

Université de Montréal

Intégration de la tarification et de l'allocation de la capacité
en transport aérien: une approche bi-niveau à grande échelle

par

Jean-Philippe Côté

Département d'informatique et de recherche opérationnelle

Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures

en vue de l'obtention du grade de

Philosophiæ Doctor (Ph.D.)

en informatique

août 2004

© Jean-Philippe Côté, 2004



RA
FL
C
C
C

AVIS

L'auteur a autorisé l'Université de Montréal à reproduire et diffuser, en totalité ou en partie, par quelque moyen que ce soit et sur quelque support que ce soit, et exclusivement à des fins non lucratives d'enseignement et de recherche, des copies de ce mémoire ou de cette thèse.

L'auteur et les coauteurs le cas échéant conservent la propriété du droit d'auteur et des droits moraux qui protègent ce document. Ni la thèse ou le mémoire, ni des extraits substantiels de ce document, ne doivent être imprimés ou autrement reproduits sans l'autorisation de l'auteur.

Afin de se conformer à la Loi canadienne sur la protection des renseignements personnels, quelques formulaires secondaires, coordonnées ou signatures intégrées au texte ont pu être enlevés de ce document. Bien que cela ait pu affecter la pagination, il n'y a aucun contenu manquant.

NOTICE

The author of this thesis or dissertation has granted a nonexclusive license allowing Université de Montréal to reproduce and publish the document, in part or in whole, and in any format, solely for noncommercial educational and research purposes.

The author and co-authors if applicable retain copyright ownership and moral rights in this document. Neither the whole thesis or dissertation, nor substantial extracts from it, may be printed or otherwise reproduced without the author's permission.

In compliance with the Canadian Privacy Act some supporting forms, contact information or signatures may have been removed from the document. While this may affect the document page count, it does not represent any loss of content from the document.

Université de Montréal
Faculté des études supérieures

Cette thèse intitulée

Intégration de la tarification et de l'allocation de la capacité
en transport aérien: une approche bi-niveau à grande échelle

présentée par

Jean-Philippe Côté

a été évaluée par un jury composé des personnes suivantes:

Michel Gendreau, président du jury

Patrice Marcotte, directeur de recherche

Gilles Savard, codirecteur de recherche

Jacques Ferland, membre du jury

Georgia Perakis, examinateur externe

Thèse acceptée le 28 janvier 2005

*If there had been a capitalist down there, the guy should
have shot down Wilbur.*

Warren Buffet

Sommaire

L'optimisation des revenus est une méthodologie utilisée dans des industries ayant une capacité de production limitée et offrant des produits ou des services de nature périssable. Bien qu'on l'applique aujourd'hui dans des domaines aussi divers que les télécommunications, l'industrie hôtelière, le transport de marchandises ou le divertissement, c'est encore dans l'industrie aérienne, où elle fit son apparition dans les années 1970, qu'on lui accorde le plus d'intérêt.

Traditionnellement, le problème d'optimisation des revenus en transport aérien est divisé en quatre sous-problèmes qui sont traités séparément par des individus, voire des départements, distincts chez les grands transporteurs aériens. Ces sous-problèmes sont :

- prévision de la demande,
- calcul des niveaux de surréservation,
- allocation de la capacité,
- tarification.

Parmi ces problèmes, l'allocation de la capacité est celui qui a reçu le plus d'attention au cours des deux dernières décennies, si bien qu'aujourd'hui la totalité des grands transporteurs internationaux utilisent des systèmes informatisés pour gérer leur inventaire de sièges. Par contre, peu d'entre eux disposent d'un système équivalent pour résoudre le problème de tarification. Cela peut s'expliquer par les difficultés à la fois techniques et théoriques entourant la résolution du problème de tarification en transport aérien.

Cette thèse explore une nouvelle approche permettant de résoudre *conjointement* et *simultanément* les problèmes d'allocation de la capacité et de tarification en transport aérien.

Nous débutons par un historique et une revue bibliographique qui illustre l'état actuel des

recherches scientifiques en optimisation des revenus et qui nous permet de mieux positionner notre approche dans le corpus existant. Nous présentons ensuite un modèle de tarification basé sur le principe de la programmation mathématique à deux niveaux. Ce modèle incorpore des aspects qui sont souvent ignorés dans la littérature scientifique, par exemple le comportement micro-économique des consommateurs, les effets de la compétition et les interactions entre les itinéraires partageant des segments de vol communs. Nous démontrons que ce modèle possède certaines propriétés qui nous permettent de déduire des règles simples décrivant la structure des solutions optimales. Nous utilisons ensuite ces règles dans un algorithme d'énumération implicite spécialisé conçu pour ce problème. Nous illustrons les performances de cet algorithme sur des données réelles provenant de l'industrie aérienne. Finalement nous présentons un cadre dynamique de tarification et d'allocation de la capacité sur un horizon divisé en périodes de réservation et proposons une stratégie de résolution globale dans le cas où le sous-problème de tarification à chaque période est résolu à l'aide de notre modèle à deux niveaux.

Mots clefs : Optimisation des revenus, tarification, programmation mathématique à deux niveaux, optimisation globale, optimisation en nombres entiers, programmation dynamique.

Abstract

Revenue optimization is a methodology used in industries characterized by a limited production capacity and providing perishable goods or services. Although it is nowadays encountered in fields as diverse as telecommunications, hospitality, freight transportation or entertainment, it is still in the airline industry, its birthplace during the 1970's, that revenue optimization receives the most interest.

Traditionnally, the airline revenue optimization problem has been divided into four sub-problems that are adressed by distinct individuals (often working in distinct departments) at carriers. These sub-problems are :

- forecasting,
- overbooking,
- capacity allocation,
- pricing.

Amongst those problems, capacity allocation (often called "Yield Management" or "Revenue Management") has received the most attention over the last two decades. Today, all major international carriers rely on computerized systems to manage their seat inventory. Few of them, however, exploit equivalent systems for solving the pricing problem. This can be explained by the many challenges, technical as well as theoretical, surrounding the pricing problem in the airline industry.

This thesis explores a new approach for *jointly* and *simultaneously* solving the capacity allocation and pricing problems in airline revenue optimization.

We begin by sketching the history of revenue optimization and present a thorough literature

review which illustrates the state of current research in this field and allows us to better position our approach within the existing corpus. We then present a pricing model based on the principles of bilevel mathematical programming. This model incorporates aspects that are often ignored in the literature, for instance micro-economic customer behaviour, competing airlines and interactions between itineraries sharing flight legs. We show that this model possesses certain properties which in turn allow us to infer simple rules describing the structure of optimal solutions. We then apply these rules to a specialized algorithm based on implicit enumeration (Branch-and-Bound) and illustrate its performance through numerical results obtained from industry data. Finally, we present a dynamic framework for pricing and capacity allocation over an horizon divided into booking periods and suggest a resolution method for the case where the pricing problem at each period is solved using our bilevel model.

Keywords Revenue optimization, pricing, bilevel mathematical programming, global optimization, integer optimization, dynamic programming.

Table des matières

Sommaire	v
Abstract	vii
Remerciements	xx
1 Introduction	1
2 Problématique	6
2.1 Définitions	6
2.2 Caractérisation et décomposition du problème	8
2.3 Nomenclature	9
2.4 Historique	10
2.5 Conditions nécessaires à l'application de l'o.d.r.	16
2.6 Méthodes de résolution	18
2.7 Caractéristiques propres à l'industrie aérienne	19
2.7.1 Nature périssable et interchangeable des produits	19

<i>TABLE DES MATIÈRES</i>	x
2.7.2 Coûts fixes élevés, coûts marginaux négligeables	20
2.7.3 Structures tarifaires complexes	23
2.8 Tendances actuelles en o.d.r.	24
2.8.1 Allocation de la capacité	24
2.8.2 Tarification	26
2.8.3 Intégration de l'allocation de la capacité et de la tarification	31
3 Revue bibliographique	32
3.1 Ouvrages généraux	32
3.2 Allocation de la capacité	33
3.2.1 Formulations par segments de vol	33
3.2.2 Formulations sur réseau	39
3.3 Tarification	44
3.4 Tarification et allocation de la capacité conjointes	45
3.5 Autres aspects	46
4 Programmation bi-niveau	49
4.1 Revue bibliographique	49
4.2 Principes de base	50
5 Modèle de tarification	56
5.1 Buts du modèle	56
5.2 Agents modélisés	57

<i>TABLE DES MATIÈRES</i>	xi
5.3 Structure de données	58
5.3.1 Vols	60
5.4 Hypothèses du modèle	62
5.5 Formulation	64
5.5.1 Notation	64
5.5.2 Objectif du premier niveau	66
5.5.3 Contraintes du premier niveau	66
5.5.4 Objectif du second niveau	67
5.5.5 Contraintes du second niveau	68
5.6 Modèle complet	69
5.7 Transformations et reformulation	70
5.7.1 Contraintes de capacité	75
5.8 Autres modèles de second niveau	78
5.9 Utilisation pratique	80
5.9.1 Contexte d'utilisation	80
5.9.2 Disponibilité des données	81
6 Méthodes de résolution	83
6.1 Jeux de données et environnement informatique	83
6.2 Résolution par un code commercial (CPLEX)	85
6.3 Structure des schémas tarifaires optimaux	86

<i>TABLE DES MATIÈRES</i>	xii
6.4 Algorithme d'énumération implicite spécialisé	92
6.4.1 Implantation	94
6.4.2 Élimination des variables binaires	96
6.4.3 Stratégie d'exploration	96
6.4.4 Sélection de la variable de branchement	101
6.4.5 Sélection du nœud courant	104
6.4.6 Comparaisons avec CPLEX	108
6.5 Résolution des relaxations linéaires par pénalité	117
6.5.1 Méthode de pénalité - cadre général	118
6.5.2 Résolution des sous-problèmes par la méthode de Frank-Wolfe	119
6.5.3 Résolution des sous-problèmes par la méthode de projection	124
6.5.4 Intégration dans le processus d'énumération implicite	130
6.6 Généralisation à plusieurs critères	138
6.6.1 Comparaisons avec CPLEX	141
7 Cadre dynamique	149
7.1 Tarification et allocation de la capacité conjoints	150
7.2 Formulation et résolution	151
7.2.1 Tarifs communs aux périodes	152
7.2.2 Tarifs propres à chaque période	156
8 Conclusion	169

<i>TABLE DES MATIÈRES</i>	xiii
Annexes	173
A Procédure d'étalonnage	174
A.1 Optimisation inverse	174
A.1.1 Revue bibliographique	175
A.2 Étalonnage du modèle BAFO	176
A.2.1 Formulation	176
A.2.2 Disponibilité des données	178
B Utilité et rationalité	180
B.1 Notion d'utilité	180
B.1.1 Définition et justification	180
B.1.2 Développement axiomatique	184
B.1.3 Utilité d'alternatives à attributs multiples	188
B.1.4 Additivité des fonctions d'utilité	188
B.2 Notion de rationalité	190
B.2.1 Description mathématique	190
B.2.2 Justification	190
C Librairie PyPlex	192
C.1 Motivation	192
C.2 Présentation de PyPlex	193
C.3 Implantation de PyPlex	196

<i>TABLE DES MATIÈRES</i>	xiv
C.4 Le module MathProg	200
C.5 Bénéfices	207
D Glossaire	209
Bibliographie	211

Liste des tableaux

2.1	Comparaison entre la mentalité traditionnelle et la mentalité axée sur la tarification optimale du produit.	28
6.1	Taille des instances utilisées lors des tests numériques.	84
6.2	Résultats obtenus avec CPLEX sur la formulation (BAFO-MIP ii).	85
6.3	Tests numériques sur les stratégies d'exploration mixtes.	99
6.4	Tests numériques sur les règles de sélection de la variable de branchement.	103
6.5	Tests numériques sur les critères de maintien du monceau.	107
6.6	Comparaisons entre CPLEX et l'algorithme spécialisé.	109
6.7	Impact du paramètre de pénalité μ sur le sous-problème primal (avec FW).	123
6.8	Impact du paramètre de pénalité μ sur le sous-problème dual (avec FW).	124
6.9	Impact du paramètre δ sur le sous-problème primal (avec FW).	124
6.10	Impact du paramètre δ sur le sous-problème dual (avec FW).	125
6.11	Impact du paramètre μ sur le sous-problème primal (avec Projection).	129
6.12	Impact du paramètre μ sur le sous-problème dual (avec Projection).	130

6.13 Impact du paramètre ω sur le sous-problème primal (avec Projection).	130
6.14 Impact du paramètre ω sur le sous-problème dual (avec Projection).	130
6.15 Performance des heuristiques primales pour trouver un point initial au problème primal.	136
6.16 Résultats obtenus avec CPLEX sur la formulation (BAFO-MIP $_{ii}$) avec deux critères. 141	
6.17 Comparaisons entre CPLEX et l'algorithme spécialisé.	142
7.1 Performances comparées de l'heuristique de gradient projeté.	162
7.2 Comparaisons entre les prix «optimaux» et les prix d'offre pour l'instance 100. . .	167
C.1 Performances comparées de certains langages.	195

Table des figures

5.1	Un arc de segment $s \in S$ reliant les nœuds (λ^1, τ^1) et (λ^2, τ^2)	59
5.2	Un arc d'escale $e \in E$ reliant les nœuds (λ^2, τ^2) et (λ^2, τ^3)	60
5.3	Un vol composé des segments s_1 et s_2 entrecoupés par l'escale e	61
6.1	Exemple des sorties obtenues avec l'implantation de l'algorithme spécialisé.	95
6.2	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 100.	110
6.3	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 101.	110
6.4	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 102.	111
6.5	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 103.	111
6.6	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 104.	112
6.7	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 105.	112
6.8	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 106.	113
6.9	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 107.	113
6.10	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 108.	114
6.11	Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 110.	114

6.12 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 111.	115
6.13 La fonction $h(\lambda) = (7 - \lambda)^+ + (5 - \lambda)^+ + (3 - \lambda)^+ + (1 - \lambda)^+$	127
6.14 Exemple de résolution de l'équation $h(\lambda) = b$	129
6.15 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 100 (avec produits).	143
6.16 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 101 (avec produits).	143
6.17 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 102 (avec produits).	144
6.18 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 103 (avec produits).	144
6.19 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 104 (avec produits).	145
6.20 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 105 (avec produits).	145
6.21 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 106 (avec produits).	146
6.22 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 107 (avec produits).	146
6.23 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 108 (avec produits).	147
6.24 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 110 (avec produits).	147
6.25 Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 111 (avec produits).	148

TABLE DES FIGURES

xix

7.1	Résultats de l'heuristique sur l'instance 100 avec 4 périodes.	163
7.2	Résultats de l'heuristique sur l'instance 100 avec 10 périodes.	163
7.3	Résultats de l'heuristique sur l'instance 103 avec 4 périodes.	164
7.4	Résultats de l'heuristique sur l'instance 103 avec 10 périodes.	164
C.1	Organisation des fichiers de la librairie PyPlex.	196
C.2	Hierarchie des classes du module MathProg.	200
C.3	Interactions logicielles.	207

Remerciements

En écrivant cette thèse, j'ai eu la chance de me joindre à une équipe de recherche dynamique, motivée et compétente et de participer à un projet de recherche des plus intéressants. Je tiens d'abord à remercier Partice Marcotte et Gilles Savard, mes directeurs de recherche, pour leur intérêt, leur confiance, leur disponibilité et la façon dont ils dirigent l'équipe de recherche. L'ambiance de travail à la fois amicale et professionnelle qu'ils ont créée encourage l'initiative et l'innovation. J'espère que notre collaboration se poursuivra pour de nombreuses (et profitables) années. Je veux aussi remercier les autres membres de l'équipe de recherche, en particulier mon «colocataire» Alexandre Schoeb avec qui j'ai eu de nombreuses discussions productives sur certains aspects de cette thèse. Merci aussi à Jean-François Pagé et Julie Bédard d'Air Canada. Leur disponibilité et leur expertise m'auront permis de mieux comprendre des réalités de l'industrie aérienne.

Le Centre de Recherche sur les Transports, où j'ai réalisé la majeure partie de ces travaux, offre aux chercheurs et aux étudiants un environnement idéal pour poursuivre leurs recherches. Merci au personnel administratif pour s'occuper de toutes ces petites choses que les chercheurs ont souvent tendance à oublier. Merci aussi aux administrateurs du système informatique pour maintenir hub, hub2 et leurs spokes dans un état optimal malgré les abus que nous leur faisons subir.

Finalement, et par-dessus tout, je veux remercier mes parents pour leur support indéfectible et inconditionnel tout au long de mes études, du premier jour de la maternelle jusqu'à aujourd'hui. Ils ne comprendront peut-être pas tous les étranges symboles mathématiques que renferme ce document, mais ils y sont pour beaucoup. Merci aussi à ma sœur Marie-Christine. Ses constants rappels du fait qu'à l'âge de 27 ans je fréquente toujours l'école m'ont motivé à redoubler d'efforts et à garder les idées claires pour ne pas trop m'écarter des objectifs que je m'étais fixés.

Chapitre 1

Introduction

L'aviation civile est une affaire de gros sous.

Pour les passagers (et très souvent leurs employeurs), l'achat d'un billet d'avion demeure une dépense considérable qui implique une réflexion préalable. On achète rarement un billet d'avion sur un coup de tête, et bien qu'on ait observé ces dernières années une augmentation du nombre de réservations effectuées par l'entremise de services en-ligne, plusieurs consommateurs font encore appel à un agent professionnel pour les aider à se retrouver parmi la pléthore de tarifs offerts, de conditions, de restrictions et de programmes de fidélisation.

Pour les transporteurs (et leurs actionnaires et, souvent depuis quelques années, leurs crédateurs), les sommes en jeu sont beaucoup plus importantes. L'industrie du transport aérien possède un «coût d'entrée» extrêmement élevé. Les plus récents aéronefs se vendent des dizaines, voire des centaines de millions de dollars¹, le prix du carburant fluctue généralement vers le haut, les aéroports surachalandés haussent leurs droits d'atterrissage, sans parler des coûts d'assurance et de sécurité. En dépit de ces coûts fixes très élevés, les coûts marginaux que doit assumer un transporteur lors de la vente d'un siège supplémentaire sont pratiquement négligeables. Ce phénomène, combiné à la nature périssable du service qu'ils offrent, motive les transporteurs à exploiter au maximum chaque unité de capacité disponible; la différence entre profit et perte réside souvent dans la vente (habituellement à fort prix) des quelques derniers sièges disponibles

¹L'Airbus A380, qui sera dans quelques années le plus gros et le plus moderne de tous les avions civils, se détaille un quart de milliard de dollars américains l'unité.

sur un vol.

Jusqu'à la fin des années 1970, les tarifs pour le transport aérien étaient fixés par des agences gouvernementales (et le sont encore pour certains marchés internationaux). Ayant les mains liées par les structures tarifaires peu flexibles qui leur étaient imposées, les transporteurs étaient plus ou moins impuissants et devaient se soumettre aux tendances du marché. Cependant, depuis la vague de déréglementation des années 1970-80, les transporteurs disposent de plus de latitude pour maximiser leurs revenus. C'est dans ce contexte qu'est apparue la méthodologie que nous connaissons aujourd'hui sous le nom d'optimisation des revenus¹.

Encore aujourd'hui, l'aspect central de cette méthodologie demeure l'allocation de la capacité. En divisant (de façon imaginaire) la cabine d'un appareil en classes tarifaires correspondant habituellement (mais pas nécessairement) à des niveaux de prix distincts, les transporteurs tentent de contrôler l'accès aux sièges offerts à rabais et destinés à une clientèle élastique afin que ceux-ci ne soient pas disponibles pour les passagers plus inélastiques et affichant une propension à payer supérieure. À l'aide de restrictions artificielles, les transporteurs arrivent à segmenter leur clientèle et à augmenter leurs revenus en fermant les classes tarifaires les moins profitables au fur et à mesure que le nombre de réservations augmente.

On remarquera que pour la très grande majorité des transporteurs la tarification est un élément complètement séparé du processus d'allocation de la capacité. Les méthodes de gestion de l'inventaire de sièges qui sont utilisées en pratique considèrent les tarifs comme une donnée exogène, qu'il s'agisse de moyenne de tarifs historiques ou encore de tarifs actuellement en vigueur. Quant à la tarification elle-même, elle s'effectue aussi en vase clos, sans tenir compte des limites de réservation imposées par les contrôleurs. De plus, la détermination des tarifs demeure un processus généralement manuel, effectué par un analyste humain qui base ses décisions sur son expérience et son intuition, sans avoir recours à une aide informatisée.

Cette dichotomie entre les processus d'allocation de la capacité et de tarification (et entre les individus qui en sont respectivement responsables) donne parfois lieu à des situations absurdes qui ont un impact négatif sur les revenus de l'entreprise. Par exemple, lorsque la situation sur le marché évolue rapidement, on peut observer des «inversions» de classes. Un produit voit son tarif augmenté (diminué) jusqu'à un niveau qui le placerait dans la classe tarifaire suivante (précédente), de sorte que les schémas d'allocation ne sont plus valides et doivent être réévalués.

¹Le chapitre 2 contient un historique beaucoup plus détaillé de l'optimisation de revenus en transport aérien.

Les observateurs de l'industrie aérienne et les transporteurs eux-mêmes ont reconnu les bénéfices que pourraient avoir le développement d'outils de tarification informatisés et leur intégration dans le processus d'allocation de la capacité. Des obstacles se dressent cependant sur cette voie. La tarification est un processus complexe qui pose plusieurs défis. D'abord, d'un point de vue technique, la collecte, le formatage et l'interprétation de données tarifaires (qu'elles soient courantes ou historiques) demandent un investissement considérable. Du point de vue théorique, il n'y a pas de consensus dans la communauté scientifique sur la façon d'aborder et de modéliser le problème de tarification en transport aérien, et peu importe l'approche retenue, la taille et la complexité des problèmes (comportement des passagers, importance de la compétition, structure de réseau, alliances commerciales, etc.) nécessitent le développement d'algorithmes spécialisés et sophistiqués. Les aspects organisationnels et humains entrent aussi en ligne de compte car la fusion des processus de tarification et d'allocation de la capacité implique certainement une révision de pratiques bien ancrées dans l'industrie depuis une quinzaine d'années. Par conséquent, les systèmes intégrés de tarification et d'allocation de la capacité sont encore à l'état embryonnaire dans l'industrie du transport aérien¹. Les bénéfices potentiels sont par contre trop importants pour que l'industrie n'investisse pas dans le développement et le déploiement de tels systèmes.

Cette thèse propose une nouvelle approche pour la résolution conjointe et simultanée des problèmes de tarification et d'allocation de la capacité en transport aérien. L'élément central de notre approche est un modèle de tarification basé sur le principe de la programmation mathématique à deux niveaux. Le premier niveau de décision est occupé par le transporteur, lequel cherche à maximiser les revenus sur l'ensemble des routes qu'il opère. Le second niveau correspond aux passagers qui cherchent à minimiser la désutilité qu'entraîne l'achat d'un titre de transport en considérant l'intégralité de l'offre, y compris les compétiteurs. Ce modèle de tarification s'intègre dans un cadre dynamique d'allocation de la capacité, permettant ainsi la résolution conjointe des deux problèmes.

¹Lors de la récente conférence annuelle du groupe d'étude en optimisation des revenus de la société AGIFORS, qui s'est tenue du 28 au 31 mars 2004 à Auckland en Nouvelle-Zélande, les représentants de la société Delta (le troisième plus important transporteur du monde) ont offert une présentation sur un projet interne visant l'implantation d'un système d'allocation de la capacité au niveau du réseau (alors que cette allocation s'effectue présentement au niveau du segment de vol, ce qui ignore les interactions entre les segments et leur contributions aux revenus du réseau). Le sommaire de leur présentation indique que le déploiement de ce projet est prévu pour le premier semestre de 2004, ce qui laisse présager que l'implantation d'un système intégré de tarification et d'allocation de la capacité ne se produira pas à court terme. Il s'agit plutôt d'un des défis auquel feront face les transporteurs au cours des prochaines décennies.

La thèse est organisée comme suit. Le chapitre 2 présente de façon détaillée la problématique à laquelle nous nous intéressons. On y retrouve un historique de l'optimisation des revenus en transport aérien, une description des conditions nécessaires à son application et un aperçu des tendances actuelles dans ce domaine. Le chapitre 3 contient une revue bibliographique approfondie illustrant les solutions proposées jusqu'à maintenant aux différents aspects du problème d'optimisation des revenus.

Le chapitre 4 présente la programmation mathématique à deux niveaux, l'approche sur laquelle est basée notre modèle de tarification. Nous y faisons une revue bibliographique et en présentons les principes généraux. Le chapitre 5 décrit le modèle et la structure de données qui lui est sous-jacente.

Le chapitre 6, le plus important de la thèse, présente certaines propriétés du modèle et démontre qu'on peut en déduire des règles lexicographiques décrivant la structure des solutions optimales. Ces règles sont à leur tour utilisées dans un algorithme spécialisé d'énumération implicite conçu spécialement pour résoudre le modèle. Nous décrivons cet algorithme et présentons des résultats numériques obtenus en l'utilisant sur des données réelles provenant de l'industrie. Nous comparons aussi ses performances avec celles d'un solveur commercial.

Le chapitre 7 présente un cadre dynamique de tarification et d'allocation de la capacité dans lequel notre modèle de tarification joue le rôle de sous-routine. Nous illustrons les propriétés de cette approche dynamique et proposons une méthode de résolution heuristique de type (sous-)gradient projeté. Les performances de cette méthode sont encore une fois illustrées en utilisant des données réelles.

Le chapitre 8 est la conclusion de la thèse. En annexe A, on retrouve une brève description de la procédure d'étalonnage des paramètres du modèle. Nous faisons référence à cette procédure à quelques reprises au long du texte, mais comme son développement n'est pas directement lié aux travaux entourant la rédaction de cette thèse, nous avons choisi de la présenter en annexe plutôt que dans le corps de la thèse. Nous nous intéressons en annexe B aux concepts d'utilité et de rationalité, lesquels jouent un rôle important dans notre modèle. En allant notamment puiser dans des ouvrages de théorie des jeux et d'analyse micro-économique, nous justifions certaines hypothèses sur lesquelles s'appuie notre approche. L'annexe C contient une courte introduction à la librairie PyPlex, un outil logiciel développé dans le cadre de cette thèse pour faciliter l'implantation et la validation de modèles mathématiques et d'algorithmes. Finalement,

l'annexe D contient un glossaire de termes propres au domaine de l'optimisation des revenus. Une liste de références bibliographiques clos la thèse.

Nous terminons cette introduction par quelques remarques sur la notation utilisée dans ce document. Les mots apparaissant en caractères **gras** dans le texte sont des termes importants définis pour la première fois, tandis que les expressions apparaissant en *gras italique* correspondent à une rubrique du glossaire. Dans la présentation de modèles de programmation mathématique, les variables primales sont représentées par des lettres latines minuscules et les variables duales par des lettres grecques minuscules. Les paramètres de ces modèles (issus de la procédure d'éta-lonnage) apparaissent en gras (e.g. α plutôt que α). Les lettres majuscules (e.g. A , F , Γ , etc.) représentent des matrices ou des ensembles (et dans certaines cas spéciaux, des vecteurs). Les attributs de produits tarifaires sont représentés par des lettres en caractères calligraphiques (e.g. \mathcal{A} , \mathcal{D} , etc.). Nous utiliserons les symboles \mathbb{R} pour dénoter l'ensemble des nombres réels, \mathbb{R}_{0+} pour les nombres réels non négatifs, \mathbb{Z} pour les entiers, \mathbb{Z}_+ pour les entiers positifs (i.e. exclu-ant 0) et \mathbb{B} pour les nombres «binaires», i.e. $\mathbb{B} = \{0, 1\}$. De plus, afin d'alléger la notation et d'éviter l'emploi de l'opérateur de transposition, on suppose que les vecteurs de gauche sont des vecteurs-ligne et ceux de droite des vecteurs-colonne.

Chapitre 2

Problématique

Ce chapitre introduit la problématique. Nous y définissons le problème général et proposons une décomposition de ce problème en sous-problèmes interreliés. Nous présentons aussi certaines caractéristiques propres à l'industrie du transport aérien qui justifient l'utilisation du modèle proposé.

2.1 Définitions

Yield Management, Revenue Management, Perishable Asset Revenue Management, Enterprise Profit Optimization sont autant d'expressions désignant un ensemble de techniques ayant fait leur apparition dans l'industrie aérienne américaine au cours des années 70. Pour des raisons que nous invoquerons plus loin, nous appellerons l'ensemble de ces techniques «optimisation des revenus», ou o.d.r. Comme la majorité des nouveaux processus d'affaire, cette méthodologie a été développée afin de répondre à des changements à la fois internes et externes à l'industrie du transport aérien. Ces changements significatifs dans le comportement des transporteurs américains furent principalement engendrés par un événement précis : la déréglementation de l'industrie aérienne aux États-Unis, qui s'est échelonnée de 1978 à 1983 [Cro95, Cro98, MvR99].

Bien que faisant l'objet d'une littérature scientifique et technique relativement abondante, aucune définition exacte de cette méthodologie ne s'est imposée au fil des années. Selon une définition informelle, presque folklorique, il s'agit de «la pratique qui consiste à vendre le bon

produit au bon client au bon moment et au bon prix» [SLD92]. Kimes [Kim90] la décrit simplement comme «une méthode pour gérer la capacité de façon profitable». Cross [Cro95, Cro98] la définit comme «un processus qui consiste à prévoir le comportement futur des consommateurs d'un point de vue micro-économique pour ensuite optimiser la disponibilité des produits et leur prix dans le but de maximiser les revenus». Cet auteur ajoute que cette discipline fait intervenir à la fois des analystes humains et des systèmes informatiques sophistiqués dans un ensemble de processus d'affaire (*business processes*) ayant comme objectif de comprendre le marché, d'anticiper le comportement des consommateurs et de réagir rapidement lorsque de nouvelles opportunités de revenu se présentent. Cette dernière définition nous paraît tout à fait appropriée et nous l'adopterons pour décrire la méthodologie que nous appelons optimisation des revenus. Remarquons cependant que certains perçoivent l'o.d.r. d'une toute autre façon. C'est le cas de l'hebdomadaire *The Economist*, qui dans un article récent décrivait (non sans sarcasme) l'o.d.r. comme «[...] a rich science delicately engineered by people with large brains and fancy computer programs to squeeze every last cent from a plane load of passengers» [Ano02]. Cet article faisait remarquer que, du moins du point de vue du consommateur, l'o.d.r. n'est pas nécessairement un bienfait mais plutôt un mal nécessaire.

Peu importe la définition, le problème fondamental auquel ont à faire face les transporteurs aériens d'aujourd'hui demeure le même : dans un environnement déréglementé et hautement compétitif, les transporteurs doivent assumer des coûts fixes très élevés afin d'offrir une quantité limitée de produits (i.e. sièges sur un vol), lesquels sont, du moins aux yeux des consommateurs, plus ou moins équivalents et interchangeable d'un transporteur à l'autre^{1 2}. Afin d'être profitable, il est essentiel que les transporteurs gèrent avec soin les produits qu'ils offrent et déterminent de façon réfléchie les prix auxquels ces produits sont offerts, dans le but de tirer le maximum de revenus de chaque siège au moment du départ.

Bien qu'elle fit son apparition dans l'industrie aérienne et qu'elle eût été développée principalement par des spécialistes de ce domaine, l'o.d.r. est aujourd'hui utilisée dans d'autres industries telles l'hôtellerie [YI97], les télécommunications [Hum01], le transport maritime, le rail [CILP99],

¹Les compagnies aériennes ont depuis quelques années beaucoup investi afin d'atténuer les effets de l'«interchangeabilité» de leurs produits, notamment par des stratégies de *branding* agressives leur permettant de se démarquer de leurs concurrents et par la mise en place de programmes de fidélisation. Malgré tous ces efforts, des sièges sur un vol entre deux villes demeurent, pour la plupart des consommateurs, des produits équivalents, peu importe qu'ils soient offerts par un transporteur ou un autre.

²Nous discuterons plus en détail des caractéristiques particulières de l'industrie aérienne dans une section subséquente.

la location de voitures, etc. Nous verrons un peu plus bas que ces industries à première vue diverses partagent avec le transport aérien des caractéristiques communes les rendant propices à l'application de l'o.d.r.

2.2 Caractérisation et décomposition du problème

Afin de caractériser l'o.d.r. en transport aérien, Weatherford et Bodily [WB92] adoptent un point de vue inspiré de la programmation mathématique et la décrivent comme la combinaison d'objectifs posés par une compagnie, lesquels sont associés à des groupes de contraintes, le tout formant un programme mathématique.

Les objectifs plausibles sont :

1. maximisation du profit
2. maximisation des revenus¹
3. maximisation de l'utilisation de la capacité
4. maximisation du revenu moyen par usager
5. minimisation de la perte de sympathie des clients envers la compagnie
6. maximisation de l'utilité des clients.

Les contraintes possibles sont :

1. contraintes opérationnelles : capacité des appareils, intégrité des horaires et des routes
2. contraintes de marketing : niveau de service minimal
3. contraintes stratégiques : aspirations de l'entreprise, par exemple devenir le meneur en termes de prix sur un certain marché
4. contraintes découlant des actions de concurrents : les prix, horaires et routes des concurrents peuvent aussi imposer des contraintes supplémentaires.

Selon Smith, Leimkuhler et Darrow [SLD92], cette approche «unifiée» est inapplicable en pratique, car elle engendre des problèmes d'optimisation dont la taille dépasse largement les capacités technologiques actuelles. L'expérience acquise chez American Airlines par ces auteurs montre qu'il est nécessaire de décomposer le problème d'o.d.r. en transport aérien en sous-problèmes

¹Dans les cas où les coûts sont plus ou moins fixes, la maximisation des revenus est équivalente à la maximisation du profit.

pouvant être traités séquentiellement. Kimes [Kim90] adopte aussi cette approche et propose la décomposition suivante :

1. Prévision de la demande

Analyse du comportement de la clientèle à partir de données historiques fournies par le système de réservations informatisé dans le but de prévoir la demande pour les produits futurs.

2. Détermination de la politique de surréservation

Prédiction, pour chaque vol, du nombre de passagers *aphanes*, de façon à ce que la politique de *surréservation* sur ce vol minimise le nombre de passagers *emphanes* auxquels on aura refusé l'accès à bord, tout en s'assurant qu'un maximum de sièges seront occupés au décollage.

3. Allocation de la capacité

Détermination du nombre de sièges pouvant être réservé dans chaque *classe tarifaire* ainsi que la période pendant laquelle ces classes tarifaires sont disponibles.

4. Tarification

Détermination du prix des *bases tarifaires* offertes.

C'est cette décomposition, généralement acceptée dans la littérature, que nous utiliserons.

2.3 Nomenclature

Il existe une certaine confusion dans la littérature scientifique et technique quant à l'appellation du problème et des solutions proposées. Le terme le plus commun est sans doute *Yield Management*. Il s'agit du terme inventé à la fin des années 70 par Robert Crandall, alors président de American Airlines, compagnie où cette technique fit son apparition [Cro98]. À l'origine, ce terme ne désignait que le troisième des sous-problèmes énumérés ci-dessus, soit l'allocation de la capacité, car il s'agit du premier aspect auquel se sont intéressé les compagnies aériennes¹. Avec les années, cette expression en est venue à désigner l'ensemble des activités de gestion de la demande et de l'inventaire et de tarification chez les transporteurs aériens. Cependant, certains

¹Certains transporteurs avaient commencé à s'intéresser au problème de surréservation dès les années 60, mais rarement de façon systématique, en affectant des employés spécialement à cette tâche. Ce n'est que vers le début des années 80 que de nouveaux départements de *Yield Management* et de *Revenue Management* regroupant à la fois des spécialistes en surréservation et en contrôle d'inventaire furent créés chez les grands transporteurs américains.

auteurs (et firmes offrant des systèmes informatiques dans le domaine) utilisent encore le terme *Yield Management* pour désigner exclusivement le contrôle d'inventaire. D'autres auteurs ont suggéré le terme *Revenue Management*, soit pour désigner la gestion de l'inventaire de sièges¹, soit pour désigner la pratique qui englobe à la fois le *Yield Management* original, i.e. le contrôle d'inventaire, et le traitement des autres sous-problèmes. C'est ce terme, francisé en «Gestion des Revenus», qui fut utilisé dans [Côt01]. Toutefois, ce terme suggère que la technique en question est un processus passif, ou au mieux réactif, quant aux facteurs dont dépendent les revenus d'une compagnie. En réalité, il s'agit d'un processus pro-actif visant à comprendre le comportement des consommateurs et à influencer leurs actions en contrôlant les prix et la quantité des produits disponibles. Nous avons donc choisi d'adopter le terme «optimisation des revenus», ou o.d.r., car celui-ci suggère une façon de faire délibérée, réfléchie et volontariste en rapport avec les activités générant des revenus pour une compagnie.

2.4 Historique

Les principes fondamentaux sur lesquels sont basées les méthodes modernes d'o.d.r. sont étudiés depuis longtemps en sciences économiques. Déjà en 1776, l'économiste écossais Adam Smith remarquait dans son ouvrage *The Wealth of Nations* que les consommateurs et les producteurs sont en quête d'information leur permettant de faire des choix économiques reflétant leurs propres intérêts en employant les ressources rares de façon productive. On peut voir dans cet énoncé les bases des études subséquentes sur le comportement micro-économique des consommateurs et des producteurs.

Les facteurs généraux auxquels sont soumis les transporteurs aériens modernes sont aussi connus depuis que l'être humain s'adonne à des échanges économiques. La force des consommateurs, la demande versus l'offre, l'existence de choix à meilleur marché, les variations cycliques et saisonnières, sont autant de phénomènes qui ont caractérisé la plupart des activités économiques au cours de l'histoire. Ces facteurs influencent l'industrie aérienne depuis ses débuts. Ce fut d'ailleurs le cas lors du premier vol commercial de l'histoire, qui eut lieu le premier janvier 1914

¹Il existe une différence technique entre le *Yield Management* et le *Revenue Management* dans le contexte de la gestion de l'inventaire de siège et de l'allocation de la capacité aux classes tarifaires d'un appareil. La première approche cherche à maximiser le rendement, i.e. le revenu qu'engendre chaque siège divisé par la distance parcourue, alors que la seconde cherche à mesurer le revenu comme mesure absolue. Toutefois, peu d'articles font cette différence si bien que les deux termes sont généralement interchangeables dans la littérature spécialisée.

[Cro95, Leg91]. Afin de rentabiliser l'achat d'un aéronef, le pilote américain Tony Jannus offrit aux intéressés la possibilité de traverser la baie de Tampa (en Floride) entre les villes de Tampa Bay et St-Petersburg à bord de son appareil, un trajet de 29km. Jannus avait fixé le prix du billet à 5\$, mais il fut en mesure d'expérimenter une des premières réalités de l'o.d.r. : la valeur d'un billet n'est pas la même aux yeux de tous les passagers. En effet, l'ex-maire de St-Petersburg, A.C. Pheil, offrit à Jannus 400\$ pour le privilège d'acheter le premier billet et ainsi devenir le premier passager commercial de l'histoire de l'aviation. Bien que le produit eût été le même, M. Pheil paya 80 fois plus que les autres passagers. Il s'agit du premier exemple de segmentation de la clientèle en transport aérien. Jannus devait aussi expérimenter une autre réalité propre à l'o.d.r. : la saisonnalité. Après quatre mois d'opérations profitables, il dut abandonner son service en avril, à la fin de l'hiver, par manque de clientèle.

L'aviation commerciale a véritablement pris son essor après la première guerre mondiale. Les années 20 virent l'apparition de nombreux transporteurs nationaux, dont certains existent encore aujourd'hui. À cette époque, le transport postal était la principale source de revenus des compagnies aériennes, les passagers n'étant acceptés que lorsque les sacs de courrier ne remplissaient pas complètement la soute. Les transporteurs de cette époque avaient donc compris une autre réalité propre à l'industrie aérienne : lorsqu'un avion décolle, un espace non occupé dans cet avion peut être produit à coût pratiquement nul. Mieux vaut donc l'offrir à coût réduit, car ce coût se traduit presque entièrement en un profit qui autrement n'aurait pas été généré.

On assista pendant les années 30 à l'apparition de nouveaux avions plus confortables et ayant un plus grand rayon d'action, ce qui stimula grandement le transport de passagers. Dans le but de mieux circonscrire cette industrie grandissante, le président F.D. Roosevelt ratifia le 23 juin 1938 le *Civil Aeronautics Act*. Ce texte de loi entraîna la création du *Civil Aeronautics Board* (C.A.B.), dont la mission était de préserver un équilibre compétitif en ce qui a trait aux routes, aux capacités offertes ainsi qu'aux tarifs exigés par les transporteurs aériens aux États-Unis. Le C.A.B. établit entre autres des normes de rendement au mille que les transporteurs devaient respecter. Tout nouveau tarif devait être approuvé par le C.A.B.

Jusqu'à la fin des années 40, les transporteurs américains n'exploitaient qu'un seul type de cabine sur tous leurs appareils et n'offraient qu'une seule classe tarifaire, la «première». En 1948 un transporteur nommé Capitol Airlines offrit un nouveau type de cabine et de classe tarifaire, la classe *coach*, sur quelques-unes de ses lignes. Capitol réussit à convaincre le C.A.B. de lui accorder un prix plancher plus bas en offrant son service *coach* sur des appareils d'avant-

guerre, donc moins confortables, tels des Boeing B-247 et des Douglas DC-3. Le C.A.B. accorda à Capitol un prix plancher de 4ç par mille, comparativement à 6ç par mille pour la première classe. L'apparition de ce nouveau type de produit stimula la demande et les autres transporteurs emboîtèrent rapidement le pas à Capitol en offrant eux-mêmes une classe *coach*. En fait, la demande fut tellement grande que les transporteurs retirèrent quelques mois plus tard leurs vieux appareils, même si ceux-ci avaient été utilisés pour justifier la classe *coach* au C.A.B, et les remplacèrent par des Douglas DC-6 et des Lockheed Constellation plus modernes et surtout plus gros.

La prochaine vague de nouveaux tarifs eut lieu vers la fin des années soixante, parallèlement au développement des premiers systèmes de réservation informatisés. Les transporteurs américains reçurent du C.A.B. l'autorisation d'offrir des tarifs réduits du type *excursion fares*. Il s'agissait de billets qui devaient être réservés de 7 à 21 jours à l'avance et auxquels étaient associées d'autres restrictions quant à la durée du séjour entre l'aller et le retour. Il n'y avait toutefois pas de contrôle sur la quantité de billets de ce type qui étaient offerts ; s'ils remplissaient les exigences, tous les passagers d'un vol pouvaient, en théorie, acheter des billets *excursion fare*.

À l'extérieur des États-Unis, un des pionniers dans ce domaine fut BOAC (British Overseas Airways Corporation, aujourd'hui British Airways), qui dès 1971 offrait des tarifs *Earlybirds* sur les billets réservés au moins 21 jours à l'avance. Cette innovation permettait à la compagnie de tirer des revenus de certains sièges qui autrement eussent demeuré inoccupés au décollage. Elle posait cependant le problème du calcul de la quantité de sièges qui devaient être protégés pour les passagers les plus payants achetant leurs billets plus tard. Littlewood, alors à l'emploi de BOAC, proposa un ensemble de règles permettant un contrôle plus strict des inventaires de sièges en présence de deux classes tarifaires [Lit72]. Ces règles utilisaient l'information statistique historique sur les habitudes de réservation des passagers de BOAC, laquelle était désormais disponible grâce au développement et à la mise en place de systèmes de réservation informatisés à peu près à la même époque.

La demande pour des billets à prix réduit augmenta significativement au début des années 70, si bien que le C.A.B. se vit dans l'obligation d'accorder des permis d'un nouveau type appelé *public charter*. Les transporteurs détenant ce type de permis pouvaient offrir des tarifs réduits, sans toutefois offrir le même niveau de service au sol et en vol que les transporteurs traditionnels. Ces nouveaux joueurs entraient néanmoins en compétition directe avec les transporteurs établis sur certains marchés importants tels New York–Los Angeles, ce qui n'était pas sans inquiéter

les dirigeants de ces derniers. C'est au cours d'une réunion chez American Airlines (AA) dont le but était de trouver des solutions pour contrer cette nouvelle menace que l'idée du *Yield Management* fut évoquée pour la première fois. En se demandant comment ils pouvaient offrir des sièges à un coût de production aussi bas que celui des *public charters*, les dirigeants de AA se rendirent compte qu'ils étaient en mesure de proposer des sièges à un coût pratiquement nul pour la compagnie. En effet, le taux d'occupation des vols de AA était à cette époque d'environ 70%. Or, le coût marginal d'un siège autrement inoccupé est pratiquement nul (on doit en réalité calculer quelques dizaines de litres de carburant, un repas et la manutention de quelques bagages supplémentaires). AA pouvait ainsi offrir ce 30% de sièges normalement invendus à un prix égal, voire inférieur, à celui des *public charters*. Le problème était de s'assurer que ces sièges ne fussent pas achetés par des passagers possédant la capacité et la volonté d'acheter des sièges plus dispendieux. La réponse de AA fut donc de proposer des billets de type *Super Savers*, auxquels étaient associées de fortes restrictions. La stratégie fonctionna puisque les *public charters* disparurent du marché dans les années qui suivirent. L'effort ne fut cependant pas vain puisqu'il préparait AA au plus grand changement de l'histoire de l'aviation commerciale aux États-Unis, soit la déréglementation complète du transport aérien qui s'échelonna de 1978 à 1983.

En Europe, les premiers marchés à être déréglementés furent ceux entre la Grande-Bretagne et l'Irlande et entre la Grande-Bretagne et les Pays-Bas. Déjà à la fin des années 80, Aer Lingus, British Airways et KLM avaient développé des systèmes sophistiqués de contrôle d'inventaire. Ces compagnies furent imitées par le majorité des transporteurs européens majeurs au début des années 90.

People Express : victime de l'o.d.r.

People Express (PE) fit partie de la première vague de transporteurs bon marché qui suivirent la déréglementation. La compagnie, fondée en 1981, offrait un minimum de service à ses passagers, ce qui lui permettait de vendre ses billets à un prix allant de 50 à 70% du prix des billets offerts par les transporteurs traditionnels.

Dans le but d'économiser lors de la création de l'entreprise, les dirigeants avaient opté pour une structure tarifaire simpliste : un seul type de billet était offert à prix unique (bien entendu, les appareils de PE ne comportaient pas de cabine d'affaire ou de type première). Entre autres, le système de réservation informatique de la compagnie ne pouvait accommoder qu'un seul prix

pour chaque marché. Cette simplicité dans la stratégie de mise en marché fut d'ailleurs utilisée par PE comme argument publicitaire : les annonces de la compagnie ridiculisaient les structures tarifaires extrêmement complexes des transporteurs traditionnels et vantaient la simplicité et l'efficacité du prix unique. Initialement, PE occupait des marchés délaissés par les transporteurs établis, comme par exemple Newark (New Jersey) vers Norfolk (Virginie) ou Columbus (Ohio), si bien qu'elle n'entraît pas en compétition avec les autres compagnies aériennes mais plutôt avec des modes de transport alternatifs tels l'autocar et l'automobile.

Les grands transporteurs estimaient que PE n'attirait que des passagers qui n'auraient de toute façon jamais voyagé sur un de leurs vols. Cette perception changea brusquement en 1984 lorsque PE s'attaqua à certains marchés importants tels New York-Chicago, en siphonnant une partie significative de la clientèle des transporteurs traditionnels. La réponse de ces derniers, avec AA en tête, fut d'appliquer une stratégie agressive d'o.d.r. AA offrit par exemple des billets de type *Ultimate Super Savers*, qui étaient souvent moins chers que ceux de PE. Ces billets étaient cependant accompagnés de restrictions très sévères, de sorte que seuls les passagers les plus élastiques s'en procuraient. Or ces passagers représentaient précisément la clientèle visée par PE. Celle-ci, ne disposant ni de système d'o.d.r. ni de système de réservation lui permettant de différencier ses produits, se vit dans l'obligation d'égaliser les prix de la compétition sur *tous* ses sièges et *tous* ses vols. Cependant, même à prix égal, les passagers préféraient aller chez les transporteurs traditionnels car ceux-ci offraient différents services (repas en vol, sélection préalable de siège, manutention de bagages) absents chez PE.

L'impact de la stratégie menée par AA et quelques autres transporteurs majeurs fut dévastatrice pour PE : le taux d'occupation sur ses vols passa de 70-90% en 1985 à 20-40% au début de 1987. PE déclara faillite le 2 février 1987 [Leg91] et fut absorbée par Continental Airlines. Il s'agit de la première victime importante de l'o.d.r. de l'ère post-déréglementation aux États-Unis.

Les limites de l'o.d.r. : le catastrophique été 92

Les événements ayant eu lieu dans l'industrie aérienne américaine au printemps et à l'été 92 illustrent les limites des stratégies d'o.d.r. dans un marché instable. C'est encore une fois American Airlines (AA) et son président de l'époque Robert Crandall qui ont été tenus responsables de la guerre de prix ayant secoué l'industrie pendant cette période et des pertes considérables encourues subséquemment par les transporteurs [ZRS92, Cro98] .

Au début des années 90, malgré un certain ralentissement économique, AA était en pleine crois-

sance. La compagnie avait au cours de années précédentes investi plusieurs milliards dans l'acquisition de nouveaux appareils et presque doublé son nombre d'employés. Elle avait aussi racheté plusieurs routes internationales d'autres transporteurs, notamment vers l'Amérique Latine et l'Europe, ce qui en faisait un transporteur d'envergure internationale. S'appuyant sur des stratégies d'o.d.r. sophistiquées qu'elle avait elle-même mises au point et avait été la première à utiliser, AA était considérée comme le meneur en termes de tarifs sur le marché américain, les autres transporteurs devant la plupart du temps suivre les tendances lancées par AA¹.

Le 9 avril 1992, AA lança une nouvelle structure tarifaire appelée *Value Pricing*. Ce nouveau programme ramenait à quatre le nombre de produits tarifaires disponibles sur chaque marché (alors que ce nombre pouvait auparavant dépasser la vingtaine sur certains marchés) et proposait des réductions spectaculaires, par exemple 38% en classe affaire. AA s'appuya sur la simplicité de sa nouvelle structure tarifaire et la stimulation de la demande entraînée par les réductions de tarifs pour justifier son nouveau programme. D'autres transporteurs, comme TWA et America West, qui étaient déjà en difficulté financière, ont rapidement dénoncé le nouveau programme de AA, prétendant qu'il s'agissait d'une stratégie de tarification prédatrice visant à éliminer les concurrents les plus faibles pour ensuite rehausser les prix. Ces compagnies s'appuyaient sur les déclarations de AA qui prétendait ne pas s'attendre à ce que son nouveau programme génère des revenus supplémentaires à court terme. Cela n'empêcha toutefois pas ses concurrents d'égaliser rapidement ses tarifs et de déclencher une guerre de prix qui dura tout l'été 92. L'industrie aérienne américaine avait tout simplement perdu le contrôle. On assista alors à des promotions des plus imaginatives («deux billets pour le prix d'un» ou encore «*Adults Fly for Free*»), à une diminution moyenne du prix des billets de l'ordre de 50% par rapport à l'année précédente et une augmentation spectaculaire du nombre de voyageurs. Les grands gagnants furent les consommateurs et l'industrie touristique [MB92].

Paradoxalement, c'est en appliquant une stratégie d'o.d.r. que Continental, l'un des transporteurs engagés dans cette guerre de prix, put afficher une meilleure performance que les autres. Ayant remarqué que les réservations s'effectuaient beaucoup plus rapidement que prévu et que la guerre de prix avait engendré une demande supplémentaire, Continental décida de protéger un plus grand nombre de sièges qu'elle fut en mesure de vendre à prix plus élevé lorsque les vols de la compétition eurent tous été remplis.

¹À cet effet, Edward Bauvais, le président d'America West à cette époque, déclarait : «*What we have here is de facto regulation—with American as the regulator. They set the price and punish anyone who deviates*». [ZRS92]

La guerre de prix et ses conséquences amenèrent les transporteurs américains à revoir leurs stratégies d'o.d.r. et de marketing. Certains décidèrent de fusionner leur département de tarification avec leur département de contrôle d'inventaire (ces deux entités étant auparavant indépendantes chez la majorité des transporteurs et le sont encore aujourd'hui chez certains), ce qui permet de mieux coordonner les tarifs et l'offre de sièges.

Après quelques années marquées par des pertes colossales, estimées à 13 milliards US\$ aux États-Unis seulement pour la période 1990-1994, la plupart des transporteurs américains ont renoué avec la rentabilité en 1995, avec semble-t-il un certain changement de mentalité. John Dashburg, le président de Northwest déclarait en 1995 [Ost95] : *« Airlines are no longer engaged in monumental efforts to steal market share. The change has been profound. There's a recognition that ubiquity is not a financeable concept. As airlines came to that conclusion in the early Nineties, they began to focus on markets where they have historic strength »*. Concrètement, les grands transporteurs rationalisèrent leurs activités, élaguèrent leur réseau et réduisirent leur capacité en annulant notamment des commandes de nouveaux appareils et en mettant d'autres au rancart, ce qui permit à l'industrie d'afficher des profits pour plusieurs années consécutives, jusqu'au ralentissement économique du début du millénaire, aggravé par les événements que l'on connaît.

2.5 Conditions nécessaires à l'application de l'o.d.r.

L'o.d.r. est utilisée dans plusieurs industries très différentes à première vue mais qui partagent certaines caractéristiques justifiant son application. Selon Kimes [Kim90], ces caractéristiques sont :

1. Capacité fixe

L'o.d.r. s'adresse avant tout aux industries ayant une capacité (ou un inventaire) fixe, ou ne pouvant être modifiée qu'à fort coût, de façon planifiée. Par exemple, reconfigurer la cabine d'un avion afin d'y ajouter de nouveaux sièges pour accommoder une demande plus grande que prévue sur un vol particulier est irréalisable en pratique.

2. Péremption de l'inventaire

Si une certaine unité de capacité n'est pas vendue à une date fixe, alors elle est perdue et n'engendre aucun revenu.

3. Produits vendus à l'avance

Les produits, qui sont en fait des unités de capacité, sont vendus, ou à tout le moins réservés, avant la date de leur consommation.

4. Coût fixe élevé

Le coût encouru pour offrir une unité de capacité est élevé, que cette unité soit vendue ou non.

5. Coût variable bas

Le coût supplémentaire encouru par la vente d'une unité de capacité autrement inutilisée est faible par rapport au coût total. Par exemple, il est très dispendieux de faire décoller un avion, qu'il soit rempli ou non. Cependant, la vente d'un siège additionnel amène des coûts supplémentaires négligeables, lesquels se limitent à la manutention de quelques bagages, à un repas et à une petite quantité de carburant (environ 2,15 ℓ de kérosène supplémentaire par passager par 100 kilomètres pour un Airbus A-320 doté de moteurs turbofan [Leg91])¹.

6. Demande variant selon le temps

La demande est stochastique.

En plus de posséder ces caractéristiques essentielles, les entreprises qui désirent appliquer une stratégie d'o.d.r. doivent disposer d'informations sur [Kim90, WB92] :

1. La segmentation du marché

Les entreprises doivent être en mesure d'identifier les différentes catégories de consommateurs et classer ceux-ci selon leur volonté à payer plus pour un meilleur service. Cette segmentation est obtenue à l'aide de restrictions ou «barrières» souvent artificielles (voir [Bot96]), lesquelles empêchent les consommateurs disposant de ressources financières plus appréciables et ayant une volonté de payer plus grande d'acheter des produits à rabais. Un exemple de ces restrictions en transport aérien est l'obligation (pour certaines classes de billets) que s'écoule au moins un samedi entre le vol d'aller et le vol de retour.

2. La demande historique et les habitudes de réservation

Les entreprises doivent avoir à leur disposition des données statistiques sur la demande et les habitudes de réservation pour chaque catégorie de consommateurs afin de pouvoir établir des distributions de probabilité sur la demande future.

¹Sur un vol Montréal-Paris de 5530 km, cela signifie 120 ℓ de kérosène par passager, ou encore 95 kg, le kérosène ayant une densité d'environ 0.8 kg/ ℓ . Au prix actuel de 350 \$US la tonne métrique de kérosène, le coût en carburant d'un passager supplémentaire sur un vol Montréal-Paris est donc de 33 \$US.

3. Les prix en vigueur

Les entreprises doivent continuellement et en temps réel connaître les prix offerts par leurs concurrents.

2.6 Méthodes de résolution

Traditionnellement, quatre types de méthodes sont utilisées pour résoudre des problèmes d'o.d.r. [Kim90] :

1. Méthodes basées sur la programmation mathématique.
2. Méthodes basées sur les théories économiques.
3. Méthodes basées sur l'analyse de courbes de seuil et de modèles de revenus marginaux.
4. Méthodes utilisant des systèmes experts et faisant appel à l'intelligence artificielle¹.

Jusqu'au début des années 90, il semble que les deuxième et troisième approches aient été les plus populaires, notamment à cause de la puissance informatique et du temps de résolution considérables requis par les approches basées sur la programmation mathématique et l'intelligence artificielle [Kim90]. Cependant, les récentes innovations dans ces domaines, tant sur le plan théorique que technique, ainsi que l'augmentation de la puissance des ordinateurs, font en sorte que ces méthodes sont maintenant largement utilisées.

Il est important de remarquer que les problèmes d'o.d.r. doivent être résolus de façon continue et répétée; toute solution doit être obtenue relativement rapidement, efficacement et à un coût raisonnable. Ainsi, l'optimalité d'une solution, bien que souhaitable, n'est pas absolument nécessaire, en autant que cette solution puisse être obtenue rapidement et possède un niveau acceptable de précision [Kim90, SLD92].

¹Un *système expert* est un programme informatique qui tente d'imiter le raisonnement logique d'un humain expert dans le domaine. Ces systèmes, basées sur l'évaluation de prédicats logiques, sont typiquement implantés à l'aide de langages spécialisés tels le PROLOG.

2.7 Caractéristiques propres à l'industrie aérienne

Dans un discours prononcé au milieu des années 90 devant des étudiants en gestion de l'Université de Caroline du Nord, l'assureur et investisseur américain Warren Buffet, président de Berkshire Hathaway (et deuxième homme le plus riche au monde au moment d'écrire ces lignes) déclara que, lorsque les frères Wright ont effectué le premier vol de l'histoire de l'aviation à Kitty Hawk (Caroline du Nord) le 17 décembre 1903, *«if there had been a capitalist down there, the guy should have shot down Wilbur»* [Smi95].

Cette pointe d'humour noir illustre bien la relation haine-amour entre les investisseurs et une industrie habituée aux déficits. Comme le fait remarquer Smith [Smi95], depuis l'époque d'Howard Hughes les investisseurs ont souvent été attirés par les compagnies aériennes pour des raisons apparemment irrationnelles, et plusieurs y ont perdu des fortunes (y compris M. Buffet), un peu comme dans l'industrie du cinéma ou du sport professionnel. La rentabilité d'une compagnie aérienne dépend en effet d'un ensemble de facteurs qui doivent tous être réunis. Or la façon de fonctionner des transporteurs aériens (le *business model*) a souvent par le passé fait fi de l'un ou l'autre de ces facteurs, conduisant irrémédiablement à des pertes.

Nous discutons ici de quelques-unes des caractéristiques propres à l'industrie aérienne qui la rendent si particulière et rendent nécessaire l'utilisation de méthodes spécialisées telle l'o.d.r.

2.7.1 Nature périssable et interchangeable des produits

Un siège sur un avion est un produit périssable, un peu comme un pot de yogourt dans le comptoir réfrigéré d'un super-marché : si le siège demeure invendu lorsque l'avion décolle, il devient alors périmé et ne procure aucun revenu à la compagnie (un tel produit est dit «gaspillé»). D'où la tentation d'offrir ce produit à rabais lorsque la date de péremption approche. *«You can either get a penny for it or you can get nothing for it, so you sell it for a penny»*, comme le faisait remarquer John Dasburg, président de Northwest Airlines [Smi95]. Toutefois, contrairement au yogourt, la quantité de sièges disponible est limitée et diminue à mesure que la date de départ approche, ce qui peut faire augmenter la valeur des sièges toujours disponibles. Les transporteurs aériens doivent donc constamment jongler entre le risque de gaspiller leurs produits et la possibilité de vendre ces produits à meilleur prix si la demande se fait sentir peu de temps avant le départ.

De plus, en dépit des efforts considérables faits par les transporteurs depuis une dizaine d'années dans le but de se démarquer de la compétition et de fidéliser leur clientèle, un billet d'avion demeure un produit fondamentalement interchangeable pour la majorité des consommateurs. Rien ne ressemble plus à un billet pour un trajet entre les villes A et B qu'un autre billet pour le même trajet, qu'il soit offert ou non par le même transporteur. Les coûts de transfert d'un transporteur à l'autre sont aussi pratiquement nuls. Alors que dans le cas d'une machine-outil ou d'un système informatique, un individu ou une organisation a avantage à toujours faire affaire avec le même fournisseur (pour réduire les coûts de formation et de maintenance, assurer la compatibilité, etc.), un passager peut facilement passer d'un transporteur à un autre sans que cela n'entraîne de coût supplémentaire.

2.7.2 Coûts fixes élevés, coûts marginaux négligeables

Les coûts engendrés par la mise en marché d'un produit dans l'industrie aérienne sont énormes, allant de l'amortissement de l'achat de l'appareil au carburant, en passant par les frais d'aéroport et les salaires de l'équipage. La quantité de capital requise afin d'opérer une compagnie aérienne est imposante et constamment en mouvement (les financiers utilisent le concept de *cash-flow* pour décrire ce phénomène). Or les coûts marginaux entraînés par la vente d'un produit, concrètement un siège sur un vol, sont négligeables par rapport au prix de vente de ce siège. Aucune autre industrie ne possède autant d'occasions de revenu ponctuelles qui apparaissent et disparaissent à mesure que le marché évolue (d'autant plus que les produits sont périssables). Prises individuellement, ces opportunités de revenu peuvent paraître insignifiantes, mais leur impact cumulatif sur les performances financières d'un transporteur peut être spectaculaire.

Transport aérien et cœur vide

Cette particularité de l'industrie aérienne ainsi que la crise traversée par l'industrie aérienne (en particulier aux États-Unis) au début des années 90 ont amené certains économistes à s'interroger sur la possibilité que cette industrie comporte un «cœur vide». Ce concept emprunté à la théorie des jeux coopératifs permettrait notamment d'expliquer l'instabilité de l'industrie, les guerres tarifaires que se sont souvent livrées les transporteurs majeurs sur le marché nord-américain ainsi que les faillites spectaculaires ayant entraîné la disparition de nombreux transporteurs depuis la déréglementation¹.

¹ «[...] Québecair, Transworld, Northeast, Eastern, Western, pis Pan American» [CP68] ont tous fait faillite

De façon informelle, on peut décrire un jeu coopératif comme un ensemble d'acteurs économiques (les «joueurs») produisant ou consommant des biens et services qu'ils transigent entre eux. Le cœur de ce jeu (i.e. activité économique) est l'ensemble des situations dans lesquelles aucun sous-ensemble de joueurs ne peuvent améliorer leur sort en rejetant la situation en question et en transigeant uniquement entre eux. En d'autres mots, le cœur se compose de tous les états d'équilibre pour lesquels aucun gain supplémentaire n'est possible pour aucun sous-ensemble de joueurs [But96, But02]. Un secteur économique avec un cœur vide est caractérisé par l'incapacité des joueurs à atteindre une situation d'équilibre, chaque joueur ayant toujours la possibilité d'améliorer son profit en rompant l'équilibre existant. Bref, les compétiteurs ne sont pas en mesure d'atteindre un équilibre de Pareto.

