

UNIVERSITÉ DE MONTRÉAL

RÉDUCTION DES SAUTS D'INTÉGRITÉ
DANS LES PROBLÈMES D'AFFECTATION DE LOCOMOTIVES
POUR UN TRANSPORTEUR DE MARCHANDISES

STÉPHANE ROUILLON
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

THÈSE PRÉSENTÉE EN VUE DE L'OBTENTION
DU DIPLÔME DE PHILOSOPHIAE DOCTOR (Ph.D.)
(MATHÉMATIQUES DE L'INGÉNIEUR)

JUIN 2004



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 0-612-98185-1

Our file Notre référence

ISBN: 0-612-98185-1

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée:

RÉDUCTION DES SAUTS D'INTÉGRITÉ
DANS LES PROBLÈMES D'AFFECTATION DE LOCOMOTIVES
POUR UN TRANSPORTEUR DE MARCHANDISES

présentée par: ROUILLON Stéphane

en vue de l'obtention du diplôme de: Philosophiae Doctor

a été dûment acceptée par le jury d'examen constitué de:

M. AUDET Charles, Ph.D., président

M. DESAULNIERS Guy, Ph.D., membre et directeur de recherche

M. SOUMIS François, Ph.D., membre et codirecteur de recherche

M. ROUSSEAU Louis-Martin, Ph.D., membre

M. RENAUD Jacques, Ph.D., membre externe

A tous ceux pour qui le
pouvoir n'est pas un but en
soi.

REMERCIEMENTS

Je tiens tout d'abord à remercier particulièrement mes directeur et codirecteur de recherche, M. Guy Desaulniers et François Soumis pour leur patience et surtout leur confiance. Merci aussi à mon "collègue" François Lessard qui m'a fait découvrir la recherche opérationnelle.

Mes sincères remerciements à l'ensemble de l'équipe du *GERAD* avec qui ce fut un plaisir de travailler: à Beyime Tachefine et Koorush Ziarati, les spécialistes du groupe *rail*, à Norbert Lingaya et Daniel Villeneuve du groupe *Gencol* pour leurs précieuses explications, à Pierre Girard et François Guertin pour leur soutien informatique ainsi qu'à tous ceux et celles qui m'ont aidé à réaliser ces quelques pages.

RÉSUMÉ

Cette thèse expose quelques techniques permettant d'améliorer les temps de calculs et la qualité des solutions entières obtenues pour des problèmes d'affectation qui peuvent se formuler comme des problèmes multiflot avec contraintes liantes. Parmi ces problèmes, nous traitons particulièrement d'un problème de recouvrement généralisé, soit le problème d'affectation de locomotives pour les trains de marchandises. La taille d'un problème industriel de ce genre est telle que l'on ne peut pas énumérer l'ensemble des solutions et l'on doit se contenter d'explorer une fraction d'un arbre de recherche. Nos outils sont donc développés dans le cadre d'une recherche systématique incomplète par une méthode heuristique d'énumération implicite d'un problème en nombres entiers dont la relaxation linéaire peut se résoudre par génération de colonnes — une méthode de type *branch-and-price*. En utilisant trois différents scénarios appliqués à ce type de problème ferroviaire, nous validons les techniques développées soit pour perfectionner les tactiques de branchement aux noeuds de l'arbre de recherche, soit pour accélérer l'exploration de ce même arbre de recherche.

Le chapitre 1 contient la définition du problème d'affectation de locomotives aux trains de marchandises, ou plus simplement le cadre de ce travail. On y retrouve tout d'abord la formulation du problème

multiflot avec contraintes additionnelles, précisé pour le cas du problème d'affectation de locomotives. La section suivante présente comment la structure angulaire par bloc, particulière à ce type de formulation, peut-être décomposée en un problème maître et plusieurs sous-problèmes — un par commodité. Cette décomposition permet de résoudre une relaxation linéaire du problème par un processus de génération de colonnes. La dernière section indique comment incorporer ces relaxations linéaires dans le cadre d'une énumération implicite pour explorer une fraction de l'arbre des solutions disponibles.

Le chapitre 2 répertorie les principales parutions dans ce domaine. Quelques rappels d'approches fondamentales de résolution en nombres entiers précèdent les sections de la revue de la littérature plus spécifique aux tactiques de branchement et aux stratégies d'exploration, les deux principales composantes permettant d'explorer l'arbre de recherche. La dernière section rapporte les publications principales relatives aux problèmes ferroviaires dans ce domaine.

Le chapitre 3 étudie les tactiques de branchement dans le contexte de sous-problèmes de plus court chemin avec chemins interdits. En effet pour notre problème, les sous-problèmes obtenus pour chaque commodité par la décomposition de Dantzig-Wolfe sont des problèmes de plus court chemin. Pour explorer l'arbre des solutions de façon rapide et systématique, nous intégrons à chaque noeud la prise de décisions multiples et l'interdiction de chemins. La troisième section de ce chapitre présente trois tactiques de branchement conjuguant ces

éléments. La dernière section contient les résultats numériques obtenus avec les trois tactiques de branchement pour trois scénarios différents avec la stratégie d'exploration classique en profondeur-d'abord.

Le chapitre 4 expose une autre stratégie d'exploration appelée exploration par meilleures attentes. La première section indique les principales composantes de cette stratégie: une extrapolation de l'objectif espéré, une mesure d'intégralité et une recherche en coups de sonde. Par la suite, nous révisons ces composantes en les évaluant non seulement à partir des flots fractionnaires, mais aussi en tenant compte de l'impact des coûts des variables du modèle sur la qualité de la solution. Enfin, nous présentons les résultats numériques obtenus.

Le chapitre 5 exploite la structure de coût du problème de recouvrement généralisé pour améliorer le branchement. Tout d'abord, nous prédisons de nouvelles estimations des détériorations causées par une décision élémentaire. Ces estimations proviennent de problèmes d'optimisation simplifiés pour chaque arc du réseau associé à une commodité. A partir de ces nouvelles estimations, de nouveaux critères de branchement sont construits pour identifier les variables de chemin fixées à l'unité ainsi que le dernier chemin interdit. Il devient même efficace de fixer des variables de chemin à des valeurs entières non-unitaires. La dernière section montre les résultats numériques obtenus avec nos scénarios de référence précédents.

Le chapitre 6 décrit les étapes permettant de construire un schéma de branchement prédicteur-correcteur de recherche dans l'arbre des solutions. Un premier noeud correspondant au noeud de gauche des branchements employés au chapitre 3 constitue le noeud-prédicteur. Une analyse de ce noeud permet de réviser les détériorations associées à chaque décision élémentaire, puis de réordonner ces décisions. Celles-ci sont ensuite regroupées pour construire un ou plusieurs noeud(s)-correcteur(s). La dernière section présente les résultats obtenus avec ou sans retour-arrière pour un branchement prédicteur-correcteur appliqué aux scénarios de référence.

Chacun des quatre chapitres d'applications représente une étape supplémentaire pour essayer d'isoler l'élément ayant le plus d'impact sur la qualité des solutions. Le chapitre 3 permet une recherche systématique de l'arbre des solutions en permettant une partition des noeuds-fils. La résolution bénéficie ainsi de la force brute de l'ordinateur en explorant à chaque noeud une solution différente. En généralisant simplement la méthode de Ziarati et al. [64] en permettant un retour-arrière par profondeur-d'abord, on obtient un gain de 38% du saut d'intégrité relativement à la solution initiale pour la moyenne des scénarios étudiés. Toujours estimé après 24 heures de temps de calcul CPU, le chapitre 4 introduit une stratégie d'exploration qui permet un gain supplémentaire de 4% en moyenne pour l'ensemble des tactiques de branchement. En outre, cela met en évidence la supériorité de la tactique de branchement dynamique ambivalente. Celle-ci permet un gain supplémentaire de 5% de saut d'intégrité

relatif. Le chapitre 5 ne produit pas en général de meilleurs résultats que les techniques précédentes à l'exception du scénario 1 lorsque le paramètre d'accélération est réduit à sa valeur minimale ($A=1$). Nous ne retenons donc pas l'approche d'estimation des détériorations associées aux décisions élémentaires grâce à des problèmes linéaires simplifiés. Le dernier chapitre confirme la supériorité du branchement dynamique ambivalent lorsque combiné à un schéma prédicteur-correcteur. La réduction additionnelle relativement faible (4%) en terme de saut d'intégrité relatif représente toutefois plusieurs heures de temps de calcul. La combinaison des quatre composantes citées plus haut constitue, après 24 heures de temps de calcul, une réduction de saut d'intégrité de 47% relativement aux solutions obtenues par la méthode initiale de Ziarati et al. [64] après 15 minutes pour la moyenne des scénarios étudiés. En utilisant le scénario 2 comme référence, ces économies peuvent être estimées à plus de six millions de dollars (6 340 000 \$) annuellement pour un transporteur de marchandises comme le *Canadian National Inc.* Après seulement deux heures de temps de calcul, la réduction moyenne du saut d'intégrité relatif de 43% représente déjà plus de cinq millions de dollars (5 100 000 \$) en utilisant le scénario 2 comme référence. Ces estimations sont confirmées par les résultats obtenus pour un scénario d'une semaine. Enfin, les outils proposés sont aussi fonctionnels pour tout problème multiflot à contraintes liantes comportant des sous-problèmes de plus court chemin avec ou sans ressource.

ABSTRACT

This thesis presents a number of techniques that improve the computation times and the quality of integer solutions obtained for assignment problems that can be represented as multicommodity network flow problems with linking constraints. Among these problems, we focus more specifically on a generalized set covering problem, i.e. the locomotive assignment problem for freight trains. An industrial problem of this kind is of such magnitude that listing every solution is impossible and we must limit ourselves to investigate a fraction of a search tree. Our tools are thus developed within the framework of an incomplete systematic search using a heuristic branch-and-price implicit enumeration. We apply three different scenarios to our typical railway problem and validate the developed techniques either to perfect the tactics of branching at the nodes of the search tree or to accelerate the exploration of the same search tree.

Chapter 1 provides a definition of the locomotive assignment problem in freight trains, the problem we use as reference throughout this work. First, the multicommodity network flow problem with additional constraints is formulated, tailored to the locomotive assignment problem. The next section explains the angular block structure specific to this type of formulation: a master problem and several subproblems (one per commodity). This structure allows for solving a linear relaxation of the problem by a column generation

process. The last section shows how these linear relaxations can be embedded in an implicit enumeration in order to explore a fraction of the solution tree.

Chapter 2 lists the main publications in this field. A few reminders of fundamental solution approaches to obtain integer solutions precede the sections presenting the literature on branching tactics and exploration strategies. Both are the main constituents to search the solution tree. The last section reviews the main papers relating to railway problems in this field.

Chapter 3 studies branching tactics in the context of shortest path subproblems with forbidden paths. Indeed in the case at hand, the subproblems obtained for every commodity by the Dantzig-Wolfe decomposition are shortest path problems. To explore the solution tree rapidly and systematically, we develop every node using multiple decisions and forbidden paths. The third section of this chapter presents three branching tactics conjugating these elements. The last section contains the numerical results obtained with the three branching tactics for three different scenarios using the classical depth-first exploration strategy.

Chapter 4 presents another investigation strategy called exploration by best expectations. The first section sets out the main components of this strategy: an extrapolation of the expected

objective, a measure of integrality and a probing search. We then revise these components, estimating them not only from a fractional flow perspective, but also taking into account the impact of the costs of the model variables on the quality of the solution. Finally, we present the numerical results obtained.

Chapter 5 considers the cost structure of the generalized set covering problem to improve the branching. First, we predict new estimates of the deteriorations caused by an elementary decision. These estimates result from simplified optimization problems for every network arc associated with a commodity. From these new estimates, new branching criteria are built to identify the path variables fixed at unity as well as the last forbidden path. It even becomes effective to fix path variables to non-unitarian integer values. The last section shows the numerical results obtained with our previous reference scenarios.

Chapter 6 describes the steps for building a predictor-corrector branching scheme search for the solution tree. A first node, corresponding to the left branching node used in chapter 3, constitutes the predictor-node. By analyzing this node we can first revise the deteriorations associated with every elementary decision, then re-order these decisions. The decisions are then classified to build one or several corrector-node(s). The last section presents the results

obtained with or without backtracking for a predictor-corrector branching tactic applied to our reference scenarios.

Each of the four application chapters represents an additional step to isolate the element with the most impact on the quality of the solutions. Chapter 3 presents a systematical search in the solution tree. Thus, the resolution benefits from the raw power of the computer by investigating a different solution every time. By simply generalizing the Ziarati *et al.* [64] method with a depth-first backtracking technique, we obtain a 38% earning of the integrality gap with regard to the initial solution for the average of the scenarios under study. Chapter 4 introduces an exploration strategy that allows an additional earning estimated at 4% on average for all branching tactics, after 24 hours of computation time (CPU). Furthermore, it highlights the superiority of the dynamic and ambivalent branching. This tactic allows an additional 5% reduction of the relative gap. The techniques discussed in chapter 5 do not produce better overall results than the previous techniques, with the exception of scenario 1, where the acceleration parameter is reduced to its minimal value ($A=1$). Thus, the simplified linear problem approach is not retained to estimate the deteriorations associated with each elementary decision. Finally, the last chapter confirms the superiority of the dynamic ambivalent tactic when combined with a predictor-corrector scheme. However, the additional but relatively weak reduction (4%) in terms of relative integrality gap represents several hours of computation time.

After 24 hours of computation time, the four quoted components previously described taken as a whole produce a 47% reduction of the integrality gap compared to the solutions obtained by the initial Ziarati et al. [64] method for the average of the scenarios under study. Using scenario 2 as a reference, the savings can be estimated at approximately six million dollars (\$ 6,340,000) annually for a railway freight carrier such as the Canadian National of North America Inc. Still using scenario 2 as a reference, after only two hours of computation time, the relative 43% integrality gap mean reduction represents more than five million dollars (\$ 5,100,000). These estimations were confirmed by results obtained for a weekly scenario. Finally, the proposed tools are also functional for any multicommodity network flow problem with linking constraints containing shortest path subproblems with or without resources.

TABLE DES MATIÈRES

DÉDICACE	iv
REMERCIEMENTS	v
RÉSUMÉ	vi
ABSTRACT	xi
TABLE DES MATIÈRES	xvi
LISTE DES TABLEAUX	xx
LISTE DES FIGURES	xxi
LISTE DES SIGLES ET ABREVIATIONS	xxvii
LISTE DES ANNEXES	xxviii
INTRODUCTION	1
CHAPITRE 1: DÉFINITION DU PROBLÈME	5
1.1) LE PROBLÈME MULTIFLOTS AVEC CONTRAINTES ADDITIONNELLES	5
1.1.1) Formulation générale	5
1.1.2) Application au domaine ferroviaire	9
1.2) DÉCOMPOSITION DE DANTZIG-WOLFE.....	13
1.2.1) Problème maître	14
1.2.2) Sous-problème	16
1.2.3) Processus de génération de colonnes	16
1.2.4) Impact d'une décision de branchement	20
1.3) LA MÉTHODE D'ÉNUMÉRATION IMPLICITE.....	21
1.3.1) Tactiques de branchement	21
1.3.2) Stratégies d'exploration	25

CHAPITRE 2: RECHERCHE BIBLIOGRAPHIQUE	27
2.1) APPROCHES DE RÉOLUTION EN NOMBRES ENTIERS.....	27
2.2) LES TACTIQUES DE BRANCHEMENT.....	30
2.3) LES STRATÉGIES D'EXPLORATION.....	32
2.4) LE PROBLÈME D'AFFECTATION DE LOCOMOTIVES.....	37
CHAPITRE 3: TACTIQUES DE BRANCHEMENT AVEC CHEMINS	
INTERDITS	40
3.1) PRISE DE DÉCISIONS MULTIPLES.....	42
3.2) BRANCHEMENT INTERDISANT DES CHEMINS.....	48
3.2.1) Méthode des déviations	49
3.2.2) Méthode des k plus courts chemins	53
3.2.3) Comparaison entre les méthodes	55
3.3) DIFFÉRENTES TACTIQUES DE BRANCHEMENT.....	56
3.3.1) Interdiction du dernier meilleur chemin	57
3.3.2) Interdiction du pire chemin	61
3.3.3) Prise de décisions sur les chemins	
ambivalents	62
3.4) RÉSULTATS NUMÉRIQUES.....	65
CHAPITRE 4: STRATÉGIE D'EXPLORATION PAR MEILLEURES	
ATTENTES	73
4.1) OUTILS DE RECHERCHE BASÉS SUR LES FLOTS	
FRACTIONNAIRES	74
4.1.1) Extrapolation de l'objectif espéré	74
4.1.1.1) Références de l'extrapolation.....	76
4.1.1.2) Mesure de l'intégralité.....	79

4.1.2)	Recherche en coups de sonde	82
4.1.3)	Résultats numériques	85
4.2)	OUTILS DE RECHERCHE BASÉS SUR LES COÛTS.....	96
4.2.1)	Révision de la mesure d'intégralité	96
4.2.2)	Résultats numériques	100
4.2.3)	Révision des références d'extrapolation	106
CHAPITRE 5: TACTIQUES DE BRANCHEMENT EXPLOITANT LA		
	STRUCTURE DE COÛT	110
5.1)	ESTIMATION DES DÉTÉRIORATIONS.....	111
5.1.1)	Évaluation des détériorations pour un arc ..	112
5.1.1.1)	Détériorations pour un segment de train	116
5.1.1.2)	Détériorations pour les autres types d'arcs	121
5.1.2)	Nouveaux critères de branchement pour un chemin	123
5.1.2.1)	Détériorations pour un chemin.....	124
5.1.2.2)	Nouveaux critères de sélection.....	127
5.2)	VARIABLES FIXÉES À DES VALEURS ENTIÈRES NON-UNITAIRES	130
5.3)	RÉSULTATS NUMÉRIQUES.....	134
CHAPITRE 6: UN BRANCHEMENT PRÉDICTEUR-CORRECTEUR		
6.1)	ANALYSE DES DÉCISIONS ÉLÉMENTAIRES A POSTERIORI	143
6.1.1)	Révision des détériorations	143

6.1.2) Critères de réordonnancement	147
6.2) REGROUPEMENT DES DÉCISIONS ÉLÉMENTAIRES.....	148
6.3) CONSTRUCTION DES NOEUDS-CORRECTEURS.....	157
6.4) RÉSULTATS NUMÉRIQUES.....	160
6.4.1) Résultats sans retour-arrière	161
6.4.2) Résultats avec retour-arrière	164
6.4.3) Résultats pour un scénario d'une semaine ...	170
CONCLUSION	173
BIBLIOGRAPHIE	180
ANNEXE	189

LISTE DES TABLEAUX

Tableau 3.1: Caractéristiques des scénarios étudiés	66
Tableau 4.1: Gain de saut d'intégrité relatif après quatre heures de calculs	90
Tableau 4.2: Gain de saut d'intégrité relatif après vingt-quatre heures de calculs	91
Tableau 4.3: Caractéristiques des combinaisons étudiées	94
Tableau 5.1: Valeur de la fonction objectif pour la meilleure solution entière après six heures de temps de calcul (<i>CPU</i>)	135
Tableau 5.2: Saut d'intégrité pour la meilleure solution entière après six heures de temps de calcul (<i>CPU</i>)	136
Tableau 5.3: Qualité des prédictions de détérioration pour la première solution entière	137
Tableau 5.4: Qualité des prédictions de détérioration des décisions à flot non-unitaire pour la première solution entière ..	138
Tableau 5.5: Différence entre détériorations prévues et constatées pour les trois scénarios	139
Tableau 6.1: Valeur de la première solution entière	162
Tableau 6.2: Résultats pour un scénario d'une semaine	172

LISTE DES FIGURES

Figure 1.1: Réseau espace-temps des routes admissibles aux locomotives	10
Figure 1.2: Structure d'un problème de génération de colonnes	17
Figure 1.3: Processus de résolution d'un problème de génération de colonnes	19
Figure 3.1: Exemple d'arbre de branchement binaire	43
Figure 3.2: Report du calcul de la relaxation linéaire	44
Figure 3.3: Equivalence entre décisions élémentaires et multiples	45
Figure 3.4: Représentation schématique du rejet des noeuds de droite .	46
Figure 3.5: Comparaison de la diversité d'un échantillonnage	47
Figure 3.6: Résultat de l'application de l'algorithme de plus court chemin avec pré-étiquetage des chemins interdits	50
Figure 3.7: Branchement interdisant le dernier meilleur chemin	58
Figure 3.8: Caractéristiques d'un modèle de distribution des coûts des solutions entières d'un sous-arbre voisin L_2^v	59
Figure 3.9: Sous-arbre à plus forte densité de meilleures solutions ..	60
Figure 3.10: Branchement interdisant le pire chemin	61
Figure 3.11: Branchement sur les chemins ambivalents	64

Figure 3.12: Evolution du coût de la meilleure solution entière pour le scénario 1 en stratégie d'exploration profondeur- d'abord	68
Figure 3.13: Evolution du coût de la meilleure solution entière pour le scénario 2 en stratégie d'exploration profondeur- d'abord	69
Figure 3.14: Evolution du coût de la meilleure solution entière pour le scénario 3 en stratégie d'exploration profondeur- d'abord	70
Figure 3.15: Evolution moyenne du coût de la meilleure solution entière pour tous les scénarios en stratégie d'exploration profondeur-d'abord	71
Figure 4.1: Modèle d'interpolation de la valeur de l'objectif entre la solution du noeud-racine et la meilleure solution entière courante	75
Figure 4.2: Comparaison des techniques d'extrapolation de l'objectif espéré	77
Figure 4.3: Extrapolations locales systématiquement optimistes	79
Figure 4.4: Comparaison des mesures d'intégralité	81
Figure 4.5: Exploration en coups de sonde	83

Figure 4.6: Evolution du coût de la meilleure solution entière pour le scénario 1 en stratégie d'exploration par meilleures attentes	86
Figure 4.7: Evolution du coût de la meilleure solution entière pour le scénario 2 en stratégie d'exploration par meilleures attentes	87
Figure 4.8: Evolution du coût de la meilleure solution entière pour le scénario 3 en stratégie d'exploration par meilleures attentes	88
Figure 4.9: Evolution moyenne du coût de la meilleure solution entière pour les scénarios en stratégie d'exploration par meilleures attentes	89
Figure 4.10: Comparaison du coût de la meilleure solution entière entre stratégies d'exploration par meilleures attentes et profondeur-d'abord	92
Figure 4.11: Evolution en escalier des valeurs de l'objectif de la solution relaxée	97
Figure 4.12: Comparaison des mesures d'intégralité entre les deux versions de la stratégie d'exploration par meilleures attentes	99

Figure 4.13: Evolution du coût de la meilleure solution entière pour le scénario 1 en stratégie d'exploration par meilleures attentes version 2	101
Figure 4.14: Evolution du coût de la meilleure solution entière pour le scénario 2 en stratégie d'exploration par meilleures attentes version 2	102
Figure 4.15: Evolution du coût de la meilleure solution entière pour le scénario 3 en stratégie d'exploration par meilleures attentes version 2	103
Figure 4.16: Evolution moyenne du coût de la meilleure solution entière pour les scénarios en stratégie d'exploration par meilleures attentes version 2	104
Figure 4.17: Comparaison du coût de la meilleure solution entière entre les diverses stratégies d'exploration	105
Figure 4.18: Extrapolation adaptée pour la stratégie d'extrapolation par meilleures attentes	108
Figure 5.1: Analyse du gaspillage pour une contrainte de couverture .	117
Figure 5.2: Analyse de la détérioration	119
Figure 5.3: Interdiction de l'un des deux chemins d'une solution	126
Figure 5.4: Autre paire de chemins formant la même solution	126
Figure 5.5: Branchement multiple	132

Figure 5.6: Branchement binaire	133
Figure 6.1: Détériorations maximale et réelle associées à une décision élémentaire	144
Figure 6.2: Détermination des limites de séparation par référence ...	153
Figure 6.3: Détermination des limites de séparation par différences .	155
Figure 6.4: Exemples de regroupements à corriger	156
Figure 6.5: Construction du noeud-prédicteur	157
Figure 6.6: Branchement corrigé théorique	158
Figure 6.7: Branchement corrigé pratique	159
Figure 6.8: Comparaison du coût de la meilleure solution entière pour le scénario 1 avec et sans schéma prédicteur- correcteur	165
Figure 6.9: Comparaison du coût de la meilleure solution entière pour le scénario 2 avec et sans schéma prédicteur- correcteur	166
Figure 6.10: Comparaison du coût de la meilleure solution entière pour le scénario 3 avec et sans schéma prédicteur- correcteur	167
Figure 6.11: Comparaison du coût moyen de la meilleure solution entière pour tous les scénarios avec et sans schéma prédicteur-correcteur	168

Figure 6.12: Décomposition de l'horizon de temps étudié en fenêtres

d'optimisation 170

Figure 7.1: Comparaison du coût de la meilleure solution entière

pour tous les scénarios en ajoutant chaque composante ... 177

LISTE DES SIGLES ET ABREVIATIONS

<i>CN: Canadian National Inc</i>	65
<i>CPU: Central Process Unit</i>	39
<i>DP: Drawbar Pull</i>	10
<i>GERAD: Groupe d'Études et de Recherche en Analyse des Décisions</i>	v
<i>HP: Horse Power</i>	10
<i>LR: Relaxation Linéaire (Linear Relaxation)</i>	44
<i>MIP: Mixed Integer Program</i>	38
<i>NP: Non-déterministe Polynômial</i>	29
<i>PCP: Point de Changement de Puissance</i>	9
<i>SPFP: Shortest Path problem with Forbidden Paths</i>	48

LISTE DES ANNEXES

Annexe 1: Notations employées par Ziarati [63]	189
--	-----

INTRODUCTION

GÉNÉRALITÉS

L'explosion combinatoire engendrée par les problèmes en nombres entiers constitue un défi passionnant pour la recherche opérationnelle. Pour résoudre de tels systèmes, on représente généralement l'ensemble des états possibles à l'intérieur d'un arbre de décision. Un noeud initial représente l'ensemble de tous les degrés de liberté du système. Chaque noeud peut être développé en plusieurs noeuds-fils et on définit le passage du noeud-parent au noeud-fils par des prises de décision qui contraignent davantage le problème parent. Chaque décision précise un ensemble de valeurs entières assignables à une ou plusieurs variables. Ce processus peut être répété afin d'énumérer tous les états du système et identifier une solution optimale.

Les développements des ressources informatiques permettent maintenant de trouver des solutions optimales en nombres réels pour des problèmes de grande taille. Toutefois, la résolution optimale en nombres entiers de ces problèmes peut nécessiter l'évaluation d'un nombre exponentiel de solutions. Aucun ordinateur, aussi puissant soit-il, ne peut parvenir à cette fin en un temps raisonnable pour de grands problèmes. En acceptant de restreindre l'énumération à une fraction du domaine des possibilités, on peut réduire le temps de calcul en écartant des sous-ensembles de grande taille. Nous verrons comment obtenir de bons résultats sans générer tous ces noeuds, à l'aide de méthodes heuristiques de séparation et évaluation progressive par génération de colonnes — *branch-and-price* [11] — contenant des plans coupants — *branch-and-cut*.

Les domaines d'applications de la programmation en nombres entiers sont innombrables: le design de molécules [54], les tournées de véhicules [42], la localisation d'entrepôts [55], la construction de ponts [59], le regroupement de terminaux électriques [48], la reconnaissance de forme [62], les jeux de décisions [7], certains problèmes matriciels [4], la construction des blocs de wagons [50], la construction d'horaires [23], ... On restreint toutefois le cadre de cette recherche à la résolution en nombres entiers du problème de flot multi-commodités — ou multiflots — avec contraintes additionnelles. Le modèle multiflots s'applique, par exemple, en télécommunications [38], en transport aérien [23], à la gestion d'équipages [21], à l'élaboration de routes de train [8] ou à la gestion d'un réseau informatique [14]. Plus précisément, on considère tout problème où on peut regrouper des sous-ensembles de contraintes et leur associer des sous-objectifs adéquats pour représenter plusieurs sous-problèmes de "plus court chemin" avec ou sans ressources. Ce sous-problème est courant en recherche opérationnelle.

Le sous-problème de plus court chemin sans ressource est un problème de flot. Sa représentation matricielle est unimodulaire [3] et sa résolution en variables continues produit des variables entières. Cette dernière remarque est aussi valable dans le cas des problèmes de plus court chemin avec fenêtre de temps ou avec ressources: il existe une solution optimale entière [23]. Cependant, le modèle multiflots contient des contraintes liantes: généralement, il possède une solution optimale fractionnaire et donc un saut — ou *gap* — d'intégrité. L'obtention d'une solution optimale entière pour le problème initial nécessite alors un arbre de décision et n'est pas triviale.

BUT DE LA RECHERCHE

La grande taille des systèmes définis précédemment explique le choix d'une approche heuristique. L'objectif consiste à trouver une solution satisfaisante, possiblement sous-optimale, en ne parcourant qu'une infime fraction de l'arbre d'énumération. Il faut que la méthode heuristique soit rapide et robuste pour garantir une recherche adéquate.

Cette recherche consiste donc à réduire la différence entre le coût de la meilleure solution obtenue et la borne la plus restrictive connue. Puisque dans cette étude on considère un contexte de minimisation, une borne inférieure constitue une limite optimiste de l'objectif et donc la borne la plus restrictive est la plus grande borne inférieure. Nous nous concentrerons sur trois stratégies permettant de parvenir à une réduction du saut d'intégrité: on peut améliorer la borne optimiste du problème, évoluer vers de meilleures solutions ou accélérer la convergence du processus de résolution.

Relaxer les contraintes d'intégralité à n'importe quel noeud est une méthode assez directe pour trouver une borne optimiste au problème. Un terme supplémentaire peut aussi être ajouté à cette borne pour représenter la détérioration de l'objectif due aux décisions restantes, ou on peut améliorer la borne en rapprochant la définition du problème de l'enveloppe convexe de ses solutions admissibles en y ajoutant des coupes.

La qualité de la tactique de branchement est le point clef de l'orientation de la recherche vers une meilleure solution. Si un branchement permet d'isoler une densité supérieure de solutions de bonne qualité, il sera statistiquement plus susceptible de trouver de meilleures solutions.

Finalement, l'accélération permet d'explorer plus rapidement les noeuds de l'arbre et donc d'obtenir une meilleure solution pour un même temps de calcul. C'est la stratégie de sélection de l'ordre d'exploration des noeuds qui permet d'identifier rapidement la meilleure solution sous-optimale considérée.

CHAPITRE 1

DÉFINITION DU PROBLÈME

Ce premier chapitre comprend la définition du problème de flot multi-commodités avec contraintes additionnelles, la description de la méthode de décomposition de Dantzig-Wolfe et un exposé sur les principaux éléments d'une méthode d'énumération implicite. Le problème multiflots avec contraintes additionnelles, constitue le cadre théorique de cette étude. Toutes les observations et les résultats obtenus sont aussi applicables aux autres problèmes de ce type. Cette classe de problèmes possède une structure spéciale: on montrera comment la décomposition de Dantzig-Wolfe tire partie de cette structure pour permettre la résolution optimale de la relaxation linéaire de problèmes de grande taille. Finalement, la dernière section explique comment, en exploitant à répétition ce processus, la méthode d'énumération implicite guide la recherche à travers l'arbre de décisions pour obtenir une solution entière.

1.1) LE PROBLÈME MULTIFLOTS AVEC CONTRAINTES ADDITIONNELLES

Dans un premier temps nous présentons la formulation générale du problème de multiflots avec contraintes additionnelles, puis son application au problème spécifique de l'affectation de locomotives pour tirer des trains dont les horaires sont déjà pré-établis.

1.1.1) Formulation générale

Le problème multiflots qui nous intéresse possède la structure générale suivante:

- une fonction objectif linéaire;
- des contraintes spécifiques à une commodité définissant un problème de plus court chemin avec ou sans ressources. Par conséquent, chaque flot provient d'un seul noeud-source pour se rendre à un seul noeud-puits. Lorsqu'on considère aussi qu'il n'y a pas de contraintes de capacité individuelle sur les arcs, cela permet de ramener tout flot différent de l'unité à un flot d'une unité en ajustant les coûts des arcs pour cette commodité;
- des contraintes liantes additionnelles qui lient les commodités entre elles et qui font possiblement appel à des variables autres que les variables de flot (par exemple, des variables d'écart ou de surplus).

Mathématiquement, le problème multiflots avec contraintes additionnelles est décrit dans Desaulniers et al. [22] dans le cadre des problèmes déterministes de tournées de véhicule et d'horaires d'équipages. Adapté au contexte de cette étude, il se formule ainsi:

$$\min_{X,Y} \sum_{k \in K} \sum_{(i,j) \in A^k} c_{ij}^k X_{ij}^k + \sum_{s \in S} c_s Y_s \quad (1.1)$$

$$\text{soumis à: } \sum_{k \in K} \sum_{(i,j) \in A^k} a_{v,ij}^k X_{ij}^k + \sum_{s \in S} a_{vs} Y_s = a_v \quad \forall v \in V \quad (1.2)$$

$$a_s \leq Y_s \leq b_s \quad \forall s \in S \quad (1.3)$$

$$\sum_{j: (o(k),j) \in A^k} X_{o(k),j}^k = n^k \quad \forall k \in K \quad (1.4)$$

$$\sum_{j: (i,j) \in A^k} X_{ij}^k - \sum_{j: (j,i) \in A^k} X_{ji}^k = 0 \quad \forall k \in K, \forall i \in N^k \setminus \{o(k), d(k)\} \quad (1.5)$$

$$\sum_{i: (i,d(k)) \in A^k} X_{i,d(k)}^k = n^k \quad \forall k \in K \quad (1.6)$$

$$X_{ij}^k \left(f_{ij}^{kr} (T_i^k) - T_j^{kr} \right) \leq 0 \quad \forall k \in K, \forall r \in R^k, \forall (i,j) \in A^k \quad (1.7)$$

$$a_i^{kr} \leq T_i^{kr} \leq b_i^{kr} \quad \forall k \in K, \forall r \in R^k, \forall i \in N^k \quad (1.8)$$

$$X_{ij}^k \geq 0 \text{ et entiers} \quad \forall k \in K, \forall (i,j) \in A^k. \quad (1.9)$$

Dans les expressions précédentes, la notation suivante est employée:

k : commodité spécifique de l'ensemble des commodités K ;

$G^k = (N^k, A^k)$: graphe acyclique associé à la commodité $k \in K$;

N^k : ensemble des noeuds comprenant le noeud-source
 $o(k)$ et le noeud-puits $d(k)$;

A^k : ensemble des arcs ;

c_{ij}^k : coût du passage d'une unité de flot sur l'arc $(i, j) \in A^k$;

X_{ij}^k : variable binaire de flot sur l'arc $(i, j) \in A^k$;

n^k : nombre d'unités disponibles pour la commodité $k \in K$;

R^k : ensemble des ressources associées à la commodité $k \in K$;

T_i^{kr} : variable indiquant la valeur de la ressource $r \in R^k$ lors de
la visite du noeud $i \in N^k$ par la commodité $k \in K$;

$T_i^k = (T_i^{kr})_{r \in R^k}$: vecteur des variables de ressource au noeud $i \in N^k$
pour la commodité $k \in K$;

$f_{ij}^{kr}(\cdot)$: fonction de prolongation de la ressource $r \in R^k$ sur l'arc
 $(i, j) \in A^k$ pour la commodité $k \in K$;

a_i^{kr}, b_i^{kr} : bornes inférieures et supérieures sur les valeurs
permises de la variable T_i^{kr} , $k \in K$, $r \in R^k$ et $i \in N^k$;

S : ensemble des indices des variables supplémentaires ;

Y_s : variable supplémentaire d'indice $s \in S$;

c_s : coût unitaire de la variable Y_s , $s \in S$;

a_s, b_s : bornes inférieures et supérieures sur les valeurs permises
de la variable Y_s , $s \in S$;

V : ensemble des contraintes globales ;

$a_{v,ij}^k$: coefficient indiquant la contribution de la variable X_{ij}^k ,
 $k \in K$, $(i, j) \in A^k$, à la contrainte globale $v \in V$;

a_{vs} : coefficient indiquant la contribution de la variable Y_s ,
 $s \in S$, à la contrainte globale $v \in V$;

a_v : membre de droite de la contrainte globale $v \in V$.

La fonction objectif (1.1) est linéaire et représente la somme des coûts reliés aux flots et aux variables d'écart. Les contraintes (1.2) lient les commodités et pourraient, entre autres, représenter des contraintes de couverture de tâches sur lesquelles on permet un sous-recouvrement. Les contraintes (1.3) limitent les valeurs permises aux variables d'écart. Les contraintes (1.4) et (1.6) induisent une seule unité de flot dans le réseau par commodité. Les contraintes (1.5) appliquent la loi de conservation du flot aux noeuds. Les contraintes (1.7) sont des contraintes de prolongation de ressources. Les contraintes (1.8) limitent les valeurs permises aux ressources. Les contraintes (1.9) sont des contraintes binaires sur le flot. Les contraintes (1.2) constituent des contraintes liantes alors que le premier terme de l'objectif (1.1) et les contraintes (1.4)-(1.9) peuvent être regroupées par commodité pour former des problèmes de plus court chemin avec ou sans ressources.