Une petit exemple, adapté de [But02], permet d'illustrer ce concept. Supposons qu'un transporteur X possède un monopole sur un marché. Il en coûte 140\$ au transporteur pour opérer son appareil, peu importe que celui-ci soit plein ou vide. Son appareil possède une capacité de deux passagers, et on suppose que les deux sièges sont offerts au même prix. Supposons de plus que trois passagers, appelons-les a , b et c , désirent acheter un billet sur ce vol. a est prêt à payer 100\$, b 80\$ et c 75\$. Dans ce cas-ci, le cœur se compose de toutes les situations où a et b obtiennent un billet, mais pas c , la différence entre les situations du cœur étant le tarif payé par a et b . En effet, si le prix demandé pour un billet est supérieur à 75\$, alors X , a et b forment une coalition et X est assuré d'un profit d'au moins $2 \times 75\$ - 140\$ = 10\$$. Par contre, si le prix demandé pour un billet est inférieur ou égal à 75\$, alors c est aussi intéressé à acheter un billet et entre dans le jeu, ce qui signifie un excès de demande et une baisse de profit pour X . Dans ce cas, le passager n'obtenant pas de billet à intérêt à briser la coalition entre le transporteur et les deux autres passagers afin d'en former une autre dans laquelle il obtient un billet. Cette situation ne fait donc pas partie du cœur.

Admettons maintenant qu'un autre transporteur, Y , fasse son apparition sur le marché. Y opère un appareil identique à celui de X et possédant la même structure de coûts. Cette fois-ci, l'offre excède la demande : il y a quatre sièges disponibles et trois passagers intéressés à voyager. On peut déduire que les deux transporteurs doivent faire le même profit, car sinon un des deux pourrait abaisser légèrement son prix et dérober les passagers de l'autre. Par exemple, X forme une coalition avec a et b et vend ses deux sièges. Dans ce cas, Y peut à son tour abaisser son prix et former une nouvelle coalition avec a et b (et possiblement c si le tarif exigé est inférieur à 75\$),

au cours de années 80 ou au début des années 90. On peut ajouter à cette liste des noms comme Braniff, Capitol, Southern, et plus récemment Swissair et Sabena.

et ainsi de suite, jusqu'à ce que cette guerre tarifaire mène les deux transporteurs à un profit nul. Une situation avec profit nul ne fait pas partie du cœur, car les deux transporteurs ont intérêt à hausser leurs prix. Bref, il est impossible d'atteindre un équilibre entre les compétiteurs, et on doit conclure que le cœur sur ce marché est vide.

Cet exemple est bien entendu simpliste et irréaliste, mais certains auteurs se sont attardés au problème en s'appuyant sur des bases mathématiques rigoureuses¹ et prétendent que de telles situations sont bien réelles, mais difficiles à observer empiriquement. Il existe cependant certaines conditions sous lesquelles un cœur vide peut exister. La plus importante [But96] est que le coût unitaire de production ne soit pas une fonction non décroissante du volume produit. En d'autres termes, les coûts de production fixes sont élevés et les coûts marginaux encourus par la production d'unités supplémentaires sont nuls (ou à tout le moins négligeables), ce qui est une des propriétés de l'industrie aérienne. Sjostrom [Sjo93] s'est interrogé sur l'existence d'un problème de cœur vide dans l'industrie maritime et sur les solutions possibles pour y remédier, mais ses conclusions peuvent aussi s'appliquer au domaine aérien. Cet auteur affirme entre autres que :

- plus la différence entre les coûts moyens de production des fournisseurs est grande, plus fortes sont les chances d'atteindre un équilibre compétitif,
- les chances d'avoir un cœur vide sont plus grandes lorsque la demande est inélastique,
- plus la corrélation entre la capacité offerte et la demande est grande, plus fortes sont les chances que le cœur soit vide,
- les ententes entre les joueurs pour créer un cœur sont plus susceptibles de se produire en période de ralentissement économique.

Les spécialistes ne sont cependant pas tous unanimes quant à l'existence d'un cœur vide dans l'industrie aérienne et certains n'adhèrent pas aux conclusions quelque peu fatalistes des auteurs cités ci-dessus selon lesquels cette industrie est condamnée à continuellement s'auto-détruire, les compétiteurs étant incapables d'atteindre un équilibre profitable pour tous. C'est le cas notamment d'Alfred E. Kahn, l'architecte de la déréglementation aux États-Unis, qui dans [Smi95] affirme de pas détenir de «preuves conclusives» lui permettant d'adhérer aux thèses des partisans du cœur vide.

Quoi qu'il en soit, l'industrie aérienne possède une structure de coûts distinctive qui rend certaines opérations, notamment la tarification, particulièrement délicates.

¹Voir par exemple Telser [Tel96].

2.7.3 Structures tarifaires complexes

Jusqu'au début des années 80, les transporteurs évoluaient dans un environnement hautement stable et réglementé, que ce soit par des organismes externes, comme le C.A.B. américain, ou encore des organismes internes tels l'I.A.T.A. (*International Air Transport Association*), créée en 1945 par les transporteurs eux-mêmes dans le but de s'auto-discipliner et de faciliter les relations avec les consommateurs, les agences gouvernementales, etc. Or depuis la déréglementation, on a assisté à la multiplication des produits et tarifs offerts par les transporteurs ainsi qu'à l'apparition d'une pléthore de conditions associées à ces produits, si bien qu'aujourd'hui seuls les experts s'y retrouvent. À titre d'exemple, sur un marché nord-américain typique, chaque transporteur peut offrir plus d'une vingtaine de bases tarifaires différentes, lesquelles peuvent être modifiées jusqu'à trois fois chaque jour.

Afin de gérer cette montagne d'information, l'industrie s'est dotée de moyens informatiques élaborés, comme par exemple le système ATPCo¹. Cependant, selon Kretsch [Kre95], la plupart des transporteurs ne sont toujours pas en mesure de traiter et d'analyser toute cette information, et encore moins de mettre en place des outils informatiques permettant d'automatiser le processus. Selon cet auteur, ces difficultés résultent de la complexité des structures tarifaires. Aujourd'hui, un produit tarifaire se compose de quatre éléments :

1. la base tarifaire et son prix,
2. les règles d'application,
3. les notes supplémentaires (e.g. tarif de vente ou de groupe),
4. les trajets ou marchés applicables.

Ces quatre éléments sont indépendants et doivent être traités séparément, ce qui complique le développement de formats électroniques simples et rend l'utilisation de structures de données lourdes et d'infrastructures informatiques complexes pratiquement inévitables. Kretsch [Kre95] prétend qu'une base de données relationnelle dotée d'une interface usager conviviale est le type de système le plus approprié, mais l'implantation, l'administration et la maintenance d'un tel

¹ATPCo, *Airline Tariff Publishing Company*, fut fondée dans les années 60 par un regroupement de transporteurs américains dans le but de centraliser l'impression et la distribution de livrets contenant tous les tarifs des transporteurs participants. Au fil des années, des transporteurs canadiens et internationaux se sont ajoutés, soit comme actionnaires ou comme clients, si bien que le système publie aujourd'hui les tarifs de 550 compagnies aériennes à travers le monde. Au cours des années 80, devant l'explosion du nombre de produits tarifaires, l'entreprise s'est dotée de moyens informatiques sophistiqués, de telle sorte qu'aujourd'hui les compagnies souscrivant à ce service ont accès de façon électronique et en temps réel aux données tarifaires de leurs compétiteurs.

système nécessite des investissements considérables échelonnés sur plusieurs années.

Même si un transporteur est en mesure de traiter l'information tarifaire de ses compétiteurs, l'analyse de cette information, de façon manuelle ou informatisée, demeure un tâche difficile, compliquée notamment par deux réalités de l'industrie aérienne. D'abord, comme il n'existe pas de standard quant aux codes désignant les bases tarifaires, il est difficile d'établir des correspondances directes entre les produits de transporteurs différents. Aussi, certains tarifs ne sont pas publiés de façon électronique. C'est le cas notamment sur le marché est-asiatique, où les transporteurs communiquent encore leurs tarifs de façon privée aux distributeurs.

Une fois que la structure tarifaire des compétiteurs a été traitée et analysée, un transporteur peut procéder à l'optimisation de ses propres tarifs. Pour des raisons exposées aux sections précédentes, la tarification dans l'industrie aérienne est essentiellement basée sur la demande. Kretsch [Kre95] affirme qu'un tel processus d'optimisation des tarifs, s'il est conduit correctement, peut réduire le nombre de contrôles d'inventaires requis tout en stimulant la demande. Il prétend entre autre que «la clef de l'optimisation des tarifs est le développement d'un modèle qui puisse mesurer la réaction d'un passager au prix et aux autres attributs d'un vol». Il conclut en affirmant que le développement d'un tel modèle est difficile mais possible, et que les bénéfices potentiels sont énormes.

2.8 Tendances actuelles en o.d.r.

Nous discutons ici des tendances actuelles en optimisation des revenus reflétées par la littérature récente à ce sujet. Nous nous intéressons principalement aux sous-problèmes d'allocation de la capacité et de tarification, puisqu'il s'agit des deux aspects sur lesquels portent nos travaux.

2.8.1 Allocation de la capacité

Afin de mieux comprendre les tendances actuelles dans le domaine des systèmes d'allocation de la capacité, il est bon de décrire brièvement l'évolution de ce type de système et l'utilisation qu'en ont fait les compagnies aériennes depuis son apparition. Cet historique se retrouve notamment dans [Bel98].

Le but d'un système d'allocation de la capacité (du moins en transport aérien) est de mettre en marché des sièges dont la vente à plein prix est peu probable en offrant ces sièges à un tarif avantageux, en s'assurant toutefois que les passagers qui ont une propension à payer plus grande n'aient pas accès à ces sièges écoulés à rabais.

La première génération de systèmes d'allocation de la capacité s'appuyait sur le système de réservation informatisé de la compagnie (CRS, *Computerized Reservation System*). De façon régulière (e.g. toutes les nuit), on extrayait un instantané de ce système afin que les analystes puissent réviser la disponibilité des différentes classes tarifaires en tenant compte des nouvelles réservations enregistrées dans le CRS. Il s'agissait avant tout d'un travail réactif effectué de façon manuelle. Par la suite, on a doté les CRS de fonctions leur permettant de comparer les réservations effectuées pour un vol avec des données historiques pour des vols comparables. Ces systèmes de deuxième génération pouvaient, en se basant sur ces constats historiques, suggérer aux analystes d'augmenter ou de diminuer la capacité de certaines classes tarifaires. Quant aux systèmes de troisième génération, apparus au cours de années 80, ils incorporent des modèles mathématiques permettant d'établir des prévisions basées à la fois sur le nombre effectif de réservations et les données historiques, d'optimiser l'allocation des sièges aux différentes classes tarifaires et de déterminer la politique de surréservation pour chaque vol.

Sans entrer dans les détails techniques de chaque modèle (ce qui sera fait lors de la revue bibliographique), on peut tracer l'évolution des approches utilisées, en se basant notamment sur la portée des modèles.

Les premiers modèles utilisés étaient de type *leg- and class-based*, c'est-à-dire qu'ils considéraient indépendamment chaque *segment* de vol et établissaient des limites sur le nombre de sièges pouvant être réservé dans chaque classe tarifaire. Cette approche est appropriée pour un transporteur n'effectuant que des liaisons directes, i.e. où tous les *vols* ne se composent que d'un seul segment. Cependant, depuis la fin des années 80, dans le but de réduire les coûts et d'augmenter le nombre de marchés desservis, la majorité des compagnies aériennes exploitent un réseau de type *radial*, où les vols transitent par le *moyeu*. Dans un tel cas, les modèles d'allocation de la capacité doivent tenir compte des interactions entre les marchés partageant des segments communs dans le réseau. La majorité des modèles proposés sont ainsi basés sur des techniques d'optimisation de réseau. Toutefois, ces modèles ne prennent pas en considération les aspects stochastiques et dynamiques de problème, pourtant bien présents. De plus, ces modèles ne sont pas en mesure de générer des schémas d'allocation de la capacité qui tiennent compte de

l'*emboîtement* des classes tarifaires. Cet aspect réseau du problème d'allocation de la capacité correspond à ce que Kimes [Kim90] appelle des problèmes «multi-étapes». Dans son article, écrit en 1990, cet auteur suggère que le traitement de tels problèmes devrait faire l'objet de recherches afin que les stratégies d'o.d.r. en transport aérien soient vraiment efficaces. Au cours de la dernière décennie, d'autres auteurs ont suivi cette recommandation et se sont attaqués au problème d'allocation de la capacité sur réseau.

L'allocation de la capacité au niveau du réseau pose aussi un problème logistique. Comme l'indique Vinod [Vin95], les CRS sauvegardent traditionnellement leurs informations au niveau du segment et non du vol ; la plupart des compagnies aériennes ne disposent donc pas *a priori* d'informations historiques sur le réseau entier. Il faut préalablement reconstruire ces données, ce qui peut être long et coûteux. Aussi, la complexité algorithmique des techniques classiques d'optimisation de réseau a longtemps porté les transporteurs à croire que de telles méthodes étaient inapplicables en pratique. Il semble que cette opinion ait été répandue dans l'industrie jusqu'au milieu des années 90.

La tendance observée depuis une dizaine d'années dans ce domaine est donc de raffiner les modèles d'allocation de la capacité stochastiques de type *leg-based* de façon à ce qu'il puissent tenir compte des effets réseau. Ceci se traduit par l'utilisation de différents gadgets de modélisation tels les «classes virtuelles». Ce type de modèle est parvenu à maturité et est aujourd'hui appliqué par de nombreux transporteurs majeurs internationaux.

2.8.2 Tarification

L'allocation de la capacité et le contrôle d'inventaire de sièges étant parvenus à une certaine maturité, la tarification sera sans doute l'aspect du problème d'o.d.r. qui se verra accorder le plus d'attention au cours des prochaines années. Deux articles récents, celui de Cross [Cro98] et celui de Garvett et Michaels [GM98] présentent la tarification comme *the next Big Thing* dans l'industrie aérienne.

Cross identifie certains facteurs qui illustrent l'importance de la tarification pour une compagnie aérienne :

- **Le prix est le principal critère d'achat**

Plus que jamais, le prix est le premier critère sur lequel les consommateurs basent leur décision

au moment de l'achat d'un billet d'avion. Cette tendance est commune à tous les segments de la clientèle, peu importe qu'il s'agisse de clients voyageant à rabais ou s'intéressant aux produits haut-de-gamme.

- **L'accès à l'information procure un plus grand choix**

L'émergence de nouvelles sources d'information, en particulier l'Internet, permet aux consommateurs de prendre des décisions plus réfléchies lors de l'achat d'un billet. Il est entre autre possible de comparer les tarifs offerts par diverses compagnies.

- **Les clients corporatifs ont une attitude différente**

Plusieurs compagnies, dans le but de réduire les coûts de voyage de leur personnel, sont, comme les voyageurs particuliers, de plus en plus à l'affût des rabais. Certains délaissent la classe affaire au profit de la classe économique.

- **Égaler ou ne pas égaler**

Au lieu d'égaliser automatiquement les prix offerts par leurs compétiteurs, les transporteurs commencent à analyser l'impact de telles décisions et adoptent des politiques tarifaires plus sophistiquées.

- **Les objectif corporatifs sont reflétés dans les politiques tarifaires**

Certaines compagnies aériennes ont réalisé que les tarifs qu'elles offraient ne correspondaient pas toujours à leurs objectifs à moyen et à long terme. Par exemple, si des dirigeants décident d'augmenter la part de marché détenue par leur compagnie pour une certaine paire origine-destination, leur politique de prix et de marketing doit s'ajuster en conséquence. La tendance est à une plus grande interaction entre la direction d'une compagnie et son service de tarification.

Dans un discours prononcé en 1992 par Robert Crandall, l'ancien président d'American Airlines (et père spirituel le l'o.d.r.) discute de quelques-uns de ces facteurs. Cet extrait se trouve dans Cross [Cro98].

«One of the basic truths of our industry is that consumers are price driven. As a consequence, no carrier can or will give any competitor a price advantage. [...] Like it or not, airline economics are subject to the reality that consumers regard one airline's seats as a ready substitute for any other airline's seats. As a result, very few people will pay anything extra to fly one carrier versus another—even though they may consider one product marginally superior to another. Moreover, computerized reservation systems give airline consumers complete and instant information about every carrier's price. Thus the movement of customers to any carrier offering a lower price is

both swift and certain—which is why airlines match whoever flies the lowest fares with such alacrity.»

Cross résume son propos comparant la mentalité traditionnelle d'une compagnie aérienne et la mentalité d'une compagnie aérienne ayant adopté une stratégie de tarification délibérée et réfléchie. Cette comparaison est reproduite au tableau 2.1.

Mentalité «traditionnelle»	Mentalité «tarification optimale»
Les transporteurs contrôlent le prix que les consommateurs payent pour des produits et services.	Les transporteurs déterminent les prix selon le comportement des consommateurs; les meilleurs transporteurs réagissent aux signes annonciateurs de changements dans le comportement des consommateurs.
Les consommateurs paient pour les produits et services qu'ils achètent.	Les consommateurs paient pour l'opportunité d'acheter au prix et aux conditions en vigueur à un instant particulier.
Les transporteurs segmentent leur clientèle selon des critères démographiques et sociaux tels l'âge, le sexe, le revenu, etc.	Les transporteurs segmentent leur clientèle selon les variations dans le comportement d'achat, en se basant sur la sensibilité au prix.
Les prix sont déterminés sur une base coût plus profit.	Les transporteurs ajustent leurs prix pour accommoder les passagers sensibles aux coûts tout en réalisant des profits.
Les ventes sont organisées par le département de marketing, souvent dans le but d'atteindre des objectifs de volume.	Les ventes et les rabais sont organisés par le département de tarification dans le but de maximiser les revenus.

TAB. 2.1 – Comparaison entre la mentalité traditionnelle et la mentalité axée sur la tarification optimale du produit.

Garvett et Michaels [GM98] s'intéressent à tout ce qui entoure le développement et la mise en place d'un *Pricing Decision Support System* (PDSS) dans une compagnie aérienne. Ils définissent un PDSS comme un outil permettant à un transporteur d'évaluer la structure tarifaire d'un marché, y compris les conditions d'entrée et les coûts d'opportunité, et d'identifier les opportunités de revenu en prévoyant ou simulant la demande, le trafic, le rendement, la diversion du trafic

d'autres compagnies, etc. Les auteurs affirment qu'au moment d'écrire leur article (en 1998) de tels systèmes étaient encore à l'état embryonnaire.

L'article de Garvett et Michaels identifie les objectifs qu'un PDSS devrait permettre d'atteindre. Ces objectifs sont :

- **Maximisation des revenus**

L'objectif principal de la majorité des compagnies aériennes est bien entendu de maximiser le retour sur les investissements de leurs actionnaires. À court terme, une fois les horaires fixés et les ressources affectées, un transporteur élabore la structure tarifaire qui maximise son revenu, étant donné la situation ponctuelle du marché, i.e. demande et structures tarifaires des concurrents. Cette tarification optimale est basée sur une segmentation stricte de la clientèle permettant d'imposer des tarifs élevés aux segments inélastiques et de stimuler la demande des segments élastiques par des tarifs concurrentiels. Cette façon de faire est à l'opposé de la stratégie du « moi-aussi », qui consiste à systématiquement égaler les prix offerts par les concurrents, que ceux-ci soient intelligents ou stupides.

- **Tarification selon les coûts et la valeur**

Traditionnellement, le prix d'un produit était déterminé par les coûts de production de ce produit plus la marge de profit désirée. Cependant, dans l'industrie aérienne, la demande et la structure du marché ont une influence considérable sur les tarifs proposés. Afin de maximiser les revenus, la tarification doit tenir compte non seulement des coûts de production, mais aussi de la valeur des produits aux yeux des consommateurs.

- **Processus de distribution de de vente**

Un PDSS doit faciliter l'administration et la communication de structures tarifaires complexes comme celles en vigueur aujourd'hui dans l'industrie aérienne. Cette structure doit être compréhensible pour tous les intervenants de la chaîne de distribution.

- **Objectifs secondaires**

Les auteurs identifient certains objectifs secondaires d'un PDSS : contrôler l'utilisation des produits tarifaires, se conformer à certaines contraintes légales imposées par les autorités, etc.

Selon les auteurs, le principal moyen pour atteindre ces objectifs est une segmentation poussée de la clientèle basée sur des critères tels que la qualité de service, l'itinéraire, la saisonnalité, le jour et la date, le point de vente, etc. Ceux-ci précisent aussi que, étant donné le nombre de produits offerts, la fréquence des changements, etc., l'implantation d'un PDSS représente un défi technique considérable. Les auteurs insistent sur le délai de réaction du système aux change-

ments dans les structures tarifaires de la compétition. L'importance de ce délai est cruciale si la compagnie désire profiter des opportunités de revenus ponctuelles qui apparaissent et disparaissent continuellement. Ils vont même jusqu'à classer les compagnies aériennes en deux grandes catégories, «*the quick and the dead.*»

Garvett et Michaels discutent aussi des fonctionnalités qu'un PDSS devrait idéalement offrir. Parmi les éléments principaux, on retrouve :

- **L'évaluation de structures tarifaires par marché**

Sur un marché donné, les structures tarifaires de l'ensemble des compagnies peuvent comprendre des centaines de produits (i.e. bases tarifaires), chacun étant possiblement modifié plusieurs fois par jour. Il est extrêmement difficile, voire impossible, pour un analyste humain d'administrer autant d'information à caractère dynamique. Un PDSS doit fournir à un analyste la possibilité d'évaluer les différentes structures tarifaires en vigueur sur chaque marché et de présenter cette information de façon claire et concise.

- **Comparaisons entre les marchés**

Des informations complémentaires peuvent être obtenues en comparant les structures tarifaires et les rendements de marchés distincts, par exemple en fonction de la distance séparant les origines et les destinations.

- **Analyse des tarifs de la compétition**

La possibilité de consulter, de comparer et d'analyser les tarifs de la compétition est l'un des aspects importants d'un PDSS. De plus, certains ajustement tarifaires triviaux sur des marchés secondaires peuvent être automatisés par le système.

- **Contrôle des changements tarifaires**

Par sa capacité de traiter de grandes quantités d'information, un PDSS indique à l'analyste quels changements récents parmi les tarifs de la compétition sont les plus significatifs et nécessitent prioritairement son attention.

- **Analyse de scénarios et simulation**

Un analyste désire mesurer l'impact des changements qu'il apporte à la structure tarifaire de sa compagnie. Le PDSS doit fournir la possibilité à l'analyste de considérer et de simuler différents scénarios en lui indiquant comment ces changements vont affecter la demande, le trafic, les parts de marché, etc. Les auteurs désignent cette fonctionnalité comme le «*what-if pricing*».

- **Facilitation du processus de publication des tarifs**

Une fois décidés, les changements aux structures tarifaires doivent être publiés par l'intermé-

diaire d'un système tel ATPCo. Cette étape peut être prise en charge par le PDSS.

Garvett et Michaels concluent leur article en mentionnant des gains potentiels de l'ordre de 5 à 10%. Il tiennent cependant à mettre les compagnies aériennes en garde : un tel système ne sera pas développé du jour au lendemain.

2.8.3 Intégration de l'allocation de la capacité et de la tarification

Il existe une relation de dualité entre l'allocation de la capacité et la tarification, chaque aspect ayant une influence sur la demande et la demande influençant à son tour à la fois l'allocation de la capacité disponible et les tarifs des produits offerts. Plusieurs auteurs ont suggéré l'exploration d'approches permettant d'intégrer ces deux aspects, i.e. résoudre conjointement les problèmes de contrôle d'inventaire de sièges et de tarification, de façon à ce que les quantités de sièges alloués à chaque classe tiennent compte explicitement des tarifs en vigueur et vice-versa. C'est le cas notamment de Kimes [Kim90], selon qui l'intégration de ces deux aspects est «extrêmement souhaitable». À notre connaissance, aucune méthode permettant de résoudre simultanément et conjointement les problèmes de contrôle d'inventaire et de tarification n'a encore été proposée¹. McGill et van Ryzin [MvR99] affirment avoir pris connaissance de quelques documents de travail et de thèses consacrés à ce sujet, mais aucun document n'a été publié à grande échelle. Comme nous le verrons à la section suivante, seuls Kuyumcu et Garcia-Diaz [KGD00] ont récemment proposé une approche qu'ils présentent, à notre avis à tort, comme un méthode de résolution conjointe.

¹Certains auteurs, par exemple Gallego et van Ryzin [GvR97] et You [You99], affirment que des modèles de tarification sophistiqués peuvent implicitement contrôler la quantité de sièges disponibles, notamment en assignant un prix «infini» à des sièges que l'on désire retirer du marché. Leurs modèles ne produisent cependant pas de schéma d'allocation de la capacité disponible.

Chapitre 3

Revue bibliographique

Ce chapitre passe en revue plusieurs articles consacrés à l'o.d.r. en général et plus particulièrement aux problèmes d'allocation de la capacité et de tarification. Il présente brièvement les principales solutions à ces deux problèmes proposées par la communauté scientifique depuis une vingtaine d'années. Nous nous intéressons avant tout aux articles concernant le domaine du transport aérien car il s'agit de notre champ d'application principal.

3.1 Ouvrages généraux

Relativement peu d'articles généraux sur l'o.d.r. en transport aérien ont été publiés dans des revues scientifiques. Dans un article récent, McGill et van Ryzin [MvR99] présentent l'o.d.r. et ses principaux aspects et font en parallèle une revue bibliographique poussée citant des articles scientifiques publiés depuis les années 50. Kimes [Kim90] présente l'o.d.r. de façon concise, en énumère les caractéristiques principales et en vulgarise les concepts fondamentaux. Smith, Leimkuhler et Darrow [SLD92] présentent une étude de cas intéressante en exposant les méthodes d'o.d.r. développées par le transporteur aérien American Airlines pour son propre compte. L'article de Weatherford et Bodily [WB92] aborde le problème d'o.d.r. de façon scientifique et présente une taxinomie basée sur la difficulté algorithmique des sous-problèmes s'y rattachant. Botimer [Bot96] aborde les implications commerciales et sociales de l'utilisation de la recherche opérationnelle dans la résolution de problèmes d'o.d.r. Finalement, le livre édité par Yeoman

et Ingold [YI97] expose succinctement plusieurs aspects de l'o.d.r. (économiques, commerciaux, légaux, etc.) et présente une multitude d'études de cas, la plupart se rapportant cependant à l'industrie hôtelière et touristique.

3.2 Allocation de la capacité

On peut classer les méthodes d'allocation de la capacité et de contrôle d'inventaire de sièges de différentes façons. Une première distinction existe entre les méthodes qui considèrent l'ensemble du réseau et tiennent compte des interactions entre les segments, notamment dans un réseau radial, et les méthodes qui considèrent chaque segment de façon indépendante. Il existe une autre distinction entre d'un côté les méthodes dites «statiques», i.e. qui supposent que le processus de réservation est divisé en un certain nombre d'intervalles dans lesquels une seule classe tarifaire est offerte, les classes les moins dispendieuses étant réservées en premier, et de l'autre côté les méthodes dites «dynamiques» qui ne posent pas d'hypothèse quant à l'ordre d'arrivée des requêtes de réservation. Notons que la plupart de ces méthodes supposent aussi que tous les passagers sont emphanes, que la demande pour chaque classe est indépendante et exogène au modèle et qu'il n'y a pas de réservation de groupe.

3.2.1 Formulations par segments de vol

Méthodes statiques

Les méthodes statiques posent l'hypothèse (très forte) que les sièges sont réservés¹ par ordre croissant de tarif et elles établissent *a priori* des schémas d'allocation fixes. Elles supposent de plus l'absence d'aphanie et d'annulation.

La première méthode de contrôle d'inventaire par segment fut celle de Littlewood en 1972 [Lit72]. En supposant que les prix soient connus et fixés, cette méthode suggère une façon de déterminer si une nouvelle réservation devrait être acceptée en basse-classe (tarif à rabais) sur un segment

¹Il existe une différence pratique entre la *réservation* d'un billet et son *achat*. Cette différence n'a toutefois d'importance que lorsque la méthode de contrôle d'inventaire considère la possibilité d'annulation. Or la majorité des méthodes statiques (et la totalité de celles que nous présentons dans cette section) supposent que les passagers sont emphanes et qu'il n'y a pas d'annulation. Nous pouvons donc utiliser de façon interchangeable les concepts de réservation et d'achat de billet pour décrire ces méthodes.

unique avec deux classes tarifaires séparées mais partageant la même cabine. Soit t_i le tarif d'un billet en classe i , s_i le nombre de sièges attribués (i.e. protégés) à la classe i et r_i la demande (i.e. le nombre de requêtes) en classe i . On suppose que les classes sont étiquetées de telle sorte que $t_i > t_{i+1}$ pour $i = 1, 2, \dots$. En d'autres mots la classe 1 est la plus dispendieuse, et ainsi de suite.

La règle de Littlewood stipule qu'une réservation en basse-classe peut être acceptée tant que

$$t_2 \geq t_1 \cdot \Pr\{r_1 > s_1\}. \quad (3.1)$$

De façon intuitive, cette règle suggère d'accepter la vente d'un siège additionnel en basse-classe si le revenu ainsi obtenu égale ou excède le revenu espéré par la vente d'un siège en haute-classe. La plus petite valeur de s_1 pour laquelle l'inégalité (3.1) est satisfaite représente le niveau de protection en haute-classe. Notons que cette règle ne produit pas de schémas d'allocation qui tiennent compte de l'emboîtement des classes tarifaires.

Brumelle et al. [BMO⁺90] s'intéressent à la règle de Littlewood dans le cas où les demandes pour les deux classes sont positivement corrélées ou encore lorsque le *surclassement* engendre une dépendance entre les demandes.

En 1987, Belobaba [Bel87, Bel89, Bel98] proposa une généralisation de la règle de Littlewood pouvant s'appliquer à plusieurs classes tarifaires et produisant des schémas d'allocation de sièges emboîtés. Cette méthode se base sur le concept d'*Expected Marginal Seat Revenue* (EMSR). Elle suppose que les prix sont connus et fixes durant toute la période de réservation. Comme la méthode détermine des limites de réservation au niveau de la classe tarifaire et qu'une classe peut contenir plusieurs bases tarifaires, chacune ayant un prix différent, les prix utilisés sont la moyenne des prix des bases tarifaires pour chaque classe. On suppose de plus que la demande pour chaque classe tarifaire sur chaque segment est connue. Soit p_i la fonction de densité pour le nombre de requêtes de réservation en classe i . Belobaba affirme utiliser généralement une densité normale pour modéliser la demande aléatoire r_i de chaque classe i . Supposons que s_i sièges aient été alloués à la classe i . La probabilité cumulative qu'aucune requête ne soit rejetée est

$$P_i(s_i) = \Pr\{r_i \leq s_i\} = \int_0^{s_i} p_i(z) dz. \quad (3.2)$$

De façon équivalente, la probabilité que le nombre de requêtes en classe i excède le nombre de

sièges alloués à cette classe est

$$\Pr\{r_i > s_i\} = \int_{s_i}^{\infty} p_i(z) dz = 1 - P_i(s_i) = \bar{P}_i(s_i). \quad (3.3)$$

Le *Expected Marginal Seat Revenue* du $z^{\text{ième}}$ siège en classe i est défini comme le tarif moyen en classe i (que nous dénotons par t_i) multiplié par la probabilité de vendre z sièges ou plus dans cette classe, i.e.

$$EMSR_i(z) = t_i \cdot \bar{P}_i(z). \quad (3.4)$$

Soit maintenant s_j^i le nombre de sièges protégés de la classe j et accessibles seulement à la classe i . En supposant qu'il y ait k classes tarifaires, la valeur optimale du niveau de protection s_j^i en classe i doit satisfaire

$$EMSR_i(s_j^i) = t_j, \forall i < j, i \in \{1, 2, \dots, k\}. \quad (3.5)$$

Cela signifie que les sièges de revenu marginal espéré supérieur à t_j ne devraient pas être vendus en classe j mais plutôt dans une classe de niveau plus élevé. Le schéma d'allocation optimal est donc obtenu lorsque la contribution d'un siège supplémentaire vendu dans une classe tarifaire égale celle d'un autre siège vendu dans la classe suivante. On calcule alors la limite de réservation l_j en classe j comme

$$l_j = \max \left\{ 0, u - \sum_{i < j} s_j^i \right\}, \quad (3.6)$$

où u est la capacité totale (de l'appareil ou de la cabine économique). De plus, le niveau de protection emboîté de la classe j est

$$n_j = l_j - l_{j+1}, \quad j = 1, 2, \dots, k - 1. \quad (3.7)$$

Pour un appareil avec k classes tarifaires, on a alors la relation

$$u = \sum_{j < k} n_j + l_k. \quad (3.8)$$

On remarque que la classe de plus bas niveau (i.e. la $k^{\text{ième}}$) ne possède pas de niveau de protection, toutes les autres classes pouvant lui « voler » des sièges. Notons que cette présentation correspond à la variante « a » de la méthode EMSR dans laquelle le niveau de protection est établi pour chaque classe individuelle par rapport à la première classe de niveau inférieur. Il en existe une variante « b » dans laquelle le niveau est établi conjointement pour toutes les classes supérieures par rapport à la première classe de niveau inférieur. Considérons, à titre d'exemple, trois classes tarifaires, étiquetées Y, Q, L, en ordre décroissant de valeur. Dans la variante « a » de la méthode EMSR, le niveau de protection de la classe Y est établi par rapport à la classe

Q et le niveau de protection de la classe Q est établi par rapport à la classe L. Dans la variante «b», on établie toujours le niveau de protection de la classe Y par rapport à la classe Q, mais on établie ensuite de niveau de protection conjoint des classes Y et Q par rapport à la classe L.

Il est possible d'obtenir des conditions d'optimalité équivalentes pour les niveaux de protection en adoptant une approche basée sur la programmation mathématique. Soit $R : \mathbb{R}^k \mapsto \mathbb{R}$ la fonction de revenu espéré sur un segment j , i.e. $R(s)$ représente le revenu espéré lorsque $s = (s_1, s_2, \dots, s_k)$ sièges sont alloués sur ce segment. On peut alors définir le revenu marginal espéré d'un siège en classe i comme

$$EMSR_i = \frac{\partial R}{\partial s_i}, \quad (3.9)$$

i.e. le taux de variation de la fonction de revenu espéré R dans la direction correspondant à l'allocation de la classe i . Un schéma d'allocation optimal peut être obtenu en résolvant le programme

$$\max_s \left\{ R(s) \mid \sum_{i=1}^k s_i \leq u \right\}, \quad (3.10)$$

où u est encore une fois la capacité totale disponible. Soit ξ le multiplicateur de Lagrange associé à la contrainte de capacité. La condition d'optimalité du premier ordre stipule que

$$\frac{\partial R}{\partial s_i} = \xi, \quad i = 1, 2, \dots, k. \quad (3.11)$$

En d'autres mots, dans un schéma d'allocation optimal, le revenu marginal espéré par siège est le même pour chaque classe tarifaire et égal à ξ . On n'a donc pas intérêt à modifier l'allocation pour ajouter des sièges à une classe au dépens d'une autre.

L'avantage de la méthode EMSR est qu'elle est simple à implanter et produit des schémas d'allocation emboîtés. Cependant, il fut démontré par plusieurs auteurs (notamment par Curry en 1990 [Cur90]) que cette méthode est sous-optimale, sauf dans le cas de deux classes. Il s'agit donc d'une heuristique permettant d'approximer les schémas d'allocation optimaux.

Curry [Cur90] et Brumelle et McGill [BM93] proposent des conditions d'optimalité pour lesquelles, sous certaines conditions de continuité de la fonction de revenu espéré, un schéma d'allocation de sièges comporte des niveaux de protection optimaux. Les conditions d'optimalité de Brumelle et McGill prennent la forme

$$\begin{aligned} t_2 &= t_1 \cdot \Pr\{r_1 > s_1^*\} \\ t_3 &= t_2 \cdot \Pr\{(r_1 > s_1^*) \cap (r_1 + r_2 > s_2^*)\} \\ &\vdots \end{aligned}$$

$$t_{k+1} = t_k \cdot \Pr\{(r_1 > s_1^*) \cap (r_1 + r_2 > s_2^*) \cap \dots \cap (r_1 + r_2 + \dots + r_k > s_k^*)\},$$

où s_i^* est le niveau de protection optimal pour la classe i . On calcule alors la limite de réservation de la classe i comme $l_i = u - s_{i-1}^*$, où u est la capacité totale de l'appareil. Brumelle et McGill proposent une méthode exacte exploitant ces conditions d'optimalité.

Wollmer [Wol92] présente une formulation exacte permettant de déterminer les schémas d'allocation optimaux d'un segment unique dans un contexte statique. Il compare sa méthode avec la méthode EMSR de Belobaba à l'aide d'un programme de simulation. L'auteur affirme que les rendements générés par chaque méthode sont similaires mais que la méthode EMSR a tendance à surestimer le revenu espéré.

Robinson [Rob93] compare l'efficacité des méthodes exactes et approximées pour l'établissement de schémas d'allocation emboîtés. L'auteur propose de nouvelles conditions d'optimalité qui relaxent l'hypothèse d'arrivée séquentielle des requêtes de réservation, par ordre croissant de tarif. Robinson démontre que ces conditions d'optimalité permettent d'obtenir des schémas d'allocation à l'aide de méthodes de type Monte Carlo.

Van Ryzin et McGill [vRM00] présentent un algorithme adaptatif permettant de déterminer des niveaux de protection emboîtés respectant les conditions d'optimalité énoncées ci-dessus. Leur algorithme a la particularité de ne pas utiliser d'estimés de demande *a priori*, mais plutôt de s'ajuster en se basant sur des données historiques. Cette méthode requiert donc une période préalable de collecte de données couvrant un nombre de vols assez grand pour permettre à l'algorithme de bien fonctionner, ce qui limite ses applications.

Talluri et van Ryzin [TvR01] proposent un modèle de contrôle d'inventaire tenant compte explicitement du comportement des consommateurs. Ces auteurs tentent de modéliser les choix des consommateurs devant effectuer un achat parmi plusieurs produits tarifaires offerts simultanément de façon à déterminer quel ensemble de produits devrait être offert à un moment donné. Ils modélisent la probabilité d'achat de chaque produit tarifaire comme une fonction des produits offerts, les probabilités étant obtenues à l'aide d'un modèle LOGIT multinomial. Selon les auteurs, cette approche permet de modéliser les phénomènes d'*achat ascensionnel* et *descensionnel*.

Netessine et Shumsky [NS01] s'intéressent à un cas simple, soit deux compagnies offrant chacune un vol à segment unique sur un marché donné avec deux seules classes tarifaires emboîtées, en se

basant sur la théorie des jeux. Les auteurs énoncent entre autres des conditions sous lesquelles les compétiteurs peuvent atteindre un équilibre de Nash. En se basant sur des expériences numériques, ils comparent la disponibilité des sièges lorsqu'une compagnie aérienne se trouve en situation de monopole ou en compétition avec d'autres compagnies. En supposant que les prix demeurent fixes, leurs résultats suggèrent qu'une situation de monopole favorise les consommateurs ayant une propension à payer moindre, au dépens des consommateurs plus fortunés se dirigeant vers les produits de meilleure qualité, qui seront dans ce cas plus chers.

Méthodes dynamiques

Les méthodes dynamiques ne supposent pas que les clients effectuent leurs réservations dans un ordre quelconque et ne produisent pas de schéma d'allocation des sièges *a priori*. Ces méthodes contrôlent plutôt l'inventaire en décidant si chaque réservation devrait être acceptée ou rejetée. Cette décision se base sur l'état actuel de l'inventaire et l'espérance de revenu futur.

Lee et Hersh [LH93] présentent un modèle de programmation dynamique en temps discret où la demande pour chaque classe tarifaire est modélisée par un processus de Poisson. Un tel processus d'arrivée engendre un problème de décision markovien, les demandes de réservation effectuées avant le temps t n'affectant pas celles effectuées après le temps t . L'espace d'états de la chaîne de Markov ainsi engendrée est déterminé par le nombre de périodes avant le départ et la capacité toujours disponible, une transition ayant lieu à chaque nouvelle période ou lorsque la capacité diminue. Soit $R(c, n)$ le revenu total espéré n périodes avant le départ avec une capacité résiduelle de c unités. Selon le modèle, une requête de réservation en classe i se produisant n périodes avant le départ est acceptée si, et seulement si,

$$t_i \geq R(c, n-1) - R(c-1, n-1), \forall i \in I, \forall c \in \{1, 2, \dots, u\}, \forall n \in \{1, 2, \dots, N\}, \quad (3.12)$$

où I est l'ensemble des classes tarifaires sur ce segment, u la capacité de l'appareil et N le nombre de périodes considérées. La relation (3.12) stipule qu'une requête de réservation dans une certaine classe ne peut être acceptée que si le tarif de cette classe égale ou excède le coût d'opportunité du siège, défini comme la valeur marginale de ce siège à la période n . Les auteurs donnent une formulation récursive permettant de calculer dynamiquement les valeurs $R(c, n)$.

Lautenbacher, Shaler et Subramanian [SSL99] présentent une méthode dynamique s'appuyant sur celle de Lee et Hersh [LH93], en tenant toutefois compte des possibilités d'annulation et d'aphanie des passagers. Ce dernier article incorpore ainsi des éléments propres à la surréservation. Ils considèrent aussi les cas où le processus d'arrivée des clients est continu, formant ainsi

un cas limite du modèle discret.

Lautenbacher et Shaler [LS99] considèrent des vols à segment unique sur un seul marché où la demande est stochastique et plusieurs classes tarifaires emboîtées sont offertes. Ces auteurs s'intéressent à la fois aux cas dynamique et statique, la question étant de déterminer à tout moment si une réservation dans une classe particulière devrait être acceptée ou rejetée, de façon à maximiser le revenu global sur l'ensemble des classes tarifaires pour la durée de la période pendant laquelle les réservations sont acceptées. Chaque problème est modélisé à l'aide d'un processus de décision markovien qui engendre une équation récursive pouvant être résolue par des techniques de programmation dynamique. Les auteurs démontrent que les approches statique et dynamique ne sont en fait que des cas spéciaux d'un modèle markovien unifié.

Vulcano, van Ryzin et Maglaras [VvRM01] proposent un modèle d'enchère dynamique dans lequel les consommateurs, agissant rationnellement, compétitionnent entre eux pour la capacité disponible et soumettent des enchères au vendeur. Les auteurs comparent leur approche avec les méthodes statiques traditionnelles. Ils énumèrent les conditions sous lesquelles un mécanisme d'enchères optimales est préférable aux approches traditionnelles et appuient leurs résultats théoriques par des expériences numériques.

Feng et Xiao [FX01] suggèrent un modèle dynamique sur un sous-ensemble d'arcs d'un réseau radial. Précisément, ces auteurs considèrent des passagers en provenance de plusieurs origines, transitant à un moyeu et se dirigeant ensuite vers une seule destination. Leur modèle n'est donc pas à proprement parler un modèle par segment, puisqu'il est en mesure d'exploiter certains effets réseau engendrés par les connections au moyeu.

3.2.2 Formulations sur réseau

Il existe deux grandes classes de méthodes de contrôle d'inventaire sur réseau. La première regroupe les méthodes par segment «améliorées» afin qu'elles tiennent compte de certaines interactions entre les parties du réseau communes à différents itinéraires. Bien que sous-optimales et incapables d'identifier la contribution d'un itinéraire particulier aux revenus de l'ensemble du réseau, ces méthodes ont l'avantage d'être basées sur des approches par segments déjà largement utilisées. Leur développement et leur implantation nécessitent donc des investissements moindres que des méthodes basées sur des approches complètement nouvelles, ce qui fait que, se-

lon Belobaba [Bel98], plusieurs transporteurs utilisent toujours ce genre de méthodes «hybrides» situées à mi-chemin entre les méthodes par segments et les véritables méthodes d'optimisation de réseau.

Par exemple, l'utilisation de classes virtuelles est une heuristique développée afin de pouvoir utiliser des méthodes traditionnelles de contrôle d'inventaire par segment sur un réseau [Bel98, SLD92]. Chaque segment est divisé non pas en classes tarifaires mais en classes virtuelles qui correspondent à des intervalles de revenus sur l'ensemble du réseau. Sur un segment utilisé par plus d'un vol, les niveaux de protection sont établis par classe virtuelle, de sorte que les sièges qui rapportent le plus de revenu sur le réseau sont protégés de façon prioritaire. Ce type d'heuristique, essentiellement «vorace», ne considère que la contribution totale d'un vol aux revenus, et peut être sous-optimale, par exemple dans le cas d'un vol à deux segments, de A à B et B à C, où il est plus avantageux d'accepter deux passagers «locaux» (i.e. l'un allant de A à B et l'autre de B à C) payant chacun 600\$ qu'un seul passager allant de A à C et payant 1000\$. De plus, comme les niveaux de protection sont établis indépendamment pour chaque segment, cette méthode ne tient pas compte des interactions entre les flots de passagers transitant sur le réseau.

L'autre classe de méthodes d'allocation de la capacité sur réseau repose sur des modèles de programmation mathématique (linéaire ou en nombres entiers). Ces modèles tiennent explicitement compte de la structure de réseau et peuvent tirer avantage de la forte combinatoire d'un réseau aérien moderne.

Glover et al. [GGLM92] et Alstrup [ABMV96] proposent des modèles de flot à coût maximum (l'objectif étant de maximiser les revenus de la compagnie) qui considèrent la demande comme une quantité exogène et déterministe. Ce genre de modèle permet d'exploiter les effets réseau en ignorant toutefois les aspects stochastiques du problème. Soit F l'ensemble des vols opérés par un transporteur et $P(f)$ l'ensemble des produits tarifaires offerts sur le vol $f \in F$ ¹. Les variables de décision sont x_{fp} , soit le nombre de passagers (flot) achetant le produit tarifaire $p \in P(f)$ sur le vol $f \in F$. On suppose que le tarif t_{fp} du produit $p \in P(f)$ sur le vol $f \in F$ est connu, de même que la demande b_{fp} pour cette paire produit/vol. Les modèles de flot à coût

¹Nous utiliserons le terme «produit tarifaire» pour désigner soit la classe tarifaire, soit la base tarifaire, indépendamment du niveau de granularité des modèles auxquels nous nous intéressons. Les modèles de contrôle d'inventaire présentés dans cette section utilisent la classe tarifaire comme unité d'allocation de base. Par contre, les modèles que nous introduirons subséquentement sont plus précis et utilisent la base tarifaire comme unité d'allocation.

maximum prennent la forme

(MAXFLOW)

$$\begin{aligned}
 \max_x \quad & \sum_{f \in F} \sum_{p \in P(f)} t_{fp} x_{fp} \\
 \text{s.l.c.} \quad & \sum_{\substack{f|s \in S(f) \\ p \in P(f)}} x_{fp} \leq u_s \quad \forall s \in S & (\xi_s) \\
 & x_{fp} \leq b_{fp} \quad \forall f \in F, \forall p \in P(f) \\
 & x_{fp} \geq 0 \quad \forall f \in F, \forall p \in P(f),
 \end{aligned}$$

où S est l'ensemble des segments, u_s la capacité du segment $s \in S$ et $S(f)$ l'ensemble des segments utilisés par le vol $f \in F$.

Curry [Cur90] divise chaque *marché* en un ensemble de classes tarifaires emboîtées et formule un programme non linéaire permettant d'obtenir des limites de réservation pour chaque classe sur l'ensemble du réseau. L'auteur démontre que la fonction objectif de ce programme, laquelle exprime le revenu de la compagnie de façon déterministe selon la quantité de siège allouée à chaque classe, est concave. Il utilise une approximation linéaire de cette fonction afin de résoudre son modèle par programmation linéaire. Le modèle de Curry est statique et se base sur les mêmes hypothèses que les méthodes de segment statiques décrites ci-haut (i.e. emphanie des passagers, impossibilité d'annulation, réservations séquentielles selon le tarif moyen de chaque classe tarifaire, etc.).

D'autres auteurs intègrent certains aspects stochastiques à leurs modèles d'optimisation sur réseau. Par exemple, Wong, Koppelman et Daskin [WKD93] utilisent le *Expected Marginal Seat Revenue* comme coefficient de la variable de flot sur chaque segment dans la fonction objectif de leur modèle de flot à coût maximum. De Boer, Freling et Piersma [dBFP99] comparent les approches déterministes et stochastiques et présentent des résultats numériques.

Gallego et al. [GIPD04] s'intéressent à l'offre de produits flexibles (i.e. le consommateur effectue une réservation pour un sous-ensemble de produits et le vendeur s'engage à lui offrir un produit parmi ce sous-ensemble à une date ultérieure) dans un contexte dynamique. Ils proposent une approche basée sur la génération de colonnes afin de déterminer quels ensembles de produits flexibles devraient être offerts à un instant précis et utilisent un modèle de choix discret pour modéliser le comportement des consommateurs.

Bertsimas et Popescu [BP03] proposent de nouvelles politiques de contrôle d'inventaire basées

sur le principe de la programmation dynamique approximée. À chaque période de leur modèle, la fonction de coût est évaluée à l'aide d'un programme linéaire. Ces auteurs fournissent divers résultats théoriques et numériques sur leur approche.

Belobaba [Bel98] remarque que le principal avantage de tels modèles basés sur la programmation linéaire est qu'ils peuvent tenir compte des effets combinatoires et ainsi exploiter de façon optimale la capacité disponible sur l'ensemble du réseau. Par contre, cet auteur voit beaucoup d'inconvénients à ce type de méthode. D'abord, elles ne tiennent généralement pas compte des aspects stochastiques et dynamiques du problème. Bien que certains efforts aient été faits dans ce domaine, notamment à l'aide de modèles de programmation stochastique, un modèle d'optimisation mathématique incorporant complètement les aspects stochastiques et dynamiques du problème serait insoluble en pratique, comme le faisaient remarquer Smith et al. [SLD92]. Ensuite, les schémas d'allocation de sièges que produisent ces méthodes ne sont pas emboîtés, mais plutôt «partitionnés». Belobaba utilise cet adjectif pour illustrer le fait que les valeurs x_{fp} indiquent le nombre de sièges alloués exclusivement au produit tarifaire p sur le vol f . De telles allocations partitionnées sont le propre des modèles de programmation mathématique qui allouent des ressources limitées (dans ce cas-ci des sièges), la solution optimale étant un point extrême du domaine admissible. Certains travaux visant à transformer la solution d'une méthode de réseau en schéma d'allocation emboîté ont cependant été effectués récemment (voir par exemple [dBFP99] et [PP02]). Finalement, Belobaba affirme que la demande espérée pour une paire produit/vol (la quantité que nous dénotons par b_{fp} dans le modèle MAXFLOW) possède typiquement une moyenne inférieure à un mais une variance relativement grande. Il en résulte que la magnitude des variables x_{fp} est souvent très faible, ce qui fait qu'un certain produit tarifaire pourrait ne recevoir aucun siège dans une allocation, entraînant ainsi le *déversement* d'une partie de la demande.

Magré ces inconvénients, les modèles de programmation mathématique sur réseau trouvent une utilisation comme sous-routine dans certaines méthodes de prix d'offre.

Approche de «prix d'offre»

On définit généralement le prix d'offre (*bid price* dans la littérature anglophone) d'un siège sur un segment comme la valeur marginale de ce siège pour ce segment, c'est-à-dire le coût d'opportunité (possiblement espéré si le modèle incorpore les aspects stochastiques) engendré par la réservation ou la vente de ce siège sur ce segment. Le prix d'offre d'un siège sur un vol est la somme des prix

d'offre de ce siège sur tous les segments qui composent le vol. Le contrôle d'inventaire par prix d'offre consiste à accepter une réservation pour un siège (i.e. produit tarifaire) sur un vol si, et seulement si, le tarif de ce siège est égal ou supérieur à son prix d'offre. Le prix d'offre constitue ainsi le tarif minimum pour lequel une réservation devrait être acceptée.

L'avantage de cette approche est qu'elle remplace les schémas d'emboîtement complexes et les niveaux de protection multiples par une règle simple permettant d'accepter ou de rejeter une réservation. Le problème est de calculer les prix d'offre.

La méthode la plus couramment utilisée pour calculer les prix d'offres repose sur les formulations de programmations mathématiques sur réseau de type MAXFLOW, qu'elles soient déterministes ou stochastiques. Avec une telle formulation, le prix d'offre d'un siège sur un segment est la valeur de la variable duale associée à la contrainte de capacité sur ce segment (nous avons dénoté cette variable par ξ_s dans le modèle MAXFLOW). Bien entendu, si la contrainte de capacité n'est pas active pour un segment, le prix d'offre sur ce segment est nul, ce qui signifie qu'aucun coût d'opportunité n'est encouru pour offrir un siège supplémentaire sur ce segment.

Cette approche se basant sur la valeur des variables duales d'un programme linéaire déterministe fut étudiée de façon détaillée par Williamson [Wil92]. D'autres articles en discutent, par exemple [Tal01, PP02, Bel98, Vin95]. Talluri et van Ryzin [TvR99] incorporent les aspects stochastiques de la demande dans leur modèle de programmation mathématique sur réseau et proposent une procédure appelée *Randomized Linear Programming* permettant de résoudre ce modèle et d'obtenir les prix d'offre. Ces mêmes auteurs [TvR98] analysent de façon formelle l'approche de prix d'offre et démontrent que, de façon générale, les prix d'offre calculés à partir de la valeur des variables duales sont sous-optimaux. Les auteurs attribuent cette sous-optimalité à deux causes. D'abord, la vente d'un siège peut représenter un «grand» changement dans la capacité résiduelle qui outrepasserait le niveau d'information marginale que peut fournir une variable duale. Deuxièmement, les revenus futurs dépendent possiblement de la capacité résiduelle de façon non linéaire, ce qui fait que les prix d'offre-variables duales ne sont qu'une approximation des véritables coûts d'opportunité entraînés par la vente d'un siège supplémentaire. L'article de Talluri et van Ryzin donne aussi certaines conditions sous lesquelles les prix d'offres-variables duales peuvent asymptotiquement correspondre aux prix d'offre optimaux. Il faut entre autre que la capacité totale et le volume des ventes soient grands. Malgré leur sous-optimalité, les prix d'offres-variables duales demeurent une heuristique largement utilisée, notamment à cause de leur interprétation intuitive et de la facilité à résoudre les programmes linéaires sur lesquels ils sont basés.

3.3 Tarification

La majorité des articles traitent la tarification sous un angle propre aux sciences économiques, en analysant les effets des politiques de tarification sur l'ensemble de l'industrie aérienne. Un nombre plus restreint d'articles s'intéressent à la tarification d'un point de vue stratégique ou décisionnel, en utilisant des méthodes propres à la recherche opérationnelle. Parmi ces articles, on peut citer Gallego et van Ryzin [GvR94, GvR97] et You [You99]. Dans les deux cas, les auteurs utilisent un processus stochastique afin de modéliser la tarification de vols multi-segments comportant plusieurs classes tarifaires et proposent des approches de programmation dynamique, exactes et heuristiques, pour résoudre les équations de récurrence engendrées par leurs modèles. Ces auteurs soulignent les relations de dualité existant entre le contrôle d'inventaire et la tarification et affirment aussi que leurs modèles de tarification peuvent implicitement résoudre le problème d'allocation de la capacité. Ils soutiennent qu'imposer un prix très élevé pour un billet dans une classe tarifaire donnée équivaut en pratique à fermer cette classe (donc à en déterminer la capacité finale) et à déplacer la demande sur d'autres classes.

Afin d'illustrer cette approche, nous décrivons brièvement le modèle dynamique de Gallego et van Ryzin [GvR97]. Une firme produit n types de produits à partir de m types de ressources. La quantité initiale (i.e. au temps 0) de chaque ressource est donnée par le vecteur $x = (x^1, x^2, \dots, x^m) \in \mathbb{Z}_+^m$. La firme offre ses produits sur l'horizon temporel $[0, t]$, où $t > 0$. Aucune ressource supplémentaire ne peut être acquise sur cet intervalle. La production d'une unité du produit j requiert a_{ij} unités de la ressource i . Ces données sont contenues dans la matrice $A \in \mathbb{Z}_+^{m \times n}$ telle que $[A]_{ij} = a_{ij}$. Soit N_s^j le processus de comptage indiquant le nombre d'unités du produit j vendues sur l'intervalle $[0, s]$, où $0 \leq s \leq t$ et p_s^j le prix du produit j au temps s . Selon cette notation, $dN_s^j = 1 \cdot ds$ indique qu'une unité du produit j est vendue à l'instant s (sinon $dN_s^j = 0$). Pour alléger la notation, on forme les vecteurs $N_s = (N_s^1, N_s^2, \dots, N_s^n)$ et $p_s = (p_s^1, p_s^2, \dots, p_s^n)$. Au temps s , la demande pour chacun des produits est déterminée par des processus de Poisson d'intensité $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_n)$. Ce vecteur d'intensités dépend de l'instant et du prix en vigueur en cet instant. On suppose donc que le vecteur d'intensités est donné par une fonction $\lambda : \mathbb{R}^{n+1} \mapsto \mathbb{R}^n$, i.e. $\lambda = \lambda(p, s)$. Soit U l'ensemble des politiques tarifaires admissibles. Cet ensemble est entre autre déterminé par la quantité de ressources disponibles, i.e. $\int_0^t A dN_s \leq x$. L'objectif du modèle est de trouver, parmi U , la politique qui maximise le revenu espéré au temps t étant donné le vecteur de ressources initiales x , que l'on dénote $R(t, x)$.

On obtient ainsi

$$R_u(t, x) = E_u \left[\int_0^t p_s \cdot dN_s \right], \quad (3.13)$$

et la politique optimale u^* est

$$u^* = \arg \sup_{u \in U} R_u(t, x). \quad (3.14)$$

Remarquons que R satisfait les conditions limites

$$\begin{aligned} R_u(t, x) &= 0, \quad \forall t \text{ et } \forall x \text{ t.q. } \exists i, j : x^i < a_{ij}, \\ R_u(0, x) &= 0, \quad \forall x \text{ et } \forall j. \end{aligned}$$

Comme indiqué ci-dessus, ce modèle engendre, par ses conditions d'optimalité, des équations de récurrence qui sont résolues par des méthodes de programmation dynamique.

3.4 Tarification et allocation de la capacité conjointes

Il semble y avoir eu très peu de recherche sur des approches permettant de traiter simultanément et directement ces deux aspects (et non pas implicitement comme dans le cas des modèles de tarification mentionnés ci-haut). McGill et van Ryzin [MvR99] affirment avoir pris connaissance de quelques documents de travail et de thèses consacrés à ce sujet, mais aucun document n'a été publié à grande échelle.

Récemment, Kuyumcu et Garcia-Diaz [KGD00] proposaient un modèle de programmation en nombres entiers que les auteurs prétendent capable de résoudre conjointement les problèmes de tarification et d'allocation de la capacité. Cette affirmation nous paraît exagérée. Leur modèle permet de déterminer, parmi un large ensemble de produits tarifaires, quel sous-ensemble devrait être retenu ainsi que les quantités de sièges allouées à chaque classe. Les auteurs utilisent le *Expected Marginal Seat Revenue* comme coefficient dans leur fonction objectif. Afin de calculer cette quantité, ils supposent que le prix de chaque produit est connu et préalablement fixé, donc que la tarification proprement dite a déjà été effectuée. Cette approche permet une certaine flexibilité tarifaire seulement lorsque l'ensemble de produits admissibles est très élevé. Or le nombre de variables de décision dépend de la taille de cet ensemble. Les auteurs développent aussi des formulations alternatives et des méthodes de coupe permettant de réduire les temps de résolution de leurs modèles.

3.5 Autres aspects

Le reste de cette section présente brièvement certains articles traitant d'autres aspects du problème d'o.d.r. que nos travaux ne considèrent pas directement.

Prévision de la demande et modélisation du comportement de la clientèle

La prévision de la demande et la modélisation des consommateurs constituent le premier maillon de la chaîne en o.d.r., car les résultats de ce processus sont utilisés subséquemment par les méthodes de surréservation, d'allocation de la capacité et de tarification. C'est d'ailleurs sur cet aspect qu'ont porté les premiers articles scientifiques s'intéressant au processus de réservation et de vente dans le domaine aérien.

Beckman et Bobkoski [BB57], dans un article paru en 1957, tentent de modéliser le processus d'arrivée des passagers à l'aide de plusieurs lois statistiques et concluent que la loi Gamma permet de mieux approximer ce phénomène. Cependant, les données avec lesquelles les paramètres de leurs distributions ont été estimés datent du début des années 50, à l'époque où les transporteurs n'offraient qu'une seule classe tarifaire. Il y a donc lieu de se demander si ces résultats reflètent toujours adéquatement le comportement des consommateurs.

Plus récemment, Weatherford et al. [WBP93] ont présenté un modèle probabiliste d'arrivée et de comportement des passagers en se limitant à deux classes tarifaires avec possibilité d'achat ascensionnel lorsque la classe inférieure a été fermée. Soumis et Nagurney [SN93] considèrent un modèle d'équilibre qui, étant donné un horaire de vol et une loi de probabilité pour la demande, permet de prévoir la répartition des passagers dans un contexte multi-classe et multi-marché. Leur modèle tient compte des préférences personnelles des passagers en utilisant des «coefficients d'attraction» pour chaque vol. Les auteurs proposent un algorithme de type Gauss-Seidel afin de résoudre leur modèle et présentent des résultats obtenus à partir de données réelles.

Surréservation

La surréservation et l'allocation de la capacité sont liés et les auteurs qui s'intéressent à l'un de ces deux aspects ont souvent publié des articles se rapportant à l'autre. Parmi les travaux les plus souvent cités dans ce domaine, on retrouve notamment ceux de Rothstein [Rot71] et Alstrup et al. [ABMV96].

Applications à l'industrie aérienne

Nous citons trois articles qui traitent du développement et de l'implantation de méthodes d'o.d.r. chez des transporteurs aériens majeurs. D'abord, Smith, Leimkuhler et Darrow [SLD92] décrivent la façon dont American Airlines utilise l'o.d.r. pour la surréservation et le contrôle d'inventaire. Ce transporteur est un pionnier dans le domaine et plusieurs méthodes développées par ses chercheurs ont été reprises par d'autres compagnies. Anderson [And89] démontre comment le transporteur Scandinavian Airlines System (SAS) utilise la recherche opérationnelle, non seulement pour l'o.d.r. mais aussi pour résoudre d'autres problèmes, tels la construction de l'horaire, l'affectation d'appareils ou la prévision du nombre de repas. Le récent article de Barnhart, Belobaba et Odini [BBO03] illustre l'importance de la recherche opérationnelle en transport aérien en faisant une large place à l'o.d.r.

Aspects économiques, légaux et sociaux

Certains auteurs analysent l'o.d.r. en adoptant un point de vue autre que celui de la recherche opérationnelle et tentent d'en mesurer les effets économiques et sociaux. Par exemple, Botimer [Bot96] prétend que l'application de l'o.d.r. est bénéfique à la fois pour les transporteurs aériens et pour les consommateurs en général car elle assure des billets aux passagers accordant la plus grande valeur au voyage. Cette conclusion nous paraît toutefois discutable.

Weatherford et Pfeiffer [WP94] analysent les effets économiques des réservations à l'avance, notamment sur la stimulation de la demande. Gale [Gal93] s'intéresse aux phénomènes d'équilibre se produisant sur un marché où les produits sont différenciés par leur date d'achat.

Le livre de Yeoman et Ingold [YI97] contient notamment une section traitant des aspects légaux reliés à l'application de l'o.d.r.

Aspects divers

En plus de l'optimisation des revenus, les transporteurs aériens doivent faire face à d'autres problèmes opérationnels importants. Ces problèmes à première vue indépendants peuvent avoir un impact non négligeable sur les stratégies d'o.d.r. d'une compagnie aérienne. Afin d'illustrer ces impacts, nous citons trois articles s'intéressant à d'autres aspects de l'industrie aérienne et décrivons l'influence que chacun de ces aspects peut avoir sur les stratégies d'o.d.r.

Berge et Hopperstad [BH93] proposent des modèles et des algorithmes permettant d'affecter de

façon dynamique les appareils selon la demande. Actuellement, l'affectation d'appareils se fait longtemps à l'avance, dès que l'horaire des vols est connu, et les transporteurs ont très peu de latitude pour modifier cette affectation, leur marge de manœuvre diminuant à l'approche du départ. Ces auteurs énumèrent certaines conditions sous lesquelles il serait possible de modifier l'affectation de façon dynamique selon la demande espérée, permettant ainsi d'exploiter de façon optimale la flotte dont dispose le transporteur. L'une de ces conditions est l'adoption de familles d'appareils possédant une interface commune, de sorte qu'un même équipage puisse piloter différents types d'appareils, sans avoir besoin de formation supplémentaire. La polyvalence des pilotes est donc une condition essentielle à ce type d'assignation dynamique. Le problème d'affectation est étroitement lié au problème d'o.d.r. En effet, la capacité que doivent gérer les méthodes de contrôle d'inventaire sur un segment de vol donné dépend directement du type d'appareil assigné à ce segment. Idéalement, ces deux problèmes devraient être résolus de façon conjointe, en y ajoutant possiblement la conception d'itinéraires. Cependant, pour des raisons pratiques et théoriques, ces problèmes sont actuellement résolus séparément.

Hassin et Shy [HS00] s'intéressent au phénomène grandissant des *liaisons partagées*. En analysant de telles ententes entre des compagnies compétitionnant sur le même marché, ces auteurs démontrent qu'une liaison partagée permet d'augmenter les revenus de chaque compagnie participant à l'entente mais a des effets négatifs pour les consommateurs. Les liaisons partagées ont un impact significatif sur le processus d'o.d.r., notamment sur la tarification, puisque la répartition des revenus doit être calculée cas par cas, selon les ententes avec chaque compagnie.

Finalement, Berry, Carnall et Spiller [BCS97] s'intéressent aux aspects micro-économiques d'un réseau de type radial, en mesurant notamment les effets d'un tel réseau sur la demande de différents types de passagers et sur les structures tarifaires proposées par un transporteur. Le développement d'un réseau radial a un effet immédiat sur le processus d'o.d.r. d'une compagnie aérienne car un tel réseau engendre des problèmes d'optimisation beaucoup plus difficiles à résoudre en raison de la forte combinatoire et du nombre élevé d'itinéraires possibles dans un tel réseau.

Chapitre 4

Programmation bi-niveau

Nous présentons dans ce chapitre la programmation mathématique bi-niveau, l'approche sur laquelle nos modèles sont basés. Nous passons notamment en revue des articles généraux parus à ce sujet, de même que certains articles traitant de l'application de cette technique à des problèmes de tarification sur réseau.

4.1 Revue bibliographique

La première formulation du problème de programmation bi-niveau fut celle proposée par Bracken et McGill en 1973 [BM73]. Ce sont toutefois Candler et Norton [CN77] qui ont suggéré en 1977 l'appellation bi-niveau pour désigner cette classe de problèmes. Depuis le début des années 80, ce problème s'est vu accorder une attention plus particulière. De nombreux chercheurs, motivés par les relations entre la programmation bi-niveau et la théorie des jeux de Stackelberg, ont étudié ce problème afin d'en découvrir les propriétés théoriques et proposer des méthodes de résolution. La base de données bibliographique (en format BIBTEX) de Vicente et Calamai constitue une excellente source d'information. Pour une présentation plus concise de la programmation bi-niveau, on suggère les courts articles à ce sujet parus dans *The Encyclopedia of Optimization* : le premier, par Vicente [Vic01], traite d'aspects généraux tandis que les trois autres, par Marcotte et Savard, discutent respectivement des formulations [MS01c], applications [MS01b] et algorithmes [MS01a] de résolution de la programmation bi-niveau.

La programmation bi-niveau, un cas spécial de programmation multi-niveau, est appropriée pour modéliser un grand nombre de situations pratiques impliquant un processus de décision hiérarchique. Parmi les domaines d'application déjà étudiés, on retrouve :

1. La gestion

Coordination de firmes multi-divisionnelles ; localisation optimale de centres de production ; allocation de crédit.

2. La planification

Planification optimale de politiques énergétiques ou agricoles.

3. Le design

Problèmes de design optimal.

4. Le transport

Conception de réseaux ; estimation de la demande et de l'utilisation ; tarification optimale.

Nos travaux s'intéressent précisément à ce dernier champ d'application, c'est-à-dire la tarification optimale de services de transport. Quelques articles à ce sujet, par Brotcorne, Labbé, Marcotte et Savard, retiennent notre attention. Le premier [LMS99] présente un modèle générique de taxation basé sur la programmation bi-niveau, en explore les propriétés et présente certaines méthodes de résolution heuristiques, les méthodes exactes ne pouvant résoudre que de petites instances de ce problème. Le second [LMS98], tout en reprenant la présentation générique du problème, s'intéresse à une application de taxation optimale de réseau autoroutier. On y présente un modèle détaillé dont on démontre la NP-complétude, on y décrit des méthodes de résolution, et on y présente des résultats numériques. D'autres modèles et algorithmes sont présentés dans [BLMS00, BLMS01].

4.2 Principes de base

La programmation mathématique bi-niveau est la classe de problèmes d'optimisation mathématique où un sous-ensemble des variables de décision doit faire partie de l'ensemble des solutions optimales d'un autre problème d'optimisation paramétrisé par les variables du premier problème. Cette technique est appropriée pour modéliser les problèmes dans lesquels une partie des variables de décision n'est pas contrôlée par l'agent principal, appelé meneur, mais plutôt par un second agent, appelé suiveur, qui optimise son propre objectif. Les variables que le suiveur ne contrôle pas lui apparaissent alors comme des constantes, de sorte que sa stratégie optimale

dépend des décisions préalablement prises par le meneur.

Soient x et y les vecteurs de variables de décision contrôlées respectivement par le meneur et le suiveur. En supposant que l'objectif du meneur soit $F(x, y)$ et celui du suiveur $f(x, y)$, un problème bi-niveau peut s'exprimer comme

$$\begin{aligned} \min_{x,y} \quad & F(x, y) \\ \text{s.l.c.} \quad & (x, y) \in X \\ & y \in \arg \min_{y' \in Y(x)} f(x, y'), \end{aligned}$$

où X est l'ensemble des solutions réalisables du problème de premier niveau (problème du meneur) et $Y(x)$ celui du problème de deuxième niveau (problème du suiveur), qui dépend des décisions prises au premier niveau.

Comme ce document traite principalement de programmes bi-niveau, nous abuserons légèrement de la notation afin d'alléger la présentation des modèles subséquents. Ainsi, selon notre nouvelle convention notationnelle (parfois appelée «forme verticale»), le modèle ci-dessus s'exprime comme

$$\begin{aligned} \min_{x,y} \quad & F(x, y) \\ & (x, y) \in X \\ \min_y \quad & f(x, y) \\ & y \in Y(x). \end{aligned}$$

Lorsque le problème de deuxième niveau est convexe, on parle alors d'un programme mathématique bi-niveau convexe. L'avantage de ce type de programme est que le problème de deuxième niveau peut être remplacé (sous des conditions de régularité et de différentiabilité appropriées) dans la formulation globale par ses conditions d'optimalité de Karush-Kuhn-Tucker, qui sont alors nécessaires et suffisantes. Cependant, malgré l'appellation «convexe», le problème global peut avoir un domaine réalisable non convexe, non connexe ou même vide en présence de contraintes de premier niveau.

Dans les modèles bi-niveau que nous présenterons, l'objectif du premier niveau sera bilinéaire¹ tandis que les contraintes de premier niveau ainsi que le problème de deuxième niveau seront linéaires. On parlera donc de problèmes (bi-niveau) bilinéaires-linéaires. Comme le deuxième

¹Une fonction bilinéaire est un cas spécial de fonction quadratique de la forme $xQy + bx + cy$, c'est-à-dire une fonction linéaire en x pour tous les y et vice-versa.

niveau est linéaire (donc convexe), nous appliquerons la technique décrite ci-dessus et remplacerons ce problème par ses conditions d'optimalité de KKT. De plus, certaines manipulations algébriques nous permettront de transformer l'objectif bilinéaire en une fonction linéaire. À ce point, seules les contraintes de complémentarité découlant des conditions d'optimalité de KKT ne sont pas linéaires. Celles-ci sont alors remplacées par de nouvelles contraintes utilisant des variables binaires. Les problèmes résultants sont donc des programmes mathématiques linéaires mixtes, c'est-à-dire avec variables continues et variables entières (ou plus précisément binaires).

Nous pouvons illustrer cette démarche en présentant les modèles de tarification de réseau non capacitaire de Labbé, Marcotte et Savard (voir [LMS99, LMS98]). Ces travaux serviront de point de départ à notre approche. Nous supposons ici que nous sommes en présence d'un réseau orienté dont les arcs possèdent un coût fixe. Par exemple, dans un réseau autoroutier, le coût fixe d'un arc peut être sa longueur ou encore le temps requis pour le traverser (nous pouvons multiplier ces quantités par un facteur donné, s'exprimant par exemple en dollars par minute, afin d'obtenir des unités monétaires pour le coût fixe d'un arc). De plus, nous supposons qu'un sous-ensemble des arcs sont «taxables», c'est-à-dire que le meneur peut leur attribuer un coût supplémentaire. Toujours dans notre exemple de réseau autoroutier, cette taxe correspond à un péage que l'autorité responsable impose sur diverses sections du réseau. Le meneur (premier niveau) cherche bien entendu à maximiser ses revenus, tandis que les usagers du réseau (second niveau) désirent se rendre à destination en minimisant leur coût individuel, peu importe que ceux-ci soient fixes ou imposés par le meneur. Ainsi, en supposant que le flot sur les arcs taxables et non taxables est représenté respectivement par les vecteurs x et y , que le coût fixe des arcs taxables et non taxables est représenté respectivement par les vecteurs de constantes c et d et finalement que le vecteur t représente les taxes sur les arcs taxables, nous pouvons exprimer ce modèle comme

$$\begin{aligned} & \max_{t,x,y} && tx \\ & \min_{x,y} && (c+t)x + dy \\ \text{s.l.c.} &&& Ax + By = b \quad (\lambda) \\ &&& x, y \geq 0, \end{aligned}$$

où λ dénote le vecteur de variables duales associées aux contraintes d'égalité du problème de second niveau.

Les contraintes d'égalité de ce modèle assurent la satisfaction de la demande et la conservation du flot sur le réseau. Remarquons de plus que les arcs possèdent une capacité illimitée dans

ce modèle. Ainsi, pour un couple origine-destination donné, tous les usagers emprunteront le chemin ayant le coût total le moins élevé.

En remplaçant le problème de deuxième niveau par ses conditions d'optimalité de KKT, nous obtenons le programme à un seul niveau suivant :

$$\begin{aligned}
 \max_{t,x,y,\lambda} \quad & tx \\
 \text{s.l.c.} \quad & Ax + By = b \\
 & x, y \geq 0 \\
 & \lambda A \leq c + t \\
 & \lambda B \leq d \\
 & (c + t - \lambda A)x = 0 \\
 & (d - \lambda B)y = 0,
 \end{aligned}$$

où λ est le vecteur de variables duales associées aux contraintes de conservation de flot.

En se basant sur les contraintes de complémentarité et en effectuant quelques manipulations algébriques élémentaires, nous pouvons déduire que

$$\begin{aligned}
 tx &= (\lambda A - c)x \\
 &= \lambda(b - By) - cx \\
 &= \lambda b - (cx + dy).
 \end{aligned}$$

De plus, comme le meneur désire maximiser ses revenus, il est dans son intérêt d'imposer des taxes aussi élevées que possibles. Ainsi, en nous basant sur la première contrainte de complémentarité, nous pouvons déduire que $t_i = (\lambda A - c)_i$, lorsque $x_i > 0$, et lorsque $x_i = 0$, c'est-à-dire lorsque le flot est nul sur un arc taxable donné, le meneur peut imposer une taxe arbitraire supérieure ou égale à $(\lambda A - c)_i$, et en particulier la taxe $(\lambda A - c)_i$. Bref, nous pouvons poser que

$$t = \lambda A - c.$$

Ces observations nous permettent de récrire le modèle complet comme

$$\begin{aligned}
 \max_{x,y,\lambda} \quad & \lambda b - (cx + dy) \\
 \text{s.l.c.} \quad & Ax + By = b
 \end{aligned}$$

$$\begin{aligned}x, y &\geq 0 \\ \lambda B &\leq d \\ (d - \lambda B)y &= 0.\end{aligned}$$

S'il n'était de la contrainte de complémentarité restante, le dernier modèle serait linéaire. La présence de cette contrainte constitue d'ailleurs la principale difficulté algorithmique des modèles de tarification étudiés. Cependant, il est possible de transformer ce dernier modèle en un programme linéaire mixte, c'est-à-dire comportant des variables continues et des variables entières. En effet, soit $\{\lambda^l\}_{l \in L}$ l'ensemble des points extrêmes du polyèdre dual et $\{x^j, y^j\}_{j \in J}$ l'ensemble des points extrêmes du polyèdre primal. Nous pouvons alors remplacer la contrainte de complémentarité $(d - \lambda B)y = 0$ par l'ensemble de contraintes suivant :

$$\begin{aligned}Mw &\geq d - \lambda B \\ Nz &\geq y \\ w_i + z_i &\leq 1 \quad \forall i \\ w_i, z_i &\in \mathbb{B} \quad \forall i.\end{aligned}$$

où

$$M \geq \max_i \max_{l \in L} \{(d - \lambda^l B)_i\}$$

et

$$N \geq \max_i \max_{j \in J} \{y_i^j\}.$$

Le modèle final est alors

$$\begin{aligned}\max_{t, x, y, \lambda} & \lambda b - (cx + dy) \\ \text{s.l.c.} & Ax + By = b \\ & x, y \geq 0 \\ & \lambda B \leq d \\ & Mw \geq d - \lambda B \\ & Nz \geq y \\ & w_i + z_i \leq 1 \quad \forall i \\ & w_i, z_i \in \mathbb{B} \quad \forall i.\end{aligned}$$

Dans nos modèles, nous utiliserons cette démarche afin de convertir les contraintes de complémentarité en contraintes linéaires faisant appel à des variables binaires. Notons cependant

qu'en pratique, une telle démarche n'utiliserait pas deux mais bien un seul vecteur de variables binaires, c'est-à-dire que les nouvelles contraintes prendraient la forme

$$\begin{aligned} Mw &\geq d - \lambda B \\ N(1 - w) &\geq y \\ w_i &\in \mathbb{B} \quad \forall i. \end{aligned}$$

Cette formulation alternative, bien qu'équivalente au point de vue mathématique, permet possiblement de réduire le nombre d'opérations de branchement requises par un algorithme d'énumération implicite de type *Branch-and-Bound*, en plus de diminuer le nombre de variables binaires et de contraintes.

Chapitre 5

Modèle de tarification

Ce chapitre présente le modèle de programmation mathématique à deux niveaux qui servira de base à nos travaux. Ce modèle fut introduit dans [Côt01] et repris dans [CMS03]. En plus de présenter la formulation mathématique, on discute des hypothèses sur lesquelles le modèle s'appuie ainsi que de certains aspects reliés à son utilisation dans un contexte pratique.

5.1 Buts du modèle

Le modèle que nous proposons permet de résoudre de façon conjointe et simultanée les problèmes de tarification et de contrôle d'inventaire en optimisation des revenus dans le domaine aérien. Plus précisément, le modèle calcule les tarifs optimaux de produits tarifaires en prévoyant la réaction de passagers à ces tarifs, de sorte qu'il estime aussi le nombre de passagers (i.e. flot) pour chaque produit tarifaire sur chaque vol. Ces flots prévus peuvent être utilisés pour l'allocation de sièges aux classes tarifaires, mais le modèle peut aussi tenir compte d'une allocation effectuée préalablement, en utilisant des limites de réservation exogènes.

Le modèle permet d'adopter différentes stratégies de tarification, en se basant sur les concepts apparentés mais distincts de part de marché et de part de revenus. La part de marché d'un transporteur sur un marché donné est la fraction des passagers utilisant les vols de ce transporteur, alors que la part de revenus est la fraction des revenus totaux disponibles sur ce marché qui reviennent au transporteur. La compagnie utilisant notre approche peut imposer des ob-

jectifs de part de marché ou de part de revenus à atteindre sur certains marchés. Ces objectifs commerciaux se traduisent par des contraintes supplémentaires dans le modèle, de sorte que les structures tarifaires que celui-ci produit reflètent ces objectifs. Si la compagnie adopte une stratégie agressive et désire augmenter ses parts de marché ou de revenus, alors le modèle, prévoyant la réaction des consommateurs, déterminera probablement (mais pas nécessairement) des tarifs plus avantageux, alors qu'une stratégie plus conservatrice se traduira possiblement par des prix plus élevés.

5.2 Agents modélisés

Un modèle d'optimisation hiérarchique à deux niveaux comporte deux agents distincts, un meneur et un suiveur, aux objectifs souvent contradictoires.

Le meneur est la compagnie aérienne utilisant notre approche. L'objectif de la compagnie est de maximiser les revenus sur l'ensemble des vols qu'elle offre. Notre modèle ne tient pas compte des coûts encourus pour offrir ces vols ; nous supposons que ces coûts sont fixes, ce qui est à toutes fins pratiques le cas dans le domaine aérien.

Le meneur doit faire face à certaines contraintes. Nous avons déjà mentionné les objectifs de vente, i.e. de parts de marché ou de revenus. Il est aussi possible qu'une autorité, par exemple une agence gouvernementale, impose des limites sur les tarifs offerts pour un marché donné. Ces limites se traduisent par des bornes (supérieures ou inférieures) sur les variables de décision.

Le deuxième niveau du modèle représente l'ensemble des passagers. Ceux-ci désirent se rendre à destination en minimisant le coût perçu du voyage. Ce coût comprend non seulement le tarif mais aussi d'autres coûts (réels ou imaginaires) engendrés par le choix d'un vol plutôt qu'un autre. Bref, les passagers cherchent à minimiser la désutilité qu'ils encourent pour se rendre à destination, laquelle dépend de critères précis. Le modèle suppose que les passagers considèrent tous les vols, i.e. ceux offerts par le meneur et aussi ceux offerts par les compétiteurs du meneur présents sur chaque marché. Puisque les décisions des passagers sont indépendantes, le problème de deuxième niveau prend la forme d'un problème de plus court chemin dans le réseau induit par l'ensemble des vols du meneur et de ses compétiteurs.

La capacité des appareils et des différentes classes tarifaires doit être respectée. D'un point de

vue conceptuel, nous pouvons poser ces contraintes au premier ou au deuxième niveau. Les poser au deuxième niveau signifie que les passagers sont conscients des capacités et des limites de réservation au moment de choisir leur billet, alors que les poser au premier niveau signifie que les passager ne voient pas ces limites. Or un vol ne sera offert aux passagers que s'il possède une capacité résiduelle non nulle. Ainsi, même si les passagers ne sont pas conscients de contraintes de capacité présentes au premier niveau, ils doivent implicitement les respecter. Nous démontrerons qu'il est mathématiquement équivalent de poser ces contraintes au premier ou au deuxième niveau.

5.3 Structure de données

Le modèle que nous allons développer s'appuie implicitement sur une structure de données particulière. Celle-ci n'apparaît pas directement dans la formulation, mais son importance pour la compréhension du modèle justifie que l'on s'y attarde. Sa présentation permet aussi de formaliser certains concepts définis ci-haut. Il s'agit d'un réseau spatio-temporel, que l'on peut représenter dans la plan en agrégeant les trois dimensions de l'espace sur un axe, que nous appellerons axe λ (pour «lieu»), alors que l'autre axe, que nous appellerons axe τ (pour «temps») symbolise le temps. Ainsi, un point (λ, τ) dans ce système de coordonnées représente un lieu physique dans le temps.

Dans le cas qui nous intéresse, les abscisses λ sont en fait des villes. Un arc reliant les nœuds (λ^1, τ^1) et (λ^2, τ^2) représente donc un segment quittant la ville λ^1 à l'heure τ^1 et arrivant à la ville λ^2 à l'heure τ^2 (voir figure 5.1), tandis qu'un arc reliant les nœuds (λ^1, τ^1) et (λ^1, τ^2) représente une escale dans la ville λ^1 qui dure de τ^1 à τ^2 (voir figure 5.2).

Si on ajoute l'ensemble de tous les arcs de segment S et l'ensemble de tous les arcs d'escale E pour un ensemble de villes V à l'intérieur d'une fenêtre de temps donnée, on obtient un réseau $R = (V, S \cup E)$ orienté et acyclique¹. De là, on peut définir un vol comme un chemin se composant d'arcs de segment et d'arcs d'escale. De façon formelle, un vol f avec n segments et $n - 1$ escales est une séquence de nœuds (villes dans le temps)

$$f = \langle (\lambda^0, \tau^0), (\lambda^1, \tau^1), (\lambda^1, \tau^2), \dots, (\lambda^k, \tau^{2k-1}), (\lambda^k, \tau^{2k}), \dots, (\lambda^n, \tau^{2n-1}) \rangle$$

¹Cette propriété découle du fait qu'il est impossible de revenir en arrière sur l'axe τ , du moins avec les connaissances scientifiques en vigueur lors de la rédaction de cette thèse.

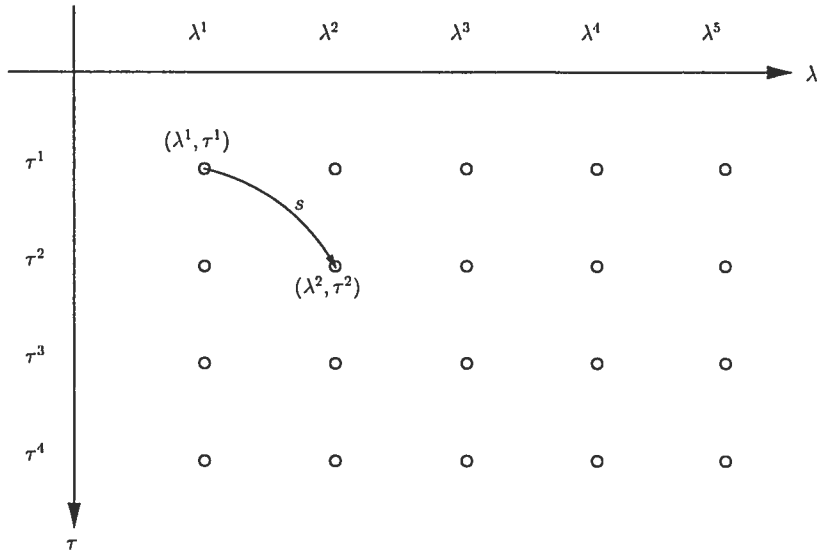


FIG. 5.1 – Un arc de segment $s \in S$ reliant les nœuds (λ^1, τ^1) et (λ^2, τ^2) .

reliées entre eux alternativement par des arcs de segment $s \in S$ et des arcs d'escale $e \in E$.

Chaque vol f est emprunté par un certain nombre de passager x_f . De plus, à chaque vol f est associée une durée \mathcal{D}_f , qui est la somme des durées des segments et escales composant ce vol, c'est-à-dire

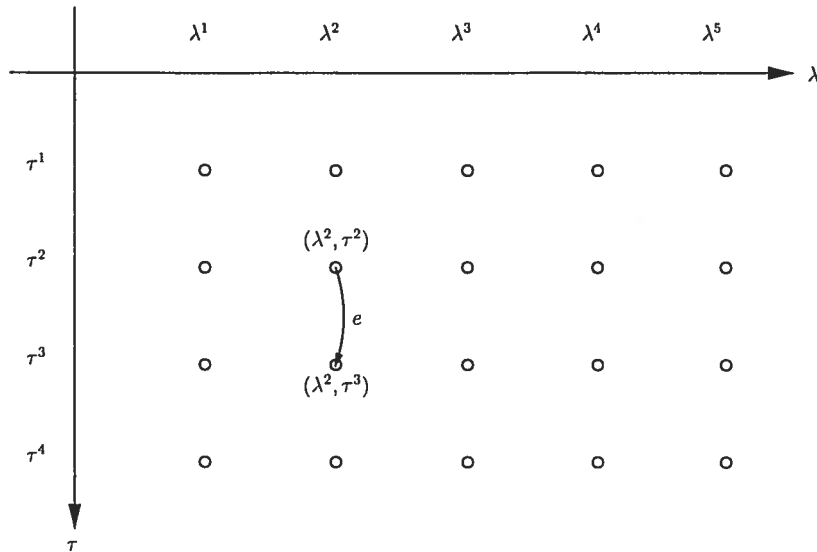
$$\mathcal{D}_f = \sum_{i=1}^{2n-1} (\tau^i - \tau^{i-1}) = \tau^{2n-1} - \tau^0,$$

où n représente le nombre de segments composant le vol. On désigne par F l'ensemble de tous les vols présents sur le réseau R et par $F(k)$ l'ensemble des vols couvrant le marché $k \in K$, K étant l'ensemble des marchés. Si deux vols $f_1 = \langle (\lambda_1^o, \tau_1^o), \dots, (\lambda_1^d, \tau_1^d) \rangle$ et $f_2 = \langle (\lambda_2^o, \tau_2^o), \dots, (\lambda_2^d, \tau_2^d) \rangle$ couvrent le même marché, alors $\lambda_1^o = \lambda_2^o$ et $\lambda_1^d = \lambda_2^d$, c'est-à-dire que les composantes spatiales de leur origine et destination sont respectivement les mêmes ; les fenêtres temporelles couvertes par ces vols peuvent cependant être différentes.

Remarquons que, de façon concrète, un arc de segment $s \in S$ est assuré par un aéronef possédant une capacité limitée dont nous devons tenir compte. Nous dénoterons la capacité du segment $s \in S$ par u_s . Ainsi, la somme du nombre de passagers sur les vols empruntant le segment s devra respecter cette capacité, c'est-à-dire

$$\sum_{f|s \in S(f)} x_f \leq u_s, \forall s \in S,$$

où $S(f)$ représente l'ensemble des segments composant le vol f .

FIG. 5.2 – Un arc d'escale $e \in E$ reliant les nœuds (λ^2, τ^2) et (λ^2, τ^3) .

5.3.1 Vols

Nos formulations sont basées sur la connaissance de l'ensemble F . Si F n'est pas connu, une étape préliminaire s'impose afin d'obtenir l'ensemble des vols, à partir d'un horaire énumérant tous les segments pour une période de temps donnée. La procédure suivante décrit la démarche à suivre.

Procédure CALCVOLS

Étant donné l'ensemble des segments S effectués à l'intérieur d'une fenêtre de temps donnée $]\tau^0, \tau^{\max}[$, ainsi que l'ensemble des marchés K :

Début

1. Créer l'ensemble F et l'initialiser $F \leftarrow \emptyset$.
2. Construire l'ensemble des arcs d'escales E : pour chaque arc de segment $s = ((\lambda^1, \tau^1), (\lambda^2, \tau^2)) \in S$, créer un nouvel arc d'escale $e = ((\lambda^2, \tau^2), (\lambda^2, \tau'))$, pour chaque $\tau' \geq \tau^2$ tel que $\tau' - \tau^2 \leq \Delta_e^{\max}$, où Δ_e^{\max} représente la durée maximale d'une escale.
3. Construire le réseau $R = (V, S \cup E)$, où V est l'union de l'ensemble des villes-origines et villes-destinations présentes dans K à l'intérieur de la fenêtre de temps $]\tau^0, \tau^{\max}[$.

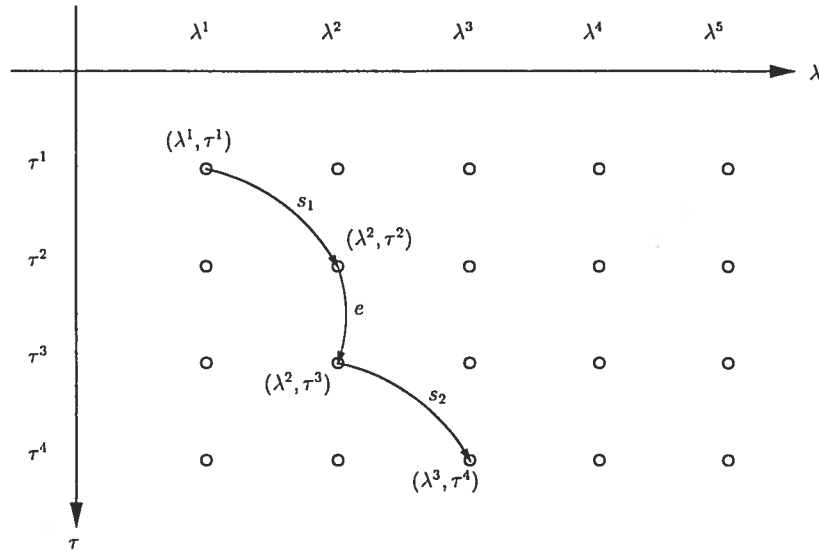


FIG. 5.3 – Un vol composé des segments s_1 et s_2 entrecoupés par l'escale e .

4. Pour chaque $k = (\lambda^{o(k)}, \lambda^{d(k)}) \in K$, c'est-à-dire pour chaque couple origine-destination :
 - (a) Créer l'ensemble $F(k)$ et l'initialiser $F(k) \leftarrow \emptyset$.
 - (b) Créer les nœuds temporaires $(\lambda^{o(k)}, \tau^0)$ et $(\lambda^{d(k)}, \tau^{\max})$, que nous appellerons respectivement super-origine et super-destination du marché k , dans le réseau R .
 - (c) Relier la super-origine $(\lambda^{o(k)}, \tau^0)$ à tous les nœuds $(\lambda^{o(k)}, \tau')$ où $\tau' \in]\tau^0, \tau^{\max}[$ par un arc artificiel et temporaire dans R .
 - (d) Relier tous les nœuds $(\lambda^{o(k)}, \tau'')$ à la super-destination $(\lambda^{d(k)}, \tau^{\max})$ où $\tau'' \in]\tau^0, \tau^{\max}[$ par un arc artificiel et temporaire dans R .
 - (e) Appliquer un algorithme de plus courts chemins entre la super-origine et la super-destination afin d'obtenir le 1^{ier}, 2^{ième}, ..., $n^{\text{ième}}$ plus court chemin dans R entre ces nœuds.
 - (f) Pour chaque chemin f obtenu à l'étape précédente, enlever le premier et le dernier arc de ce chemin (car ils sont artificiels). Enlever aussi l'avant dernier arc s'il s'agit d'un arc d'escale. Faire $F(k) \leftarrow F(k) \cup f$.
 - (g) Effacer tous les arcs et nœuds temporaires ajoutés à R lors des étapes précédentes.
 - (h) Faire $F \leftarrow F \cup F(k)$.
5. Retourner F , l'ensemble des vols.

Fin.

Cette procédure complète la définition formelle d'un vol, c'est-à-dire un chemin ou itinéraire dans le réseau spatio-temporel.

5.4 Hypothèses du modèle

Les principales hypothèses sur lesquelles s'appuie notre modèle sont :

- **Segmentation de la clientèle**

Hypothèse La clientèle est segmentée en **groupes d'utilisateurs**. Chacun de ces groupes accorde une valeur particulière aux critères dont dépend la désutilité d'un vol.

Justification Il est connu que la clientèle des transporteurs aériens est diversifiée. Traditionnellement, on divise la clientèle en deux grandes catégories, soit les touristes et les voyageurs d'affaires. Ces deux catégories peuvent elles-mêmes être sous-divisées en groupes aux caractéristiques propres¹. De plus, les individus de chacun de ces groupes possèdent une propension à payer différente pour les avantages et services auxquels sont associés les produits qu'ils achètent. Ces différences sont d'ailleurs exploitées depuis longtemps par les transporteurs aériens, qui imposent des «barrières» visant à prévenir l'achat de produits à rabais par les consommateurs les plus fortunés.

- **Comportement des consommateurs**

Hypothèse Au moment de l'achat, les consommateurs agissent de façon rationnelle et disposent d'information parfaite quant aux produits offerts par tous les transporteurs présents sur le marché.

Justification Le développement de nouveaux moyens de communication et de canaux de vente, en particulier Internet, permet aux voyageurs d'obtenir rapidement de l'information sur l'ensemble des produits et tarifs offerts par les transporteurs majeurs. Ces informations servent souvent de complément aux informations obtenues par les canaux traditionnels, tels les agences de voyages ou les services de vente des transporteurs eux-mêmes. Les voyageurs d'aujourd'hui ont ainsi accès à de l'information sans doute imparfaite mais néanmoins riche et continuellement remise à jour. Bien entendu, certains consommateurs, même s'ils disposent

¹Poussé à l'extrême, ce raisonnement suggère que chaque consommateur possède un comportement lui étant propre, i.e. compose son propre groupe, ce qui est sans doute le cas en réalité. Cependant il est impossible en pratique de cibler individuellement chaque consommateur et de mettre en marché un produit lui étant uniquement destiné. Le regroupement de consommateurs selon certains critères est donc inévitable.

d'information quasi-parfaite, vont prendre des décisions irrationnelles au moment de l'achat. Les raisons de ces décisions relèvent de la psychologie plutôt que de la recherche opérationnelle (voir annexe B). On peut cependant supposer que ces individus irrationnels sont en minorité et que la plupart des consommateurs, pour lesquels un voyage aérien représente une dépense considérable, vont choisir le produit maximisant leur utilité. Cette hypothèse de rationalité est en accord avec les arguments énoncés à l'annexe B.

- **Demande**

Hypothèse La demande est une quantité exogène, connue et déterministe sur chaque marché. Nous supposons de plus que la demande est connue pour chaque groupe d'usagers.

Justification La demande totale sur chaque marché suit des tendances historiques connues et est estimée de façon interne par le service de marketing de chaque transporteur. Ces informations sont donc disponibles. Quant à la fraction de la demande correspondant à chaque groupe d'usagers, elle peut être calculée en créant des «paniers» de produits tarifaires auxquels nous supposons qu'un seul groupe ait accès. Cette hypothèse permet de calculer la fraction de la demande pour chaque groupe à partir des données de vente historiques pour chacun de ces paniers.

- **Capacité de la compétition**

Hypothèse Les vols de la compétition possèdent une capacité illimitée.

Justification Cette hypothèse peut être justifiée à l'aide d'un argument économique. Le transport aérien est une industrie très compétitive, en particulier sur les marchés transfrontaliers et internationaux. Si, sur un certain marché, une partie de la demande demeure insatisfaite, il se trouvera alors un compétiteur pour offrir de nouveaux vols et saisir cette opportunité de revenu. Ce nouveau compétiteur devra cependant ajuster ses prix avec ceux déjà en vigueur sur le marché, de sorte que, la capacité de la compétition peut être augmentée pour satisfaire la demande, mais les prix proposés doivent demeurer sensiblement les mêmes. Cette hypothèse simplificatrice permet de diminuer considérablement la taille des problèmes et d'obtenir plus rapidement une solution optimale. Cependant, elle possède aussi ses inconvénients. Sur certains marchés, nous pouvons nous retrouver en présence d'une situation de monopole ou d'oligopole où l'offre n'arrive pas à satisfaire la demande. Les passagers entrent alors en compétition pour obtenir les billets disponibles, ce qui favorise les transporteurs et leur permet d'ajuster leurs prix à la hausse. Notre modèle ne peut pas tenir compte de ce scénario improbable quoique possible.

5.5 Formulation

5.5.1 Notation

Les symboles suivants dénotent des ensembles :

- K : ensemble des marchés (i.e. couples origine-destination)
- G : ensemble des groupes d'utilisateurs
- $P(k)$: ensemble des produits tarifaires offerts sur le marché $k \in K$
- $P(f)$: ensemble des produits tarifaires offerts sur le vol $f \in F$
- $F(k)$: ensemble des vols offerts sur le marché $k \in K$
- F : ensemble de tous les vols offerts, i.e. $F = \bigcup_{k \in K} F(k)$
- S : ensemble des segments de vol
- $S(f)$: ensemble des segments de vol composant le vol $f \in F$
- H : ensemble des classes tarifaires
- $H(f)$: ensemble des classes tarifaires offertes sur le vol $f \in F$

Afin d'associer les produits tarifaires à leur classe respective, nous utiliserons la fonction $h : P \mapsto H$. Ainsi, $h(p)$ représente la classe tarifaire à laquelle est associé le produit p ¹. Dans le même ordre d'idées, nous utiliserons parfois la notation $k(f)$ pour indiquer le marché sur lequel est exploité le vol f .

Lorsque nécessaire, nous utiliserons des exposants pour dénoter des ensembles propres au meneur (1) ou à la compétition (2). Par exemple, $P^1(k)$ représente l'ensemble des produits tarifaires offerts par le meneur sur le marché $k \in K$ et $P^2(k)$ celui de produits tarifaires offerts par la compétition sur ce même marché. Nous avons ainsi $P(k) = P^1(k) \cup P^2(k)$.

Les variables de décision du modèle sont :

- t_{fp}^1 : tarif du produit $p \in P^1$ sur le vol $f \in F^1$ du meneur (premier niveau)
- x_{fgp} : nombre de passagers du groupe $g \in G$ ayant acheté le produit $p \in P$ sur le vol $f \in F^1$ (deuxième niveau)
- y_{fgp} : nombre de passagers du groupe $g \in G$ ayant acheté le produit $p \in P$ sur le vol $f \in F^2$ (deuxième niveau)

¹Notons qu'en pratique une telle fonction est triviale car la classe tarifaire à laquelle est associé un produit correspond à la première lettre du *Fare Basis Code* de ce produit.

Le modèle utilise certaines données exogènes, que nous supposons connues et disponibles. Ces quantités sont :

t_{fp}^2 : tarif du produit $p \in P^2$ sur le vol $f \in F^2$ de la compétition

b_{gk} : demande (i.e. nombre de passagers) du groupe $g \in G$ sur le marché $k \in K$

r_{hf} : limite de réservation de la classe $h \in H$ sur le vol $f \in F$

u_s : capacité maximale du segment $s \in S$

\bar{t}_{fp}^1 : borne supérieure sur le tarif du produit $p \in P$ sur le vol $f \in F^1$ du meneur

\underline{t}_{fp}^1 : borne inférieure sur le tarif du produit $p \in P$ sur le vol $f \in F^1$ du meneur

Le modèle est aussi conditionné par un ensemble de paramètres externes. Afin de souligner l'importance de ces paramètres, ceux-ci seront dénotés par des lettres grecques en caractères gras.

Le premier sous-ensemble de paramètres permet de modéliser le comportement des consommateurs face aux différents critères d'achat ayant une influence sur l'utilité (ou la désutilité) qu'implique le choix d'un produit par rapport à un autre. Les valeurs de ces paramètres sont obtenues à l'aide de la procédure d'étalonnage décrite en annexe A. Dans un premier temps, nous nous intéresserons à deux critères d'achat auxquels correspondent deux vecteurs de paramètres. Le premier représente la valeur que les passagers accordent au temps, i.e. à la durée d'un vol et possiblement au nombre d'escales. Le second représente la valeur que les passagers accordent à la qualité de service associée à un produit tarifaire. Cette «qualité» est en fait une valeur numérique calculée à partir des avantages et restrictions associés à un produit. Ces vecteurs sont dénotés par :

α : valeur du temps des passagers

β : valeur de la qualité de service des passagers

Remarquons que ces paramètres sont estimés pour chaque groupe d'utilisateurs sur chaque marché. La taille des vecteurs précédents est donc $|G| \cdot |K|$. Lorsque nécessaire, nous utiliserons la notation α_{gk} pour dénoter la valeur d'un paramètre (dans ce cas-ci la valeur du temps) d'un groupe particulier $g \in G$ sur un marché $k \in K$ donné. Afin d'alléger la notation, nous utiliserons parfois le symbole $\theta = (\alpha, \beta)$, i.e. le «vecteur» dont les composantes sont les vecteurs de paramètres.

Le second sous-ensemble de paramètres utilisés par le modèle conditionne les résultats afin que ceux-ci reflètent les objectifs commerciaux de la compagnie. Comme décrit ci-dessus, ces objectifs prennent la forme de parts de marché ou de parts de revenus que la compagnie désire atteindre

et qui seront dénotées par :

$\bar{\sigma}$: bornes supérieures sur les parts de marché

$\underline{\sigma}$: bornes inférieures sur les parts de marché

$\bar{\rho}$: bornes supérieures sur les parts de revenu

$\underline{\rho}$: bornes inférieures sur les parts de revenu

Ces quantités sont des vecteurs de dimension $|K|$, i.e. les bornes sont propres à chaque marché.

5.5.2 Objectif du premier niveau

Le meneur maximise ses revenus sur l'ensemble des vols qu'il offre pour tous les produits et tous les groupes d'usagers :

$$\max_{t^1, x, y} \sum_{f \in F^1} \sum_{p \in P(f)} \sum_{g \in G} t_{fp}^1 x_{fgp}.$$

5.5.3 Contraintes du premier niveau

Parts de marché

Soit

$$V_k = \sum_{f \in F(k)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp}$$

le nombre total de passagers sur le marché $k \in K$. Les contraintes de parts de marché pour le meneur prennent la forme

$$\underline{\sigma}_k V_k \leq \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp} \leq \bar{\sigma}_k V_k, \forall k \in K.$$

Parts de revenus

Soit

$$R_k = \sum_{f \in F(k)} \sum_{p \in P(f)} \sum_{g \in G} t_{fp}^1 x_{fgp}$$

le revenu total sur le marché $k \in K$. Les contraintes de parts de revenu pour le meneur prennent

la forme

$$\underline{\rho}_k R_k \leq \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} t_{fp}^1 x_{fg} \leq \bar{\rho}_k R_k, \forall k \in K.$$

Bornes sur les tarifs

Pour différentes raisons, le meneur peut s'imposer ou se voir imposer des bornes supérieures ou inférieures sur ses tarifs. Ces contraintes prennent la forme

$$\underline{t}_{fp}^1 \leq t_{fp}^1 \leq \bar{t}_{fp}^1, \forall f \in F^1, \forall p \in P(f).$$

5.5.4 Objectif du second niveau

Les passagers, situés au deuxième niveau, désirent se rendre à destination en minimisant le coût perçu du voyage. Il s'agit en fait de la désutilité encourue par l'achat d'un billet par rapport à un autre. Ce coût est calculé à partir des caractéristiques du produit tarifaire qu'un passager achète et est propre à chaque groupe d'utilisateur. Le coût perçu reflète donc les valeurs que chaque groupe accorde aux différents critères d'achat.

Connaissant le vecteur de paramètres $\theta = (\alpha, \beta)$, le coût perçu du produit p sur le vol f pour un passager du groupe g est

$$\pi_{fgp}^i(\theta) = t_{fp}^i + \alpha_{gk} \mathcal{D}_f + \beta_{gk} \mathcal{Q}_p,$$

où $i \in \{1, 2\}$, k est le marché sur lequel est offert le vol f , \mathcal{D}_f la durée du vol f et \mathcal{Q}_p la qualité du produit p ¹.

L'objectif du second niveau prend ainsi la forme

$$\min_{x,y} \sum_{f \in F^1} \sum_{p \in P(f)} \sum_{g \in G} \pi_{fgp}^1(\theta) x_{fgp} + \sum_{f \in F^2} \sum_{p \in P(f)} \sum_{g \in G} \pi_{fgp}^2(\theta) y_{fgp}.$$

¹On peut constater que selon cette définition de la désutilité (que nous appelons «coût perçu» dans le modèle), la relation de préférence stricte pour les attributs «prix», «durée» et «qualité de service» est une relation d'ordre stricte partiel. Le théorème B.3 indique l'existence de fonctions d'utilité, que nous définissons respectivement comme t_{fp} , $\alpha_{gk} \mathcal{D}_f$ et $\beta_{gk} \mathcal{Q}_p$. On vérifiera aussi que la relation de préférence (\succsim) est une relation d'ordre faible et que comme les attributs t_{fp} , \mathcal{D}_f et \mathcal{Q}_p sont des scalaires, les conditions du théorème B.5 sont satisfaites. On peut donc conclure à l'additivité des fonction de désutilité pour obtenir l'expression du coût perçu ci-dessus.

5.5.5 Contraintes du second niveau

Capacité

Les passagers doivent respecter la capacité limite des appareils sur lesquels ils voyagent. Cette contrainte prend la forme

$$\sum_{f|s \in S(f)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp} \leq u_s, \forall s \in S^1.$$

Limites de réservation

La contrainte engendrée par les limites de réservation est

$$\sum_{p|h(p)=h} \sum_{g \in G} x_{fgp} \leq r_{hf}, \forall h \in H, \forall f \in F^1.$$

Dans le cas où les limites de réservations ont été préalablement fixées¹, il est possible d'utiliser ces limites directement dans le modèle. Bien entendu, on suppose que les limites de réservation respectent la capacité des appareils, de sorte que la contrainte de capacité est implicitement satisfaite et donc inutile.

Satisfaction de la demande

Les passagers désirent se rendre à destination, ce qui s'exprime comme

$$\sum_{f \in F^1(k)} \sum_{p \in P(f)} x_{fgp} + \sum_{f \in F^2(k)} \sum_{p \in P(f)} y_{fgp} = b_{gk}, \forall g \in G, \forall k \in K.$$

Non négativité des flots

Le nombre de passagers pour chaque combinaison vol-groupe-produit doit être non négatif, i.e.

$$x_{fgp} \geq 0, \forall f \in F^1, \forall p \in P(f), \forall g \in G$$

$$y_{fgp} \geq 0, \forall f \in F^2, \forall p \in P(f), \forall g \in G.$$

¹Dans ce cas, le modèle est beaucoup plus restreint quant aux schémas d'allocation qu'il peut produire. On l'utilise donc principalement pour résoudre le problème de tarification.

5.6 Modèle complet

Nous sommes maintenant en mesure de formuler le modèle, que nous dénoterons par BAFO (pour *Bilevel Airline Fare Optimization*) :

(BAFO)

$$\begin{aligned}
 & \max_{t^1, x, y} \sum_{f \in F^1} \sum_{p \in P(f)} \sum_{g \in G} t_{fp}^1 x_{fgp} \\
 \text{s.l.c.} \quad & \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp} \geq \underline{\sigma}_k V_k, \quad \forall k \in K \\
 & \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp} \leq \bar{\sigma}_k V_k, \quad \forall k \in K \\
 & \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} t_{fp} x_{fgp} \geq \underline{\rho}_k R_k, \quad \forall k \in K \\
 & \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} t_{fp} x_{fgp} \leq \bar{\rho}_k R_k, \quad \forall k \in K \\
 & t_{fp}^1 \geq \underline{t}_{fp}^1, \quad \forall f \in F^1, \forall p \in P(f) \\
 & t_{fp}^1 \leq \bar{t}_{fp}^1, \quad \forall f \in F^1, \forall p \in P(f)
 \end{aligned}$$

$$\begin{aligned}
 & \min_{x, y} \sum_{f \in F^1} \sum_{p \in P(f)} \sum_{g \in G} \pi_{fgp}^1(\theta) x_{fgp} + \sum_{f \in F^2} \sum_{p \in P(f)} \sum_{g \in G} \pi_{fgp}^2(\theta) y_{fgp} \\
 \text{s.l.c.} \quad & \sum_{f \in S(f)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp} \leq u_s, \quad \forall s \in S^1 \quad (\xi_s) \\
 & \sum_{f \in F^1(k)} \sum_{p \in P(f)} x_{fgp} + \sum_{f \in F^2(k)} \sum_{p \in P(f)} y_{fgp} = b_{gk}, \quad \forall g \in G, \forall k \in K \quad (\eta_{gk}) \\
 & x_{fgp} \geq 0, \quad \forall f \in F^1, \\
 & \quad \quad \quad \forall p \in P(f), \forall g \in G \\
 & y_{fgp} \geq 0, \quad \forall f \in F^2, \\
 & \quad \quad \quad \forall p \in P(f), \forall g \in G.
 \end{aligned}$$

où

$$V_k = \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} x_{fgp}$$

et

$$R_k = \sum_{f \in F^1(k)} \sum_{p \in P(f)} \sum_{g \in G} t_{fp}^1 x_{fgp}.$$

Les variables apparaissant entre parenthèses dans le modèle, soit (ξ_s) et (η_{gk}) , sont les variables duales associées respectivement aux contraintes de capacité et de faisabilité du réseau (satisfaction de la demande).

5.7 Transformations et reformulation

Nous reprenons le modèle (BAFO) présenté ci-dessus en enlevant toutefois les contraintes de parts de marché et de revenus ainsi que les bornes sur les tarifs. Ces contraintes, intéressantes en pratique, n'apportent rien du point de vue théorique et ne font qu'alourdir la présentation. Nous choisissons donc de les ignorer. Dans un premier temps, en plus du tarif lui-même, nous n'allons considérer qu'un seul critère d'achat, soit la durée du trajet. Les résultats théoriques présentés ci-dessous se généraliseront à un nombre arbitraire de critères. Comme la durée ne dépend que du vol, et non du produit, nous éliminons les références au produit tarifaire dans le modèle de base. Les variables de décision seront donc t_f^1 , x_{fg} et y_{fg} (l'indice p dénotant le produit disparaît du modèle). Aussi, α_{gk} dénote la valeur du temps pour les usagers du groupe g sur le marché k et le coût perçu du voyage prend la forme

$$\pi_{fg}^i(\theta) = t_f^i + \alpha_{gk} \mathcal{D}_f,$$

où $i \in \{1, 2\}$ et $\theta = (\alpha)$, i.e. le vecteur de paramètres comportementaux ne comporte qu'un seul élément, la valeur du temps. Le modèle que nous utiliserons comme point de départ est

(BAFO i)

$$\begin{aligned} & \max_{t^1, x, y} \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} t_f^1 x_{fg} \\ & \min_{x, y} \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} \pi_{fg}^1(\theta) x_{fg} + \sum_{k \in K} \sum_{f \in F^2(k)} \sum_{g \in G} \pi_{fg}^2(\theta) y_{fg} \\ & \text{s.l.c.} \quad \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg} \leq u_s, \quad \forall s \in S^1 \quad (\xi_s) \\ & \quad \sum_{f \in F^1(k)} x_{fg} + \sum_{f \in F^2(k)} y_{fg} = b_{gk}, \quad \forall g \in G, \forall k \in K \quad (\eta_{gk}) \\ & \quad x_{fg} \geq 0, \quad \forall f \in F^1, \forall g \in G \\ & \quad y_{fg} \geq 0, \quad \forall f \in F^2, \forall g \in G. \end{aligned}$$

Le dual du problème de second niveau est

$$\max_{\eta, \xi} \sum_{k \in K} \sum_{g \in G} b_{gk} \eta_{gk} + \sum_{s \in S^1} u_s \xi_s$$

$$\begin{aligned}
 \text{s.l.c. } \eta_{gk} + \sum_{s \in S(f)} \xi_s &\leq t_f^1 + \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^1(k), \quad (x_{fg}) \\
 &\forall g \in G \\
 \eta_{gk} &\leq t_f^2 + \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^2(k), \quad (y_{fg}) \\
 &\forall g \in G \\
 \xi_s &\leq 0, \quad \forall s \in S^1.
 \end{aligned}$$

Pour un vecteur de prix t donné, le problème de second niveau est linéaire en x et y . On peut donc lui substituer ses conditions d'optimalité primales-duales, qui sont nécessaires et suffisantes, dans la formulation globale afin d'obtenir le problème à un seul niveau suivant :

$$(\text{BAFO-NL}i) \tag{5.1}$$

$$\max_{t^1, x, y, \eta, \xi} \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} t_f^1 x_{fg} \tag{5.2}$$

$$\text{s.l.c. } \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg} \leq u_s, \quad \forall s \in S^1 \tag{5.3}$$

$$\sum_{f \in F^1(k)} x_{fg} + \sum_{f \in F^2(k)} y_{fg} = b_{gk}, \quad \forall g \in G, \forall k \in K \tag{5.4}$$

$$x_{fg} \geq 0, \quad \forall f \in F^1, \forall g \in G \tag{5.5}$$

$$y_{fg} \geq 0, \quad \forall f \in F^2, \forall g \in G. \tag{5.6}$$

$$\eta_{gk} + \sum_{s \in S(f)} \xi_s - t_f^1 \leq \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^1(k), \tag{5.7}$$

$$\forall g \in G \tag{5.8}$$

$$\eta_{gk} - t_f^2 \leq \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^2(k), \tag{5.9}$$

$$\forall g \in G \tag{5.10}$$

$$\xi_s \leq 0, \quad \forall s \in S^1 \tag{5.11}$$

$$-\xi_s (u_s - \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg}) = 0, \quad \forall s \in S^1 \tag{5.12}$$

$$(\alpha_{gk} \mathcal{D}_f + t_f^1 - \eta_{gk} - \sum_{s \in S(f)} \xi_s) x_{fg} = 0, \quad \forall k \in K, \forall f \in F^1(k), \tag{5.13}$$

$$\forall g \in G \tag{5.14}$$

$$(\alpha_{gk} \mathcal{D}_f + t_f^2 - \eta_{gk}) y_{fg} = 0, \quad \forall k \in K, \forall f \in F^2(k), \tag{5.15}$$

$$\forall g \in G. \tag{5.16}$$

De par la fonction objectif et les contraintes de complémentarité induites par les conditions d'optimalité primales-duales, ce programme est non linéaire. Cependant, des manipulations algébriques élémentaires combinées à l'introduction de variables binaires nous permettent de le reformuler en un programme linéaire mixte équivalent.

Remarquons d'abord que les contraintes de complémentarité sont de nature logique; elles imposent que le terme de droite ou le terme de gauche prenne une valeur nulle dans une solution optimale (il s'agit d'un «ou» non exclusif). Une telle situation est facilement modélisée à l'aide de variables binaires. Associons une variable $z_s \in \mathbb{B}$ à chaque contrainte (5.12), $v_{fg} \in \mathbb{B}$ à chaque contrainte (5.14) et $w_{fg} \in \mathbb{B}$ à chaque contrainte (5.14). Nous sommes alors en mesure de reformuler les contraintes de complémentarité comme

$$\begin{aligned}
 -\xi_s &\leq M_s z_s, & \forall s \in S^1 \\
 u_s - \sum_{f|s \in S(f)} \sum_{g \in G} v_{fg}^1 &\leq N_s(1 - z_s), & \forall s \in S^1 \\
 x_{fg} &\leq M_{fg}^1 v_{fg}, & \forall k \in K, \forall f \in F^1(k), \\
 & & \forall g \in G \\
 \alpha_{gk} \mathcal{D}_f + t_f^1 - \eta_{gk} - \sum_{s \in S(f)} \xi_s &\leq N_{fg}^1(1 - v_{fg}), & \forall k \in K, \forall f \in F^1(k), \\
 & & \forall g \in G \\
 y_{fg} &\leq M_{fg}^2 w_{fg}, & \forall k \in K, \forall f \in F^2(k), \\
 & & \forall g \in G \\
 \alpha_{gk} \mathcal{D}_f + t_f^2 - \eta_{gk} &\leq N_{fg}^2(1 - w_{fg}), & \forall k \in K, \forall f \in F^2(k), \\
 & & \forall g \in G \\
 z_s &\in \mathbb{B} & \forall s \in S^1 \\
 v_{fg} &\in \mathbb{B} & \forall f \in F^1, \forall g \in G \\
 w_{fg} &\in \mathbb{B} & \forall f \in F^1, \forall g \in G,
 \end{aligned}$$

où les M et N sont des scalaires de taille suffisante afin que les contraintes qui leurs sont associées soient satisfaites peu importe que la variable binaire correspondante prenne la valeur 0 ou 1, de façon à ne pas éliminer de solutions admissibles.

Comme nous l'avons mentionné, la fonction objectif du programme ci-dessus est non linéaire (elle est en fait bilinéaire). Le lemme suivant démontre qu'il est possible de reformuler l'objectif

en une fonction linéaire.

LEMME 5.1 *Reformulation de la fonction objectif*

$$\begin{aligned}
 \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} t_f^1 x_{fg} &= \sum_{k \in K} \sum_{g \in G} \eta_{gk} b_{gk} \\
 &\quad - \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} \alpha_{gk} \mathcal{D}_f x_{fg} \\
 &\quad - \sum_{k \in K} \sum_{f \in F^2(k)} \sum_{g \in G} (t_f^2 + \alpha_{gk} \mathcal{D}_f) y_{fg} \\
 &\quad + \sum_{s \in S^1} \xi_s u_s.
 \end{aligned}$$

Preuve

En partant de l'objectif bilinéaire de ce programme, on a

$$\begin{aligned}
 \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} t_f^1 x_{fg} &= \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} (\eta_{gk} x_{fg} - \alpha_{gk} \mathcal{D}_f x_{fg}) \\
 &\quad + \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} \sum_{s \in S(f)} \xi_s x_{fg} \quad (\text{par (5.14)}) \\
 &= \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} (\eta_{gk} y_{fg} - \alpha_{gk} \mathcal{D}_f x_{fg}) \\
 &\quad + \sum_{s \in S^1} \xi_s u_s \quad (\text{par (5.12)}) \\
 &= \sum_{k \in K} \sum_{g \in G} (\eta_{gk} (b_{gk} - \sum_{f \in F^2(k)} y_{fg})) \\
 &\quad - \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} \alpha_{gk} \mathcal{D}_f x_{fg} \\
 &\quad + \sum_{s \in S^1} \xi_s u_s \quad (\text{par (5.4)}) \\
 &= \sum_{k \in K} \sum_{g \in G} \eta_{gk} b_{gk} \\
 &\quad - \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} \alpha_{gk} \mathcal{D}_f x_{fg} \\
 &\quad - \sum_{k \in K} \sum_{f \in F^2(k)} \sum_{g \in G} (t_f^2 + \alpha_{gk} \mathcal{D}_f) y_{fg} \\
 &\quad + \sum_{s \in S^1} \xi_s u_s, \quad (\text{par (5.16)})
 \end{aligned}$$

la dernière expression étant linéaire. □

Afin d'alléger la présentation des modèles subséquents, nous posons

$$Z(x, y, \eta, \xi) = \sum_{k \in K} \sum_{g \in G} \eta_{gk} b_{gk} + \sum_{s \in S^1} \xi_s u_s$$

$$- \sum_{k \in K} \sum_{g \in G} \left(\sum_{f \in F^1(k)} \alpha_{gk} \mathcal{D}_f x_{fg} + \sum_{f \in F^2(k)} (t_f^2 + \alpha_{gk} \mathcal{D}_f) y_{fg} \right).$$

Ce résultat nous permet de récrire le modèle sous la forme d'un programme linéaire mixte, c'est-à-dire

(BAFO-MIP i)

$$\begin{aligned}
 & \max_{\substack{t_1, x, y, \\ \eta, \xi, z, v, w}} Z(x, y, \eta, \xi) \\
 & \text{s.l.c.} \quad \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg} \leq u_s, & \forall s \in S^1 \\
 & \sum_{f \in F^1(k)} x_{fg} + \sum_{f \in F^2(k)} y_{fg} = b_{gk}, & \forall g \in G, \forall k \in K \\
 & \quad x_{fg} \geq 0, & \forall f \in F^1, \forall g \in G \\
 & \quad y_{fg} \geq 0, & \forall f \in F^2, \forall g \in G \\
 & \quad \eta_{gk} + \sum_{s \in S(f)} \xi_s - t_f^1 \leq \alpha_{gk} \mathcal{D}_f, & \forall k \in K, \forall f \in F^1(k), \\
 & & \forall g \in G \\
 & \quad \eta_{gk} - t_f^2 \leq \alpha_{gk} \mathcal{D}_f, & \forall k \in K, \forall f \in F^2(k), \\
 & & \forall g \in G \\
 & \quad \xi_s \leq 0, & \forall s \in S^1 \\
 & \quad -\xi_s \leq M_s z_s, & \forall s \in S^1 \\
 & \quad u_s - \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg} \leq N_s(1 - z_s), & \forall s \in S^1 \\
 & \quad x_{fg} \leq M_{fg}^1 v_{fg}, & \forall k \in K, \forall f \in F^1(k), \\
 & & \forall g \in G \\
 & \quad \alpha_{gk} \mathcal{D}_f + t_f^1 - \eta_{gk} - \sum_{s \in S(f)} \xi_s \leq N_{fg}^1(1 - v_{fg}), & \forall k \in K, \forall f \in F^1(k), \\
 & & \forall g \in G \\
 & \quad y_{fg} \leq M_{fg}^2 w_{fg}, & \forall k \in K, \forall f \in F^2(k), \\
 & & \forall g \in G \\
 & \quad \alpha_{gk} \mathcal{D}_f + t_f^2 - \eta_{gk} \leq N_{fg}^2(1 - w_{fg}), & \forall k \in K, \forall f \in F^2(k), \\
 & & \forall g \in G \\
 & \quad z_s \in \mathbb{B}, & \forall s \in S^1 \\
 & \quad v_{fg} \in \mathbb{B}, & \forall f \in F^1, \forall g \in G
 \end{aligned}$$

$$w_{fg} \in \mathbb{B}, \quad \forall f \in F^2, \forall g \in G.$$

5.7.1 Contraintes de capacité

Dans le modèle initial, nous avons choisi de poser les contraintes de capacité sur les segments au deuxième niveau. Conceptuellement, cela signifie que les passagers en sont conscients, i.e. qu'ils connaissent explicitement le nombre de sièges disponibles sur chaque segment de vol. Bien que cette information soit généralement publique (les passagers peuvent consulter un site spécialisé afin de connaître le type d'appareil desservant leur vol et le nombre de places que contient celui-ci), la très grande majorité des passagers n'en sont pas conscients. Ainsi, du point de vue de la modélisation, nous pouvons placer les contraintes de capacité au premier niveau, où elles sont invisibles aux yeux des passagers et influencent seulement les décisions du meneur. Le prochain théorème démontre qu'il est aussi mathématiquement équivalent de ramener ces contraintes au premier niveau.

THÉORÈME 5.2 Valeur des multiplicateurs associés aux contraintes de capacité

Il existe une solution optimale du modèle (BAFO-NL i) telle que $\xi_s = 0$, pour tout segment $s \in S^1$.