Les contraintes globales (1.2) ont parfois différentes formes. Par exemple, lorsque tout arc $(i, j) \in A^v$ représente un vol sur lequel on doit affecter un avion et un seul, les contraintes peuvent s'écrire:

$$\sum_{k \in K} X_{ij}^k = 1 \quad \forall (i, j) \in A^v. \quad (1.10)$$

On parle alors d'un problème multiflots de type partitionnement d'ensemble. Lorsque plusieurs flots sur le réseau contribuent à l'accomplissement d'une tâche sur un même arc $(i, j) \in A^w$, comme dans le cas où l'arc (i, j) représente un segment de train sur lequel on doit affecter au moins E_{ij} unités, les contraintes (1.2) prennent la forme:

$$\sum_{k \in K} e^k X_{ij}^k \geq E_{ij} \quad \forall (i, j) \in A^w, \quad (1.11)$$

où e^k représente la contribution d'une unité de flot de la commodité k à l'accomplissement d'une tâche et E_{ij} la demande pour satisfaire la tâche en question sur l'arc (i, j) . On obtient alors un problème multiflots de type recouvrement généralisé. Nous verrons ensuite à la sous-section 1.2.4 pourquoi ce dernier type de problème peut être plus long à résoudre.

1.1.2) Application au domaine ferroviaire

Un problème typique en transport ferroviaire est l'affectation des locomotives aux trains de marchandises. Nous utiliserons cet exemple pour valider les méthodes suggérées puisque c'est un problème multiflots de type recouvrement généralisé. C'est même un exemple très complexe car les contraintes de maintenance des commodités représentant des locomotives critiques comportent des fenêtres de temps. Le modèle employé tout au long de cette étude ainsi que sa notation proviennent de Ziarati [63]. La notation est présentée en annexe.

Il est possible de modéliser graphiquement toutes les routes admissibles pour des locomotives à travers un réseau espace-temps $G=(N,A)$. Les noeuds représentent un point de changement de puissance (PCP), généralement une gare, à divers moments dans le temps et les noeud-source et noeud-puits. L'ensemble des arcs A peut être décomposé en cinq sous-ensembles disjoints et complémentaires:

- I : ensemble des arcs d'entrée — d'input;
- W : ensemble des segments de trains;
- F : ensemble des arcs de voyage à vide — ou *deadhead*;
- Z : ensemble des arcs de repos;
- J : ensemble des arcs de sortie.

La figure 1.1 suivante illustre comment représenter un exemple de réseau acyclique G résultant. Les arcs d'entrée relient le noeud-source aux points de disponibilité initiale des locomotives et permettent de respecter les capacités initiales à chaque PCP. Les segments de trains représentent une portion de la route d'un train pendant laquelle aucune modification ne doit être apportée au convoi — ou *consist* — de locomotives. Chaque segment de train comporte plusieurs contraintes de recouvrement de type (1.11). Les arcs de voyage à vide représentent les trajets possibles de locomotives se déplaçant sans wagon. Finalement, les arcs de repos permettent de représenter les locomotives qui demeurent au même PCP pendant une certaine période de temps.

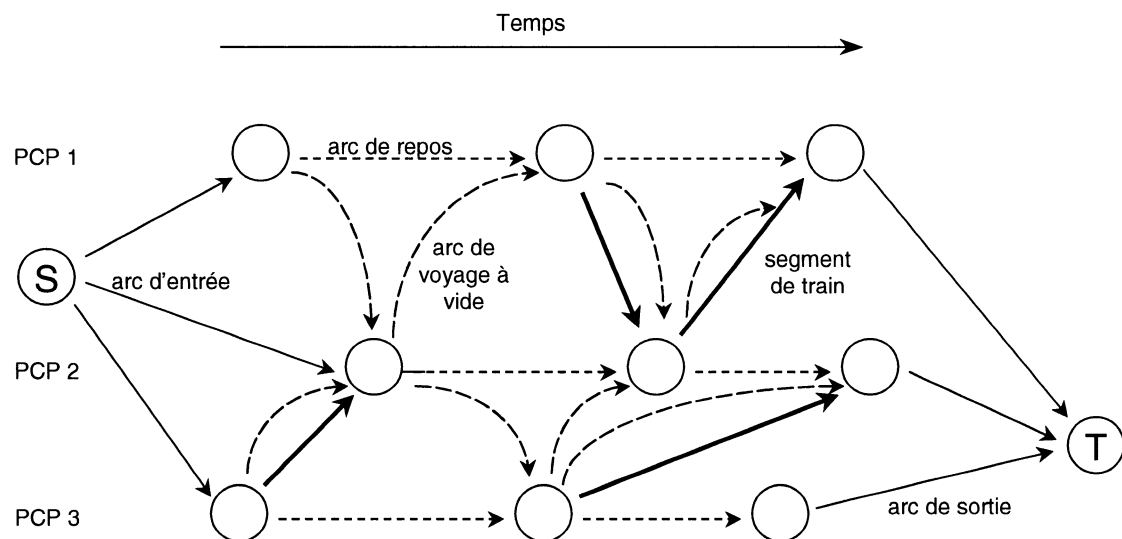


Figure 1.1: Réseau espace-temps des routes admissibles aux locomotives

On cherche donc à minimiser les coûts d'opération liés aux déplacements et aux pénalités de sous-recouvrement en puissance ou en tonnage (HP ou DP). D'autres expressions possibles de la fonction

objectif emploient des pénalités pour modéliser des locomotives artificielles ou l'introduction de retard sur les trains. On permet ainsi de combler un assemblage de locomotives incomplet. Notez que ces objectifs, bien que de nature différente, sont tous linéaires. Pour des précisions sur la notation employée, le lecteur est invité à consulter l'annexe 1. Voici donc l'objectif de ce problème type:

$$\min \sum_{k \in K} \sum_{(i,j) \in A^k} (c_{ij}^k X_{ij}^k + d_{ij}^k F_{ij}^k) + \sum_{s \in S^n \cup S^p \cup S^q} c_s Y_s. \quad (1.12)$$

Le modèle contient plusieurs contraintes liantes: le nombre minimal de locomotives dans un convoi (1.13), la puissance minimale d'un convoi (HP) (1.14), le poids minimal d'un convoi (DP) (1.15), la disponibilité des ateliers pour l'entretien des locomotives (1.16), les besoins d'un poste local (Outpost) (1.17), la garantie d'une puissance minimale aux gares critiques ou PCP (Point de Changement de Puissance) pour une solution cyclique (1.18) et des bornes sur les variables d'écart permettant le sous-recouvrement (1.19). Voici leur formulation:

$$\sum_{k \in K} \sum_{(i,j) \in A^k} a_{w,ij}^k X_{ij}^k + \sum_{s \in S^n} a_{ws} Y_s = n_w \quad \forall w \in W^n \quad (1.13)$$

$$\sum_{k \in K} \sum_{(i,j) \in A^k} a_{w,ij}^k X_{ij}^k p^k + \sum_{s \in S^p} a_{ws} Y_s = p_w \quad \forall w \in W^p \quad (1.14)$$

$$\sum_{k \in K} \sum_{(i,j) \in A^k} a_{w,ij}^k X_{ij}^k q^k + \sum_{s \in S^q} a_{ws} Y_s = q_w \quad \forall w \in W^q \quad (1.15)$$

$$\sum_{k \in K} \sum_{(i,j) \in A^k} g_{m,ij}^{kd} X_{ij}^k \leq n_m^d \quad \forall m \in M, \forall d \in D \quad (1.16)$$

$$\sum_{k \in K} \sum_{(i,j) \in A^k} g_{o,ij}^{kd} X_{ij}^k \geq n_o^d \quad \forall o \in O, \forall d \in D \quad (1.17)$$

$$\sum_{k \in K} \sum_{(i,j) \in A^k} g_{h,ij}^k X_{ij}^k p^k \geq p_h \quad \forall h \in H \quad (1.18)$$

$$a_s \leq Y_s \leq b_s \quad \forall s \in S^n \cup S^p \cup S^q. \quad (1.19)$$

Ce modèle contient en outre des contraintes supplémentaires: sur le nombre maximal d'essieux, le nombre maximal de locomotives actives et de déplacements à vide dans un convoi, le nombre de locomotives disponibles par classes et le nombre maximal disponible par arc d'entrée. Toutes ces contraintes faibles sont rarement violées et sont donc traitées dynamiquement (c.f. sous-section 1.3.1) comme dans une méthode de séparation et évaluation progressive avec plans coupants.

Les autres contraintes peuvent être réunies pour une même commodité. Elles forment alors un problème de plus court chemin, avec fenêtre de temps dans le cas des commodités représentant des locomotives critiques:

$$\sum_{j:(o(k),j) \in A^k} X_{o(k),j}^k = 1 \quad \forall k \in K \quad (1.20)$$

$$\sum_{j:(i,j) \in A^k} (X_{ij}^k + F_{ij}^k) - \sum_{j:(j,i) \in A^k} (X_{ji}^k + F_{ji}^k) = 0 \quad \forall k \in K, \forall i \in N^k \quad (1.21)$$

$$\sum_{j:(j,d(k)) \in A^k} X_{j,d(k)}^k = 1 \quad \forall k \in K \quad (1.22)$$

$$X_{ij}^k (T_i^k + t_{ij}^k - T_j^k) \leq 0 \quad \forall k \in K^c, \forall (i,j) \in A^k \quad (1.23)$$

$$a_i^k \leq T_i^k \leq b_i^k \quad \forall k \in K^c, \forall i \in N^k \quad (1.24)$$

$$X_{ij}^k, F_{ij}^k \geq 0 \text{ et entiers} \quad \forall k \in K, \forall (i,j) \in A^k. \quad (1.25)$$

On reconnaît la forme générale d'un problème de plus court chemin, à la différence additionnelle qu'un transvasement entre les variables de flot X_{ij}^k et F_{ij}^k est possible, ce qui représente le changement de statut d'une locomotive, soit active, soit en voyage à vide. Les contraintes (1.23) et (1.24) qui ajoutent des fenêtres de temps au problème de plus court chemin ne sont considérées que pour les classes de locomotives critiques. Pour notre problème, chaque classe de locomotives est représentée par une commodité. Les locomotives critiques — devant passer à un atelier d'entretien avant un temps donné — et les

locomotives artificielles représentent des commodités distinctes pour chaque classe.

La précision du modèle varie selon son utilité. L'horizon du temps de simulation indique souvent le type de gestion recherchée: planification sur plusieurs mois, gestion stratégique de quelques semaines, application tactique de quelques jours ou réponse opérationnelle en cas de panne. Le modèle de Ziarati [63] emploie des horizons partiels de temps avec un chevauchement entre deux périodes successives. On peut donc adapter la précision temporelle selon le temps de calcul disponible. Un modèle cyclique est souvent utilisé pour représenter la gestion d'une solution périodique alors qu'une gestion opérationnelle requiert les détails précis comme la présence d'une double voie, le temps de retournement d'une locomotive ou la vitesse variable d'un train le long de son trajet. Le détail du modèle ne permet pas toutefois de répondre à ces dernières exigences opérationnelles.

1.2) DÉCOMPOSITION DE DANTZIG-WOLFE

La programmation linéaire a connu son essor avec le développement de la méthode du simplexe par Dantzig [20]. Avec l'aide de Wolfe, il propose la méthode de génération de colonnes, aussi appelée décomposition de Dantzig-Wolfe. Cette technique est reprise par Lasdon [40] qui détaille tout le processus (simplexe phase I, problème maître et sous-problème, génération de colonnes) pour les grands systèmes. La section suivante présente donc le problème maître, les sous-problèmes, le processus de génération de colonnes et l'impact d'une décision de branchement sur ce processus pour le problème d'affectation de

locomotives décrit plus tôt. Une connaissance détaillée de la décomposition de Dantzig-Wolfe est utile pour comprendre comment une tactique de branchement peut s'y intégrer efficacement.

1.2.1) Problème maître

Toutes les contraintes liantes fortes sont incorporées à l'intérieur du problème maître. On obtient le problème maître en remplaçant les variables de flots X_{ij}^k et F_{ij}^k dans les expressions (1.12) - (1.19) par les équations suivantes:

$$X_{ij}^k = \sum_{p \in \Omega^k} x_{ijp}^k \theta_p^k \quad \forall k \in K, \forall (i, j) \in A^k \quad (1.26)$$

$$F_{ij}^k = \sum_{p \in \Omega^k} f_{ijp}^k \theta_p^k \quad \forall k \in K, \forall (i, j) \in A^k \quad (1.27)$$

$$\text{où: } \sum_{p \in \Omega^k} \theta_p^k = n^k \quad \forall k \in K \quad (1.28)$$

$$\theta_p^k \geq 0 \text{ et entiers} \quad \forall k \in K, \forall p \in \Omega^k \quad (1.29)$$

$$X_{ij}^k \geq 0, F_{ij}^k \geq 0 \text{ et entiers} \quad \forall k \in K, \forall (i, j) \in A^k. \quad (1.30)$$

Dans les relations précédentes, Ω^k représente l'ensemble des points extrêmes du domaine réalisable du sous-problème associé à la commodité $k \in K$, et $p \in \Omega^k$ correspond à un chemin élémentaire dans le réseau G^k avec un flot n^k . x_{ijp}^k et f_{ijp}^k sont des coefficients binaires, égaux à l'unité si l'arc $(i, j) \in A^k$ appartient au chemin p , respectivement pour les locomotives actives de classe k et les mises en place de même classe; sinon ils sont nuls. Notez finalement qu'il suffit d'imposer la condition d'intégralité aux flots θ_p^k (1.29) de tous les chemins p pour obtenir l'intégrité de X_{ij}^k et F_{ij}^k (1.30).

De la même façon qu'en (1.12), l'objectif ci-dessous contient les coûts de déplacement des locomotives selon leur statut et les coûts de sous-recouvrement :

$$\min_{\theta, Y} \sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} (x_{ijp}^k c_{ij}^k + f_{ijp}^k d_{ij}^k) \theta_p^k + \sum_{s \in S^n \cup S^p \cup S^q} c_s Y_s \quad (1.31)$$

Les contraintes représentent les restrictions décrites précédemment pour notre modèle :

$$\sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} a_{w,ij}^k x_{ijp}^k \theta_p^k + \sum_{s \in S^n} a_{ws} Y_s = n_w \quad \forall w \in W^n \quad (1.32)$$

$$\sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} a_{w,ij}^k x_{ijp}^k \theta_p^k p^k + \sum_{s \in S^p} a_{ws} Y_s = p_w \quad \forall w \in W^p \quad (1.33)$$

$$\sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} a_{w,ij}^k x_{ijp}^k \theta_p^k q^k + \sum_{s \in S^q} a_{ws} Y_s = q_w \quad \forall w \in W^q \quad (1.34)$$

$$\sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} g_{m,ij}^{kd} x_{ijp}^k \theta_p^k \leq n_m^d \quad \forall m \in M, \forall d \in D \quad (1.35)$$

$$\sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} g_{o,ij}^{kd} x_{ijp}^k \theta_p^k \geq n_o^d \quad \forall o \in O, \forall d \in D \quad (1.36)$$

$$\sum_{k \in K} \sum_{p \in \Omega^k} \sum_{(i,j) \in A^k} g_{h,ij}^k x_{ijp}^k \theta_p^k p^k \geq p_h \quad \forall h \in H \quad (1.37)$$

$$a_s \leq Y_s \leq b_s \quad \forall s \in S^n \cup S^p \cup S^q \quad (1.38)$$

$$\sum_{p \in \Omega^k} \theta_p^k = n^k \quad \forall k \in K \quad (1.39)$$

$$\theta_p^k \geq 0 \text{ et entiers} \quad \forall k \in K, \forall p \in \Omega^k \quad (1.40)$$

Les contraintes de puissance et de tonnage rendent le problème difficile et induisent un saut d'intégrité à la racine fortement supérieur à celui du même problème avec uniquement des contraintes unitaires de type (1.32). En effet, les relations (1.33) et (1.34) sont souvent satisfaites en affectant des fractions de locomotives sur les trains.

1.2.2) Sous-problème

En utilisant toujours la notation de Ziarati [63] présentée en annexe, voici la formulation obtenue pour le sous-problème:

$$\begin{aligned} \min \sum_{(i,j) \in A^k} (c_{ij}^k - \sum_{w \in W^n} a_{w,ij}^k \alpha_w - \sum_{w \in W^p} a_{w,ij}^k p^k \beta_w - \sum_{w \in W^q} a_{w,ij}^k q^k \gamma_w - \\ \sum_{m \in M} \sum_{d \in D} g_{m,ij}^{kd} \delta_m - \sum_{o \in O} \sum_{d \in D} g_{o,ij}^{kd} \sigma_o - \sum_{h \in H} b_{h,ij}^k p^k \varepsilon_h) X_{ij}^k + \sum_{(i,j) \in A^k} d_{ij}^k F_{ij}^k - \sum_{j: (o(k),j) \in A^k} \mu^k X_{o(k),j}^k \end{aligned} \quad (1.41)$$

sous les contraintes (1.20)-(1.25). La fonction sous-objectif (1.41) utilise les variables duales respectives de (1.32)-(1.37) α_w , β_w , γ_w , δ_m , σ_o , ε_h , le coût d'un voyage à vide et la variable duale de (1.39) μ_k . L'origine de cette relation est expliquée dans la prochaine sous-section. On y précise aussi le calcul des variables duales.

1.2.3) Processus de génération de colonnes

D'après le théorème fondamental de la programmation linéaire: pour un problème borné, dont l'objectif est linéaire et le domaine défini par des inégalités linéaires, la frontière du domaine réalisable contient forcément un point extrême solution. Cette propriété est aussi vérifiée lorsque le domaine est défini par un convexe fermé. Pour un domaine défini par des inégalités linéaires, l'algorithme du simplexe permet de trouver la solution optimale. Le détail de cette procédure et certains cas pathologiques (domaine ouvert, non-borné, dégénérescence, problème dual, algorithme dual, objectif non-linéaire, ...) sont décrits dans [28,44]. On considère que le lecteur est familier avec la méthode du simplexe. Le système linéaire des contraintes du problème s'écrit:

$$[A] X = b, \quad X \geq 0. \quad (1.42)$$

Le nombre de points extrêmes d'un modèle linéaire est généralement exponentiel en terme du nombre de variables. Toutefois, dans certains cas, la matrice des contraintes possède une structure particulière. On peut alors réduire le nombre de points à explorer avant de trouver l'optimum. Quelques contraintes peuvent être regroupées pour mettre en évidence plusieurs sous-problèmes de même nature et indépendants. La décomposition de Dantzig-Wolfe [20] exploite cette structure pour restreindre le nombre de points X considérés. La figure 1.2 illustre de façon schématique le problème linéaire dans le cas où 3 sous-problèmes sont présents.

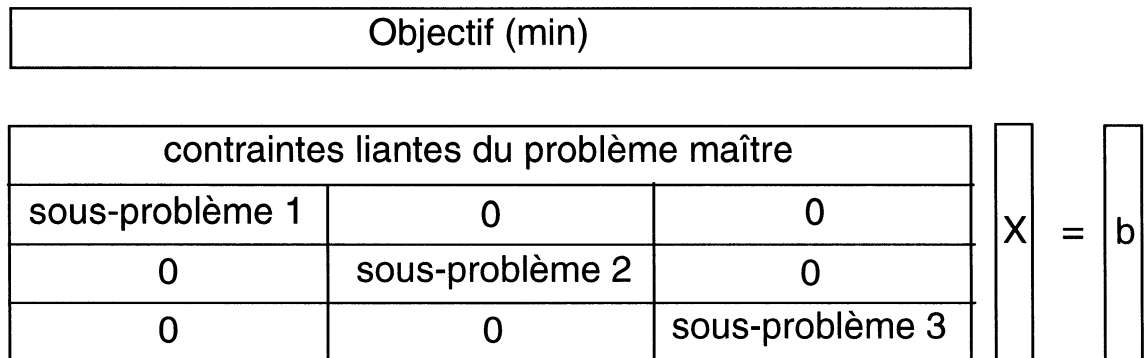


Figure 1.2: Structure d'un problème de génération de colonnes

La sous-matrice $[A_{PM}]$ désigne les coefficients des contraintes liantes du problème maître et chaque sous-matrices $[A_k]$ les coefficients des contraintes des sous-problèmes indépendants. De la même façon, les sous-vecteurs b_{PM} et b_k représentent respectivement les membres de droite des contraintes liantes et ceux des sous-problèmes. Les sous-vecteurs X_k désignent les variables associées à la commodité k . L'équation (1.42) est donc équivalente au système:

$$[\mathbf{A}_{PM}] \mathbf{X} = \mathbf{b}_{PM} \quad (1.43)$$

$$\text{et:} \quad [\mathbf{A}_k] \mathbf{X}_k = \mathbf{b}_k \quad \forall k \in K \quad (1.44)$$

$$\text{où:} \quad \mathbf{X} \geq 0. \quad (1.45)$$

Tel que décrit par Lasdon [40], il est possible d'exprimer toute solution d'un des problèmes (1.44) respectant la contrainte de non-négativité par une combinaison convexe de ses points extrêmes et de ses rayons extrêmes. Toutefois, pour le type de problème que nous considérons, toutes les commodités disposent d'un flot maximal et le domaine est borné. Il suffit donc de ne considérer qu'une combinaison convexe des points extrêmes:

$$\mathbf{X}_k = \sum_j \lambda_{kj} \mathbf{X}_k^j \quad (1.46)$$

$$\text{où:} \quad \sum_j \lambda_{kj} = 1, \quad \lambda_{kj} \geq 0 \quad \forall k \in K. \quad (1.47)$$

En utilisant cette transformation, il est possible d'exprimer le problème maître, constitué des contraintes liantes ainsi que des $|K|$ contraintes de convexité (1.47), en fonction des variables λ_{ij} . Ces variables représentent les poids respectifs de la combinaison convexe permettant de représenter la solution courante du problème maître en fonction des points extrêmes des sous-problèmes indépendants.

A partir d'une solution initiale — obtenue par une phase I du simplexe si nécessaire, le problème maître définit une base $[\mathbf{B}]$ et son vecteur coût $\mathbf{c}_B^T$, puis génère des variables duales $\boldsymbol{\pi}^T$ pour chaque contrainte liante et pour chaque contrainte de convexité représentant un des k sous-problèmes:

$$\boldsymbol{\pi}^T = (\boldsymbol{\pi}_{PM}^T, \boldsymbol{\pi}_0^T) = \mathbf{c}_B^T [\mathbf{B}]^{-1}. \quad (1.48)$$

Ces variables représentent la réduction du coût total provoquée par la diminution unitaire de la variable d'écart des contraintes

correspondantes. Elles sont incorporées dans l'objectif des sous-problèmes (1.44) :

$$\min (\mathbf{c}_k^T - \boldsymbol{\pi}_{PM}^T [\mathbf{A}_k]) \mathbf{X}_k - \pi_{0k} \quad (1.49)$$

où $\mathbf{c}_k^T$ représente le sous-vecteur des coûts associé à tout sous-problème k . Après optimisation, ces sous-problèmes proposent une ou plusieurs colonnes $\mathbf{A}_k^l$ de coût réduit $\bar{f}_k^l$ négatif pour entrer dans la base du problème maître :

$$\bar{f}_k^l = (\mathbf{c}_k^T - \boldsymbol{\pi}_{PM}^T [\mathbf{A}_k]) \mathbf{X}_k^l - \pi_{0k} \leq 0. \quad (1.50)$$

Cet échantillon de colonnes représente des bonnes décisions suggérées par les sous-problèmes au problème maître pour améliorer son objectif. En appliquant l'algorithme du simplexe au problème maître, on réoptimise la solution courante. Puis on réévalue les variables duales $\boldsymbol{\pi}^T$. On répète le tout jusqu'à ce qu'aucune colonne de coût réduit négatif ne soit produite. La solution est alors optimale. Ce processus s'appelle la génération de colonnes et est recommandé pour les systèmes de grande taille [40] dont la structure peut être exploitée. Il est illustré à la figure 1.3.

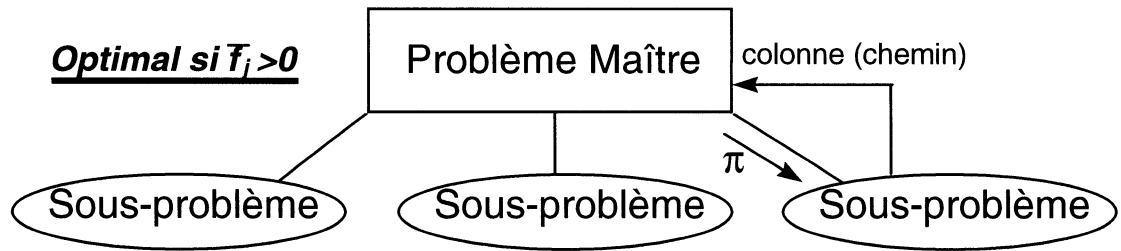


Figure 1.3: Processus de résolution d'un problème de génération de colonnes

A tout moment, les colonnes $\mathbf{A}_k^l$ de la matrice des coefficients des contraintes du problème maître appartiennent à l'un des 3 ensembles

suivants: la base du problème maître, les colonnes générées antérieurement mais actuellement hors-base ou les colonnes non-générées. Ce sont toutes ces colonnes non-générées, parce que finalement dominées, qui permettent de réduire le temps de résolution du problème initial.

On peut aussi accélérer la résolution en renvoyant dans le problème maître toutes les colonnes de coût réduit $\bar{f}_k^l$ négatif trouvées, disjointes si possible. Dans le cas du problème multiflots, chaque commodité induit un sous-problème de type "plus court chemin" sur le réseau. Chaque colonne représente un chemin. On peut facilement identifier des colonnes indépendantes comme des chemins ne recouvrant aucune tâche commune. Pour un même sous-problème cela peut être vérifié en ne renvoyant que les chemins arc-disjoints.

1.2.4) Impact d'une décision de branchement

Puisque les variables d'un problème multiflots correspondent à des flots sur des arcs, il est possible de fixer une variable à une valeur entière sur un arc ou tout au long d'un chemin. Pour bénéficier de la décomposition de Dantzig-Wolfe, il est préférable de prendre des décisions sur les variables de chemins. On conserve ainsi la structure des sous-problèmes, réduisant même parfois la taille du réseau correspondant. Cette idée devra être conservée pour toute approche de branchement, sinon il n'est plus possible d'employer le processus de génération de colonnes de façon itérative à l'intérieur d'un arbre de décision. En général, la prise d'une décision permet aussi de réduire la taille du problème maître en satisfaisant certaines contraintes de

couverture de tâche. Toutefois, ce n'est pas systématiquement le cas pour un problème de type recouvrement généralisé. La décision de fixer une variable de chemin à une valeur entière ne permet pas toujours de réduire la taille du problème maître.

1.3) LA MÉTHODE D'ÉNUMÉRATION IMPLICITE

Contrairement aux problèmes pour lesquels la matrice $[A]$ est unimodulaire, la résolution d'un programme linéaire en variables continues ne produit pas directement une solution en nombres entiers. On ne peut résoudre qu'une relaxation linéaire du problème, en conservant les contraintes de non-négativité mais en retirant les contraintes d'intégralité de (1.40). La section suivante décrit les qualités des deux éléments principaux de la méthode d'énumération implicite: la tactique de branchement qui doit isoler une grande densité de bonnes solutions entières et la stratégie d'exploration qui doit rapidement guider la recherche vers la meilleure solution entière.

1.3.1) Tactiques de branchement

Une bonne tactique de branchement consiste à déterminer sur quelles variables de décisions brancher pour obtenir rapidement ce que l'on cherche. Cette sous-section prend en compte différents aspects qui pourraient permettre d'identifier ces variables de décision, selon le contexte de recherche courant.

La relaxation linéaire génère une première borne optimiste qui permet parfois de rejeter le sous-ensemble courant de l'arbre de branchement avec certitude. On cherche donc à améliorer le calcul de

la borne pessimiste en découvrant rapidement une solution réalisable (entière). Les décisions de branchement doivent facilement s'incorporer dans la technique de résolution par génération de colonnes, pour conserver la même technique de résolution à chaque noeud.

Pour que la méthode d'énumération implicite à grande échelle soit efficace, il faut d'abord éviter toute répétition dans les noeuds représentant des sous-ensembles de solutions. Par exemple, un dédoublement de 10% de l'ensemble des solutions à l'intérieur de deux noeuds-frères sur 20 niveaux implique que chaque solution finale soit représentée 7 fois. Une telle répétition induit des pertes de temps et de mémoire équivalentes. Un branchement sans partition reste fonctionnel, mais offre un mauvais rendement en temps de calcul. Pour une telle stratégie, la programmation dynamique [41] conviendrait mieux.

Le développement d'un noeud introduit deux problèmes: quelles décisions prendre pour générer les noeuds-fils et combien de noeuds générer? On doit remarquer avant tout que plusieurs décisions peuvent être prises entre un noeud-père et un noeud-fils. Dans le contexte des techniques heuristiques de recherche dans un arbre de relaxations linéaires produites par génération de colonnes, la stratégie couramment employée par les algorithmes d'exploration d'un arbre de décisions n'impose une valeur entière qu'à une seule variable à la fois. Balas [10] a montré l'équivalence possible entre une variable entière et une série de variables binaires. Sélectionner une seule variable entière revient donc à fixer plusieurs variables binaires entre deux noeuds. De la même façon, on peut regrouper artificiellement plusieurs

variables entières possédant des bornes supérieures en une seule: sa borne devient égale au produit des bornes précédentes. La prise de décision simple ou multiple n'est donc qu'une différence d'apparence dans la formulation employée. Seule l'implantation informatique d'une décision multiple est plus laborieuse.

Comment peut-on classer ces décisions? Parmi tous les indicateurs, nous en connaissons deux permettant d'ordonner celles-ci et de choisir celles qui doivent être prises. Le coût réduit d'une variable et la variable duale d'une contrainte constituent des mesures du rythme actuel de détérioration de l'objectif. L'un approxime la détérioration par variation unitaire de la variable, l'autre par variation unitaire du membre de droite de la contrainte. Les produits du coût réduit d'une variable par sa partie fractionnaire et celui de la variable duale d'une contrainte par son gaspillage — ou dépassement du membre de droite — fournissent donc des limites minimales de la détérioration — ou pénalité — engendrée sur l'objectif par une décision. Lorsqu'une branche concerne plusieurs variables différentes, ces paramètres permettent d'ordonner les noeuds intermédiaires non-représentés. Un noeud-fils représente ces décisions et son ou ses frères devrai(en)t représenter les autres possibilités sans répétitions.

En même temps que l'on branche sur les variables, il est possible de gérer dynamiquement l'ensemble des contraintes. Trois aspects sont visés: retirer des contraintes désuètes — déjà satisfaites, modifier le membre de droite des contraintes selon le nouveau domaine des possibilités et ajouter des coupes, parfois même heuristiques, pour obtenir rapidement une solution entière. Les réductions de la taille

des problèmes engendrées par la gestion dynamique des contraintes pourraient influencer la sélection des variables de décision.

La première étape est nécessaire pour réduire la taille de la relaxation linéaire et accélérer l'évaluation des noeuds à chaque génération. On peut retirer une contrainte qui sera satisfaite pour tout noeud-fils, ou négliger un ensemble de contraintes faibles, qui devront être vérifiées après chaque résolution et réintroduite une par une en cas de violation. Ce dernier cas est appliqué avec succès aux contraintes peu restreignantes de notre problème d'affectation de locomotives. Le réajustement du membre de droite avec la coupe entière réalisable la plus profonde est une amélioration simpliste de l'enveloppe réalisable du problème. On se contente de l'implanter à la racine de l'arbre.

La dernière stratégie provient de l'idée générale de trouver l'enveloppe convexe du domaine réalisable. De nombreuses classes de facettes sont maintenant générées par analyse polyédrale pour divers problèmes. Ziarati [63] a montré que certaines coupes sont des facettes du problème initial, lorsqu'uniquement deux types de locomotives sont disponibles pour construire un convoi. Généralement, plusieurs classes sont disponibles même si la majorité des convois ne contiennent finalement que deux types. Les coupes "2-consist", appliquées aux deux types de locomotives les plus utilisés par la solution en variables continues, constituent donc un outil heuristique intéressant pour obtenir rapidement une solution entière.

1.3.2) Stratégies d'exploration

Malgré la qualité des tactiques de branchement évoquées précédemment, explorer un arbre de recherche dans sa totalité reste impossible avec les tailles des problèmes étudiés actuellement. Une stratégie d'exploration d'un arbre gigantesque semble donc toujours heuristique dans le sens où elle rejette arbitrairement des sous-ensembles de solutions peu prometteurs lorsque le temps de calcul permis s'achève.

La méthode généralement utilisée est une amélioration du vieux réflexe d'essai et d'erreur: l'énumération implicite. Une bonne stratégie d'exploration garantissant la découverte d'une bonne solution réalisable — entière — le plus rapidement possible, le calcul d'une borne permet alors de rejeter certains sous-ensembles avec certitude. Lorsque les décisions précédentes mènent à un cul-de-sac, il devient nécessaire d'effectuer un retour-arrière — ou *backtracking*. La résolution d'un noeud peut mener à divers résultats disjoints: une solution non-réalisable, une solution non-entière qui peut avoir du potentiel ou être sans espoir, une solution entière moins bonne ou meilleure que la solution entière courante. Le reste de l'arbre est composé de noeuds non-explorés. Les deux techniques suivantes constituent le fondement des stratégies d'exploration: les stratégies "meilleur-d'abord" et "profondeur-d'abord" — ou respectivement *best-first* et *depth-first*.

Avec la technique meilleur-d'abord, selon un critère d'évaluation, on développe systématiquement le meilleur candidat non-exploré. Cette stratégie repose fortement sur la qualité du critère

utilisé. Malheureusement, au fur et à mesure que l'on fixe des variables, on provoque une détérioration de l'objectif. Pour cette raison, l'approche meilleur-d'abord a tendance à osciller entre toutes les décisions favorables. La mémoire utilisée et le temps de découverte d'une première solution entière augmentent donc vertigineusement pour les grands systèmes. Cette stratégie reste toutefois très bonne pour trouver l'optimum de petits et moyens systèmes.

Avec la technique profondeur-d'abord, on développe systématiquement le dernier candidat créé. Cette stratégie balaie l'arbre de gauche à droite. Malheureusement, aucune information ne provient de la qualité de l'objectif précédent. Pour cette raison l'approche profondeur-d'abord a tendance à traverser bêtement l'arbre. La mémoire utilisée est relativement faible mais la qualité des résultats dépend fortement de la qualité du branchement initial. Cette stratégie est très efficace pour obtenir rapidement une solution réalisable.

La littérature a déjà proposé de nombreuses manières de combiner ces stratégies. Par exemple, on recommande au moins d'employer l'approche en profondeur jusqu'à l'obtention d'une solution pour élaguer un minimum l'arbre, avant de passer à la stratégie meilleur-d'abord. La clef du succès semble résider dans le critère qui guide l'orientation de l'exploration.

CHAPITRE 2

RECHERCHE BIBLIOGRAPHIQUE

Deux approches étaient généralement considérées pour la résolution de problèmes en nombres entiers: l'énumération implicite — *branch-and-bound* — et la génération de l'enveloppe convexe par coupes. Ces méthodes ont ensuite été mélangées pour produire une approche hybride d'énumération avec coupes — *branch-and-cut*. Finalement, ces dernières années, l'imbrication d'un processus de génération de colonnes à l'intérieur d'un arbre de recherche en nombres entiers [11] — *branch-and-price* — est devenu plus courant. Ce second chapitre résume les publications effectuées dans les domaines de résolution en nombres entiers, en insistant sur les tactiques de branchement et les stratégies d'exploration.

2.1) APPROCHES DE RÉOLUTION EN NOMBRES ENTIERS

À l'origine un problème en nombres entiers peut varier de nature: on cherche simplement une solution réalisable — il n'y a pas d'objectif, le respect des contraintes est suffisant — ou le système de contraintes est exprimé sous forme d'inégalités. On corrige alors la définition du problème en effectuant une phase I du simplexe ou en ajoutant des variables d'écart ou de surplus au système [20,28]. L'ajout de dimensions supplémentaires permet de retrouver la formulation d'un problème linéaire classique avec contraintes d'intégralité.

Land et Doig [39] introduisent une méthode systématique de programmation avec des variables entières en 1960. L'arbre de

décisions est émondé par la meilleure solution réalisable connue. Par la suite, la méthode d'énumération implicite — ou d'énumération et élimination — est appliquée à toutes sortes de problèmes. On peut trouver des exemples dans McMillan [46], Gondran et Minoux [30], Alj et Faure [5], Al-Khayyal [4], Nemhauser et Wolsey [49], et d'innombrables publications subséquentes... On améliorera aussi la borne de cette technique pour accélérer l'élagage, réduisant ainsi la taille d'un arbre à des dimensions raisonnables. Pospichal et Kvasnicka [54] emploient cette méthode pour énumérer des molécules: ils insistent sur la non-répétition des états étudiés pour identifier la combinaison cherchée.

Rapidement, on observe une autre façon d'obtenir une solution entière par la résolution de problèmes de réseaux. La matrice unimodulaire de ces problèmes contient forcément une solution optimale entière, tel que montré dans Gondran et Minoux [30], Nemhauser et Wolsey [49], et Ahuja et al. [3]. Le problème du plus court chemin, avec ou sans fenêtres de temps [23], appartient à la famille des problèmes qui possèdent une solution optimale entière. Ces problèmes sont souvent employés comme sous-problème. Par exemple, Miaou et Chin [47] et Charland [14] emploient le problème des "k plus courts chemins" pour générer plusieurs colonnes simultanément.

Desrosiers et al. [23] illustrent diverses applications du problème de flot avec multi-commodités et contraintes de ressources. Ils montrent aussi comment ajouter quelques termes permettant de resserrer la borne optimiste dans certains cas. On y explique aussi la généralisation de la notion de fenêtre de temps à celle de ressource.