Preuve

Soit $(\hat{t}, \hat{x}, \hat{y}, \hat{\eta}, \hat{\xi})$ une solution optimale de (BAFO-NL i) (on supposera que ces symboles sont des collections de variables, i.e. des vecteurs, et que les indices appropriés sont sous-entendus). Supposons que $\hat{\xi} \neq 0$, car sinon il n'y a rien à démontrer. Nous construisons la solution alternative $(\hat{t}, \hat{x}, \hat{y}, \hat{\eta}, \hat{\xi})$ telle que

$$\hat{t}_f^1 = \hat{t}_f^1 - \sum_{s \in S(f)} \hat{\xi}_s \geq \hat{t}_f^1, \quad \forall f \in F^1$$

et

$$\hat{\xi}_s = 0, \quad \forall s \in S^1.$$

Cette nouvelle solution est admissible pour le problème. En effet,

$$\hat{\eta}_{gk} + \sum_{s \in S(f)} \hat{\xi}_s - \hat{t}_f^1 = \hat{\eta}_{gk} + 0 + \sum_{s \in S(f)} \hat{\xi}_s - \hat{t}_f^1 \leq \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^1(k) \forall g \in G.$$

La contrainte (5.8) est donc respectée. Aussi,

$$\alpha_{gk} \mathcal{D}_f + \hat{t}_f^1 - \hat{\eta}_{gk} - \sum_{s \in S(f)} \hat{\xi}_s = \alpha_{gk} \mathcal{D}_f + \hat{t}_f^1 - \hat{\eta}_{gk} - \sum_{s \in S(f)} \hat{\xi}_s - 0, \quad \forall k \in K, \forall f \in F^1(k) \forall g \in G.$$

lequel est équivalent au programme bi-niveau

(BAFOii)

$$\begin{aligned}
 & \max_{t,x,y} \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} t_f^1 x_{fg} \\
 & \text{s.l.c.} \quad \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg} \leq u_s, \quad \forall s \in S^1 \\
 & \min_{x,y} \sum_{k \in K} \sum_{f \in F^1(k)} \sum_{g \in G} \pi_{fg}^1(\theta) x_{fg} + \sum_{k \in K} \sum_{f \in F^2(k)} \sum_{g \in G} \pi_{fg}^2(\theta) y_{fg} \\
 & \text{s.l.c.} \quad \sum_{f \in F^1(k)} x_{fg} + \sum_{f \in F^2(k)} y_{fg} = b_{gk}, \quad \forall g \in G, \forall k \in K \quad (\eta_{gk}) \\
 & \quad \quad \quad x_{fg} \geq 0, \quad \forall f \in F^1, \forall g \in G \\
 & \quad \quad \quad y_{fg} \geq 0, \quad \forall f \in F^2, \forall g \in G.
 \end{aligned}$$

□

En utilisant encore une fois des variables binaires pour récrire les contraintes de complémentarité sous forme linéaire, nous obtenons le modèle suivant :

(BAFO-MIPii)

$$\begin{aligned}
 & \max_{t,x,y,\eta,v,w} Z(t, \eta, x, y) \\
 & \text{s.l.c.} \quad \sum_{f|s \in S(f)} \sum_{g \in G} x_{fg} \leq u_s, \quad \forall s \in S^1 \\
 & \quad \quad \quad \sum_{f \in F^1(k)} x_{fg} + \sum_{f \in F^2(k)} y_{fg} = b_{gk}, \quad \forall g \in G, \forall k \in K \\
 & \quad \quad \quad x_{fg} \geq 0, \quad \forall f \in F^1, \forall g \in G \\
 & \quad \quad \quad y_{fg} \geq 0, \quad \forall f \in F^2, \forall g \in G \\
 & \quad \quad \quad \eta_{gk} - t_f^1 \leq \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^1(k), \\
 & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \forall g \in G \\
 & \quad \quad \quad \eta_{gk} - t_f^2 \leq \alpha_{gk} \mathcal{D}_f, \quad \forall k \in K, \forall f \in F^2(k), \\
 & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \forall g \in G \\
 & \quad \quad \quad x_{fg} \leq M_{fg}^1 v_{fg}, \quad \forall k \in K, \forall f \in F^1(k), \\
 & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \forall g \in G \\
 & \quad \quad \quad \alpha_{gk} \mathcal{D}_f + t_f^1 - \eta_{gk} \leq N_{fg}^1 (1 - v_{fg}), \quad \forall k \in K, \forall f \in F^1(k), \\
 & \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \forall g \in G \\
 & \quad \quad \quad y_{fg} \leq M_{fg}^2 w_{fg}, \quad \forall k \in K, \forall f \in F^2(k),
 \end{aligned}$$

$$\begin{aligned}
& \forall g \in G \\
& \alpha_{gk} \mathcal{D}_f + t_f^2 - \eta_{gk} \leq N_{fg}^2 (1 - w_{fg}), \quad \forall k \in K, \forall f \in F^2(k), \\
& \forall g \in G \\
& v_{fg} \in \mathbb{B}, \quad \forall f \in F^1, \forall g \in G \\
& w_{fg} \in \mathbb{B}, \quad \forall f \in F^2, \forall g \in G.
\end{aligned}$$

où

$$Z(t, \eta, x, y) = \sum_{k \in K} \sum_{g \in G} \eta_{gk} b_{gk} - \sum_{k \in K} \sum_{g \in G} \left(\sum_{f \in F^1(k)} \alpha_{gk} \mathcal{D}_f x_{fg} + \sum_{f \in F^2(k)} (t_f^2 + \alpha_{gk} \mathcal{D}_f y_{fg}) \right).$$

5.8 Autres modèles de second niveau

Le modèle de second niveau vise à décrire le comportement d'achat des consommateurs de façon à ce que le meneur puisse prévoir leurs décisions et intégrer au processus de tarification la réaction qu'ils auront face à ses propres tarifs et ceux de la compétition. Dans le cadre de cette thèse, nous avons utilisé un modèle de comportement linéaire au second niveau. Une telle formulation est simple, intuitive et peut, grâce à ses propriétés de convexité et de différentiabilité, être remplacée dans la formulation globale par ses conditions d'optimalité primales-duales, obtenant ainsi un modèle à un seul niveau. Le principal inconvénient de cette approche de modélisation réside dans le caractère extrémal des solutions qu'elle produit. Dans le contexte du transport aérien, l'extrémalité des solutions est atténuée par la forte dimensionnalité du problème, elle-même engendrée par la multitude de vols, de produits et de tarifs offerts. Cependant, dans d'autres domaines d'application où cette variété de l'offre est absente (par exemple dans l'industrie hôtelière), un modèle de comportement linéaire peut produire des solutions d'un réalisme moindre.

Un élément de modélisation de plus en plus important en transport aérien est l'inertie de la clientèle, c'est-à-dire l'apparente irrationalité dans les décisions d'achat des consommateurs membres de programmes de fidélisation offerts par les compagnies aériennes. Certains passagers choisiront explicitement un produit plus coûteux s'il leur donne droit à des primes ou leur permet d'accumuler des crédits subséquentement échangeables contre d'autres titres de voyage sur le même transporteur. Même si ces décisions peuvent paraître irrationnelles dans les données de vente que maintiennent les transporteurs (qui, pour des raisons techniques, n'indiquent pas encore

l'adhérence d'un passager à un programme donné), elle sont réfléchies. En effet, pour ces passagers, un transfert de transporteur engendre un coût d'opportunité supplémentaire se traduisant par la perte d'une prime ou de crédits de voyage. Nous suggérons d'intégrer cet aspect à l'approche de tarification en utilisant un modèle de comportement quadratique au second niveau. Plus précisément, nous reprenons le modèle linéaire présenté ci-dessus en y ajoutant un terme de pénalité quadratique qui représente de coût (désutilité) supplémentaire engendré par le transfert de transporteur pour certains passagers.

Soit \bar{x}_{fgp} le flot moyen en usagers du groupe g observé historiquement sur le vol f et le produit p (il s'agit du niveau d'information le plus désagrégé; on peut imaginer en pratique une approche analogue à un niveau d'agrégation plus élevé). Nous ajoutons à l'objectif du problème de second niveau le terme

$$\psi(x) = \sum_{k \in K} \sum_{g \in G} \gamma_{kg} \sum_{f \in F^1(k)} \sum_{p \in P(f)} \frac{1}{2} \|x_{fgp} - \bar{x}_{fgp}\|^2.$$

Le poids du coût de transfert quadratique dans la composition de la désutilité totale dépend du facteur γ_{kg} . La valeur de ce paramètre doit être estimée à partir de données de ventes historiques, par exemple en utilisant une procédure similaire à celle décrite en annexe A.

Notons que

$$\nabla \psi(x) = \sum_{k \in K} \sum_{g \in G} \gamma_{kg} \sum_{f \in F^1(k)} \sum_{p \in P(f)} (x_{fgp} - \bar{x}_{fgp}).$$

Nous sommes donc toujours en mesure de remplacer le problème de second niveau par ses conditions d'optimalité pour ultiment obtenir un problème linéaire en nombres entiers mixte, comme l'avons fait auparavant. Toutefois, la présence du terme quadratique a pour effet de lier les sous-problèmes primal et dual dans la formulation globale, ce qui complique les choses du point de vue algorithmique.

Selon le domaine d'application, on peut aussi envisager un modèle de comportement qui ne soit pas directement basé sur la programmation mathématique. Par exemple, la famille de modèles de choix discrets de type LOGIT est couramment utilisée en sciences économiques et en modélisation des systèmes de transport. En s'appuyant sur des hypothèse probabilistes (concernant les propriétés de la loi statistique Gumbel), un modèle LOGIT décrit, sous forme analytique, le comportement d'agents économiques placés devant un ensemble d'alternatives aux attributs distincts. Bien qu'il soient intéressants au point de vue théorique, le principal défi relatif à l'intégration de tels modèles non linéaires dans notre approche de tarification est le développement d'algorithmes spécialisés permettant de les résoudre.

5.9 Utilisation pratique

La relative complexité (en termes d'indices, d'ensembles considérés, du nombre de variables de décision, etc.) du modèle BAFO s'explique par le fait que celui-ci traite un problème bien réel (et très complexe) auquel doivent quotidiennement faire face les transporteurs aériens. Cependant, l'utilisation pratique d'un tel modèle soulève quelques questions importantes auxquelles nous tentons ici de répondre.

5.9.1 Contexte d'utilisation

Étant donné l'état actuel du marché (horaires, estimés de demande, tarifs de la compétition), le modèle permet de résoudre le problème de tarification en produisant des prix pour chaque combinaison vol/produit et fournit aussi des estimés de flot qui peuvent être utilisés pour la gestion de l'inventaire à court terme. En pratique, ces deux problèmes sont traités par des employés (voire des départements) différents chez les grands transporteurs. Au-delà de la complexité théorique de ces deux problèmes et des difficultés algorithmiques découlant de leur résolution conjointe, il existe dans le domaine aérien des contraintes organisationnelles qui rendent difficile, du moins dans un avenir rapproché, l'utilisation d'une telle approche et l'exploitation de toutes ses caractéristiques. Le problème de gestion d'inventaire s'est vu accorder au cours des dernières décennies beaucoup plus d'intérêt que le problème de tarification, de sorte qu'aujourd'hui, la majorité (sinon la totalité) des grands transporteurs internationaux utilisent des systèmes automatisés d'allocation de la capacité, bien que les approches les plus communes en pratique produisent des résultats sous-optimaux. Les efforts actuels se concentrent sur le problème de tarification. Les analystes responsables de la tarification prennent des décisions en se basant avant tout sur leur expérience et leur intuition ; peu de transporteurs disposent d'outils informatisés d'aide à la décision dans ce domaine. Pour ces raisons, le modèle que nous avons présenté est avant tout destiné à résoudre le problème de tarification dans un contexte tactique sur un horizon allant de quelques heures à quelques jours. Nous présenterons au chapitre 7 un cadre dynamique de tarification et d'allocation de la capacité conjoints dans lequel notre modèle de tarification bi-niveau joue un rôle de sous-routine. Cette approche combinée nous permettra de véritablement résoudre simultanément les problèmes de tarification et d'allocation de la capacité en transport aérien.

5.9.2 Disponibilité des données

Notre modèle utilise une quantité importante de données qui doivent être assemblées, formatées et interprétées avant d'être utilisées. La disponibilité de toute cette information est une condition essentielle à l'application pratique du modèle. Nous dressons ici la liste des différents types de données et décrivons les sources desquelles ces données peuvent être extraites.

- **Horaires des vols**

Les horaires des vols, ou plus précisément des segments de vol, sont connus et publiés de façon électronique par des entreprises privées spécialisées dans le domaine. Puisque notre modèle utilise le «vol» (i.e. une séquence de segments de vol ou encore un itinéraire dans le réseau) comme unité de base, les vols doivent être calculés à partir des horaires. La procédure «CALCVOLS» décrite ci-dessus peut être utilisée dans ce contexte.

- **Estimés de demande**

La prévision de la demande est à la base des processus d'optimisation des revenus et de marketing dans l'industrie aérienne. Les transporteurs ont mis en place des services internes dont la tâche est précisément d'établir des prévisions de demande. Ces services disposent aujourd'hui de logiciels statistiques sophistiqués pour les assister dans leur tâche. La précision de ces outils serait de 10 à 12 %.

- **Prix de la compétition**

Notre approche suppose que les prix des tarifs de la compétition sont connus et remis à jour régulièrement. Cette condition, courante en o.d.r., est facilement remplie car la plupart des grands transporteurs souscrivent à un système commun de publication des tarifs nommé ATPCo. En y publiant leurs tarifs, ils ont accès à ceux de leurs compétiteurs. Ce service est disponible de façon électronique et les tarifs qu'il contient sont ajustés plusieurs fois par jour.

- **Paramètres comportementaux**

Les valeurs des paramètres décrivant le comportement de la clientèle (valeur du temps, de la qualité de service, etc.) sont obtenus à l'aide de la procédure d'étalonnage décrite en annexe A. Cette procédure utilise des données de vente historiques. Ce type de données est disponible par l'entremise de vendeurs privés tels MIDT. Certains transporteurs disposent aussi d'un service interne visant à amasser et traiter ce type d'information.

L'information requise pour l'utilisation du modèle est disponible pour la plupart des transporteurs aériens. Cependant, elle provient de systèmes distincts, datant d'époques différentes et aux interfaces rarement compatibles. La phase préliminaire de collecte des données s'avère ainsi un

tâche complexe du point de vue technique.

Chapitre 6

Méthodes de résolution

Ce chapitre décrit les méthodes que nous avons utilisées pour résoudre le modèle introduit au chapitre précédent. Ces méthodes sont basées sur le principe de l'énumération implicite. Notre point de départ est le modèle en nombres entiers (BAFO-MIP ii) qui peut être résolu par des techniques classiques d'énumération implicite. Cette approche de type «force brute» atteint toutefois rapidement ses limites et n'est applicable que sur des instances de petite taille ayant peu d'intérêt pratique. Nous illustrons ensuite une méthode d'énumération implicite (de type *Branch-and-Bound*) développée spécialement afin de résoudre ce problème. En effet, nous démontrons que les solutions optimales du modèle BAFO possèdent une structure particulière de laquelle nous pouvons déduire des règles lexicographiques applicables en chaque nœud d'un arbre d'énumération implicite. Notre méthode s'appuie sur cette propriété et exploite des stratégies d'exploration et de branchement adaptées au modèle. Les relaxations linéaires de l'arbre peuvent être résolues par un code générique de programmation linéaire, mais nous proposons des méthodes alternatives et démontrons comment elles s'intègrent dans la méthode globale d'énumération implicite.

6.1 Jeux de données et environnement informatique

Ce chapitre contient de nombreux résultats numériques qui illustrent la performance des diverses approches. Les résultats présentés furent obtenus à partir de données réelles (horaire, demande,

tarifs, etc.) fournies par un transporteur majeur. Entre autres, les tests que nous avons effectués utilisent des données en vigueur sur le marché trans-frontalier nord-américain le 14 mars 2001. Le tableau 6.1 résume la taille des instances sur lesquelles nous avons testé les différentes méthodes de résolution. On remarquera que les neuf premières instances (100 à 108) correspondent à trois sous-réseaux qui se distinguent par le nombre de groupe d'usagers considérés. Les deux dernières instances (110 et 111) sont de taille comparable à l'instance 100 et seront utilisées afin d'illustrer les effets de différentes stratégies (exploration de l'arbre, choix du nœud, choix de la variable de branchement, etc.) sur la performance globale de l'algorithme. Nous utilisons ces instances de taille plus petite car nous sommes en mesure d'obtenir leur solution en quelques minutes, ce qui nous permet d'effectuer un plus grand nombre de tests.

Instance	K	F ¹	F ²	F	S ¹	S ²	S	P	G
100	3	42	48	90	39	62	101	381	3
101	3	42	48	90	39	62	101	381	9
102	3	42	48	90	39	62	101	381	15
103	12	105	97	202	89	104	193	1 076	3
104	12	105	97	202	89	104	193	1 076	9
105	12	105	97	202	89	104	193	1 076	15
106	86	426	1 343	1 769	222	955	1 177	11 957	3
107	86	426	1 343	1 769	222	955	1 177	11 957	9
108	86	426	1 343	1 769	222	955	1 177	11 957	15
110	6	24	153	177	20	176	196	916	3
111	4	24	44	68	23	61	84	423	3

TAB. 6.1 – Taille des instances utilisées lors des tests numériques.

Le logiciel CPLEX (version 8.1) fut utilisé pour résoudre la formulation de base (en nombres entiers) ainsi que pour résoudre les relaxations linéaires dans l'algorithme d'énumération spécialisé. Les données étaient stockées dans un serveur MySQL (version 4.1).

Deux implantations différentes de l'algorithme spécialisé ont été réalisées. La première, écrite en langage Python, fait appel à la librairie PyPlex (voir annexe C) et fut utilisée pour valider l'algorithme sur des données réelles.

La seconde implantation est écrite en langage C++ est beaucoup plus performante que la première. Les résultats numériques présentés dans ce chapitre furent obtenus avec cette deuxième version.

Les tests présentés dans ce chapitre (et les suivants) furent conduits sur un processeur Intel Xeon cadencé à 2.8 GHz avec 1 mégaoctet de mémoire disponible et fonctionnant sous le système

d'exploitation Linux version 2.4.18. Le compilateur utilisé fut GCC version 3.0.

6.2 Résolution par un code commercial (CPLEX)

Le modèle (BAFO-MIP ii) peut être résolu à l'aide de logiciels commerciaux de programmation en nombres entiers. Nous donnons au tableau 6.2 les résultats numériques obtenus en utilisant le logiciel CPLEX pour résoudre les différentes instances. Les temps limites de résolution ont été fixés à six heures pour les instances 100-102, 110 et 111, douze heures pour les instances 103-105 et vingt-quatre heures pour les instances 106-108. En aucun cas CPLEX a été en mesure d'identifier la solution optimale à l'intérieur des limites de temps que nous avons imposées.

Instance	Première solution entière			Meilleure solution entière			
	Valeur	Nœud	Temps (s)	Valeur	Nœud	Temps (s)	Gap(%)
100	251 023.75	0+ ¹	0.66	263 186.66	30 800	180.54	5.89
101	344 311.10	0+	7.64	344 311.10	0+	7.64	9.36
102	349 315.82	0+	9.69	368 626.67	0+	18.85	6.24
103	351 180.55	0+	2.30	371 316.78	20 000	694.75	7.17
104	513 017.59	0+	16.19	544 421.33	71 230	8 558.08	7.52
105	548 480.18	0+	40.17	575 840.12	62 400	20 153.13	7.72
106	1 371 400.49	0+	9.61	1 455 705.60	39 600	5 152.41	9.95
107	2 346 811.82	0+	68.50	2 423 811.19	52 420	34 401.95	10.11
108	2 673 229.21	0+	131.30	2 722 271.50	32 800	11 787.24	9.22
110	95 029.28	0+	0.95	112 642.64	36 000	344.42	5.13
111	68 871.11	0+	0.47	72 213.60	19 740	182.76	7.97

TAB. 6.2 – Résultats obtenus avec CPLEX sur la formulation (BAFO-MIP ii).

Afin de réaliser ces tests, la configuration par défaut de CPLEX a été utilisée. Notons que dans ce tableau et les suivants, la colonne intitulée «Gap» représente l'écart relatif entre la valeur de la meilleure borne entière et la valeur de la meilleure solution de la relaxation linéaire parmi les nœuds déjà explorés, au moment où la solution courante a été identifiée. Il s'agit ainsi du «gap» produit par le logiciel CPLEX.

¹La méthode de résolution de problème linéaires en nombres entiers du solveur CPLEX (version 8.0 et les suivantes) utilise par défaut diverses heuristiques lors de la résolution de la relaxation linéaire au nœud 0 afin d'obtenir rapidement une solution entière admissible. Lorsque CPLEX obtient une telle solution grâce à une heuristique, il affiche la notation «0+» dans la colonne correspondant au numéro du nœud. La solution ainsi obtenue est entière, mais cela ne signifie pas qu'il s'agit de la relaxation du problème entier à la racine comme on pourrait s'y attendre. Nous avons utilisé la même notation au dans la tableau 6.2 pour indiquer que la solution au nœud 0 a été obtenue à l'aide d'heuristiques.

6.3 Structure des schémas tarifaires optimaux

Les résultats numériques présentés au tableau 6.2 confirment les limites pratiques de l'approche «force brute» qui consiste à utiliser un logiciel de programmation en nombre entiers générique avec la formulation (BAFO-MIP*ii*). Dans cette section, nous démontrons que les schémas tarifaires optimaux possèdent une structure particulière qu'il est possible d'exploiter. Nous y décrivons aussi une méthode d'énumération implicite de type *Branch-and-Bound* qui exploite directement des règles lexicographiques induites par cette structure.

Nous débutons par reformuler le modèle de base en utilisant une notation vectorielle se prêtant mieux à l'exposition de résultats théoriques.

Le vecteur x représentera les flots du meneur. Si nous ne considérons que deux critères, soit prix et durée, les composantes du vecteur x sont les variables de décision x_{fg} , c'est-à-dire le flot en usagers du groupe $g \in G$ sur le vol du meneur $f \in F^1$. En supposant que $m = |G|$ et $n^1 = |F^1|$, le vecteur x prend la forme

$$x = \left(\underbrace{x_{11}, x_{12}, \dots, x_{1m}}_{\text{vol 1}}, \dots, \underbrace{x_{n^1 1}, x_{n^1 2}, \dots, x_{n^1 m}}_{\text{vol } n^1} \right).$$

Si on considère aussi le critère «qualité de service», lequel est associé aux produits tarifaires plutôt qu'aux vols, les composantes du vecteur x sont alors les variables x_{fgp} . Dans les deux cas, le vecteur y est de forme analogue, sauf qu'il représente les vols (et les produits) de la compétition plutôt que du meneur et est donc de dimension mn^2 au lieu de mn^1 , où $n^2 = |F^2|$. Afin d'alléger la présentation suivante, nous n'allons utiliser, en plus du prix, qu'un seul critère d'achat, soit la durée de vol. Les résultats se généralisent facilement au cas à plusieurs critères.

Le vecteur c , de même taille que le vecteur x , représente les coûts supplémentaires des vols (et produits), c'est-à-dire les coûts qui s'additionnent au tarif dans la définition du coût perçu total. Soit α_{gk} la valeur du temps des usagers du groupe g sur le marché k et soit f l'indice d'un vol sur ce marché. Nous avons ainsi

$$c_{fg} = \alpha_{gk} \mathcal{D}_f,$$

où \mathcal{D}_f est la durée du vol f . Le coût perçu total pour ce vol est donc $t_f^1 + c_{fg} = t_f^1 + \alpha_{gk} \mathcal{D}_f$.

Le vecteur d , de même taille que le vecteur y , représente les coûts perçus totaux pour les vols

de la compétition. Ainsi,

$$d_{fg} = t_f^2 + \alpha_{gk} D_f.$$

Le vecteur b représente la demande totale (par groupe et par marché) tandis que le vecteur u correspond à la capacité des segments du meneur. Les matrices A et B permettent d'associer les vols (respectivement du meneur et de la compétition) aux marchés sur lesquels ils sont offerts. La matrice C associe les vols du meneur aux segments qu'ils utilisent, alors que la matrice D permet de lier les prix (représentés par le vecteur t) aux vols (et, s'il y a lieu, produits) auxquels ils sont associés.

En utilisant cette nouvelle notation, on peut récrire le modèle bi-niveau comme

(BAFOvec)

$$\begin{aligned} \max_t \quad & tDx = \eta b - (cx + dy) \\ \text{s.l.c} \quad & Cx \leq u \\ \min_{x,y} \quad & (c + tD)x + dy \\ & Ax + By = b \quad (\eta) \\ & x, y \geq 0. \end{aligned}$$

En remplaçant le problème de second niveau par ses conditions d'optimalité, on obtient le problème à un seul niveau

(BAFOvec-NL)

$$\begin{aligned} \max_{t,\eta,x,y} \quad & \eta b - (cx + dy) \\ \text{s.l.c} \quad & Cx \leq u \\ & Ax + By = b \\ & x, y \geq 0 \\ & \eta A - tD \leq c \\ & \eta B \leq d \\ & (c + tD - \eta A)x = 0 \\ & (d - \eta B)y = 0. \end{aligned}$$

Nous introduisons maintenant les vecteurs de variables binaires $v \in \mathbb{B}^{m \times n^1}$ et $w \in \mathbb{B}^{m \times n^2}$ qui nous permettent de reformuler le modèle sous forme d'un programme en nombres entiers mixtes

s'écrivant comme

(BAFOvec-MIP)

$$\begin{aligned}
 & \max_{t, \eta, x, y, v, w} \quad \eta b - (cx + dy) \\
 & \text{s.l.c} \quad \quad \quad Cx \leq u \\
 & \quad \quad \quad Ax + By = b \\
 & \quad \quad \quad x, y \geq 0 \\
 & \quad \quad \quad \eta A - tD \leq c \\
 & \quad \quad \quad \eta B \leq d \\
 & \quad \quad \quad \eta A - tD - c \leq Mv \\
 & \quad \quad \quad x \leq M(1 - v) \\
 & \quad \quad \quad \eta B - d \leq Nw \\
 & \quad \quad \quad y \leq N(1 - w) \\
 & \quad \quad \quad v \in \mathbb{B}^{mn^1} \\
 & \quad \quad \quad w \in \mathbb{B}^{mn^2}
 \end{aligned}$$

où M et N sont des scalaires de taille appropriée.

Supposons que sur le marché $k \in K$, le meneur offre n_k^1 vols (comparativement à n_k^2 pour la compétition) et que $m = |G|$ groupes d'utilisateurs sont présents. Un vecteur de tarifs pour ce marché prend donc la forme

$$t^k = (t_1^k, t_2^k, \dots, t_{n_k^1}^k).$$

Nous allons supposer que les vols offerts sur ce marché sont étiquetés en ordre strictement croissant de durée, i.e. le vol 1 est le plus court et le vol n_k^1 le plus long^{1 2}. Nous supposons aussi que les groupes d'utilisateurs présents sur ce marché sont étiquetés en ordre strictement décroissant de valeur du temps, le groupe 1 correspondant aux passagers les plus «pressés» et ayant la plus grande propension à payer pour un vol plus court. Selon ces hypothèses, nous avons

$$\mathcal{D}_1 < \mathcal{D}_2 < \dots < \mathcal{D}_{n_k^1} \quad \text{et} \quad \alpha_{1k} > \alpha_{2k} > \dots > \alpha_{m_k k}.$$

¹Par un léger abus de langage, nous allons supposer que les objets f représentent à la fois les vols et leurs indices. Ainsi, si $f = 1$, on parlera du vol 1, en sous-entendant qu'il s'agit du vol d'indice 1. Il en va de même pour les indices g des groupes. C'est la valeur numérique de ces indices qui servira de base aux règles lexicographiques que nous introduisons ci-dessous.

²En pratique, si deux vols ont exactement la même durée, il suffit de soustraire une fraction de seconde à la durée de l'un d'entre eux pour que notre hypothèse soit respectée.

Le coût perçu du vol f pour les passagers du groupe g est donc $\pi_{fg}(\alpha_{gk(f)}) = t_f + c_{fg} = t_f + \alpha_{gk(f)}\mathcal{D}_f$. La notation $k(f)$ représente ici le marché auquel est associé le vol f .

DÉFINITION 6.1

Pour chaque groupe $g \in G$ sur chaque marché $k \in K$, soit

$$\hat{\eta}_g^k = \min_{f \in F^2(k)} \{d_{fg} = t_f^2 + \alpha_{gk(f)}\mathcal{D}_f\}.$$

Le coût perçu maximum que les usagers du groupe $g \in G$ sont prêts à déboursier sur le marché $k \in K$ est donc $\hat{\eta}_g^k$. \square

Remarquons que, selon le modèle,

$$\eta B \leq d,$$

i.e. la valeur maximale du coût perçu pour chaque groupe d'usagers sur chaque marché est borné supérieurement par le coût perçu des vols de la compétition. Ceci implique que si, sur un marché donné, le meneur désire attirer les usagers d'un groupe particulier, il doit s'assurer que le coût perçu de ses propres vols aux yeux de usagers de ce groupe n'excède pas celui du moins cher des vols de la compétition. C'est d'ailleurs de cette façon que l'on peut interpréter la contrainte

$$\eta A - tD \leq c.$$

Remarquons de plus que sur chaque marché k

$$\hat{\eta}_1^k \geq \hat{\eta}_2^k \geq \dots \geq \hat{\eta}_m^k.$$

DÉFINITION 6.2

Sur le marché k , le tarif t_f^k du vol f est dit **ajusté** s'il existe au moins un groupe d'usagers g pour lequel

$$t_f^k = \hat{\eta}_g^k - c_g^k.$$

\square

DÉFINITION 6.3

Sur le marché k , le **vecteur d'affectation** des groupes d'usagers est le vecteur $\Gamma^k = (\bar{g}_1, \bar{g}_2, \dots, \bar{g}_{n_k^1})$ tel que

$$\bar{g}_f = \min \{g \in G \mid x_{fg}^k > 0\}, \quad f = 1, 2, \dots, n_k^1.$$

En d'autres termes, \bar{g}_f est le plus petit indice des groupes affectés au vol f . On dira que ce vecteur est **ordonné** si $\bar{g}_f \leq \bar{g}_{f+1}$, pour $f = 1, 2, \dots, n_k^1 - 1$. \square

LEMME 6.1

Dans une solution optimale, il existe, sur chaque marché $k \in K$, au moins un groupe g pour lequel $\eta_g^k = \hat{\eta}_g^k$.

Preuve Par contradiction, supposons que $\exists k \in K$ tel que $\eta_g^k < \hat{\eta}_g^k, \forall g \in G$. Ainsi, sur ce marché, aucun groupe n'a intérêt à aller sur un vol de la compétition. Soit $\epsilon = \min_{g \in G} \{\hat{\eta}_g^k - \eta_g^k\}$. Le meneur peut augmenter le tarif de tous ses vols sur ce marché d'un montant ϵ et demeurer compétitif par rapport à la compétition, sans perdre de clientèle. Aussi, après avoir procédé à cette augmentation de tarif, il y aura au moins un groupe g pour lequel $\eta_g^k = \hat{\eta}_g^k$. Or le revenu du meneur après l'augmentation de tarif est supérieur au revenu initial, une contradiction, puisqu'on a supposé que la solution initiale est optimale. \square

LEMME 6.2

Dans toute solution admissible du modèle, pour un marché k donné, si le groupe g est présent sur le vol f et le groupe g' est présent sur le vol f' tel que $f < f'$, alors $g \leq g'$.

Preuve Supposons, par contradiction, que $g > g'$. Les contraintes (5.8) (admissibilité duale) et (5.12) (complémentarité) impliquent que

$$\begin{aligned} \eta_g^k &= t_f^k + \alpha_g^k \mathcal{D}_f \\ \eta_g^k &\leq t_{f'}^k + \alpha_g^k \mathcal{D}_{f'} \\ \eta_{g'}^k &\leq t_f^k + \alpha_{g'}^k \mathcal{D}_f \\ \eta_{g'}^k &= t_{f'}^k + \alpha_{g'}^k \mathcal{D}_{f'}. \end{aligned}$$

Nous avons donc

$$\begin{aligned} t_f^k + \alpha_g^k \mathcal{D}_f &\leq t_{f'}^k + \alpha_g^k \mathcal{D}_{f'} \quad \text{et} \\ t_{f'}^k + \alpha_{g'}^k \mathcal{D}_{f'} &\leq t_f^k + \alpha_{g'}^k \mathcal{D}_f. \end{aligned}$$

En additionnant ces deux inégalités, nous obtenons

$$\alpha_g^k \mathcal{D}_f + \alpha_{g'}^k \mathcal{D}_{f'} \leq \alpha_g^k \mathcal{D}_{f'} + \alpha_{g'}^k \mathcal{D}_f,$$

qui peut se réarranger comme

$$(\alpha_{g'}^k - \alpha_g^k) \mathcal{D}_{f'} \leq (\alpha_{g'}^k - \alpha_g^k) \mathcal{D}_f.$$

Nous avons supposé que $g > g'$, ce qui signifie que $\alpha_g^k < \alpha_{g'}^k$. La quantité $\alpha_{g'}^k - \alpha_g^k$ est ainsi strictement positive. La dernière inégalité nous permet donc de conclure que $\mathcal{D}_{f'} \leq \mathcal{D}_f$, une contradiction. \square

Le théorème 6.5 définit un ensemble de règles lexicographiques sur les indices des vols (et des produits s'ils sont considérés, comme ce sera le cas dans la généralisation de la méthode que nous présenterons dans une section subséquente) que doit respecter toute solution optimale du modèle. L'algorithme spécialisé que nous allons présenter impose ces règles en chaque nœud de l'arbre d'énumération et ainsi n'explore que les nœuds possédant cette structure particulière, réduisant du même coup la taille du domaine admissible. Ces règles, qui reviennent à fixer certaines variables de flot à 0 ou certaines contraintes à égalité, peuvent être perçues comme des coupes ajoutées au problème dual de la relaxation linéaire en chaque nœud. Bref, cet algorithme peut être décrit comme une méthode de type *Branch-and-Bound* ou encore de type *Branch-and-Cut* sur la formulation duale.

Avant de passer à la présentation de l'algorithme comme tel, nous faisons une dernière observation concernant les flots sur les vols de la compétition. Par hypothèse, ces vols ont une capacité illimitée. Si, pour un groupe g sur un marché k donné, les vols du meneur n'attirent pas la totalité de la demande b_{gk} , alors les usagers de ce groupe n'ayant pas été affectés sur les vols du meneur se dirigeront sur le vol de la compétition qui leur permet de minimiser le coût perçu (le coefficient dans l'objectif du vecteur de flot y sur les vols de la compétition est en effet $-d$, soit l'inverse du coût perçu sur ces mêmes vols). Or comme ce vol possède une capacité illimitée, il attirera la totalité de la demande résiduelle en usagers du groupe g sur le marché k . Ceci implique que nous pouvons ignorer les contraintes de complémentarité

$$(d - \eta B)y = 0$$

et identifier *a posteriori* quel vol de la compétition sera utilisé par chaque groupe sur chaque marché. Il s'agit en fait du vol de la compétition sur lequel est basé le tarif «ajusté» selon la définition que nous avons donnée ci-dessus. Nous pouvons donc ignorer ces contraintes (et leurs variables binaires correspondantes) dans la résolution du modèle pour ensuite calculer, une fois les flots optimaux x sur les vols du meneur connus, quels flots résiduels se répartiront sur les vols de la compétition (en ainsi obtenir la valeur du vecteur y). C'est pourquoi les variables binaires w sont absentes de l'algorithme d'énumération que nous allons maintenant présenter.

6.4 Algorithme d'énumération implicite spécialisé

On présume que sur chaque marché $k \in K$, pour chaque groupe $g \in G$, les indices inférieurs et supérieurs des vols empruntés par les groupes, lesquels sont dénotés respectivement par \underline{f}_g^k


```

23:    $X_i^1 \leftarrow X_i \cup \{v_{fg}^k = 1\}$ 
24:   if  $g > 1$  and  $v_{f,g-1}^k.isFixed()$  and  $v_{f,g-1}^k = 1$  then
25:      $X_i^0 \leftarrow X_i^0 \cup \{v_{f,g'}^k = 0 \mid \forall g' \in G, g' > g\}$ 
26:   if  $g < m_k$  and  $v_{f,g+1}^k.isFixed()$  and  $v_{f,g+1}^k = 1$  then
27:      $X_i^0 \leftarrow X_i^0 \cup \{v_{f,g'}^k = 0 \mid \forall g' \in G, g' < g\}$ 
28:   for all  $f' \in F^1(k) - \{f\}$  do
29:     for all  $g' \in G - \{g\}$  do
30:       if  $g' = 1$  then
31:         if  $f' > \underline{f}_2^k$  then
32:            $X_i^0 \leftarrow X_i^0 \cup \{v_{f',g'}^k = 0\}$ 
33:         if  $(g \neq 2)$  or  $(g = 2$  and  $f' > f)$  then
34:            $X_i^1 \leftarrow X_i^1 \cup \{v_{f',g'}^k = 0\}$ 
35:         else if  $g' = m_k$  then
36:           if  $f' < \overline{f}_{m_k-1}^k$  then
37:              $X_i^0 \leftarrow X_i^0 \cup \{v_{f',g'}^k = 0\}$ 
38:           if  $(g \neq m_k - 1)$  or  $(g = m_k - 1$  and  $f' < f)$  then
39:              $X_i^1 \leftarrow X_i^1 \cup \{v_{f',g'}^k = 0\}$ 
40:         else
41:           if  $f' > \underline{f}_{g'+1}^k$  or  $f' < \overline{f}_{g'-1}^k$  then
42:              $X_i^0 \leftarrow X_i^0 \cup \{v_{f',g'}^k = 0\}$ 
43:           if  $(g \notin \{g' - 1, g' + 1\})$  or  $(g = g' + 1$  and  $f' > f)$  or  $(g = g' - 1$  and  $f' < f)$  then
44:              $X_i^1 \leftarrow X_i^1 \cup \{v_{f',g'}^k = 0\}$ 
45:    $T.addNode(X_i^0)$ 
46:    $T.addNode(X_i^1)$ 
47:   if  $f < \underline{f}_g^k$  then
48:      $\underline{f}_g^k \leftarrow f$ 
49:   else if  $f > \overline{f}_g^k$  then
50:      $\overline{f}_g^k \leftarrow f$ 
51: return sol*
```

6.4.1 Implantation

Comme mentionné au début de ce chapitre, une implantation en langage C++ de cet algorithme a été réalisée. Celle-ci fait appel au solveur CPLEX pour les relaxations linéaires et obtient ses données en consultant un serveur MySQL. À titre d'exemple, la figure 6.4.1 illustre les résultats affichés à l'écran par cette implantation sur une petite instance (soit l'instance 100).

```

Job 100 has 3 markets, 3 groups, 381 products, 90 flights, 101 legs
Job id: 100
Relax strategy: Cplex (1)
Branch strategy: Maximum complementarity (1)
Node selection strategy: Maximum objective (1)
Tree exploration strategy: Classical (1)
Strategy: DEPTH FIRST
-----
Iter  Node  Depth  Heap Sz  Nb Fixed(%)  Curr Sol  Heap Top  Best Int  Total Comp  Real Obj  Non Comp Nb  Gap(%)  Time(s)
*   5    9      5        0  (0.0)  262197.86  262197.86  262197.86  48288.08  0.00  0  -  0.41
Strategy: BEST FIRST
100  165   12    37  1323  (50.7)  263513.38  264474.82  262197.86  1024.94  0.00  4  0.87  8.24
*  178  163   12    47    0  (0.0)  263186.66  263992.31  263186.66  0.00  0.00  0  0.31  14.84
<  200  263   11    48  1145  (43.9)  255146.67  263862.44  263186.66  0.00  0.00  0  -  16.51
*  243  310   13    34    0  (0.0)  263397.44  263715.94  263397.44  0.00  0.00  0  0.12  19.32
<  300  337   10    15  1062  (40.7)  263391.29  263551.29  263397.44  0.00  0.00  0  -  22.36
Branch and bound over
Optimal solution value: 263397.44
Time: 24.91 seconds
Total number of nodes: 349
Checking solution...
Solution is valid
    
```

FIG. 6.1 – Exemple des sorties obtenues avec l'implantation de l'algorithme spécialisé.

Le pseudo-code ci-dessus fait abstraction de plusieurs détails reliés à l'implantation de l'algorithme. Les sections suivantes ont pour but de décrire de façon plus précise les opérations ignorées dans la présentation de l'algorithme.

6.4.2 Élimination des variables binaires

Afin de faciliter la présentation de notre algorithme, nous avons utilisé explicitement des variables binaires v . Notre implantation n'utilise toutefois pas de variables binaires. À chaque nœud de l'arbre d'énumération, au lieu de tester l'intégralité de variables binaires, nous vérifions si les contraintes de complémentarité sont satisfaites par la solution de la relaxation. On recherche donc une solution «complémentaire» plutôt qu'entière. Cette approche a deux avantages. D'abord, d'un point de vue technique, elle diminue le nombre total de variables et allège l'implantation, et d'un point de vue théorique, elle permet possiblement de parvenir plus rapidement à une solution admissible dans une branche de l'arbre et ainsi d'obtenir une borne valide. Alors que la complémentarité d'une solution implique l'intégralité d'une solution du problème mixte équivalent, l'inverse n'est pas nécessairement vrai. Il se peut que, dans la formulation utilisant des variables binaires, une de ces variables soit fractionnaire alors que la contrainte de complémentarité correspondante soit satisfaite. Il est donc inutile de brancher sur une telle variable. Cependant, un algorithme générique d'énumération implicite ne peut utiliser cette information et procédera à un branchement inutile.

6.4.3 Stratégie d'exploration

Les premiers tests que nous avons réalisés utilisent une stratégie classique d'exploration de l'arbre d'énumération. Dans un premier temps, nous effectuons une recherche en profondeur afin d'obtenir le plus rapidement possible une solution admissible. Une fois cette première solution obtenue, nous passons à une recherche en largeur, laquelle permet de minimiser le nombre total de nœuds explorés [Wol98]. Cette stratégie d'exploration est sous-entendue dans l'opération `findNextNode()` du pseudo-code. La recherche en largeur peut s'effectuer selon plusieurs critères dont nous discuterons ci-dessous. Par exemple, dans le cas du critère «meilleure borne d'abord», on retient, parmi tous les nœuds déjà explorés (i.e. dont on a résolu la relaxation linéaire) possédant des fils non explorés, celui dont la valeur est maximale puis on explore un de ses nœuds-fils, d'abord celui de gauche, s'il n'est pas lui-même exploré, puis celui de droite. Le

nœud avec la meilleure borne est obtenu par l'entremise d'une structure de données auxiliaire, un monceau binaire classique qui est maintenu tout au long de l'exécution de l'algorithme. Le monceau est aussi nettoyé lorsqu'une nouvelle solution admissible est identifiée afin d'éliminer immédiatement les nœuds qui seraient de toute façon subséquentement élagués par borne grâce à cette solution.

Il est aussi possible d'utiliser une stratégie d'exploration mixte faisant intervenir alternativement une recherche en largeur et en profondeur. On se base alors sur un critère pré-déterminé afin de connaître la stratégie à appliquer à un moment donné de l'exploration de l'arbre d'énumération. Le critère que nous avons retenu est le nombre de variables binaires dont l'intégralité en un nœud n'est pas satisfaite (ou de façon équivalente, le nombre de contraintes de complémentarité dont la valeur est non nulle en ce nœud). Selon cette stratégie, l'algorithme démarre comme auparavant avec une recherche en profondeur ayant pour but d'obtenir le plus rapidement possible une première solution admissible (et donc une première borne). Ensuite, l'algorithme procède à une recherche en largeur à l'aide d'un monceau mais revient à une recherche en profondeur si le nombre de variables binaires non entières du nœud au sommet du monceau est inférieur à un certain seuil. Une fois tous les descendants de ce nœud explorés, l'algorithme revient à une recherche en largeur.

L'idée sous-tendant une telle stratégie est d'obtenir rapidement, par une recherche en profondeur, une nouvelle borne à partir d'un nœud prometteur presque complètement «résolu». Dans une recherche en largeur classique, cette borne n'est obtenue que lorsque ce nœud arrive au sommet du monceau. La difficulté est d'identifier le seuil approprié pour passer à une recherche en profondeur à partir d'un nœud prometteur.

Nous avons testé cette stratégie mixte sur les instances 100-102, 110 et 111. Les seuils retenus pour la recherche en profondeur en un nœud sont 15, 9, 5 et 3 variables binaires non entières. Les résultats sont présentés au tableau 6.3. On y donne aussi les résultats obtenus avec un seuil de 0, i.e. avec une stratégie d'exploration classique. Si le symbole «>» apparaît sur une ligne, cela signifie que le test fut interrompu à cause d'un manque de mémoire. Pour ces tests, les autres règles étaient fixées à leur valeur par défaut.¹ Notons que dans ce tableau et les suivants, la colonne intitulée «Gap» indique l'écart relatif entre la valeur de la meilleure solution

¹Les règles par défaut sont : exploration de l'arbre par stratégie classique (profondeur jusqu'à la première solution entière puis largeur), sélection de la variable de branchement par complémentarité maximale et maintien du monceau par valeur de l'objectif maximale.

admissible connue et la valeur de la meilleure solution de la relaxation linéaire parmi les nœuds déjà explorés, au moment où la solution courante a été identifiée ¹. Précisions aussi qu'un astérisque précédent la valeur d'une solution indique que cette solution a été reconnue comme optimale par l'algorithme, i.e. que l'arbre d'énumération a été parcouru en entier.

¹Cette définition du «gap» est la même que celle utilisée par CPLEX.

Instance	Première solution				Meilleure solution					
	Seuil	Valeur	Nœud	Temps (s)	Valeur	Nœud	Temps (s)	Gap(%)	Nœuds explorés	Temps total (s)
100	0	262 197.86	5	0.28	*263 397.44	243	12.89	0.12	349	16.67
	15	262 197.86	5	0.28	*263 397.44	395	16.89	-	487	20.66
	9	262 197.86	5	0.27	*263 397.44	182	8.16	-	393	16.66
	5	262 197.86	5	0.27	*263 397.44	118	6.20	-	397	17.17
101	3	262 197.86	5	0.27	*263 397.44	100	5.42	-	375	16.91
	1	356 153.95	12	1.95	*356 565.09	12 451	1 895.57	0.02	14 089	2 096.53
	15	356 153.95	12	1.95	356 153.95	12	1.95	-	>623 000	>79 256.72
	9	356 153.95	12	2.01	356 153.95	12	2.01	-	>675 900	>87 691.27
102	5	356 153.95	12	1.94	356 153.95	12	1.94	-	>1 314 100	>166 635.19
	3	356 153.95	12	1.94	356 153.95	12	1.94	-	>683 800	>87 249.26
	1	370 443.51	18	4.90	*370 847.43	23 079	6 009.66	0.01	24 795	6 364.46
	15	370 443.51	18	4.95	370 847.43	507	134.20	-	>411 300	>89 933.64
110	9	370 443.51	18	4.95	370 443.51	18	4.95	-	>461 300	>100 930.76
	5	370 443.51	18	5.21	370 443.51	18	5.21	-	>318 600	>71 084.86
	3	370 443.51	18	6.99	370 443.51	18	6.99	-	>305 400	>83 252.72
	0	111 899.71	6	0.44	*112 689.26	1 228	115.88	0.03	1 412	130.68
111	15	111 899.71	6	0.38	*112 689.26	1 520	86.51	-	1 793	101.82
	9	111 899.71	6	0.39	*112 689.26	676	38.42	-	1 653	93.37
	5	111 899.71	6	0.39	*112 689.26	448	26.55	-	1 475	85.03
	3	111 899.71	6	0.38	*112 689.26	184	12.78	-	1 421	90.86
112	0	72 180.31	3	0.14	*72 232.79	17	3.57	0.05	111	3.89
	15	72 180.31	3	0.14	*72 232.79	72	1.74	-	123	2.81
	9	72 180.31	3	0.09	*72 232.79	72	1.75	-	123	2.82
	5	72 180.31	3	0.09	*72 232.79	47	1.17	-	111	2.46
113	3	72 180.31	3	0.09	*72 232.79	39	0.98	-	111	2.43

TAB. 6.3 – Tests numériques sur les stratégies d'exploration mixtes.

Ces résultats nous permettent de faire plusieurs observations. D'abord, on remarque que la première solution entière est la même pour toutes les instances et est obtenue au même nœud dans tous les cas. Ce comportement est normal puisque que l'algorithme effectue d'abord une recherche en profondeur dans l'arbre pour obtenir rapidement une première solution entière. On peut toutefois observer que le temps requis pour trouver cette solution peut, pour une même instance, varier légèrement d'un test à l'autre, bien que l'algorithme suive la même «trajectoire» dans l'arbre d'énumération. Ceci s'explique par le fait que notre algorithme, bien que déterministe en lui-même, est implanté à l'aide de fonctions de la librairie standard des langages C et C++. Certaines de ces fonctions comportent un aspect probabiliste qui peut faire varier le temps d'exécution d'une fois à l'autre. Le programme inscrit aussi des résultats intermédiaires dans un fichier externe qui physiquement repose sur un disque dur auquel le programme accède par un réseau local. Dépendamment du taux d'utilisation du disque et du réseau en question, les temps requis pour inscrire les résultats intermédiaires dans le fichier peuvent varier.

Ces tests indiquent aussi que, pour les instances où l'arbre a été complètement exploré, le temps d'exécution n'est pas nécessairement une fonction croissante du nombre total de nœuds explorés. En effet, lorsque l'algorithme est en mode «recherche en profondeur», les solutions des relaxations linéaires à chaque nœud sont calculées à l'aide de la méthode duale du simplexe, en partant de la solution optimale obtenue au nœud-père. Cette solution demeure duale-admissible et constitue un bon point de départ, de sorte qu'un nombre moindre d'itérations du simplexe est nécessaire. Les relaxations sont donc résolues plus rapidement, ce qui améliore le temps total de résolution.

Pour ce qui est de la stratégie d'exploration comme telle, le tableau 6.3 démontre que celle-ci peut avoir des bénéfices. Pour quatre des cinq instances (100, 102, 110 et 111), la stratégie d'exploration mixte a permis à l'algorithme de trouver beaucoup plus rapidement la solution optimale (par exemple au nœud 507 plutôt qu'au nœud 23 079 pour l'instance 102 avec un seuil de 15), bien que le nombre total de nœuds explorés diffère peu d'un test à l'autre. Cependant, sur les instances plus grandes (101 et 102), l'algorithme s'est engouffré dans un «puits sans fond» alors qu'il était en mode «recherche en profondeur», jusqu'à ce que la taille de l'arbre devienne trop importante et que le programme se termine par manque de mémoire (par exemple après plus de 1 300 000 nœuds et près de 48 heures dans le cas de l'instance 101 avec un seuil de 5). En pratique, une telle stratégie devrait donc être utilisée avec précaution et inclure un mécanisme permettant de revenir à une recherche en largeur si la recherche en profondeur ne permet pas d'obtenir une solution entière lorsque rendue à une distance critique de la racine.

6.4.4 Sélection de la variable de branchement

Le pseudo-code de l'algorithme présente la sélection de la variable de branchement comme

$$\text{let } v_{fg}^k \in X_i - \mathbb{B},$$

ce qui ne précise pas la règle de sélection. En réalité, nous avons testé plusieurs règles que nous décrivons ci-dessous. Afin de faciliter la présentation de ces règles, nous allons poser

$$\epsilon_{fg} = (t_f + c_{fg} - \eta_{gk(f)}) x_{fg},$$

i.e. ϵ_{fg} est la valeur de la contrainte de complémentarité correspondant au vol du meneur f et au groupe d'utilisateurs g .

Les règles que nous avons considérées sont :

1. Complémentarité maximale

Soient

$$(\hat{f}, \hat{g}) \in \arg \max_{\substack{f \in F^1 \\ g \in G}} \{ \epsilon_{fg} | \epsilon_{fg} \neq 0 \}.$$

La variable de branchement est $v_{\hat{f}, \hat{g}}^{k(\hat{f})}$.

2. Complémentarité minimale

Soient

$$(\hat{f}, \hat{g}) \in \arg \min_{\substack{f \in F^1 \\ g \in G}} \{ \epsilon_{fg} | \epsilon_{fg} \neq 0 \}.$$

La variable de branchement est $v_{\hat{f}, \hat{g}}^{k(\hat{f})}$.

3. Choix aléatoire

Soit $v_{fg}^k \in X_i - \mathbb{B}$ la variable de branchement. Dans notre implantation, ce choix se fait à l'aide d'un générateur de nombre pseudo-aléatoire, nommément la fonction `random()` de la librairie mathématique du langage C.

4. Complémentarité maximale pondérée

Dans ce cas-ci, nous considérons toutes les variables correspondant aux contraintes de complémentarité non satisfaites (i.e. pour lesquelles $\epsilon_{fg} \neq 0$) et nous divisons la valeur de la «non complémentarité» par la somme des valeurs des deux termes de la contrainte (à laquelle on ajoute une petite quantité pour éviter les divisions par 0). Le but d'une telle opération est, étant donnée un ensemble de contraintes de la forme $ab = 0$, de ne pas désavantager celles pour lesquelles les termes a et b sont relativement petits.

Ainsi, soient

$$(\hat{f}, \hat{g}) \in \arg \max_{\substack{f \in F^1 \\ g \in G}} \left\{ \epsilon_{fg} / (t_f + c_{fg} - \eta_{gk(f)} + x_{fg} + 0.001) \mid \epsilon_{fg} \neq 0 \right\}.$$

La variable de branchement est $v_{\hat{f}, \hat{g}}^{k(\hat{f})}$.

5. Complémentarité minimale pondérée

Soient

$$(\hat{f}, \hat{g}) \in \arg \min_{\substack{f \in F^1 \\ g \in G}} \left\{ \epsilon_{fg} / (t_f + c_{fg} - \eta_{gk(f)} + x_{fg} + 0.001) \mid \epsilon_{fg} \neq 0 \right\}.$$

La variable de branchement est $v_{\hat{f}, \hat{g}}^{k(\hat{f})}$.

Le tableau 6.4 présente des résultats numériques obtenus en utilisant chacune de ces stratégies sur les instances 100-102, 111 et 110. Pour ces tests, les autres règles étaient fixées à leur valeurs par défaut.

À la lumière de ces résultats, il est clair que la règle 1, i.e. «complémentarité maximale», est la plus efficace pour choisir la variable de branchement à chaque nœud, bien que la règle 4 «complémentarité maximale pondérée» donne aussi de bons résultats sur certaines instances. La performance de la règle 1 s'explique lorsqu'on remet le problème dans son contexte originel, c'est-à-dire un problème bi-niveau. La complémentarité doit être satisfaite pour respecter les conditions d'optimalité du problème de second niveau. Or les contraintes avec la plus grande «valeur de complémentarité» sont celles qui nous éloignent le plus d'une solution optimale au second niveau. Ainsi, dans l'arbre d'énumération, nous avons avantage à traiter d'abord ces contraintes et brancher sur la variable leur correspondant.

6.4.5 Sélection du nœud courant

Lorsque l'algorithme explore l'arbre d'énumération en mode «meilleure borne d'abord» («*Best-first*»), que ce soit dans un mode d'exploration «classique» (i.e. recherche en profondeur jusqu'à ce que la première solution entière soit obtenue puis meilleure borne à partir de ce point) ou encore dans un mode «mixte» où l'on passe à la recherche en profondeur pour «terminer» une branche pour ensuite revenir en mode meilleure borne, il maintient un monceau contenant les nœuds explorés jusque-là. Le prochain nœud à être exploré est simplement celui qui se trouve au sommet du monceau. Il est alors extrait et le monceau est remis à jour.

Dans un schéma de *Branch-and-Bound* classique, le critère utilisé pour maintenir le monceau est la valeur de la fonction objectif associée à chaque nœud (ou encore un estimé de cette valeur dans une approche heuristique). Dans le cas d'un problème de maximisation, on choisit alors le nœud exploré ayant la plus grande valeur de l'objectif, dans l'espoir de trouver rapidement une meilleure solution entière et de minimiser le nombre total de nœuds explorés.

Dans notre cas, le problème que nous tentons de résoudre est initialement un programme bi-niveau que nous reformulons en problème avec contraintes de complémentarité puis en programme linéaire à variables mixtes. La difficulté algorithmique provient précisément des contraintes de complémentarité bilinéaires. En remplaçant le problème en nombres entiers dans ce contexte, on peut imaginer d'autres critères de sélection du nœud courant qui exploitent l'information provenant du problème de complémentarité. Les critères que nous avons retenus sont le nombre de contraintes de complémentarité non satisfaites et la valeur totale des contraintes de complémentarité non satisfaites. Par souci de symétrie, nous avons testé des versions «minimum» et

«maximum» de chacun de ces critères.

De façon formelle, au nœud i de l'arbre d'énumération, soit

$$N_i = \{(f, g) \in F^1 \times G \mid (t_f + c_{fg} - \eta_{gk(f)}) x_{fg} \neq 0\}$$

l'ensemble des indices de vols et de groupes correspondant aux contraintes de complémentarité non satisfaites. Soit aussi

$$\sigma_i = \sum_{(f,g) \in N_i} (t_f + c_{fg} - \eta_{gk(f)}) x_{fg}$$

la somme des valeurs des contraintes non satisfaites. Nous supposons aussi que z_i dénote la valeur de l'objectif au nœud i .

Les critères utilisés lors de la construction et du maintien du monceau pour comparer les nœuds i et j sont :

1. Valeur de l'objectif maximale

$$z_i > z_j$$

2. Valeur de l'objectif minimale

$$z_i < z_j$$

3. Complémentarité non satisfaite maximale

$$\sigma_i > \sigma_j$$

4. Complémentarité non satisfaite minimale

$$\sigma_i < \sigma_j$$

5. Nombre maximum de contraintes de complémentarité non satisfaites

$$|N_i| > |N_j|$$

6. Nombre minimum de contraintes de complémentarité non satisfaites

$$|N_i| < |N_j|$$

Si ces conditions sont satisfaites, le nœud i devient le parent du nœud j dans le monceau, sinon il devient son fils.

Le tableau 6.5 présente des résultats numériques obtenus en utilisant chacun de ces critères sur les instances 100-102, 111 et 110. Pour ces tests, les autres règles étaient fixées à leur valeurs par défaut.

Instance	Première solution			Meilleure solution			Temps total (s)			
	Règle	Valeur	Nœud	Temps (s)	Valeur	Nœud		Temps (s)	Gap(%)	Nœuds explorés
100	1	262 197.86	5	0.28	*263 397.44	243	12.89	0.12	349	16.67
	2	262 197.86	5	0.27	*263 397.44	205	9.29	5.81	415	18.16
	3	262 197.86	5	0.27	*263 397.44	1 067	49.35	0.12	1 074	49.63
	4	262 197.86	5	0.27	*263 397.44	205	9.01	5.81	415	17.57
	5	262 197.86	5	0.27	*263 397.44	1 017	48.86	0.12	1 022	49.09
	6	262 197.86	5	0.27	*263 397.44	205	9.01	5.81	415	17.57
101	1	356 153.95	12	1.95	*356 565.09	12 451	1 895.57	0.02	14 089	2 096.53
	2	356 153.95	12	1.94	*356 565.09	6 140	866.79	5.60	19 043	2 545.36
	3	356 153.95	12	1.96	*356 565.09	23 540	3 179.31	0.02	23 552	3 182.61
	4	356 153.95	12	1.95	*356 565.09	7 016	1 020.24	5.60	17 985	2 483.13
	5	356 153.95	12	2.04	*356 565.09	36 840	6 107.80	0.02	36 864	6 111.40
	6	356 153.95	12	2.00	*356 565.09	6 200	953.07	5.60	19 231	4 299.38
102	1	370 443.51	18	4.90	*370 847.43	23 079	6 009.66	0.01	24 795	6 364.46
	2	370 443.51	18	9.70	*370 847.43	3 733	1 622.64	5.60	24 967	11 837.08
	3	370 443.51	18	6.68	*370 847.43	76 360	40 542.63	0.01	76 412	40 631.44
	4	370 443.51	18	14.01	*370 847.43	3 245	2 378.61	5.72	33 073	23 891.36
	5	370 443.51	18	14.15	*370 847.43	68 533	54 042.29	0.02	68 743	54 197.07
	6	370 443.51	18	13.87	*370 847.43	3 291	2 333.60	5.60	24 719	16 842.88
110	1	111 899.71	6	0.44	*112 689.26	1 228	115.88	0.03	1 412	130.68
	2	111 899.71	6	0.95	*112 689.26	895	123.93	5.09	1 951	268.65
	3	111 899.71	6	1.05	*112 689.26	2 222	341.27	0.03	2 232	342.41
	4	111 899.71	6	1.01	*112 689.26	895	124.73	5.09	1 951	271.37
	5	111 899.71	6	0.96	*112 689.26	2 226	389.42	0.08	2 288	397.03
	6	111 899.71	6	1.04	*112 689.26	895	123.21	5.09	1 949	268.79
111	1	72 180.31	3	0.14	*72 232.79	17	3.57	0.05	111	3.89
	2	72 180.31	3	0.13	*72 232.79	100	3.21	7.89	123	3.85
	3	72 180.31	3	0.12	*72 232.79	93	5.11	0.05	111	5.88
	4	72 180.31	3	0.12	*72 232.79	98	4.55	7.89	123	5.76
	5	72 180.31	3	0.12	*72 232.79	95	3.47	0.05	117	4.03
	6	72 180.31	3	0.12	*72 232.79	100	3.22	7.89	123	3.90

TAB. 6.5 – Tests numériques sur les critères de maintien du monceau.

Ces résultats indiquent que les critères de maintien du monceau les plus performants sont 2, 4 et 6. Ce sont ceux qui, en général, permettent d'obtenir le plus rapidement la solution optimale et de minimiser le nombre total de nœuds explorés. La raison expliquant la performance du critère 2 (valeur de l'objectif minimale) est sans doute l'importance relative du saut d'intégralité entre la valeur de la relaxation à la racine et la valeur de la solution optimale. Ce saut est de 5.49% pour l'instance 100, 5.30% pour 101, 5.30% pour 102, 4.89% pour 110 et 7.34% pour 111. On a donc intérêt à privilégier les nœuds ayant une valeur moindre (donc possiblement plus «près» de la solution optimale) que les nœuds ayant une plus grande valeur (donc possiblement plus «près» de la relaxation à la racine). Pour ce qui est des critères 4 (complémentarité non satisfaite minimale) et 6 (nombre minimal de contraintes de complémentarité non satisfaites), les raisons expliquant leur performance sont reliées. Moins une solution comporte de contraintes de complémentarité non satisfaites (ou moindre est la somme des valeurs de ces contraintes), plus près elle est d'une solution entière et possiblement optimale.

6.4.6 Comparaisons avec CPLEX

Le tableau 6.6 compare les résultats obtenus sur toutes les instances avec d'une part le solveur commercial CPLEX et d'autre part l'algorithme d'énumération implicite spécialisé présenté ci-dessus. La configuration retenue pour l'algorithme spécialisé était : exploration «classique» (exploration en profondeur jusqu'à ce que la première solution entière soit obtenue et exploration en largeur par la suite), sélection de la variable de branchement par «complémentarité maximale» (règle 1) et maintien du monceau par «valeur minimale de l'objectif» (règle 2). Pour ce qui est du solveur CPLEX, la configuration par défaut fut utilisée (ce qui inclut l'utilisation d'heuristiques). Nous n'avons pas appliqué de critère d'arrêt spécifique pour l'algorithme spécialisé ; nous avons plutôt arrêté les calculs lorsque la recherche semblait stagner. Par contre, nous ne présentons que les résultats qui furent obtenus à l'intérieur d'un délai identique à celui que nous avons accordé à CPLEX, soit 6 heures pour les instances 100-102, 110 et 111, douze heures pour les instances 103-105 et vingt-quatre heures pour les instances 106-108. La colonne intitulée «Rapport» au tableau 6.6 indique, en pourcentage, le rapport entre la valeur de la meilleure solution obtenue par l'algorithme spécialisé et la valeur de la meilleure solution obtenue par CPLEX. Chaque comparaison est ensuite résumée de façon graphique aux pages suivantes. Sur ces graphiques, un cercle vide représente une solution admissible identifiée par CPLEX alors qu'un triangle noir représente une solution identifiée par l'algorithme spécialisé.

Instance	CPLEX				Algo. spéc.				Rapport (%)
	Valeur	Nœud	Temps (s)	Gap(%)	Valeur	Nœud	Temps (s)	Gap(%)	
100	263 186.66	30 800	180.54	5.89	*263 397.44	205	8.87	5.81	100.08
101	344 311.10	0+	7.64	9.36	*356 153.95	12	1.99	5.60	103.44
102	368 626.67	0+	18.85	6.24	*370 443.51	18	4.90	5.60	100.49
103	371 316.78	20 000	694.75	7.17	*373 278.74	105 402	12 363.62	6.58	100.53
104	544 421.33	71 230	8 558.08	7.52	544 970.83	97 378	35 487.02	7.39	100.10
105	575 840.12	62 400	20 153.13	7.72	576 990.42	66 056	38 300.97	7.49	100.20
106	1 455 705.60	39 600	5 152.41	9.95	1 420 762.13	6 950	22 860.00	12.66	97.60
107	2 423 811.19	52 420	34 401.95	10.11	2 462 240.45	6 218	61 675.73	8.39	101.59
108	2 722 271.50	32 800	11 787.24	9.22	2 776 498.79	927	17 487.58	7.08	101.99
110	112 642.64	36 000	344.42	5.13	*112 689.26	895	50.63	5.09	100.04
111	72 213.60	19 740	182.76	7.97	*72 232.79	100	2.39	7.89	100.03

TAB. 6.6 – Comparaisons entre CPLEX et l'algorithme spécialisé.

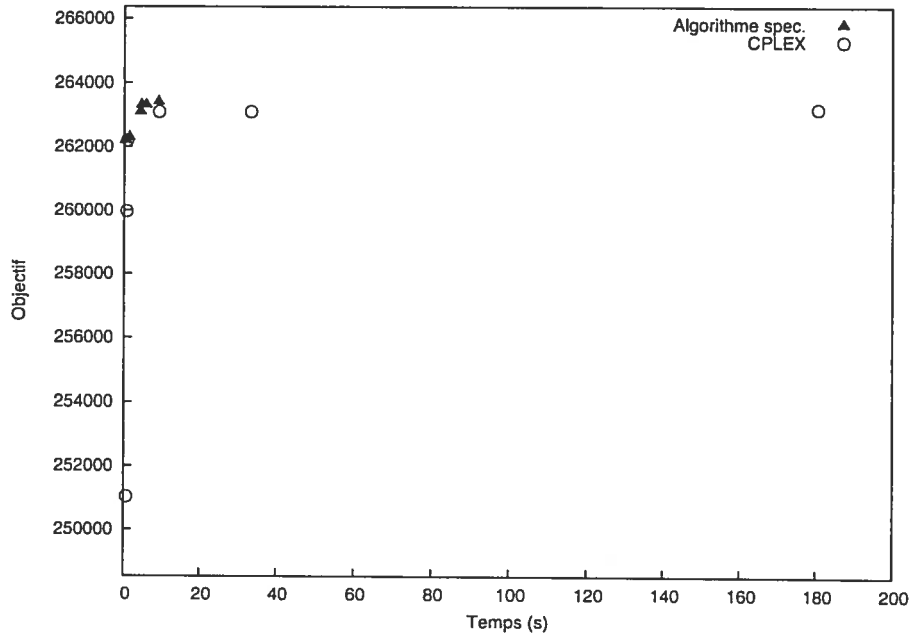


FIG. 6.2 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 100.