Le livre de Nemhauser et Wolsey [49] résume l'autre approche pour la résolution de problèmes en nombres entiers. Il traite d'un sujet fondamental de la théorie polyédrale: la classification des inégalités valides en facettes pour des familles de problèmes NP-complets. L'ajout de coupes pour cerner l'enveloppe convexe devient un axe de recherche fondamental, poursuivant l'idée d'introduire des coupes entières de Gomory [29]. Nemhauser et Sigismondi [48] illustrent comment générer des familles de coupes et soulever — *lifter* — des inégalités valides pour cerner l'enveloppe convexe.

Les procédures de *lifting* constituent la composante essentielle d'une méthode d'énumération implicite avec plans coupants. Toute coupe introduite à un noeud de branchement doit être soulevée pour contribuer à la définition du domaine convexe du problème initial. Hoffman et Padberg [37] résument ainsi les quatre composantes d'un algorithme d'énumération implicite avec plans coupants: un pré-processeur qui resserre la définition du problème en modifiant les membres de droites et en retirant les contraintes redondantes; une heuristique permettant de trouver une bonne solution entière rapidement; un générateur de coupes pour cerner l'enveloppe convexe initiale; et finalement, une tactique de branchement permettant d'explorer des solutions éloignées. Ce dernier concept ressemble à l'idée de vallées de solutions différentes exposée plus tard. Un autre exemple de cette méthode est présenté par Bixby et Lee [13] dans le cadre du transport par camions.

Un dernier élément qui n'est pas directement relié au branchement est à noter: la gestion dynamique des contraintes. Pour diminuer la taille des relaxations linéaires, on retire les contraintes déjà satisfaites. Certains algorithmes proposent d'enlever des contraintes

dans le même but de réduction de taille, sans voir leur satisfaction garantie. Il devient nécessaire de rajouter ces contraintes violées finalement pour retrouver une solution réalisable comme lors d'une méthode primale-duale. Pour rajouter ces contraintes et s'assurer de ne pas cycler, la suite des objectifs, après chaque ajout de contraintes intermédiaires, doit être strictement croissante — en cas de minimisation de l'objectif. Si cette croissance n'est pas stricte, on recommande une règle anti-cyclage rigoureuse — Jaumard et al. [38]. Un branchement permettant d'écarter de nombreuses contraintes, même temporairement, est à considérer.

2.2) LES TACTIQUES DE BRANCHEMENT

Little et al. [42] appliquent la méthode d'énumération implicite aux problèmes de tournées. Ils recommandent l'emploi d'un arbre débalancé, avec l'exploration du noeud-fils prometteur. Ils sont cités par Lawler et Wood [41] dans leur résumé des travaux précédents en nombres entiers. Ces derniers rapportent leur choix de la variable de branchement x_i : celle où $x_i = 0$ provoque la pire détérioration de la borne; en explorant $x_i = 1$, on a l'avantage d'avoir réduit la taille du problème. Les qualités d'un branchement asymétrique sont toujours débattues de nos jours.

Dans le même genre, Taha [57] propose de choisir une variable de branchement qui diminue rapidement la taille du problème, en commençant par s'intéresser aux variables avec peu de liberté où la détérioration est extrême. Il recommande de choisir la variable x_s telle que:

$$s \in \underset{i}{argmax} \{ \min [d_i f_i, u_i (1 - f_i)] \} \quad (2.1)$$

où f_i, d_i et u_i représentent respectivement la partie fractionnaire des composantes $\bar{x}_i$ au noeud-parent et les estimations de coût unitaire pour diminuer $\bar{x}_i$ à $\lfloor \bar{x}_i \rfloor$ et augmenter $\bar{x}_i$ à $\lceil \bar{x}_i \rceil$. Le choix binaire de $x_i = \lfloor \bar{x}_i \rfloor$ ou $x_i = \lceil \bar{x}_i \rceil$ comme prochain noeud d'exploration dépend du terme minimal en (2.1). Typiquement, une méthode de pénalité correspond à cette description et s'avère très efficace pour de petits problèmes. Tomlin [58] propose un raffinement intéressant des pénalités associées au respect des contraintes d'intégralité lorsqu'on ne peut pas profiter d'une structure de coût particulière au problème.

La véritable percée en matière de branchement découle de l'article de Balas [10], décrite aussi par Alj et Faure [6], sans avoir été mise en évidence par ces auteurs. Le branchement multiple permet de fixer plusieurs variables entre un noeud-père et un noeud-fils. Bien sûr, ce type de branchement est heuristique si on ne peut pas modifier ce branchement plus tard, en isolant les décisions actuellement agglomérées en une seule branche. Yu et Yuan [62] exploitent ce type de décisions pour la reconnaissance de formes. Tseng et al. [59] l'incorporent par la suite dans une étude plus complète sur l'assemblage de fermes dans la construction de ponts. Ils permettent de nombreux fils, des décisions multiples et cinq critères différents de branchement: distance fractionnaire minimale ou maximale, différence minimale ou maximale entre les distances fractionnaires et différence de coût maximale — ou pénalité. Plus la dimension Nb du voisinage considéré est grande, plus la qualité des solutions peut s'améliorer:

$$\lfloor \bar{x}_s \rfloor - Nb \leq x_s \leq \lceil \bar{x}_s \rceil + Nb. \quad (2.2)$$

Encore une fois, la méthode de pénalité donne les meilleurs résultats. Enfin, l'utilisation de multiples décisions accélère le temps de calcul

mais détériore légèrement la qualité de la solution finale. Ce comportement du branchement multiple est confirmé par les résultats de Gamache et al. [26]. Ils obtiennent une réduction du temps de calcul par un facteur de 7,5 mais constatent aussi une détérioration relative non-négligeable du saut d'intégrité. En effet, les effets combinés de plusieurs décisions sont parfois mauvais et l'on ne peut plus les dissocier. Cet aspect sera développé en détail dans le chapitre 6.

2.3) LES STRATÉGIES D'EXPLORATION

Lawler et Wood [41] résument les stratégies d'exploration classiques: largeur-d'abord et profondeur-d'abord. Ces stratégies cèdent rapidement la place à la stratégie meilleur-d'abord: le choix du critère de sélection de cette stratégie devient l'intérêt principal. Les notions de sous-optimalité et de retour-arrière font partie de leur résumé. Une solution est appelée Δ sous-optimale si la valeur de son objectif détériore celle de la solution optimale d'un terme Δ au plus.

La littérature plus avancée concernant les algorithmes de recherche provient principalement de la communauté scientifique en intelligence artificielle. Harris [35] et Pearl [52] traitent des arbres minmax utilisés pour les jeux de dames ou d'échecs. La stratégie alpha-beta définit à chaque noeud deux bornes supplémentaires au-delà desquelles toute poursuite des calculs est sans effet. Même si les arbres minmax ne correspondent pas au modèle que nous traitons, certains concepts de leurs stratégies d'exploration sont récupérables: notamment la technique d'Anantharaman et al. [7] qui appliquent avec succès au jeu d'échecs le développement d'une ligne prometteuse. Cette

idée est voisine de la technique de coup de sonde présentée dans la section 4.1.2.

Un algorithme heuristique provient généralement de la relaxation de contraintes du problème initial. Pour un problème d'énumération implicite, le critère est normalement constitué d'une détérioration passée g entre le noeud-racine et le noeud courant, et d'une détérioration future h . Harris [35] met en évidence cette forme générale du critère d'exploration: $\hat{f} \approx g + h$. Une heuristique admissible est systématiquement optimiste: pour une minimisation, $\hat{h} \leq h$ où $\hat{h}$ représente une prévision de h . Un algorithme heuristique est appelé ϵ -admissible si:

$$\hat{h}(n) \leq h(n) + \epsilon . \quad (2.3)$$

De la même façon, un algorithme heuristique est dit $\mathcal{E}$ -optimal si:

$$h(n) - \epsilon \leq \hat{h}(n) \leq h(n) + \epsilon . \quad (2.4)$$

Pohl [53] confirme que les heuristiques constituent la meilleure stratégie pour contrer l'explosion combinatoire, en précisant qu'elles gagnent en efficacité lorsqu'on adapte leur précision localement. Elles excellent près de la solution.

Dans leur livre, Nemhauser et Wolsey [49] énoncent aussi l'existence d'un algorithme polynomial pour la recherche de la faisabilité de certains problèmes combinatoires. En fait, cela n'est possible que pour des classes d'heuristiques qui permettent de maintenir une relation d'ordre. Bagchi et Mahanti [9] définissent les types d'heuristiques qui répondent à ce critère: non-déroutante ou propre. Soient m et n deux noeuds dans le graphe de recherche. Soient P_1 et P_2 deux chemins représentant les décisions menant respectivement du noeud-racine aux noeuds m et n . Soient finalement

$c(P_1)$ et $c(P_2)$ les détériorations de l'objectif accumulées le long de P_1 et P_2 . Une heuristique est dite non-déroutante — *non-misleading* — si :

$$c(P_1) + \hat{h}(m) < c(P_2) + \hat{h}(n) \Rightarrow c(P_1) + h(m) \leq c(P_2) + h(n) \quad \forall (m, n). \quad (2.5)$$

Une heuristique est dite propre si la condition (2.5) est vérifiée pour tous chemins P_1 et P_2 non sous-chemin l'un de l'autre. Dans les deux cas, le temps de recherche peut être ramené à un ordre polynomial $O(x^2)$ où x est le nombre de variables. En outre si l'heuristique est consistante, ou du moins admissible, le coût des décisions de la solution réalisable est minimal. Une heuristique est dite consistante si $\hat{h}(m) - \hat{h}(n) \leq k(m, n)$, $\forall (m, n)$ où $k(m, n)$ représente le coût additionnel sur la branche (m, n) . Malheureusement, les heuristiques disponibles pour prévoir les coûts supplémentaires dus à la satisfaction des contraintes liantes du problème multiflots ne conservent pas de relation d'ordre.

Cependant, on peut essayer de corriger leurs erreurs. C'est Pearl [52] qui formule la correction lorsqu'on connaît le support supérieur r_i des erreurs relatives induites par chaque heuristique $\hat{h}_i$. Le support supérieur r_i représente l'erreur maximale de surestimation par l'heuristique $\hat{h}_i$. En outre, Pearl [52] résume les caractéristiques de complétude, d'admissibilité et de dominance des heuristiques. Le principe "petit-rapide" est aussi mis en évidence. L'étude de l'accélération de la résolution par $f = (1-w)g + wh$ montre que $1/2 \leq w \leq 1$, avec des préférences à $w=1$ pour le problème de faisabilité et à $w=1/2$ lorsque le but est d'atteindre un objectif optimal. Finalement, il montre comment combiner différentes heuristiques lorsqu'elles induisent des erreurs proportionnelles :

$$\hat{h} = \max_i \left\{ \hat{h}_i / (1 + r_i) \right\}. \quad (2.6)$$

Haralick et Elliot [34] résument ainsi le principe petit-rapide: essayer d'abord ce qui devrait échouer rapidement. Ils ajoutent les conseils suivants: regarder et anticiper le futur pour réussir dans les décisions présentes, garder en mémoire le passé pour éviter de répéter les mêmes erreurs et regarder l'avenir sans se soucier du passé pour le problème de faisabilité. Gashnig [27] met en pratique le second principe en introduisant le *backmarking*. L'idée consiste à indiquer, lorsque c'est possible, quelle décision empêche un sous-ensemble de décisions futures; lorsqu'une impasse est atteinte, au lieu d'effectuer un retour-arrière usuel, on retourne directement au noeud-aïeul sain.

La meilleure étude générale de l'exploration avec objectif est proposée par Taha [57]. Il considère trois critères de sélection du noeud. D'abord celui de la meilleure projection:

$$Z_{esp} = Z_{lr} - \lambda S \quad (2.7)$$

$$\text{où:} \quad S = \sum_i \min(f_i, 1 - f_i) \quad (2.8)$$

$$\text{et} \quad \lambda = (Z_0 - Z_c^*) / S_0 \quad (2.9)$$

avec la notation suivante:

Z_{esp} : valeur espérée de l'objectif en poursuivant la recherche à partir du noeud courant v ayant pour solution $\bar{\mathbf{x}}^v$;

Z_{lr} : objectif de la relaxation linéaire ou borne optimiste;

λ : taux de détérioration unitaire de l'objectif;

S : "distance fractionnaire" à parcourir;

f_i : partie fractionnaire de la $i^{\text{ème}}$ composante de $\bar{\mathbf{x}}^v$;

Z_0 : objectif relaxé obtenu à la racine de l'arbre;

Z_c^* : meilleure solution entière courante ou borne pessimiste;

S_0 : distance fractionnaire initiale.

Le second critère est comparable mais emploie différentes évaluations de λ . Les nouveaux pseudo-coûts sont:

$$d_i = (Z^c - Z^{c1}) / f_i \quad (2.10)$$

$$u_i = (Z^{c'} - Z^{c'2}) / (1 - f_i) \quad (2.11)$$

$$\text{avec:} \quad Z_{esp} = Z_{lr} - \sum_i \min\{u_i f_i, d_i(1 - f_i)\}. \quad (2.12)$$

Les objectifs relaxés Z^c , Z^{c1} , $Z^{c'}$ et $Z^{c'2}$ sont ceux des points le long de la meilleure solution entière courante tels que x_i a été fixée à sa borne inférieure entre les noeuds c et $c1$, et x_i à sa borne supérieure entre les noeuds c' et $c'2$. Le dernier critère exploite les valeurs de l'objectif espéré, en tenant compte en plus de la distance à la borne pessimiste. Le pourcentage d'erreur tient donc compte de la proximité de la solution. On cherche la valeur maximale suivante, la division du gain sur la borne par la détérioration maximale jusqu'à la borne:

$$\eta = 100(Z_c^* - Z_{esp}) / (Z_{lr} - Z_c^*). \quad (2.13)$$

Ces critères permettent généralement d'identifier un seul noeud suivant à explorer. Mans et Roucairol [45] ont toutefois étudié les effets de différents choix de bris d'égalité pour la technique meilleur-d'abord, appliqué à la programmation sur des processeurs en parallèle. Rosing et Revelle [55] exploitent aussi l'obtention de différentes solutions parallèles. Ils développent un algorithme de concentration d'heuristique à deux étages, pour identifier une meilleure solution contenant des décisions communes. En comparant plusieurs solutions sous-optimales obtenues heuristiquement, ils proposent d'étudier tous les noeuds du sous-arbre suivant: toutes les

décisions communes à toutes les heuristiques sont fixées et on explore le sous-ensemble de combinaisons des décisions rencontrées dans au moins une solution heuristique. On espère obtenir ainsi une meilleure solution proche des précédentes.

2.4) LE PROBLÈME D'AFFECTION DE LOCOMOTIVES

En 1957, Bartlett et Charnes [12] furent parmi les premiers à étudier le problème d'affectation de locomotives pour tirer des trains. Toutefois, c'est Florian et al. [25] qui formulèrent l'approche en terme de réseau multicommodité que nous employons. Ils utilisèrent une méthode de partitionnement de Benders pour résoudre ce problème sur un horizon d'une semaine en considérant 200 segments de trains: ils ont ainsi obtenus de bons résultats quant à la planification stratégique de la flotte nécessaire. Par la suite, Smith et Sheffi [56] introduisirent des fonctions convexes de pénalité de sous-recouvrement ce qui génère des modèles plus réalistes. Ils ont résolu leur problème sur un horizon de 3 jours jusqu'à concurrence de 102 segments de trains grâce à un algorithme heuristique efficace, mais cette méthode est difficilement applicable dès que plus de deux commodités sont employées car la complexité de la phase d'échange — *interchange* — croît rapidement.

Le problème de planification cyclique des locomotives a été traité par Chih et al. [15], et dernièrement par Ahuja et al. [2] sur un horizon d'une semaine. Leur version du problème se souciait de minimiser les réarrangements d'assemblages de locomotives aux PCP, mais ne considérait pas les contraintes de maintenance. Leur approche en deux temps déterminait tout d'abord les connections possibles entre

segments de train d'une même journée, puis utilisait ces solutions journalières pour bâtir une solution hebdomadaire. A chaque étape, ils utilisaient le logiciel CPLEX pour résoudre un problème d'optimisation à variables mixtes — ou *MIP* — et essayaient d'améliorer la solution entière obtenue en explorant à grande échelle son voisinage. Cette approche leur a permis d'améliorer fortement les solutions manuelles obtenues par *CSX Transportation Inc.* pour des problèmes industriels utilisant 3 316 locomotives de cinq types différents et comportant 3 324 trains.

Après les méthodes de partitionnement et de substitution, apparaît l'approche par génération de colonnes imbriquée dans un arbre d'énumération implicite, ce qui permet une exploration systématique du domaine des solutions discrètes. Ziarati et al. [64] illustrent l'efficacité de cette méthode en l'appliquant au problème d'affectation de locomotives. Ils obtiennent un saut d'intégrité moyen autour de 5,5% pour des problèmes sur un horizon d'une semaine considérant 1500 segments de trains. Un autre exemple de cette méthode est présenté par Nou et al. [51]. En résumé, plusieurs façons d'évaluer la fonction objectif sont considérées, prenant en compte les retards et les pénalités de sous-recouvrement au lieu de locomotives artificielles — louées par un concurrent. Ziarati et al. [64] présentent une réduction moyenne de 2% du saut d'intégrité avec pénalités de sous-recouvrement lorsque des retards sont introduits pour permettre l'ajout de locomotives à un train sous-couvert. La taille et le chevauchement de fenêtres d'optimisation intermédiaires permettant de représenter l'horizon de temps par morceaux y sont aussi étudiés: plus la fenêtre d'optimisation ou le chevauchement entre ces fenêtres sont grands, plus l'optimisation est efficace. Naturellement, la mémoire nécessaire et

le temps *CPU* de résolution augmentent en proportion. Le détail de cette étude provient de la thèse de Ziarati [63]. Avec ces améliorations, le saut d'intégrité diminue typiquement de 11,2% obtenu par les solutions appliquées dans l'industrie à 4,2% avec les méthodes de Ziarati. Nous poursuivons cette approche.

Les coupes "2 types de locomotives par *consist*" — aussi appelées coupes 2-*consist* — peuvent être appliquées aux deux types de locomotives principalement utilisées par la solution en variables continues. Ziarati et al. [65] obtiennent ainsi de plus petits sauts d'intégrité lorsque la solution finale contient des convois peu diversifiés — en moyenne 2,2 types de locomotives par convoi. Pour ce genre de problèmes, les coupes 2-*consist* constituent un outil heuristique intéressant pour obtenir rapidement une solution entière. Toutefois les solutions entières ne sont que légèrement meilleures pour des scénarios où plusieurs fenêtres d'optimisation se chevauchent. Par ailleurs, Ziarati [63] a démontré que ces coupes sont des facettes du problème initial lorsque seulement deux types de locomotives sont disponibles pour bâtir le convoi.

L'affectation simultanée de locomotives et de wagons est une généralisation du problème que nous étudions. En utilisant une décomposition de Benders, Cordeau et al. [16-18] présentent des exemples d'application à VIA pour des trains de passagers avec une demande de locomotives exprimée en nombre d'unités (1 ou 2). Un modèle permettant de combiner à une moindre échelle jusqu'à deux assemblages de wagons lors d'un même déplacement est aussi disponible dans Lübbecke et Zimmermann [43]. Plus de détails à propos de publications dans le domaine du rail sont répertoriés dans Cordeau et al. [19].

CHAPITRE 3

TACTIQUES DE BRANCHEMENT AVEC CHEMINS INTERDITS

Le problème d'affectation de locomotives pour le transport de marchandises est un problème d'optimisation de grande taille qui doit être résolu en nombres entiers. Il s'agit d'une application particulière du modèle multiflots avec contraintes additionnelles en nombres entiers et les méthodes suggérées s'appliquent à l'ensemble de cette classe de problèmes. Plus précisément, bien que la décomposition de Dantzig-Wolfe puisse s'appliquer en utilisant des sous-problèmes quelconques, on suppose ici que la structure bloc angulaire — *c.f.* figure 1.1 — se compose de sous-problèmes indépendants de type plus court chemin avec ressource(s). En exploitant cette structure particulière, la technique de résolution isole ces sous-problèmes par commodité distincte pour résoudre le problème maître par génération de colonnes.

Dans le cadre du problème d'affectation de locomotives pour le transport de marchandises, on définit une commodité par type de locomotives et par locomotive critique. Ceci correspond à agréger les commodités correspondant à des locomotives de même type. Cette agrégation est possible car les arcs qui sont associés aux locomotives d'un même type sont équivalents, c'est-à-dire qu'ils possèdent les mêmes caractéristiques: un même coût, une paire de noeuds incidents identiques, une consommation de ressource identique, etc. En général, cette agrégation permet d'accélérer les temps de résolution en diminuant le nombre de sous-problèmes à résoudre et le nombre de colonnes suggérées au problème maître ainsi qu'en évitant les branchements en cascades. Cette dernière difficulté survient lorsque

l'interdiction d'un chemin à une locomotive engendre l'utilisation d'une autre locomotive identique. En interdisant le chemin à toutes ces locomotives regroupées en une même commodité, des remplacements équivalents sont évités.

Lors du processus d'énumération implicite, le type de décisions de branchement doit permettre de préserver autant que possible la structure du modèle à chaque noeud de l'arbre d'énumération. Dans un contexte de décomposition de Dantzig-Wolfe, la structure bloc angulaire du problème maître et des sous-problèmes doit donc être conservée. Plusieurs types de branchement ont été suggérés par le passé. Pour les problèmes de type partitionnement d'ensemble, apparaissant par exemple en transport aérien, la génération des noeuds-fils lors du processus d'énumération s'effectue souvent à l'aide de variables dites d'inter-tâche. Malheureusement, cette sorte de branchement ne convient pas aux problèmes de type recouvrement généralisé où plusieurs unités de flot sont nécessaires pour accomplir une tâche, que ce soit en distribution par camion — *truck load* [13], pour la construction d'horaires d'agents de bord [21,23,36,37] ou dans le domaine ferroviaire [8,19,63-65]. Plus généralement, tout branchement sur des arcs entre en conflit avec l'agrégation d'un problème de recouvrement généralisé: une décision d'inter-tâche ou de flot sur un arc ne peut pas être imposée à une seule locomotive de la commodité sans détruire la structure du modèle. Une autre option est applicable mais limitée: un branchement fondé sur les ressources peut être envisagée en ce qui concerne les locomotives critiques. Finalement, l'utilisation d'un branchement sur les variables de chemin semble la seule voie qui permette de conserver efficacement la structure du modèle mathématique, en mettant simplement à jour les membres de droite du système de contraintes. En outre, le

système d'équations décrit en (1.31)-(1.40) est compact et emploie une commodité par type de locomotive en accord avec l'agrégation.

C'est dans ce contexte que l'on se concentre sur le développement de stratégies permettant de trouver rapidement une bonne solution entière à un problème de grande taille. On définit dans cette étude un problème de grande taille tel qu'il ne peut pas être exploré totalement par énumération implicite, même en ayant initialement une solution entière optimale. Nous proposons une méthode de branchement efficace en utilisant les deux éléments suivants: une prise de décisions multiples à chaque noeud pour accélérer l'obtention d'une solution entière et une représentation de chemins interdits pour partitionner l'ensemble des solutions. En appliquant ces deux principes, la technique de recherche heuristique obtenue est de type systématique incomplète. "Systématique" car chaque noeud-feuille représente une solution distincte, et "incomplète" car, par définition, un problème de grande taille ne peut pas être exploré totalement dans le temps imparti. Les sections suivantes présentent ces deux éléments, différentes façons de les combiner et des résultats numériques issus de ces combinaisons.

3.1) PRISE DE DÉCISIONS MULTIPLES

Bien que l'analyse qui suit s'applique aussi au cas non binaire, les figures montrent généralement un branchement binaire pour ne pas surcharger la représentation graphique. La figure 3.1 représente un exemple d'arbre de branchement binaire typique. La légende énumère les états potentiels des noeuds de l'arbre. Les noeuds représentés par des cercles possèdent une solution optimale fractionnaire, ceux représentés

celui de la figure 3.1 mais contient plusieurs noeuds-fils et des décisions multiples. L'équivalence est illustrée à la figure 3.3 pour un noeud-père noté v . Elle est aussi valable lorsque l'arbre n'est pas binaire.

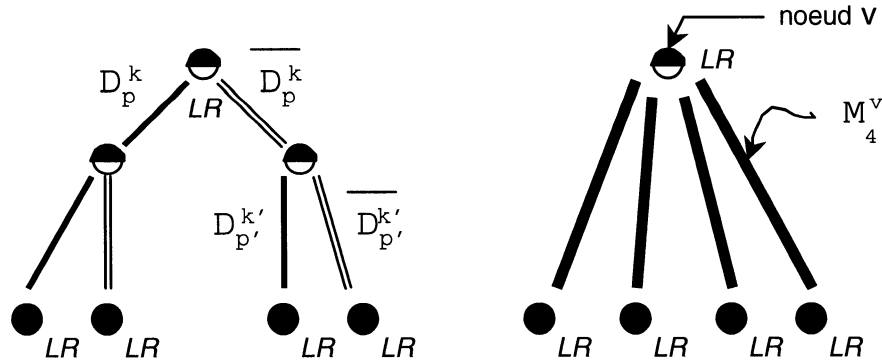


Figure 3.3: Equivalence entre décisions élémentaires et multiples

La représentation graphique de droite, qui omet les noeuds intermédiaires, sera parfois employée avec des traits plus large pour désigner les décisions multiples. La χ -ième décision multiple au noeud v , notée M_χ^v , est donc constituée d'un ensemble de décisions élémentaires — de type (3.1) — entre deux noeuds (père-fils) du nouvel arbre d'énumération T . Dans l'exemple ci-dessus, la quatrième décision au noeud v se décrit par:

$$M_4^v = \{ \overline{D}_p^k, \overline{D}_{p'}^{k'} \}. \quad (3.2)$$

Le report de la résolution des relaxations linéaires rencontrées aux noeuds intermédiaires permet de réduire la profondeur de l'arbre et d'économiser du temps de calcul. Toutefois, l'ensemble $\hat{M}_\chi^v$ des décisions élémentaires additionnelles de M_χ^v — i.e. toutes les décisions de M_χ^v sauf celle qui aurait été retenue si la décision était unique — peut en être affecté. Ainsi, le processus d'exploration est accéléré à condition de pouvoir trouver quand même

une solution entière ε -optimale — c.f. section 2.3 — en un temps relativement court. Pour éviter d'éliminer une solution entière ε -optimale, il est recommandé de limiter le nombre n_χ^v de décisions élémentaires fixées par une décision multiple M_χ^v par une borne $n_{\max}$ qui soit inférieure ou égale au nombre de commodités:

$$n_\chi^v \leq n_{\max} \leq |K|. \quad (3.3)$$

Sinon, il y aurait forcément au moins deux décisions élémentaires prises pour une même commodité sans tenir compte de l'effet de leur combinaison. Cette accélération ne permet toutefois pas d'espérer l'énumération complète d'un arbre de branchement typique. L'équivalence théorique entre décisions multiples binaires et décision unique entière ayant déjà été mise en évidence par Balas [10], aucun désavantage n'est introduit par cette technique si ce n'est une implantation informatique légèrement plus complexe.

Tous les noeuds-fils pourraient être générés, mais en pratique une fraction des noeuds de droite est rejetée — par convention on appelle "noeud de gauche" le premier noeud-fils exploré et "noeuds de droite" ses noeuds-frères. La figure 3.4 montre un branchement multiple contenant trois décisions élémentaires avec rejet d'une fraction des noeuds de droite.

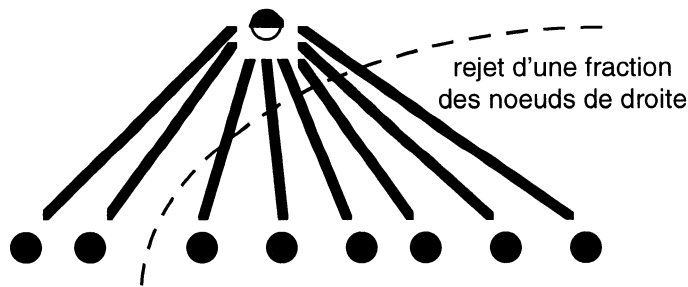


Figure 3.4: Représentation schématique du rejet des noeuds de droite

La grande taille des problèmes considérés justifie le caractère incomplet de l'exploration de l'arbre des possibilités. En effet, un problème typique peut nécessiter plusieurs années de calcul, ne serait-ce que pour prouver l'optimalité d'une solution. Il est donc raisonnable d'arrêter la recherche avant de compléter toute l'énumération. Malheureusement, l'interruption du processus d'énumération usuel en profondeur-d'abord restreint l'énumération à des solutions voisines. Le rejet heuristique d'une fraction des noeuds de droite lors de la prise de décisions multiples permet d'élargir la zone de cet échantillonnage, tel qu'illustré schématiquement à la figure 3.5. Le chapitre 4 propose une technique de recherche permettant d'obtenir le même effet, encore plus rapidement.

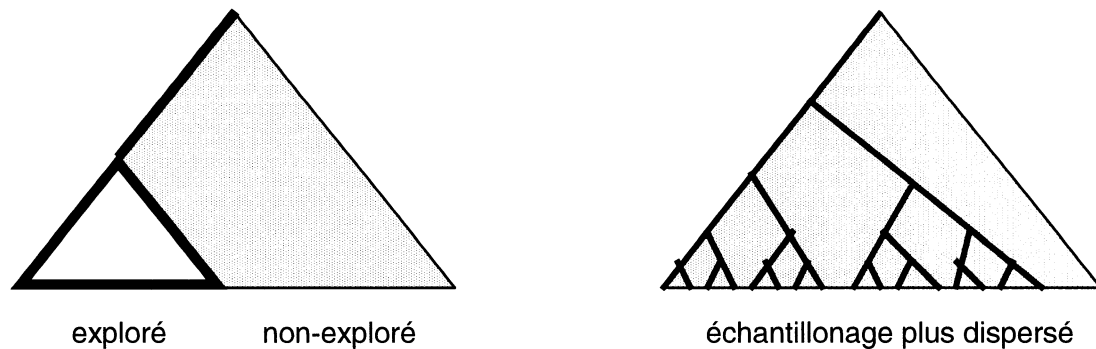


Figure 3.5: Comparaison de la diversité d'un échantillonnage

Le report du calcul de la relaxation linéaire et le rejet d'une fraction des noeuds de droite sont toutefois dissociables. Plusieurs variables peuvent être fixées entre deux noeuds sans éliminer heuristiquement des solutions mais la mémoire nécessaire à la représentation de l'arbre dans son entier devient démesurée. Le rejet d'une fraction des noeuds de droite — contenant possiblement la ou les solutions optimales — est le compromis employé car des noeuds voisins des noeuds rejetés et contenant des solutions entières de qualité

comparable à celle des solutions rejetées sont conservés. Ce résultat a été confirmé par Tseng et al. [59] et Gamache et al. [26] qui ont montré que les décisions multiples sont sous-optimales en ce qui concerne la qualité de la solution. Un raffinement du nombre de décisions prises le long du chemin reliant le noeud-racine et le noeud de la meilleure solution entière courante est donc conseillé. Cet aspect est considéré à la section 3.3.3.

3.2) BRANCHEMENT INTERDISANT DES CHEMINS

Pour utiliser efficacement la méthode d'énumération implicite à grande échelle, nous devons éviter toute répétition dans la représentation des solutions potentielles — tel que souligné par Pospichal et Kvasnicka [54]. Dans le cas du modèle (1.31)-(1.40), une décision élémentaire consiste à imposer un chemin à une locomotive d'un type donné. Pour établir une partition de l'ensemble des solutions, un chemin imposé à un noeud pour une locomotive doit pouvoir être interdit du réseau de la même classe de locomotives dans tous les autres noeuds-frères. Le sous-problème de génération de colonnes devient alors une variante du problème de plus court chemin où plusieurs chemins sont interdits. Cet ensemble de chemins interdits est noté F , leur nombre est noté $|F|$, et le problème résultant est noté *SPFP — Shortest Path with Forbidden Paths*.

Chaque noeud de l'arbre T est produit à partir d'un noeud-père v et d'une décision multiple M_χ^v qui contient une liste de décisions élémentaires supplémentaires, dont les chemins interdits additionnels. Un sous-problème de génération de colonnes à un noeud de l'arbre de décision consiste donc à générer un plus court chemin du réseau

correspondant parmi tous les chemins non-interdits. Nous considérons et comparons les deux approches suivantes pour résoudre ce sous-problème *SPFP* dans le cas sans ressource: la méthode des déviations et la méthode des k plus courts chemins.

3.2.1) Méthode des déviations

Le traitement des chemins interdits utilise leur représentation en séquence d'arcs. En effet, Villeneuve et Desaulniers [61] ont mis au point une procédure permettant d'interdire des séquences d'arcs en les parcourant pour les pré-étiqueter avant de résoudre les sous-problèmes. Le cas particulier de l'implantation pour interdire des chemins complets du noeud-source au noeud-puits et non des séquences d'arcs quelconques est décrit par Villeneuve [60]. En résumé, la technique consiste à enregistrer d'avance une liste d'étiquettes aux noeuds des chemins interdits avec la liste des arcs le long desquels elles ne peuvent pas être prolongées. Ces étiquettes agiront comme sources supplémentaires à partir des chemins interdits lors de la résolution par l'algorithme de plus court chemin usuel. Chaque prolongement permis à partir d'une de ces étiquettes représente une première déviation d'un chemin interdit. On note G le réseau initial et G^* le réseau augmenté des déviations permises. Une équivalence rigoureuse en terme de graphe est disponible dans Villeneuve et Desaulniers [61].

Un exemple montre comment obtenir le plus court chemin lorsque les chemins en pointillés de longueur 4 et 7 sont interdits. Sur fond gris sont indiqués les coûts des arcs. En comparant l'appartenance des arcs aux chemins interdits, l'algorithme doit générer une liste

d'étiquettes à chaque noeud pour trouver le chemin de longueur 5. Les valeurs des étiquettes à un noeud sont énumérées dans les rectangles ou simplement inscrite dans le noeud quand l'étiquette est unique. Les valeurs des étiquettes créées lors du pré-étiquetage des chemins interdits sont soulignées. La dominance entre les meilleurs chemins possibles est indiquée au noeud-puits.

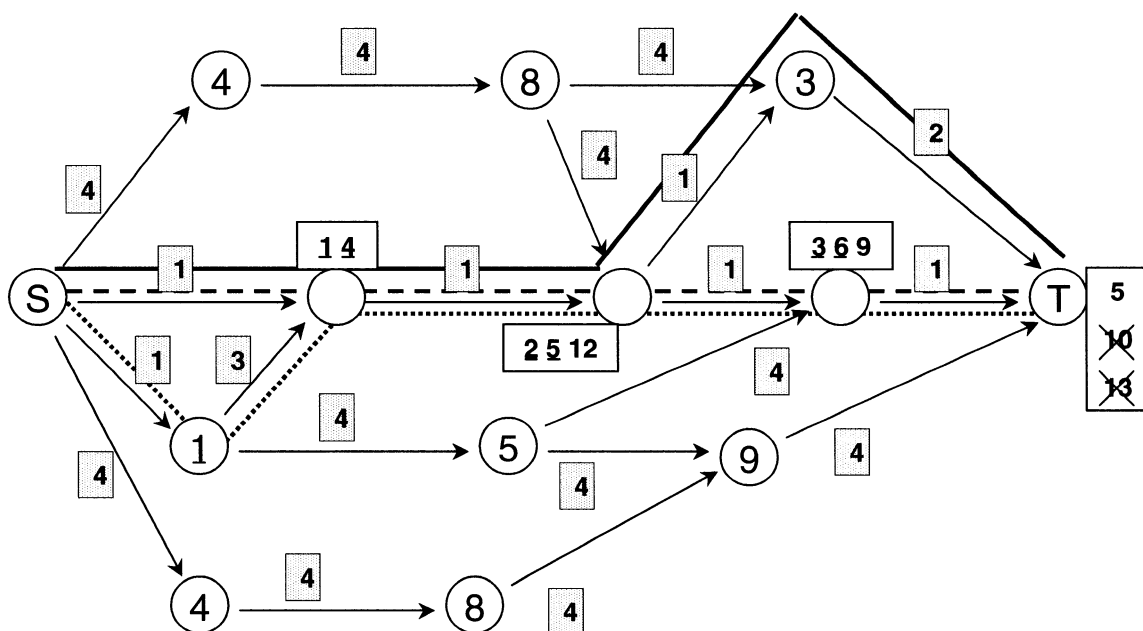


Figure 3.6: Résultat de l'application de l'algorithme de plus court chemin avec pré-étiquetage des chemins interdits

Les variations des variables duales qui définissent les coûts sur les arcs du sous-problème peuvent modifier la longueur des chemins entre deux appels du générateur de colonnes, et donc entre deux relaxations linéaires. En conséquence, il faut proscrire les chemins interdits à un noeud-aïeul précédent sans tenir compte de leur longueur: le nombre de chemins à proscrire n'est pas suffisant et l'enregistrement individuel, arc par arc, des chemins interdits est donc nécessaire.

C'est ce que permet la méthode proposée en interdisant n'importe quels $|F|$ chemins plutôt que seulement les $|F|$ plus courts chemins.