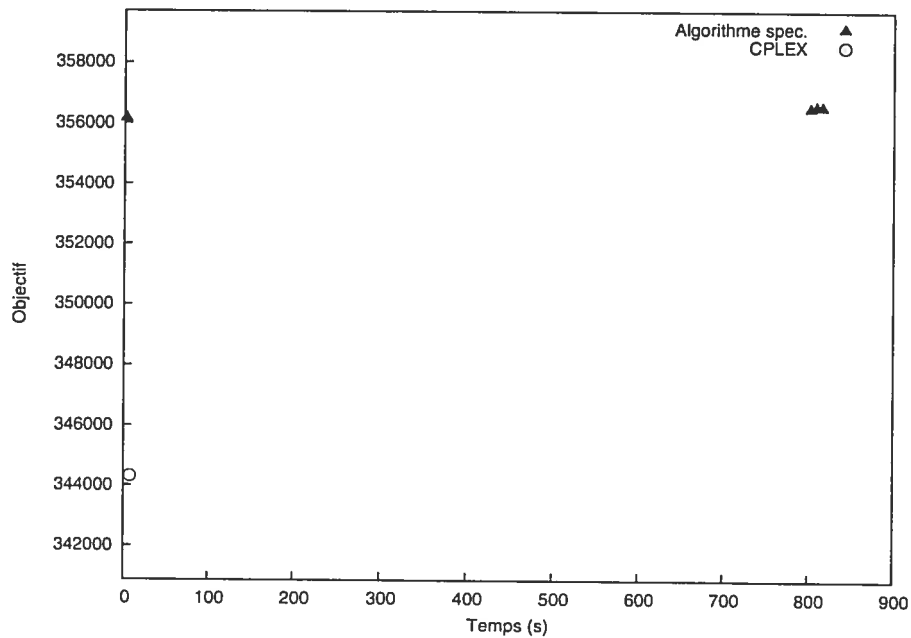


FIG. 6.3 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 101.

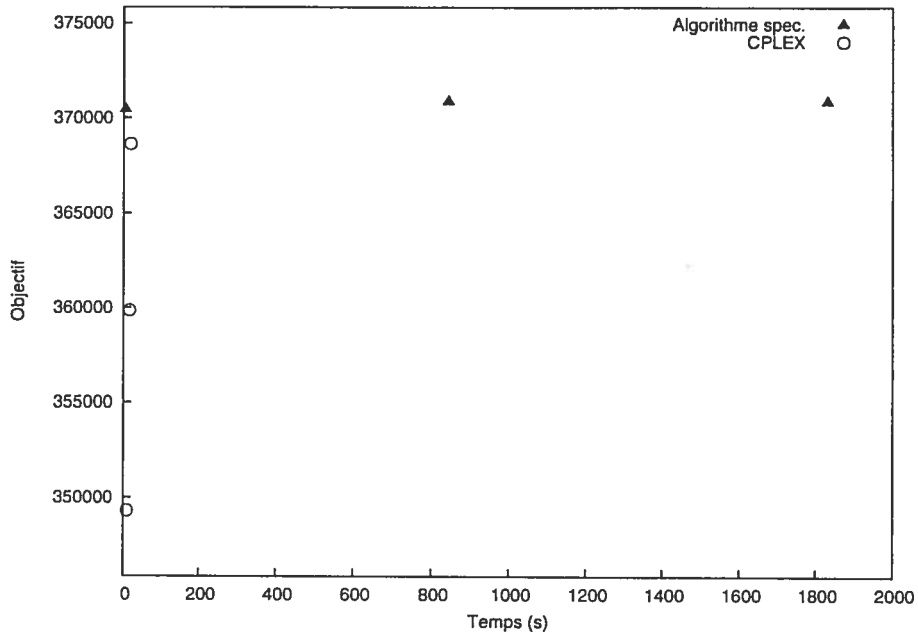


FIG. 6.4 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 102.

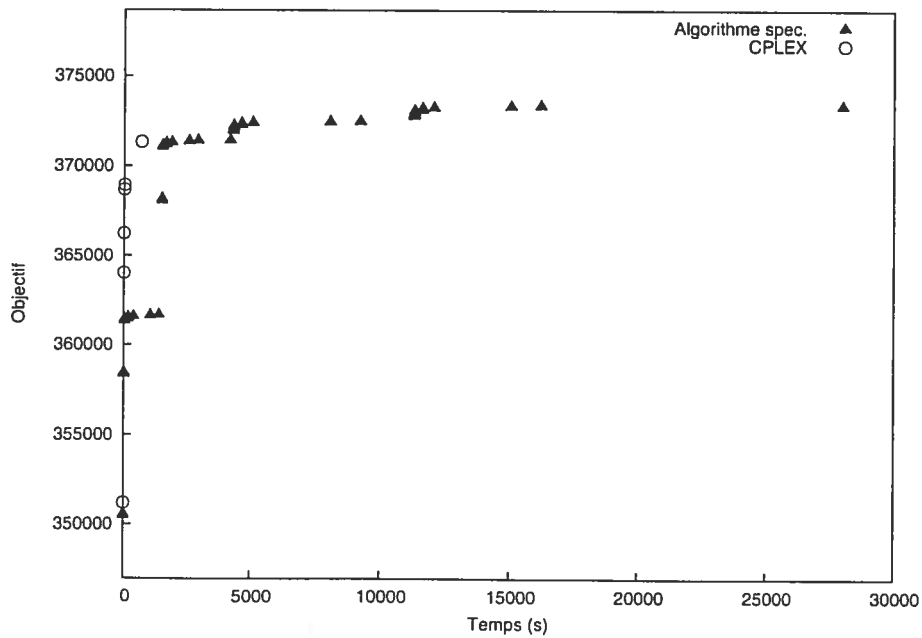


FIG. 6.5 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 103.

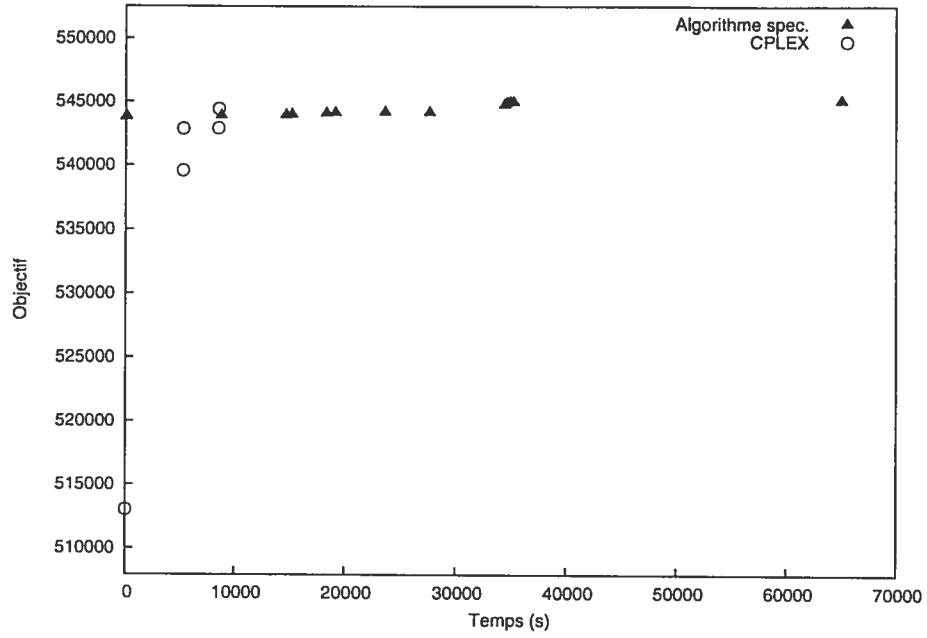


FIG. 6.6 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 104.

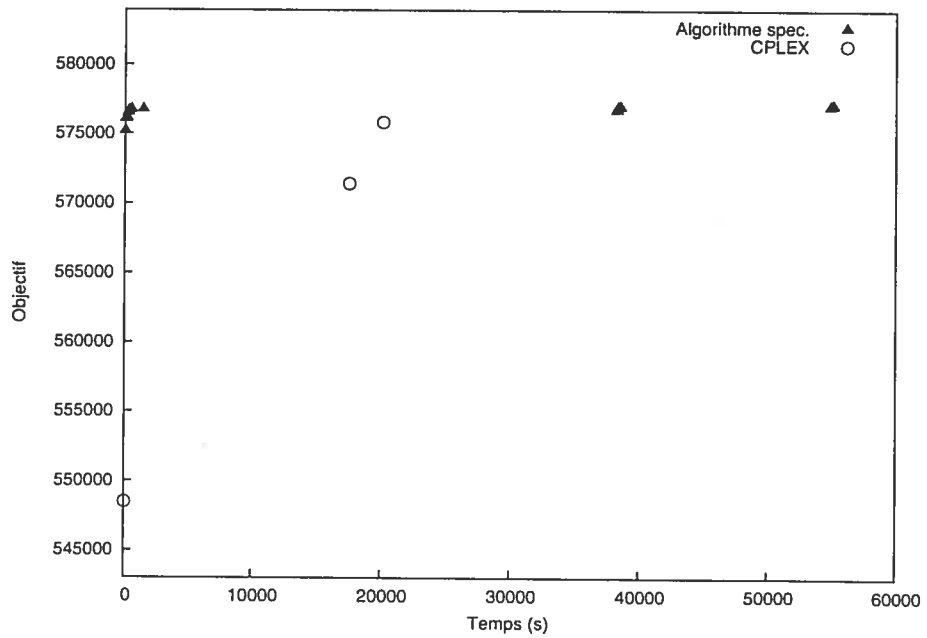


FIG. 6.7 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 105.

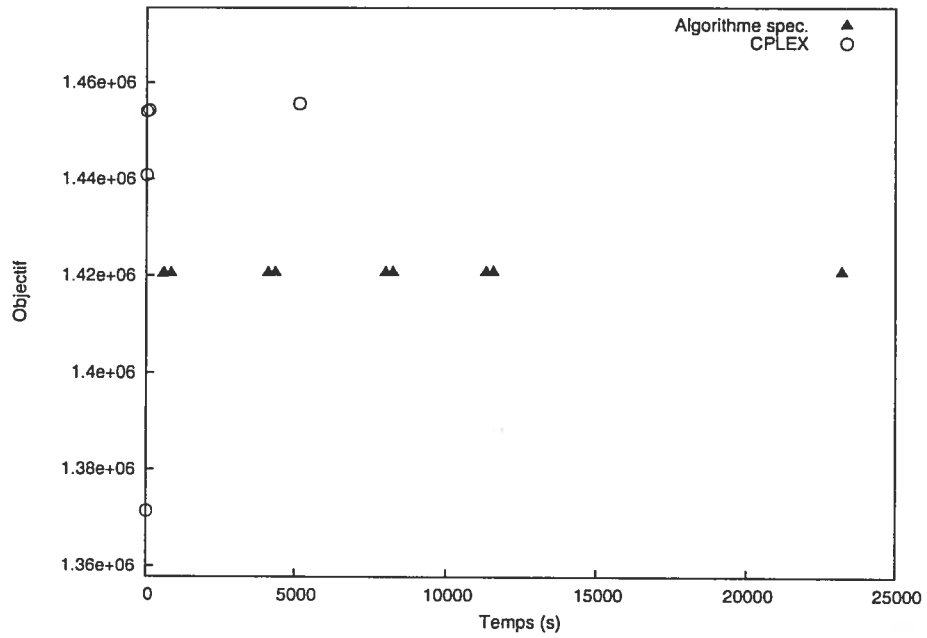


FIG. 6.8 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 106.

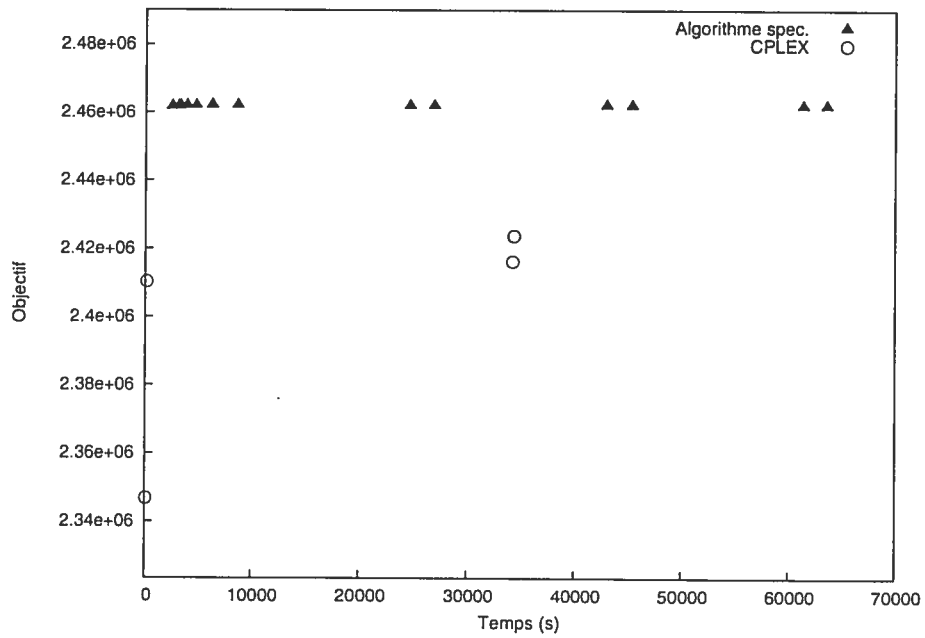


FIG. 6.9 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 107.

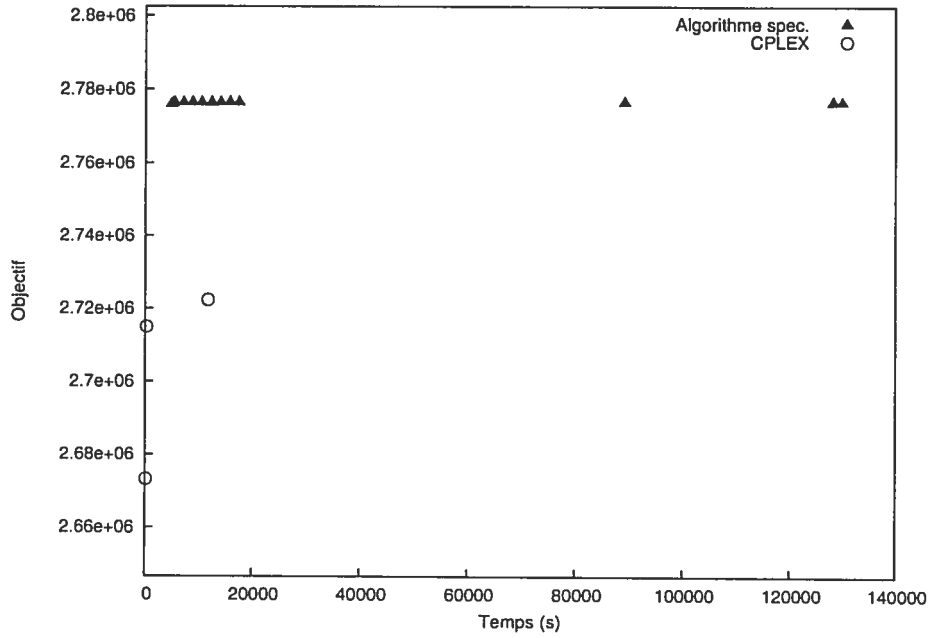


FIG. 6.10 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 108.

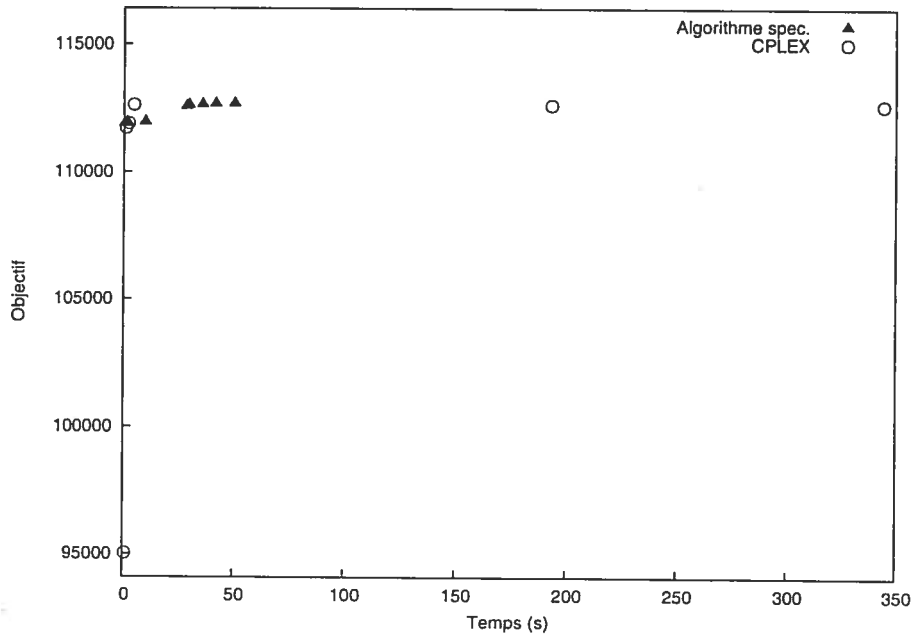


FIG. 6.11 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 110.

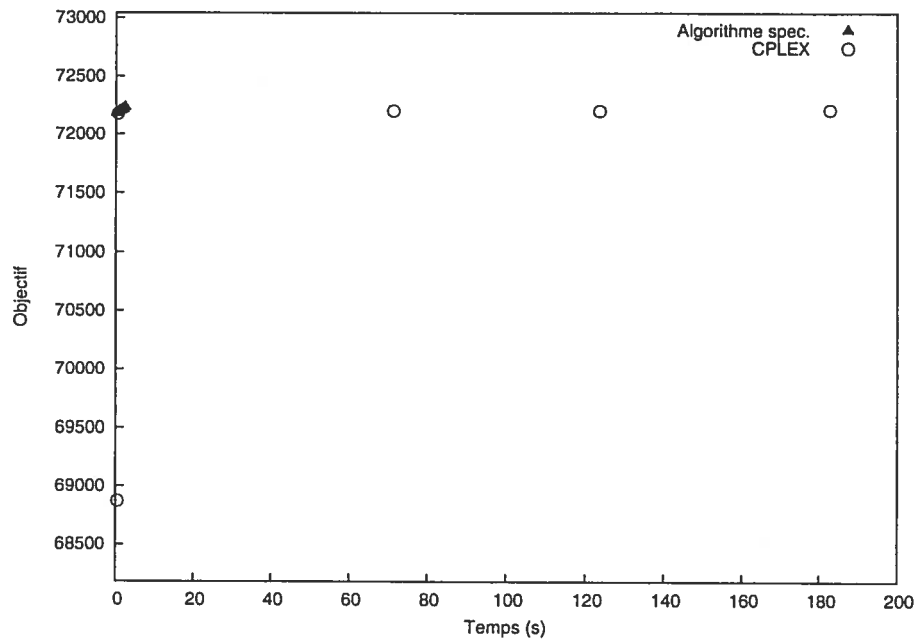


FIG. 6.12 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 111.

Les résultats présentés au tableau 6.6 nous permettent de faire plusieurs observations. Sauf dans le cas de l'instance 106, on constate que l'algorithme spécialisé a été en mesure d'identifier de meilleures solutions que CPLEX. Pour les six plus petites instances, l'algorithme spécialisé a aussi été en mesure de parcourir entièrement l'arbre d'énumération et de démontrer l'optimalité des solutions obtenues. Les instances 101 et 102 sont particulièrement intéressantes. Dans les deux cas, la meilleure et seule solution entière que CPLEX et l'algorithme spécialisé ont été en mesure d'identifier est la première obtenue. CPLEX a calculé cette solution à l'aide d'heuristiques au nœud racine (d'où la notation «0+») tandis que l'algorithme spécialisé a identifié cette solution en effectuant une recherche en profondeur. Pour les deux instances, la solution identifiée par l'algorithme spécialisé est meilleure que celle de CPLEX (l'optimalité de ces solutions a été démontrée par l'algorithme).

Les résultats obtenus pour les instances de plus grande taille illustrent une faiblesse de notre implantation. Bien qu'en général l'algorithme spécialisé identifie de meilleures solutions, le nombre de nœuds qu'il est en mesure d'explorer est de beaucoup inférieur à celui de la méthode d'énumération implicite de CPLEX. Par exemple, dans le cas de l'instance 108, CPLEX a pu explorer 32 000 nœuds en 11 787.24 secondes (pour une moyenne d'environ 2.7 nœuds à la seconde) alors que l'algorithme spécialisé a parcouru 927 nœuds en 17 487.58 secondes (pour une moyenne d'environ 0.053 nœud à la seconde, soit environ 50 fois moins que CPLEX). Ce phénomène

possède une explication technique bien simple. CPLEX garde en mémoire la base optimale à chaque nœud ayant des fils non explorés, ce qui fait que lorsqu'il doit résoudre un de ces fils, il peut repartir de la base optimale au nœud-père en utilisant la méthode duale du simplexe, qui est très efficace dans ce cas car la solution optimale au nœud-père demeure duale-admissible au nœud-fils. Cependant, pour des problèmes de grande taille, cette information ne peut être entièrement conservée en mémoire vive et doit être sauvegardée dans un fichier temporaire sur disque. Ce fichier est maintenu à l'aide de structures de données relativement complexes afin que l'on soit en mesure de rapidement retrouver l'information sur un nœud donné¹. Notre implantation, moins sophistiquée sur la plan technique, ne maintient pas de fichier temporaire et ne conserve pas en mémoire les bases optimales de nœuds déjà résolus (sauf en recherche en profondeur, où on réoptimise à partir de la solution au nœud-père avec la méthode duale du simplexe). Toutefois, bien qu'elle parcoure jusqu'à cinquante fois moins de nœuds, notre approche identifie en général de meilleures solutions car elle explore un espace plus restreint. Avec le même «taux d'exploration» que CPLEX, on peut supposer que l'algorithme spécialisé serait en mesure d'identifier des solutions encore meilleures ou de démontrer l'optimalité de celles obtenues.

¹Sans doute pour ne pas révéler de secrets commerciaux, la documentation de CPLEX est avare de détails sur la nature exacte de l'information gardée en mémoire par la méthode d'énumération implicite et la façon dont cette information est utilisée. À ce sujet, le guide de l'utilisateur [Ilo02] affirme «*The information about a tree that ILOG CPLEX accumulates in memory can be substantial. In particular, ILOG CPLEX saves a basis for every unexplored node. Furthermore, when ILOG CPLEX uses the best bound or best estimate strategies of node selection, the list of unexplored nodes itself can become very long for large or difficult problems. How large the unexplored node list can be depends on the actual amount of memory available, the size of the problem, and algorithm selected.*» Afin de mesurer de façon plus précise l'impact de l'information conservée en mémoire sur les temps de résolution globaux, nous avons mesuré, pour l'instance 108, le temps de résolution de la relaxation linéaire correspondant au nœud zéro par la méthode duale du simplexe (celle que nous utilisons dans notre implantation). Le solveur linéaire de CPLEX met 79.12 secondes à résoudre cette relaxation, mais ce temps inclut une étape de *preprocessing* de 62.58 secondes au cours de laquelle la taille du problème est réduite par diverses substitutions. Comme nous utilisons toujours la même formulation, cette étape préalable n'est effectuée qu'une fois dans notre implantation, à la racine de l'arbre. Ainsi, en excluant ce *preprocessing* qui n'est effectué qu'une fois, nous avons donc un temps de résolution de 16.54 secondes avec le solveur linéaire de CPLEX, ce qui correspond à un taux d'exploration de 0.060 nœud à la seconde, lequel est très près de ce que nous avons observé pour cette instance. La légère différence peut s'expliquer par le fait que dans certains cas, le nœud courant est effectivement le fils du dernier nœud résolu. La méthode duale est efficace dans ces cas. Toutefois, le prix à payer en termes de performance lorsque la majorité des relaxations sont résolues avec une base initiale quelconque est très élevé.

6.5 Résolution des relaxations linéaires par pénalité

À chaque nœud de l'arbre d'énumération, nous devons résoudre un programme linéaire de la forme

$$\begin{aligned}
 \max_{t, \eta} \quad & \eta b - (cx + dy) \\
 \text{s.l.c} \quad & Cx \leq u \\
 & Ax + By = b \\
 & x, y \geq 0 \\
 & \eta A - tD \leq c \\
 & \eta B \leq d
 \end{aligned}$$

Cette formulation correspond à la première relaxation, i.e. celle résolue au nœud 0 de l'arbre d'énumération. À mesure que l'on descend dans l'arbre, des contraintes de la forme $\eta A - tD \leq c$ sont fixées à égalité et des variables x sont fixées à zéro¹.

Jusqu'à maintenant, ces problèmes ont été résolus en faisant appel au solveur linéaire du logiciel CPLEX. Nous explorons dans cette section des méthodes alternatives pour résoudre les relaxations en chaque nœud, sans faire appel à un solveur externe.

CPLEX étant une implantation de la méthode du simplexe déjà très performante, il y a peu d'espoir d'obtenir de meilleures performances avec une autre implantation de cette méthode. Dans cette section nous étudions une approche de pénalité utilisant diverses stratégies pour résoudre les sous-problèmes pénalisés que nous formulons ci-dessous.

Dans tous les cas, nous tirons avantage de la séparabilité du problème relaxé, avec d'une part les variables η et t et d'autre part les variables x et y . On a donc les problèmes :

(PRIM) :

$$\begin{aligned}
 \min_{x, y} \quad & cx + dy \\
 \text{s.l.c} \quad & Cx \leq u \\
 & Ax + By = b
 \end{aligned}$$

¹Afin de simplifier la présentation, nous allons utiliser cette formulation «à la racine» dans la présentation des méthodes de résolution et préciserons subséquemment comment intégrer ces méthodes dans le processus d'énumération implicite.

$$x, y \geq 0.$$

et

(DUAL) :

$$\begin{aligned} \max_{t, \eta} \quad & \eta b \\ \text{s.l.c} \quad & \eta A - tD \leq c \quad (v) \\ & \eta B \leq d \quad (w). \end{aligned}$$

Le dual du problème DUAL est

(DUAL-D) :

$$\begin{aligned} \min_{v, w} \quad & cv + dw \\ \text{s.l.c} \quad & Av + Bw = b \quad (\eta) \\ & Dv = 0 \quad (t) \\ & v, w \geq 0. \end{aligned}$$

On remarquera que les problèmes (PRIM) et (DUAL-D) possèdent des structures très semblables et ne diffèrent que par les contraintes de capacité $Cx \leq u$ dans le problème «primal» et les contraintes de prix $Dv = 0$ dans le problème «dual». L'idée développée dans cette section est de pénaliser ces contraintes et d'utiliser une méthode commune permettant de résoudre à la fois le sous-problème pénalisé primal et son équivalent dual.

6.5.1 Méthode de pénalité - cadre général

Soient

$$z_P(x, y) = cx + dy \quad \text{et} \quad z_D(v, w) = cv + dw$$

les objectifs initiaux des problèmes primal et dual. Considérons les fonctions de pénalité quadratiques

$$\psi_P(x) = \frac{1}{2} \|\max\{0, Cx - u\}\|^2 \quad \text{et} \quad \psi_D(v) = \frac{1}{2} \|Dv\|^2.$$

Soit

$$X = \left\{ (x, y) \mid Ax + By = b; x, y \geq 0 \right\}.$$

Les problèmes pénalisés que nous devons résoudre sont

$$\text{(PRIM-PEN)} : \min_{(x,y) \in X} \varphi_P(x,y) = z_P(x,y) + \mu \psi_P(x)$$

et

$$\text{(DUAL-PEN)} : \min_{(v,w) \in X} \varphi_D(v,w) = z_D(v,w) + \mu \psi_D(v)$$

où μ est un scalaire positif.

La méthode de pénalité générique est la suivante :

Soit le problème initial

$$\min \{ \varphi(x) = z(x) + \mu \psi(x) \mid x \in X \}.$$

- Initialisation

Poser $k \leftarrow 0$ et soit le scalaire $\epsilon > 0$. Choisir une solution admissible initiale x_0 , un paramètre de pénalité $\mu_0 > 0$ et un scalaire $\delta > 1$.

- Boucle principale

1. En partant du point x_k , calculer

$$x_{k+1} \in \arg \min_{x \in X} \varphi(x).$$

2. Si $\mu_k \psi(x_{k+1}) < \epsilon$, terminer.

3. $\mu_{k+1} \leftarrow \delta \mu_k$, $k \leftarrow k + 1$. Aller en 1.

6.5.2 Résolution des sous-problèmes par la méthode de Frank-Wolfe

Nous choisissons de linéariser les fonctions objectif pénalisées primale et duale (i.e. φ_P et φ_D) dans le but de résoudre les sous-problèmes pénalisés à l'aide de la méthode de Frank-Wolfe.

Nous avons

$$\begin{aligned} \nabla \varphi_P(x,y) &= \nabla z_P(x,y) + \mu \begin{pmatrix} \nabla \psi_P(x) \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} c \\ d \end{pmatrix} + \mu \begin{pmatrix} C' \cdot \max\{0, Cx - u\} \\ 0 \end{pmatrix} \end{aligned}$$

et

$$\begin{aligned}\nabla\varphi_P(x, y) &= \nabla z_D(v, w) + \mu \begin{pmatrix} \nabla\psi_D(v) \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} c \\ d \end{pmatrix} + \mu \begin{pmatrix} D'Dv \\ 0 \end{pmatrix},\end{aligned}$$

où ' dénote l'opérateur de transposition (dont l'emploi est ici inévitable).

La méthode de Frank-Wolfe procède de façon itérative. Supposons qu'à l'itération $l - 1$ nous ayons obtenu une solution admissible (x_{l-1}, y_{l-1}) du sous-problème primal. À l'itération l , le problème de Frank-Wolfe que nous devons résoudre est

(FW-PRIM(μ) $_l$) :

$$\begin{aligned}\min_{x, y} \quad & (c + \mu C' \cdot \max\{0, Cx_{l-1} - u\})x + dy \\ \text{s.l.c} \quad & Ax + By = b \\ & x, y \geq 0.\end{aligned}$$

De façon similaire, soit (v_{l-1}, w_{l-1}) une solution admissible du sous-problème dual obtenue à l'itération $l - 1$. Le problème de Frank-Wolfe que nous devons résoudre à l'itération l est

(FW-DUAL(μ) $_l$) :

$$\begin{aligned}\min_{v, w} \quad & (c + \mu D'Dv_{l-1})v + dw \\ \text{s.l.c} \quad & Av + Bw = b \\ & v, w \geq 0.\end{aligned}$$

(FW-PRIM(μ) $_l$) et (FW-DUAL(μ) $_l$) peuvent être perçus comme des problèmes de plus courts chemins avec capacité infinie. Dans notre contexte, les chemins sont préalablement énumérés. On peut donc facilement résoudre ces problèmes en identifiant, pour chaque groupe sur chaque marché, le chemin de coût minimum et affecter la totalité de la demande correspondante sur ce chemin. Ces sous-problèmes de Frank-Wolfe peuvent ainsi être résolus en temps linéaire.

Recherche linéaire

Soit (x, y) une solution courante du problème primal et (\bar{x}, \bar{y}) la nouvelle solution obtenue en résolvant le sous-problème de Frank-Wolfe formulé ci-dessus. La direction de descente à cette

itération est ainsi $(p, q) = (\bar{x} - x, \bar{y} - y)$. De façon analogue, dans le cas dual, cette direction de descente est $(r, s) = (\bar{v} - v, \bar{w} + -w)$. Soient λ et ν les pas optimaux dans les sous-problèmes primal et dual, respectivement. Nous devons donc résoudre les problèmes de recherche linéaire

$$\min_{\lambda \in [0,1]} \zeta_P(\lambda) = \varphi_P \left(\begin{pmatrix} x \\ y \end{pmatrix} + \lambda \begin{pmatrix} p \\ q \end{pmatrix} \right) \quad \text{et} \quad \min_{\nu \in [0,1]} \zeta_D(\nu) = \varphi_D \left(\begin{pmatrix} v \\ w \end{pmatrix} + \nu \begin{pmatrix} r \\ s \end{pmatrix} \right).$$

Il existe diverses façons de résoudre ces problèmes à une seule variable. Nous présentons deux méthodes, une exacte et une inexacte, pour résoudre le problème de recherche linéaire. Cette présentation est tirée de [BSS93].

- Recherche par section dorée (recherche exacte)

Cette méthode permet de résoudre le problème de recherche linéaire avec une précision désirée ϵ pour une fonction ζ strictement quasi-convexe. À chaque itération k , elle réduit la taille d'un intervalle d'incertitude $[a_k, b_k]$ que l'on sait contenir le minimum de la fonction ζ jusqu'à ce que $\|b_k - a_k\| < \epsilon$. Soit $\sigma = \frac{\sqrt{5}-1}{2} \approx 0.618$ (il s'agit du «nombre d'or»). La méthode procède comme suit :

- **Initialisation** Soit $[a_1, b_1]$ l'intervalle initial. $l_1 \leftarrow a_1 + (1 - \sigma)(b_1 - a_1)$ et $m_1 \leftarrow a_1 + \sigma(b_1 - a_1)$. Évaluer ζ aux points l_1 et m_1 .
- **Boucle principale**
 1. Si $\|b_k - a_k\| < \epsilon$, arrêter.
 2. Si $\zeta(l_k) > \zeta(m_k)$, alors $a_{k+1} \leftarrow l_k$, $b_{k+1} \leftarrow b_k$, $l_{k+1} \leftarrow m_k$ et $m_{k+1} \leftarrow a_{k+1} + \sigma(b_{k+1} - a_{k+1})$. Évaluer ζ en m_{k+1} . Aller en 4.
 3. Sinon, $a_{k+1} \leftarrow l_k$, $b_{k+1} \leftarrow m_k$, $m_{k+1} \leftarrow l_k$ et $m_{k+1} \leftarrow a_{k+1} + (1 - \sigma)(b_{k+1} - a_{k+1})$. Aller en 4.
 4. $k \leftarrow k + 1$. Retourner en 1.

À chaque itération, une seule évaluation de la fonction ζ est nécessaire et on réduit la taille de l'intervalle d'incertitude d'un facteur σ . La taux de réduction à l'itération k est donc σ .

- Méthode d'Armijo (recherche inexacte)

La méthode d'Armijo se veut une façon pratique de procéder à une recherche linéaire sans devoir résoudre à l'optimalité un second problème d'optimisation (en une seule variable). Supposons qu'au point x nous cherchions à minimiser la fonction φ dans la direction d . Soit

$$\zeta(\lambda) = \varphi(x + \lambda d).$$

La règle dépend de deux paramètres que nous dénotons γ et ϵ . Soit de plus la fonction

$$\hat{\zeta}(\lambda) = \zeta(0) + \lambda\epsilon\dot{\zeta}(0),$$

où $\dot{\zeta}$ est la dérivée première de la fonction ζ . La règle est la suivante : on choisit un pas $\bar{\lambda}$. Si $\zeta(\bar{\lambda}) \leq \hat{\zeta}(\bar{\lambda})$, alors on trouve le plus grand entier $r \geq 0$ tel que $\zeta(\gamma^r \bar{\lambda}) \leq \hat{\zeta}(\gamma^r \bar{\lambda})$. Le pas retenu est $\gamma^r \bar{\lambda}$. Par contre, si $\zeta(\bar{\lambda}) > \hat{\zeta}(\bar{\lambda})$, alors on trouve le plus grand entier $r \geq 0$ tel que $\zeta(\bar{\lambda}/\gamma^r) \leq \hat{\zeta}(\bar{\lambda}/\gamma^r)$. Le pas retenu est alors $\bar{\lambda}/\gamma^r$.

La méthode d'Armijo permet de procéder à une recherche inexacte avec une précision suffisante pour garantir la convergence globale de l'algorithme. Cependant, elle repose sur un paramètre γ qui doit lui-même être étalonné pour assurer une bonne performance. Comme nous travaillons uniquement avec des fonctions strictement convexes dont l'évaluation est relativement peu coûteuse, nous avons choisi d'utiliser exclusivement la recherche exacte par section dorée dans les résultats numériques présentés ci-dessous.

Critère d'arrêt

À chaque itération de la méthode de pénalité, i.e. pour une valeur du paramètre de pénalité μ donnée, la fonction φ que nous cherchons à minimiser est convexe. L'approximation linéaire, sur laquelle est basée la méthode de Frank-Wolfe sous-estime donc cette fonction. Soit x le point courant et y le nouveau point. Nous avons

$$\varphi(y) \geq \tilde{\varphi}(y) = \varphi(x) + \nabla\varphi(x)'(y - x),$$

où $\tilde{\varphi}(y)$ dénote l'approximation linéaire de φ en y . En d'autres mots, $\tilde{\varphi}(y)$ est une borne inférieure sur la valeur optimale de la fonction φ . Un critère d'arrêt couramment utilisé est la valeur (relative) de l'erreur d'approximation, i.e. l'algorithme s'arrête lorsque

$$\frac{\varphi(y) - \tilde{\varphi}(y)}{|\tilde{\varphi}(y)|} < \epsilon,$$

où ϵ est un paramètre donné. Un critère alternatif de type heuristique mesure la dérivée directionnelle (relative) au point courant et termine l'algorithme lorsque celle-ci est arbitrairement près de zéro :

$$\left| \frac{\nabla\varphi(x)'(y - x)}{\varphi(x)} \right| < \epsilon.$$

L'idée sur laquelle repose ce dernier critère est de terminer l'algorithme lorsque la décroissance de la fonction objectif à une itération donnée n'atteint pas un certain seuil de tolérance ϵ . Bien qu'en

pratique cette quantité tend généralement à décroître, rien ne garantit que cette décroissance soit monotone. Il se peut donc que ce critère termine l'algorithme «avant l'heure», d'où son caractère heuristique.

Résultats numériques

Nous illustrons la performance de l'approche de pénalité lorsque la méthode de Frank-Wolfe est utilisée pour résoudre les sous-problèmes. Ces tests furent réalisés sur la relaxation initiale (i.e. correspondant au nœud 0 de l'arbre d'énumération) de l'instance 100. Nous présentons séparément les résultats pour les sous-problèmes primal et dual. Notons que les solutions exactes de ces sous-problèmes ont pour valeur 271 500.93 et 550 205.39. Le solveur linéaire du logiciel CPLEX résout conjointement ces deux sous-problèmes en un temps de 50 millièmes de seconde.

Afin d'appliquer la méthode de pénalité, nous devons disposer d'un point initial. Pour les tests qui suivent, ce point a été obtenu en assignant la totalité du flot sur les vols de la compétition. La prochaine section discute de façons d'obtenir un bon point initial. Les tableaux de résultats indiquent le nombre d'itérations «majeures» (i.e. de la méthode de pénalité) et «mineures» (i.e. de la méthode utilisée pour résoudre les sous-problèmes, dans ce cas-ci Frank-Wolfe). Pour chaque test, la méthode a été interrompue après 10 itérations majeures si la solution optimale n'a pas été obtenue. Dans les tableaux de résultats, nous dénoterons par z_0 la valeur de la solution initiale, z_* la valeur de la solution finale, ψ_* la valeur de la fonction de pénalité à la solution finale et $\mu \psi_*$ le poids du terme de pénalité à la solution finale.

Nous débutons par illustrer aux tableaux 6.7 et 6.8 l'impact de la valeur initiale du facteur de pénalité μ . C'est cette valeur qui apparaît à la première colonne de ces tableaux.

μ	δ	ϵ	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
10.0	2.0	0.01	1 055 281.75	8	478	291 105.53	30.65	410.86	30.68
1.0	2.0	0.01	1 055 281.75	10	168	276 789.85	17.08	874.46	11.55
0.1	2.0	0.01	1 055 281.75	5	78	272 917.87	1.97	1006.84	5.70

TAB. 6.7 – Impact du paramètre de pénalité μ sur le sous-problème primal (avec FW).

Les tableaux 6.9 et 6.10 illustrent l'impact du paramètre δ .

À la lumière de ces résultats, on remarque d'abord que le sous-problème dual est plus facile à

μ	δ	ϵ	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
10.0	2.0	0.01	1 055 281.75	9	94	550 137.63	0.0067	34.30	6.59
100.0	2.0	0.01	1 055 281.75	7	59	550 191.03	0.00058	7.42	4.46
1000.0	2.0	0.01	1 055 281.75	7	12	550 205.39	0.0	0.0	1.33

TAB. 6.8 – Impact du paramètre de pénalité μ sur le sous-problème dual (avec FW).

μ	δ	ϵ	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
0.1	2.0	0.01	1 055 281.75	5	78	272 917.87	17.08	874.46	5.70
0.1	5.0	0.01	1 055 281.75	10	126	273 640.41	0.050	9844.82	8.61
0.1	10.0	0.01	1 055 281.75	10	193	274 355.69	0.0061	606511.40	12.83

TAB. 6.9 – Impact du paramètre δ sur le sous-problème primal (avec FW).

résoudre que le sous-problème primal. Ceci s'explique par la nature des contraintes pénalisées dans chaque cas, soit $Cx \leq u$ pour le primal et $Dv = 0$ pour le dual. La méthode de pénalité semble mieux se comporter pour des contraintes d'égalité. Avec un jeu de paramètres approprié, il est même possible de trouver la solution optimale du problème dual. C'est ce que nous avons fait avec $\mu = 1000.0$ et $\delta = 2.0$. Dans le cas du sous-problème primal, l'approche de pénalité combinée à la méthode de Frank-Wolfe semble toujours surestimer la valeur de la solution optimale. Aucun jeu de paramètre considéré ne permet de s'approcher à plus de 0.55% de la solution optimale du problème. On constate de plus que le nombre d'itérations nécessaire pour parvenir à la précision désirée est relativement élevé, de sorte que les temps de résolution obtenus ne sont pas du tout compétitifs avec la méthode du simplexe (et son implantation dans CPLEX).

6.5.3 Résolution des sous-problèmes par la méthode de projection

Toujours dans le but de résoudre les relaxations linéaires par pénalité, nous décrivons dans cette section une façon alternative de résoudre les sous-problèmes pénalisés en utilisant la méthode de projection.

Considérons le problème

$$\min_{(x,y) \in X} \varphi(x,y) = z(x,y) + \mu \psi(x)$$

où μ est un scalaire positif. Soit

$$X = \{(x,y) \mid Ax + By = b; x, y \geq 0\}.$$

μ	δ	ϵ	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu\psi_*$	Temps (s)
100.0	2.0	0.01	1 055 281.75	7	59	550 191.03	0.00058	7.42	4.46
100.0	5.0	0.01	1 055 281.75	3	56	550 195.20	0.00394	4.93	3.70
100.0	10.0	0.01	1 055 281.75	2	47	550 191.63	0.00072	7.21	3.57

TAB. 6.10 – Impact du paramètre δ sur le sous-problème dual (avec FW).

(Le développement qui suit étant indépendant des formulations primale et duale, nous omettons les indices P et D dans la notation des fonctions). Nous supposons que la quantité b , qui correspond à la demande dans le modèle, est strictement positive.

Nous choisissons ici de résoudre le problème pénalisé par une méthode de projection du gradient sur l'ensemble admissible X .

Soit (x, y) le point courant. Comme nous sommes en présence d'un problème de minimisation, nous suivons la direction donnée par l'opposé du gradient de la fonction φ au point (x, y) . Soit ω un scalaire non négatif. Ce paramètre représente la longueur du pas dans la direction donnée par l'inverse du gradient. En suivant cette direction, nous obtenons le point

$$(v, w) = (x, y) - \omega \nabla \varphi(x, y),$$

lequel n'est pas nécessairement admissible. Nous devons donc le projeter sur l'ensemble réalisable X afin d'obtenir le nouveau point courant (\bar{x}, \bar{y}) , i.e.

$$(\bar{x}, \bar{y}) = \text{proj}_X(v, w).$$

L'opérateur de projection minimise la distance (selon la norme euclidienne) entre le point (v, w) et l'ensemble X . Nous avons donc

$$(\bar{x}, \bar{y}) \in \arg \min_{(x,y) \in X} \frac{1}{2} \|(x, y) - (v, w)\|^2.$$

La fonction lagrangienne de ce problème est

$$\mathcal{L}(x, y, \lambda) = \frac{1}{2} \|(x, y) - (v, w)\|^2 + \lambda (Ax + By - b),$$

dont le gradient est

$$\nabla_{(x,y)} \mathcal{L}(x, y, \lambda) = \begin{pmatrix} x - v \\ y - w \end{pmatrix} + \begin{pmatrix} \lambda A \\ \lambda B \end{pmatrix}.$$

Les conditions d'optimalité primales-duales du problème de projection prennent ainsi la forme

$$\begin{aligned} \begin{pmatrix} x - v \\ y - w \end{pmatrix} + \begin{pmatrix} \lambda A \\ \lambda B \end{pmatrix} &= 0 \\ Ax + By &= b \\ x, y &\geq 0. \end{aligned}$$

Nous cherchons maintenant à résoudre ce système, i.e. à identifier des valeurs pour les variables x , y et λ qui satisfassent simultanément ces conditions. Notons que, dans notre contexte, le problème lagrangien est séparable par marché et par groupe. Des conditions d'optimalités énumérées ci-dessus, nous pouvons déduire que, pour le marché k ,

$$x_{fg} = \max \{0, v_{fg} - \lambda_k\} = (v_{fg} - \lambda_k)^+$$

où f est l'indice d'un vol du meneur et g l'indice d'un groupe d'usagers et a^+ représente la partie positive de la quantité a . De façon analogue, nous avons

$$y_{fg} = \max \{0, w_{fg} - \lambda_k\} = (w_{fg} - \lambda_k)^+$$

où f est cette fois-ci l'indice d'un vol de la compétition. La contrainte de satisfaction de la demande implique que, pour toute combinaison groupe/marché (dénotés respectivement par les indices g et k), nous avons

$$\begin{aligned} \sum_{f \in F^1(k)} x_{fg} + \sum_{f \in F^2(k)} y_{fg} &= \sum_{f \in F^1(k)} (v_{fg} - \lambda_k)^+ + \sum_{f \in F^2(k)} (w_{fg} - \lambda_k)^+ \\ &= b_{gk}. \end{aligned}$$

Cette dernière équation, qui ne contient qu'une seule variable, soit λ_k , est résolue par une procédure spécialisée que nous décrivons maintenant.

Supposons que sur le marché k , $i = (f, g)$ représente une combinaison vol/groupe d'usagers. Soit m^1 le nombre de ces combinaisons pour les vols du meneur et m^2 le nombre pour les vols de la compétition, avec $n = m^1 + m^2$. Nous allons représenter par a_j les flots non projetés $(v, w) = (x, y) - \omega \nabla \varphi(x, y)$ obtenus en suivant la direction donnée par l'inverse du gradient. Nous avons donc $a_j = v_i$, où $i = 1, 2, \dots, m^1$ dans le cas d'un vol du meneur et $a_j = w_i$, où $i = 1, 2, \dots, m^2$ dans le cas d'un vol de la compétition. En laissant tomber l'indice k pour la variable λ , nous définissons la fonction $h : \mathbb{R} \rightarrow \mathbb{R}$ comme

$$h(\lambda) = \sum_{j=1}^n (a_j - \lambda)^+.$$

Avec cette nouvelle notation et en laissant aussi tomber les indices g et k pour la quantité b , nous sommes en mesure de récrire l'équation précédente comme

$$h(\lambda) = b.$$

Notons que h est une fonction convexe linéaire par morceau dont les abscisses des points de rencontre des morceaux correspondent aux valeurs a_j . À titre d'exemple, la fonction

$$h(\lambda) = (7 - \lambda)^+ + (5 - \lambda)^+ + (3 - \lambda)^+ + (1 - \lambda)^+$$

est illustrée à la figure 6.13.

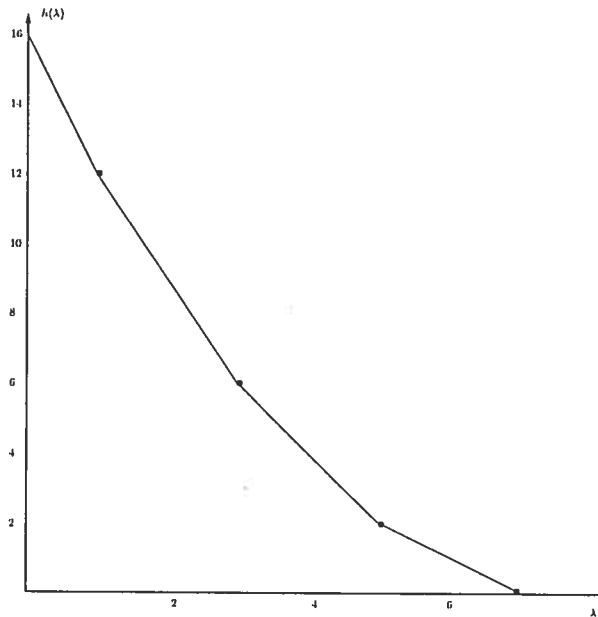


FIG. 6.13 – La fonction $h(\lambda) = (7 - \lambda)^+ + (5 - \lambda)^+ + (3 - \lambda)^+ + (1 - \lambda)^+$.

La procédure de résolution est la suivante :

- **Initialisation**

1. Trier les a_j par ordre décroissant, i.e. de telle sorte que (après réétiquetage, s'il y a lieu)

$$a_1 \geq a_2 \geq \dots \geq a_n.$$

2. Soit $\bar{a}_1 = a_1$ et, pour $j = 2, 3, \dots, n$, $\bar{a}_j = \bar{a}_{j-1} + a_j$.

- **Boucle principale** Soit $\epsilon > 0$. $r \leftarrow 0$ et $s \leftarrow n$.

1. $\lambda_r \leftarrow (\bar{a}_s - b)/s$.

2. Si $h(\lambda_r) - b < \epsilon$, alors terminer car λ_r résout $h(\lambda) = b$ à la précision ϵ . Sinon aller en 3.
3. $s \leftarrow \max \{s' \mid a_{s'} > \lambda_r\}$, $r \leftarrow r + 1$. Retourner en 1.

L'opération de tri à l'étape d'initialisation s'effectue en temps $O(n \log n)$. Comme $h(\lambda)$ est non négative pour tout λ et $b > 0$ par hypothèse, la boucle principale de la procédure est effectuée au plus n fois, i.e. une fois pour chaque «morceau» de la fonction h , jusqu'à ce que l'on identifie le «morceau» qui intersecte la droite d'ordonnée à l'origine b et que l'on calcule le point exact d'intersection sur ce morceau, ce qui se fait en temps $O(1)$. Bref, la procédure totale de résolution de l'équation $h(\lambda) = b$ s'effectue en temps $O(n \log n)$.

Nous illustrons le déroulement de la procédure de résolution en l'appliquant à la fonction $h(\lambda) = (7 - \lambda)^+ + (5 - \lambda)^+ + (3 - \lambda)^+ + (1 - \lambda)^+$ et en posant $b = 3$.

- **Initialisation**

Après le tri, nous avons $a_1 = 7$, $a_2 = 5$, $a_3 = 3$ et $a_4 = 1$. De là, on calcule $\bar{a}_1 = 7$, $\bar{a}_2 = 12$, $\bar{a}_3 = 15$ et $\bar{a}_4 = 16$. Soit $r \leftarrow 0$ et $s \leftarrow 4$. Posons $\epsilon = 0.001$.

- **Boucle principale**

- **Itération 0**

$\lambda_0 \leftarrow (\bar{a}_4 - b)/4 = (16 - 3)/4 = \frac{13}{4}$. On a $h(\lambda_0) = \frac{11}{2}$ et donc $h(\lambda_0) - b = \frac{11}{2} - 3 = \frac{5}{2} \geq \epsilon$. On calcule $s \leftarrow \max \{s' \mid a_{s'} > \frac{11}{2}\} = \max \{1, 2\} = 2$. $r \leftarrow r + 1 = 1 + 1 = 2$ et on va à la prochaine itération.

- **Itération 1**

$\lambda_1 \leftarrow (\bar{a}_2 - b)/2 = (12 - 3)/2 = \frac{9}{2}$. On a $h(\lambda_1) = 3$ et donc $h(\lambda_1) - b = 3 - 3 = 0 < \epsilon$. On termine car $\lambda_1 = \frac{9}{2}$ est la solution de l'équation $h(\lambda) = b$.

La figure 6.14 illustre la progression de la procédure de résolution sur cet exemple.

Une fois le point $(\bar{x}, \bar{y}) = \text{proj}_X(v, w)$ calculé, on procède à une recherche linéaire dans la direction $(p, q) = (x - \bar{x}, y - \bar{y})$. Les techniques utilisées pour effectuer cette recherche de même que les critères d'arrêt sont décrits à la section précédente sur la méthode de Frank-Wolfe.

Résultats numériques

Les tableaux suivants présentent différents résultats numériques obtenus en utilisant l'approche de pénalité avec la méthode de projection du gradient pour résoudre les sous-problèmes. Comme ce fut le cas avec la méthode de Frank-Wolfe, ces résultats furent obtenus en résolvant l'instance

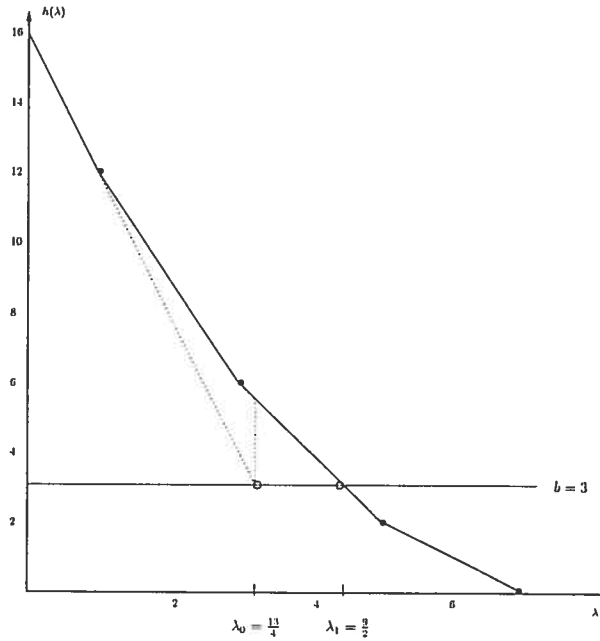


FIG. 6.14 – Exemple de résolution de l'équation $h(\lambda) = b$.

100 au nœud 0. Les symboles utilisés dans ces tableaux sont les mêmes que précédemment.

Les tableaux 6.11 et 6.12 illustrent l'impact du paramètre μ sur la résolution des sous-problèmes primal et dual.

μ	δ	ϵ	ω	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
10.0	2.0	0.01	10.0	1 055 281.75	10	226	292 478.23	0.27	1384.73	38.63
1.0	2.0	0.01	10.0	1 055 281.75	10	41	274 866.07	1.01	517.37	8.13
0.1	2.0	0.01	10.0	1 055 281.75	10	43	272 937.73	8.68	444.60	8.22

TAB. 6.11 – Impact du paramètre μ sur le sous-problème primal (avec Projection).

Ces résultats démontrent que la méthode de projection est, pour une valeur du paramètre ω donnée (dans ce cas-ci $\omega = 10.0$) beaucoup plus sensible à la valeur du facteur de pénalité μ que la méthode de Frank-Wolfe. Ce fait est particulièrement évident dans le cas du sous-problème dual, alors que l'algorithme semble diverger pour des valeurs de μ élevées (de telles valeurs accélèrent la convergence pour la méthode de Frank-Wolfe). L'impact du paramètre ω est mesuré aux tableaux 6.13 et 6.14.

Ces résultats indiquent que la méthode de projection, étant donnée une valeur appropriée du

μ	δ	ϵ	ω	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
10.0	2.0	0.01	10.0	1 055 281.75	9	46	550 126.77	0.0078	39.94	8.05
100.0	2.0	0.01	10.0	1 055 281.75	10	20	561 516.69	1.39	71 363.49	5.13
1000.0	2.0	0.01	10.0	1 055 281.75	10	10	1 052 605.70	0.42	214 456.57	2.47

TAB. 6.12 – Impact du paramètre μ sur le sous-problème dual (avec Projection).

μ	δ	ϵ	ω	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
0.1	2.0	0.01	1000.0	1 055 281.75	10	42	273 185.43	35.58	1821.50	8.37
0.1	2.0	0.01	10.0	1 055 281.75	10	43	272 937.73	8.68	444.60	8.22
0.1	2.0	0.01	1.0	1 055 281.75	10	38	272 332.34	5.12	262.19	7.08

TAB. 6.13 – Impact du paramètre ω sur le sous-problème primal (avec Projection).

paramètre ω , peut obtenir des résultats d'une précision similaire à la méthode de Frank-Wolfe. Les temps de résolution sont cependant supérieurs à ceux obtenus avec cette dernière. L'impact du paramètre ω , qui doit être «étalonné» comme le paramètre μ , ajoute une autre dimension à cette méthode, ce qui en complique l'utilisation en pratique.

6.5.4 Intégration dans le processus d'énumération implicite

Les méthodes de résolution des relaxations linéaires décrites ci-dessus sont destinées à être utilisées dans le processus d'énumération implicite spécialisé illustré précédemment. Cependant, les performances numériques jusqu'à maintenant décevantes de ces méthodes ne justifient pas à ce moment leur application dans notre approche. Toutefois, par souci d'exhaustivité, cette section décrit la façon dont ces méthodes peuvent être intégrées au processus global. Bien que nous ayons illustré deux algorithmes pour résoudre les sous-problèmes linéaires, soit la méthode de Frank-Wolfe et la méthode du gradient projeté, ce qui suit ne dépend pas de l'algorithme

μ	δ	ϵ	ω	z_0	It. maj.	It. min.	z_*	ψ_*	$\mu \psi_*$	Temps (s)
10.0	2.0	0.01	1000.0	1 055 281.75	9	49	550 155.72	0.0052	26.62	9.19
10.0	2.0	0.01	100.0	1 055 281.75	9	45	550 139.05	0.0067	34.30	8.25
10.0	2.0	0.01	10.0	1 055 281.75	9	46	550 126.77	0.0078	39.94	8.05
10.0	2.0	0.01	1.0	1 055 281.75	10	35	575 927.00	0.21	1089.44	7.09

TAB. 6.14 – Impact du paramètre ω sur le sous-problème dual (avec Projection).

utilisé.

Résolution aux nœuds autres que la racine

La présentation des méthodes de résolution des relaxations s'est faite en utilisant la formulation du nœud 0, i.e. dans laquelle aucune contrainte de complémentarité n'a été fixée. Nous devons donc adapter la présentation de ces méthodes à un nœud arbitraire de l'arbre. Au nœud i , soit I_i un ensemble d'indices représentant les contraintes de la forme $\eta A - tD \leq c$ ayant été fixées à égalité aux nœuds antérieurs. Ces indices sont en réalité des couples (f, g) d'indices de vols et de groupes. Nous avons donc

$$\eta_{gk(f)} - t_f = c_{fg}, \quad \forall (f, g) \in I_i.$$

De façon analogue, soit J_i l'ensemble des couples (f, g) correspondant aux contraintes de non négativité sur les flots $x \geq 0$ ayant été fixées à 0. Nous avons ainsi

$$x_{fg} = 0, \quad \forall (f, g) \in J_i.$$

Au nœud i , les sous problèmes que nous devons résoudre prennent la forme

(PRIM(i)) :

$$\begin{aligned} \min_{x,y} \quad & cx + dy \\ \text{s.l.c} \quad & Cx \leq u \\ & Ax + By = b \\ & x, y \geq 0 \\ & x_{fg} = 0 \quad \forall (f, g) \in J_i. \end{aligned}$$

et

(DUAL(i)) :

$$\begin{aligned} \max_{t,\eta} \quad & \eta b \\ \text{s.l.c} \quad & \eta A - tD \leq c \\ & \eta_{gk(f)} - t_f = c_{fg} \quad \forall (f, g) \in I_i \\ & \eta B \leq d. \end{aligned}$$

Ce dernier problème est remplacé par son dual :

(DUAL(i)-D) :

$$\begin{aligned} \min_{v,w} \quad & cv + dw \\ \text{s.l.c} \quad & Av + Bw = b \\ & Dv = 0 \\ & v_{fg} \geq 0 \quad \forall (f, g) \notin J_i \\ & w \geq 0. \end{aligned}$$

Comme décrit précédemment, ces problèmes sont alors résolus en pénalisant les contraintes $Cx \leq u$, dans le cas de (PRIM(i)), et les contraintes $Dv = 0$, dans le cas de DUAL(i)-D.

Identification de la solution initiale

À mesure que nous explorons l'arbre d'énumération, nous pouvons utiliser comme point initial de la méthode de pénalité la solution obtenue au nœud-père. En effet, le problème résolu au nœud-père ne diffère du problème courant que par la fixation d'un certain nombre de contraintes et de variables. Nous pouvons donc penser que la solution obtenue au nœud-père est relativement «près» de la solution au nœud courant et peut constituer un bon point de départ. D'ailleurs, dans un arbre d'énumération où la méthode du simplexe est utilisée pour résoudre les résolutions linéaires, la solution au nœud-père est utilisée comme base initiale au nœud courant (car elle est duale admissible) et la solution optimale est obtenue en effectuant quelques itérations duales.

Cependant, à la racine (i.e. au nœud 0) de l'arbre d'énumération, nous devons calculer une solution initiale qui est ensuite utilisée comme point de départ à la méthode de pénalité. Nous avons intérêt à ce que cette solution initiale soit le plus près possible de la solution optimale.

Dans le cas du problème DUAL(0)-D, il existe une solution optimale analytique. En effet, au nœud 0, aucune des variables de flot v n'a encore été «relâchée». En d'autres mots, toutes ces variables sont contraintes à prendre des valeurs non négatives. Or les contraintes $Dv = 0$ impliquent que la somme des valeurs des variables de flot sur les vols (et les produits, si ceux-ci sont considérés) reliés à un tarif soit nulle. Comme chaque variable v apparaît dans exactement une de ces contraintes, on peut conclure que dans la solution optimale au nœud 0, les variables

v prennent toutes la valeur 0. Le problème que nous devons résoudre peut donc s'écrire comme

(DUAL(i)-D) :

$$\begin{aligned} \min_w \quad & dw \\ \text{s.l.c} \quad & Bw = b \\ & w \geq 0. \end{aligned}$$

Il s'agit, pour chaque groupe d'usager sur chaque marché, de trouver le vol de la compétition ayant un coût perçu total minimal. Comme par hypothèse les vols de la compétition n'ont pas de contrainte de capacité, on peut affecter à ce vol à coût minimal toute la demande du groupe sur le marché. Le pseudo-code suivant illustre cette procédure.

Procédure findInitialPointDual()

```

1:  $v \leftarrow 0$ 
2:  $w \leftarrow 0$ 
3: for all  $k \in K$  do
4:   for all  $g \in G$  do
5:     let  $f \in \arg \min_{f' \in F^2(k)} \{d_{f'g}\}$ 
6:      $w_{fg} \leftarrow b_{gk}$ 
7: return  $(v, w)$ 

```

Dans le cas du sous-problème primal, il n'existe pas de solution analytique au nœud 0. Nous avons implanté deux procédures heuristiques qui répartissent le flot en remplissant à capacité les vols du meneur (en supposant que la demande soit suffisante) en ordre croissant de coût perçu, puis en affectant la demande résiduelle sur les vols de la compétition de coût perçu minimum. La première exploite la structure décomposable du problème et procède à la répartition du flot marché par marché, alors que la seconde considère tous les vols simultanément, en ignorant les marchés. Notons que dans les deux cas, nous obtenons une solution admissible pour le problème non pénalisé, i.e pour laquelle $\psi_P(x) = 0$.

Ces procédures sont décrites par les pseudo-codes suivants.

Procédure findInitialPointPrimalHeuristic1()

```

1:  $x \leftarrow 0$ 
2:  $y \leftarrow 0$ 
3:  $\bar{u} \leftarrow u$ 
4: for all  $k \in K$  do

```

```

5:   for all  $g \in G$  do
6:      $n \leftarrow |F^1(k)|$ 
7:      $\text{costs} \leftarrow 0$  //  $\text{costs}$  est un tableau de dimension  $n$ 
8:      $i \leftarrow 0$ 
9:     for all  $f \in F^1(k)$  do
10:       $\text{costs}[i] \leftarrow (c_{f,g}, f)$  // La première composante est le coût et la seconde l'indice du vol
11:       $i \leftarrow i + 1$ 
12:      $\text{costs.sort}()$  // Le tableau est trié en ordre croissant de coûts
13:      $\delta \leftarrow b_{g,k}$ 
14:     for  $i \leftarrow 0$  to  $n$  do
15:        $f \leftarrow \text{costs}[i][2]$ 
16:        $\theta \leftarrow \infty$ 
17:       for all  $s \in S(f)$  do
18:          $\theta \leftarrow \min\{\theta, \bar{u}_s\}$ 
19:       if  $\theta > 0$  then
20:          $\sigma \leftarrow \min\{\theta, \delta\}$ 
21:          $x_{f,g} \leftarrow \sigma$ 
22:         for all  $s \in S(f)$  do
23:            $\bar{u}_s \leftarrow \bar{u}_s - \sigma$ 
24:            $\delta \leftarrow \delta - \sigma$ 
25:         if  $\delta = 0$  then
26:           break
27:       if  $\delta > 0$  then
28:         let  $f \in \arg \min_{f' \in F^2(k)} \{d_{f',g}\}$ 
29:          $y_{f,g} \leftarrow \delta$ 
30: return  $(x, y)$ 

```

Procédure findInitialPointPrimalHeuristic2()

```

1:  $x \leftarrow 0$ 
2:  $y \leftarrow 0$ 
3:  $\bar{u} \leftarrow u$ 
4:  $\bar{b} \leftarrow b$ 
5:  $n \leftarrow |F^1| \times |G|$ 
6:  $\text{costs} \leftarrow 0$ 
7:  $i \leftarrow 0$ 
8: for all  $f \in F^1$  do
9:   for all  $g \in G$  do

```



```

10:   costs[i] ← (cf,g, f, g)
11:   i ← i + 1
12: costs.sort()
13: for i ← 0 to n do
14:   f ← costs[i][2]
15:   g ← costs[i][3]
16:   θ ← ∞
17:   for all s ∈ S(f) do
18:     θ ← min{θ,  $\bar{u}_s$ }
19:   if θ > 0 then
20:     σ ← min{θ,  $\bar{b}_{f(k),g}$ }
21:     xf,g ← σ
22:      $\bar{b}_{f(k),g}$  ←  $\bar{b}_{f(k),g}$  - σ
23:     for all s ∈ S(f) do
24:        $\bar{u}_s$  ←  $\bar{u}_s$  - σ
25:   for all k ∈ K do
26:     for all g ∈ G do
27:       if  $\bar{b}_{f(k),g}$  > 0 then
28:         let f ∈ arg minf' ∈ F2(k) {df',g}
29:         yf,g ←  $\bar{b}_{f(k),g}$ 
30: return (x, y)

```

Le tableau 6.15 illustre la performance de cette procédure lorsqu'utilisée pour identifier la solution initiale du sous-problème primal. La colonne intitulée «Différence» indique l'écart relatif entre la solution obtenue par l'heuristique et la solution optimale.

Instance	Simplexe (CPLEX)		Heuristique 1			Heuristique 2		
	Valeur	Temps (s)	Valeur	Temps (s)	Différence(%)	Valeur	Temps (s)	Différence(%)
100	271 500.93	0.02	271501.07	< 0.0055 ¹	0.000052	285 907.34	<0.0055	5.67
101	1 561 162.23	0.05	2 390 345.99	0.02	53.11	1 602 529.92	0.01	2.65
102	2 878 929.47	0.1	4 797 494.06	0.02	66.64	2 944 360.33	0.03	2.27
103	385 056.91	0.05	387 603.65	0.01	0.66	402 140.62	0.01	4.43
104	2 173 194.86	0.15	3 095 382.70	0.02	42.43	2 247 434.05	0.06	3.41
105	4 038 818.88	0.28	6 197 216.10	0.03	53.44	4 156 216.86	0.12	2.91
106	2 189 999.68	0.54	2 520 466.50	0.02	15.09	2 377 976.65	0.08	8.58
107	10 681 271.59	2.25	15 174 803.27	0.11	42.07	11 608 141.07	0.38	8.68
108	19 565 577.89	3.8	28 577 646.30	0.19	46.1	20 844 866.34	0.65	6.54
110	95 247.38	0.05	96 834.15	0.01	1.67	100 201.45	0.01	5.20
111	119 198.96	0.02	124 161.79	< 0.0055	4.16	123 408.79	0.01	3.53

TAB. 6.15 – Performance des heuristiques primales pour trouver un point initial au problème primal.

Nous pouvons remarquer que l'heuristique 1 fonctionne bien sur des instances de petite taille, notamment celles avec peu de groupes d'utilisateurs. Cependant, sa performance est mauvaise sur des instances de moyenne et grande taille, avec une première solution souvent à plus de 50% de la solution optimale. L'heuristique 2 s'avère beaucoup plus stable, avec une marge d'erreur moyenne d'environ 4.9% sur l'ensemble des instances.

Calcul des multiplicateurs

Au nœud i de l'arbre d'énumération, nous résolvons le problème (DUAL(i)-D), dont les variables de décision sont v et w dans les formulations que nous avons présentées. Une fois la solution optimale de ce problème obtenue, nous devons à partir de celle-ci calculer la valeur des variables duales aux contraintes de (DUAL(i)-D), soit η et t , afin de les utiliser dans les opérations de branchement de l'arbre d'énumération.

Étant donnée une solution (v, w) de (DUAL(i)-D), la procédure suivante calcule les valeurs des variables η et t associées.

Procédure computeDualValues(x, y)

```

1: for all  $k \in K$  do
2:    $\theta \leftarrow \infty$ 
3:   for all  $f \in F^2$  do
4:      $\theta \leftarrow \min\{\theta, d_{f, g|G}\}$ 
5:      $\eta_{g|G, k} \leftarrow \theta$ 
6:     for  $i \leftarrow |G| - 1$  downto 1 do
7:        $\delta \leftarrow \max_{f \in F^1(k)} \{c_{f, g_i} - c_{f, g_{i+1}} \mid v_{f, g_i} \neq 0 \wedge v_{f, g_{i+1}} \neq 0\} \cup \{-\infty\}$ 
8:       if  $\delta > -\infty$  then
9:          $\theta \leftarrow \eta_{g_{i+1}, k} + \delta$ 
10:      else
11:         $\theta \leftarrow \min_{f \in F^2(k)} \{d_{f, g_i}\}$ 
12:         $\eta_{g_i, k} \leftarrow \theta$ 
13: for all  $k \in K$  do

```

¹Sous l'architecture Intel i386, l'horloge du système possède une fréquence d'environ 18Hz, ce qui signifie qu'on enregistre un «tic» tous les 55 millisecondes. Un programme dont la durée d'exécution est inférieure s'exécute possiblement entre deux «tics» consécutifs, ce qui signifie que le temps d'exécution affiché dans ce cas peut être nul. En réalité, il est compris entre 0 et 55 millisecondes. Afin d'obtenir une précision supérieure, il faut utiliser directement l'horloge du processeur et donc programmer en assembleur, ce que nous n'avons pas fait.

```

14: for all  $f \in F^1(k)$  do
15:    $t_f \leftarrow \max_{g \in G} \{\eta_{g,k} - c_{f,g} | v_{f,g} \neq 0\}$ 
16: return  $(\eta, t)$ 

```

Cette procédure peut être interprétée en considérant la formulation (DUAL), i.e.

(DUAL) :

$$\begin{array}{ll}
 \max_{t, \eta} & \eta b \\
 \text{s.l.c} & \eta A - tD \leq c \\
 & \eta B \leq d.
 \end{array}$$

Nous calculons d'abord les valeurs des variables η , en commençant par le groupe $g_{|G|}$, i.e. le groupe qui, selon notre hypothèse, possède la plus faible propension à payer. Rappelons que les variables η , qui sont les multiplicateurs duaux des contraintes de satisfaction de la demande du problème de second niveau du modèle initial, représentent, dans une solution optimale, le coût perçu total que les usagers d'un groupe sont prêts à déboursier sur un marché donné. Pour le groupe $g_{|G|}$, cette valeur dépend uniquement des vols de la compétition et est conditionnée par les contraintes $\eta B \leq b$ du modèle (DUAL). Ainsi, pour chaque marché k , nous calculons cette valeur pour le groupe $g_{|G|}$ (lignes 3-5). Les valeurs η des autres groupes peuvent ensuite être calculées à partir de cette valeur de base. Si un groupe g_i (où $1 \leq i \leq |G| - 1$) partage au moins un vol avec le groupe g_{i+1} , c'est-à-dire si $v_{f,g_i} \neq 0$ et $v_{f,g_{i+1}} \neq 0$, alors le coût perçu maximal du groupe g_i sur le marché k est le coût perçu maximal du groupe g_{i+1} plus la différence de désutilité maximale sur les vols que ces groupes utilisent en commun (ligne 7). Sinon, le coût perçu maximal du groupe g_i est conditionné par les vols de la compétition (ligne 11).

Une fois les valeurs des variables η calculées, nous traitons les variables t . Celles-ci dépendent de η et des contraintes $\eta A - tD \leq c$. Comme nous ne considérons pas les produits, le prix du vol f est la différence maximale, parmi tous les groupes empruntant ce vol, entre le coût perçu maximal du groupe et la désutilité encourue sur le vol f (ligne 15).

6.6 Généralisation à plusieurs critères

Dans le but d'alléger la présentation et de faciliter la validation des méthodes de résolution spécialisées, nous n'avons jusqu'à maintenant considéré qu'un seul critère comportemental dans la composition du coût perçu d'un vol, soit la durée du trajet. Nous illustrons dans cette section

comment la méthode d'énumération implicite spécialisée que nous avons présentée se généralise à un nombre arbitraire de critères. Pour des raisons pratiques, nous nous intéressons particulièrement au cas à deux critères comportementaux, soient la durée du trajet et la qualité de service. Le premier, associé au vol, sera comme auparavant dénoté par \mathcal{D}_f , tandis que le second, associé au produit tarifaire, sera dénoté par \mathcal{Q}_p . Il s'agit de la désutilité associée à un produit tarifaire p . Sur un marché k donné, les paramètres décrivant le comportement des usagers d'un groupe g seront représentés par un vecteur $\theta_{gk} = (\alpha_{gk}, \beta_{gk})$. Afin d'utiliser une notation vectorielle plus compacte, le symbole \mathcal{A} représentera le vecteur dont les composantes sont les valeurs numériques des attributs considérés. Par exemple, dans le cas à deux critères, nous avons $\mathcal{A}_{fp} = (\mathcal{D}_f, \mathcal{Q}_p)$.

Supposons que m_k groupes soient présents sur le marché k . Pour généraliser notre hypothèse fondamentale, nous allons supposer qu'il existe sur ce marché un ordonnancement strict, au sens vectoriel, des paramètres comportementaux des différents groupes, i.e que la relation

$$\theta_{1k} > \theta_{2k} > \dots > \theta_{m_k k}$$

soit respectée.

Cette hypothèse implique que si un groupe possède un indice inférieur à un autre, alors les usagers de groupe accordent une plus grande valeur monétaire à *tous* les critères considérés, ce qui semble restrictif à première vue. En effet, sous cette hypothèse, nous ne pouvons pas composer de groupe qui par exemple accorderait une grande valeur à la durée du trajet mais une petite valeur à la qualité de service. Cependant, en pratique, les valeurs qu'accordent les usagers aux différents critères sont fortement corrélées, de sorte que cette hypothèse n'est pas un sérieux handicap lorsque vient le temps de modéliser le comportement de la clientèle.

Notre principal défi est de généraliser le lemme 6.2 afin de démontrer que les solutions admissibles possèdent aussi une structure particulière dans le cas multi-critère. Comme nous considérons plus d'un critère, nous ne pouvons plus utiliser les indices des vols (ou des produits) comme nous le faisons précédemment pour baser notre raisonnement. Nous utiliserons plutôt des «combinaisons» d'objets que nous dénoterons par des uplets d'indices. Par exemple, dans le cas à deux critères, l'uplet $i = (f, p)$ représente la combinaison du vol f et du produit p . Selon cette notation, le vecteur d'attribut correspondant est $\mathcal{A}_i = (\mathcal{D}_f, \mathcal{Q}_p)$. La généralisation du lemme est la suivante.

LEMME 6.6

Dans toute solution admissible du modèle, sur un marché k donné, si le groupe g est présent

sur la combinaison i et le groupe g' est présent sur la combinaison i' telle que $\mathcal{A}_i < \mathcal{A}_{i'}$, alors $g \leq g'$.

Preuve Supposons, par contradiction, que $g > g'$. Les contraintes (5.8) (admissibilité duale) et (5.12) (complémentarité) impliquent que

$$\begin{aligned}\eta_g^k &= t_i^k + \theta_g^k \mathcal{A}_i \\ \eta_g^k &\leq t_{i'}^k + \theta_g^k \mathcal{A}_{i'} \\ \eta_{g'}^k &\leq t_i^k + \theta_{g'}^k \mathcal{A}_i \\ \eta_{g'}^k &= t_{i'}^k + \theta_{g'}^k \mathcal{A}_{i'}.\end{aligned}$$

Nous avons donc

$$\begin{aligned}t_i^k + \theta_g^k \mathcal{A}_i &\leq t_{i'}^k + \theta_g^k \mathcal{A}_{i'} \quad \text{et} \\ t_{i'}^k + \theta_{g'}^k \mathcal{A}_{i'} &\leq t_i^k + \theta_{g'}^k \mathcal{A}_i.\end{aligned}$$

En additionnant ces deux inégalités, nous obtenons

$$\theta_g^k \mathcal{A}_i + \theta_{g'}^k \mathcal{A}_{i'} \leq \theta_g^k \mathcal{A}_{i'} + \theta_{g'}^k \mathcal{A}_i,$$

qui peut se réarranger comme

$$(\theta_{g'}^k - \theta_g^k) \mathcal{A}_{i'} \leq (\theta_{g'}^k - \theta_g^k) \mathcal{A}_i.$$

Nous avons supposé que $g > g'$, ce qui signifie que $\theta_g^k < \theta_{g'}^k$, cette dernière inégalité étant interprétée au sens vectoriel. La quantité vectorielle $\theta_{g'}^k - \theta_g^k$ est ainsi strictement positive. La dernière inégalité nous permet donc de conclure que $\mathcal{A}_{i'} \leq \mathcal{A}_i$, une contradiction. \square

Nous exploitons ce résultat comme précédemment, c'est-à-dire en tenant compte de la structure particulière des solutions admissibles lors des opérations de branchement et de fixation des variables binaires en chaque nœud de l'arbre d'énumération. Cependant, comme nous travaillons avec plus d'un critère, nous ne pouvons pas utiliser les indices des objets considérés et maintenir un ordonnancement sur ces indices. Il faut en fait considérer explicitement les valeurs des attributs associés à chaque objet. La présentation et l'implantation de l'algorithme sont donc plus complexes. Pour cette raison, nous ne l'illustrerons pas en détail comme nous l'avons fait dans le cas à un seul critère. Cependant, les idées fondamentales demeurent les mêmes.

À titre de comparaison, nous illustrons au tableau 6.16 les performances obtenues par CPLEX pour résoudre la formulation à variable entière mixte à deux critères (temps et qualité de service).

Comme précédemment, les temps limites de résolution ont été fixés à six heures pour les instances 100-102, 110 et 111, douze heures pour les instances 103-105 et vingt-quatre heures pour les instances 106-108. Nous ne connaissons pas la qualité de la solution obtenue par CPLEX dans le cas des instances 103, 110 et 111. Ces solutions ont une valeur supérieure à celle des solutions identifiées par notre algorithme, mais CPLEX n'a pas été en mesure de parcourir entièrement l'arbre d'énumération. Nous ne savons donc pas si ces solutions sont optimales. Dans tous les autres cas, la meilleure solution de CPLEX obtenue à l'intérieur du temps alloué était sous-optimale.

Instance	Première solution entière			Meilleure solution entière			
	Valeur	Nœud	Temps (s)	Valeur	Nœud	Temps (s)	Gap(%)
100	404 667.96	0+	0.94	423 921.82	21 100	298.03	6.57
101	567 510.42	0+	4.76	588 606.41	23 255	1 735.55	11.17
102	599 456.95	0+	44.15	599 456.95	0+	44.15	14.97
103	565 587.77	0+	9.66	608 098.88	37 090	5 444.14	9.76
104	875 429.28	0+	31.61	915 189.40	98 652	28 566.84	14.33
105	957 775.10	0+	94.90	957 775.10	0+	94.90	15.31
106	1 574 342.74	0+	13.82	1 658 675.09	7 539	3 555.49	19.13
107	2 541 146.63	0+	65.83	2 541 146.63	0+	65.83	23.33
108	2 783 632.28	0+	351.89	2 783 632.28	0+	351.89	22.02
110	85 223.42	0+	1.70	106 110.20	55 026	1 305.60	6.92
111	113 046.95	0+	0.95	133 925.86	37 690	945.72	1.81

TAB. 6.16 – Résultats obtenus avec CPLEX sur la formulation (BAFO-MIP $_{ii}$) avec deux critères.

6.6.1 Comparaisons avec CPLEX

Comme nous l'avons fait dans le cas à un critère, nous présentons une étude comparative entre les résultats obtenus avec d'une part le solveur CPLEX et d'autre part l'algorithme spécialisé dans le cas multi-critère. Les conditions sous lesquelles ces tests ont été effectués sont les mêmes que précédemment. Le tableau 6.17 présente ces résultats, lesquels sont encore une fois résumés de façon graphique aux pages suivantes, en utilisant les mêmes symboles que précédemment.

Instance	CPLEx			Algo. spéc.			Rapport (%)
	Valeur	Nœud	Temps (s)	Gap(%)	Valeur	Nœud	
100	423 921.82	21 100	298.03	6.57	424 585.32	116 191	4 710.42
101	588 606.41	23 255	1 735.55	11.17	611 887.72	3 529	464.82
102	599 456.95	0+	44.15	14.97	636 236.67	368	89.12
103	608 098.88	37 090	5 444.14	9.76	604 589.49	24 098	2742.92
104	915 189.40	98 652	28 566.84	14.33	973 632.87	1 589	505.39
105	957 775.10	0+	94.90	15.31	1 015 754.63	942	538.38
106	1 658 675.09	7 539	3 555.49	19.13	1 728 571.46	5 435	17 902.75
107	2 541 146.63	0+	65.83	23.33	2 800 543.93	537	6 581.20
108	2 783 632.28	0+	351.89	22.02	3 017 210.62	727	13 903.26
110	106 110.20	55 026	1 305.60	6.92	104 403.46	302 244	16 927.89
111	133 925.86	37 690	945.72	1.81	133 557.46	236 689	4 854.41

TAB. 6.17 – Comparaisons entre CPLEx et l'algorithme spécialisé.

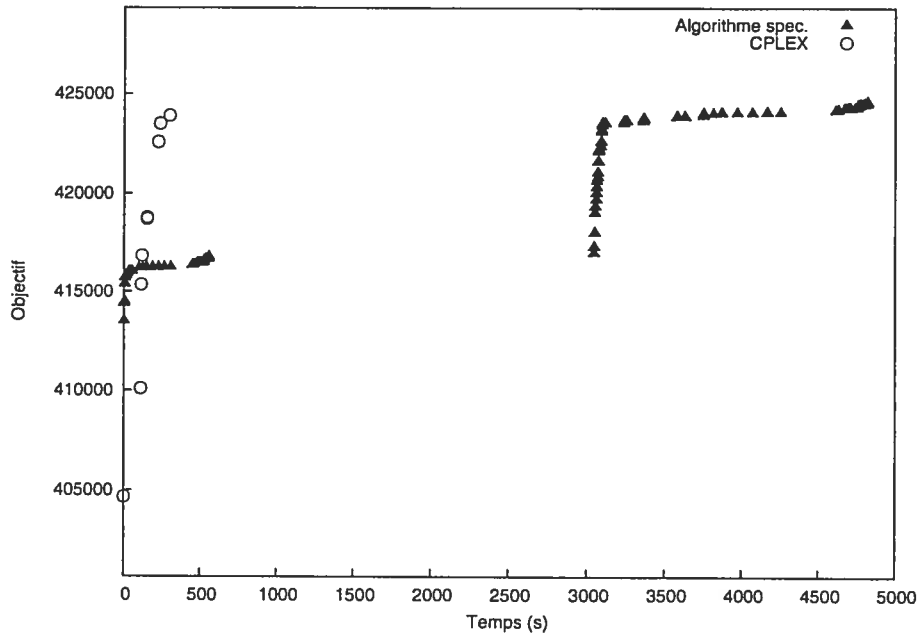


FIG. 6.15 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 100 (avec produits).

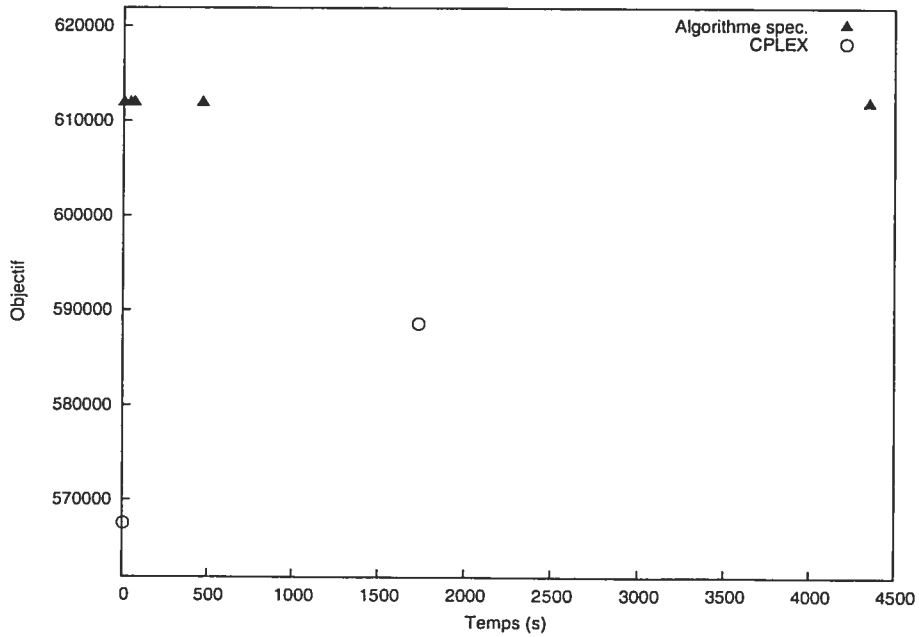


FIG. 6.16 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 101 (avec produits).

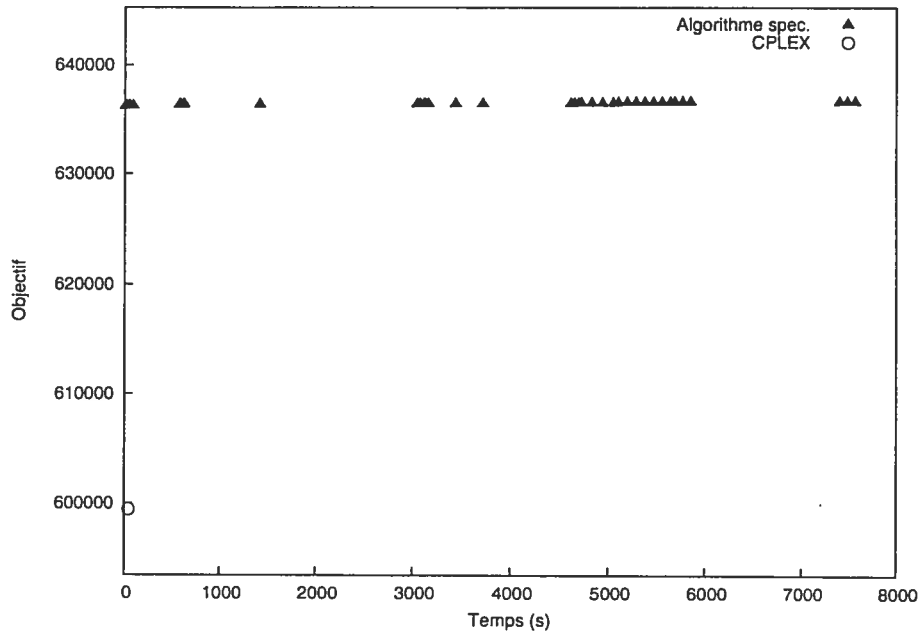


FIG. 6.17 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 102 (avec produits).

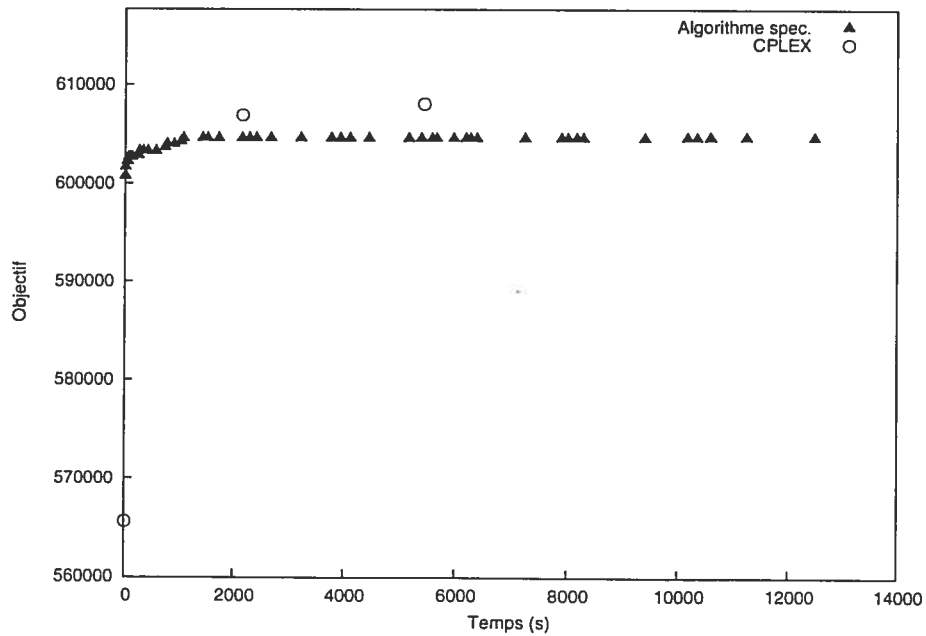


FIG. 6.18 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 103 (avec produits).

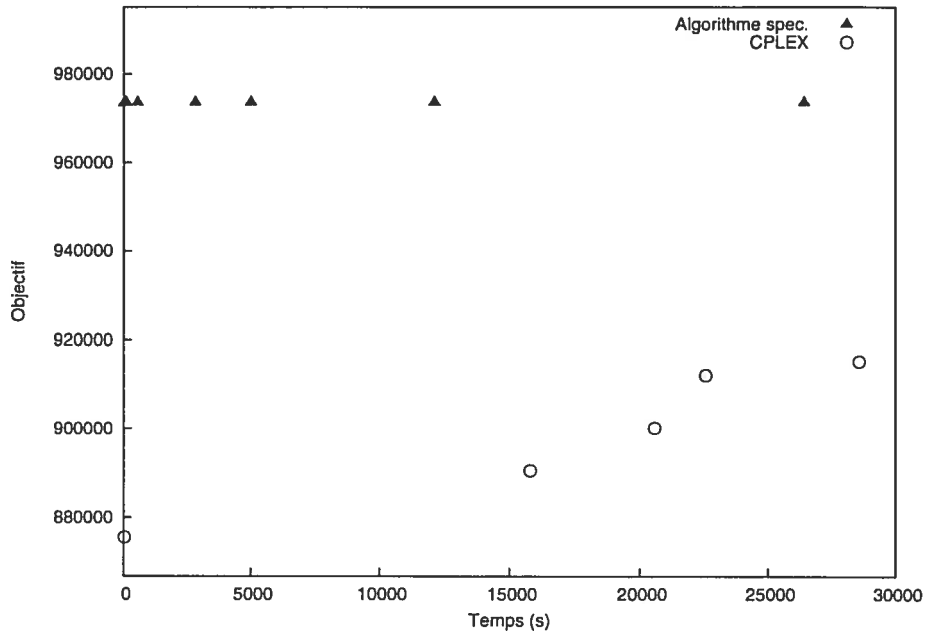


FIG. 6.19 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 104 (avec produits).

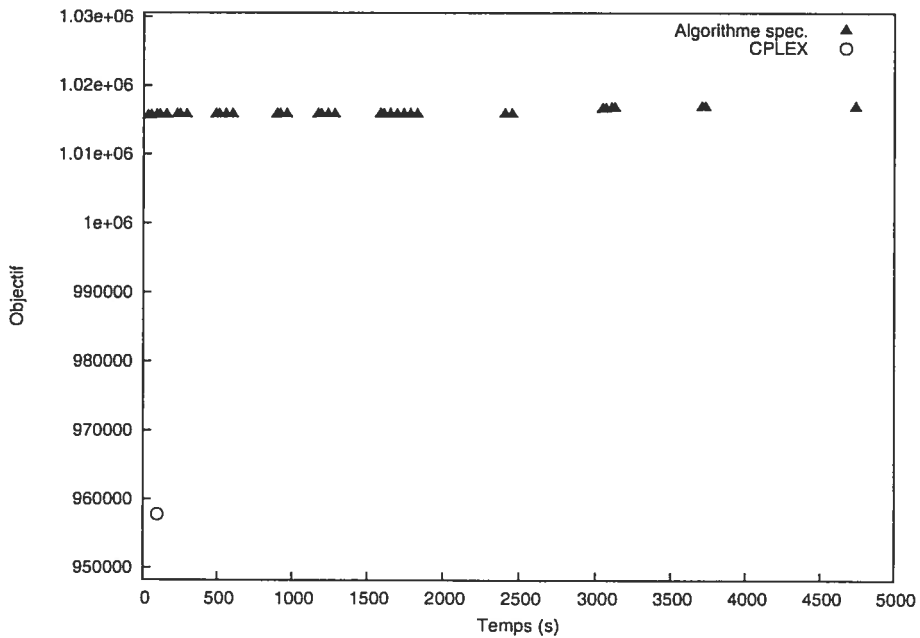


FIG. 6.20 – Comparaisons entre CPLEX et l'algorithme spécialisé sur l'instance 105 (avec produits).

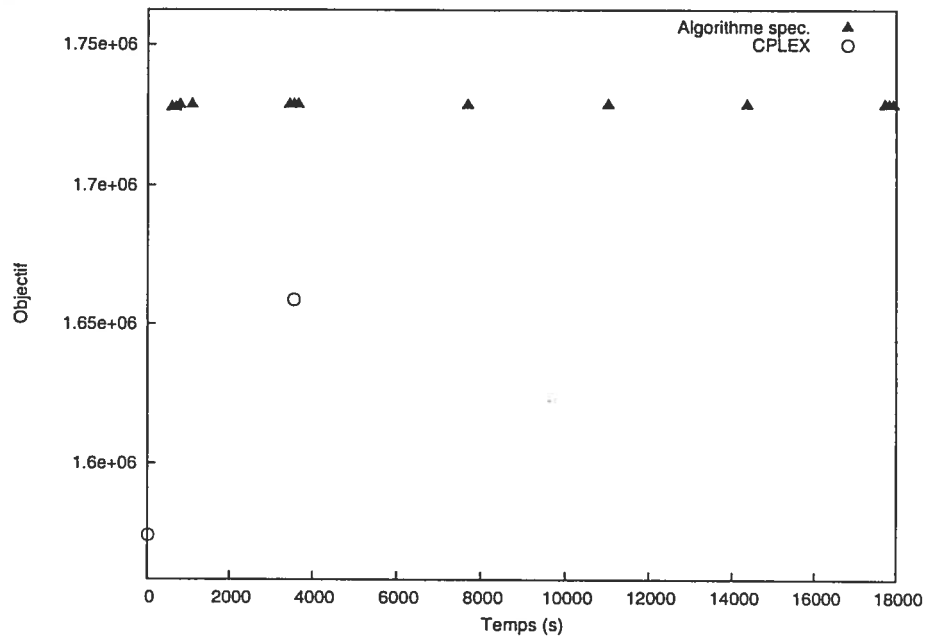


FIG. 6.21 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 106 (avec produits).

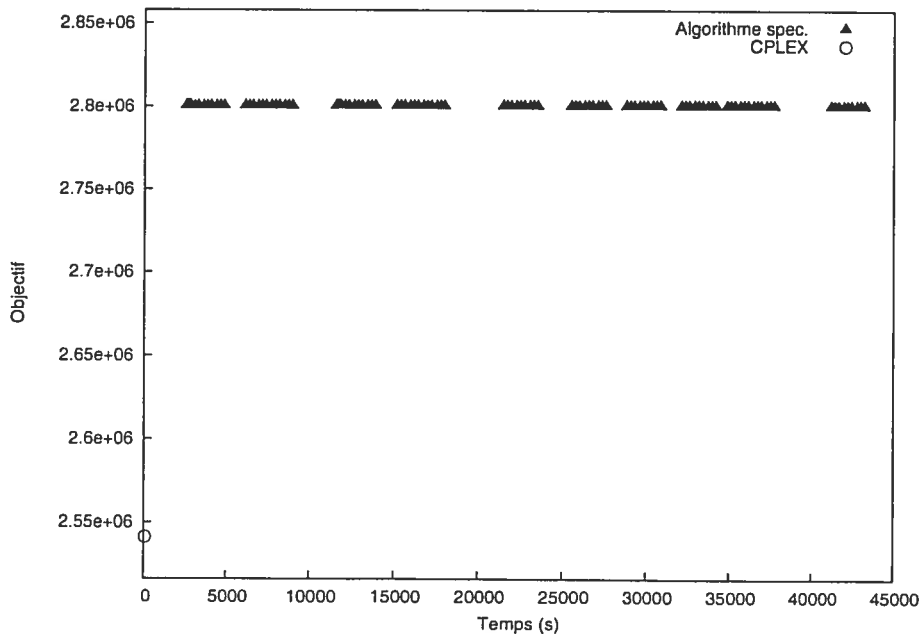


FIG. 6.22 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 107 (avec produits).

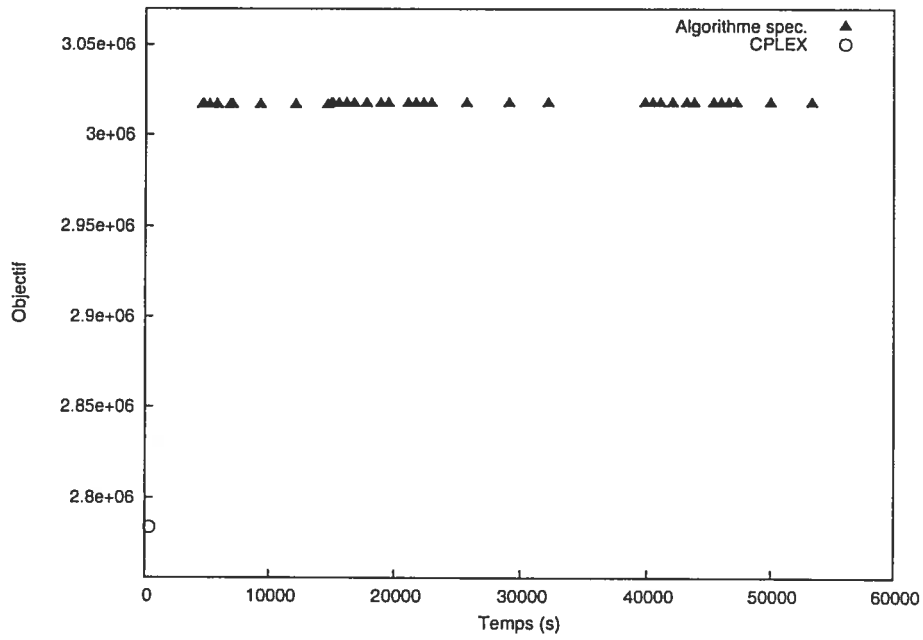


FIG. 6.23 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 108 (avec produits).

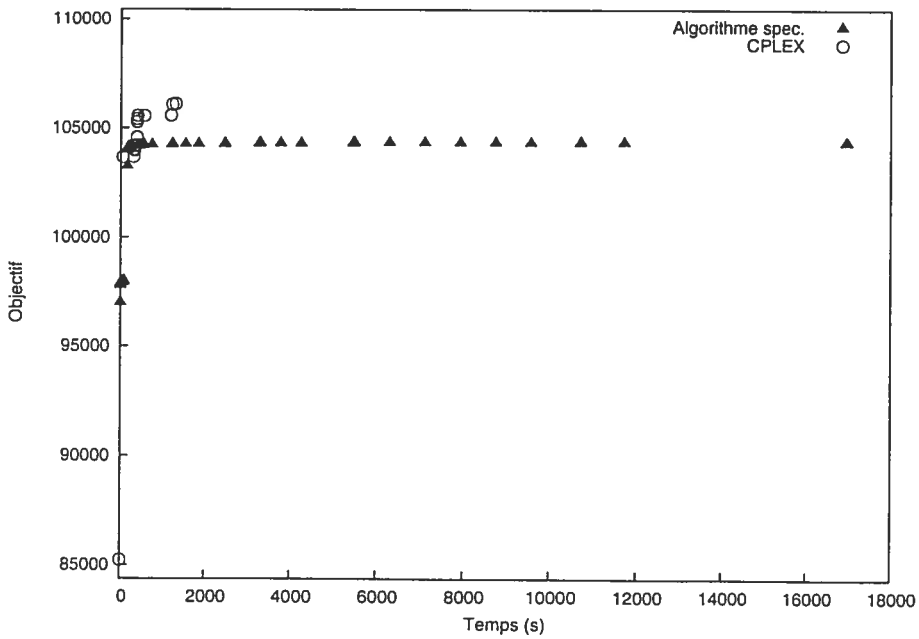


FIG. 6.24 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 110 (avec produits).

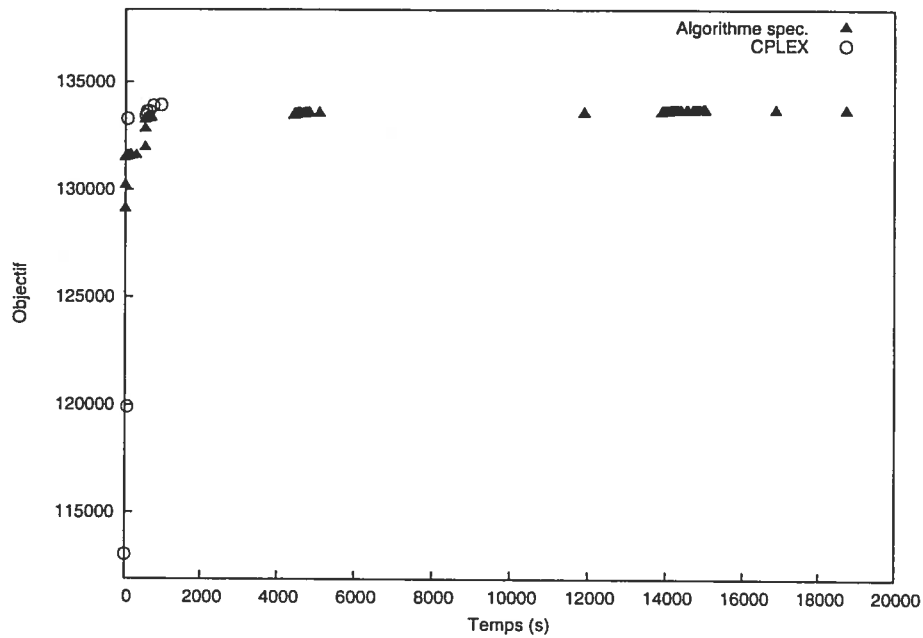


FIG. 6.25 – Comparaisons entre CPLEX et l’algorithme spécialisé sur l’instance 111 (avec produits).

Dans le cas des instances 103, 110, et 111, les solutions obtenues par CPLEX sont légèrement supérieures à celles obtenues par l’algorithme spécialisé. On peut supposer que pour ces problèmes, les heuristiques que CPLEX utilise à la racine sont particulièrement efficaces et lui permettent d’identifier rapidement une bonne solution entière. Par contre, pour les autres instances, y compris celles de grande taille, l’algorithme spécialisé obtient des solutions nettement supérieures à celles de CPLEX. On peut remarquer que le rapport entre ces solutions est plus élevé que dans le cas à un critère. Le fait de considérer plusieurs critères augmente de façon significative la taille de l’espace de solution, ce qui avantage d’autant plus l’algorithme spécialisé car celui-ci n’explore que le sous-espace des solutions possédant la structure décrite ci-dessus.

Chapitre 7

Cadre dynamique

Les formulations que nous avons présentées jusqu'à maintenant sont statiques : étant donné l'état actuel du réseau, la capacité résiduelle, les derniers estimés de demande, les prix de la compétition et un jeu de paramètres appropriés, nous cherchons à obtenir les prix (et, de façon secondaire, les estimés de flot) qui permettent de maximiser le revenu à très court terme. La durée de vie de ces solutions est courte ; une réoptimisation s'impose dès que les données du problème sont modifiées. Sur certains marchés très volatils et compétitifs, de telles modifications peuvent survenir plusieurs fois par jour. Étant donné le caractère ponctuel et éphémère des solutions, on peut se permettre dans le modèle d'ignorer les aspects dynamiques et stochastiques du problème qui n'ont qu'un impact minime à très court terme. Cela ne signifie pas que le modèle statique ne puisse être utilisé dans un contexte dynamique, où des réoptimisations régulières (e.g. aux quatre heures comme sur le marché trans-frontalier nord-américain) s'imposent. Si tel est le cas, il est légitime de se demander si la capacité dont on dispose à un moment donné peut être offerte (au prix que le modèle identifie comme «optimal») ou s'il ne serait pas plus opportun d'attendre une période subséquente, dans l'espoir que cet inventaire toujours disponible rapporte plus dans le futur. Une telle décision doit être prise en tenant compte du coût d'opportunité qu'entraîne la vente immédiate d'une unité de capacité. C'est là la fonction principale des systèmes de gestion de l'inventaire (*Yield Management*) utilisés depuis une vingtaine d'années dans l'industrie aérienne. Ces systèmes n'intègrent toutefois pas l'aspect tarification au processus de décision.

Il est aussi permis d'imaginer un contexte d'utilisation légèrement différent pour notre approche. Au lieu de se concentrer sur un horizon tactique de quelques heures nécessitant l'utilisation de

données en temps réel, on peut considérer un horizon plus large allant de quelques semaines à quelques mois. Une telle application revêt un caractère stratégique plutôt qu'opérationnel et permet de tester la validité de politiques tarifaires existantes ou d'en concevoir de nouvelles, par exemple lors de l'ouverture de nouveaux itinéraires. Dans cet environnement, on utilise des données futures plutôt que courantes et les solutions ont une durée de vie beaucoup plus longue que dans le contexte tactique. Comme nous disposons de plus de temps, la rapidité du processus de résolution n'est pas un facteur primordial et le côté algorithmique est moins important. Toutefois, comme nous cherchons à établir des politiques tarifaires à moyen ou long terme, d'autres aspects entrent en ligne de compte.

7.1 Tarification et allocation de la capacité conjoints

Dans un contexte tactique à très court terme, l'allocation de la capacité a déjà été effectuée de sorte que le modèle ne résout que le problème de tarification en tenant compte de limites de réservations exogènes.

Par contre, lorsque nous considérons un horizon plus éloigné, nous ne pouvons plus ignorer le caractère dynamique du processus de réservation et de fermeture progressive des classes tarifaires. En incorporant cet aspect au modèle, nous sommes véritablement en mesure de résoudre conjointement les deux sous-problèmes.

Selon Belobaba [Bel98], le but des méthodes d'allocation de la capacité est *«[to] make seats that are expected to go unsold available at a lower fare to passengers who would otherwise not travel, while at the same time ensuring that these lower fares are not purchased by passengers who are willing to pay a higher fare»*. Notre revue bibliographique a présenté plusieurs approches d'allocation de la capacité dynamiques¹ qui diffèrent par leurs hypothèses (processus d'arrivée

¹Rappelons qu'il existe deux grandes catégories de méthodes d'allocation de la capacité. Les premières, dites «statiques», supposent que les requêtes se font séquentiellement par ordre croissant de prix (i.e. que tous les sièges dans la classe la plus basse sont réservés avant que n'arrivent les requêtes de réservation pour la prochaine classe, et ainsi de suite jusqu'à la classe la plus haute) tandis que les secondes, dites «dynamiques», ne posent aucune hypothèse quant à l'ordre d'arrivée des requêtes. Bien que des méthodes appartenant à la première catégorie soient encore largement utilisées en pratique (à cause principalement de leur simplicité d'implantation), les travaux de recherche en allocation de la capacité depuis une quinzaine d'années se concentrent presque exclusivement sur les méthodes dynamiques car elles sont plus réalistes (voir chapitre 3). C'est sur ce type de méthode que nous baserons dans ce chapitre.

et distribution statistique des requêtes, niveau d'interaction entre les segments, contribution de chaque segment aux revenus sur l'ensemble du réseau, etc.) mais qui peuvent être décrites de façon générale par un programme dynamique.

Nous présentons maintenant un cadre général d'optimisation des revenus dans un contexte dynamique. La présentation est inspirée des méthodes d'allocation de la capacité les plus communes mais permet d'intégrer l'aspect tarification propre à notre modèle.

7.2 Formulation et résolution

Considérons un horizon temporel divisé en N périodes de réservations indexées de façon décroissante par $n = N, N - 1, \dots, 1$. La période $n = 0$ correspond à la date de départ et indique que le processus est terminé. Les méthodes d'allocation de la capacité dynamiques supposent qu'au plus une requête de réservation est effectuée à chaque période et que conséquemment N est le nombre maximum de requêtes pouvant être reçues sur le réseau considéré. Soit u_n le vecteur de capacité résiduelle sur les segments du meneur au début de la période n . Nous avons ainsi $u_N = u$, où u dénote comme aux chapitres précédents le vecteur de capacité totale des segments du meneur.

Nous commençons par présenter de façon générale les méthodes d'allocation de la capacité dynamiques. Soit $J_n(u_n)$ le revenu optimal aux périodes $n, n - 1, \dots, 0$ avec une capacité résiduelle de u_n au début de la période n . Soit ρ_n un vecteur de dimension égale au nombre de vols du meneur et dont toutes les composantes sont nulles sauf celle correspondant à l'itinéraire associé à la requête n . Cette composante est égale au nombre de sièges concernés par la requête. La quantité $J_n(u_n)$ satisfait à l'équation de Bellman

$$J_n(u_n) = \max_{v_n \in \mathbb{B}} \left\{ tD\rho_n v_n + J_{n-1}(u_n - C\rho_n v_n) \right\} \quad (7.1)$$

$$J_0(u) = 0, \quad \forall u \in \mathbb{R}_{0+}, \quad (7.2)$$

où le vecteur t (prix) et les matrices C (incidence segment-vol) et D (incidence vol-prix) ont les mêmes significations qu'aux chapitres précédents. La quantité que ce processus cherche à maximiser est $J_N(u)$. Le processus se termine dès que $u_n = 0$ ou lorsqu'on atteint la période 0. Un tel processus possède une interprétation intuitive. La requête n est acceptée (i.e $v_n = 1$) si, et seulement si, son revenu immédiat $tD\rho_n$ excède le coût d'opportunité d'allouer immédiatement un siège sur les segments qu'implique cette requête. Dans ce cas-ci, les coûts t sont fixes et

exogènes. Le développement de premier ordre de la fonction J_n nous offre une interprétation intuitive des équations de Bellman énoncées ci-dessus. En effet,

$$\nabla J_n(u_n)C\rho_n \approx J_n(u_n) - J_n(u_n - C\rho_n).$$

Autrement dit, la contribution marginale de la $n^{\text{ième}}$ requête au niveau du réseau peut être approximée par le coût d'opportunité qu'entraîne l'acceptation de cette requête.

Nous allons maintenant tenter d'intégrer notre modèle de tarification dans un cadre dynamique similaire, dans le but de résoudre simultanément le problème d'allocation de la capacité. Nous adoptons cependant une définition différente pour les périodes : chaque période correspond à une classe tarifaire et les $N - 1$ «contrôles» v_n (pour employer la terminologie habituelle en programmation dynamique) que nous devons effectuer sur cet horizon reviennent à déterminer le moment auquel chaque classe est fermée. Comme les classes sont emboîtées, on suppose qu'à la période n les classes $n, n - 1, \dots, 1$ sont offertes.

7.2.1 Tarifs communs aux périodes

En pratique, les tarifs proposés par les compagnies aériennes sont relativement stables d'une période de contrôle à l'autre, du moins du point de vue de l'allocation de la capacité. En effet, bien que le département de tarification ait la possibilité de modifier les tarifs offerts afin qu'ils reflètent les tendances du marché, le processus de gestion des classes de réservation se base en général sur des moyennes de tarifs historiques. Ces tarifs demeurent constants à mesure que les classes sont fermées¹. Par exemple, un produit tarifaire en classe Y (habituellement la plus chère et la moins restrictive de la cabine économique) sera offert à 500\$ peu importe que les classes inférieures soient ouvertes ou non. Du point de vue de l'optimisation des tarifs, une telle façon de faire est une restriction car elle nous empêche de proposer un tarif reflétant le mieux possible la situation du marché à chaque période de contrôle. Cependant, comme il s'agit de l'approche habituellement suivie dans l'industrie (et que, pour des raisons organisationnelles et humaines, de telles pratiques sont difficiles à changer lorsqu'elles sont profondément ancrées dans une entreprise), nous devons être en mesure de la traiter. C'est pourquoi nous allons dans un premier temps supposer que $t_n = t$ pour $n = N, N - 1, \dots, 1$, i.e. qu'un seul vecteur de tarifs t est offert pour l'ensemble des périodes.

¹Certaines méthodes de contrôle d'inventaire statiques ne fonctionnent que sous cette hypothèse de constance des prix.

Soit $R_n(t, v_n)$ le revenu immédiat à la période n pour un vecteur de tarifs t et une allocation de capacité v_n donnés. Ainsi, les équations de Bellman du problème d'allocation dynamique de la capacité sont

$$J_n(t, u_n) = \max_{0 \leq v_n \leq u_n} \left\{ R_n(t, v_n) + J_{n-1}(t, u_n - v_n) \right\}, \quad n = N, N-1, \dots, 1, \quad (7.3)$$

$$J_0(t, u) = 0, \quad \forall t, \forall u \in \mathbb{R}_{0+}. \quad (7.4)$$

Remarquons que la fonction de transition $\varphi_n : \mathbb{R}^{|F^1| \cdot |P^1|} \times \mathbb{R}_{0+}^{|S^1|} \mapsto \mathbb{R}^{|F^1| \cdot |P^1|} \times \mathbb{R}_{0+}^{|S^1|}$ de ce programme est

$$\varphi_n(t, u_n) = (t, u_n - v_n), \quad n = N, N-1, \dots, 1.$$

Dans notre approche, la quantité $R_n(t, v_n)$ est la valeur de la solution optimale d'un problème bi-niveau :

$$R_n(t, v) = \max_{(x,y) \in \Omega_n(t)} \left\{ tDx \mid Cx \leq v \right\},$$

où $\Omega_n(t)$ est l'ensemble des solutions optimales du problème de second niveau de notre modèle induit par le vecteur de tarifs t à la période n , i.e.

$$\Omega_n(t) = \arg \min_{x,y} \left\{ (c_n + t)x + d_n y \mid Ax + By = b_n; x, y \geq 0 \right\}.$$

Selon ce cadre dynamique, le problème global d'optimisation des tarifs et d'allocation de la capacité prend la forme

$$\max_t J_N(t, u).$$

Nous dénoterons par J_N^* la valeur de la solution optimale de ce problème.

Le prochain lemme et son corollaire démontrent que ce cadre dynamique peut, dans le cas qui nous intéresse, se résoudre comme un problème mathématique à deux niveaux.

LEMME 7.1

$$J_n(t, u_n) = \max_{v_1, v_2, \dots, v_n \geq 0} \left\{ \sum_{m=1}^n R_m(t, v_m) \mid \sum_{m=1}^n v_m = u_n \right\}, \quad n = N, N-1, \dots, 1.$$

Preuve La preuve se fait par induction.

Base ($n = 1$) :

$$\begin{aligned} J_1(t, u_1) &= \max_{0 \leq v_1 \leq u_1} \{R_n(t, v_1) + J_0(t, u_1 - v_1)\} \quad (\text{par 7.3}) \\ &= \max_{0 \leq v_1 \leq u_1} \{R_n(t, v_1) + 0\} \quad (\text{par 7.4}) \\ &= R_1(t, u_1) \\ &= \max_{v_1 \geq 0} \{R_1(t, v_1) \mid v_1 = u_1\}. \end{aligned}$$

Hypothèse :

$$J_{t, n-1}(t, u_{n-1}) = \max_{v_1, v_2, \dots, v_{n-1} \geq 0} \left\{ \sum_{m=1}^{n-1} R_m(t, v_m) \mid \sum_{m=1}^{n-1} v_m = u_{n-1} \right\}.$$

Induction :

$$\begin{aligned} J_n(t, u_n) &= \max_{0 \leq v_n \leq u_n} \{R_n(t, v_n) + J_{n-1}(t, u_n - v_n)\} \\ &= \max_{0 \leq v_n \leq u_n} \left\{ R_n(t, v_n) + \max_{v_1, v_2, \dots, v_{n-1} \geq 0} \left\{ \sum_{m=1}^{n-1} R_m(t, v_m) \mid \sum_{m=1}^{n-1} v_m = u_n - v_n \right\} \right\} \\ &\quad (\text{par l'hypothèse d'induction}) \\ &= \max_{0 \leq v_n \leq u_n} \left\{ R_n(t, v_n) + \max_{v_1, v_2, \dots, v_{n-1} \geq 0} \left\{ \sum_{m=1}^{n-1} R_m(t, v_m) \mid \sum_{m=1}^{n-1} v_m = u_n - v_n \right\} \right\} \\ &\quad (\text{car } u_{n-1} = u_n - v_n \text{ par la fonction de transition}) \\ &= \max_{v_1, v_2, \dots, v_n \geq 0} \left\{ \sum_{m=1}^n R_m(t, v_m) \mid \sum_{m=1}^n v_m = u_n \right\}. \end{aligned}$$

□

COROLLAIRE 7.2

J_N^* est la valeur de la solution optimale du problème bi-niveau

$$\max_{v_n \mid n=N, N-1, \dots, 1} t \sum_{n=1}^N Dx_n \quad (7.5)$$

$$\text{s.l.c.} \quad \sum_{n=1}^N v_n = u \quad (7.6)$$

$$n = N, N-1, \dots, 1 : \quad (7.7)$$

$$v_n \geq 0 \quad (7.8)$$

$$Cx_n \leq v_n \quad (\xi_n) \quad (7.9)$$

$$(x_n, y_n) \in \Omega_n(t). \quad (7.10)$$

□

Cette formulation «dynamique» du modèle bi-niveau permet de résoudre de façon conjointe les sous-problèmes de tarification et d'allocation de la capacité des segments aux classes tarifaires. À prime abord, on pourrait croire que le problème est séparable par période de réservation (selon l'indice n). Deux éléments rendent toutefois une telle séparation difficile. Les contraintes de capacité (7.6) sont couplantes pour les périodes et comme nous avons choisi de ne considérer qu'un seul vecteur de tarifs pour l'ensemble des périodes, les «blocs» correspondant aux périodes sont aussi couplés par ces variables. Ces «blocs» (définis implicitement par les contraintes (7.9) et l'ensemble (7.10)) sont en fait des ensembles non convexes (et possiblement non connexes). Les techniques traditionnelles de décomposition ne s'appliquent donc pas dans ce cas.

Soient $\bar{v}_n = \sum_{m=1}^n v_m$ et $\hat{v}_n = \sum_{m=n}^N v_m$. Comme nous supposons que les classes tarifaires sont emboîtées, le niveau de protection des classes $1, 2, \dots, n$ correspond alors à \bar{v} , i.e. ces classes ont au moins \bar{v} sièges qui leur sont exclusivement réservés. De façon symétrique, \hat{v}_n représente la limite de réservation de la classe n , i.e. au plus \hat{v}_n peuvent être réservés dans cette classe. Remarquons que $\hat{v}_1 = u$, ce qui signifie que la classe la plus élevée a accès à tous les sièges de l'appareil.

Les schémas d'allocation de la capacité décrits ci-dessus sont d'abord basés sur le segment de vol, comme c'est le cas pour la plupart des méthodes encore utilisées dans l'industrie aérienne. Cependant, étant donnée une allocation de capacité v_n à la période n , le modèle bi-niveau, que nous utilisons pour calculer $R_n(t, v_n)$, procède lui-même à une répartition des passagers au niveau du réseau complet. La méthode résultante est donc hybride car elle produit d'abord des allocations de capacité au niveau du segment de vol (comme c'est le cas dans le cadre traditionnel) puis raffine ces allocations au niveau du réseau en tenant compte des interactions entre les itinéraires partageant des segments de vol communs. Nous sommes donc en mesure d'établir des niveaux de protection précis pour chaque classe tarifaire (ou même chaque produit tarifaire) sur chaque vol. Comme nous le verrons dans la description des méthodes de résolution, le processus d'allocation de la capacité ne s'effectue pas de façon séquentielle (i.e. d'abord sur les segments puis sur les itinéraires partageant ces segments) mais plutôt de façon interactive entre les deux niveaux de granularité.

7.2.2 Tarifs propres à chaque période

Bien qu'il ne s'agisse pas de la façon de faire traditionnelle, certains transporteurs ont commencé ces dernières années à revoir leur pratiques tarifaires et à relaxer l'hypothèse de constance des tarifs lors de l'allocation de la capacité aux classes tarifaires. Dans la plus part des cas, les prix ne sont pas déterminés de façon optimale mais suivent des échelles pré-établies, par exemple une augmentation fixe de 20\$ à chaque nouvelle période.

Nous intéressons dans cette section au cas où la contrainte de constance des tarifs sur les périodes de réservation est relaxée et y intégrons l'aspect tarification dynamique. Ainsi, au lieu de déterminer un seul vecteur de tarifs t , nous cherchons les vecteurs t_N, t_{N-1}, \dots, t_1 propres à chaque période.

Une réflexion sur la nature des périodes de réservation s'impose si on adopte une telle approche. Nous avons précédemment supposé que les périodes $N, N-1, \dots, 1$ déterminaient les points de fermeture des N classes tarifaires. Comme les classes sont emboîtées, à la période n les classes $n, n-1, \dots, 1$ sont ouvertes. Cependant, dans le cas où les tarifs sont propres à chaque période, la nécessité de fermer les classes à mesure que progresse le processus de réservation n'est plus aussi claire. En supposant que l'on soit en mesure d'identifier avec un degré de précision acceptable les groupes d'usagers présents à chaque période (ce qui présume une analyse statistique sérieuse), il peut s'avérer avantageux d'offrir en tout temps l'ensemble des produits tarifaires disponibles (et donc de laisser ouvertes toutes les classes) et d'en ajuster le prix pour satisfaire aux besoins des usagers présents à chaque période. Considérons par exemple un vol se composant d'un seul segment de 100 places et offrant plusieurs produits. Un d'entre eux est associé à la classe L (la classe la plus basse et la plus restrictive) et un autre est associé à la classe H (une classe moyenne). Un billet en classe H est remboursable à 75% tandis qu'un billet en classe L ne l'est pas. Dans le cadre traditionnel, la classe L est offerte à 130\$ et la classe H est offerte à 285\$. Ces tarifs demeurent constants tout au long de la période de réservation. On impose une limite de réservation de 5 sièges en classe L. Les trois premiers passagers à effectuer une réservation pour ce vol sont des touristes qui planifient leur voyage longtemps à l'avance et qui désirent passer plusieurs semaines à destination. Ils sont donc indifférents aux restrictions temporelles (e.g. samedi soir à destination obligatoire) associées à leur billet. Comme ils sont les premiers à réserver, toutes les classes leurs sont accessibles. Ils recherchent le billet à meilleur prix et réservent en classe L. Sortons maintenant du cadre traditionnel et supposons que nous ayons la possibilité de modifier les tarifs à chaque période de réservation. Nos trois touristes

désirent toujours acheter un billet au meilleur prix possible. Ils considèrent donc la classe L. Cependant, comme nous disposons de plus de flexibilité, nous calculons en nous basant sur des paramètres comportementaux que nous avons obtenus à partir de données historiques que les usagers de ce groupe sont prêts à payer jusqu'à 15\$ pour bénéficier de la «remboursabilité». Ainsi, parallèlement à la classe L à 130\$, nous offrons aussi la classe H à 145\$. Si un des trois passager décide d'opter pour la classe H, les revenus passent de 390\$ à 405\$, une augmentation d'environ 3.8%. Ainsi, en étant confiant de pouvoir identifier les groupes d'usagers présents à chaque période, nous pouvons en ajustant les tarifs selon la volonté de payer de chaque passager augmenter de façon significative les revenus. C'est une illustration de ce que les économistes appellent la discrimination par le prix. En appliquant ce principe, les périodes ne correspondent plus aux points de fermeture des classes tarifaires mais plutôt à des intervalles pendant lesquels les tarifs sont ajustés afin de mieux circonscrire la propension à payer des passagers.

Comme les tarifs sont désormais propres à chaque période, nous devons modifier le cadre présenté à la section précédente. Soit $R_n(v)$ le revenu immédiat à la période n avec une allocation de capacité de v . Dans notre approche, $R_n(v)$ est la valeur de la solution optimale du problème bi-niveau

$$R_n(v) = \max_{(x,y) \in \Omega_n(t_n)} \left\{ t_n Dx \mid Cx \leq v \right\}.$$

Les équations de Bellman sont

$$J_n(u_n) = \max_{0 \leq v_n \leq u_n} \left\{ R_n(v_n) + J_{n-1}(u_n - v_n) \right\}, \quad n = N, N-1, \dots, 1. \quad (7.11)$$

$$J_0(u) = 0, \quad \forall u \in \mathbb{R}_{0+} \quad (7.12)$$

et la fonction de transition $\varphi_n : \mathbb{R}_{0+}^{|S^1|} \mapsto \mathbb{R}_{0+}^{|S^1|}$, de ce programme est

$$\varphi_n(u_n) = u_n - v_n, \quad n = N, N-1, \dots, 1.$$

Selon ce modèle dynamique, la valeur de la solution optimale des sous-problèmes de tarification d'allocation de la capacité est

$$J_N^* = J_N(u).$$

Nous reprenons le lemme présenté à la section précédent en l'adaptant toutefois au modèle dynamique où les tarifs sont propres à chaque période. Comme précédemment, le corollaire de ce lemme suggère une reformulation du modèle dynamique en un programme bi-niveau.