L'article de Villeneuve et Desaulniers [61] fournit la complexité d'une implantation de cette méthode de déviations pour résoudre le problème *SPFP* sans ressource, pour des séquences d'arcs interdites sur un graphe G contenant des cycles et plusieurs noeuds-puits:

$$O\left(S(n^*, m^*, c^*) + d_{\max}^+ \sum_{f \in F} |f| + n^{*2}\right). \quad (3.4)$$

Les symboles précédents représentent respectivement:

$d_{\max}^+$: degré extérieur maximal des noeuds du réseau G ;

f : chemin interdit dans le réseau G ;

$|f|$: nombre d'arcs permettant de décrire le chemin interdit f ;

n^* : nombre de noeuds dans le réseau G^* $\left(n^* \leq n + \sum_{f \in F} |f|\right)$;

m^* : nombre d'arcs dans le réseau G^* $\left(m^* \leq m + d_{\max}^+ \sum_{f \in F} |f|\right)$;

c^* : coût maximal sur un arc du réseau G^* ;

$S(n^*, m^*, c^*)$: complexité —en temps— d'un algorithme de plus

court chemin avec noeud-puits unique.

Le premier terme provient de la résolution usuelle d'un algorithme de plus court chemin, le second de la création d'une structure particulière pour représenter les déviations et le dernier de la translation permettant de récupérer le plus court chemin permis. C'est ce modèle que nous retenons pour traiter les chemins interdits dans la suite de cette étude.

Adaptons la complexité décrite par (3.4) de la méthode des déviations à notre contexte de génération de colonnes pour un problème

d'affectation de locomotives. Tout d'abord, le réseau G que nous considérons ne possède pas de cycle — les locomotives ne peuvent pas reculer dans le temps — et donc tout chemin est forcément élémentaire. Tout chemin contient alors au plus n arcs. La complexité de l'algorithme de plus court chemin approprié au problème d'affectation de locomotives est $O(m^* + n^*)$ car le réseau G^* est aussi acyclique par construction. Par ailleurs, puisque dans le réseau acyclique G tout chemin est élémentaire:

$$O\left(\sum_{f \in F} |f|\right) = O(|F|n). \quad (3.5)$$

Cette complexité peut encore être améliorée si on considère qu'en pratique le réseau est peu profond:

$$|f| \leq p_{\max} \ll n \quad (\forall f \in F) \Rightarrow O\left(\sum_{f \in F} |f|\right) = O(|F| p_{\max}). \quad (3.6)$$

Finalement, le sous-problème adapté au contexte d'affectation de locomotives décrit par (1.20)-(1.25) ne considère qu'un seul noeud-puits unique. Le dernier terme de (3.4) se résume à traduire un seul chemin à partir du noeud-puits de G^* vers G . Le graphe G étant acyclique, le graphe des déviations G^* contient donc des chemins de longueur maximale n en général, et de $p_{\max}$ dans notre contexte. Pour la méthode des déviations adaptée au contexte d'affectation de locomotives, la complexité (3.4) devient:

$$O(m^* + n^* + |F| d_{\max}^+ p_{\max} + n) \quad (3.7)$$

$$\text{où} \quad n^* \leq n + \sum_{f \in F} |f| \quad (3.8)$$

$$\text{et} \quad m^* \leq m + d_{\max}^+ \sum_{f \in F} |f|. \quad (3.9)$$

Donc la complexité de la méthode des déviations appliquée à notre contexte particulier est:

$$O(m + n + |F| d_{\max}^+ p_{\max}). \quad (3.10)$$

3.2.2) Méthode des k plus courts chemins

Une autre technique de résolution du sous-problème *SPFP* est la suivante: générer les k plus courts chemins lorsque $k-1$ chemins sont interdits, puis comparer un par un les k chemins trouvés à la liste des chemins proscrits. Cette idée a déjà été décrite par Hansen et al. [31-33] dans un contexte comparable: il s'agissait d'intercepter les k solutions générées par un algorithme appliqué au sous-problème pour les comparer à une liste de $k-1$ solutions interdites à des itérations précédentes du générateur de colonnes, avant d'envoyer les solutions restantes au problème maître.

Les algorithmes de k plus courts chemins de meilleure complexité rencontrés dans la littérature proviennent d'Eppstein [24] de façon générale et aussi dans le cas plus particulier d'un graphe sans variable de coût négatif. En notant $S(n,m,c)$ la complexité d'un algorithme de plus court chemin usuel pour un noeud-puits unique, l'algorithme d'Eppstein [24] produit une représentation implicite des k plus courts chemins selon les complexités suivantes:

$$\text{- dans le désordre: } O(S(n,m,c) + m + n + k); \quad (3.11)$$

$$\text{- en ordre croissant des coûts: } O(S(n,m,c) + m + n + k \log k). \quad (3.12)$$

L'expression précédente emploie les notations habituelles: k représente le nombre de plus courts chemins cherchés, m le nombre d'arcs et n le nombre de noeuds du réseau G . La complexité de $S(n,m,c)$ est $O(mn)$ de façon générale, $O(m+n \log n)$ lorsqu'il n'y a pas de variable de coût négatif sur les arcs et $O(m+n)$ lorsque le graphe est acyclique. Dans un contexte de génération de colonnes, l'ordonnancement des k plus courts chemins n'est pas forcément nécessaire puisque le sous-problème

peut suggérer plusieurs chemins permis dont la variable associée possède un coût réduit négatif. Nous retenons donc par la suite l'algorithme de k plus courts chemins dans le désordre, avec :

$$k = |F| + 1 \Leftrightarrow |F| = k - 1. \quad (3.13)$$

En nous référant de nouveau au travail de Villeneuve et Desaulniers [61], l'algorithme de résolution du problème *SPFP* par la méthode des k plus courts chemins se décompose en trois parties. La première consiste à créer un automate pour déterminer en $O(|p|)$ si un chemin p appartient à l'ensemble des chemins interdits F . L'algorithme de Aho et Corasick [1] construit l'automate en :

$$O\left(\sum_{f \in F} |f|\right). \quad (3.14)$$

La seconde partie est la résolution proprement dite du problème de k plus court chemin dont la complexité est décrite par (3.11) pour la version des plus courts chemins en désordre. La dernière étape permet de valider chaque plus court chemin. D'une façon générale, la complexité de la méthode des k plus courts chemins pour résoudre le problème *SPFP* est :

$$O\left(S(n, m, c) + m + n + |F| + \sum_{f \in F} |f| + |F|m\right). \quad (3.15)$$

Toutefois, dans notre contexte de génération de colonnes pour un problème d'affectation de locomotives, le réseau G que nous considérons est acyclique : $S(n, m, c) = O(m + n)$. En outre, tout chemin est forcément élémentaire et contient au plus n arcs, donc le dernier terme devient $O(|F|n)$. Finalement, l'hypothèse de faible profondeur du réseau exprimée par (3.6) reste applicable à l'expression précédente ainsi que pour calculer la complexité de construction de l'automate. En regroupant toutes ces expressions, nous obtenons la complexité de la

méthode des k plus courts chemins pour résoudre le problème *SPFP* dans le présent contexte :

$$O(m+n+|F| p_{\max}). \quad (3.16)$$

Pour fin de comparaison, l'algorithme de k plus courts chemins dans le désordre sans chemins interdits dans le même contexte d'affectation de locomotives a pour complexité selon (3.11) :

$$O(m+n+|F|). \quad (3.17)$$

La méthode des k plus courts chemins pour résoudre le problème *SPFP* reste donc de complexité comparable à l'algorithme de k plus courts chemins appliqué au problème usuel lorsque $|F| p_{\max} \ll m$.

3.2.3) Comparaison entre les méthodes

Il est difficile de comparer ces deux méthodes dans un contexte général car les complexités décrites par (3.4) et (3.15) dépendent de nombreux facteurs. Toutefois, nous pouvons comparer ces deux méthodes dans notre contexte particulier de génération de colonnes appliqué à l'affectation de locomotives en examinant les complexités décrites par (3.10) et (3.16). En terme de pire complexité, la méthode des k plus courts chemins semble systématiquement préférable.

Toutefois, soulignons qu'il est possible de généraliser la méthode des déviations de Villeneuve et Desaulniers [61] pour résoudre des séquences d'arcs interdites, et ce pour un sous-problème de plus court chemin sur un graphe cyclique avec ressources et plusieurs noeuds-puits. Malheureusement, à notre connaissance, il n'existe pas d'algorithme de k plus courts chemins avec contraintes de ressources.

C'est donc une implantation simplifiée de la méthode des déviations que nous utilisons pour la suite de cette étude.

3.3) DIFFÉRENTES TACTIQUES DE BRANCHEMENT

Le branchement sur les variables de chemin permet de conserver la structure du modèle en branchant sur les variables du problème. Il reste à déterminer la meilleure combinaison de décisions permettant d'isoler une densité élevée de solutions de bonne qualité. La densité de solutions de bonne qualité se définit comme suit. Soient L_χ^v le sous-arbre de L obtenu après avoir ajouté la décision M_χ^v au noeud v , N_χ^v l'ensemble des noeuds du sous-arbre L_χ^v dont la solution relaxée est entière, et N_χ^{v*} l'ensemble de ces noeuds possédant une solution entière de coût moindre que Z_c^* , celui de la meilleure solution entière courante $\mathbf{x}_c^*$, la densité recherchée s'exprime alors par le rapport des cardinalités suivant :

$$\left| N_\chi^{v*} \right| / \left| N_\chi^v \right| \in [0,1]. \quad (3.18)$$

En combinant les décisions multiples et l'interdiction de chemins, diverses tactiques de branchement peuvent être construites. Les sections suivantes en suggèrent trois puis les comparent. Nous utiliserons aussi le terme de "vallée de solutions" pour identifier un sous-arbre L_χ^v , dont la densité de solutions entières meilleures que la solution courante devrait être supérieure à celle de la forêt encore non-explorée. L_1^v représente donc le premier sous-arbre exploré à partir du noeud v , L_2^v une seconde tentative, et ainsi de suite...

3.3.1) Interdiction du dernier meilleur chemin

La première tactique de branchement employée provient de celle proposée par Ziarati [63]. En notant $\bar{\theta}_p^k$ la valeur des variables de flot θ_p^k des chemins de la solution de la relaxation linéaire au noeud courant, les parties fractionnaires f_p^k s'expriment:

$$f_p^k = \bar{\theta}_p^k - \lfloor \bar{\theta}_p^k \rfloor. \quad (3.19)$$

En conservant la même notation, la définition des compléments à l'entier suivant S_p^k s'exprime ainsi:

$$S_p^k = \lceil \bar{\theta}_p^k \rceil - \bar{\theta}_p^k = 1 - f_p^k. \quad (3.20)$$

Nous proposons d'étudier Φ_p^k , la différence entre les détériorations de l'objectif Δ_p^k et ϵ_p^k lorsque la décision D_p^k est respectivement prise ou interdite. En conservant les indices précédents, mais en utilisant la notation de Taha (2.10)-(2.12):

$$\Phi_p^k = \left| \Delta_p^k - \epsilon_p^k \right| \approx \left| u_p^k S_p^k - d_p^k f_p^k \right|. \quad (3.21)$$

Ziarati [63] propose de classer les chemins considérés selon les compléments à l'entier suivant S_p^k croissants: un nombre n_1^v inférieur à un maximum prédéterminé — c.f. équation (3.3) — des meilleures décisions est fixé. Typiquement, n_1^v est égal au nombre de commodités. Il est alors possible de produire un échantillonnage représentatif de l'arbre en interdisant dans un noeud de droite le dernier chemin fixé, le plus douteux, tel que représenté à la figure 3.7.

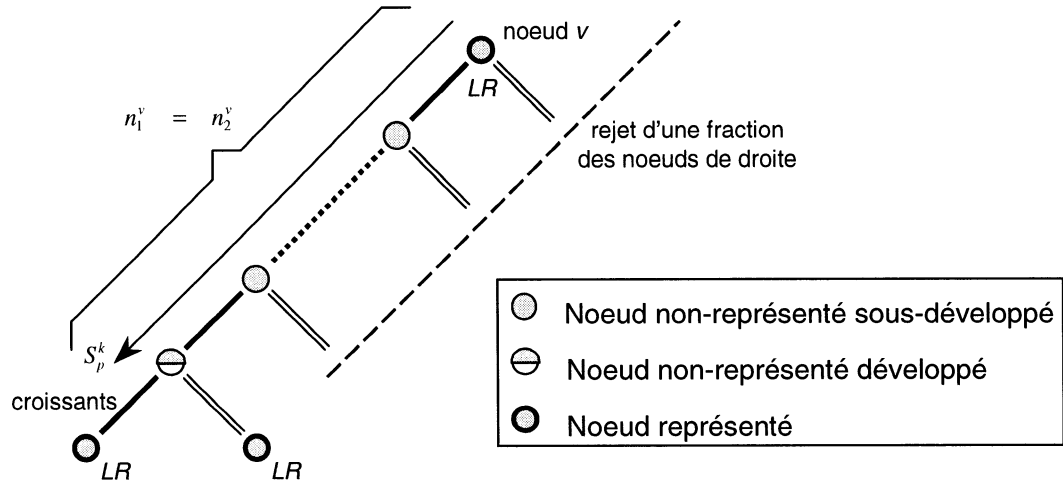


Figure 3.7: Branchement interdisant le dernier meilleur chemin

L'avantage principal de cette tactique est qu'elle regroupe les pires décisions vers le bas de l'arbre. La solution initiale est relativement bonne et le sous-arbre contenant des décisions interdisant des chemins dont les flots fractionnaires sont inférieurs à 0,5 contient une forte densité de solutions potentiellement intéressantes. En effet, l'analyse de la distribution des solutions entières contenues dans un sous-arbre voisin L_2^v des noeuds menant à la meilleure solution entière courante laisse entrevoir un modèle respectant les trois caractéristiques suivantes. Tout d'abord, la valeur moyenne $\tilde{Z}_2^{v*}$ des solutions entières du sous-arbre L_2^v est approximée par la valeur de la solution optimale courante Z_1^{v*} du sous-arbre L_1^v à laquelle est ajoutée la détérioration Φ_p^k (3.21) causée par la dernière décision complémentaire. Ensuite, la valeur des solutions entières est bornée, par la valeur connue de la relaxation linéaire Z_{lr}^v au noeud v et par la valeur plus restrictive mais inconnue de la meilleure solution entière $Z_2^{v*} \geq Z^*$ du sous-arbre L_2^v . Enfin, la variance est liée au nombre de décisions restant à prendre et à la nature du branchement. Tous ces éléments suggèrent un modèle de distribution illustré à la figure 3.8.

Dans l'exemple de gauche de la figure suivante, aucune nouvelle solution entière de meilleure qualité n'existe dans le sous-arbre L_2^v . Toutefois, lorsque la variance de la distribution est relativement plus forte que la détérioration, la densité de meilleures solutions entières augmente comme dans l'exemple de droite.

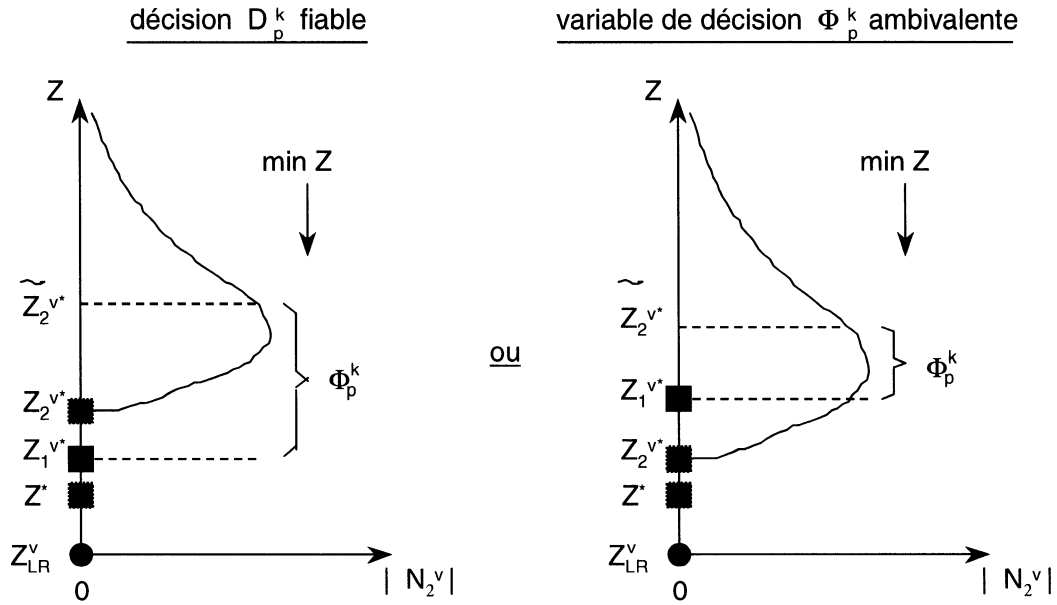


Figure 3.8: Caractéristiques d'un modèle de distribution des coûts des solutions entières d'un sous-arbre voisin L_2^v

Le triangle blanc de la figure 3.9 identifie un sous-arbre où la détérioration Φ_p^k (3.21) est faible et où la densité (3.18) devrait être plus élevée. Cette zone représente les cas où la dernière meilleure décision suggérée par la relaxation linéaire est de mauvaise qualité ($D_p^k | f_p^k \leq 0,5$). La remise en question d'un chemin de ce type remplace généralement une mauvaise suggestion par une ultérieurement jugée légèrement plus mauvaise ($D_{p'}^{k'} | f_{p'}^{k'} \leq f_p^k$). Cette estimation peut toutefois s'avérer fausse. En explorant de nombreuses branches

estimées légèrement détériorées par rapport à la branche de la solution courante, la probabilité d'identifier une meilleure solution augmente.

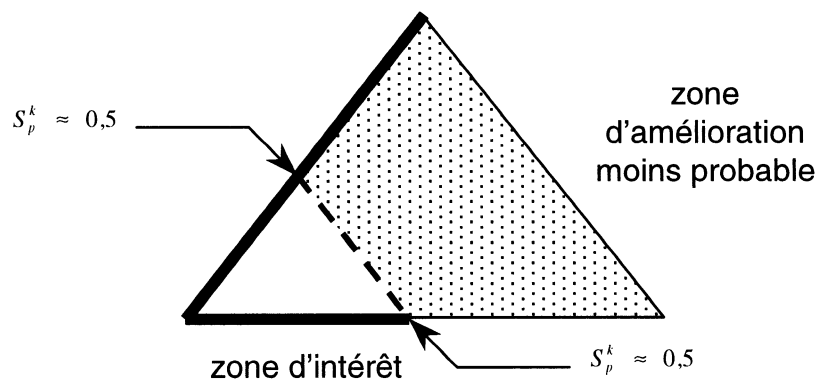


Figure 3.9: Sous-arbre à plus forte densité de meilleures solutions

Le problème principal de cette tactique, c'est qu'elle génère des solutions comparables. La première solution entière obtenue est relativement bonne mais la remise en question des moins pires décisions finales ne modifie pas beaucoup la nature de la solution. En effet, la figure 3.9 montre clairement que toutes les solutions dans la zone intéressante possèdent le même ensemble de décisions initiales. A la suite de ces décisions, certaines contraintes peuvent devenir fortement restrictives et provoquer des "goulots" ne possédant plus qu'une ou deux solutions à un coût raisonnable. En résumé, la variance des solutions est faible et, hors de la zone d'intérêt montrée à la figure 3.9, la détérioration est trop forte. Pour récupérer ces degrés de liberté gaspillés, il est nécessaire de changer les décisions initiales ou d'identifier des décisions ambivalentes à faible détérioration: c'est ce que nous essayons dans les deux tactiques de branchement suivantes.

3.3.2) Interdiction du pire chemin

Pour générer des solutions fortement disparates, une autre tactique de branchement est proposée. L'idée originelle provient de la logique suivante: moins la solution entière s'éloigne de la solution relaxée optimale, moins l'objectif du point entier peut se détériorer. En ce sens, fixer une variable de flot fractionnaire 0,13 à 0 semble moins dommageable que de fixer une variable de flot fractionnaire 0,73 à 1. En outre, l'interdiction d'un chemin laisse généralement beaucoup plus de degrés de liberté pour compléter une même tâche et interdire le pire chemin devrait avoir moins d'impact que d'en imposer un. Le noeud de gauche contient donc toutes les bonnes décisions et en plus interdit dès le départ le chemin de flot fractionnaire minimal. Le noeud de droite fixe à l'unité tous les flots de ces chemins. La figure 3.10 représente le branchement obtenu.

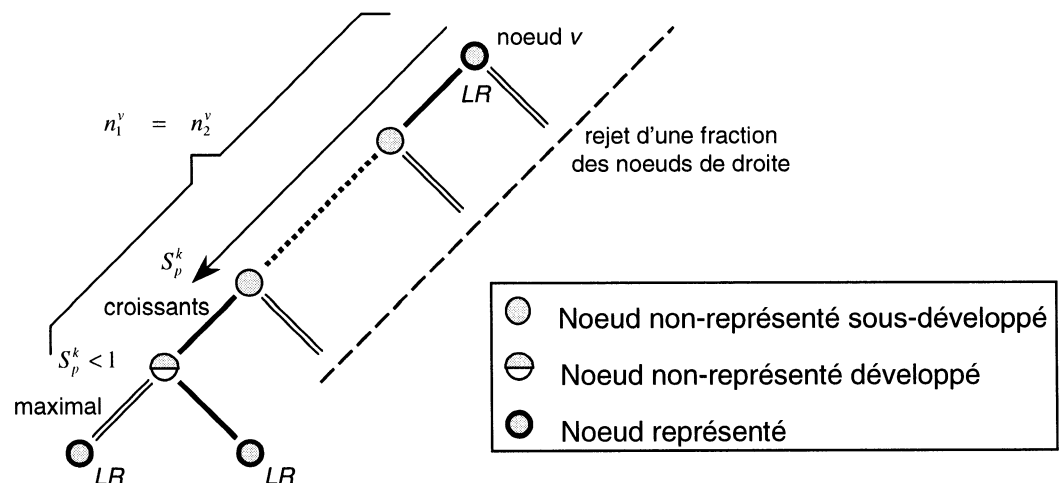


Figure 3.10: Branchement interdisant le pire chemin

Quelques problèmes surviennent toutefois: en pratique, interdire le chemin ne fait que remettre à plus tard la décision de choisir

quelle locomotive accomplira les tâches correspondantes. Si la portion de travail est accomplie par une autre locomotive de flot antérieurement nul, la décision est nuisible et la solution devient simplement plus coûteuse. Les quelques fois où le flot de la variable interdite est transféré sur au moins une autre de flot non-nul, la solution prend une composition différente: la variance sera plus importante.

En comparaison avec la tactique initiale, la première solution entière légèrement détériorée reste bonne et l'addition de variété génère quelque fois plus de variance dans la qualité des solutions. Malheureusement, plus on remet en question de décisions et plus il est certain que des chemins de flot initialement nul apparaissent dans la solution. Même si la densité de meilleures solutions entières peut s'améliorer pour des sous-arbres L_2^v contenant une seule remise en question, la détérioration augmente trop fortement lorsque plusieurs décisions élémentaires ont été remises en cause dans la solution.

3.3.3) Prise de décisions sur les chemins ambivalents

Pour remédier à ce problème et regrouper les avantages des deux tactiques précédentes en un seul "schéma", nous proposons une dernière approche. Le but de celle-ci est de générer des sous-arbres différents et à l'intérieur de chacun d'eux, isoler une densité supérieure de solutions de bonne qualité. En branchant sur les variables d'une décision principale dont le flot fractionnaire est proche de 0,5, la solution entière s'éloigne davantage de la solution relaxée et la composition des convois diffère entre les noeuds-frères. Cette tactique introduit une faible détérioration et une forte variance. Chaque remise en question d'une décision lors de l'étape de retour-

arrière propose alors une solution de nature différente et potentiellement meilleure. L'accélération par prise de décisions multiples s'effectue en fixant à 1 les flots fractionnaires proche de l'unité.

Un début de réponse concernant de trop fortes détériorations a trait au nombre de décisions arbitrairement fixé. Pour ce faire, au lieu de fixer *a priori* le nombre de décisions à prendre à chaque noeud de branchement, ce nombre est établi à chaque noeud en fonction des évaluations des détériorations prévues. Tout d'abord, un branchement sur une décision principale dont le niveau d'indécision est élevée réduit les chances de rejeter un noeud de droite contenant une solution optimale. Cette décision nous permet de définir le niveau de confiance $\Phi_{\min}$ comme le minimum des différences $\Phi_p^k((k,p) \in K \times \Omega^k)$. Ce minimum permet d'identifier une paire commodité-chemin $(k^* \in K, p^* \in \Omega^{k^*})$:

$$(k^*, p^*) = \arg \min (\Phi_p^k) \quad (3.22)$$

et la décision principale $D_{p^*}^{k^*}$ ou $\bar{D}_{p^*}^{k^*}$ associée à la partie fractionnaire $f_{p^*}^{k^*} = f_{\text{pri}}$ correspondante.

Si les coûts réduits de toutes les variables fractionnaires peuvent être supposés comparables $(u_p^k \approx d_p^k \approx \bar{c}, \forall (k,p) \in K \times \Omega^k)$, l'équation (3.21) devient:

$$\Phi_p^k \approx |\bar{c}(2f_p^k - 1)| \quad \forall (k,p) \in K \times \Omega^k, \quad (3.23)$$

$$\text{et} \quad \Phi_{\min} \approx \bar{c} \min_{(k,p) \in K \times \Omega^k} |2f_p^k - 1| \approx \bar{c} |2f_{\text{pri}} - 1|. \quad (3.24)$$

La différence minimale $\Phi_{\min}$ est donc fonction de la partie fractionnaire f_{pri} la plus proche de 0,5. Il suffit alors de comparer les prévisions de détériorations de l'objectif entier ΔZ_p^k au niveau de

confiance $\Phi_{\min}$ pour sélectionner les décisions complémentaires à la décision principale:

$$\Delta Z_p^k \approx \bar{c} (1 - f_p^k) \quad (3.25)$$

$$\text{et} \quad \Delta Z_p^k \leq \Phi_{\min} \Leftrightarrow \bar{c} (1 - f_p^k) \leq 2\bar{c} |f_{\text{pri}} - 0,5|. \quad (3.26)$$

Nous obtenons donc l'ensemble complémentaire $\hat{M}_1^v$ des décisions "fiabiles" en utilisant une limite inférieure $f_{\lim}$ sur la partie fractionnaire:

$$\hat{M}_1^v = \left\{ D_p^k \mid (k, p) \in K \times \Omega^k \text{ et } f_p^k \geq f_{\lim} = 1 - 2 |f_{\text{pri}} - 0,5| \right\}. \quad (3.27)$$

Il n'est plus évident de garder ou d'interdire le chemin ambivalent dans le noeud de gauche. Le noeud le plus prometteur, celui qui s'éloigne un minimum de la solution optimale relaxée, est exploré en premier. Le chemin est donc interdit dans le noeud de gauche si son flot fractionnaire est inférieur à 0,5 et dans le noeud de droite sinon. Le branchement résultant est représenté à la figure 3.11.

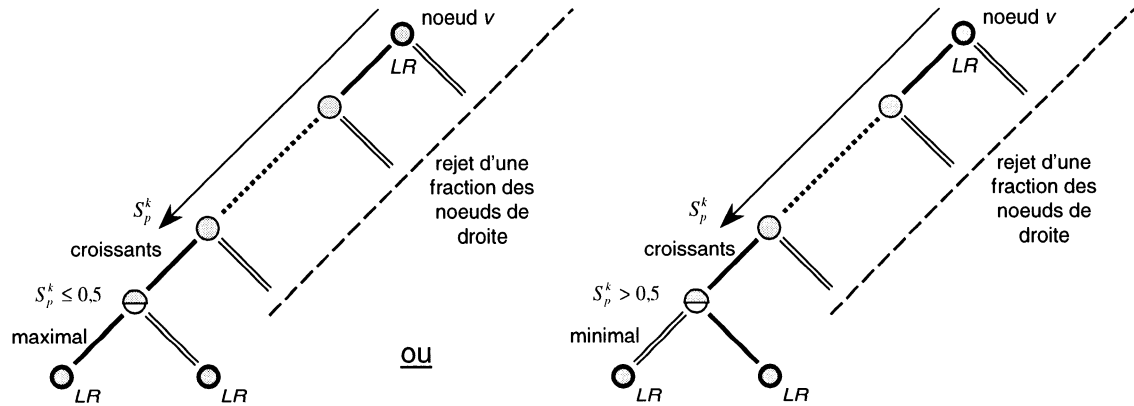


Figure 3.11: Branchement sur les chemins ambivalents

Cette tactique de branchement produit une assez bonne solution initiale, des noeuds-frères introduisant de la variété dans les solutions et un branchement multiple pour accélérer le temps de calcul en protégeant des solutions ε -optimales. Toutefois, pour obtenir des résultats dans des temps comparables aux autres tactiques, un coefficient d'accélération A est introduit. En se référant à la définition de l'ensemble $\hat{M}_1^v$ des décisions fiables (3.27) :

$$\hat{M}_1^v = \left\{ D_p^k \mid (k, p) \in K \times \Omega^k \text{ et } f_p^k \geq f'_{\text{lim}} = 1 - 2A \left| f_{\text{pri}} - 0,5 \right| \right\}. \quad (3.28)$$

Dans les tests effectués, ce paramètre A peut prendre deux valeurs : l'une de l'ordre de grandeur du nombre de commodités, l'autre deux fois plus petite. Cette tactique de branchement permet ainsi de lier le nombre de décisions élémentaires contenues dans une décision multiple au niveau de confiance $\Phi_{\min}$ avec lequel on pense pouvoir rejeter un grand nombre de noeuds de droite.

3.4) RÉSULTATS NUMÉRIQUES

Pour valider l'ensemble de cette étude, nous présentons les résultats obtenus pour trois scénarios de taille et de nature différentes. Toutes ces simulations sont calculées à partir d'une base de données du CN établie en 1998. Le scénario le plus facile est évalué sur une fenêtre de temps de deux jours avec une surabondance de locomotives disponibles : c'est un problème stratégique peu contraint permettant de sélectionner l'équipement. Le second est un scénario typique quand au niveau de complexité nécessaire pour résoudre des problèmes industriels. Il correspond à des conditions normales d'exploitation : on utilise une autre fenêtre de temps de deux jours et la disponibilité des locomotives est restreinte à un niveau réaliste. Finalement, le dernier scénario emploie une fenêtre de temps de trois

jours mais ne considère pas les locomotives critiques pour introduire plus de diversité entre les problèmes: c'est un cas plus difficile de disponibilité d'équipement très contraignante. Le tableau 3.1 résume quelques caractéristiques permettant de mesurer et de comparer la complexité de chacun de ces scénarios. Le nombre de variables fractionnaires de la solution relaxée est l'indice principal du niveau de complexité du problème.

Tableau 3.1: Caractéristiques des scénarios étudiés

	Scénario 1	Scénario 2	Scénario 3
Fenêtre de temps	2 jours	2 jours	3 jours
Locomotives critiques	oui	oui	non
Surabondance de locomotives	oui	non	non
Nombre de rangées du problème maître après agrégation des réseaux	1939	2223	2151
Nombre de segments de train	4031	2908	3167
Nombre de noeuds	4898	2453	3303
Nombre d'arcs	662295	208954	45091
Nombre de réseaux	308	301	19
Nombre de variables fractionnaires de la relaxation linéaire	823	936	1309
Saut d'intégrité avec l'approche initiale	0,216 %	4,4 %	9,36 %

Les graphiques suivants illustrent l'évolution de la qualité de la meilleure solution entière connue en fonction du temps *CPU* pour chacun de ces scénarios. L'axe vertical représente la valeur normalisée de la fonction objectif de la meilleure solution entière trouvée en tout

temps. La valeur minimale (0) utilisée est celle de la relaxation linéaire du problème — après avoir ajouté les quelques contraintes dynamiques communes à tout l'arbre de recherche — et la valeur maximale (1) représente le coût de la solution entière trouvée par la technique précédente de Ziarati [63], soit celui de la première solution entière du branchement sur le dernier meilleur chemin à un rythme de 40 décisions élémentaires par décision multiple. Le temps *CPU* est indiqué en heures et équivaut à 24 heures de temps de calcul sur une machine de 400 MHz. Ces temps pourront être réduits à quelques heures avec les machines de 2 ou 3 GHz. La légende indique le type de branchement — *Bdern_meilleur*, *Bpire_chemin*, *Bdyn_ambiv* — et la valeur de leur principal paramètre: le nombre de décisions élémentaires par décision multiple dans le cas des branchements sur le dernier meilleur chemin ou sur le pire chemin — *D40* ou *D20* — et le coefficient d'accélération dans le cas du branchement dynamique sur les chemins ambivalents — *A40* ou *A20*. La stratégie de recherche employée pour tous les résultats présentés dans ce chapitre est de type profondeur-d'abord.

La figure 3.12 montre l'évolution de la qualité des solutions pour les six combinaisons possibles appliquées au premier scénario. Après deux heures, quatre combinaisons sur six permettent de réduire le saut d'intégrité de près de 65% comparativement à la solution unique obtenue après 15 minutes par la recherche initiale de Ziarati. Malheureusement, les deux dernières subissent une détérioration relative de 283% (non-visible sur la figure) pour un même temps *CPU*. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité de l'ordre de 67% après vingt-quatre heures de calcul. La zone d'intérêt du branchement sur le dernier meilleur chemin à 40 décisions élémentaires par décision multiple est

complètement explorée après six heures de calcul mais quelques solutions entières très légèrement meilleures apparaissent ensuite. La tactique de branchement dynamique ambivalent indique aussi qu'aucune meilleure solution entière n'est disponible dans le sous-arbre exploré, quel que soit le niveau d'accélération choisi.

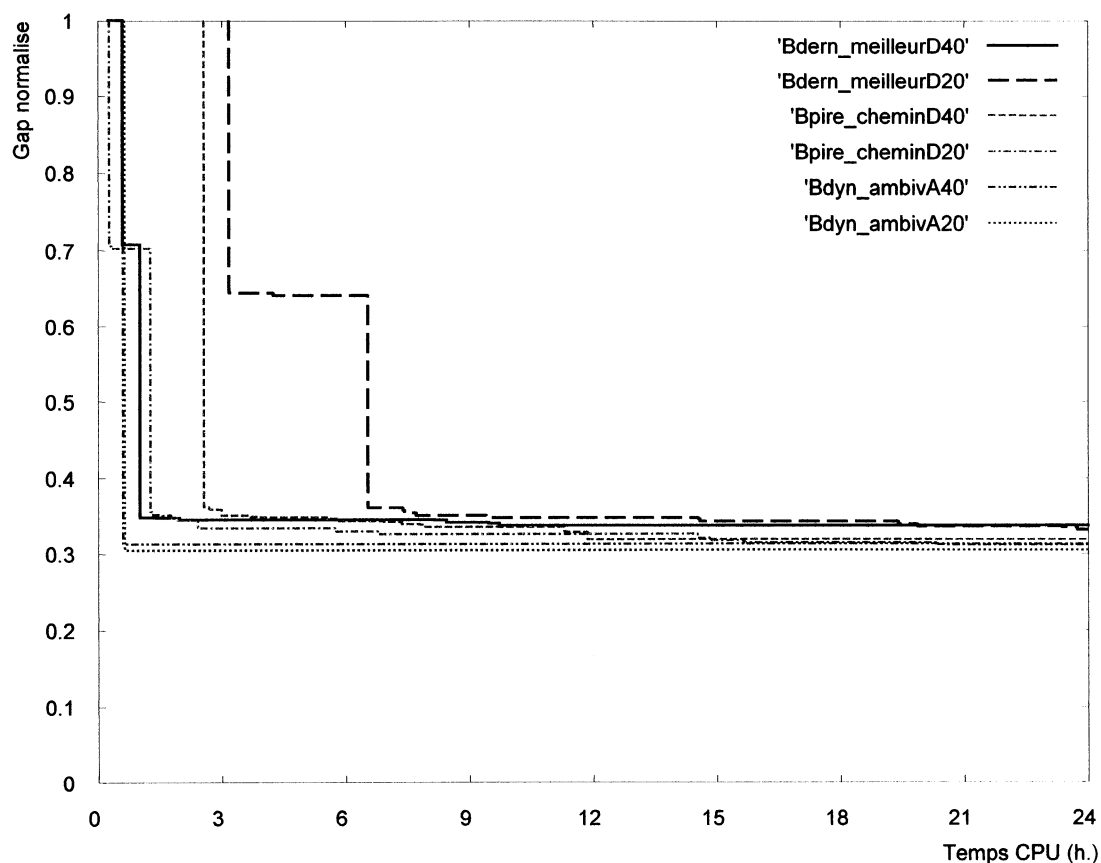


Figure 3.12: Evolution du coût de la meilleure solution entière pour le scénario 1 en stratégie d'exploration profondeur-d'abord

La figure 3.13 montre l'évolution de la qualité des solutions pour ces mêmes tactiques de branchement appliquées au second scénario. Après deux heures, la réduction moyenne du saut d'intégrité est de 22,5% comparativement à la solution obtenue par l'approche initiale

après 15 minutes. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité entre 21% et 36% après vingt-quatre heures de calcul, à l'exception de la combinaison *Bdyn_ambivA20* qui ne sauve que 9%. La tactique de branchement dynamique ambivalent montre toujours le même comportement quel que soit le niveau d'accélération choisi: aucune meilleure solution entière notable n'est disponible dans le sous-arbre exploré. La zone d'intérêt du branchement sur le dernier meilleur chemin à 40 décisions élémentaires par décision multiple est complètement explorée après douze heures de calcul mais quelques solutions entières très légèrement meilleures apparaissent encore.