LEMME 7.3

$$J_n(u_n) = \max_{v_1, v_2, \dots, v_n \geq 0} \left\{ \sum_{m=1}^n R_m(v_m) \mid \sum_{m=1}^n v_m = u_n \right\}, \quad n = N, N-1, \dots, 1.$$

Preuve La preuve se fait par induction en utilisant le même argument qu'au lemme 7.1. Le paramètre t disparaît toutefois car les tarifs sont maintenant des variables de décision propres à chaque période. □

COROLLAIRE 7.4

J_N^* est la valeur de la solution optimale du problème bi-niveau

$$\max_{t_n, v_n | n=N, N-1, \dots, 1} \sum_{n=1}^N t_n D x_n \tag{7.13}$$

$$\text{s.l.c.} \quad \sum_{n=1}^N v_n = u \tag{7.14}$$

$$n = N, N-1, \dots, 1 : \tag{7.15}$$

$$v_n \geq 0 \tag{7.16}$$

$$C x_n \leq v_n \quad (\xi_n) \tag{7.17}$$

$$(x_n, y_n) \in \Omega_n(t_n). \tag{7.18}$$

□

Propriétés de la fonction de revenu

Soit la fonction

$$Q(v_1, v_2, \dots, v_N) = \sum_{n=1}^N R_n(v_n).$$

Nous avons vu au lemme 7.3 que la solution optimale du problème de tarification et d'allocation dynamique de la capacité est obtenue en maximisant Q sous la contrainte que la somme des capacités allouées à chaque période égale la capacité totale des segments. Dans notre approche $R_n(v_n)$ est la solution du problème bi-niveau

$$R_n(v_n) = \max_{(x, y) \in \Omega_n(t)} \{ t D x \mid C x \leq v_n \}.$$

Nous illustrons maintenant certaines propriétés de la fonction Q que nous exploiterons pour résoudre le problème global de tarification et d'allocation de la capacité.

LEMME 7.5

Soit $\lambda \in]0, 1[$ et $\mu \in \{\lambda, 1 - \lambda\}$. Alors,

$$R_n(\mu v) \geq \mu R_n(v),$$

pour $n = N, N - 1, \dots, 1$ et $v \geq 0$.

Preuve Soit $(\hat{t}, \hat{x}, \hat{y})$ une solution optimale du problème

$$P1 : \max_{(x,y) \in \Omega_n(\hat{t})} \{tDx \mid Cx \leq v\}.$$

Considérons maintenant le problème

$$P2 : \max_{(x,y) \in \Omega_n(\hat{t})} \{tDx \mid Cx \leq \mu v\}.$$

Soit $\bar{x} = \mu \hat{x}$ et \bar{y} une solution du système $\{B\bar{y} = b - A\bar{x}, (d - \hat{\eta}B)\bar{y} = 0\}$, où $\hat{\eta}$ est la variable duale de second niveau associée à la solution $(\hat{t}, \hat{x}, \hat{y})$. Une telle solution \bar{y} existe car, par hypothèse, les vols de la compétition ont une capacité illimitée et comme nous l'avons démontré au lemme 6.1, il existe sur chaque marché au moins un groupe d'utilisateurs qui, dans un schéma tarifaire donné, encoure son coût perçu maximal, lequel dépend de l'offre de la compétition. Nous démontrons d'abord que $(\hat{t}, \bar{x}, \bar{y})$ est une solution admissible de P2. En effet, $(\bar{x}, \bar{y}) \in \Omega_n(\hat{t})$ (par les conditions d'optimalité primales-duales du problème de second niveau que décrit implicitement $\Omega_n(\hat{t})$). De plus, $C\bar{x} = C\mu\hat{x} \leq \mu v$ car $C\hat{x} \leq v$. Comme $(\hat{t}, \bar{x}, \bar{y})$ est admissible pour P2, nous pouvons conclure que la valeur de la solution optimale de P2 est bornée inférieurement par $\hat{t}D\bar{x} = \hat{t}D\mu\hat{x} = \mu(\hat{t}D\hat{x})$, cette dernière expression étant par supposition la valeur de la solution optimale de P1 multipliée par le scalaire μ . Conséquemment, $R_n(\mu v) \geq \mu R_n(v)$. \square

LEMME 7.6

La fonction $R_n(v)$ est concave en v , pour $n = N, N - 1, \dots, 1$.

Preuve Soit $\lambda \in]0, 1[$. Nous avons

$$\begin{aligned} R_n(\lambda v + (1 - \lambda)w) &= \max_{(x,y) \in \Omega_n(\hat{t})} \{tDx \mid Cx \leq \lambda v + (1 - \lambda)w\} \\ &\geq \max_{(x,y) \in \Omega_n(\hat{t})} \{tDx \mid Cx \leq \lambda v\} + \max_{(x,y) \in \Omega_n(\hat{t})} \{tDx \mid Cx \leq (1 - \lambda)w\} \\ &\quad (\text{par la nature des contraintes de capacité}). \\ &= R_n(\lambda v) + R_n((1 - \lambda)w) \end{aligned}$$

$$\geq \lambda R_n(v) + (1 - \lambda)R_n(w) \quad (\text{par le lemme 7.5}).$$

□

COROLLAIRE 7.7

La fonction $Q(v_1, v_2, \dots, v_N)$ est concave.

Preuve Puisque

$$Q(v_1, v_2, \dots, v_N) = \sum_{n=1}^N R_n(v_n),$$

i.e. une somme de fonctions concaves, Q est elle-même concave. □

Résolution par gradient projeté

Dans le cadre dynamique que nous avons développé, le problème global de tarification et d'allocation de la capacité est

$$\text{(GLOB)} \quad (7.19)$$

$$\max_{v_1, v_2, \dots, v_N \geq 0} Q(v_1, v_2, \dots, v_N) \quad (7.20)$$

$$\text{s.l.c.} \quad \sum_{n=1}^N v_n = u. \quad (7.21)$$

Comme nous l'avons vu, ce problème consiste à maximiser une fonction concave sous un ensemble de contraintes linéaires de type simplicial. La fonction Q est définie implicitement et nous devons résoudre un problème à deux niveaux pour l'évaluer en un point donné, ce qui peut s'avérer coûteux, particulièrement sur des instances de grande taille. Cependant, si on retire la contrainte (7.21), le problème devient séparable par période de réservation, i.e. selon l'indice n .

Nous choisissons de résoudre le problème GLOB par la méthode du gradient projeté. La question que nous devons nous poser est dans ce cas : quel gradient projeter ? Rappelons que

$$Q(v_1, v_2, \dots, v_N) = \sum_{n=1}^N R_n(v_n)$$

où

$$R_n(v_n) = \max_{(x,y) \in \Omega_n(t)} \{tDx \mid Cx \leq v_n\}.$$

Ainsi, étant donné un vecteur d'allocation de capacité (v_1, v_2, \dots, v_n) respectant la contrainte (7.21), nous devons pour évaluer Q en ce point résoudre une séquence de N problèmes à deux

niveaux. La fonction $R_n(v_n)$ n'est donc pas différentiable et nous ne disposons pas d'expression exacte pour ∇Q .

Soit $\xi_n(v_n)$ le vecteur de multiplicateurs duaux associé à la contrainte $Cx \leq v_n$ (7.17) dans la solution optimale de $R_n(v_n)$. Ces multiplicateurs peuvent s'interpréter comme la contribution marginale d'une unité supplémentaire de capacité au revenu R_n à la période n , étant donnée une allocation de v_n . Il s'agit d'une interprétation très semblable à celle qu'aurait le gradient $\nabla R_n(v_n)$, si un tel vecteur pouvait être calculé. Se basant sur cet argument, nous posons

$$\nabla Q \approx \Xi,$$

où $\Xi = (\xi_1(v_1), \xi_2(v_2), \dots, \xi_N(v_N))$ est le vecteur obtenu par la concaténation des vecteurs de multiplicateurs duaux aux contraintes de capacité à chaque période. Il s'agit bien sur d'une approximation car le mieux que nous puissions espérer est que Ξ soit un sous-gradient de Q , i.e. $\Xi \in \partial Q$. La méthode que nous présentons est donc de nature heuristique.

En supposant que $\mathcal{U} = \left\{ (v_1, v_2, \dots, v_n) \in \mathbb{R}_{0+}^{|S^1| \times N} \mid \sum_{n=1}^N v_n = u \right\}$, notre méthode du gradient projeté est décrite comme :

- **Initialisation**

Identifier une allocation de capacité initiale $V^0 = (v_1^0, v_2^0, \dots, v_N^0) \in \mathcal{U}$. Soit ω un scalaire positif. $k \leftarrow 0$.

- **Boucle principale**

1. Calculer Ξ^k en résolvant $R_1(v_1^k), R_2(v_2^k), \dots, R_N(v_N^k)$.
2. Soit $V^{k+1} \in \text{proj}_{\mathcal{U}}(V^k + \omega \Xi^k) = \arg \min_{U \in \mathcal{U}} \frac{1}{2} \|U - (V^k + \omega \Xi^k)\|^2$
3. Si V^{k+1} satisfait le critère d'arrêt, alors terminer.
4. Sinon, $k \leftarrow k + 1$ et retourner en 1.

Nous avons illustré à la section 6.5.3 une procédure permettant de résoudre le problème de projection (en norme euclidienne) sur un ensemble simplicial. C'est sur cette même procédure qu'est basée notre implantation de la méthode décrite ci-dessus.

La méthode possède une interprétation économique intéressante. En pratique, on peut imaginer que l'allocation de capacité initiale V^0 soit une moyenne historique sur les segments concernés. En résolvant les problèmes R_n pour chaque période $n = N, N-1, \dots, 1$, nous obtenons le vecteur Ξ qui approxime localement le taux marginal de croissance des revenus en fonction de la capacité

allouée et de la situation sur le marché. Nous entreprenons donc de modifier cette allocation de capacité historique dans la direction qui nous semble la plus profitable étant donné l'état actuel du marché, en respectant toutefois la capacité maximale des segments (d'ou l'opération de projection).

Nous illustrons les performances la méthode du gradient projeté en comparant ses résultats avec ceux obtenus par le solveur CPLEX sur une formulation optimale. Nous avons considéré quatre instances, soit 100, 103, 110 et 111 (voir chapitre 6) et avons, pour chaque instance, considéré deux cas, avec respectivement 4 et 10 périodes de réservation. À chaque itération de la méthode, pour chaque période considérée, nous terminons la résolution à la première solution admissible identifiée, ce qui renforce le caractère heuristique de cette approche. Les résultats sont présentés au tableau 7.1.

Instance	N	CPLEX		Heuristique		
		Valeur	Temps (s)	Valeur	Temps (s)	Rapport
100	4	426 073.44	2 333.91	413 967.93	24.89	97.2%
	10	431 296.01	10 314.53	412 389.58	39.49	95.6%
103	4	617 498.22	7 738.00	596 773.35	114.74	96.6%
	10	628 175.95	5 840.77	605 979.19	495.16	95.5%
110	4	106 029.01	445.85	96 960.47	72.11	90.4%
	10	107 868.09	6 696.26	99 361.12	494.68	92.1%
111	4	134 563.77	2 292.07	129 371.53	22.71	96.1%
	10	135 081.82	833.74	130 820.78	64.00	96.8%

TAB. 7.1 – Performances comparées de l'heuristique de gradient projeté.

Nous n'avons pas utilisé de critère d'arrêt particulier dans l'obtention des résultats du tableau 7.1 ; nous avons plutôt interrompu l'algorithme au moment où la croissance de la fonction objectif nous semblait ralentir. En fait, nous avons pu observer que plus la méthode disposait de temps, plus nous nous approchions de la solution optimale. Ainsi, dans un contexte pratique, la précision de la solution dépendra du temps que l'utilisateur est prêt à accorder à l'algorithme. Nous illustrons la progression des solutions en reprenant, sous forme graphique, les résultats obtenus sur les instances 100 et 103, en considérant 4 et 10 périodes de réservation. Remarquons que dans les graphiques ci-dessous, le temps (axe horizontal) est représenté sur une échelle logarithmique. Dans tous les cas, la procédure heuristique a été en mesure d'identifier de bonnes solutions beaucoup plus tôt que la formulation exacte et semble converger rapidement vers 97-98% de la valeur de la meilleure solution connue.

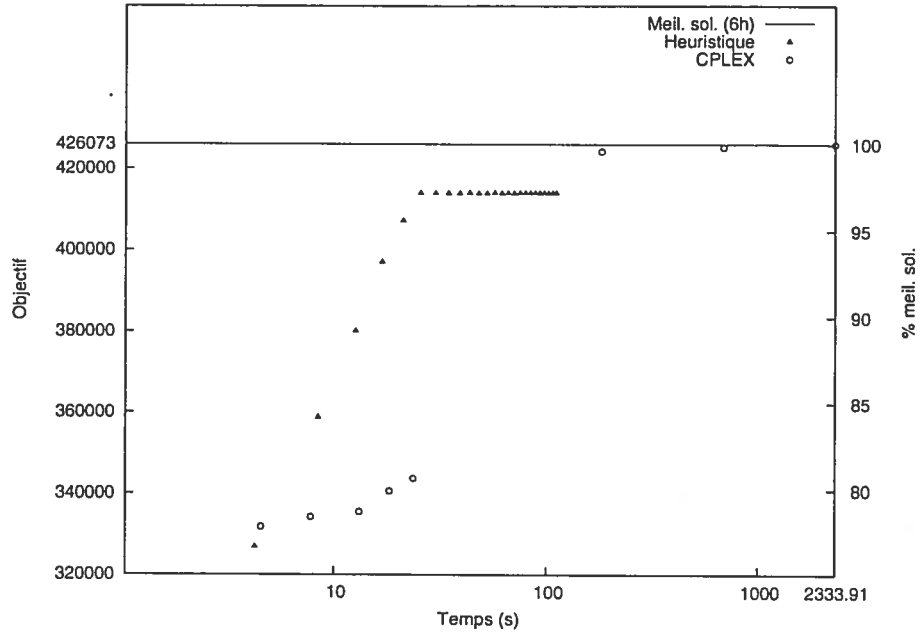


FIG. 7.1 – Résultats de l'heuristique sur l'instance 100 avec 4 périodes.

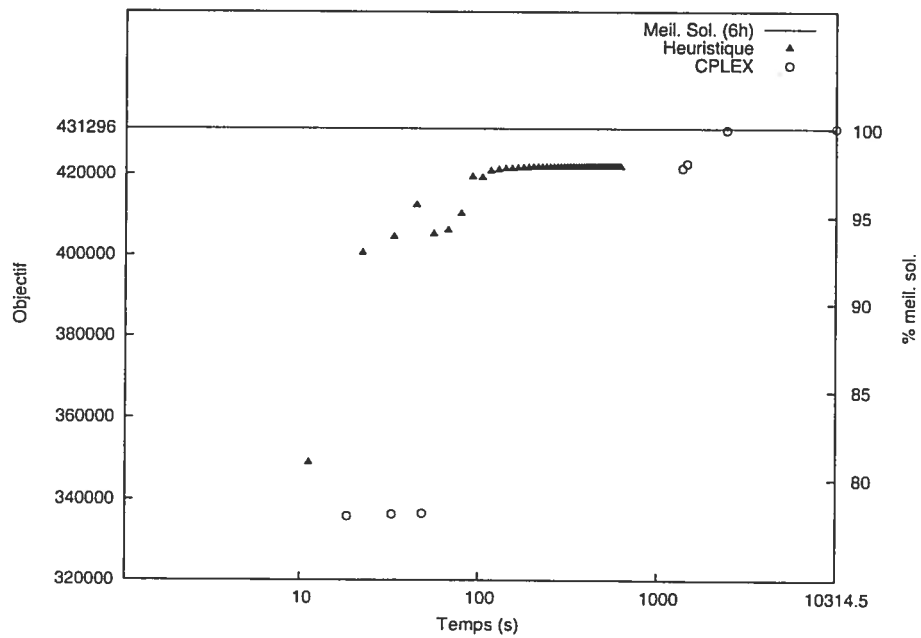


FIG. 7.2 – Résultats de l'heuristique sur l'instance 100 avec 10 périodes.

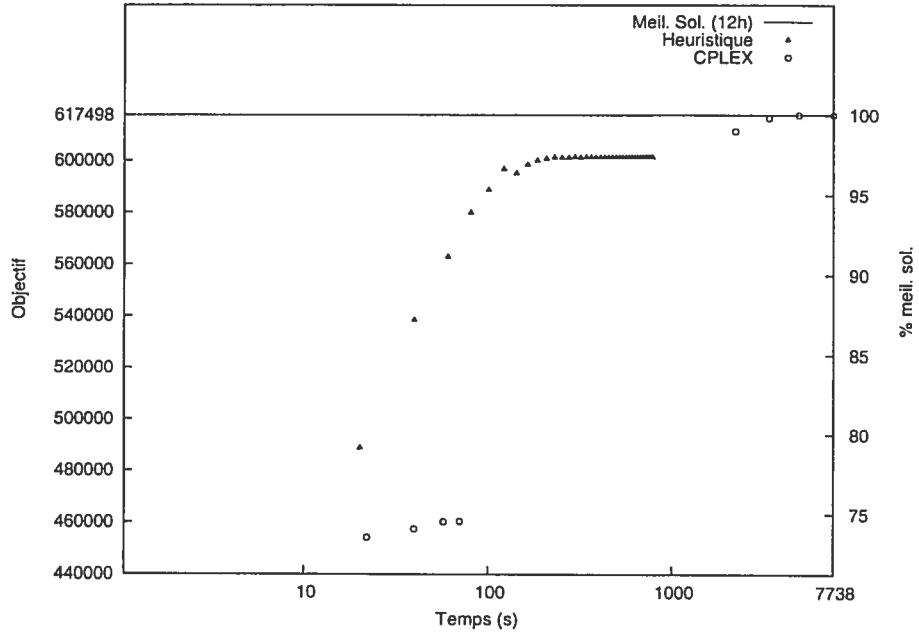


FIG. 7.3 – Résultats de l'heuristique sur l'instance 103 avec 4 périodes.

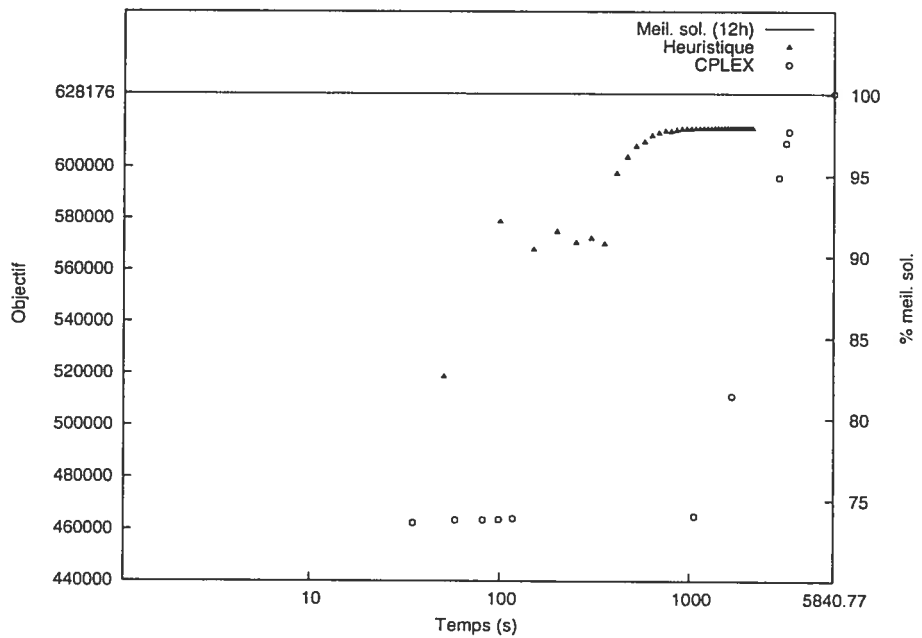


FIG. 7.4 – Résultats de l'heuristique sur l'instance 103 avec 10 périodes.

Relations avec les méthodes de prix d'offre

Nous avons brièvement décrit au chapitre 3 les méthodes de prix d'offre (connues sous le nom anglais de *bid-price methods*). Elles sont principalement utilisées comme sous-routine dans les méthodes d'allocation de la capacité pour calculer la valeur marginale d'un siège au niveau du réseau. Cette valeur est ensuite utilisée pour déterminer si une requête de réservation pour ce siège devrait être acceptée. La façon la plus courante de les obtenir est basée sur la programmation linéaire.

Soit \bar{t} un vecteur de tarifs observés. Il peut s'agir des tarifs courants ou encore de moyennes historiques. Soit r un estimé de demande pour les produits tarifaires en vigueur¹. Typiquement, r est une variable aléatoire dont les paramètres sont estimées à partir de données historiques. La distribution la plus couramment utilisée est la loi normale [TvR98, Bel89]. En supposant un vecteur d'allocation de capacité v , le programme linéaire (déterministe) utilisé par les méthodes de prix d'offre est

$$\begin{aligned}
 & \text{(BP)} \\
 & \max_x \quad \bar{t}Dx \\
 & \text{s.l.c.} \quad Cx \leq v \quad (\lambda) \\
 & \quad \quad \quad x \leq E[r] \\
 & \quad \quad \quad x \geq 0,
 \end{aligned}$$

où la fonction $E[\cdot]$ dénote l'espérance mathématique et les matrices C et D conservent leur signification précédente. Une version plus sophistiquée de ce problème utilise directement la variable aléatoire r plutôt que son espérance $E[r]$. Une telle formulation doit alors être résolue par des techniques propres à la programmation stochastique. Certains auteurs [TvR99, dBFP99, dBFP02] ont comparé les deux approches et sont parvenus à la conclusion que l'approche stochastique ne produit pas systématiquement de meilleurs résultats que l'approche déterministe. Ces auteurs démontrent que sous certaines conditions l'approche déterministe est même préférable à l'approche stochastique. Il semble donc que la complexité supplémentaire qu'implique la considération de la stochasticité de la demande dans les modèles de prix d'offre ne se traduise pas immédiatement par des bénéfices concrets. L'évaluation des paramètres de la distribution

¹Nous avons jusqu'à maintenant supposé que nous connaissions la demande *globale* sur chaque marché. Cependant, les méthodes des prix d'offre basées sur la programmation linéaire présument que la demande est connue de façon beaucoup plus détaillée, c'est-à-dire pour *chaque* produit tarifaire sur chaque *vol*. Comme il ne s'agit pas de la même quantité, nous utilisons un symbole différent pour la dénoter, soit r plutôt que b .

statistique de r peut aussi comporter certains problèmes. D'abord, la demande pour une combinaison vol/produit peut avoir une valeur espérée faible mais une variance importante. Sur certaines routes peu utilisées, par exemple dans un large réseau à structure radiale, il peut être difficile d'observer un nombre suffisant de réservations pour permettre une estimation fiable des paramètres de la distribution de la demande pour chaque combinaison vol/produit. Une autre difficulté inhérente à l'estimation de la demande à partir de données de vente historiques est la «détronquation» des données (ce qui dans la littérature anglophone est connu sous le nom de *uncensoring* ou *unconstraining*). Les données dont nous disposons ne concernent que les clients qui ont acheté un billet; nous ne connaissons rien de ceux qui désiraient effectuer une réservation mais qui n'ont pas trouvé de siège disponible (c'est pourquoi les données de vente sont dites «tronquées»). La détronquation de données de vente historiques est une tâche fondamentalement statistique qui outrepassé la portée de cette thèse et nous ne nous y attarderons pas. McGill et van Ryzin [MvR99] donnent plusieurs références sur ce sujet et la thèse doctorale de Zeni [Zen01] y est entièrement consacrée.

L'intérêt du programme BP réside dans le vecteur de multiplicateurs λ . Ces valeurs indiquent la contribution marginale au niveau du réseau d'un siège supplémentaire sur chaque segment. En additionnant les valeurs correspondantes aux segments d'un vol, on obtient un «prix d'offre» qui nous permet d'approximer le coût d'opportunité d'un siège sur ce vol et ainsi déterminer si une requête de réservation pour ce siège devrait être acceptée.

Les vecteurs ξ que nous obtenons en résolvant le problème GLOB ont une interprétation économique similaire aux prix d'offre λ du modèle BP. Cependant, le modèle GLOB, de par la structure bi-niveau des sous-problèmes résolus à chaque période, tient compte de la compétition et du comportement des passagers, des éléments qui sont absents du modèle de prix d'offre BP. Le modèle GLOB résout aussi le problème de tarification en identifiant des vecteurs de tarifs t_n propres à chaque période de réservation.

Afin d'illustrer la différence entre les prix d'offre calculés à partir de variables duales et les prix optimaux (au sens de notre modèle), nous allons comparer, pour certaines combinaisons vols/produits, les tarifs donnés par les variables décisions t_n et les prix d'offres calculés à partir des variables duales du modèle BP.

La méthodologie que nous avons utilisé pour obtenir les prix d'offre est la suivante. Nous avons d'abord utilisé le modèle GLOB pour obtenir, à chaque période n , un vecteur de prix \bar{t}_n et un

vecteur de flots \bar{x}_n pour les vols du meneur. Ces vecteurs de flots servent de base au vecteur de demande r du modèle BP. Nous ajoutons à ces flots un «bruit» aléatoire ϵ uniformément distribué sur $[-\frac{1}{2}, \frac{1}{2}]$. Ainsi, dans le modèle de prix d'offre, la demande à la période n pour le produit p sur le vol f est

$$r_{fp} = (\bar{x}_{nfp} + \epsilon)^+.$$

Nous utilisons ensuite cette demande dans le modèle BP avec \bar{t}_n comme vecteur de prix. Le prix d'offre de la combinaison vol/produit (f, p) à la période n est alors

$$\tau_{fnp} = \sum_{s \in S(f)} \lambda_s.$$

Les résultats présentés ci-dessous furent obtenus en appliquant cette méthode sur l'instance 100 avec 4 périodes de réservation. Nous présentons un sous ensemble des résultats à la période 1 et à la période 3.

n	f	g	p	t_{nfp}	τ_{fnp}	x_{nfgp}
1	77653	13	216973	238.32	41.01	1.50
	77909	1	216973	240.32	240.32	6.00
	77909	5	216973	240.32	240.32	5.62
	77909	13	216973	240.32	240.32	3.38
	84582	1	216972	168.79	168.79	30.69
	84582	13	216972	168.79	168.79	7.71
	84586	5	216972	168.79	166.02	10.07
	84586	13	216972	168.79	166.02	17.23
	84589	5	216972	168.79	0.00	18.70
	101291	5	216979	138.39	138.39	12.43
	101291	13	216979	138.39	138.39	21.17
	101301	13	216979	138.39	138.39	33.60
	102289	1	216979	142.39	142.39	5.48
	102289	5	216979	142.39	142.39	28.12
	102292	5	216979	142.39	136.62	27.30
	102294	13	216979	137.89	0.00	4.03
	102303	1	216979	142.39	142.39	27.30
3	77653	5	216973	238.32	238.32	6.12
	77660	13	216973	236.32	233.55	10.00
	77903	13	216973	235.32	232.55	10.00
	83999	13	216972	163.79	163.79	18.20
	84002	13	216972	167.29	164.52	18.20
	84576	13	216972	163.79	161.02	25.60
	84579	13	216972	164.29	159.89	18.20
	84582	1	216972	173.29	173.29	15.35
	84586	5	216972	173.29	166.02	18.20
	84589	5	216972	173.29	173.29	25.35
	84592	13	216972	166.29	163.52	18.20
	84595	13	216972	166.29	166.29	25.60
	101306	13	216979	133.39	133.39	22.40
102284	13	216979	132.39	129.62	22.40	
102286	13	216979	134.89	134.89	26.40	
102288	5	216979	136.89	136.89	22.40	
102291	5	216979	146.89	146.89	12.61	

TAB. 7.2 – Comparaisons entre les prix «optimaux» et les prix d'offre pour l'instance 100.

Les résultats de cette expérience tendent à démontrer que les prix d'offre obtenus à partir des valeurs des variables duales aux contraintes de capacité d'un modèle d'optimisation sur réseau constituent des bornes *inférieures* aux tarifs offerts au niveau du réseau. En présence d'hétérogénéité de la clientèle, le prix d'offre d'une combinaison itinéraire/produit correspond au tarif maximal du groupe d'utilisateur ayant la plus forte propension à payer dont la demande résiduelle pour cette combinaison est non nulle. Comme les résultats précédents l'indiquent, ce

tarif est inférieur ou égal au tarif calculé par le modèle GLOB.

Il est difficile de mesurer l'impact d'un contrôle d'inventaire comme celui du modèle GLOB par rapport à une approche de prix d'offre. Il serait intéressant d'effectuer des simulations, mais cela implique la modélisation du processus d'arrivée des requêtes de réservation dans un contexte de compétition et d'hétérogénéité de la clientèle, ce qui dépasse le cadre de cette thèse.

Chapitre 8

Conclusion

Cette thèse propose une nouvelle approche pour la résolution conjointe et simultanée des problèmes de tarification et d'allocation de la capacité en transport aérien.

Le développement de modèles de tarification réalistes et pratiques et leur intégration dans le processus dynamique d'allocation de la capacité constituent un défi auquel auront à faire face les transporteurs aériens dans les années à venir. Si ces efforts sont couronnés de succès, il y a fort à parier que d'autres industries aux caractéristiques similaires tenteront d'adapter ces nouvelles approches à leur problématiques propres, comme ce fut le cas au cours des années 1990 avec les méthodes de contrôle d'inventaire.

Nos travaux de recherche se voulaient avant tout pratiques; notre objectif était et demeure le développement de méthodes qui soient applicables dans l'industrie. C'est pourquoi nous avons porté une attention particulière à ce que les modèles et algorithmes que nous avons proposés soient testés et validés sur des données réelles dans un contexte que nous estimons réaliste. Cette motivation pratique ne nous a toutefois pas empêché de définir et valider les fondements théoriques de notre approche. Une revue bibliographique poussée nous a permis d'établir l'état actuel des recherches en optimisation des revenus et de positionner nos travaux en mettant en valeur leur caractère novateur. En ayant recours à la littérature économique, nous avons été en mesure d'asseoir notre approche sur des bases théoriques existantes et de justifier certaines hypothèses importantes, notamment en ce qui concerne l'utilité et la rationalité des consommateurs.

La principale contribution de la thèse est l'étude des propriétés d'un modèle de tarification bi-niveau et l'exploitation de ces propriétés dans un algorithme de résolution spécialisé. Les résultats numériques obtenus avec cet algorithme sont en général bien meilleurs que ceux obtenus avec le solveur commercial CPLEX. Entre autres, nous avons été en mesure d'obtenir plus rapidement de meilleures solutions, bien que les taux d'exploration de notre implantation soit, pour des raisons techniques, inférieurs à ceux de CPLEX. Nous avons étudié empiriquement les différentes stratégies utilisées par l'algorithme, notamment les critères d'exploration, de branchement et de sélection des nœuds. Nous avons aussi exploré des méthodes alternatives de résolution des relaxations linéaires en chaque nœud qui exploitent la structure particulière des relaxations linéaires. Les résultats numériques obtenus avec ces méthodes ont toutefois été décevants au point de vue de la performances et de la stabilité numérique.

Une fois le modèle de tarification et son algorithme de résolution bien en place, nous avons proposé un cadre plus général d'optimisation des revenus dans un contexte dynamique comme on retrouve dans l'industrie aérienne. Ce cadre, inspiré de la programmation dynamique, permet de poser conjointement les problèmes de tarification et d'allocation de la capacité. Bien que certaines de ses propriétés se généralisent à n'importe quelle méthode de tarification, nous nous sommes plus longuement intéressés au cas particulier où la tarification s'effectue par l'entremise de notre modèle bi-niveau. En s'appuyant sur les propriétés de concavité de la fonction de revenu, nous avons proposé une méthode de résolution de type gradient projeté. Cette méthode est heuristique puisque nous n'utilisons pas le gradient (qui n'existe pas toujours) mais l'approximons par un vecteur de multiplicateurs qui possède une interprétation économique très semblable. Les résultats numériques obtenus tendent à confirmer la validité de cette approche.

En annexe A, on retrouve une description de la procédure d'étalonnage du modèle à laquelle nous faisons référence à quelques reprises et que nous avons choisi de présenter en annexe par souci d'exhaustivité. Le développement de cette procédure n'a toutefois pas fait partie directement des travaux de recherche entourant cette thèse. Il s'agit plutôt d'un effort s'inscrivant dans un projet de recherche plus large auquel participent plusieurs chercheurs. Nous présentons en annexe C la librairie PyPlex, un outil logiciel que nous avons développé pour faciliter l'implantation et la validation de modèles mathématiques et d'algorithmes. Encore là, il ne s'agit pas de travaux directement concernés par cette thèse mais plutôt d'un outil complémentaire qui nous fut bien utile à plusieurs occasions. Finalement, l'annexe D contient un glossaire de termes propres au domaine de l'optimisation des revenus en transport aérien. Pour la plupart de ces concepts, il n'existait pas, à notre connaissance, d'équivalents français aux termes anglais que l'on retrouve

dans la littérature spécialisée. Les termes francisés que nous présentons sont donc une proposition en ce sens ; certains sont d'heureuses trouvailles, d'autres moins. La terminologie française en optimisation des revenus demeure donc ouverte à la discussion et de nouvelles suggestions sont les bienvenues.

Nous sommes grandement intéressés à poursuivre nos travaux afin de raffiner certains aspects présentés dans cette thèse et explorer de nouvelles avenues de recherche. En ce qui concerne l'algorithme spécialisé, nous croyons que des heuristiques basées sur la situation du marché et le comportement historique des passagers pourraient se traduire par de nouvelles règles de fixation de variables en chaque nœud de l'arbre d'énumération (les règles lexicographiques que nous avons présentées au chapitre 6 sont exactes). Nous désirons aussi étudier la topologie de réseaux de transport aériens (spatio-temporels) et tenter de mettre au point des algorithmes d'exploration de réseau qui permettent de séparer le réseau global en sous-réseaux faiblement connexes, dans le but de réduire la taille des instances que nous devons résoudre sans affecter de façon significative le revenu total. Une direction de recherche intéressante est l'incorporation à notre approche de l'incertitude propre au comportement des consommateurs et aux fluctuations de demande. En évaluant les distributions de ces quantités et en les incluant comme variables aléatoires dans le modèle, nous pourrions explicitement tenir compte de cette incertitude propre au transport aérien. Il s'agit d'un aspect excitant du point de vue de la modélisation mais qui implique des méthodes de résolution plus sophistiquées, par exemple l'emploi de la programmation stochastique.

Bien que nous nous soyons, dans le cadre de cette thèse, concentrés exclusivement sur l'industrie aérienne, une avenue de recherche très intéressante est l'adaptation et la validation des approches développées dans cette thèse à d'autres domaines partageant des caractéristiques similaires avec le transport aérien. Nous avons déjà étudié les performances de ces approches dans le cadre d'un projet de recherche en transport ferroviaire et aurons l'occasion au cours des prochains mois nous attarder au domaine des télécommunications. Les idées fondamentales demeurent les mêmes, mais les particularités de chaque champ d'application requièrent toujours de nouveaux efforts de modélisation et de développement algorithmique.

Comme nous l'avons mentionné en introduction, le développement de modèles de tarification et leur intégration au processus d'allocation de la capacité est toujours à l'état embryonnaire dans l'industrie aérienne. Il s'agit sans doute d'un problème auquel s'intéresseront les chercheurs pendant encore plusieurs années. L'approche que nous proposons dans cette thèse ne se veut pas

une solution définitive mais plutôt une nouvelle direction dans un champ de recherche encore largement ouvert.

Annexes

Annexe A

Procédure d'étalonnage

La valeur des paramètres comportementaux qu'utilise le modèle est obtenue à partir d'une procédure basée sur le principe de l'optimisation inverse. Ce chapitre décrit cette méthodologie et présente la procédure d'étalonnage du modèle.

A.1 Optimisation inverse

Supposons que nous soyons en présence d'un phénomène physique dont nous pouvons observer les effets. Nous tentons d'approximer ce phénomène à l'aide d'un modèle mathématique paramétrisé. Il nous est impossible de mesurer ou de quantifier directement la valeur de ces paramètres. Cependant, les effets observables du phénomène correspondent aux solutions de notre modèle. En supposant que les paramètres du modèle soient stables et que les caractéristiques physiques du phénomène dépendent effectivement de ces paramètres, nous pouvons considérer un effet observable du phénomène comme une solution «optimale» du problème et chercher l'ensemble de paramètres tel que la solution produite par le modèle en se basant sur ces paramètres s'approche le plus possible de la solution «optimale» observée. En d'autres mots, le problème direct consiste à prédire, à l'aide d'un ensemble de paramètres, un effet physique du phénomène, alors que le problème inverse consiste, à partir d'un effet observable, à estimer les paramètres du modèle.

Formellement, soit $f : X \mapsto \mathbb{R}$ une fonction objectif dépendant d'un vecteur de paramètres θ , lui-même élément d'un ensemble admissible Θ . Étant donné θ , nous cherchons à résoudre le

problème d'optimisation

$$\min_{x \in X} f(\theta; x),$$

où X représente l'ensemble des solutions admissibles, que nous supposons non vide.

Nous sommes dans l'impossibilité de mesurer directement θ , mais nous pouvons «observer» un vecteur admissible \bar{x} , que nous considérons comme une solution optimale du phénomène que nous cherchons à modéliser. Le problème d'optimisation inverse correspondant est alors

$$\begin{aligned} \min_{\substack{\theta \in \Theta, \\ x \in X}} \|\bar{x} - x\|_l \\ \text{s.l.c. } x \in \arg \min \{ f(\theta; x') \mid x' \in X \}. \end{aligned}$$

Il s'agit d'un problème d'optimisation à deux niveaux de dimension supérieure au problème originel.

A.1.1 Revue bibliographique

La méthodologie connue sous le nom d'«optimisation inverse» est utilisée depuis plusieurs années dans les sciences physiques (par exemple en géographie et en géologie) ainsi qu'en génie. Les modèles utilisés dans ces domaines ne sont cependant pas des modèles d'optimisation mais plutôt des descriptions mathématiques de phénomènes physiques faisant typiquement intervenir des équations différentielles paramétriques. Deux articles par Burton et Toint [BT92, BT94] établissent cependant un lien entre ces domaines et la programmation mathématique en présentant un modèle d'optimisation ayant des applications en géologie (plus précisément en «tomographie sismique»). Dans ce modèle, des zones sismiques sont représentées par les nœuds d'un graphe dont les arcs décrivent les interactions entre deux zones. En supposant que les ondes sismiques se propagent par des plus courts chemins entre les zones, les auteurs modélisent les effets des tremblements de terre en s'appuyant sur un algorithme de plus courts chemins inverse dans leur graphe sismique.

Des articles parus au cours de la dernière décennie s'intéressent à d'autres aspects de l'optimisation inverse dans le contexte de la programmation mathématique, par exemple Zhang et Liu [ZL96], et Yang, Zhang et Ma [YZM97]. Deux articles récents par Ahuja et Orlin [AO01] et Heuberger [Heu] présentent certains résultats théoriques et décrivent les domaines d'application de l'optimisation inverse. L'article de Heuberger contient aussi une quarantaine de références sur

le sujet. Feron, Gentry et Saligrama [GSF99] présentent une adaptation de l'optimisation inverse à la programmation dynamique. Finalement, Brotcorne, Labbé, Marcotte et Savard [BLMS01] utilisent l'optimisation inverse dans un modèle de tarification en transport autoroutier.

La nature du problème d'optimisation inverse présenté dans la majorité des articles à ce sujet (y compris ceux cités ci-dessus) n'est cependant pas exactement la même que celle que nous avons présentée en introduction. Ces articles définissent le problème ainsi : soit un problème $P(c)$, disons linéaire, dont le paramètre est en fait le vecteur de coûts, i.e.

$$P(c) = \min \{ cx \mid Ax = b, x \geq 0 \}$$

et soit x^0 une solution *admissible* de ce problème. Le problème inverse consiste à trouver un vecteur de coûts d qui minimise $\|d - c\|_l$ et tel que x^0 soit une solution *optimale* de $P(d)$. On doit ainsi résoudre

$$\begin{aligned} \min_{d, \mu} \quad & \|d - c\|_l \\ \text{s.l.c.} \quad & d - \mu A \geq 0 \\ & (d - \mu A) x^0 = 0, \end{aligned}$$

où μ est le vecteur de multiplicateurs de la contrainte $Ax = b$. Ce problème est linéaire lorsque $l = 1$. Dans le cas du modèle de plus courts chemins inverses utilisé en géologie, le vecteur de coûts c (coût des arcs) peut être estimé, de façon imparfaite, à partir de données sismiques historiques représentées par le vecteur x^0 . En utilisant l'optimisation inverse, les auteurs peuvent obtenir un coût d pour les arcs reliant les zones sismiques de sorte que les observations historiques x^0 constituent une solution optimale de leur modèle tout en se rapprochant le plus possible des coûts estimés c .

A.2 Étalonnage du modèle BAFO

A.2.1 Formulation

Le modèle BAFO présenté au chapitre 5 repose sur un ensemble de paramètres décrivant le comportement des passagers lors de l'achat de billets. Ces paramètres ont été dénotés par le vecteur $\theta = (\alpha, \beta, \dots)$ et sont propres à chaque groupe d'utilisateurs g sur chaque marché k . Par exemple, α_{gk} représente la valeur du temps (en dollars par heure, disons) propre aux usagers de ce

Norme utilisée

Dans le modèle ci-dessus, nous n'avons pas spécifié la norme¹ utilisée dans la fonction objectif (i.e. nous n'avons pas spécifié la valeur du paramètre l). Comme nous désirons un objectif linéaire, nous choisissons la norme «valeur absolue», i.e. $l = 1$. À chaque terme variable x , nous associons deux nouvelles variables w^+ et w^- . Le terme $|x - \bar{x}|$ de la fonction objectif est remplacé par $w^+ + w^-$ et nous ajoutons les contraintes suivantes à la formulation :

$$\begin{aligned} w^+ + w^- &= x - \bar{x} \\ w^+ &\geq 0 \\ w^- &\leq 0. \end{aligned}$$

Les variables y sont traitées de la même façon.

En reformulant les contraintes de complémentarité sous forme linéaire à l'aide de variables binaires (comme nous l'avons fait au chapitre 5), nous obtenons un modèle d'étalonnage linéaire mixte avec variables entière (binaires).

A.2.2 Disponibilité des données

La formulation précédente du modèle d'étalonnage suppose la connaissance des prix et des flots historiques de la compétition (ce que nous désignons respectivement par les symboles \bar{t}^2 et \bar{y}). En pratique, ces informations ne sont pas toujours disponibles. Les transporteurs ont la possibilité d'acheter des données disponibles commercialement et compilées par des compagnies comme MIDT². Cependant, ces données ne couvrent que les billets ayant été réservés par l'entremise des principaux systèmes de réservation globaux. Ces systèmes ne sont généralement utilisés que par les transporteurs traditionnels, les transporteurs à coût réduit ayant leur propres canaux de distribution. Aussi, de plus en plus de billets sont réservés en-ligne directement sur le site du transporteur, sans avoir recours à un agent de voyage utilisant un système de réservations global (le nombre de réservations sur le Web atteindrait maintenant entre 20 et 30% du nombre total de billets vendus et même jusqu'à 60% sur certains marchés). Ces réservations ne sont pas

¹La norme- l d'un vecteur $x \in \mathbb{R}^n$ est la fonction à valeur réelles $\|\cdot\|_l$ définie comme $\|x\|_l \equiv \left(\sum_{i=1}^n |x_i|^l\right)^{\frac{1}{l}}$. On

a alors $\|x\|_1 = \sum_{i=1}^n |x_i|$, $\|x\|_2 = \sqrt{\sum_{i=1}^n x_i^2}$ et $\|x\|_\infty = \max_{1 \leq i \leq n} \{|x_i|\}$.

²Marketing Information Data Tapes

compilées par les systèmes globaux et seuls les transporteurs en connaissent le nombre exact. Bref, l'information sur les ventes historiques des compétiteurs est au mieux incomplète et dans certains cas carrément inexistante. Ce phénomène ne peut que prendre de l'ampleur à mesure que le nombre de réservations en-ligne augmente.

La non disponibilité des données de vente historiques de la compétition peut s'avérer problématique dans la procédure d'étalonnage de notre modèle (ou de tout autre modèle de tarification reposant sur une description numérique des habitudes d'achat des consommateurs). Toutefois, les transporteurs majeurs disposent en général de données détaillées sur leur propres ventes, qui sont typiquement compilées et sauvegardées de façon électronique. Ces données sont aussi plus complètes et cohérentes que celles compilées et revendues par une entité externe. En faisant l'hypothèse que les passagers du meneur (segmentés en groupes d'utilisateurs cohérents) ont le même comportement d'achat que ceux des autres transporteurs, nous pouvons nous appuyer uniquement sur les données de vente historique du meneur pour effectuer l'étalonnage. Cette hypothèse permet de contourner le problème de la non disponibilité ou de la mauvaise qualité des données de vente des compétiteurs et simplifie la procédure du point de vue technique puisque nous devons traiter qu'un seul format de données historiques, celui du meneur.

Annexe B

Utilité et rationalité

Le modèle que nous développons au chapitre 5 fait appel aux notions d'utilité et de rationalité. Ces concepts sont étudiés par les économistes depuis plus d'un siècle. Le but de cette annexe est de discuter de ces notions, de donner quelques références importantes et de présenter certaines propriétés mathématiques qui permettent de formaliser leur utilisation dans notre modèle. La majorité des informations qui suivent ont été tirées d'ouvrages se rapportant aux sciences économiques, et plus particulièrement à la théorie des jeux.

B.1 Notion d'utilité

B.1.1 Définition et justification

Le Nouveau Petit Robert définit l'**utilité** comme le «caractère de ce qui est utile». Cette définition, bien qu'intuitive et directe, est trop vague pour l'utilisation que nous faisons de ce terme dans le présent contexte, i.e celui d'un agent économique devant choisir parmi un ensemble d'**alternatives**¹ (qui peuvent prendre la forme de différents objets économiques, tels des biens de consommation, des opportunités, des stratégies, etc.) celle qui lui convient le mieux. Chacune de ces alternatives est caractérisée par un ensemble d'attributs qui lui sont propres. Ainsi, nous

¹Le sens que nous donnons au mot «alternative» dans ce contexte, qui n'est rien d'autre qu'un emprunt direct à la terminologie anglaise, est critiqué dans son usage français.

posons :

DÉFINITION B.1

L'utilité est la valeur monétaire des attributs d'une alternative aux yeux d'un agent économique.

□

Connaissant l'utilité des alternatives parmi lesquelles il doit choisir, l'agent en question peut déterminer celle qui lui procurera la plus grande satisfaction, le meilleur rendement, les meilleurs avantages, etc. Une notion apparentée à l'utilité est la **préférence**. Dans ce contexte, la préférence d'un individu pour un objet économique ira à l'alternative qui lui procure la plus grande utilité, en supposant bien entendu que cet individu soit placé devant un choix entre plusieurs alternatives.

La notion d'utilité était connue des mathématiciens du XIXe siècle et du début du XXe siècle¹, mais ce n'est qu'à partir des années quarante que des économistes et des mathématiciens se sont intéressés à l'utilité d'un point de vue mathématique et ont tenté d'en concevoir une théorie axiomatique.

Les efforts de modélisation mathématique de problèmes économiques pratiques supposent généralement que le but des participants est avant tout l'argent, i.e. minimiser les coûts ou maximiser les revenus. On suppose que l'argent est une commodité infiniment divisible, substituable, transférable et identique qualitativement et quantitativement. Si nous désirons tenir compte de la notion d'utilité dans notre modélisation économique, nous devons trouver un moyen de l'exprimer de façon numérique, selon les mêmes unités que les autres éléments du modèle, qui ne sont rien d'autres que des unités monétaires (e.g. dollars). Cependant, avant même d'entreprendre cette démarche, nous devons répondre à la question suivante : Est-il justifié (ou même strictement possible) d'exprimer l'utilité comme une quantité numérique ?

Parmi les premiers auteurs à s'être penchés sur cette importante question en modélisation économique, on retrouve von Neumann et Morgenstern, qui y ont consacré une section entière dans

¹L'économiste italien Vilfredo Pareto (1848-1923) s'est longuement intéressé à la notion d'utilité, d'un point de vue surtout qualitatif. Il a notamment établi une distinction entre l'«utilité», la qualité de ce qui participe au bien-être d'un individu, et l'«ophélimité», la qualité de ce qui est désirable pour un individu. L'optimalité de Pareto est alors une situation d'«ophélimité maximale», i.e. une situation dans laquelle aucun individu ne peut améliorer sa propre situation (d'un point de vue de ce qu'il désire) en empirant la situation d'un autre individu. Dans le présent document, nous n'adoptons pas cette distinction et nous restreignons à la définition donnée ci-dessus du concept d'«utilité».

leur ouvrage séminal sur la théorie des jeux [vNM53]. Selon ces auteurs, l'utilité peut être perçue comme d'autres quantités physiques telles la masse, la force, la charge, etc. Tout comme l'utilité, ces quantités possèdent une définition intuitive et directe. Par exemple, en physique mécanique, la masse est la quantité de matière que possède un corps, ou encore le rapport constant entre la force s'exerçant sur ce corps et l'accélération qu'il subit. Elles ont aussi une influence indéniable lorsqu'observées et mesurées de façon empirique. De la même façon, on peut établir des liens entre une théorie faisant intervenir l'utilité et des phénomènes économiques observés empiriquement. La notion d'utilité n'est donc pas un concept entièrement tautologique qui n'existe que par et pour lui-même. En d'autres termes, l'utilité possède un sens théorique, mais permet aussi d'expliquer des phénomènes économiques observables quotidiennement.

L'approche consistant à traiter l'utilité de façon numérique s'inscrit aussi dans le cadre de la méthode scientifique classique : afin de se concentrer sur un problème principal (dans notre cas maximiser les revenus ou minimiser les coûts), on simplifie les sous-problèmes dont dépend le problème principal en les schématisant à l'aide de certaines hypothèses de base. Cependant, l'hypothèse que l'utilité puisse s'exprimer de façon numérique n'est pas si naïve qu'elle apparaît à première vue.

Historiquement, plusieurs quantités physiques, dont celles énumérées ci-haut, ont été difficiles à exprimer numériquement, malgré qu'elles eussent possédé une définition intuitive claire ; le développement d'une théorie acceptable pour ces quantités ne fut pas direct. Considérons par exemple la chaleur. On peut facilement procéder à une classification qualitative de deux corps selon la chaleur qu'ils possèdent, i.e. déterminer lequel de ces deux corps est le plus chaud, mais il est difficile de déterminer par quel facteur le corps le plus chaud est « plus chaud » que le corps le plus froid. La question de l'additivité se pose également : Comment additionner la chaleur de deux corps séparés ? La théorie thermique a révélé que la chaleur d'un corps peut être exprimée à l'aide de deux nombres : la quantité de chaleur (i.e. d'énergie), laquelle est additive, et la température, laquelle est non additive mais possède une échelle absolue. L'histoire scientifique nous indique que le même cheminement a été répété maintes fois dans le développement d'une théorie numérique acceptable pour d'autres quantités observables empiriquement.

Remarquons que souvent les quantités physiques permettent certaines opérations algébriques. Par exemple, la masse permet l'opération « addition » (au sens scalaire) alors que ce n'est pas le cas pour la position. La vitesse et l'accélération permettent aussi l'addition, mais sous forme vectorielle. Notons de plus que, lorsqu'une quantité physique d'intérêt est décrite de façon quan-

titative, cette description n'est pas nécessairement unique. Il peut en effet exister une famille entière de descriptions possibles. Supposons que le passage d'une description à une autre s'effectue à l'aide d'une opération algébrique ω . Ainsi, l'ensemble de toutes les descriptions possibles d'une quantité physique obtenues à l'aide de cette opération algébrique forment ce que les mathématiciens appellent un groupe. Un groupe est un ensemble G associé à une opération binaire ω possédant les propriétés suivantes :

- Fermeture : $\forall a, b \in G, a \omega b \in G$.
- Associativité : $\forall a, b, c \in G, (a \omega b) \omega c = a \omega (b \omega c)$.
- Identité : $\forall a \in G, \exists e \in G$ t.q. $a \omega e = a$.
- Inverse : $\forall a \in G, \exists d \in G$ t.q. $a \omega d = e$.

Un groupe abélien possède en plus la propriété

- Commutativité : $\forall a, b \in G, a \omega b = b \omega a$.

Suivant la nomenclature de von Neumann et Morgenstern, on dira alors qu'une quantité physique est numérique sous l'opération ω si elle définit un groupe. Par exemple, la quantité «distance» est numérique sous l'opération «multiplication par une constante positive».

À mesure que la théorie sur une quantité physique se développe, la nature de l'opération ω se précise. Revenons à la chaleur. Initialement, la température d'un corps était considérée comme une quantité numérique sous n'importe quelle transformation (i.e. opération ω) monotone positive. Suite à la création d'une échelle pour la température, on a limité ω à des transformations linéaires et depuis que le zéro absolu a été fixé, on se limite à des transformations de type «multiplication par une constante positive» pour décrire la température. On se retrouve donc aujourd'hui avec d'un côté l'échelle métrique de Kelvin, laquelle fixe le zéro absolu à $-273,15$ Celsius, et de l'autre l'échelle impériale de Rankine qui fixe le zéro absolu à $-459,67$ Fahrenheit. Ainsi une mesure en degrés Kelvin n'est qu'un multiple positif d'une mesure équivalente en degrés Rankine (i.e. la nouvelle mesure est obtenue par une transformation de la forme $x' = ax$ où $a > 0$). Par contre, si l'on fait abstraction du zéro absolu (ce qui est habituellement le cas en dehors des laboratoires de chimie et de physique) et que l'on utilise d'un côté l'échelle de Celsius et de l'autre celle de Fahrenheit, on doit avoir recours à des transformations linéaires de la forme $x' = ax + b$ pour convertir une mesure d'un système à un autre.

L'exemple précédent suggère un lien entre d'une part la nature de l'opération ω générant la famille de descriptions d'une quantité physique d'intérêt et d'autre part le type d'échelle utilisé pour cette quantité. Une classification des échelles possibles est proposée par Rapaport [Rap89]. On y retrouve :

1. L'échelle ordinale

Il s'agit du type d'échelle le plus «faible». Une échelle ordinale permet de déterminer laquelle, parmi deux quantités, est plus «grande» (ou symétriquement plus «petite») que l'autre, ou encore déterminer si les deux quantités sont égales. Elle ne permet cependant pas de déterminer par quel facteur la quantité la plus grande est «plus grande» que la plus petite (et vice-versa) ou de calculer la différence entre les deux quantités. Cette échelle est invariante sous des transformations monotones positives.

2. L'échelle différentielle

L'échelle différentielle est plus forte que l'échelle ordinale car les intervalles entre des quantités successives sur cette échelle (i.e. la différence entre ces alternatives) doivent eux-mêmes former une échelle ordinale : on doit pouvoir les ordonner selon leur magnitude et aucune transformation monotone positive de l'échelle ordinale des quantités ne peut modifier l'ordre des intervalles.

3. L'échelle à intervalles

L'échelle à intervalles permet l'ordonnement de quantités selon le quotient des différences entre ces quantités. Il s'agit du type d'échelle le plus souvent utilisé pour mesurer des quantités physiques (e.g. degrés Celcius). C'est aussi ce type d'échelle qu'utilisent plusieurs auteurs pour exprimer l'utilité dans leurs modèles. Ce type d'échelle est invariant sous des transformations linéaires de la forme $x' = ax + b$, car ce type de transformation laisse les quotients des intervalles entre deux quantités inchangés.

4. L'échelle à quotient

Il s'agit d'une échelle à intervalles dont l'origine a été fixée. L'échelle Kelvin pour la température en est un exemple. Ce type d'échelle est invariant sous des transformations de la forme $x' = ax$ où $a > 0$, i.e. multiplication par une constante positive.

La notion d'utilité sur laquelle nous nous baserons possède une échelle à quotient, la forme la plus forte. Il s'agit donc d'une quantité numérique sous l'opération «multiplication par une constante positive».

B.1.2 Développement axiomatique

Nous présentons ici certains résultats importants de la théorie axiomatique de l'utilité développée dans les années quarante, cinquante et soixante. Ces résultats ont été publiés dans de nombreux articles ainsi que dans certains livres consacrés entièrement ou en partie à ce sujet. Ce qui suit

est tiré principalement de Fishburn [Fis70] et de Leitman [Lei76]. Notons que cette présentation s'intéresse à l'utilité en absence d'incertitude, sans avoir recours à des notions probabilistes. Il existe une théorie parallèle pour l'utilité dans un contexte non déterministe.

Notation et concepts de base

Soit X un ensemble fini d'alternatives. La **préférence stricte** entre deux alternatives $x, y \in X$ sera dénotée par $x \prec y$ (l'alternative y est préférée à l'alternative x), alors que l'**indifférence** entre ces deux alternatives sera dénotée par $x \sim y$. On peut aussi avoir la **préférence simple** (i.e. non stricte) $x \succsim y$. Dans ce cas, on dit que l'alternative x n'est pas préférée à l'alternative y . Remarquons que l'on peut aussi écrire, de façon symétrique, $y \succ x$ et $y \succsim x$.

Une **fonction d'utilité** est une fonction $u : X \mapsto \mathbb{R}$ assignant à chaque alternative $x \in X$ une utilité numérique $u(x)$.

Soit β une relation binaire sur les éléments de X . On dit que la relation β est

- **réflexive** si $x\beta x, \forall x \in X$,
- **symétrique** si $x\beta y \implies y\beta x, \forall x, y \in X$,
- **antisymétrique** si $x\beta y \wedge y\beta x \implies x = y, \forall x, y \in X$,
- **transitive** si $x\beta y \wedge y\beta z \implies x\beta z, \forall x, y, z \in X$,
- **négativement transitive** si $\neg(x\beta y) \wedge \neg(y\beta z) \implies \neg(x\beta z), \forall x, y, z \in X$,
- **complète** si $x\beta y \vee y\beta x, \forall x, y \in X$,
- **faiblement complète** si $x \neq y \implies x\beta y \vee y\beta x, \forall x, y \in X$

De plus, la relation binaire β sur X est une relation

- **d'ordre faible** si elle est non symétrique et négativement transitive,
- **d'ordre strict** si elle est une relation d'ordre faible faiblement complète,
- **d'ordre strict partiel** si elle est irréflexive et transitive,
- **d'équivalence** si elle est une réflexive, symétrique et transitive.

Si β est une relation d'équivalence sur X , alors la **classe d'équivalence** générée par $x \in X$ est $\beta(x) = \{y \in Y \mid y\beta x\}$, i.e. l'ensemble de tous les éléments de X équivalents à x sous β . De plus, si β est une relation d'équivalence, alors l'ensemble des classes d'équivalence de X sous β sera dénoté par X^β .

Préférence stricte comme relation d'ordre faible

Supposons que la relation de préférence stricte \prec soit une relation d'ordre faible sur un ensemble d'alternatives X . Soient x et y deux alternatives de X . Les relations d'indifférence et de préférence simple sont respectivement définies comme

$$x \sim y \iff \neg(x \prec y) \wedge \neg(y \prec x)$$

et

$$x \succsim y \iff (x \prec y) \vee (y \sim x).$$

Nous avons alors les deux théorèmes suivants, dont les preuves se trouvent dans [Fis70].

THÉORÈME B.1 *Propriétés de \prec, \succsim, \sim*

Supposons que \prec sur X soit une relation d'ordre faible, i.e. asymétrique et négativement transitive. Alors

1. on a soit $x \prec y$, $y \prec x$ ou $x \sim y$, $\forall x, y \in X$,
2. \prec est transitive,
3. \sim est une relation d'équivalence sur X ,
4. $(x \prec y) \wedge (y \sim z) \implies x \prec z$, et $(x \sim y) \wedge (y \prec z) \implies x \prec z$,
5. \succsim est réflexive, transitive et complète,
6. la relation \prec' sur X/\sim définie par $a \prec' b \iff x \prec y$ pour $x \in a$ et $y \in b$ est une relation d'ordre stricte.

□

THÉORÈME B.2 *Existence de la fonction d'utilité*

Si \prec est une relation d'ordre stricte sur X et X/\sim est fini ou infiniment dénombrable, alors il existe une fonction à valeurs réelles $u : X \rightarrow \mathbb{R}$ telle que, $\forall x, y \in X$

$$x \prec y \iff u(x) < u(y),$$

$$x \succsim y \iff u(x) \leq u(y),$$

$$x \sim y \iff u(x) = u(y).$$

□

On constate que lorsque la relation de préférence stricte \prec est une relation d'ordre faible, il est possible de construire une fonction d'utilité à valeurs réelles aux propriétés désirées. Cependant, il y a un problème avec ce type de relation. Comme mentionné au point 3 du théorème B.1, la relation d'indifférence \sim est une relation d'équivalence sur X , i.e. elle est notamment transitive et symétrique. Or la transitivité de la relation d'indifférence peut entraîner certaines contradictions lors de la modélisation si on admet qu'un agent placé devant un choix entre plusieurs alternatives ne soit pas en mesure de choisir parmi certaines «quantités» d'une taille trop petite. À titre d'exemple, supposons que $X = \{a, b, c\}$, i.e. nous devons choisir parmi trois alternatives celle qui nous convient le mieux. Nous préférons clairement l'alternative b à l'alternative a , mais nous sommes indifférents lorsque nous comparons a à c et b à c . Selon notre notation, nous avons donc $a \prec b, a \sim c$ et $b \sim c$. Par la symétrie de \sim on a $b \sim c \iff c \sim b$ et par sa transitivité on a $(a \sim c) \wedge (c \sim b) \implies a \sim b$, une contradiction puisque $a \prec b$.

Il est cependant possible d'imposer certaines restrictions supplémentaires sur la relation de préférence stricte afin d'éviter ce genre de contradiction.

Préférence stricte comme relation d'ordre strict partiel

Supposons maintenant que la relation de préférence stricte soit une relation d'ordre strict partiel sur X , i.e. irréflexive et transitive. Nous pouvons alors avoir, pour $x, y, z \in X$, la situation $x \sim y, y \sim z$ et $x \prec z$, ce qui implique que la relation d'indifférence ne peut être transitive. Il ne s'agit donc pas d'une relation d'équivalence. Nous pouvons cependant définir une nouvelle relation, dénotée \approx , comme

$$x \approx y \iff (x \sim z \iff y \sim x, \forall z \in X).$$

En d'autres termes, nous avons $x \approx y$ lorsque, pour $z \in X$ quelconque, x est indifférent à z et y est indifférent à z .

Nous avons les deux théorèmes suivants, dont la preuve se trouve encore une fois dans [Fis70].

THÉORÈME B.3 Propriétés de \prec, \sim, \approx

1. on a soit $x \prec y, y \prec x, x \approx y$ ou $(x \sim y) \wedge \neg(x \approx y) \forall x, y \in X$,
2. \approx est une relation d'équivalence,
3. $x \approx y \iff ((x \prec z \iff y \prec z) \wedge (z \prec x \iff x \prec y), \forall z \in Z)$,
4. $((x \prec y) \wedge (y \approx z)) \implies ((x \prec z) \text{ et } ((x \approx y) \wedge (y \prec z)) \implies x \prec z$,

5. la relation \prec^* sur X^\approx définie par $a \prec^* b \iff x \prec y$ pour $x \in a$ et $y \in b$ est une relation d'ordre strict partiel.

□

THÉORÈME B.4 *Existence de la fonction d'utilité*

Si \prec est une relation d'ordre strict partiel sur X et X^\approx est fini ou infiniment dénombrable, alors il existe une fonction à valeurs réelles $u : X \rightarrow \mathbb{R}$ telle que, $\forall x, y \in X$

$$x \prec y \implies u(x) < u(y),$$

$$x \approx y \implies u(x) = u(y).$$

□

B.1.3 Utilité d'alternatives à attributs multiples

La discussion précédente portait sur le cas particulier où les alternatives considérées lors du processus de décision ne comportaient qu'un seul attribut x élément d'un ensemble X . Nous allons maintenant nous intéresser au cas général où les alternatives comportent un nombre arbitraire n d'attributs. On peut alors représenter ces attributs comme un vecteur $x = (x_1, x_2, \dots, x_n)$ élément du produit cartésien de n ensembles $X^n = \prod_{i=1}^n X_i = X_1 \times X_2 \times \dots \times X_n$, où X_i représente l'ensemble (que l'on suppose non vide) des valeurs admissibles pour le $i^{\text{ème}}$ attribut. On suppose aussi que les attributs d'une alternative sont indépendants, i.e. la préférence ou l'indifférence entre deux valeurs d'une même attribut n'influencent pas la perceptions des valeurs d'un autre attribut.