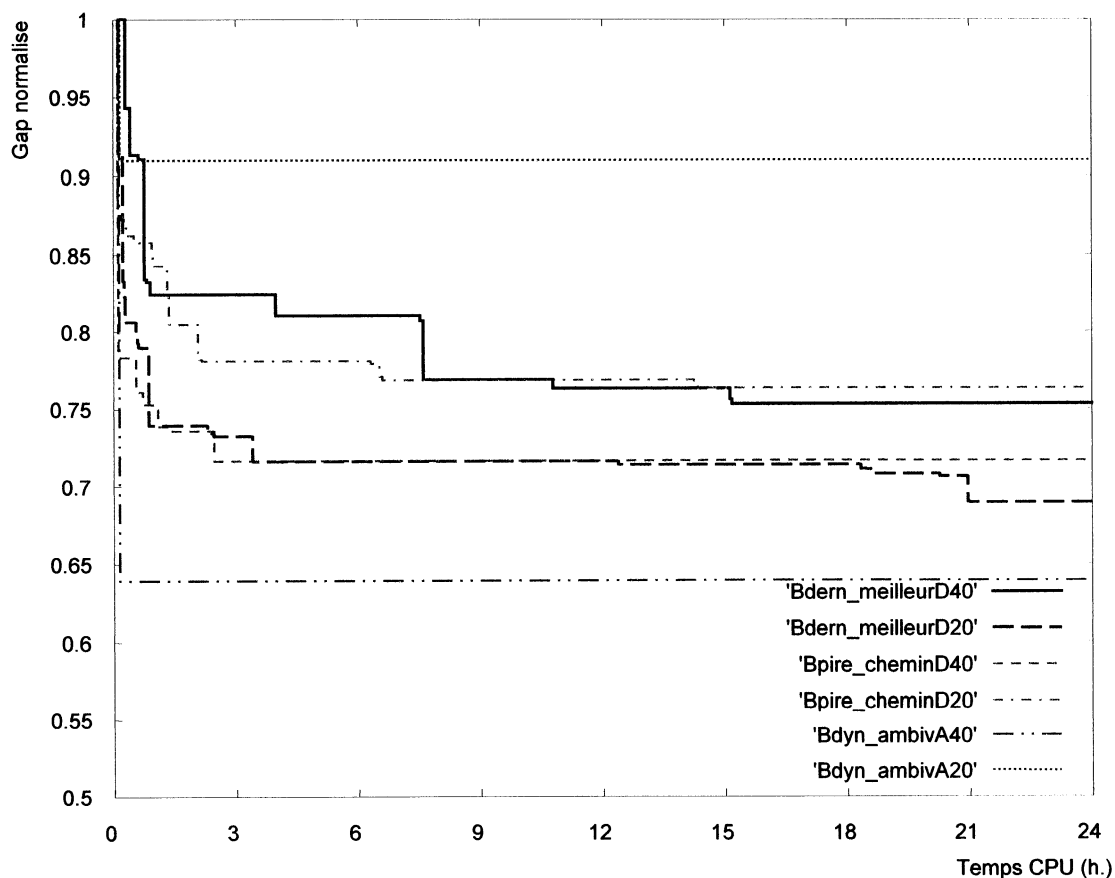


Figure 3.13: Evolution du coût de la meilleure solution entière pour le scénario 2 en stratégie d'exploration profondeur-d'abord

La figure 3.14 illustre les mêmes courbes pour le dernier scénario. Après deux heures, la réduction moyenne du saut d'intégrité est de 20% comparativement à la solution obtenue après 15 minutes grâce à l'approche initiale de Ziarati. Les deux premières tactiques de branchement permettent une amélioration relative du saut d'intégrité entre 23% et 30% après vingt-quatre heures de calcul. La tactique de branchement dynamique ambivalent ne sauve que 7% ou 16% selon le niveau d'accélération choisi et ne montre toujours aucune meilleure solution entière dans le sous-arbre exploré. La zone d'intérêt du branchement sur le dernier meilleur chemin à 40 décisions élémentaires par décision

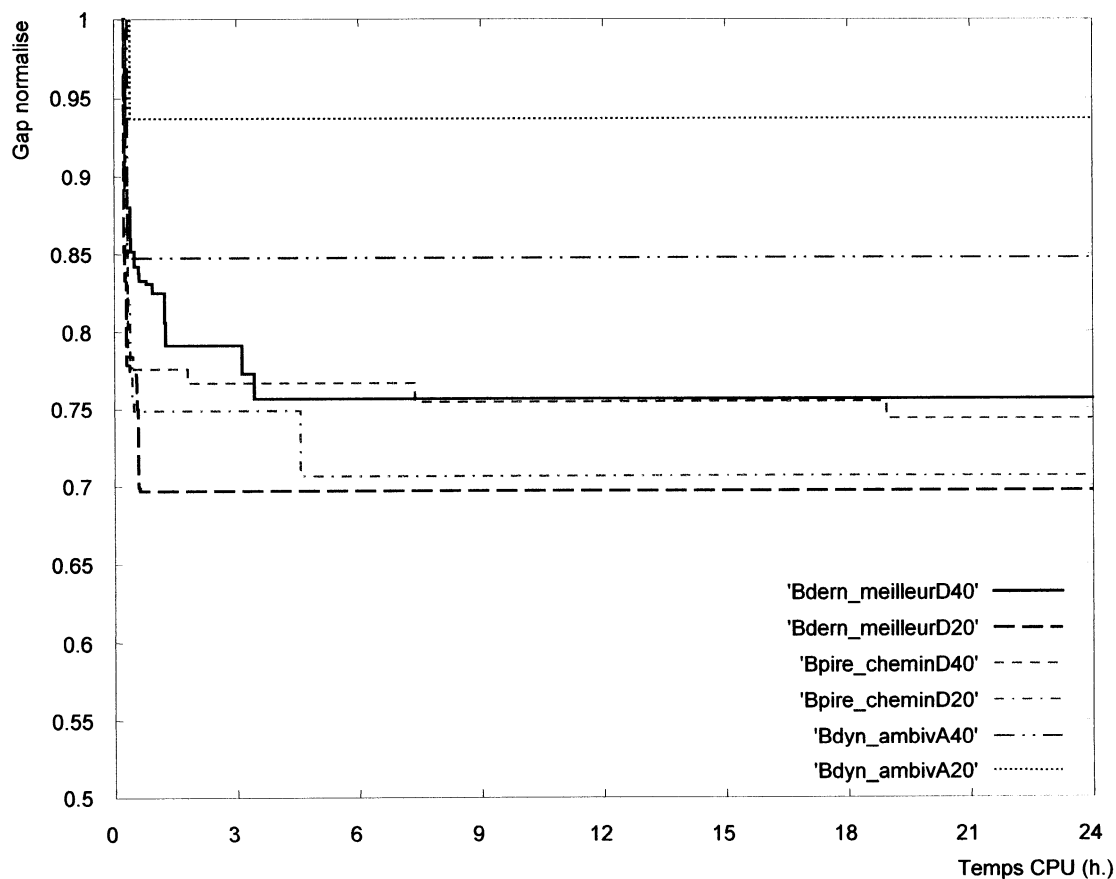


Figure 3.14: Evolution du coût de la meilleure solution entière pour le scénario 3 en stratégie d'exploration profondeur-d'abord

multiple n'est pas encore totalement explorée après vingt-quatre heures de calcul.

A partir de ces données, une dernière figure permet d'analyser le comportement général de toutes ces tactiques de branchement. En conservant la normalisation de toutes les valeurs objectifs entre la valeur de la relaxation linéaire et le coût de la solution obtenue par la tactique de Ziarati *et al.* [64], nous proposons de représenter en tout temps la moyenne arithmétique de nos trois scénarios.

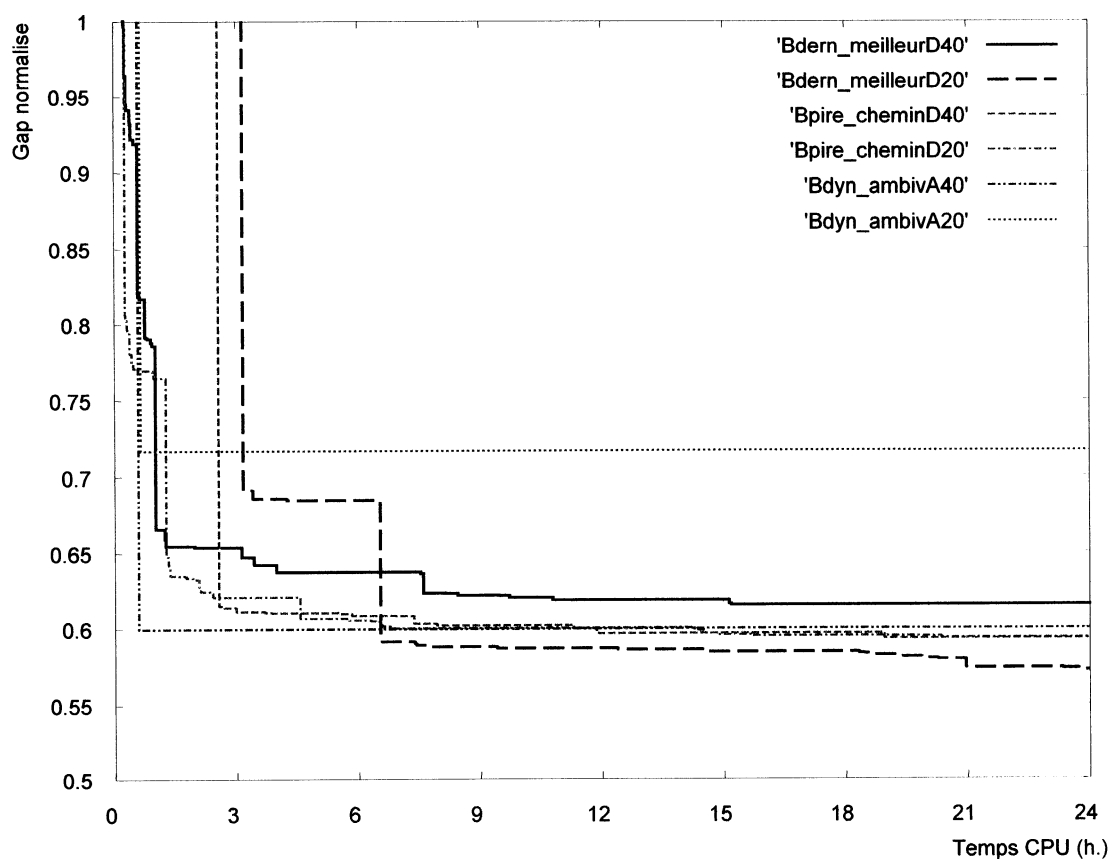


Figure 3.15: Evolution moyenne du coût de la meilleure solution entière pour tous les scénarios en stratégie d'exploration profondeur-d'abord

Tout d'abord, après deux heures de temps *CPU*, quatre combinaisons montrent une réduction moyenne de 35%. Toutefois, ce résultat est terni par la détérioration relative de 77% des deux autres combinaisons (non-visible sur la figure). Pour les six combinaisons, la détérioration relative moyenne est de 2%. Après vingt-quatre heures de calcul toutefois, toutes les tactiques de branchement montre une amélioration minimale de 28% et moyenne de 38,5%. Cette amélioration est principalement explicable par la recherche systématique que nous introduisons en utilisant des chemins interdits pour séparer en parties disjointes la filiation à chaque noeud. Le retour-arrière devient ainsi efficace car il génère toujours une solution originale à chaque nouveau calcul. La tactique de branchement dynamique ambivalent montre un comportement particulier. Malgré des améliorations moyennes de 28% et 40% du saut d'intégrité, nous remarquons surtout que la première solution entière est conservée (invisibles à l'oeil sur les courbes) et que l'exploration en profondeur-d'abord autour de cette solution n'apporte rien. Le comportement de cette tactique de branchement est généralement le suivant: prudence initiale lors des premières décisions multiples dont la décision principale a un flot proche de 0,5 puis accélération supérieure aux autres tactiques jusqu'à ce qu'il ne reste plus que des flots inférieurs à 0,5. Enfin, la tactique de branchement dynamique sur les chemins ambivalents achève en prenant un petit nombre de décisions élémentaires par décisions multiple pour obtenir une solution entière. C'est ce dernier sous-arbre que la stratégie d'exploration en profondeur-d'abord vide sans amélioration. Cette tactique de branchement semble donc bien efficace pour obtenir une très bonne solution à l'intérieur de cette dernière vallée de solutions. Les deux autres tactiques montrent en moyenne des améliorations relatives du saut d'intégrité entre 38,5% et 43% après 24 heures.

CHAPITRE 4

STRATÉGIE D'EXPLORATION PAR MEILLEURES ATTENTES

Dans ce chapitre, nous présentons une stratégie d'exploration sélectionnant les noeuds dans l'ordre du meilleur objectif espéré. Une première version est simplement basée sur les valeurs des flots fractionnaires des solutions relaxées pour améliorer l'énumération de l'arbre de recherche et une seconde prend en compte la grande variation des coûts sur les diverses commodités permettant de résoudre le problème d'affectation des locomotives pour le transport de marchandises. Chacun de ces sujets est décrit dans les sections qui suivent.

Les stratégies d'exploration classiques ont quelques désavantages mais permettent souvent de résoudre un problème précis. La stratégie meilleur-d'abord a tendance à osciller entre toutes les décisions favorables. La mémoire utilisée et le temps de découverte d'une première solution entière augmentent rapidement avec la taille des problèmes. Toutefois, on peut utiliser cette stratégie pour calculer une borne optimiste plus stricte en exploitant la force brute de l'ordinateur.

La stratégie d'exploration en profondeur-d'abord n'a pas ce problème de mémoire car elle balaie l'arbre de gauche à droite. A ce titre, on peut l'employer comme base de comparaison à la section 3.4. Malheureusement, aucune information ne provient de la qualité de la valeur de l'objectif obtenu au noeud précédent et la qualité des résultats n'influence pas l'ordre dans lequel l'arbre est exploré.

La stratégie d'exploration par meilleures attentes est équivalente à appliquer la stratégie meilleur-d'abord en remplaçant le critère d'ordonnancement de la borne inférieure par un objectif entier espéré. Ses composantes sont décrites dans les sections suivantes.

4.1) OUTILS DE RECHERCHE BASÉS SUR LES FLOTS FRACTIONNAIRES

La stratégie suivante tire profit des résultats de chaque relaxation effectuée à un noeud de l'arbre sans consommer une mémoire excessive. L'identification d'une vallée intéressante est fondée sur l'extrapolation d'une valeur d'objectif pour une solution entière à partir de chaque noeud et le gaspillage de mémoire est restreint par l'utilisation de coups de sonde.

4.1.1) Extrapolation de l'objectif espéré

Il y a de multiples façons d'extrapoler le long d'une branche de recherche la valeur de l'objectif. Toutefois, l'évolution de la valeur de l'objectif des solutions de problèmes relaxés, en fonction d'une mesure de faisabilité, semble indiquer un modèle de (dé)croissance exponentielle pour représenter la détérioration de l'objectif au fur et à mesure que sont prises des décisions, et non pas linéaire tel que modélisé par Taha [57] dans ses deux premiers modèles (2.7) et (2.12). Le comportement des détériorations est donc comparable à un modèle d'intérêts composés en finance ou de mesure des rayonnements en physique:

$$Z(S) = Z_c^* e^{-\alpha S} \quad (4.1)$$

$$\text{et} \quad \alpha = \frac{\ln(Z_c^*/Z_0)}{S_0}, \quad (4.2)$$

avec la notation suivante:

- S : distance fractionnaire à parcourir à un noeud quelconque;
- $Z(S)$: approximation de la valeur de l'objectif pour la solution entière espérée à partir du noeud courant v en fonction de S ;
- Z_c^* : valeur de l'objectif de la meilleure solution entière connue;
- α : taux de détérioration de l'objectif;
- Z_0 : valeur de l'objectif relaxé obtenu à la racine de l'arbre;
- S_0 : distance fractionnaire initiale.

La figure 4.1 illustre le résultat obtenu. Les traits plein représentent les décisions multiples reliant le noeud-racine au noeud représentant la meilleure solution entière courante. Le trait en pointillé simule l'évolution interpolée entre les points de coordonnées

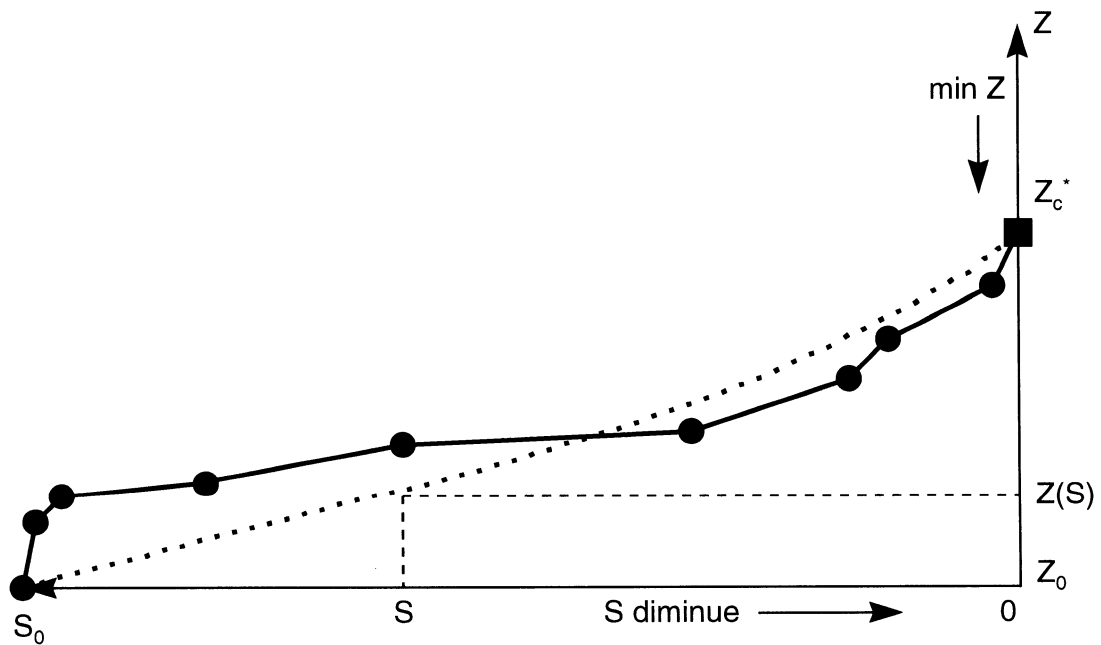


Figure 4.1: Modèle d'interpolation de la valeur de l'objectif entre la solution du noeud-racine et la meilleure solution entière courante

(S_0, Z_0) et $(0, Z_c^*)$ par le modèle (4.1). Ce modèle reste très comparable au dernier modèle (2.13) de Taha [57]. Le choix des paramètres S et α et la sélection de points de référence pour les évaluer sont décrits ci-dessous.

4.1.1.1) Références de l'extrapolation

L'objectif Z_{lr} est celui de la solution relaxée au noeud courant et sa distance fractionnaire est notée S_{lr} . L'objectif espéré Z_{esp} représente l'objectif prédit pour la solution entière optimale du sous-arbre courant. Tout au long du calcul de la solution entière, de la racine jusqu'à la feuille, chaque noeud intermédiaire peut fournir une extrapolation additionnelle de cet objectif. Trois types d'extrapolation ont été considérés: l'extrapolation naturelle globale, l'extrapolation naturelle locale et l'extrapolation optimiste.

L'extrapolation naturelle globale prend pour points de référence le noeud initial et le noeud courant. L'objectif espéré est évalué en extrapolant la tendance de (4.1) jusqu'à ce que la solution soit entière:

$$Z_{esp} = \left(\frac{Z_{lr}^{\left(\frac{S_0}{S_0 - S_{lr}}\right)}}{Z_0^{\left(\frac{S_{lr}}{S_0 - S_{lr}}\right)}} \right) = \left(\frac{Z_{lr}^{S_0}}{Z_0^{S_{lr}}} \right)^{\frac{1}{S_0 - S_{lr}}} . \quad (4.3)$$

L'extrapolation naturelle globale a été abandonnée car bien souvent les contraintes dynamiques ou d'autres aspects modifient le comportement initial du processus de calcul. Dans la majorité des problèmes contenant des contraintes traitées dynamiquement, le noeud initial est immédiatement corrigé par l'addition de plusieurs coupes. Il en résulte une rapide détérioration initiale de l'objectif avant qu'il

n'obéisse au modèle (4.1). Les extrapolations naturelles globales deviennent alors toutes artificiellement pessimistes, ce qui nuit à l'ordonnancement des branches de recherche selon les plus prometteuses. Ce phénomène est illustré à la figure 4.2 où les trois techniques d'extrapolation sont présentées. Deux branches y sont représentées: la branche menant à la meilleure solution entière connue et la branche courante contenant les noeuds indiqués par des cercles.

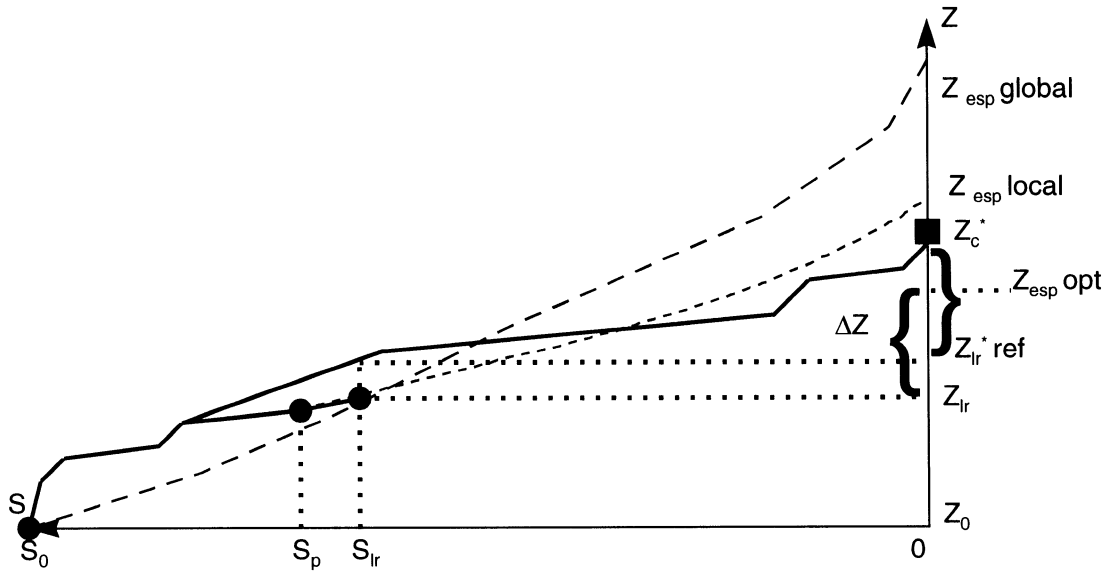


Figure 4.2: Comparaison des techniques d'extrapolation de l'objectif espéré

L'extrapolation naturelle locale effectue le même calcul mais à partir du noeud-parent et du noeud courant. En conséquence, elle est plus précise en moyenne mais plus dispersée au cours de l'évolution vers la solution entière. En utilisant l'indice p pour désigner le noeud-parent, l'expression de l'extrapolation devient:

$$Z_{esp} = \left(\frac{Z_{lr}^{S_p}}{Z_p^{S_{lr}}} \right)^{\frac{1}{S_p - S_{lr}}} . \quad (4.4)$$

L'extrapolation locale, bien que moins stable le long d'une même branche, prédit en moyenne avec plus de précision les différences entre deux objectifs espérés à comparer. Toutefois, lorsque l'extrapolation naturelle locale prédit une valeur supérieure à la meilleure solution entière courante, on sait que la branche considérée est en retard par rapport à la branche menant à cette même solution. Ces décisions moins efficaces pour atteindre l'intégralité à faible coût peuvent devenir soit un gaspillage inutile si les décisions postérieures sont comparables à celles utilisées pour obtenir la meilleure solution entière trouvée, soit un investissement pour permettre un meilleur rendement ultérieurement.

Comme ce dernier cas est le seul utile, il est plus efficace de supposer que les erreurs potentielles contenues dans les noeuds non-explorés sont des investissements avant de les ordonner. On remplace alors le taux de détérioration par celui obtenu le long de la branche de la meilleure solution entière courante pour une même valeur de S_{lr} . C'est ce que nous appelons l'extrapolation optimiste. Cette extrapolation consiste à supposer que la détérioration future à partir du noeud courant suivra celle de la meilleure solution entière courante. L'objectif espéré se calcule alors simplement:

$$Z_{esp} = Z_{lr} + \Delta Z \quad (4.5)$$

$$\text{où} \quad \Delta Z = Z_c^* - Z_{lr}^{*ref} . \quad (4.6)$$

Finalement, à cause de l'emploi de décisions multiples ou d'une mesure non-régulière de l'intégralité, l'objectif Z_{lr}^{*ref} de la relaxation linéaire de même S_{lr} le long de la branche de la solution entière optimale courante doit souvent être interpolée entre un noeud-fils et un noeud-parent, d'indices respectifs f et p :

$$S_f^* < S_{lr} < S_p^* \quad (4.7)$$

et

$$Z_{lr}^{*ref} = \left(\frac{Z_p^{*S_{lr}-S_f^*}}{Z_f^{*S_{lr}-S_p^*}} \right)^{\frac{1}{S_p^*-S_f^*}}. \quad (4.8)$$

Lorsque l'objectif espéré selon l'extrapolation locale est supérieur — dans le cas d'une minimisation — à la valeur de la solution entière courante, on remplace sa valeur par l'extrapolation optimiste, plus susceptible de suggérer une meilleure solution entière. Une mise à jour additionnelle de toutes les extrapolations optimistes est cependant nécessaire chaque fois qu'une meilleure solution entière est trouvée.

4.1.1.2) Mesure de l'intégralité

Le dernier aspect à considérer est l'unité de mesure de l'intégralité. En premier lieu, le nombre de variables fractionnaires a été employé. Malheureusement, le modèle (4.1) ne semble pas répondre au comportement avec cette mesure: les prédictions sont systématiquement optimistes et l'erreur diminue au fur et à mesure que l'on se rapproche de l'intégralité. Ce comportement est illustré à la figure 4.3.

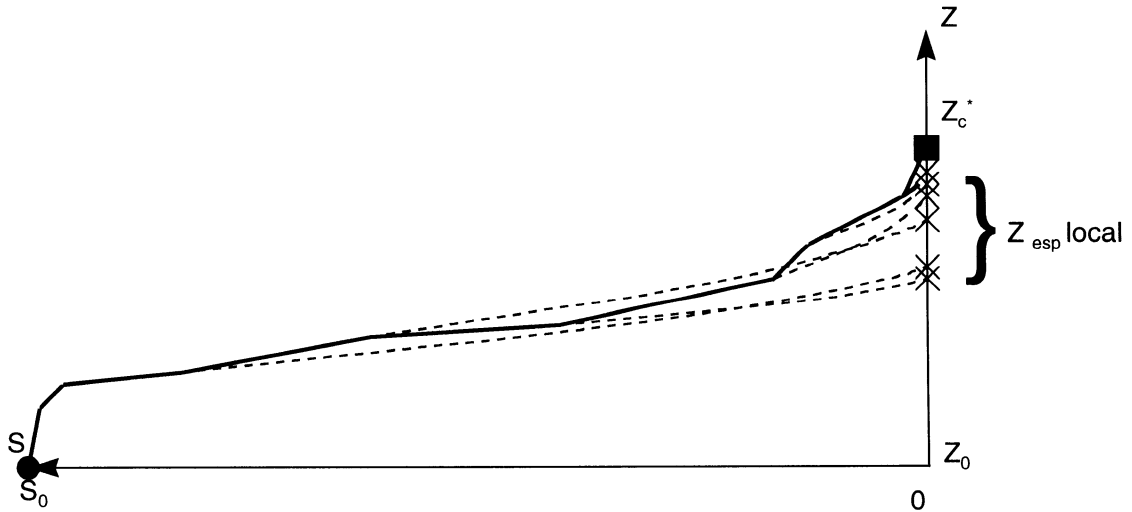


Figure 4.3: Extrapolations locales systématiquement optimistes

Le modèle (4.1) représente une détérioration composée géométriquement à chaque fois que la mesure d'intégralité avance d'une unité. En effet, l'effet de composition reflète bien le comportement anticipé des coûts réduits sur les variables d'écart des contraintes d'intégralité linéarisées — en supposant que l'on connaisse la valeur entière optimale. C'est l'augmentation de la distance fractionnaire qui n'est pas prise en compte par le nombre de variables fractionnaires. La distance fractionnaire (2.8) de Taha [57] réduit ce problème. En effet, si la base n'est modifiée que par l'ajout de la contrainte d'intégralité linéarisée à sa valeur optimale, la détérioration de l'objectif devient exactement:

$$\Delta Z_i' = \min(u_i S_i, d_i f_i) \quad \text{ou} \quad \Delta Z_i' = \bar{c}_e f_e \quad \forall x_i \text{ composante de } \bar{x}', \quad (4.9)$$

où le coût réduit $\bar{c}_e$ et la partie fractionnaire f_e sont ceux associés à la variable d'écart ou de surplus entrée en base à cause de la contrainte d'intégralité.

Un exemple numérique typique de l'impact du choix de la mesure d'intégralité est présenté. La figure 4.4 montre pour un même scénario et pour une même tactique de branchement, l'évolution de la valeur espérée de l'objectif d'une solution entière le long de la branche menant à sa meilleure solution entière. La technique d'extrapolation est du type naturelle locale. L'axe vertical indique la valeur de l'objectif espéré et l'axe horizontal la profondeur de la branche étudiée. Les valeurs des solutions entières optimales sont donc obtenus à droite au bout des branches de recherche. Les seconde et quatrième courbes indiquées dans la légende indiquent la moyenne des prévisions de l'objectif des solutions entières pour chacune des mesures d'intégralité.

En prenant le nombre de variables fractionnaires comme mesure d'intégralité, l'objectif espéré est presque toujours inférieur — trop optimiste — en comparaison de l'objectif entier finalement obtenu. L'erreur entre la moyenne des prévisions et l'objectif espéré est presque le double de celle obtenue avec la distance fractionnaire de Taha (2.8) pour mesure d'intégralité. Ce comportement caractéristique a été vérifié sur plus d'une dizaine de cas semblables. Finalement, sauf dans un des dix cas, et ce de justesse, la meilleure solution entière obtenue est toujours meilleure en utilisant la distance fractionnaire.

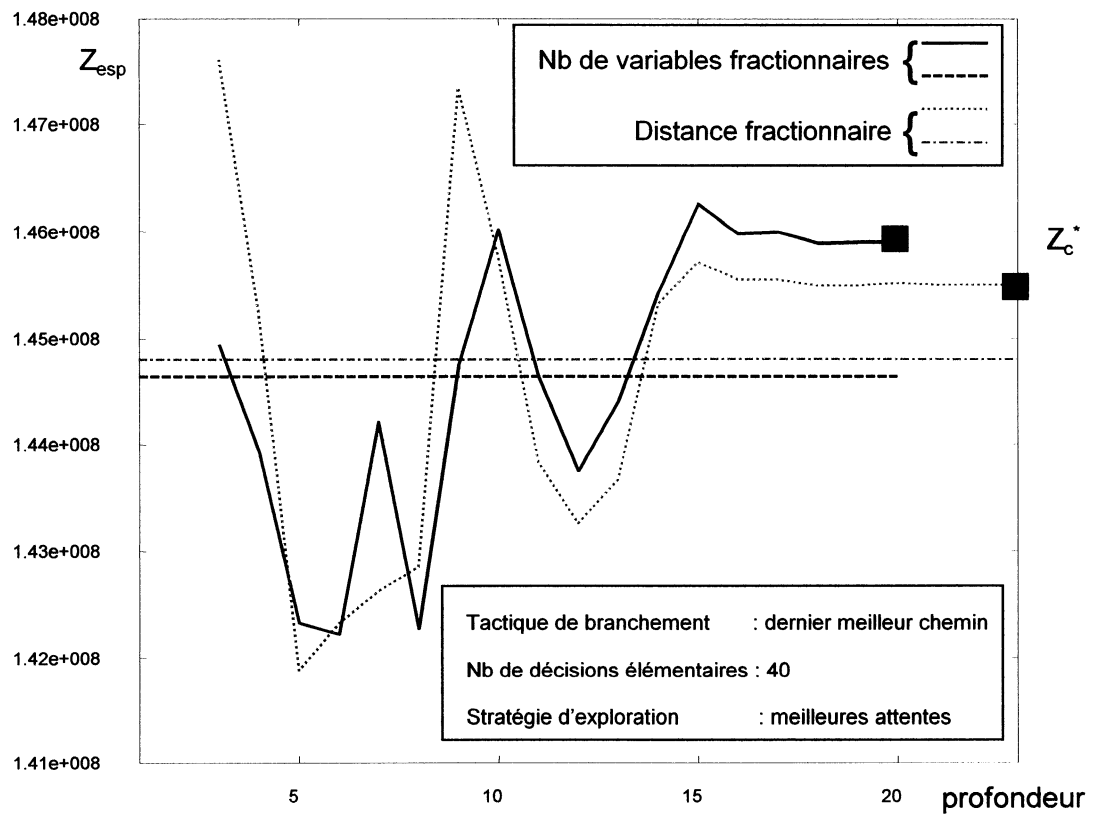


Figure 4.4: Comparaison des mesures d'intégralité

L'erreur ne provient alors plus que de l'évaluation des coûts réduits. Cet élément est rajouté au sous-chapitre 4.2.

Un dernier problème peut toutefois survenir lorsque plusieurs branches se retrouvent en compétition avec des valeurs extrapolées comparables qui se dépassent à tour de rôle. Le critère meilleur-d'abord sur les objectifs espérés introduit alors des oscillations entre tous ces noeuds consommant beaucoup de temps de calcul pour remettre à jour le problème. La section suivante présente une dernière composante pour éviter un tel gaspillage du temps de calcul.

4.1.2) Recherche en coups de sonde

Ce qui ressort essentiellement des publications concernant les techniques de recherche, c'est l'efficacité de l'alternance entre une recherche selon les meilleurs noeuds courants et les plus profonds [7]. La composante meilleur-d'abord dirige la recherche vers une région de l'arbre où il existe une bonne solution et la composante profondeur-d'abord trouve rapidement une solution. Le mélange de ces deux idées peut engendrer une stratégie efficace et robuste.

L'application du principe d'alternance entre meilleur-d'abord et profondeur-d'abord modifie ainsi la technique de recherche. Un coup de sonde, partant du meilleur noeud courant et répétant l'exploration en profondeur jusqu'à atteindre une feuille de l'arbre sans s'interroger sur la qualité des noeuds intermédiaires, introduit cet aspect d'alternance dans l'algorithme global de recherche. La répétition de ce processus devrait générer une succession de noeuds non-réalisables, entiers ou sans espoir, permettant d'identifier plus rapidement une

meilleure solution entière qu'avec une stratégie uniquement en profondeur ou meilleur-d'abord. En conséquence, aucune relaxation linéaire n'a été calculée inutilement lorsque l'exploration est interrompue par manque de temps. La figure 4.5 illustre un comportement typique de la technique de recherche en coups de sonde appliquée à l'arbre binaire de la figure 3.1.

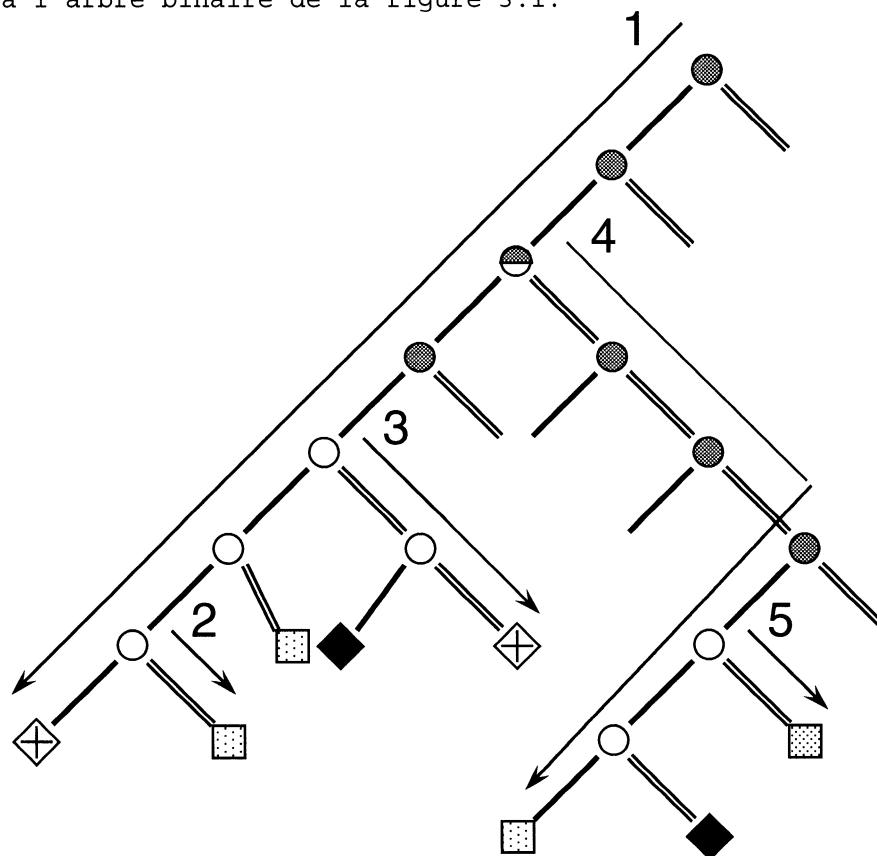


Figure 4.5: Exploration en coups de sonde

Les trois éléments précédents, soient l'extrapolation efficace d'un objectif espéré pour une solution entière, l'utilisation d'une mesure d'intégralité fiable et l'introduction de la technique de recherche en coups de sonde, se combinent pour obtenir une première version de la stratégie d'exploration des meilleures attentes.

L'algorithme suivant décrit la stratégie résultante dans un contexte de minimisation:

```

créer le noeud-racine;
tant qu'il reste un noeud non-résolu
    /* sélectionner le noeud d'objectif espéré minimal */
     $\text{noeud courant} = \operatorname{argmin} \left( Z_{\text{esp}} \mid \text{noeud non-résolu} \right);$ 
    résoudre la relaxation linéaire au noeud courant;
    /* effectuer un coup de sonde jusqu'à un noeud-feuille */
    tant que solution réalisable, fractionnaire et prometteuse  $(Z_{lr} < Z_c^*)$ 
        borner plusieurs variables continues par des valeurs entières;
        créer le noeud-fils de gauche;
        calculer  $Z_{\text{esp}}$  selon (3.38) et (2.8);
        si  $Z_{\text{esp}} \geq Z_c^*$ , corriger  $Z_{\text{esp}}$  selon (3.39)-(3.42) et (2.8);
        créer le noeud-fils de droite avec le même  $Z_{\text{esp}}$ ;
         $\text{noeud courant} = \text{noeud-fils de gauche};$ 
        résoudre la relaxation linéaire au noeud courant;
    /* mettre à jour l'arbre et la meilleure solution entière */
    si la solution courante est non-réalisable ou sans espoir
        élaguer le sous-arbre courant;
    sinon
         $Z_{\text{int}} = \text{valeur de l'objectif de la solution entière courante};$ 
        si  $Z_{\text{int}} \leq Z_c^*$ 
             $Z_c^* = Z_{\text{int}};$ 
            mettre à jour les extrapolations  $Z_{\text{esp}}$  telles que  $Z_{\text{esp}} \geq Z_c^*$ ;
            élaguer l'arbre des noeuds non-résolus tels que  $Z_{lr} \geq Z_c^*$ ;
    restituer la meilleure solution entière;

```

Les résultats obtenus avec cette stratégie sont présentés dans la section suivante.

4.1.3) Résultats numériques

Cette section contient la même analyse pour les trois tactiques de branchement et pour chacun des scénarios décrits au sous-chapitre 3.4, mais avec la stratégie d'exploration par meilleures attentes au lieu de la stratégie en profondeur-d'abord. Chaque tactique de branchement est étudiée pour deux valeurs de son paramètre, ce qui représente six combinaisons. Les graphiques suivants illustrent l'évolution de la qualité de la meilleure solution entière connue en fonction du temps de *CPU*. La mesure d'intégralité est la distance fractionnaire (2.8) proposée par Taha [57], l'objectif entier espéré est d'abord extrapolé localement puis corrigé par l'évaluation optimiste si nécessaire.

La figure 4.6 montre l'évolution de la qualité des solutions pour les six combinaisons possibles appliquées au premier scénario. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité de l'ordre de 62,5% après deux heures de calcul comparativement à la solution obtenue par l'approche initiale après 15 minutes. Rappelons qu'en profondeur-d'abord, quatre combinaisons présentaient un gain relatif moyen de 65% après deux heures, mais les deux autres une détérioration relative de 283%, pour une détérioration relative moyenne de 75,5% au total. Après vingt-quatre heures, l'amélioration relative moyenne atteint 69% comparativement à 67% en profondeur-d'abord.

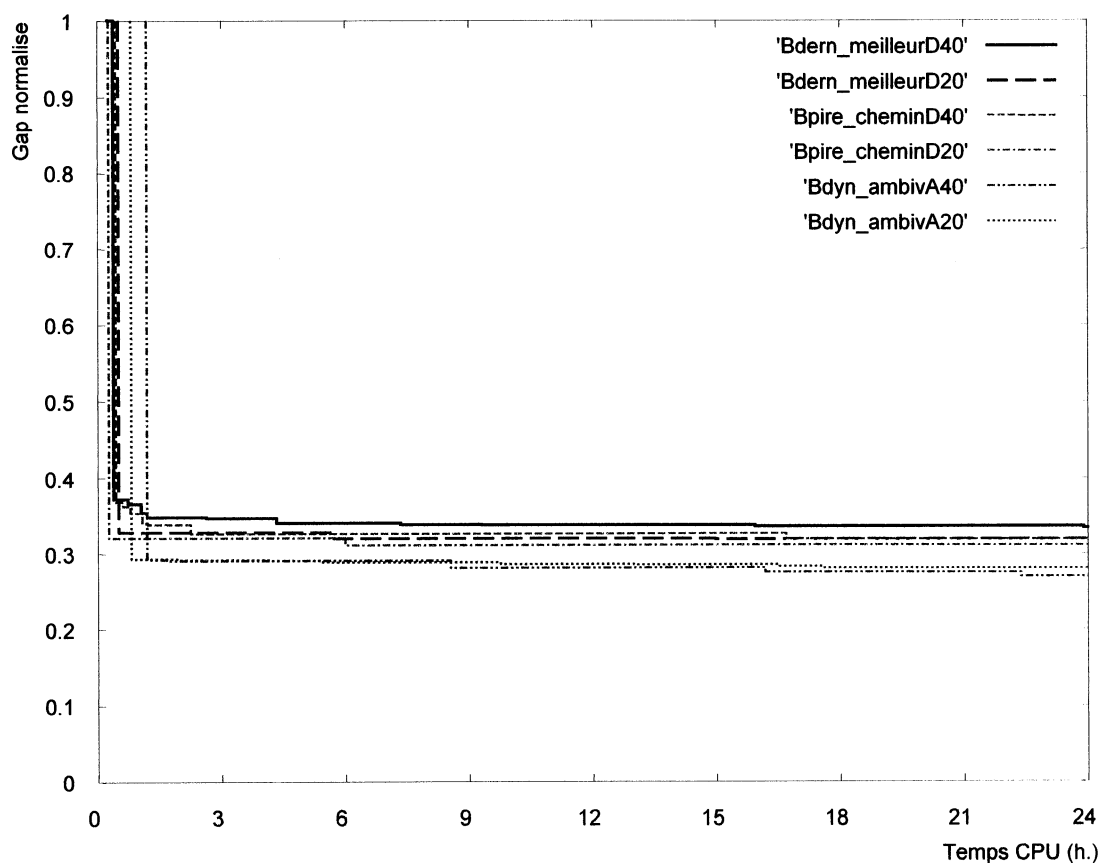


Figure 4.6: Evolution du coût de la meilleure solution entière pour le scénario 1 en stratégie d'exploration par meilleures attentes

La figure 4.7 montre l'évolution de la qualité des solutions pour ces mêmes tactiques de branchement appliquées au second scénario. La réduction moyenne du saut d'intégrité est de 22% après deux heures comparativement à la solution obtenue après 10 minutes par la méthode initiale. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité entre 23% et 35% après vingt-quatre heures de calcul. En comparaison, avec la stratégie profondeur-d'abord, l'amélioration relative du saut d'intégrité est de 22,5% en moyenne après deux heures, et entre 9% et 36 % après vingt-quatre heures.

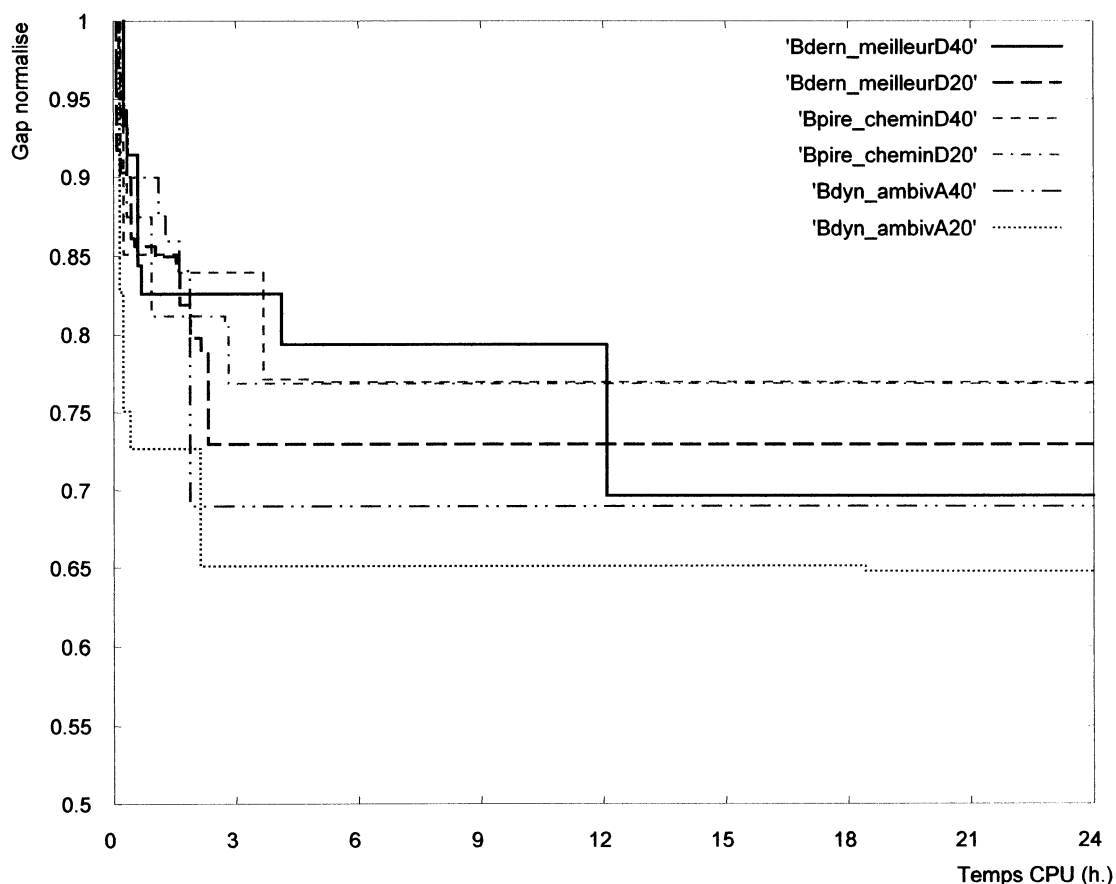


Figure 4.7: Evolution du coût de la meilleure solution entière pour le scénario 2 en stratégie d'exploration par meilleures attentes

La figure 4.8 illustre les mêmes courbes pour le dernier scénario. La réduction moyenne du saut d'intégrité est de 20,5% après deux heures comparativement à la solution obtenue après 15 minutes par la méthode initiale et au 20% de réduction moyenne en profondeur-d'abord. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité entre 27% et 32% après vingt-quatre heures de calcul pour une moyenne de 29%. Pour comparer, la stratégie d'exploration en profondeur-d'abord produit des améliorations relatives entre 7% et 30% et une moyenne de 20% pour un temps de calcul identique.

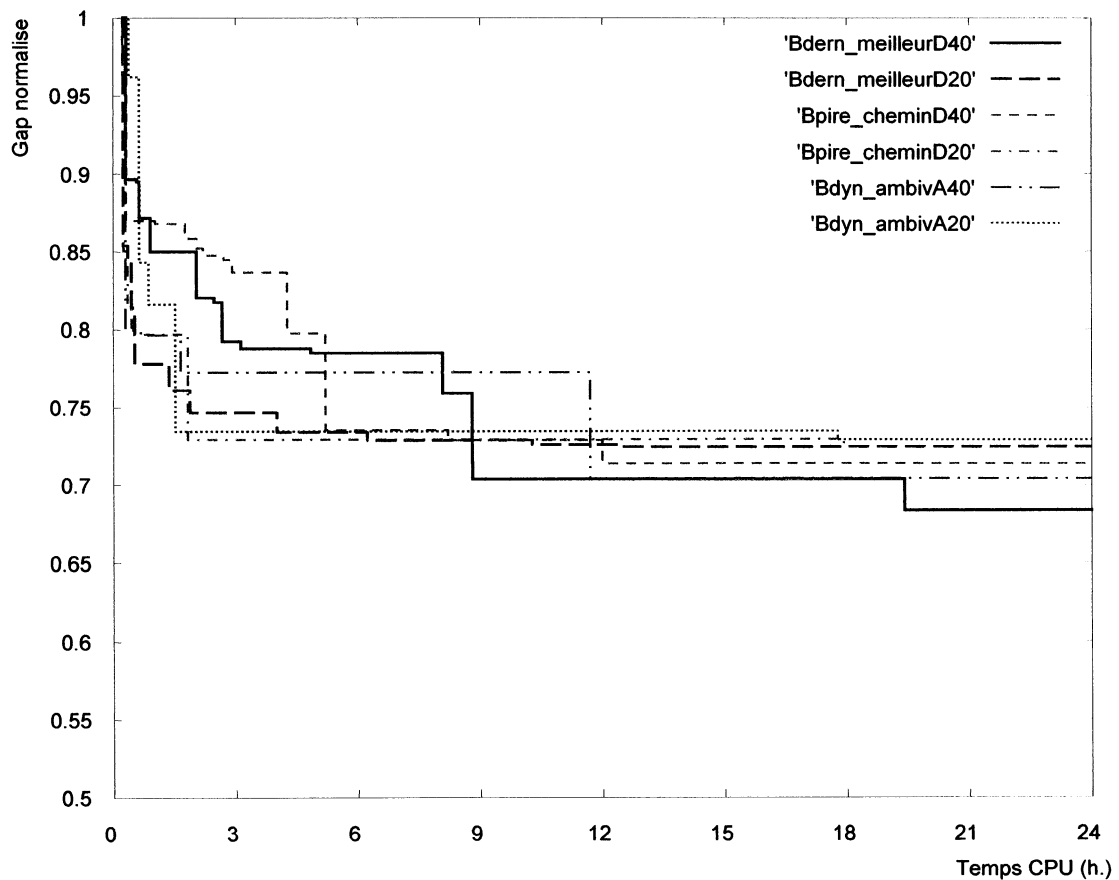


Figure 4.8: Evolution du coût de la meilleure solution entière pour le scénario 3 en stratégie d'exploration par meilleures attentes

En normalisant encore toutes les valeurs de l'objectif entre la valeur de la relaxation linéaire et le coût de la solution obtenue par la tactique de Ziarati et *al.* [64], nous proposons d'étudier encore le graphique de la moyenne arithmétique de nos trois scénarios en stratégie d'exploration par meilleures attentes, illustré à la figure 4.9. Tout d'abord, les tactiques de branchement montrent une amélioration moyenne de 37,5% après deux heures de calcul en temps *CPU*, comparativement à la détérioration de 2% de la stratégie d'exploration en profondeur-d'abord. De la même façon, l'amélioration moyenne est de 42% après vingt-quatre heures de calcul en comparaison de 38,5% pour la

stratégie en profondeur-d'abord. Si la tactique de branchement dynamique ambivalent est ignorée pour la stratégie en profondeur-d'abord — car elle y est mal adaptée, le gain relatif moyen atteignait déjà 42% pour les tactiques de branchement bien adaptées en stratégie profondeur-d'abord — *Bdern_meilleur* et *Bpire_chemin*.

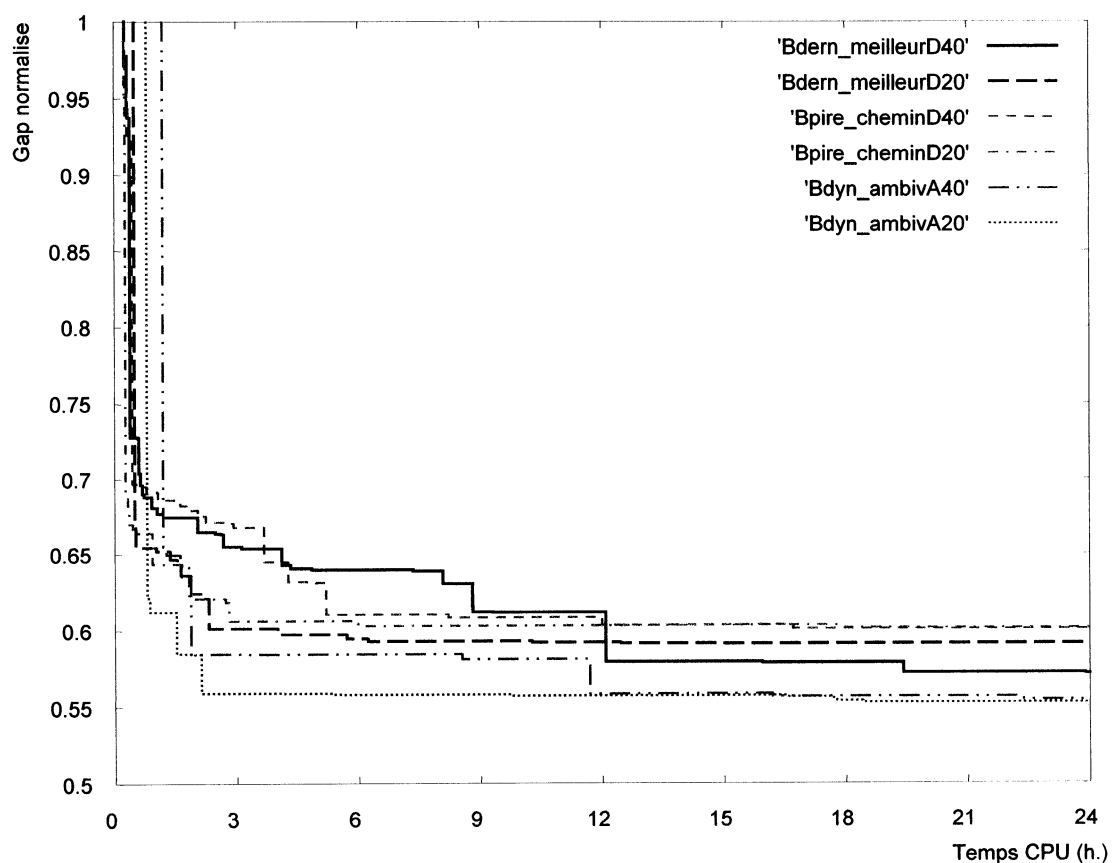


Figure 4.9: Evolution moyenne du coût de la meilleure solution entière pour les scénarios en stratégie d'exploration par meilleures attentes

Il est extrêmement difficile d'obtenir une mesure précise du temps moyen nécessaire à la stratégie de sélection par meilleures attentes pour obtenir une qualité de résultat identique à la stratégie en profondeur-d'abord. En effet, à cause du comportement discret des

Tableau 4.1: Gain de saut d'intégrité relatif après 4 heures de calculs

	Saut d'intégrité en profondeur- d'abord (%)	Saut d'intégrité en meilleures attentes (%)	Gain de saut d'intégrité relatif (%)
Scénario 1:			
<i>Bdern_meilleurD40</i>	34,48	34,65	-0,17
<i>Bdern_meilleurD20</i>	64,36	32,78	+31,58
<i>Bpire_cheminD40</i>	34,74	32,58	+2,16
<i>Bpire_cheminD20</i>	33,3	32	+1,3
<i>Bdyn_ambivA40</i>	31,2	29,03	+2,17
<i>Bdyn_ambivA20</i>	30,4	29,13	+1,27
Scénario 2:			
<i>Bdern_meilleurD40</i>	81,02	82,6	-1,58
<i>Bdern_meilleurD20</i>	71,62	72,96	-1,34
<i>Bpire_cheminD40</i>	71,63	77,12	-5,49
<i>Bpire_cheminD20</i>	78,05	76,82	+1,23
<i>Bdyn_ambivA40</i>	63,89	68,93	-5,04
<i>Bdyn_ambivA20</i>	90,94	65,07	+25,87
Scénario 3:			
<i>Bdern_meilleurD40</i>	75,62	78,75	-3,13
<i>Bdern_meilleurD20</i>	69,68	74,64	-4,96
<i>Bpire_cheminD40</i>	78,44	83,63	-5,19
<i>Bpire_cheminD20</i>	74,87	72,9	+1,97
<i>Bdyn_ambivA40</i>	84,71	77,25	+7,46
<i>Bdyn_ambivA20</i>	93,67	73,41	+20,26
<i>Moyenne</i>	64,59	59,88	+3,8

Tableau 4.2: Gain de gap relatif après 24 heures de calculs

	Saut d'intégrité en profondeur- d'abord (%)	Saut d'intégrité en meilleures attentes (%)	Gain de saut d'intégrité relatif (%)
Scénario 1:			
<i>Bdern_meilleurD40</i>	33,65	33,40	+0,25
<i>Bdern_meilleurD20</i>	33,08	31,86	+1,22
<i>Bpire_cheminD40</i>	31,8	31,83	-0,03
<i>Bpire_cheminD20</i>	31,07	31,05	+0,02
<i>Bdyn_ambivA40</i>	31,2	26,9	+4,3
<i>Bdyn_ambivA20</i>	30,4	28,04	+2,36
Scénario 2:			
<i>Bdern_meilleurD40</i>	75,26	69,63	+5,63
<i>Bdern_meilleurD20</i>	68,91	72,96	-4,05
<i>Bpire_cheminD40</i>	71,63	76,93	-5,3
<i>Bpire_cheminD20</i>	76,28	76,82	-0,6
<i>Bdyn_ambivA40</i>	63,89	68,93	-5,04
<i>Bdyn_ambivA20</i>	90,94	64,72	+26,22
Scénario 3:			
<i>Bdern_meilleurD40</i>	75,62	68,33	+7,29
<i>Bdern_meilleurD20</i>	69,68	72,44	-2,76
<i>Bpire_cheminD40</i>	74,31	71,34	+2,97
<i>Bpire_cheminD20</i>	70,61	72,39	-1,78
<i>Bdyn_ambivA40</i>	84,71	70,39	+14,32
<i>Bdyn_ambivA20</i>	93,67	72,89	+20,78
<i>Moyenne</i>	61,48	56,81	+3,66

courbes obtenues, le choix de la référence permettant d'identifier des qualités de solution comparables influence très fortement les résultats obtenus. Même en définissant un niveau de qualité propre à chacune des dix-huit courbes décrites au chapitre précédent (comme le saut d'intégrité relatif résiduel après quatre ou vingt-quatre heures de calcul par la pire stratégie), et non un niveau fixe (0,65% ou 0,6% par exemple) et en appliquant cette méthode à de nombreuses courbes, la mesure reste peu représentative. Par contre, le gain en terme de saut d'intégrité après quatre ou vingt-quatre heures demeure relativement stable quel que soit le cas étudié. Le détail de ces calculs est

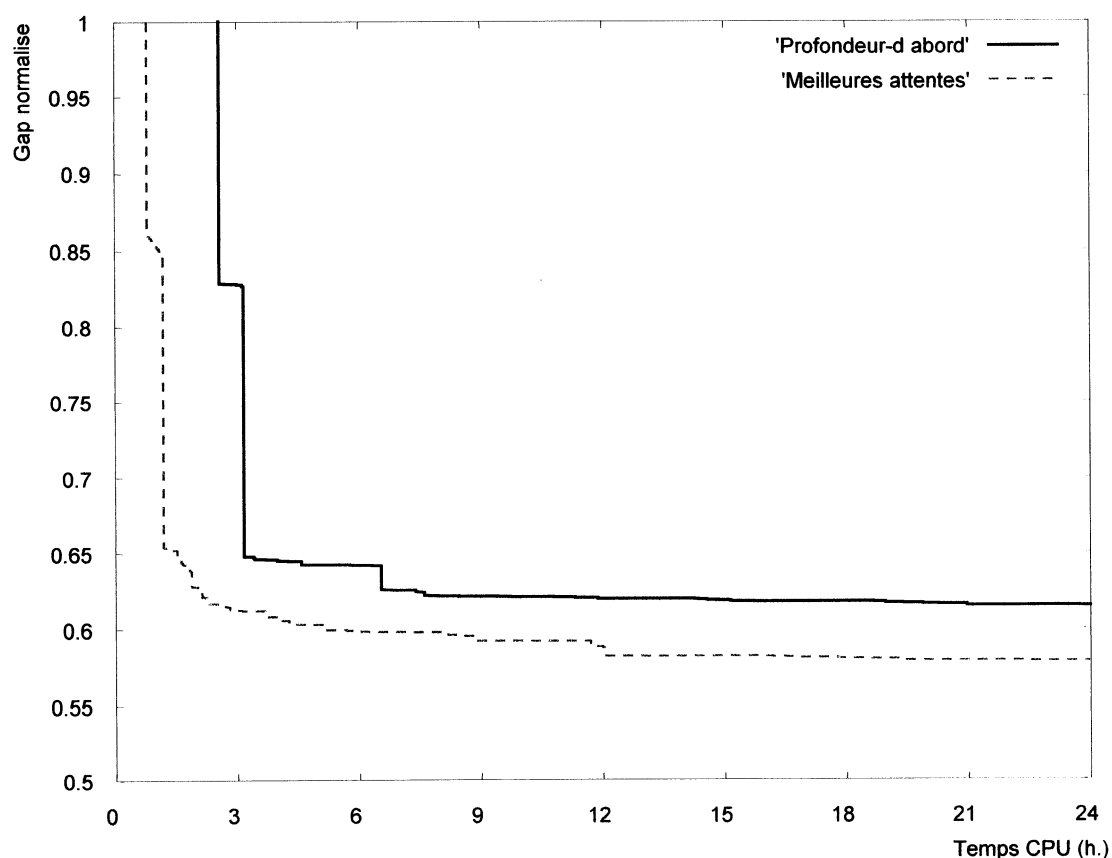


Figure 4.10: Comparaison du coût de la meilleure solution entière entre stratégies d'exploration par meilleures attentes et profondeur-d'abord

présenté aux tableaux 4.1 et 4.2. Cette évaluation suggère que la stratégie par meilleures attentes gagne en moyenne un 3,8% additionnel du saut d'intégrité relatif sur la stratégie en profondeur-d'abord après quatre heures de calcul, et 3,66% après vingt-quatre heures. Ces estimations proviennent de l'ensemble des sauts d'intégrité relatifs obtenus par les stratégies de sélection des noeuds par meilleures attentes et en profondeur-d'abord, toutes tactiques de branchement et scénarios confondus.

Ces gains moyens de saut d'intégrité paraissent aussi sur la figure 4.10 qui illustre la moyenne des dix-huit courbes pour les deux stratégies d'exploration considérées. La difficulté de gagner les derniers pourcentages de saut d'intégrité relatif apparaît clairement. Pour un cas typique régulier, la première version de la stratégie d'exploration par meilleures attentes atteint deux fois plus vite une solution entière dont le saut d'intégrité représente 65% du saut d'intégrité obtenu par Ziarati et al. [64], et conserve un avantage de 3,66% après 24 heures. Comme cet avantage est inférieur à la somme des écarts-types ($3,66 < 11,2 + 9,2 = 20,4$) des tableaux 4.1 et 4.2, toute mesure du gain de temps moyen est fortement instable. En outre, les pentes des courbes de la figure 4.10 entre 4 et 24 heures sont si faibles que l'erreur en temps devient disproportionnée. Tous ces résultats montrent toutefois que la stratégie d'exploration par meilleures attentes permet d'obtenir plus rapidement de meilleures solutions entières que celles obtenues par la stratégie en profondeur-d'abord.

Finalement, la comparaison entre les diverses tactiques de branchement est moins évidente. Toutefois, la supériorité des points pareto-optimaux de la tactique de branchement dynamique sur les chemins

ambivalents apparaît avec la stratégie d'ordonnancement par meilleures attentes. La figure 4.9 montre bien que le type de branchement est plus déterminant que le choix du paramètre — D40/D20 ou A40/A20 — pour obtenir de bonnes solutions entières rapidement. En regroupant deux à deux les échantillons de la même tactique de branchement indépendamment du paramètre, la tactique de branchement dynamique sur les chemins ambivalents apparaît supérieure à la tactique de branchement sur le dernier meilleur chemin, elle-même étant légèrement meilleure que celle interdisant le pire chemin. Le tableau 4.3 permet de compléter cette

Tableau 4.3: Caractéristiques des combinaisons étudiées

	Relaxations linéaires	Meilleures solutions	Profondeur maximale
Stratégie en profondeur d'abord:			
<i>Bdern_meilleurD40</i>	10 067,3	20,7	24,3
<i>Bdern_meilleurD20</i>	15 506	29	34,3
<i>Bpire_cheminD40</i>	10 026,7	27,3	24,7
<i>Bpire_cheminD20</i>	14 447	37,3	34,7
<i>Bdyn_ambivA40</i>	48 208	2,3	104,7
<i>Bdyn_ambivA20</i>	43 142,3	2,3	117,7
Stratégie par meilleures attentes:			
<i>Bdern_meilleurD40</i>	5 512,7	13,7	24,7
<i>Bdern_meilleurD20</i>	6 316	9,7	34
<i>Bpire_cheminD40</i>	5 439,3	8,7	26
<i>Bpire_cheminD20</i>	6 365	5,3	35,3
<i>Bdyn_ambivA40</i>	8 145,7	5	105,3
<i>Bdyn_ambivA20</i>	11 084	7,3	118,3

comparaison en indiquant les principales caractéristiques des six combinaisons étudiées: le nombre de relaxations linéaires évaluées, le nombre de meilleures solutions entières trouvées et la profondeur maximale de l'arbre d'énumération T avec les deux stratégies de recherche pour la moyenne des scénarios. Le nombre de relaxations linéaires résolues est plus important en profondeur-d'abord car le nombre de noeuds en profondeur et donc de faible taille est plus grand. Dans une moindre mesure, le même effet est constaté lorsque le paramètre d'accélération — D ou A — est faible car en prenant moins de décisions entre deux relaxations chaque réoptimisation par génération de colonnes démarre d'une solution moins perturbée et se résout plus rapidement. L'autre point intéressant est l'évidente différence de comportement de la tactique de branchement dynamique ambivalente. En stratégie profondeur-d'abord, cette tactique de branchement est mal adaptée comme en témoignent les 2,3 meilleures solutions entières trouvées en moyenne malgré un grand nombre de relaxations linéaires. La tactique de branchement dynamique ambivalente a un rythme de prise de décisions équivalent aux autres tant que les parties fractionnaires des flots de la solution relaxée sont supérieures à 0,4. Par la suite, le nombre de décisions élémentaires par décision multiple est moyennement de 3. La profondeur maximale alors obtenue est typiquement de 105 — avec $A40$ — ou 118 — avec $A20$. Pour cette tactique, le nombre de relaxations linéaires évaluées est encore plus grand, car plus de noeuds sont situés à une grande profondeur (spécialement en profondeur-d'abord) et ceux-ci prennent peu de temps, particulièrement lorsqu'un faible nombre de décisions élémentaires permet de les générer.

Toutefois, s'il est normal qu'avec le raffinement du paramètre d'accélération les résultats intermédiaires soient moins bons, au final

l'addition de précision devrait se concrétiser en de meilleures solutions. Malheureusement, les tactiques de branchement n'indiquent pas d'amélioration systématique en raffinant le paramètre de décision dans le temps imparti. Nous espérons y remédier en utilisant l'information contenue par les coûts dans le prochain sous-chapitre pour améliorer la stratégie d'exploration.

4.2) OUTILS DE RECHERCHE BASÉS SUR LES COÛTS

Les sections précédentes montrent clairement que la stratégie d'exploration par meilleures attentes est supérieure à la stratégie d'exploration en profondeur-d'abord. Toutefois, cette stratégie ne permet pas encore de classer de façon systématique les tactiques de branchement présentées au sous-chapitre 3.3. La figure 4.8 indique que pour le scénario le plus difficile de restriction d'équipements, après vingt-quatre heures de CPU, l'ordre de performance des tactiques de branchement n'est pas respecté. En outre, le raffinement des paramètres ne donne pas non plus de meilleurs résultats, quelle que soit la tactique de branchement. Les sections suivantes effectuent une révision des améliorations apportées à la stratégie d'exploration par meilleures attentes, en introduisant l'information contenue dans la fonction objectif en plus de celle représentée par les valeurs des flots fractionnaires de la solution relaxée.

4.2.1) Révision de la mesure d'intégralité

La figure 4.4 montre une amélioration significative — et systématique sur plus d'une dizaine de cas — de l'erreur moyenne obtenue par les extrapolations successives de la valeur de la fonction

objectif d'un noeud-feuille représentant une solution entière le long d'une même branche de l'arbre de recherche T . Toutefois, la proportion d'extrapolations trop optimistes représente toujours le $3/4$ du nombre d'extrapolations évaluées avec la première version de la stratégie d'exploration par meilleures attentes. En intégrant la notion de coût à la définition de la distance fractionnaire, nous espérons améliorer de nouveau la précision des prévisions de la valeur de l'objectif pour le problème en nombres entiers.

Ce comportement caractéristique a été constaté sur plusieurs cas. La figure 4.11 souligne comment ce comportement est attribuable à la superposition de plusieurs modèles de type (4.1) représentant des détériorations composées géométriquement dont les rythmes de détérioration diffèrent: les décisions de branchement sur des locomotives tirant un train ou se déplaçant à vide (sur la figure, les

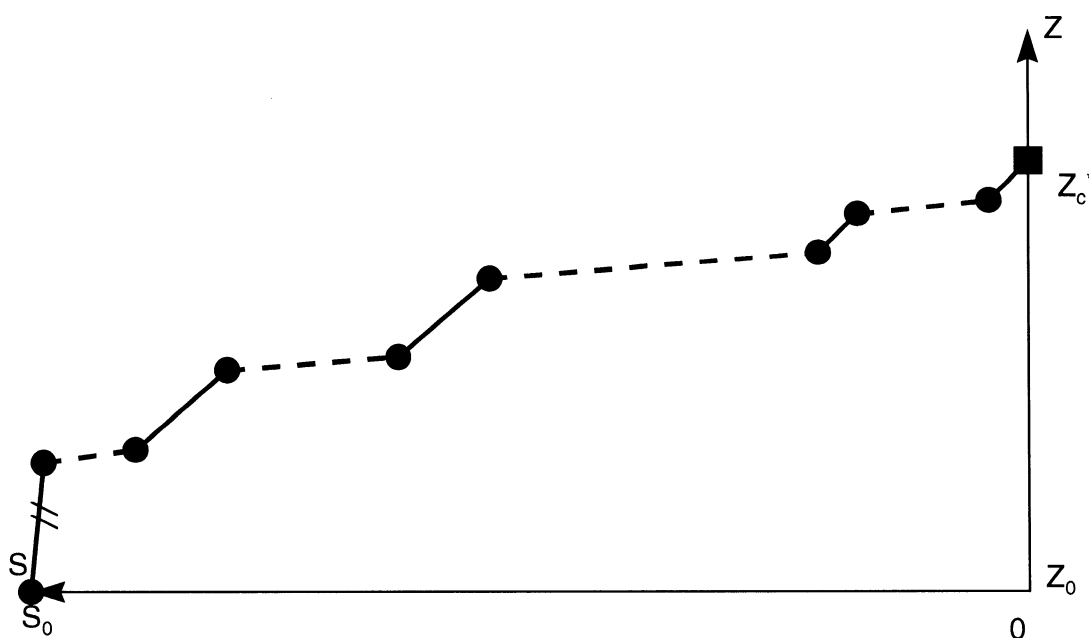


Figure 4.11: Evolution en escalier des valeurs de l'objectif de la solution relaxée

déplacements à vide sont représentés en traits d'union) n'ont pas le même impact sur la valeur de l'objectif de la prochaine solution relaxée. Dans le cas de la figure 4.4, le nombre de décisions-multiples comprenant principalement des voyages à vide est supérieur au nombre de décisions représentant l'ajout de contraintes dynamiques ou de locomotives tirant un train. On peut même considérer l'ajout de contraintes dynamiques comme une troisième catégorie de branchement avec son propre rythme de détérioration — sur la figure, la décision initiale doublement barrée représente un ajout de contraintes dynamiques.