Bien entendu, les diverses relations de préférence de d'indifférence décrites ci-dessus s'appliquent ici sous forme vectorielle. On entend par là que $x \beta y$ signifie que $x_i \beta y_i$ pour $i = 1, 2, \dots, n$, où β représente une relation binaire parmi \prec, \succ, \sim et \approx .

B.1.4 Additivité des fonctions d'utilité

Nous savons que, sous certaines conditions sur la relation de préférence stricte \prec , il existe pour un vecteur d'attributs $x = (x_1, x_2, \dots, x_n)$ des fonctions d'utilité à valeur réelles u_1, u_2, \dots, u_n telles

que $u_i : X_i \mapsto \mathbb{R}$ pour $i = 1, 2, \dots, n$. Nous désirons maintenant savoir s'il existe une fonction $u : X^n \mapsto \mathbb{R}$, où $X^n = \prod_{i=1}^n X_i$ est le produit cartésien des ensembles d'attributs admissibles. Si une telle fonction existe, nous aimerions aussi savoir s'il est possible de la construire à partir des fonctions d'utilité u_1, u_2, \dots, u_n propres à chaque attribut.

Le prochain théorème stipule que, sous certaines conditions, une telle fonction u existe et que celle-ci est la somme des fonctions d'utilité de chaque attribut considéré. Ce théorème, tel que présenté ici, se trouve dans [Rap89], mais il existe un développement similaire dans [Fis70].

Afin d'alléger la présentation, nous considérons le cas $n = 2$, i.e. où chaque alternative comporte deux attributs. Ce résultat se généralise facilement pour un nombre arbitraire d'attributs.

THÉORÈME B.5 *Existence et additivité des fonctions d'utilité*

Si

1. la relation de préférence simple \succsim est une relation d'ordre faible,
2. $\forall a, b, c \in X_1$ et $\forall x, y, z \in X_2$, $[(a, z) \succsim (c, y) \wedge ((c, x) \succsim (b, z))] \implies (a, x) \succsim (b, y)$,
3. $\forall a, b \in X_1$ et $\forall x, y \in X_2$, $\exists c \in X_1$ et $\exists z \in X_2$ t.q. $(a, x) \sim (c, y)$ et $(a, x) \sim (b, z)$,
4. pour toute séquence duale $\{(a_i, x_i)\}_{i=1,2,\dots}$ et pour tout $(b, y) \in A_1 \times A_2$, $\exists m, n \in \mathbb{Z}^+$ t.q. $(a_m, x_m) \succsim (b, y) \succsim (a_n, x_n)$ (une séquence duale est une séquence $\{(a_i, x_i)\}$ telle que $a_i \not\sim a_j$ et $x_i \not\sim x_j$ pour tout $i \neq j$),

alors il existe des fonctions $u_1 : X_1 \mapsto \mathbb{R}$, $u_2 : X_2 \mapsto \mathbb{R}$ et $u : X_1 \times X_2 \mapsto \mathbb{R}$ telles que

- I. $\forall a, b \in X_1, x, y \in X_2, (a, x) \succsim (b, y) \iff u(a, x) \leq u(b, y)$,
- II. $\forall a, b \in X_1, a \succsim_1 b \iff u_1(a) \leq u_1(b)$,
- III. $\forall x, y \in X_2, x \succsim_2 y \iff u_2(x) \leq u_2(y)$,
- IV. $\forall a \in X_1, x \in X_2, u(a, x) = u_1(a) + u_2(x)$.

□

Notons que l'on retrouve aussi une discussion sur la fonction d'utilité d'alternatives à attributs multiples dans [Lei76]. Cet ouvrage présente entre autres des conditions d'optimalité suffisantes pour une telle fonction.

B.2 Notion de rationalité

L'existence d'une fonction d'utilité permettant d'exprimer numériquement la notion de préférence n'est d'intérêt que si l'on peut supposer que les agents (ou joueurs) économiques prennent des décisions rationnelles.

B.2.1 Description mathématique

La rationalité est facile à décrire de façon mathématique [OR94]. Soit A un ensemble d'actions parmi lesquelles un décideur doit choisir, C un ensemble de conséquences possibles à ces actions, $g : A \mapsto C$ une fonction indiquant la conséquence des actions, et une relation de préférence simple \succsim (telle que décrite ci-haut) sur l'ensemble C , que l'on suppose complète, transitive et réflexive (il s'agit en fait du cas où la préférence stricte \prec est une relation d'ordre faible). Aussi, on suppose l'existence d'une fonction d'utilité $u : C \mapsto \mathbb{R}$ (on a, $\forall x, y \in C$, $x \succsim y \iff u(x) \leq u(y)$). On dit alors qu'un décideur est **rationnel** s'il choisit l'action admissible $a^* \in B$ (parmi un ensemble d'actions admissibles $B \subseteq A$) telle que $g(a) \succsim g(a^*)$, $\forall a \in B$. Nous avons donc

$$a^* \in \arg \max_{a \in B} \{u(g(a))\}.$$

B.2.2 Justification

La description mathématique précédente, aussi directe soit-elle, ne répond pas à la question fondamentale, à savoir si on peut supposer que les agents économiques agissent de façon rationnelle.

Romp [Rom97] tente de répondre à cette question à l'aide de trois arguments.

1. L'être humain est rationnel

Cet argument peut être valide dans certaines situations simples, mais placé devant une décision complexe faisant intervenir plusieurs facteurs, un individu, soit par manque d'information, soit à cause de facultés d'analyse déficientes (ou une combinaison des deux), adoptera vraisemblablement des règles simplifiant le raisonnement et conduisant à une décision sous-optimale.

2. La sélection naturelle favorise la rationalité

L'activité économique peut être comparée à un processus de sélection naturelle dans lequel les entités irrationnelles prenant continuellement des décisions sous-optimales seront éliminées. Cet argument tient dans un contexte de libre compétition, puisque les entreprises irrationnelles vont tôt ou tard disparaître du marché (on n'a qu'à penser à la bulle Internet des années 98-99), mais ne s'applique pas à d'autres contextes. Par exemple, il ne semble pas y avoir de processus darwinien à grande échelle éliminant systématiquement les consommateurs irrationnels d'une chaîne de super-marchés.

3. Toute théorie repose sur des hypothèses

Comme mentionné au cours de la discussion sur la notion d'utilité, toute théorie est par définition irréaliste et doit s'appuyer sur des hypothèses simplificatrices. Afin d'appliquer des modèles économiques à des phénomènes concrets, nous devons inmanquablement faire certaines hypothèses sur la rationalité des acteurs économiques. Ces hypothèses, même si elles sont à première vue simplistes et naïves, sont d'autant plus justifiées si les résultats des modèles se basant sur elles concordent avec les observations empiriques et permettent d'établir des prédictions.

Annexe C

Librairie PyPlex

C.1 Motivation

La recherche opérationnelle est une discipline pratique utilisée pour résoudre des problèmes concrets. Que ce soit en transport, en télécommunications, en finance, en santé ou dans quelque autre champ d'application, le chercheur opérationnel associe des objets mathématiques aux éléments du problème auquel il s'attarde et tente d'en exprimer la solution en décrivant les interactions entre ces objets. Une fois ce processus de modélisation terminé, il s'attaque à la résolution du problème, soit en concevant un nouvel algorithme ou en adaptant un algorithme existant.

Ces deux aspects, modélisation et conception d'algorithme, représentent la plus grande partie du travail d'un chercheur opérationnel. Cependant, s'il désire appliquer son modèle et son algorithme à un problème pratique, le chercheur doit typiquement procéder à une implantation informatique dont le niveau de complexité reflète celui du problème auquel il s'intéresse. Il faut souvent manipuler de larges quantités de données brutes ou encore interagir avec des bibliothèques numériques certes puissantes mais d'interface peu conviviale et demandant un certain degré de compétences informatiques.

Nous croyons qu'il est possible de simplifier le processus d'implantation informatique d'un modèle et d'un algorithme de recherche opérationnelle à l'aide d'outils logiciels appropriés. Un tel outil

devrait répondre aux critères suivant :

- établir un lien conceptuel entre les aspects théoriques (modèle et algorithmes) et les aspects pratiques (données, solveurs, etc.) de la résolution d'un problème,
- faire abstraction des lourdeurs techniques reliées à une implantation informatique (initialisation de l'environnement, allocation et gestion de la mémoire, etc.),
- permettre de décrire le modèle avec une syntaxe se rapprochant le plus possible de la notation mathématique traditionnelle,
- posséder un caractère modulaire et être facilement extensible.
- n'imposer qu'un coût supplémentaire négligeable en termes de performance et de temps d'exécution.

Il existe déjà des langages de modélisation mathématique, par exemple GAMS ou AMPL, mais ceux-ci ne sont pas de véritables langages de programmation et n'offrent pas d'interface flexible permettant d'interagir avec d'autres outils logiciels, par exemple un serveur de données ou une interface-usager graphique. C'est dans cet état d'esprit que nous avons développé la librairie PyPlex. Notre but était de permettre l'implantation facile et rapide des modèles présentés dans cette thèse et de les appliquer sur des données réelles. Nous devons donc établir un lien logiciel entre d'une part le solveur CPLEX et les données stockées dans un serveur MySQL et d'autre part les modèles et algorithmes utilisés pour résoudre le problème. PyPlex devait servir de «mortier» logiciel entre ces différents blocs composant le problème.

C.2 Présentation de PyPlex

PyPlex, comme son nom le suggère, est un module implanté en langage Python. Python est un langage semi-interprété¹, orienté-objet avec types dynamiques². Ce langage se caractérise

¹Python est «semi-interprété» au même titre que Java. Dans un premier temps, l'interpréteur procède à l'analyse lexicographique et au partage d'un fichier-source et produit un fichier en code-octet (*byte-code*) contenant des instructions binaires (indépendantes de la plate-forme d'exécution). Par la suite, le fichier en code-octet est lu et exécuté par l'interpréteur. Si aucune modification n'est apportée au fichier-source entre deux exécutions, l'interpréteur peut passer directement à la deuxième étape, ce qui diminue le temps d'exécution.

²Certains langages, par exemple C, C++, Java ou Ada utilisent des types statiques. Dans ces langages, le programmeur doit déclarer explicitement le type de chaque variable dans le code source. En principe, une variable ne peut contenir que des données du type déclaré. Cette règle est cependant respectée avec plus ou moins de rigidité selon les langages (Ada est sans doute le plus strict et C le plus permissif, ce qui n'est pas sans causer certains problèmes aux programmeurs de C peu expérimentés). D'autre langages interprétés, comme

aussi par sa syntaxe simple, légère et élégante se rapprochant du pseudo-code généralement utilisé dans les ouvrages d'algorithmique. La responsabilité de la gestion de la mémoire revient à l'interpréteur, qui possède un mécanisme de *garbage collection* lui permettant d'identifier les objets qui ne sont plus utilisés et de libérer la mémoire qu'ils occupent. L'interpréteur (i.e. l'exécutable du nom de python) comporte deux modes. Dans le premier mode, l'interpréteur lit le fichier-source qu'on lui spécifie et en exécute les commandes. Par exemple, la ligne de commande

```
cotej@hub2//home/cotej/temp[7:22pm]python test.py
```

indique à l'interpréteur d'exécuter les commandes contenues dans le fichier `test.py` (par convention, les fichiers-source en langage Python prennent l'extension `.py`). Dans le second mode, appelé interactif, le programmeur peut entrer manuellement des commandes. Celles-ci sont immédiatement exécutés par l'interpréteur, qui en affiche les résultats à la console. Une telle session interactive est initiée en appelant l'interpréteur sans argument à partir de la ligne de commande :

```
cotej@hub2//home/cotej/temp[7:27pm]python
Python 2.2.2 (#1, Nov 21 2002, 14:46:32)
[GCC 3.0.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

L'interpréteur de Python est écrit en langage C. Il est possible d'écrire des extensions à l'interpréteur en utilisant ce langage. Ces extensions prennent la forme de bibliothèques dynamiques (`.dll` sous Windows, `.so` sous Unix) qui sont chargées par l'interpréteur lorsque leur fonctionnalité est requise. Dans notre cas, le but était d'interagir avec le solveur CPLEX à l'aide du langage Python. Nous avons donc entrepris d'écrire une extension qui serve de lien entre l'interpréteur et la bibliothèque CPLEX. L'avantage principal d'une telle approche est qu'elle permet d'exploiter toute la fonctionnalité de CPLEX à l'aide d'un langage de haut-niveau facile à programmer qui fait abstraction des aspects les plus lourds de l'utilisation de CPLEX (gestion de la mémoire et des pointeurs, des pièges dans lesquels même les plus expérimentés des programmeurs tombent souvent).

Python, utilisent des types dynamiques. Leurs variables ne sont pas préalablement déclarées et l'interpréteur détermine au moment de l'exécution quel type de donnée est contenu dans une variable. Ceci facilite la tâche du programmeur, au prix d'une légère augmentation du temps d'exécution.

Le fait que nous ayons utilisé un langage (semi-)interprété pour implanter cette librairie se reflète sur les performances. Afin d'illustrer cet impact, nous nous référons à Bagley [Bag], qui a mesuré les performances de différents langages dans certains contextes précis. Nous reprenons ici quelques résultats effectués sur des tâches essentiellement numériques. Les temps d'exécution contenus dans la table C.1 sont mesurés en secondes CPU. Ces résultats suggèrent que le même

Test	C (gcc)	C++ (g++)	Java	Python
Accès à un tableau	0.11	0.12	0.82	16.93
Nombres de Fibonacci	0.24	0.26	0.86	19.68
Multiplication de matrices	0.07	0.10	0.89	27.80
Crible d'Érasthostène	0.23	0.42	1.30	47.81

TAB. C.1 – Performances comparées de certains langages.

algorithme numérique, implanté en Python, sera entre 80 et 400 fois plus lent que s'il était implanté en un langage compilé et fortement optimisé comme le C ou le C++. Tout porte à croire que notre implantation soit affligée du même désavantage. C'est pour cette raison que nous avons jusqu'à maintenant utilisé PyPlex principalement pour formuler des modèles mathématiques à partir de données contenues dans un serveur SQL et ensuite soumettre ce modèle mathématique au solveur CPLEX. Dans ce processus, l'interpréteur Python n'effectue qu'une petite partie du travail qui consiste principalement à formater des données et gérer les interactions entre les différents modules logiciels. Le gros du travail revient d'une part au serveur de données, qui doit récupérer les informations requises, et d'autre part au solveur, qui doit résoudre le modèle à l'aide d'un algorithme approprié. Par conséquent, le prix à payer (en termes de temps d'exécution) pour l'utilisation d'une interface logicielle de haut-niveau telle PyPlex est relativement faible dans ce contexte. Toutefois, rien ne nous empêche d'utiliser PyPlex et le langage Python pour effectuer une partie du travail algorithmique à la place du solveur. C'est ce que nous avons fait pour la première implantation de notre algorithme spécialisé d'énumération implicite où nous effectuons nous-mêmes les opérations de branchement, de sélection des variables et de gestion de l'arbre, des tâches auparavant entièrement laissées au solveur CPLEX. Comme l'interpréteur effectue ici un plus grande partie du travail numérique, les performances globales s'en ressentent. C'est pourquoi nous estimons qu'un tel outil logiciel peut être utilisé afin de construire rapidement un prototype et de valider un nouvel algorithme, mais devra être remplacé par une implantation de plus bas niveau dans un contexte pratique où le temps d'exécution est souvent crucial.

C.3 Implantation de PyPlex

La librairie PyPlex prend la forme d'un unique fichier-source appelé `PyPlex.c`. En plus de son propre fichier entête, `PyPlex.c` inclut les entêtes `python.h` et `cplex.h`, de façon à exploiter la fonctionnalité de ces deux entités logicielles et de servir de lien entre elles. Le fichier-source est compilé en une librairie dynamique du nom de `PyPlex.dll` sous Windows et `PyPlex.so` sous Unix¹.

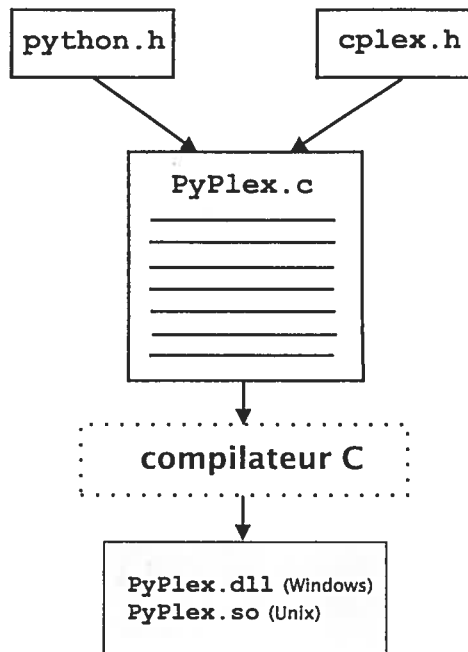


FIG. C.1 – Organisation des fichiers de la librairie PyPlex.

Afin de spécifier à l'interpréteur de Python que celui-ci doit charger la librairie PyPlex, nous utilisons la commande `import`. Cette commande se retrouve typiquement au début d'un fichier-source en langage Python. Lorsque l'interpréteur est utilisé en mode interactif, nous avons :

```
Python 2.2.2 (#1, Nov 21 2002, 14:46:32)
```

```
[GCC 3.0.3] on linux2
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

¹En mentionnant le système «Unix», nous ne prétendons pas que PyPlex soit compatible avec toutes les implantations de ce standard. Nous avons compilé et testé PyPlex sous deux variantes de Unix, soit Linux et Sun Solaris.

```
>>> import PyPlex
```

Étant donné cette commande, l'interpréteur cherche dans son environnement la librairie dynamique `PyPlex.dll` (ou `.so`). Une fois la librairie trouvée, python l'initialise et inscrit en mémoire les fonctions que celle-ci offre. Cette étape est effectuée en appelant la fonction `initPyPlex`, laquelle est définie dans `PyPlex.c` :

```
void
#ifdef WIN32
__declspec(dllexport)
#endif
initPyPlex(void)
{
    /* Immédiatement après "import PyPlex" */
    PyObject *m, *d;
    PyPlexEnvPtrType.ob_type = &PyType_Type;
    PyPlexLPPtrType.ob_type = &PyType_Type;
    /* Initialisation des symboles de CPLEX */
    initPyPlexSymbolTable();
    /* Création du module PyPlex */
    m = Py_InitModule("PyPlex", PyPlex_methods);
    d = PyModule_GetDict(m);
    ErrorObject = Py_BuildValue("s", "PyPlex.error");
    PyDict_SetItemString(d, "error", ErrorObject);
    if (PyErr_Occurred())
        Py_FatalError("Cannot initialize module PyPlex");
}
```

Une fois que la librairie `PyPlex` a été chargée, nous pouvons en exploiter la fonctionnalité. La commande de Python `dir` nous permet de consulter la liste des fonctions offertes par `PyPlex`. En lançant cette commande en mode interactif, nous obtenons :

```
...
>>> dir(PyPlex)
```

```

['CPXaddrows', 'CPXbaropt', 'CPXchgbd', 'CPXchgobjsen', 'CPXchgprobtype',
'CPXchgsense', 'CPXcloseCPLEX', 'CPXcopyctype', 'CPXcopylp', 'CPXcopyquad',
'CPXcreateprob', 'CPXdualopt', 'CPXfreeprob', 'CPXgetbase', 'CPXgetdblparam',
'CPXgeterrorstring', 'CPXgetintparam', 'CPXgetmipobjval', 'CPXgetmipslack',
'CPXgetmipx', 'CPXgetnumcols', 'CPXgetnumrows', 'CPXgetobjsen', 'CPXgetobjval',
'CPXgetpi', 'CPXgetprobtype', 'CPXgetslack', 'CPXgetstat', 'CPXgetstrparam',
'CPXgetx', 'CPXlpopt', 'CPXmipopt', 'CPXnewcols', 'CPXopenCPLEX', 'CPXprimopt',
'CPXqpwrite', 'CPXreadcopyprob', 'CPXsetdblparam', 'CPXsetintparam',
'CPXsetstrparam', 'CPXwriteprob', 'PyPlex_getSymbolValue', '__doc__', '__file__',
'__name__', 'error']
>>>

```

Plusieurs de ces fonctions (avec le préfixe CPX) sembleront familières aux lecteurs habitués à la programmation avec CPLEX. Nous avons en effet choisi d'assigner à ces fonctions le nom qu'elles possèdent dans l'interface de programmation avec CPLEX. Il ne s'agit pour l'instant que d'une partie de toutes les fonctions offertes par CPLEX, mais ce sont probablement les plus usitées (ce sont celles que nous avons du utiliser afin de résoudre nos modèles). Nous entendons compléter cette liste afin que la librairie PyPlex offre toute la fonctionnalité de CPLEX.

Illustrons maintenant le mécanisme qui nous permet d'interagir avec CPLEX à partir de l'interpréteur de Python. La première fonction qui est habituellement appelée lorsqu'on désire communiquer avec CPLEX est `CPXopenCPLEX`. Dans sa version en langage C, cette fonction, qui prend comme argument un pointeur à un entier indiquant le statut de l'opération, vérifie si une licence valide est disponible et, le cas échéant, retourne un pointeur vers un environnement CPLEX initialisé. En l'appelant à partir de l'interpréteur, nous obtenons¹ :

```

...
>>> from PyPlex import *
>>> status = 0
>>> env = CPXopenCPLEX(status)

```

¹Il existe deux façons d'importer une librairie en langage Python. La première, `import PyPlex`, charge les fonctions de PyPlex mais n'inscrit pas leur nom dans l'environnement global. Pour appeler une de ces fonctions, nous devons la «qualifier» du nom de la librairie, i.e. `PyPlex.CPXopenCPLEX()`. Cependant, nous pouvons donner à l'interpréteur la commande `from PyPlex import *`, ce qui signifie que, en plus d'importer les fonctions, l'interpréteur inscrit leur nom dans l'environnement global, de sorte qu'on puisse directement appeler `CPXopenCPLEX()` sans devoir la qualifier du nom de la librairie.


```
>>> print status
0
>>> print env
(CPLEX Environment pointer at 8171db0, 0)
>>>
```

Lorsqu'il reçoit un appel à la fonction `CPXopenCPLEX`, l'interpréteur consulte la table de symboles de la librairie afin de savoir à quelle fonction en langage C celle-ci correspond. Dans cet exemple, il s'agit de la fonction `PyPlex_CPXopenCPLEX` dont la définition se trouve dans le fichier `PyPlex.c` :

```
static PyObject* PyPlex_CPXopenCPLEX(PyObject *self, PyObject *args)
{
    int status;
    CPXENVptr CplexEnv;
    PyPlexEnvPtr* myEnv;
    if(!PyArg_ParseTuple(args, "i", &status))
        return NULL;
    if(!(myEnv = construct_PyPlexEnvPtr()))
        return NULL;
    CplexEnv = CPXopenCPLEX(&status);
    myEnv->p = CplexEnv;
    return Py_BuildValue("Oi", myEnv, status);
}
```

On remarque que cette dernière fonction contient un appel à la véritable fonction `CPXopenCPLEX` de CPLEX. Le pointeur retourné par `CPXopenCPLEX` est ensuite converti en objet Python (par la fonction `Py_BuildValue`) de sorte qu'il puisse être manipulé par l'interpréteur. Les autres fonctions de la librairie `PyPlex` ont une implantation analogue.

C.4 Le module MathProg

La librairie PyPlex permet d'interagir avec le solveur CPLEX en utilisant le langage Python. Son principal avantage elle qu'elle permet d'automatiser la gestion de la mémoire et l'initialisation de l'environnement, deux aspects qui causent souvent des problèmes aux programmeurs. Toutefois, comme elle utilise la même syntaxe de bas-niveau que l'interface originale, elle n'apporte aucun bénéfice du point de vue de l'abstraction et de la facilitation du processus d'implantation de modèles mathématiques et d'algorithmes pouvant les résoudre. Afin de combler cette lacune, nous avons développé le module¹ MathProg.

MathProg contient trois classes principales : MP (pour *mathematical program*), Var (pour *variable*) et Cons (pour *constraint*), les deux premières étant des classes abstraites ne pouvant pas avoir d'instance. Elles servent de bases à d'autres sous-classes, soit LP (pour *linear program*), MIP (pour *mixed integer program*) et QP (pour *quadratic program*) dans le cas de MP et BinVar (pour *binary variable*), IntVar (pour *integer variable*) ainsi que ContVar (pour *continuous variable*) dans le cas de Var. C'est à l'aide de ces classes que l'on peut formuler un programme mathématique dans le module MathProg. D'autres classes aussi contenues dans ce module facilitent le processus de formulation. Il s'agit de VarSet, ConsSet, Term, Expr et SolverEx (pour *solver exception*), cette dernière étant utilisée pour signaler une erreur. La figure C.2 résume la hiérarchie du module.

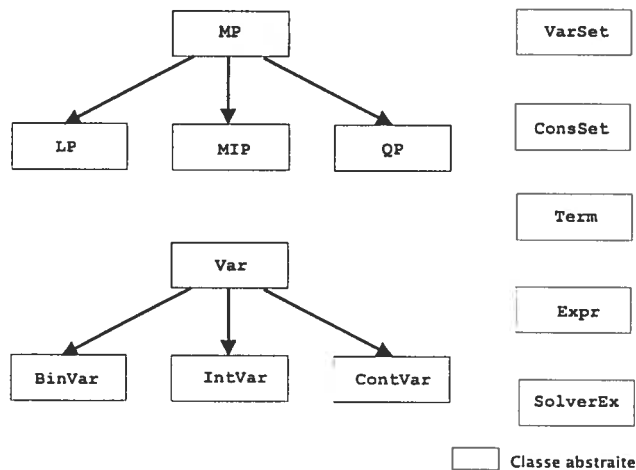


FIG. C.2 – Hiérarchie des classes du module MathProg.

¹En terminologie «pythonienne», un module est un fichier-source contenant l'implantation d'objets conceptuellement reliés et traités comme une seule unité logique.

Le prochain exemple illustre l'emploi du module. Supposons que nous voulons résoudre le problème suivant à l'aide de CPLEX ¹.

$$\begin{aligned}
 \max_{x_1, x_2, x_3} \quad & x_1 + 2x_2 + 3x_3 \\
 \text{s.l.c.} \quad & -x_1 + x_2 + x_3 \leq 20 \\
 & x_1 - 3x_2 + x_3 \leq 30 \\
 & x_1 \leq 40 \\
 & x_1, x_2, x_3 \geq 0.
 \end{aligned}$$

Le script² Python suivant, appelé `MP_test1.py`, résout ce problème à l'aide du module `MathProg` et de la librairie `PyPlex` :

```

1  from MathProg import *
2  print "MathProg test 1"
3  x1 = ContVar("x1", 0.0, 40.0)
4  x2 = ContVar("x2")
5  x3 = ContVar("x3")
6  e = SolverEx()
7  openSolver()
8  try:
9      prob = LP(MP.MAX, "Test1")
10     prob.addVars([x1,x2,x3])
11     prob.setObj(x1 + 2*x2 + 3*x3)
12     prob.addCons(Cons(-x1 + x2 + x3 , Cons.LEQ, 20.0, "c1"))
13     prob.addCons(Cons(x1 - 3*x2 + x3 , Cons.LEQ, 30.0, "c2"))
14     prob.solve()
15     prob.free()
16 except e:
17     print e
18 print "Optimal value is %f" % prob.objval
19 for v in prob.vars:

```

¹Ce problème est le premier exemple présenté dans le guide de l'utilisateur de CPLEX. Le code permettant de le résoudre se trouve dans le fichier `/examples/src/lpex1.c` de la distribution de CPLEX.

²Dans le contexte des langages de programmation interprétés, un programme complet est habituellement appelé un «script».

```

20     print "Variable %s has value %f" % (v.label, v.value)
21     print "Solver time is %.6f seconds" % prob.solvertime
22     print "Done!"
23     del prob
24     closeSolver()

```

La ligne 1 indique à l'interpréteur qu'il doit charger le module MathProg (qui charge à son tour la librairie PyPlex). Les lignes 3, 4 et 5 créent trois variables continues (de type ContVar). Le premier argument passé au constructeur de ContVar est le «nom» (label) de la variable, et les autres arguments sont respectivement les bornes inférieure et supérieure de la variable (qui sont par défaut 0 et $+\infty$). À la ligne 6, on crée un objet de type SolverEx qui sera utilisé en cas d'erreur, alors qu'à la ligne 7 la fonction openSolver initialise l'environnement CPLEX (cette fonction appelle entre autre la fonction CPXopenCPLEX dont nous avons discuté précédemment). Le problème comme tel est formulé aux lignes 8 à 13. Nous appelons ensuite la fonction solve (qui à son tour appelle la fonction de la librairie CPLEX permettant de résoudre le type de problème approprié, dans ce cas-ci un problème linéaire), et une fois le problème résolu, nous appelons la fonction free afin de libérer la licence requise lors de la résolution. Le reste du programme permet d'afficher les résultats à l'écran.

Lorsque nous lançons ce programme, nous obtenons :

```

cotej@hub2//home/cotej/windrive/work/phd/python/pyplex[7:38pm]python MP_test1.py
MathProg test 1
Optimal value is 202.500000
Variable x1 has value 40.000000
Variable x2 has value 17.500000
Variable x3 has value 42.500000
Solver time is 0.010000 seconds
Done!
cotej@hub2//home/cotej/windrive/work/phd/python/pyplex[7:38pm]

```

À titre de comparaison, le code suivant permet de formuler le même problème en utilisant directement les fonctions en langage C de la librairie CPLEX (cet extrait se trouve dans le fichier /examples/src/lpex1.c de la distribution de CPLEX).

```

static int populatebyrow (CPXENVptr env, CPXLPptr lp)

```

```

{
    int      status      = 0;
    double   obj[NUMCOLS];
    double   lb[NUMCOLS];
    double   ub[NUMCOLS];
    char     *colname[NUMCOLS];
    int      rmatbeg[NUMROWS];
    int      rmatind[NUMNZ];
    double   rmatval[NUMNZ];
    double   rhs[NUMROWS];
    char     sense[NUMROWS];
    char     *rowname[NUMROWS];

    CPXchgobjsen (env, lp, CPX_MAX); /* Problem is maximization */
    /* Now create the new columns. First, populate the arrays. */
    obj[0] = 1.0;      obj[1] = 2.0;      obj[2] = 3.0;
    lb[0] = 0.0;      lb[1] = 0.0;      lb[2] = 0.0;
    ub[0] = 40.0;     ub[1] = CPX_INFBOUND; ub[2] = CPX_INFBOUND;

    colname[0] = "x1"; colname[1] = "x2";      colname[2] = "x3";
    status = CPXnewcols (env, lp, NUMCOLS, obj, lb, ub, NULL, colname);
    if ( status ) goto TERMINATE;
    /* Now add the constraints. */
    rmatbeg[0] = 0;      rowname[0] = "c1";
    rmatind[0] = 0;      rmatind[1] = 1;      rmatind[2] = 2;      sense[0] = 'L';
    rmatval[0] = -1.0;  rmatval[1] = 1.0;  rmatval[2] = 1.0;  rhs[0] = 20.0;
    rmatbeg[1] = 3;      rowname[1] = "c2";
    rmatind[3] = 0;      rmatind[4] = 1;      rmatind[5] = 2;      sense[1] = 'L';
    rmatval[3] = 1.0;  rmatval[4] = -3.0; rmatval[5] = 1.0;  rhs[1] = 30.0;
    status = CPXaddrows (env, lp, 0, NUMROWS, NUMNZ, rhs, sense, rmatbeg,
                        rmatind, rmatval, NULL, rowname);
    if ( status ) goto TERMINATE;
TERMINATE:
    return (status);
}

```

Notons que ce code ne fait que formuler le problème; d'autres fonctions en langage C sont nécessaires afin d'initialiser l'environnement avant la formulation et de libérer la mémoire et les licences une fois le problème résolu.

Revenons au script `MP_test1.py`. Aux lignes 11, 12 et 13, la fonction objectif et les contraintes sont définies en tant que combinaison linéaires des objets `x1`, `x2` et `x3`, qui sont de type `ContVar`.

De telles combinaisons linéaires d'objets ne sont pas admises dans la syntaxe de base du langage Python. C'est ici que les classes `Expr` et `Term` entrent en jeu. En redéfinissant les opérateurs appropriés, il est possible d'implanter la grammaire

$$\begin{aligned} \langle \text{Expr} \rangle &\rightarrow \langle \text{Expr} \rangle + \langle \text{Term} \rangle \mid \langle \text{Term} \rangle \\ \langle \text{Term} \rangle &\rightarrow \langle \text{Var} \rangle \mid + \langle \text{Var} \rangle \mid - \langle \text{Var} \rangle \mid \langle \text{facteur} \rangle * \langle \text{Var} \rangle \\ \langle \text{facteur} \rangle &\rightarrow a \mid + a \mid - a \end{aligned}$$

où $a \in \mathbb{R}$. Selon cette grammaire, un objet de type `Term` est obtenu en multipliant un objet de type `Var` (ou une de ses classes dérivées) par un facteur réel, alors qu'un objet de type `Expr` est une somme d'un ou plusieurs objets de type `Term`. L'argument de la fonction `setObj` (ligne 11) et le premier argument du constructeur de la classe `Cons` (lignes 12 et 13) sont donc des objets de type `Expr`.

La grammaire est implantée dans la définition des classes concernées (dans le fichier `MathProg.py`). Voici, à titre d'exemple, la définition de la classe `Term`

```
class Term:

    def __init__(self, varindex, varcoef):
        self.varindex = varindex
        self.varcoef = varcoef

    def __add__(self, other):
        if isVar(other): # Expr = Term + Var, make a new Term with 1 as coef
            newTerm = Term(other.index, 1.0)
            return self.__add__(newTerm)
        else: # Expr = Term + Term,
            newExpr = Expr()
            newExpr.indexes.extend([self.varindex, other.varindex])
            newExpr.coefs.extend([self.varcoef, other.varcoef])
            newExpr.nb = 2
            return newExpr

    def __sub__(self, other): # Expr = Term - Term
        if isVar(other):
            newTerm = Term(other.index, -1.0)
            return self.__add__(newTerm)
        else:
            other.varcoef = -other.varcoef
```

```
return self.__add__(other)
```

Le problème que nous cherchons à résoudre à l'aide du script `MP_test1.py` ne contient que trois variables et deux contraintes. Nous pouvons donc déclarer explicitement chacune de ces variables et contraintes, comme nous le faisons dans ce script. Cependant, pour des problèmes contenant des dizaines (voire des centaines) de milliers de variables et de contraintes, une telle approche est à toutes fins pratiques impossible. C'est dans ce contexte que les classes `VarSet` et `ConsSet` trouvent leur utilité. Un objet de type `VarSet` est une collection de variables que l'on peut associer aux objets réels (i.e. extraits des données du problème) auxquels elles correspondent. Nous illustrons cet usage avec un extrait de l'implantation du modèle (BAFOv-MIP) sur des données réelles :

```
import time
import Database
from RealDataObject import *
from MathProg import *
import sys
...
x = VarSet()
y = VarSet()
T = VarSet()
eta = VarSet()
u = VarSet()
v = VarSet()
...
def createVars():
...
    # Create a flow variable for each flight/product/group combination
    for f in flights:
        for g in groups:
            if f.isLeader():
                for p in products:
                    if p.isLeader() and p.market is f.market:
                        var = ContVar("x" + "_FL" + str(f.id) + \
                                      "_GR" + str(g.id) + "_PR" + str(p.id), 0.0)
                        x[f,g,p] = var
                        prob.addVar(var)
            else:
                for p in products:
```

```

        if p.market is f.market and p.carriercode == f.legs[0].carriercode:
            var = ContVar("y" + "_FL" + str(f.id) + \
                "_GR" + str(g.id) + "_PR" + str(p.id), 0.0)
            y[f,g,p] = var
            prob.addVar(var)
        ...
    ...
def setObj():
    obj = Expr()
    # Objective: eta*b - c*x - d*y
    ...
    for f in flights:
        for g in groups:
            if f.isLeader():
                for p in products:
                    if p.isLeader() and p.market is f.market:
                        obj += -g.disutility(f,p) * x[f,g,p]
            else:
                for p in products:
                    if p.market is f.market and p.carriercode == f.legs[0].carriercode:
                        if prodflag:
                            obj += -(p.price + g.disutility(f,p)) * y[f,g,p]
                        else:
                            obj += -(f.price + g.disutility(f,p)) * y[f,g,p]
        ...
    ...
def addCons():
    # Type 1 constraint: G*x <= u (leg capacity)
    for l in legs:
        c = Expr()
        legused = False
        for f in flights:
            if f.isLeader() and f in l.flights:
                for g in groups:
                    for p in products:
                        if p.isLeader() and p.market is f.market:
                            c += x[f,g,p]
                            legused = True
        if legused:
            if legfrac:
                capfrac = l.marketsfrac[markets[0]]
            else:
                capfrac = 1.0

```



```

prob.addCons(Cons(c , Cons.LEQ, l.getTotalCap() * capfrac, "C1" + "_LG" + str(l.id)))
...

```

La classe `ConsSet` est utilisée de façon analogue.

La résolution de problèmes pratiques implique habituellement la manipulation de grandes quantités de données. Dans notre cas, les données ont été organisées selon un modèle relationnel et stockées dans un serveur de type MySQL. Le langage Python comporte une interface avec ce type de serveur, de sorte que nous pouvons facilement accéder aux données. La figure C.3 illustre les interactions entre les différents éléments du processus de résolution.

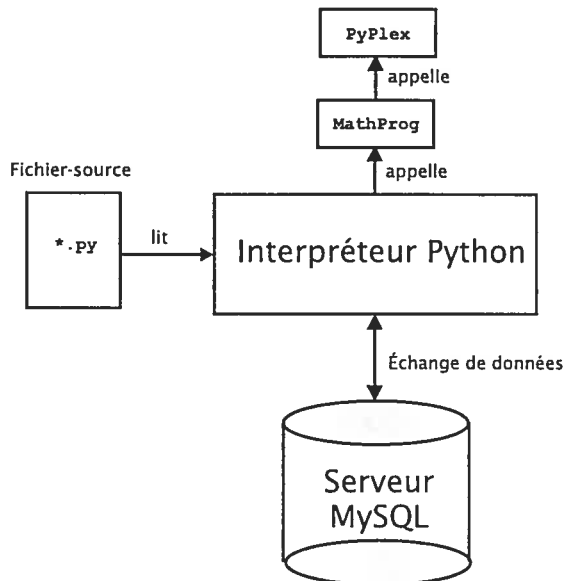


FIG. C.3 – Interactions logicielles.

C.5 Bénéfices

Nous ne croyons pas qu'un outil logiciel comme la librairie PyPlex et le module MathProg soient utiles dans un contexte industriel, du moins dans leur état actuel. D'abord, en se basant sur un langage interprété et dynamique comme Python, nous devons payer un prix élevé en termes de performance et de quantité de mémoire utilisée. De plus, la gestion et mise à jour régulière

de l'interpréteur et de l'environnement d'exécution nécessite plus d'efforts que dans le cas d'un simple programme en format binaire. Il faut entre autres s'assurer de la compatibilité entre les différentes libraires utilisées et remettre celles-ci à jour à chaque évolution. Ces tâches ne reviennent pas au programmeur mais doivent être effectués par un membre de l'organisation (typiquement un administrateur de système).

Cependant, nous entrevoyons plusieurs bénéfices à l'utilisation de tels outils dans un contexte de recherche académique ou encore pour développer le premier prototype d'une future application industrielle afin d'en valider les concepts. Ces bénéfices sont :

- accélération du temps de développement grâce à l'utilisation d'un langage de plus haut niveau et élimination de l'étape de compilation,
- abstraction des aspects les plus lourds et fastidieux du développement de logiciels (gestion de l'environnement, de la mémoire, des pointeurs, des interfaces, etc.) permettant de se concentrer sur les aspects centraux, soit les modèles et algorithmes,
- facilitation du mécanisme de collection des données par l'utilisation d'interfaces déjà disponibles,
- obtention plus rapide de résultats numériques permettant de valider ou rejeter une approche.

Annexe D

Glossaire

Ce glossaire contient la définition précise de plusieurs termes importants utilisés dans ce document. Il s'agit pour la plupart de traductions ou d'adaptations d'expressions américaines propres au transport aérien. Il n'existe pas, à notre connaissance, de terminologie française officielle dans ce domaine. Les termes que nous proposons ci-dessous se veulent ainsi une proposition en ce sens. En particulier, dans les cas où il nous fut impossible de trouver un équivalent français adéquat, nous suggérons un néologisme.

Achat ascensionnel [*cf. achat descensionnel*] Phénomène se produisant lorsqu'un passager désirant initialement acheter un billet dans un classe tarifaire de bas niveau accepte d'acheter un billet plus cher en apprenant qu'aucun billet de bas niveau n'est disponible. Le terme américain équivalent est *buy-up*.

Achat descensionnel [*cf. achat ascensionnel*] Phénomène se produisant lorsqu'un passager ayant initialement la volonté (et la capacité) d'acheter un billet dans une classe de haut niveau achète un billet moins cher en apprenant que des sièges sont disponibles dans une classe de plus bas niveau. Le terme américain équivalent est *buy-down*.

Aphane *Adj.* [*néolog., du grec class. φαίνω, moderne φαίνομαι ; sens trans. «faire apparaître, mettre en lumière», intrans. «apparaître, voir le jour».*] Signifie invisible ou incertain. Se dit des passagers ayant réservé ou acheté un billet mais ne se présentant pas lors de l'embarquement. Le terme américain équivalent est *no-show*.

Base tarifaire Ensemble d'avantages et de restrictions associés à un billet et auxquels correspond un prix unique. Une classe tarifaire est divisée en un certain nombre de bases tarifaires. C'est une base tarifaire que le client achète lorsqu'il se procure un billet. Les bases tarifaires sont identifiées par des codes d'au plus huit caractères qui habituellement indiquent la classe à laquelle la base appartient ainsi que les restrictions principales qui lui sont associés. Par exemple, le code Y21NRSU peut désigner une base tarifaire en classe Y devant être achetée au moins 21 jours avant le départ (21), qui est non remboursable (NR, *non-refundable*) et pour laquelle le passager doit passer au moins un dimanche à destination (SU, *Sunday*) entre l'aller et le retour. Remarquons qu'il n'existe pas de standard quant à la composition de ces codes, sinon qu'ils doivent posséder au plus huit caractères et que la première lettre désigne la classe. Le terme américain équivalent est *fare basis*.

Classe tarifaire Sous-ensemble de sièges d'un avion auxquels sont associées des conditions similaires. Une classe tarifaire peut correspondre à une cabine (e.g. la première classe correspond habituellement à la cabine la plus luxueuse d'un appareil), mais ce n'est pas toujours le cas. Les classes tarifaires sont habituellement identifiés par une lettre, e.g. la classe Y. Notons qu'il n'y a pas de standard quant à l'utilisation de ces codes d'une lettre désignant les classes tarifaires, chaque transporteur pouvant utiliser les codes qu'il désire. Le terme américain équivalent est *fare class*.

Déversement Phénomène se produisant lorsque des requêtes de réservation ou d'achat dans une certaine classe tarifaire sont rejetées parce que la classe en question a atteint sa limite de réservation. Le déversement est souvent le symptôme d'une politique de contrôle d'inventaire déficiente. Le terme américain équivalent est *spill*.

Emboîtement Propriété d'un schéma d'allocation de sièges aux classes tarifaires d'un vol qui prévient le rejet d'une réservation en haute-classe lorsque des sièges sont toujours disponibles dans une classe de niveau inférieur. En d'autres mots, un schéma emboîté permet à une classe payante ayant atteint sa limite de réservation d'aller « voler » un siège à une classe moins payante ayant toujours des sièges disponibles. Le terme américain équivalent est *nesting*.

Emphane *Adj.* [*néolog., cf. aphone*] Signifie visible, se manifestant physiquement. Se dit des passagers ayant réservé ou acheté un billet et présents lors de l'embarquement. Le terme américain équivalent est *go-show*.

Liaison partagée Segment de vol offert conjointement par deux compagnies aériennes qui s'en-

tendent au préalable sur le nombre de sièges vendus par chaque compagnie et sur la répartition des revenus. Les sièges sont ensuite vendus de façon indépendante par chacune des compagnies, souvent avec des numéros de vol différents. Le segment lui-même est assuré par l'une des compagnies participant à l'entente. Le terme américain équivalent est *code-share*.

Marché Paire origine-destination composée de deux aéroports, ou plus généralement, de deux villes.

Moyeu [*cf. radial, rayon*] Aéroport central par lequel transitent tous les vols dans un réseau radial. Le terme américain équivalent est *hub*.

Radial *Adj* [*cf. moyeu, rayon*] Se dit d'un réseau tentaculaire dans lequel les passagers empruntent un rayon incident à un moyeu central, d'où ils empruntent un autre rayon pour se rendre à leur destination finale. Le terme américain équivalent est *hub-and-spoke*.

Rayon [*cf. moyeu, radial*] Vol à l'origine ou à la destination du moyeu dans un réseau radial. Le terme américain équivalent est *spoke*.

Segment [*cf. vol*] Nous désignons par segment (de vol) le trajet simple effectué par un avion entre le décollage et l'atterrissage. Le terme américain équivalent est *leg*.

Surclassement Phénomène se produisant lorsqu'un passager, pour diverses raisons, se voit offrir un siège dans une classe tarifaire de niveau supérieur à celle dans laquelle il a acheté son billet.

Surréservation La surréservation en transport aérien est la pratique qui consiste à accepter plus de réservations que la capacité le permet, en sachant qu'une certaine partie de ces réservations seront annulées ou encore que les passagers ne se présenteront pas, diminuant ainsi la possibilité que des sièges soient vides au décollage. Le terme américain équivalent est *overbooking*.

Vol Nous désignons par vol le trajet complet effectué par un passager entre son origine et sa destination finale. Un vol se compose d'un ou de plusieurs segments de vol, i.e. de vols physiques effectués par un avion, possiblement entrecoupés d'escapes.

Bibliographie

- [ABMV96] J. Alstrup, S. Boas, O. B. G. Madsen et R. V. V. Vidal. « Booking Policy for Flights With Two Types of Passengers ». *European Journal of Operational Research*, volume 27, pages 274–288, 1996.
- [And89] Sven-Eric Anderson. « Operational Planning in Airline Business – Can Science Improve Efficiency? Experiences from SAS ». *European Journal of Operational Research*, volume 43, numéro 1, pages 3–12, 1989.
- [Ano02] Anonyme. « Saturday night fever ». *The Economist*, volume 363, numéro 8268, page 64, 2002.
- [AO01] Ravindra K. Ahuja et James B. Orlin. « Inverse Optimization ». *Operations Research*, volume 49, numéro 5, pages 771–783, 2001.
- [Bag] Doug Bagley. « The Great Language Shootout ». Article sur le web à l'adresse <http://www.bagley.org/~doug/shootout/>.
- [BB57] Martin J. Beckman et F. Bobkoski. « Airline Demand : An Analysis of some Frequency Distributions ». *Naval Research Logistics Quarterly*, volume 5, pages 43–51, 1957.
- [BBO03] Cynthia Barnhart, Peter P. Belobaba et Amedeo R. Odoni. « Applications of Operations Research in the Air Transport Industry ». *Transportation Science*, volume 37, numéro 4, pages 368–391, 2003.
- [BCS97] Steven Berry, Michael Carnall et Pablo T. Spiller. « Airline Hubs : Costs, Markups and the Implication of Customer Heterogeneity ». 1997.
- [Bel87] Peter P. Belobaba. « Airline Yield Management : An Overview of Seat Inventory Control ». *Transportation Science*, volume 21, numéro 2, pages 63–73, 1987.

- [Bel89] Peter P. Belobaba. « Application of a Probabilistic Decision Model to Airline Seat Inventory Control ». *Operations Research*, volume 37, numéro 2, pages 183–197, 1989.
- [Bel98] Peter P. Belobaba. « The Evolution of Airline Yield Management : Fare Class to Origin-Destination Inventory Control ». In Butler et Keller [BK98], pages 285–302.
- [BH93] Matthew E. Berge et Craig A. Hopperstand. « Demand Driven Dispatch : A Method for Dynamic Aircraft Capacity Assignment, Models and Algorithms ». *Operations Research*, volume 41, numéro 1, pages 155–168, 1993.
- [BK98] Gail F. Butler et Martin R. Keller, éditeurs. *Handbook of Airline Marketing*. Aviation Week Group, McGraw-Hill, 1998.
- [BLMS00] L. Brotcorne, M. Labbé, P. Marcotte et G. Savard. « A bilevel model and solution algorithm for a freight tariff setting problem ». *Transportation Science*, volume 34, pages 317–324, 2000.
- [BLMS01] L. Brotcorne, M. Labbé, P. Marcotte et G. Savard. « A bilevel model for toll optimization on a multicommodity transportation network ». *Transportation Science*, volume 35, pages 1–14, 2001.
- [BM73] J. Bracken et J. McGill. « Mathematical programs with optimization problems in the constraints ». *Operations Research*, volume 21, pages 37–44, 1973.
- [BM93] S. L. Brumelle et J. I. McGill. « Airline Seat Allocation with Multiple Fare Classes ». *Operations Research*, volume 41, numéro 1, pages 127–137, 1993.
- [BMO⁺90] S. L. Brumelle, J. I. McGill, T. H. Oum, K. Sawaki et M. W. Tretheway. « Allocation of Airline Seats between Stochastically Dependand Demands ». *Transportation Science*, volume 24, numéro 3, pages 183–192, 1990.
- [Bot96] T.C. Botimer. « Efficiency considerations in airline pricing and yield management ». *Transportation Research A*, volume 34, numéro 4, pages 317–337, 1996.
- [BP03] Dimitri Bertsimas et Ioana Popescu. « Revenue Management in a Dynamic Network Environment ». *Transportation Science*, volume 37, numéro 3, pages 257–277, 2003.
- [BSS93] Mokhtar S. Bazaraa, Hanif D. Sherali et C.M. Shetty. *Nonlinear programming : Theory and Algorithms (2nd Edition)*. John Wiley and Sons, 1993.
- [BT92] D. Burton et Ph. L. Toint. « On an Instance of the Inverse Shortest Paths Problems ». *Mathematical Programming*, volume 53, pages 45–61, 1992.

- [BT94] D. Burton et Ph. L. Toint. « On the Use of an Inverse Shortest Paths Algorithm for Recovering Linearly Correlated Costs ». *Mathematical Programming*, volume 63, pages 1–21, 1994.
- [But96] Kenneth J. Button. « Liberalising European Aviation : Is There An Empty Core Problem? ». *Journal of Transport Economics and Policy*, volume 30, numéro 3, pages 275–291, 1996.
- [But02] Kenneth J. Button. « Empty Cores in Airlines Markets ». Fifth Hamburg Aviation Conference, Hambourg (Allemagne), 2002.
- [CILP99] A. Ciancimino, G. Inzerillo, S. Lucidi et L. Palagi. « A Mathematical Programming Approach for the Solution of the Railway Yield Management Problem ». *Transportation Science*, volume 33, numéro 2, pages 168–181, 1999.
- [CMS03] Jean-Philippe Côté, Patrice Marcotte et Gilles Savard. « A Bilevel Modelling Approach to Pricing and Fare Optimisation in the Airline Industry ». *Journal of Revenue and Pricing Management*, volume 2, numéro 1, pages 23–36, 2003.
- [CN77] W. Candler et R. Norton. « Multilevel programming ». Rapport technique 20, World Bank Development Research Center, Washington D.C., 1977.
- [CP68] Robert Charlebois et Claude Péloquin. « Lindberg », 1968.
- [Cro95] Robert G. Cross. « An Introduction to Revenue Management ». In Jenkins [Jen95], pages 443–458.
- [Cro98] Robert G. Cross. « Trends in Airline Revenue Management ». In Butler et Keller [BK98], pages 319–331.
- [Côt01] Jean-Philippe Côté. « Un modèle bi-niveau pour la gestion du revenu en transport aérien ». Mémoire de maîtrise, Université de Montréal, juillet 2001.
- [Cur90] R.E. Curry. « Optimal Airline Seat Allocation With Fare Classes Nested by Origins and Destinations ». *Transportation Science*, volume 24, numéro 3, pages 193–204, 1990.
- [dBFP99] Sanne V. de Boer, Richard Freling et Nanda Piersma. « Stochastic Programming for Multiple-Leg Network Revenue Management ». Rapport technique EI-9935/A, ORTEC Consultants, Gouda, Pays-Bas, 1999.
- [dBFP02] Sanne V. de Boer, Richard Freling et Nanda Piersma. « Mathematical programming for network revenue management revisited ». *European Journal of Operational Research*, numéro 137, pages 72–92, 2002.

- [Fis70] Peter C. Fishburn. *Utility Theory and Decision Making*. John Wiley and Sons, 1970.
- [FP01] Christodoulos A. Floudas et Panos M. Pardalos, éditeurs. *Encyclopedia of Optimization*. Kluwer Academic Publishers, 2001.
- [FX01] Youyi Feng et Baichun Xiao. « A Dynamic Seat Inventory Control Model and its Optimal Policy ». *Operations Research*, volume 49, numéro 6, pages 938–949, 2001.
- [Gal93] Ian Gale. « Price Dispersion in a Market with Advance Purchase ». *Review of Industrial Organization*, volume 8, pages 451–464, 1993.
- [GGLM92] F. Glover, R. Glover, J. Lorenzo et C. McMillan. « The Passenger Mix Problem in the Scheduled Airlines ». *Interfaces*, volume 12, numéro 3, pages 73–79, 1992.
- [GIPD04] G. Gallego, G. Iyengar, R. Phillips et A. Dubey. « Managing Flexible Products on a Network ». CIRC Technical Report tr 2004-01, 2004.
- [GM98] Donald Garvett et Laurence Michaels. « Price Pararraying : A Direction for Quick, Decisive and Profit-Maximizing Pricing ». In Butler et Keller [BK98], pages 333–348.
- [GSF99] Sommer Gentry, Venkatesh Saligrama et Eric Feron. « Dynamic Inverse Optimization ». 1999.
- [GvR94] Guillermo Gallego et Garrett van Ryzin. « Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons ». *Management Science*, volume 40, numéro 8, pages 999–1020, 1994.
- [GvR97] Guillermo Gallego et Garrett van Ryzin. « A Multiproduct Dynamic Pricing Problem and its Applications to Network Yield Management ». *Operations Research*, volume 45, numéro 1, pages 24–41, 1997.
- [Heu] Clemens Heuberger. « Inverse Combinatorial Optimization : A Survey on Problems, Methods, and Results ». À paraître dans *Journal of Combinatorial Optimization*.
- [HS00] Orit Hassin et Oz Shy. « Code-Sharing Agreements, Frequency of Flights, and Profits in the Airline Industry ». 2000.
- [Hum01] Salal Humair. *Yield Management for Telecommunication Networks : Defining a New Landscape*. Thèse de doctorat, Massachusetts Institute of Technology, 2001.
- [Ilo02] Ilog. *CPLEX 8.1 User's manual*. Ilog S.A., 2002.
- [Jen95] Darryl Jenkins, éditeur. *Handbook of Airline Economics*. Aviation Week Group, McGraw-Hill, 1995.
- [KGD00] A. Kuyumcu et A. Garcia-Diaz. « A Polyhedral Graphs Theory Approach to Revenue Management in the Airline Industry ». *Computers and Industrial Engineering*, volume 38, numéro 3, pages 375–395, 2000.

- [Kim90] S. Kimes. « Yield management : A tool for capacity-constrained service firms ». *Journal of Operations Management*, volume 8, pages 348–363, 1990.
- [Kre95] Steven S. Kretsch. « Airline Fare Management and Policy ». In Jenkins [Jen95], pages 477–482.
- [Leg91] Jacques Legrand, éditeur. *Chronique de l'aviation*. Éditions Chronique, 1991.
- [Lei76] George Leitman. *Multicriteria Decision Making and Differential Games*. Plenum Press, 1976.
- [LH93] T. C. Lee et M. Hersh. « A Model for Dynamic Airline Seat Inventory Control with Multiple Seat Bookings ». *Transportation Science*, volume 27, numéro 3, pages 252–265, 1993.
- [Lit72] K. Littlewood. « Forecasting and control of passenger bookings ». *AGIFORS Symposium Proceedings*, volume 12, pages 95–117, 1972.
- [LMS98] M. Labbé, P. Marcotte et G. Savard. « A bilevel model of taxation and its application to optimal highway pricing ». *Management Science*, volume 44, pages 1595–1607, 1998.
- [LMS99] M. Labbé, P. Marcotte et G. Savard. « On a class of bilevel programs ». In Gianni Di Pillo et Franco Giannessi, éditeurs. *Nonlinear Optimization and Related Topics*, pages 183–206. Kluwer Academic Publishers, 1999.
- [LS99] C.J. Lautenbacher et S. Shaler. « The Underlying Markov Decision Process in the Single-Leg Airline Yield-Management Problem ». *Transportation Science*, volume 33, numéro 2, pages 136–146, 1999.
- [MB92] Maria Mallory et Greg Bowens. « Oh, What a Lovely Fare War ». *Business Week*, numéro 3272, page 37, 1992.
- [MS01a] P. Marcotte et G. Savard. « Bilevel Programming : Algorithms ». In Floudas et Pardalos [FP01], pages 156–158.
- [MS01b] P. Marcotte et G. Savard. « Bilevel Programming : Applications ». In Floudas et Pardalos [FP01], pages 158–159.
- [MS01c] P. Marcotte et G. Savard. « Bilevel Programming : Formulation ». In Floudas et Pardalos [FP01], pages 155–156.
- [MvR99] J.L. McGill et G.J. van Ryzin. « Revenue Management : Research Overview and Prospects ». *Transportation Science*, volume 33, numéro 2, pages 233–256, 1999.
- [NS01] Serguei Netessine et Robert A. Shumsky. « Revenue Management Games ». 2001.

- [OR94] Martin J. Osborne et Ariel Rubinstein. *A Course in Game Theory*. MIT Press, 1994.
- [Ost95] Andrew Osterland. « Can Airlines Stand Prosperity? ». *Financial World*, volume 164, numéro 24, pages 26–29, 1995.
- [PP02] Kevin Pak et Nanda Piersma. « Airline Revenue Management : An Overview of OR Techniques 1982-2002 ». Econometric Institute, Erasmus University, Rotterdam, Pays-Bas, 2002.
- [Rap89] Anatol Rapaport. *Decision Theory and Decision Behaviour : Normative and Descriptive Approaches*. Kluwer Publishing, 1989.
- [Rob93] L. W. Robinson. « Optimal and Approximate Control Policies for Airline Booking with Sequential Nonmonotonic Classes ». *Operations Research*, volume 43, numéro 2, pages 252–263, 1993.
- [Rom97] Graham Romp. *Game Theory : Introduction and Applications*. Oxford University Press, 1997.
- [Rot71] M. Rothstein. « An airline overbooking model ». *Transportation Science*, numéro 5, pages 180–192, 1971.
- [Sjo93] William Sjostrom. « Antitrust Immunity for Shipping Conferences : An Empty Core Approach ». *Antitrust Bulletin*, volume 38, pages 419–423, 1993.
- [SLD92] B. Smith, J. Leimkuhler et R. Darrow. « Yield management at American Airlines ». *Interfaces*, volume 22, numéro 1, pages 8–31, 1992.
- [Smi95] Timothy K. Smith. « Why Air Travel Doesn't Work ». *Fortune Magazine*, volume 131, numéro 6, pages 42–59, 1995.
- [SN93] François Soumis et Anna Nagurney. « A Stochastic, Multiclass Airline Network Equilibrium Model ». *Operations Research*, volume 41, numéro 4, pages 721–730, 1993.
- [SSL99] J. Subramanian, S. Shaler et C.J. Lautenbacher. « Airline Yield Management with Overbooking, Cancellations, and No-Shows ». *Transportation Science*, volume 33, numéro 2, pages 147–167, 1999.
- [Tal01] Kalyan T. Talluri. « Airline Revenue Management with Passenger Routing control : A New Model with Solution Approaches ». *International Journal of Services Technology and Management*, volume 2, numéro 1/2, pages 102–115, 2001.
- [Tel96] Lester G. Telser. « Competition and the Core ». *Journal of Political Economy*, volume 104, numéro 11, pages 85–107, 1996.

- [TvR98] Kalyan Talluri et Garrett van Ryzin. « An Analysis of Bid-Price Controls for Network Revenue Management ». *Management Science*, volume 44, numéro 11, pages 1577–1593, 1998.
- [TvR99] Kalyan Talluri et Garrett van Ryzin. « A Randomized Linear Programming Method for Computing Network Bid Prices ». *Transportation Science*, volume 33, numéro 2, pages 207–216, 1999.
- [TvR01] Kalyan Talluri et Garrett van Ryzin. « Revenue Management Under a General Discrete Choice Model of Consumer Behavior ». 2001.
- [Vic01] L. N. Vicente. « Bilevel Programming : Introduction, History and Overview ». In Floudas et Pardalos [FP01], pages 178–180.
- [Vin95] Ben Vinod. « Origin-and-Destination Yield Management ». In Jenkins [Jen95], pages 459–468.
- [vNM53] John von Neumann et Oskar Morgenstern. *Theory of Games and Economic Behaviour*. John Wiley and Sons, 1953.
- [vRM00] Garrett van Ryzin et Jeff McGill. « Revenue Management Without Forecasting or Optimisation : An Adaptive Algorithm for Determining Seat Protection Levels ». *Management Science*, volume 46, numéro 6, pages 760–775, 2000.
- [VvRM01] Gustavo Vulcano, Garrett van Ryzin et Costis Maglaras. « Optimal Dynamic Auctions for Revenue Management ». 2001.
- [WB92] L. Weatherford et S. Bodily. « A taxonomy and research overview of perishable asset revenue management : Yield management, overbooking, and pricing ». *Operations Research*, volume 40, numéro 5, pages 831–844, 1992.
- [WBP93] Lawrence R. Weatherford, Samuel E. Bodily et Phillip E. Pfeiffer. « Modeling the Customer Arrival Process and Comparing Decision Rules in Perishable Asset Revenue Management Situations ». *Transportation Science*, volume 27, numéro 3, pages 239–251, 1993.
- [Wil92] E. L. Williamson. *Airline Network Seat Inventory Control : Methodologies and Revenue Impacts*. Thèse de doctorat, Massachusetts Institute of Technology, 1992.
- [WKD93] J.T. Wong, F.S. Koppelman et S.D. Daskin. « Flexible Assignment Approach to Itinerary Seat Allocation ». *Transportation Research B*, volume 27B, numéro 1, pages 33–48, 1993.

- [Wol92] Richard D. Wollmer. « An Airline Seat Management Model for a Single Leg Route when Lower Fare Classes Book First ». *Operations Research*, volume 40, numéro 1, pages 26–37, 1992.
- [Wol98] Laurence A. Wolsey. *Integer Programming*. Wiley Interscience, 1998.
- [WP94] L. R. Weatherford et P. E. Pfeiffer. « The Economic Value of Using Advance Booking of Orders ». *International Journal of Management Science*, volume 22, numéro 1, pages 105–111, 1994.
- [YI97] Ian Yeoman et Anthony Ingold, éditeurs. *Yield Management : Strategies for the Service Industries*. Cassel, 1997.
- [You99] P.-S. You. « Dynamic Pricing in Airline Seat Management for Flights with Multiple Flight Legs ». *Transportation Science*, volume 33, numéro 2, pages 192–206, 1999.
- [YZM97] C. Yang, J. Zhang et Z. Ma. « Inverse Maximum Flow and Minimum Cut Problems ». *Optimization*, volume 40, pages 147–170, 1997.
- [Zen01] Richard H. Zeni. *Improved Forecast Accuracy in Airline Revenue Management by Unconstraining Demand Estimates from Censored Data*. Thèse de doctorat, Rutgers University, 2001.
- [ZL96] Jianzhong Zhang et Zhenhong Liu. « Calculating Some Inverse Linear Programming Problems ». *Journal of Computational and Applied Mathematics*, volume 72, pages 261–273, 1996.
- [ZRS92] Wendy Zellner, Andrea Rothman et Eric Schine. « The Airline Mess ». *Business Week*, numéro 3273, pages 50–55, 1992.