L'essentiel consiste à estimer les poids relatifs des détériorations respectivement induites par chaque catégorie de décisions. L'équation (4.9) indique que pour une distance fractionnaire donnée, la détérioration est proportionnelle au coût réduit, si on considère que la dernière base optimale n'est pas modifiée. Toutefois il est très coûteux de calculer chacun des coûts réduits induits par les contraintes d'intégralité sur les variables de flots fractionnaires de la solution optimale lorsqu'on ajoute ces contraintes d'intégralité au problème relaxé courant. Nous suggérons donc de considérer que ces coûts réduits sont proportionnels aux coûts des variables fractionnaires, ce qui demeure une approximation valide tant qu'il reste assez de degrés de liberté pour que la variable de flot considérée soit relativement peu dépendante de ses consœurs. Cette supposition ne serait pas valide pour certains types de problèmes. En effet, le "travail" de satisfaction des contraintes de couverture ($\geq$) accompli par une unité de flot θ_p^k est mieux décrit par:

$$S_{contributions} = \sum_{v \in V} \left(\sum_{(i,j) \in p} a_{v,ij}^k \right) / a_v \quad \forall k \in K, \forall p \in \Omega^k. \quad (4.10)$$

Idéalement, il faudrait sélectionner des variables à faible coût satisfaisant fortement les contraintes. Toutefois, pour le problème qui nous intéresse, les locomotives contribuant fortement aux demandes en *HP* et *DP* sont aussi les plus coûteuses à exploiter. Donc la satisfaction relative des contraintes peut être approximée par le coût. Dans ce cas, les poids relatifs des détériorations peuvent être considérés proportionnels aux coûts. Comme la détérioration totale est connue, le modèle d'interpolation (4.1) demeure valide si la distance fractionnaire est pondérée des coûts :

$$S = \sum_{x_i \in \bar{x}^v} c_i \min(f_i, 1 - f_i). \quad (4.11)$$

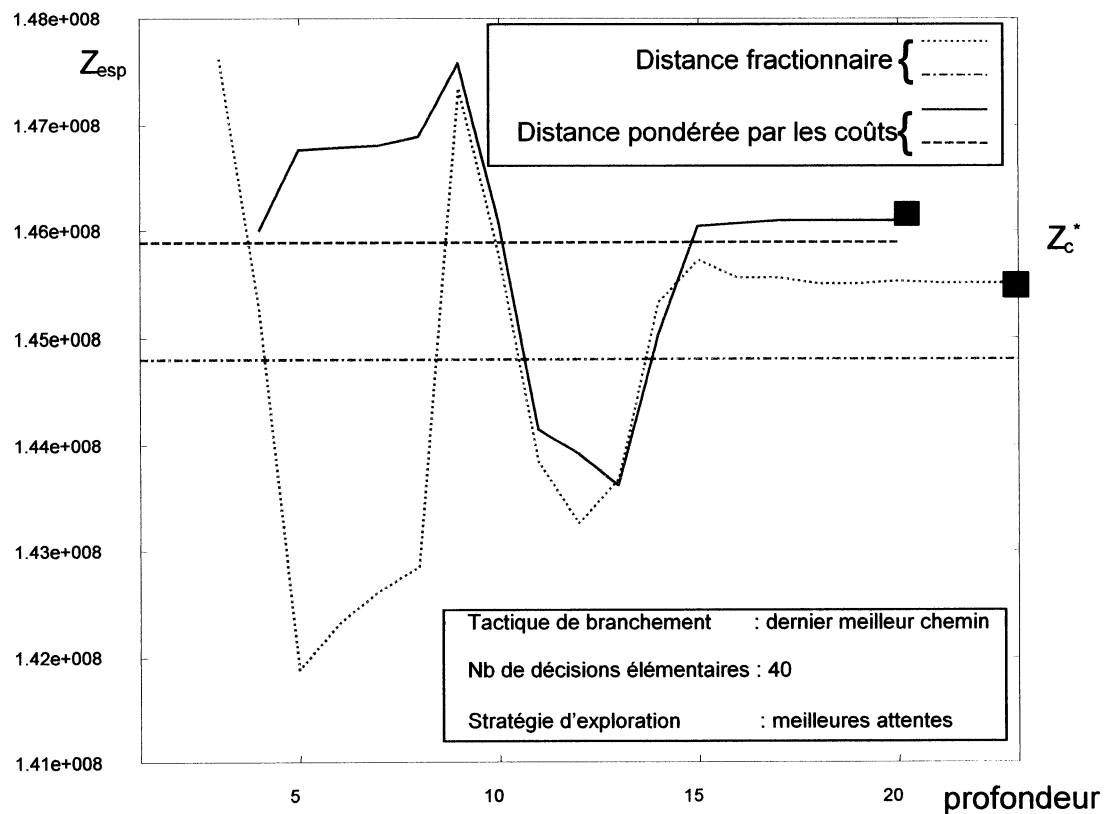


Figure 4.12: Comparaison des mesures d'intégralité entre les deux versions de la stratégie d'exploration par meilleures attentes

La figure 4.12 montre pour un même scénario et pour une même tactique de branchement, l'évolution de la valeur espérée de l'objectif d'une solution entière le long de la branche menant à sa meilleure solution entière. L'axe vertical indique la valeur de l'objectif espéré et l'axe horizontal la profondeur de la branche étudiée. Les valeurs des solutions entières optimales sont donc obtenus à droite au bout des branches de recherche. Les seconde et quatrième courbes identifiées dans la légende indiquent la moyenne des prévisions de l'objectif des solutions entières pour chacune des mesures d'intégralité. La seconde version de la méthode d'exploration par meilleures attentes emploie la nouvelle mesure d'intégralité (4.11).

L'erreur entre la moyenne des prédictions et la valeur de l'objectif de la solution entière est divisée par trois dans l'exemple précédent. En outre, ces prédictions sont également réparties autour de la valeur de l'objectif de la solution entière. Cette amélioration a pu être vérifiée pour plusieurs cas. La meilleure solution entière obtenue n'est toutefois pas systématiquement meilleure que celle obtenue par la première version, ce qui suggère une limite à l'utilité d'améliorations supplémentaires pour la stratégie d'exploration.

4.2.2) Résultats numériques

Cette section contient la même analyse qu'au sous-chapitre 3.4 pour les trois tactiques de branchement et pour chacun des scénarios décrits, mais avec la stratégie d'exploration par meilleures attentes corrigée. Chaque tactique de branchement est étudiée pour deux valeurs de son paramètre, ce qui représente six combinaisons. Les graphiques suivants illustrent l'évolution de la qualité de la meilleure solution

entière connue en fonction du temps *CPU*. La mesure d'intégralité est la distance fractionnaire pondérée (4.11).

La figure 4.13 montre l'évolution de la qualité des solutions pour les six combinaisons possibles appliquées au premier scénario. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité de l'ordre de 68% après deux heures de calcul. Rappelons qu'avec la première version de la stratégie d'exploration par meilleures attentes, le gain est de 62,5%. Après vingt-quatre heures, l'amélioration relative moyenne atteint 70%, comparativement à 69% avec la version précédente.

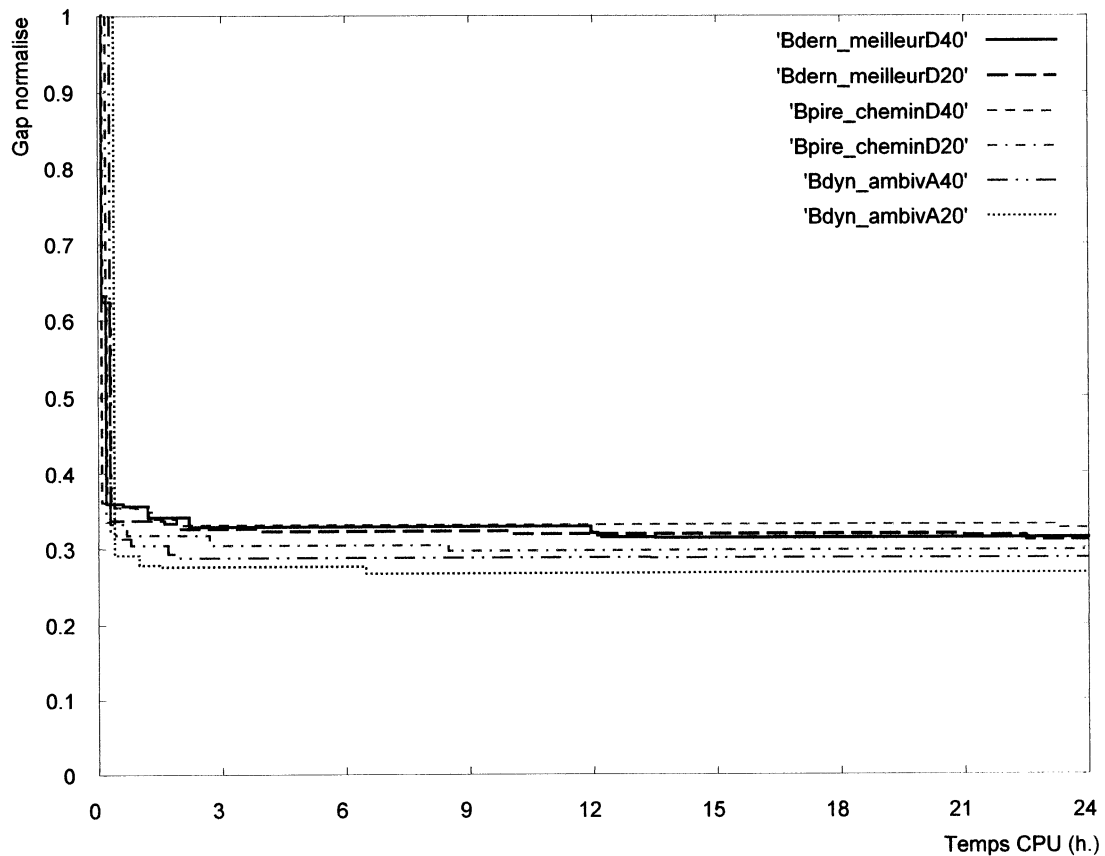


Figure 4.13: Evolution du coût de la meilleure solution entière pour le scénario 1 en stratégie d'exploration par meilleures attentes version 2

La figure 4.14 montre l'évolution de la qualité des solutions pour ces tactiques de branchement appliquées au second scénario. La réduction moyenne du saut d'intégrité est de 19% après deux heures. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité entre 18% et 41% après vingt-quatre heures de calcul. En comparaison, avec la première version de la stratégie de sélection par meilleures attentes, l'amélioration relative du saut d'intégrité est de 22% en moyenne après deux heures, et entre 23% et 35% après vingt-quatre heures.

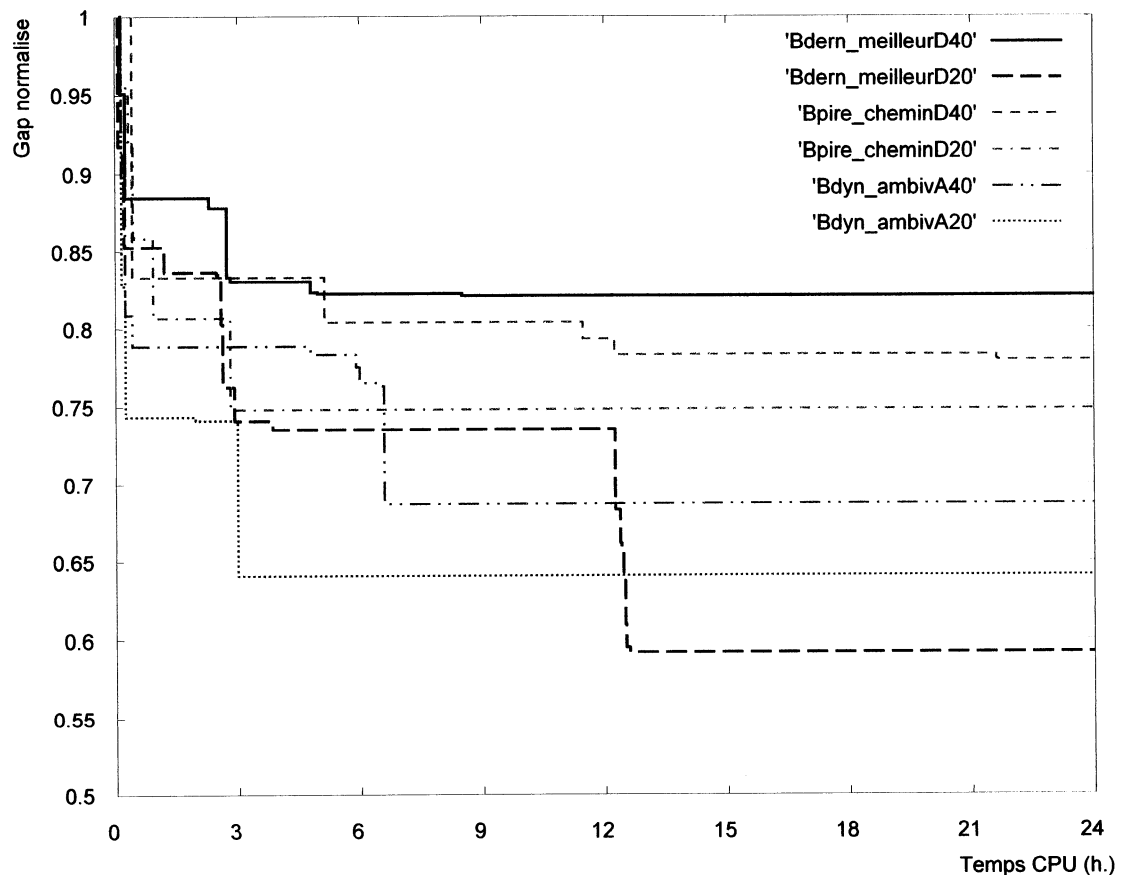


Figure 4.14: Evolution du coût de la meilleure solution entière pour le scénario 2 en stratégie d'exploration par meilleures attentes version 2

La figure 4.15 illustre les mêmes courbes pour le dernier scénario. La réduction moyenne du saut d'intégrité est de 23,5% après deux heures comparativement aux 20,5% de réduction moyenne avec la première version. Toutes les tactiques de branchement permettent une amélioration relative du saut d'intégrité entre 24% et 29% après vingt-quatre heures de calcul pour une moyenne de 27,5%. Pour comparer, la première version produit des améliorations relatives entre 27% et 32% et une moyenne de 29% pour un temps de calcul identique.

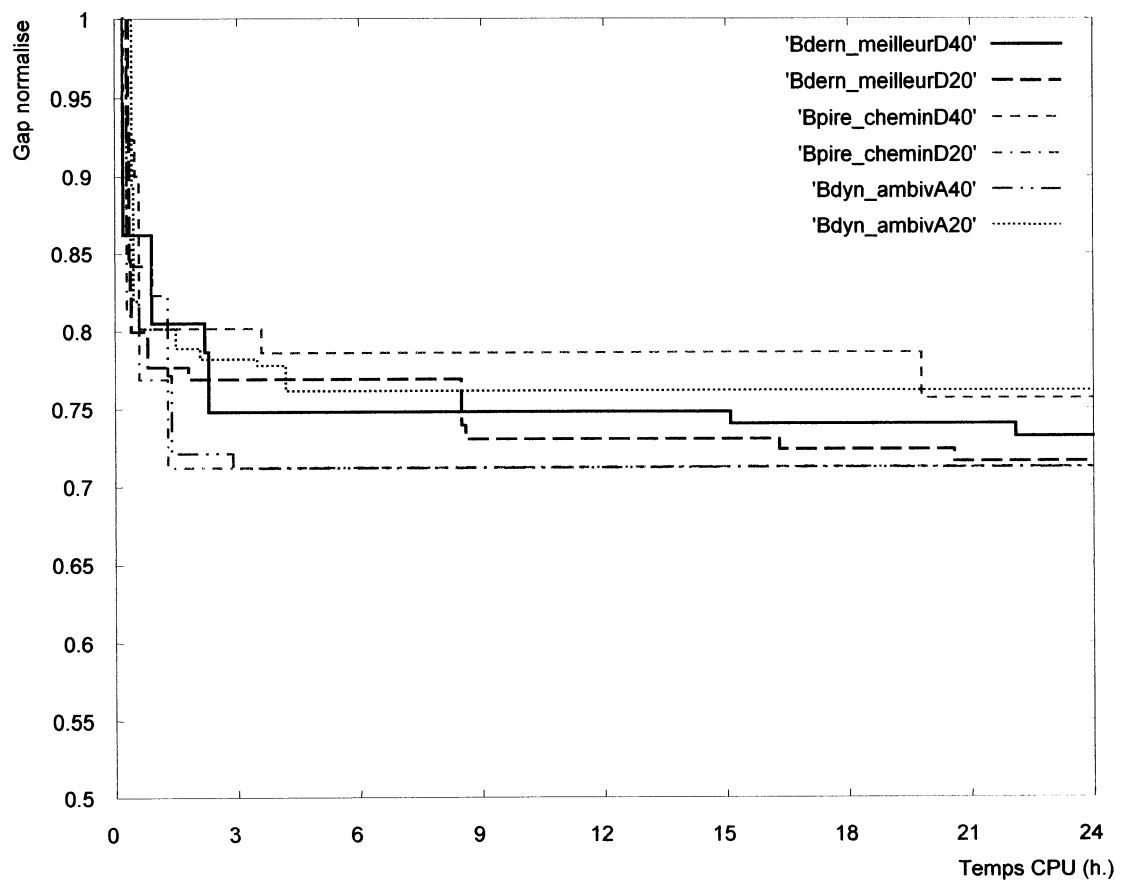


Figure 4.15: Evolution du coût de la meilleure solution entière pour le scénario 3 en stratégie d'exploration par meilleures attentes version 2

En normalisant à nouveau toutes les valeurs de l'objectif entre la valeur de la relaxation linéaire et le coût de la solution obtenue par la tactique de Ziarati et *al.* [64], nous proposons d'étudier une dernière fois le graphique de la moyenne arithmétique des trois scénarios en stratégie d'exploration par meilleures attentes, illustré à la figure 4.16. Soulignons tout d'abord qu'avec cette version de la stratégie d'exploration par meilleures attentes, le raffinement du paramètre d'accélération induit de meilleures solutions pour toutes les tactiques de branchement. Cela n'est toutefois valide que pour la moyenne des scénarios et non pour chaque exécution. Tous les gains

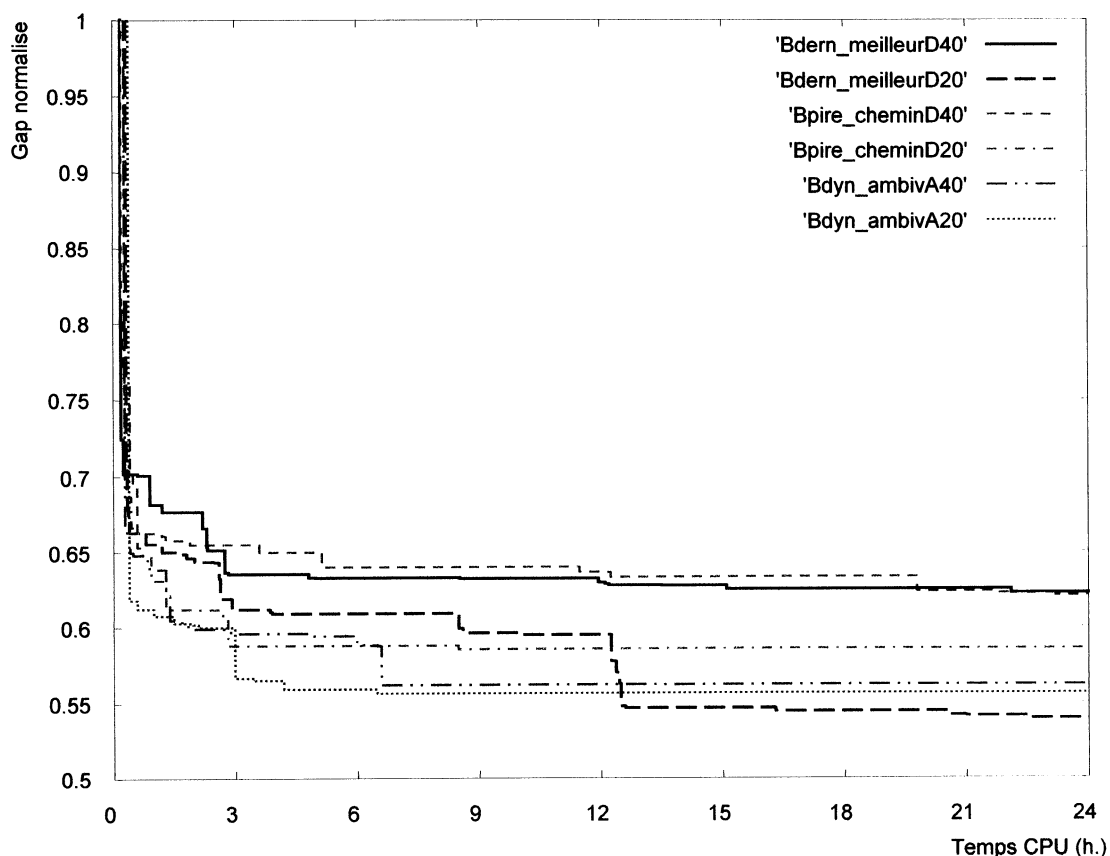


Figure 4.16: Evolution moyenne du coût de la meilleure solution entière pour les scénarios en stratégie d'exploration par meilleures attentes version 2

moyens de saut d'intégrité normalisé après deux heures, quatre heures ou vingt-quatre heures de temps *CPU* sont identiques à ceux obtenus avec la première version de la stratégie d'exploration par meilleures attentes.

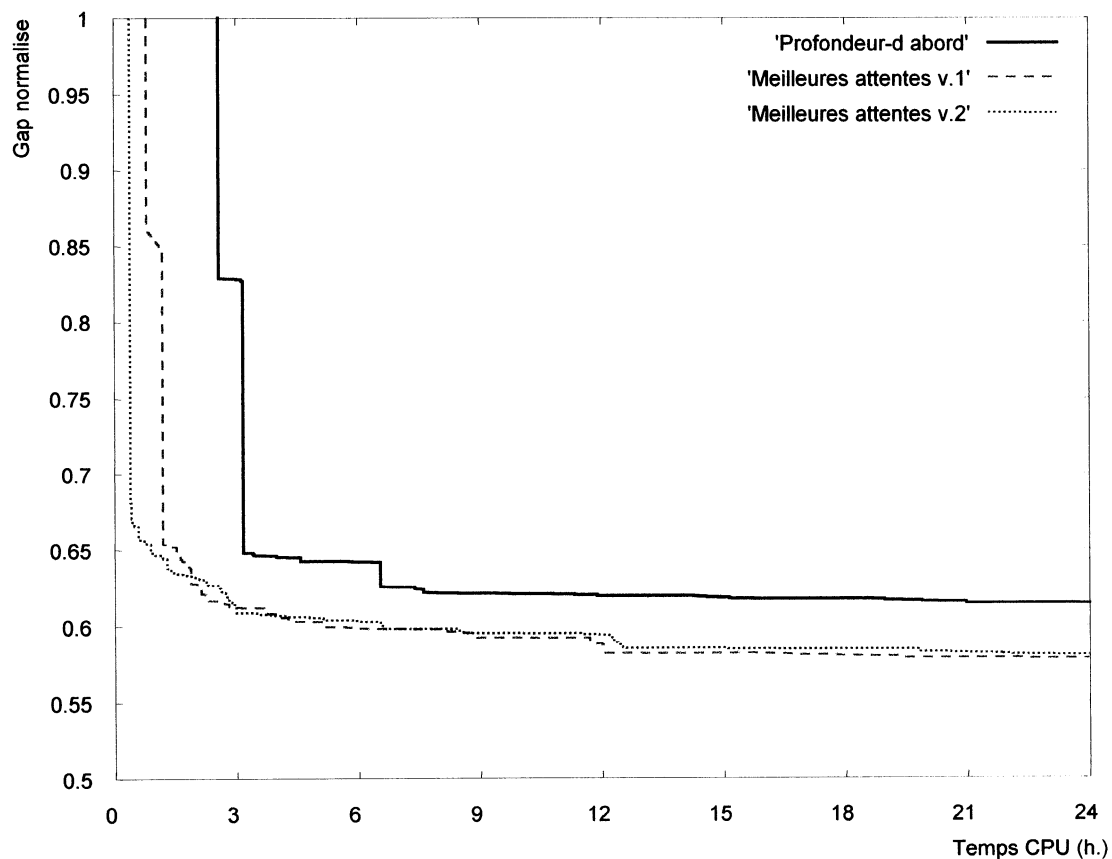


Figure 4.17: Comparaison du coût de la meilleure solution entière entre les diverses stratégies d'exploration

Ces gains moyens de saut d'intégrité paraissent aussi sur la figure 4.17 qui illustre la moyenne des 18 courbes pour chacune des stratégies d'exploration considérées. En moyenne, l'évolution du coût de la meilleure solution entière courante est presque identique avec les deux versions: la seconde version de la stratégie d'exploration par

meilleures attentes obtient seulement un léger gain de temps de calcul en début de résolution. Ces résultats semblent confirmer la supériorité de la distance fractionnaire pondérée lorsqu'elle est évaluée aux noeuds de faible profondeur, mais montrent que toute nouvelle amélioration de la stratégie d'exploration influence à peine la qualité de la solution entière à plus long terme.

Nous espérons donc améliorer ces solutions en révisant nos tactiques de branchement en introduisant une analyse de la fonction de coût au prochain chapitre. Par ailleurs, dans la suite de cette étude, on ne considère plus la tactique de branchement sur le pire chemin puisqu'elle semble définitivement moins efficace que les deux autres.

4.2.3) Révision des références d'extrapolation

Les modifications suggérées dans cette section n'ont pas engendré d'amélioration notable des résultats. Toutefois, puisqu'elles permettent de réduire encore l'erreur entre la moyenne des prédictions et la valeur de l'objectif de la solution entière, elles seront utilisées après de prochains changements apportés aux tactiques de branchement.

La figure 4.11 illustre un autre problème récurrent pour tous les scénarios traités à la section 4.1.3: le noeud-racine corrigé par contraintes dynamiques induit une très forte détérioration initiale. Celle-ci fausse l'extrapolation globale car le comportement modélisé par l'équation (4.1) n'apparaît régulier qu'après avoir éliminé les contraintes dynamiques. Ces contraintes permettent de resserrer la relaxation linéaire du problème initial mais induisent leur propre rythme de détérioration de l'objectif, nettement supérieur. Toutefois,

l'ajout de contraintes dynamiques ne génère qu'un seul noeud-fils et il est possible de considérer le dernier noeud-fils unique issu du problème initial comme le nouveau noeud-racine. Cette nouvelle référence est mieux adaptée au rythme des détériorations qui suivent.

Le dilemme entre les extrapolations globale (cette fois non-entachée par l'ajout de contraintes dynamiques) et locale réapparaît donc. L'extrapolation globale permet de stabiliser les prévisions et l'extrapolation locale récupère une meilleure approximation du rythme de détérioration future. Toutefois, il ne faut pas perdre de vue que le but visé est l'ordonnancement le plus exact des noeuds non-explorés et donc si l'erreur introduite ne perturbe que peu l'ordre de sélection des noeuds, elle est négligeable. En ce sens, l'extrapolation globale, maintenant que le noeud-racine est corrigé, n'introduit plus une erreur variable selon la profondeur et sera plus efficace. L'extrapolation locale aussi est nettement stabilisée en utilisant une mesure pondérée des coûts pour évaluer la distance fractionnaire. Les deux méthodes semblent acceptables et une méthode intermédiaire le devient aussi.

Nous suggérons donc d'employer une distance variable pour déterminer à partir du noeud courant le noeud-aïeul de référence: en adaptant l'extrapolant S_p avec une distance fractionnaire égale au double de la distance S_{lr} nécessaire pour atteindre une solution entière, nous espérons minimiser l'erreur d'extrapolation. Lorsque la distance fractionnaire pondérée S_{lr} est supérieure à la moitié de la distance fractionnaire pondérée du noeud-racine corrigé S_{ref} , nous retrouvons l'extrapolation globale naturelle (4.3). La figure 4.18 illustre le fonctionnement de l'extrapolation corrigée pour réduire l'erreur due à des rapports importants de coût dans la fonction

objectif. La correction optimiste (4.5) n'est pas indiquée mais est toujours employée.

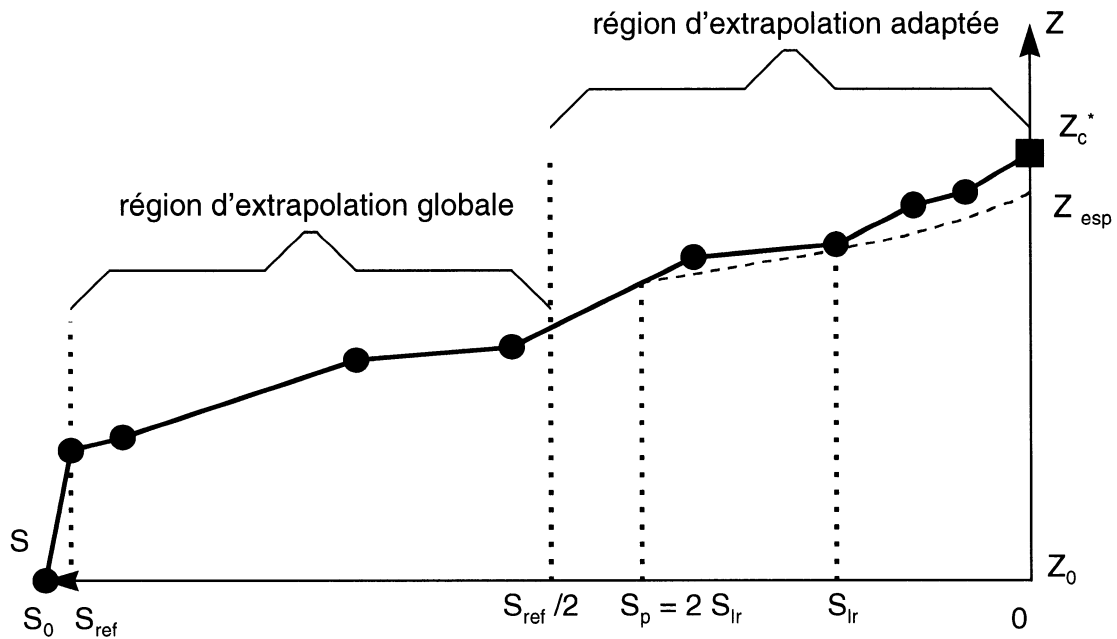


Figure 4.18: Extrapolation adaptée pour la stratégie d'extrapolation par meilleures attentes

L'extrapolation avec pas adapté est particulièrement efficace lorsque les premières décisions sont effectuées sur des variables de chemins de fort coût. En effet, l'extrapolation globale est plus susceptible de minimiser le temps de calcul pour trouver la solution entière optimale, mais l'échantillonnage qu'elle engendre est nettement plus réduit et peut nuire à la qualité des résultats intermédiaires. En prenant des décisions sur des variables de chemin de fort coût, la distance fractionnaire pondérée diminue rapidement et la région d'extrapolation adaptée est atteinte plus rapidement. En outre, une analyse des décisions induites par les décisions précédentes devrait éviter toute vallée faussement prometteuse d'où peuvent naître un grand nombre de coups de sonde générés par une extrapolation globale stable

mais erronée. Dans cette optique, l'extrapolation avec pas adapté sera introduite avec l'ajout d'une analyse de la fonction de coût ou d'une méthode de pénalité aux tactiques de branchement.

CHAPITRE 5

TACTIQUES DE BRANCHEMENT EXPLOITANT LA STRUCTURE DE COÛT

Au chapitre 4, l'introduction d'une mesure d'intégralité fondée sur les coûts dans la stratégie d'exploration nous a permis d'améliorer la prévision du coût de la prochaine solution entière. Toutefois, les résultats numériques de ce chapitre montrent que même en raffinant la qualité des prédictions des objectifs espérés, la qualité de la meilleure solution entière trouvée n'est presque plus affectée après trois heures de calcul. Nous proposons d'utiliser encore la structure de coût du problème mais cette fois-ci lors de la sélection des décisions de branchement, en espérant obtenir de meilleurs résultats comme dans le chapitre 4.

Tout d'abord, pour obtenir une meilleure estimation des détériorations engendrées par les décisions élémentaires, nous analysons séparément chaque arc employé par au moins un chemin de flot fractionnaire $\bar{\theta}_p^k$ dans la solution optimale de la relaxation linéaire. Pour chaque arc, nous définissons des problèmes linéaires simplifiés. Ceux-ci permettent d'estimer les variations de coûts associées aux décisions pour obtenir des flots entiers sur l'arc. Puis, nous transformons ces variations de coûts par arc, en détériorations de l'objectif par chemin. Ensuite, nous suggérons de nouveaux critères de sélection permettant d'identifier des décisions fiables pour accélérer le processus de décision et d'identifier une décision élémentaire ambivalente de partition.

Finalement, cette précision additionnelle permet d'accélérer encore la prise de décision en fixant des flots à des valeurs supérieures à l'unité: nous présentons donc un branchement multiple pouvant créer plus de deux noeuds-fils dans le cadre du problème de recouvrement généralisé et de son application plus spécifique au problème d'affectation de locomotives. Nous décrivons chacun de ces sujets dans les sections qui suivent.

5.1) ESTIMATION DES DÉTÉRIORATIONS

Cette section porte sur les problèmes de recouvrement généralisé, et plus spécifiquement, sur le problème d'affectation de locomotives à des trains. Pour des problèmes de type partitionnement d'ensemble, une analyse similaire est possible mais semble nettement moins prometteuse. Nous désirons évaluer les détériorations par chemin associées aux variables de flot θ_p^k pour choisir les décisions de branchement qui méritent d'être prises. Ainsi, pour le problème d'affectation de locomotives, chaque arc couvert par des flots $\bar{\theta}_p^k$ fractionnaires est étudié dans le cadre de problèmes linéaires simplifiés. Ceux-ci permettent d'estimer les détériorations causées par les décisions élémentaires liant les locomotives de la commodité k à l'arc $(i, j) \in A^k$ considéré. Pour effectuer ce calcul, nous proposons d'isoler chaque arc du réseau associé à la commodité k et d'évaluer séparément et indépendamment les pentes des variations de coûts u_{ij}^k et d_{ij}^k associées à cet arc. On note donc respectivement Δ_{ij}^k et ε_{ij}^k les variations de

coûts associées aux décisions $x_{ij}^k \geq \lceil \bar{x}_{ij}^k \rceil$ et $x_{ij}^k \leq \lfloor \bar{x}_{ij}^k \rfloor$ à la suite de la réoptimisation du problème. De la même façon, d'autres problèmes linéaires encore plus simples sont résolus pour les autres types d'arcs. En recombinaison ces informations pour tous les arcs (de segment de train, de voyage à vide, de repos, ...) d'un chemin p , nous produisons les estimations des détériorations Δ_p^k et ε_p^k associées à un chemin p de l'ensemble Ω^k des chemins associés à la commodité k . Ces détériorations permettent alors d'établir de nouveaux critères pour choisir les décisions de notre tactique de branchement. Nous noterons les décisions élémentaires de partition selon le flot sur un arc $(i, j) \in A^k$:

$$D_{ij}^k : x_{ij}^k \geq \lceil \bar{x}_{ij}^k \rceil \quad \text{ou} \quad \bar{D}_{ij}^k : x_{ij}^k \leq \lfloor \bar{x}_{ij}^k \rfloor \quad \forall k \in K, \forall (i, j) \in A^k, \quad (5.1)$$

et selon le flot sur un chemin $p \in \Omega^k$:

$$D_p^k : \theta_p^k \geq \lceil \bar{\theta}_p^k \rceil \quad \text{ou} \quad \bar{D}_p^k : \theta_p^k \leq \lfloor \bar{\theta}_p^k \rfloor \quad \forall k \in K, \forall p \in \Omega^k. \quad (5.2)$$

5.1.1) Évaluation des détériorations pour un arc

Nous décrivons tout d'abord les problèmes simplifiés pour un arc générique $(i, j) \in A^k$. Notez que nous nous contentons de construire deux problèmes simplifiés par commodité impliquée au noeud courant dans la couverture des tâches de l'arc (i, j) . Ce nombre déjà important pourrait rendre très inefficace la résolution de tous ces problèmes restreints pour identifier les décisions de branchement au seul noeud courant. Nous suggérons donc les simplifications suivantes: en dualisant toutes les contraintes du problème qui ne contiennent pas

l'arc étudié avec des multiplicateurs de Lagrange appropriés, il est possible de conserver l'essentiel de la solution précédente et de réduire énormément la taille des problèmes restreints. Pour estimer les multiplicateurs de Lagrange, les valeurs des variables duales $\tilde{\gamma}_i^k$ et $\tilde{\gamma}_j^k$ associées à la solution de la relaxation linéaire au noeud courant sont employées: ces variables duales correspondent aux contraintes de conservation de flot (1.5) appliquées respectivement aux noeuds i et j . Cette estimation n'est toutefois valable que proche de la solution optimale courante et le problème simplifié ne permet que de déterminer de petites corrections, $d\tilde{x}_{ij}^k$ sur $\bar{x}_{ij}^k$ et $d\tilde{y}_{ij}^s$ sur $\bar{y}_{ij}^s$, respectant l'une des restrictions (5.1). Toutes les variables notées à l'aide d'un tilde ($\sim$) respectent cette restriction. La solution corrigée s'exprime en fonction de $\tilde{x}_{ij}^k$ et $\tilde{y}_{ij}^s$:

$$\tilde{x}_{ij}^k = \bar{x}_{ij}^k + d\tilde{x}_{ij}^k \quad \forall k \in K, \forall (i, j) \in A^k \quad (5.3)$$

$$\tilde{y}_{ij}^s = \bar{y}_{ij}^s + d\tilde{y}_{ij}^s \quad \forall k \in K, \forall (i, j) \in A^k, \forall s \in S \quad (5.4)$$

$$\tilde{x}_{ij}^k \geq 0 \quad \forall k \in K, \forall (i, j) \in A^k \quad (5.5)$$

$$a_s \leq \tilde{y}_{ij}^s \leq b_s \quad \forall k \in K, \forall (i, j) \in A^k, \forall s \in S. \quad (5.6)$$

En remplaçant les variables du problème multiflot (1.1)-(1.8) par leur expression fractionnaire avec restriction et en retirant tous les termes constants, on construit les deux problèmes simplifiés — l'un avec $D_{ij}^{\bar{k}}$ et l'autre avec $\bar{D}_{ij}^{\bar{k}}$ — pour un arc $(i, j) \in A^{\bar{k}}$ associé à la commodité $\bar{k} \in K$ donnés:

$$\min \sum_{k \in K} (c_{ij}^k + \tilde{\gamma}_i^k - \tilde{\gamma}_j^k) d\tilde{x}_{ij}^k + \sum_{s \in S} c^s d\tilde{y}_{ij}^s \quad (5.7)$$

sous les contraintes

$$\sum_{k \in K} a_{v,ij}^k d\tilde{x}_{ij}^k + \sum_{s \in S} a_{vs} d\tilde{y}_{ij}^s = a_v - \sum_{k \in K} a_{v,ij}^k \bar{x}_{ij}^k - \sum_{s \in S} a_{vs} \bar{y}_{ij}^s \quad \forall v \in V \quad (5.8)$$

$$a_s - \bar{y}_{ij}^s \leq d\tilde{y}_{ij}^s \leq b_s - \bar{y}_{ij}^s \quad \forall s \in S \quad (5.9)$$

$$d\tilde{x}_{ij}^k \geq -\bar{x}_{ij}^k \quad \forall k \in K \quad (5.10)$$

$$D_{ij}^{\bar{k}} : d\tilde{x}_{ij}^{\bar{k}} \geq \lceil \bar{x}_{ij}^{\bar{k}} \rceil - x_{ij}^{\bar{k}} \quad \text{ou} \quad \bar{D}_{ij}^{\bar{k}} : d\tilde{x}_{ij}^{\bar{k}} \leq \lfloor \bar{x}_{ij}^{\bar{k}} \rfloor - x_{ij}^{\bar{k}}. \quad (5.11)$$

Les valeurs optimales produites par les fonctions objectifs des problèmes simplifiés génèreront les détériorations $\Delta_{ij}^{\bar{k}}$ et $\mathcal{E}_{ij}^{\bar{k}}$ cherchées pour tout arc $(i, j) \in A^{\bar{k}}$ de la commodité $\bar{k} \in K$ étudiée. Par inspection, on remarque que la restriction (5.11) étudiée devrait toujours être saturée s'il existe une solution. Ce problème simplifié ne peut malheureusement pas considérer toutes les conditions d'intégralité sur $\tilde{x}_{ij}^k$ pour plusieurs raisons. Ziarati et al. [65] contournaient cet obstacle en restreignant le nombre de commodités aux deux plus utilisées et en dénombrant les convois optimaux une seule fois en début de résolution. Nous ne voulons pas négliger les convois à plus de deux commodités. En outre, la restriction vient aussi modifier les facettes à considérer et le respect des conditions d'intégralité mène souvent à des solutions entières hors de l'hypercube unitaire entourant la solution relaxée sans restriction, ce qui invaliderait l'estimation des coûts du problème simplifié.

En effet, les coûts de l'objectif (5.7) sont ceux du problème initial corrigés par les variables duales $\tilde{\gamma}_i^k$ et $\tilde{\gamma}_j^k$ qui proviennent

des contraintes de conservation de flot (1.5) -ou (1.21) si on considère la notation spécifique aux arcs de voyage à vide. Ces variables représentent les coûts minimaux d'acheminement d'une unité de flot de la commodité k respectivement au noeud i et au noeud j . Comment estimer ces variables duales? Par définition, lorsque l'arc (i,j) contribue à un des chemins du réseau employés dans la solution optimale de la relaxation linéaire au noeud courant de l'arbre de branchement:

$$\gamma_j^k = c_{ij}^k + \gamma_i^k \quad \forall \bar{x}_{ij}^k > 0. \quad (5.12)$$

Ces variables duales ne peuvent donc pas constituer une approximation valable de la détérioration additionnelle employée dans l'objectif (5.7) qui deviendrait nulle. Le respect de la conservation du flot indique que la correction de flot $d\bar{x}_{ij}^k$ doit être enlevée au plus court chemin jusqu'au noeud i , retirée de l'arc (i,j) et être ajoutée à un autre chemin atteignant le noeud j sans emprunter l'arc (i,j) , ce qui prend en compte l'économie espérée par la libération d'une locomotive de ce type. En d'autres termes, $\tilde{\gamma}_j^k - c_{ij}^k - \tilde{\gamma}_i^k$ représente le coût de détour et $c_{ij}^k + \tilde{\gamma}_i^k$ représente le coût du plus court chemin passant par l'arc (i,j) . Nous proposons donc d'employer la valeur de γ_i^k obtenue par la relaxation linéaire du noeud de branchement courant et de corriger celle de γ_j^k en réeffectuant la dominance entre les étiquettes de plus court chemin au noeud j sans permettre de prolonger l'étiquette le long de l'arc (i,j) :

$$\tilde{\gamma}_i^k = \gamma_i^k \quad (5.13)$$

et

$$\tilde{\gamma}_j^k = \min_{i' \in \text{pred}(j) \mid i' \neq i} \{ \gamma_{i'}^k + c_{i'j}^k \}, \quad (5.14)$$

où $pred(j)$ désigne l'ensemble des noeuds prédécesseurs au noeud j . L'égalité (5.14) désigne la longueur au noeud j du plus court chemin ne passant pas par l'arc (i,j) .

Les expressions (5.13) et (5.14) devraient être adaptées dans le cas de locomotives critiques avec ressources car plusieurs étiquettes sont disponibles aux noeuds i et j avec des valeurs de ressource différentes. Dans ce cas, il serait nécessaire de pondérer les valeurs des étiquettes au noeud i par la fraction des flots des chemins correspondants. Le rapport du flot d'un chemin sur le flot total représente en effet une bonne approximation de la probabilité d'employer ce chemin dans la solution entière finale. Heureusement, dans notre modèle, le retard de maintenance sur les locomotives critiques est représenté par plusieurs arcs d'entrée de coûts différents, ce qui élimine les problèmes de plus court chemin avec ressource.

À partir de ce modèle général, nous présentons maintenant les problèmes simplifiés pour chaque type d'arc.

5.1.1.1) Détériorations pour un segment de train

La figure 5.1 illustre simplement la couverture de la demande en puissance ou en tonnage pour un train. L'exemple montre le coût Z des convois en fonction du niveau de satisfaction de la contrainte. On suppose que seulement trois types de locomotives — A, B et C —

représentés en traits d'union et classés en ordre croissant de pente ou de coût unitaire — par unité *HP* ou *DP* — sont disponibles: leur contribution respective est de 500, 700 et 350 — *HP* ou *DP*. En conséquence, le membre de droite de la contrainte peut être ramené au premier convoi réalisable, soit dans notre exemple de 1100 à 1200 — *HP* ou *DP*. Les carrés indiquent plusieurs combinaisons entières possibles. D'autres combinaisons existent et ne sont pas représentées. La solution fractionnaire optimale de cet exemple simple — indiquée par un cercle — est de 2,4 locomotives de type A. Cette solution fractionnaire nous invite à considérer la solution entière composée de 3 locomotives de type A. Cette solution entière entraîne un dépassement du membre de droite de la contrainte, c'est ce que l'on appelle le gaspillage. On obtient donc une première approximation de la détérioration entre la solution relaxée et la solution entière, soit

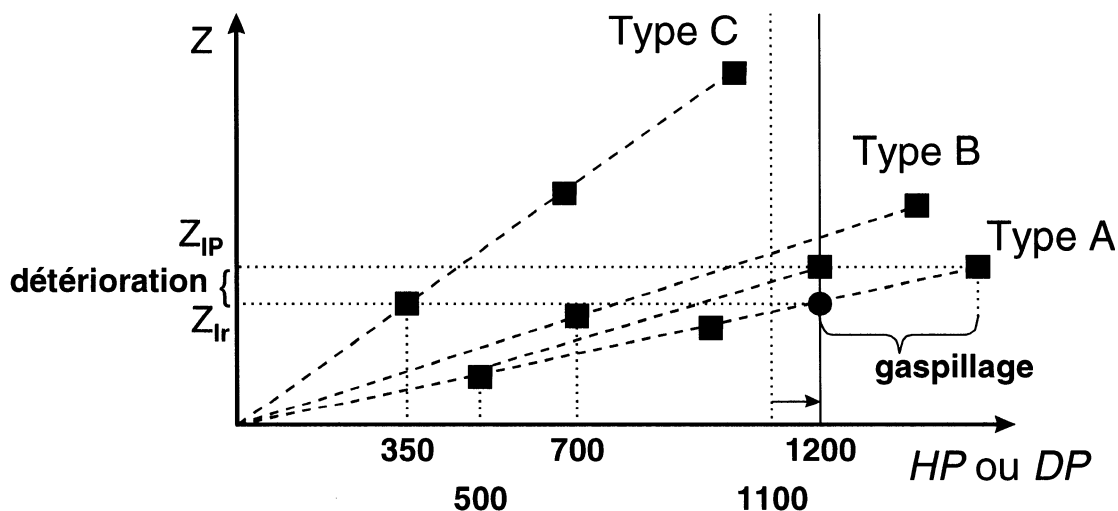


Figure 5.1: Analyse du gaspillage pour une contrainte de couverture

le produit du gaspillage et de la variable duale associée à la contrainte dans le problème en variables continues.

En réalité, le modèle est plus complexe: plusieurs contraintes — en puissance, en tonnage et parfois en unités — doivent être satisfaites et une variable continue permet jusqu'à 10% de sous-recouvrement lors de la satisfaction de ces contraintes. En ajoutant aussi les contraintes dynamiques, il est même possible d'obtenir un convoi optimal composé de plusieurs types de locomotive. Par ailleurs, l'exemple de la figure 5.1 montre qu'un autre convoi composé d'une locomotive de type A et d'une autre de type B accomplit cette même tâche à un coût identique. Ainsi, le choix du meilleur convoi devrait tenir compte de l'utilité de libérer une locomotive d'un type particulier pour tirer d'autres trains ainsi que des conditions d'intégralité si possible.

Pour chaque segment de train étudié, nous devons donc générer deux fois plus de problèmes restreints qu'il y a de commodités dont le flot sur l'arc soit fractionnaire. La figure 5.2 illustre le résultat obtenu dans un cas simple avec deux types de locomotives disponibles et donc quatre restrictions possibles. L'exemple représente par des carrés plusieurs convois possibles respectant les deux contraintes du segment de train étudié. La solution du problème relaxé se situe à l'intersection des droites définissant les deux contraintes. Les quatre autres cercles représentent les nouvelles solutions obtenues pour les problèmes modifiés relaxés. Chacune des quatre restrictions indiquées en pointillés — la quatrième s'appuie le long de l'axe

horizontal — produit une nouvelle solution relaxée qui indique une détérioration minimale de la valeur de l'objectif.

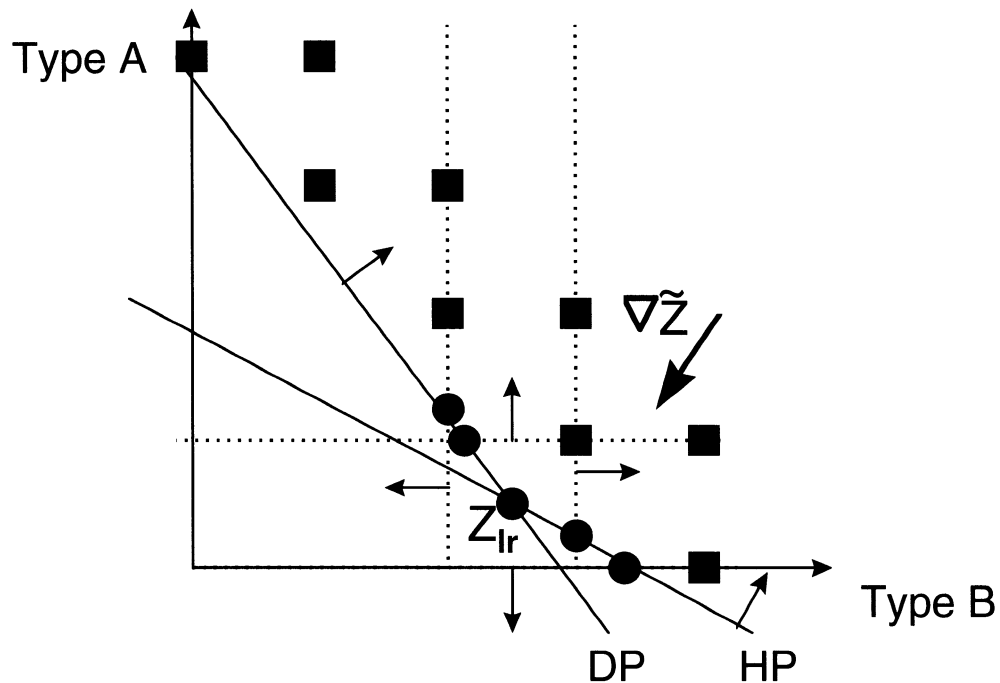


Figure 5.2: Analyse de la détérioration

Le problème original définit pour tout noeud de branchement un problème maître exprimé par les relations (1.12) à (1.19), contenant des contraintes liantes spécifiques aux trains, et plusieurs sous-problèmes décrits par les relations (1.20) à (1.25), représentant tous les chemins possibles. Pour un arc donné, nous ne conservons dans le problème simplifié que les contraintes relatives à l'arc étudié et dualisons les autres. Le problème linéaire simplifié possède une solution et une valeur objectif dont la détérioration est inférieure à celle recherchée puisqu'il ne respecte pas les contraintes d'intégralité, ce qui serait trop coûteux. Mathématiquement, les deux

problèmes simplifiés s'écrivent pour un arc $(i, j) \in W$ et une commodité $\bar{k} \in K$ donnés:

$$\min \sum_{k \in K} (c_{ij}^k + \tilde{\gamma}_i^k - \tilde{\gamma}_j^k) d\tilde{x}_{ij}^k + \sum_{s \in S} c^s d\tilde{y}_{ij}^s \quad (5.15)$$

sous les contraintes

$$\sum_{k \in K} d\tilde{x}_{ij}^k + d\tilde{y}_{ij}^n \geq n_{ij} - \sum_{k \in K} \bar{x}_{ij}^k - \bar{y}_{ij}^n \quad (5.16)$$

$$\sum_{k \in K} d\tilde{x}_{ij}^k p^k + d\tilde{y}_{ij}^{HP} \geq p_{ij} - \sum_{k \in K} \bar{x}_{ij}^k p^k - \bar{y}_{ij}^{HP} \quad (5.17)$$

$$\sum_{k \in K} d\tilde{x}_{ij}^k q^k + d\tilde{y}_{ij}^{DP} \geq q_{ij} - \sum_{k \in K} \bar{x}_{ij}^k q^k - \bar{y}_{ij}^{DP} \quad (5.18)$$

$$\sum_{k \in K} g_{m,ij}^{kd} d\tilde{x}_{ij}^k \leq n_m^d - \sum_{k \in K} \sum_{(i',j') \in A^k} g_{m,i'j'}^{kd} \bar{x}_{i'j'}^k \quad \forall m \in M, \forall d \in D \quad (5.19)$$

$$\sum_{k \in K} g_{o,ij}^{kd} d\tilde{x}_{ij}^k \geq n_o^d - \sum_{k \in K} \sum_{(i',j') \in A^k} g_{o,i'j'}^{kd} \bar{x}_{i'j'}^k \quad \forall o \in O, \forall d \in D \quad (5.20)$$

$$\sum_{k \in K} g_{h,ij}^k d\tilde{x}_{ij}^k p^k \geq p_h - \sum_{k \in K} \sum_{(i',j') \in A^k} g_{h,i'j'}^k \bar{x}_{i'j'}^k \quad \forall h \in H \quad (5.21)$$

$$-\bar{y}_{ij}^n \leq d\tilde{y}_{ij}^n \leq 0,9n_{ij} - \bar{y}_{ij}^n \quad (5.22)$$

$$-\bar{y}_{ij}^{HP} \leq d\tilde{y}_{ij}^{HP} \leq 0,9p_{ij} - \bar{y}_{ij}^{HP} \quad (5.23)$$

$$-\bar{y}_{ij}^{DP} \leq d\tilde{y}_{ij}^{DP} \leq 0,9q_{ij} - \bar{y}_{ij}^{DP} \quad (5.24)$$

$$d\tilde{x}_{ij}^k \geq -\bar{x}_{ij}^k \quad \forall k \in K \quad (5.25)$$

$$D_{ij}^{\bar{k}} : d\tilde{x}_{ij}^{\bar{k}} \geq \lceil \bar{x}_{ij}^{\bar{k}} \rceil - x_{ij}^{\bar{k}} \quad \text{ou} \quad \bar{D}_{ij}^{\bar{k}} : d\tilde{x}_{ij}^{\bar{k}} \leq \lfloor \bar{x}_{ij}^{\bar{k}} \rfloor - x_{ij}^{\bar{k}}. \quad (5.26)$$

L'objectif (5.15) du problème simplifié représente le coût des corrections apportées aux flots et aux variables d'écart dans la nouvelle solution respectant une ou l'autre des restrictions (5.26). Les contraintes (5.16) - (5.18) représentent les contraintes de couverture sur les trains de la nouvelle solution, les contraintes (5.19) - (5.21) les contraintes d'entretien, les besoins des postes locaux et la garantie de puissance minimale aux PCP. Finalement les

contraintes (5.22)-(5.24) représentent les bornes des variables d'écart et les contraintes (5.25) garantissent des flots non-négatifs.

5.1.1.2) Détériorations pour les autres types d'arcs

Une analyse similaire permet d'obtenir les problèmes simplifiés pour des arcs de voyage à vide, de repos, d'entrée et de sortie. Tous ces cas sont beaucoup plus simples car ils n'impliquent pas de contrainte liante. Pour respecter la notation particulière qui permet de différencier les arcs $(i, j) \in A^k$ de locomotive active et de voyage à vide, on utilise simplement des coûts d_{ij}^k et des corrections notées $\tilde{d}_{ij}^k$. Pour un arc de voyage à vide, le problème simplifié s'écrit:

$$\min \left(d_{ij}^{\bar{k}} + \tilde{\gamma}_i^{\bar{k}} - \tilde{\gamma}_j^{\bar{k}} \right) \tilde{d}_{ij}^{\bar{k}} \quad (5.27)$$

sous les contraintes

$$\tilde{d}_{ij}^{\bar{k}} \geq -f_{ij}^{\bar{k}} \quad (5.28)$$

$$D_{ij}^{\bar{k}} : \tilde{d}_{ij}^{\bar{k}} \geq \left\lfloor f_{ij}^{\bar{k}} \right\rfloor - f_{ij}^{\bar{k}} \quad \text{ou} \quad \bar{D}_{ij}^{\bar{k}} : \tilde{d}_{ij}^{\bar{k}} \leq \left\lceil f_{ij}^{\bar{k}} \right\rceil - f_{ij}^{\bar{k}}. \quad (5.29)$$

La solution évidente consiste à saturer la restriction (5.29).

Pour des arcs de repos, d'entrée ou de sortie, le problème simplifié devient:

$$\min \left(\tilde{\gamma}_i^{\bar{k}} - \tilde{\gamma}_j^{\bar{k}} \right) \tilde{x}_{ij}^{\bar{k}} \quad (5.30)$$

sous les contraintes

$$\tilde{x}_{ij}^{\bar{k}} \geq -x_{ij}^{\bar{k}} \quad (5.31)$$

$$D_{ij}^{\bar{k}} : \tilde{x}_{ij}^{\bar{k}} \geq \left\lfloor x_{ij}^{\bar{k}} \right\rfloor - x_{ij}^{\bar{k}} \quad \text{ou} \quad \bar{D}_{ij}^{\bar{k}} : \tilde{x}_{ij}^{\bar{k}} \leq \left\lceil x_{ij}^{\bar{k}} \right\rceil - x_{ij}^{\bar{k}}. \quad (5.32)$$

La solution évidente consiste à saturer la restriction (5.32), en respectant toutefois la contrainte dynamique de disponibilité des locomotives pour les arcs d'entrée, ce qui contribue au respect des conditions d'intégralité. Quelques cas particuliers peuvent aussi survenir quant au calcul du coût. Lorsqu'il n'y a pas de prédécesseur au noeud j autre que le noeud i , l'équation (5.14) ne peut plus être appliquée. Cette situation, impossible initialement pour un segment de train ou un voyage à vide car chacun garantit un détour à son arc jumeau, peut se concrétiser pour un arc de repos ou d'entrée. En outre, elle peut survenir pour un segment de train ou un voyage à vide plus en profondeur de l'arbre de recherche si l'arc jumeau est retiré du réseau parce que les tâches qui y sont associées sont satisfaites. Dans ce cas, l'alternative consiste à généraliser l'équation (5.14) à un noeud j' , successeur au noeud j et atteignable sans passer par j . En considérant que le trajet du noeud i jusqu'au noeud j' constitue le nouvel arc étudié, on obtient:

$$\tilde{\gamma}_{j'}^k = \min_{i' \in \text{pred}(j') \mid i' \notin j \rightarrow j'} \left\{ \gamma_{i'}^k + c_{i'j'}^k \right\}. \quad (5.33)$$

En remplaçant le trajet reliant le noeud i au noeud j' par un arc dont le coût soit la somme des coûts des arcs précédents, nous constatons que le problème simplifié équivalent revient à ignorer tout arc lorsqu'il est le seul incident à un noeud. Cette analyse n'est pas valide lorsque le retrait de l'arc (i, j) déconnecte le réseau. Toutefois, ce dernier cas n'apparaît pas dans le cadre du problème d'affectation de locomotives aux trains de marchandises.

La valeur optimale des problèmes simplifiés produite par les fonctions objectifs (5.15), (5.27) et (5.30) génère les détériorations $\Delta_{ij}^{\bar{k}}$ et $\mathcal{E}_{ij}^{\bar{k}}$ cherchées.

5.1.2) Nouveaux critères de branchement pour un chemin

Le but d'une tactique de branchement efficace est d'identifier rapidement les variables dont la valeur optimale est commune à plusieurs solutions entières satisfaisantes de l'arbre d'exploration. Nous avons déjà expliqué à la section 1.2 qu'il est nécessaire de prendre des décisions sur des variables de chemin pour préserver l'efficacité de la résolution de notre problème par une méthode de génération de colonnes. En effet, puisque les différentes locomotives d'une même commodité ne suivent pas le même chemin, il serait très complexe d'imposer des restrictions sur des arcs à seulement certaines d'entre elles. Donc, pour identifier les décisions qui génèrent le meilleur sous-arbre candidat pour contenir la solution entière optimale, les détériorations précédentes doivent être associées aux variables de chemin. Le calcul des détériorations par chemin à partir des détériorations par arc est ainsi présenté dans la section suivante. En outre, pour tirer profit de notre stratégie d'exploration, la tactique de branchement identifie aussi la décision la plus ambivalente, capable d'introduire le plus de variété dans notre recherche à peu de coût. Nous suggérons donc deux nouveaux critères prenant en compte les estimations des détériorations Δ_p^k et $\mathcal{E}_p^k$,

respectivement obtenues en prenant les décisions D_p^k ou $\bar{D}_p^k$: l'un pour sélectionner plusieurs décisions élémentaires aptes à former une décision multiple commune et de qualité, l'autre pour identifier la meilleure décision de branchement ambivalente pour générer deux sous-arbres prometteurs fortement différents.

5.1.2.1) Détériorations pour un chemin

Jusqu'ici les décisions des tactiques de branchement ne tenaient compte que des valeurs des flots fractionnaires de la solution du problème relaxé. Ainsi, on utilisait le critère de la section 4.2 qui suppose que les détériorations unitaires sont proportionnelles aux coefficients de la fonction objectif. Dans ce chapitre, l'estimation précédente des détériorations par arc permet de tenir compte de la structure de coût du problème et de ses réseaux pour établir l'ordre de priorité entre les diverses décisions de branchement disponibles.

Lorsque nous désirons augmenter le flot le long du chemin p , nous devons forcer le flot sur chacun de ces arcs à atteindre le flot recherché sur le chemin p et donc la détérioration totale est la somme de chacune des détériorations par arc. Toutefois, si le flot sur l'arc est supérieur au flot sur le chemin car il contribue aussi à d'autres chemins de flot non-nul, la détérioration doit être réduite en conséquence. Pour un chemin p dont le flot $\bar{\theta}_p^k$ est fractionnaire:

$$\Delta_p^k = \begin{cases} 0 & \text{si } \lceil \bar{\theta}_p^k \rceil < \bar{x}_{ij}^k \\ \sum_{(i,j) \in p} \Delta_{ij}^k & \text{si } \lceil \bar{\theta}_p^k \rceil > \bar{x}_{ij}^k \end{cases} \quad \forall k \in K, \forall p \in \Omega^k. \quad (5.34)$$

Lorsque le flot le long du chemin doit être réduit, il suffit de restreindre le flot sur au moins un arc du chemin p pour obtenir la réduction de flot désirée et donc la détérioration minimale sur tous les arcs du chemin p constitue la détérioration recherchée. Les détériorations par arc ε_{ij}^k permettent d'évaluer la détérioration minimale associée à un détour. L'expression suivante résume la conversion aux détériorations par chemin pour $\bar{D}_p^k$:

$$\varepsilon_p^k = \min_{(i,j) \in p} \varepsilon_{ij}^k \quad \forall k \in K, \forall p \in \Omega^k. \quad (5.35)$$

En pratique, les détériorations ε_p^k sont généralement nulles car empêcher de générer un chemin permet quand même de générer d'autres chemins produisant la même solution. Cette redondance est illustrée par les figures 5.3 et 5.4. La figure 5.3 montre pour solution une paire de chemins — les traits plein et pointillé de S à T. En interdisant l'un des deux chemins initiaux, on obtient à la figure 5.4 deux autres chemins produisant une solution identique.

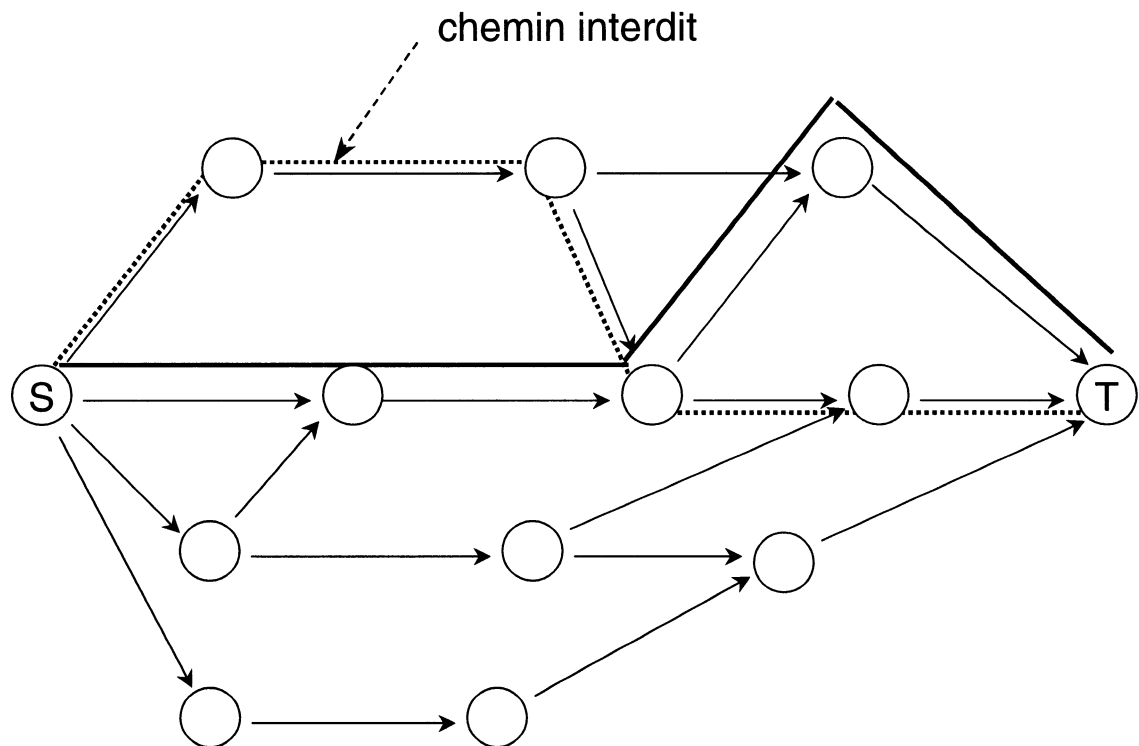


Figure 5.3: Interdiction de l'un des deux chemins d'une solution

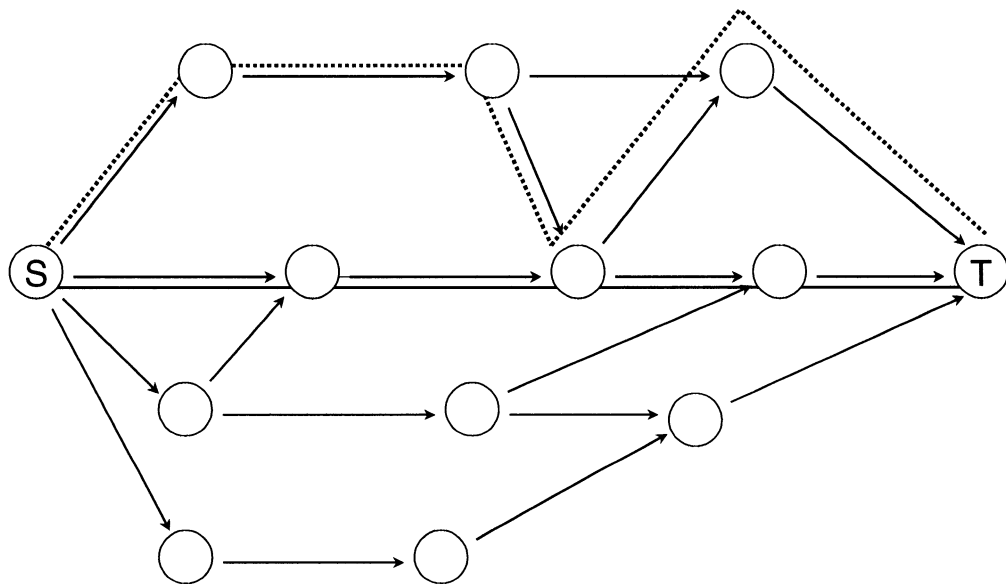


Figure 5.4: Autre paire de chemins formant la même solution

5.1.2.2) Nouveaux critères de sélection

Ces détériorations prévues pour des décisions associées aux variables de chemin permettent de définir de nouveaux critères de sélection. Ces critères permettent d'identifier les décisions fiables et la décision la plus ambivalente. Dans le but d'atteindre rapidement une solution entière sans détériorer fortement la valeur de l'objectif, les décisions fiables devraient idéalement respecter les caractéristiques suivantes:

- 1) fixer une variable de fort coût;
- 2) fixer une variable causant une faible détérioration;
- 3) fixer une variable proche de l'entier supérieur.

En effet, la première caractéristique accroît le nombre de noeuds élagués dans l'arbre de branchement en faisant progresser rapidement la part des tâches accomplies et les valeurs des bornes aux noeuds résolus en haut de l'arbre de recherche. La seconde minimise la détérioration de la valeur de l'objectif des relaxations linéaires le long de la branche explorée. Finalement, la troisième réduit la taille du problème courant en s'éloignant le moins de la solution optimale. En rassemblant toutes ces caractéristiques, nous proposons pour l'ensemble des décisions considérées le critère reposant sur le rapport suivant: le numérateur représente la détérioration associée à la décision et le dénominateur le "travail" effectué par la variable considérée. En utilisant la mesure d'intégralité justifiée par les résultats du chapitre 4, le dénominateur devient approximativement

équivalent au coût de la variable fixée. La décision D_p^k la plus fiable sera donc obtenue par:

$$\arg \min_{p,k} r_p^k, \quad (5.36)$$

$$\text{où} \quad r_p^k = \frac{\Delta_p^k}{c_p^k} \quad \forall k \in K, \forall p \in \Omega^k, \quad (5.37)$$

$$c_p^k = \sum_{(i,j) \in p} I_{ij}^k \quad \forall k \in K, \forall p \in \Omega^k, \quad (5.38)$$

$$I_{ij}^k = \begin{cases} d_{ij}^k & \text{si } a_{ij} \in F \\ c_{ij}^k & \text{si } a_{ij} \in A^k \setminus F \end{cases} \quad \forall k \in K, \forall p \in \Omega^k, \forall (i,j) \in p. \quad (5.39)$$

Une analyse identique sur les qualités d'une décision de branchement ambivalente implique une faible différence des détériorations entre les noeuds de droite et de gauche. Nous construisons donc un critère q_p^k pour identifier la meilleure décision de branchement disponible pour un chemin ambivalent:

$$q_p^k = \frac{|\varepsilon_p^k - \Delta_p^k|}{c_p^k}. \quad (5.40)$$

Ce critère permet d'identifier la variable θ_p^k choisie pour la décision ambivalente et de définir une limite $r_{\lim}$ garantissant des décisions D_p^k relativement fiables:

$$\arg \min_{p,k} q_p^k, \quad (5.41)$$

$$r_{\lim} = \min_{p,k} q_p^k. \quad (5.42)$$

Le premier noeud-frère exploré devrait être celui qui fixe le flot le long du chemin ambivalent à sa valeur supérieure. Ainsi, les décisions qui couvrent d'importantes tâches se retrouvent au début de l'arbre, et donc la profondeur de l'arbre devrait être minimale. Cette différence des détériorations relatives prévues entre les meilleures solutions entières des sous-arbres de gauche et de droite est liée au niveau de confiance de notre décision ambivalente.

Il est toujours possible de gérer l'accélération par des décisions élémentaires fiables selon la qualité de la décision de branchement ambivalente comme nous l'avons fait à la sous-section 3.3.3, avec ou sans facteur d'accélération A . Il suffit de comparer les prévisions de détériorations relatives de la solution entière (5.37) au niveau de confiance pour sélectionner les décisions complémentaires à la décision ambivalente principale. Nous obtenons donc l'ensemble des décisions complémentaires $\hat{M}_1^v$ en comparant la limite inférieure $r_{\lim}$ aux critères r_p^k donnés par l'expression (5.37) :

$$\hat{M}_1^v = \{D_p^k | (k, p) \in K \times \Omega^k \text{ et } r_p^k \leq A r_{\lim}\}. \quad (5.43)$$

Ces critères de sélection des décisions de branchement constituent une généralisation cohérente des relations (3.22) et (3.27) : ils permettent de retrouver les critères basés sur les valeurs des flots fractionnaires lorsqu'on applique l'approximation de proportionnalité entre les coûts et les détériorations unitaires qui a permis d'améliorer la précision moyenne des prédictions à la section 4.2.

En effet, on peut se permettre de conserver le coût sur un chemin comme mesure du travail fourni par la locomotive correspondante. Une mesure plus précise de la contribution à la couverture des contraintes fournie par une décision serait :

$$C_p^k = \sum_{v \in V} \left(\sum_{(i,j) \in p} a_{ij,v}^k / b_v \right) \quad \forall k \in K, \forall p \in \Omega^k. \quad (5.44)$$

En sélectionnant des flots θ_p^k contribuant à de nombreuses contraintes et dont les contributions sont importantes, on encourage l'obtention de sous-arbres de solutions très différents. En effet, ces sous-arbres partageraient plus rarement des décisions lorsque de tels chemins sont traités en haut de l'arbre. Toutefois, des travaux précédents n'ayant pas produit d'amélioration appréciable nous n'utilisons que le coefficient c_p^k dans cette étude. D'autres types de problèmes pourrait nécessiter plus de précision.

5.2) VARIABLES FIXÉES À DES VALEURS ENTIÈRES NON-UNITAIRES

Cette section ne convient qu'aux problèmes de type recouvrement généralisé. La figure 5.2 illustre comment plusieurs solutions entières optimales peuvent se situer hors de l'hypercube unitaire contenant la solution relaxée optimale. En effet, pour les problèmes de type recouvrement généralisé, les membres de droite sont souvent supérieurs à l'unité. Ne fixer qu'une unité de flot à la fois n'empêche pas l'obtention d'un flot de 2 locomotives ou plus de même type le long d'un chemin lorsqu'on prend une décision par branche puisque le même chemin peut encore être suggéré par le sous-problème au

noeud suivant. Toutefois, à l'intérieur d'une tactique de branchement avec décisions multiples, une autre décision moins efficace peut venir s'intercaler avant la répétition de cette décision idéale. Pour cette catégorie de problème, on peut fixer une variable de flot à une valeur supérieure à l'unité pour garder une prise de décision multiple efficace. En outre, fixer une variable directement à 2 ou plus permet d'éviter des calculs supplémentaires.

Pour permettre de fixer des flots à des valeurs entières non-unitaires tout en respectant le principe de séparation sans répétition, nous employons les décisions de branchement D_p^k et $\bar{D}_p^k$ décrites par les décisions (5.2). Dans un cas, on borne inférieurement la variable, dans l'autre, on la borne supérieurement et on interdit sa génération dans le sous-problème. La décision de type D_p^k s'effectue facilement, en attribuant un flot $\lceil \bar{\theta}_p^k \rceil$ le long du chemin p pour la commodité k , en corrigeant les membres de droite des contraintes de façon appropriée et sans autre modification. L'autre décision $\bar{D}_p^k$ peut être implantée de deux façons. La première nécessite plusieurs noeuds. En effet, pour imposer une limite supérieure à un flot le long d'un chemin, on peut interdire ce dernier après avoir fixé le flot à sa valeur recherchée. Il faut alors considérer autant de noeuds qu'il y a de valeurs de flot entières strictement inférieures à $\bar{\theta}_p^k$, soit $\lceil \bar{\theta}_p^k \rceil$ noeuds. En outre, chacun de ces noeuds interdit le chemin p à la commodité k . La figure 5.5 montre le branchement multiple résultant.

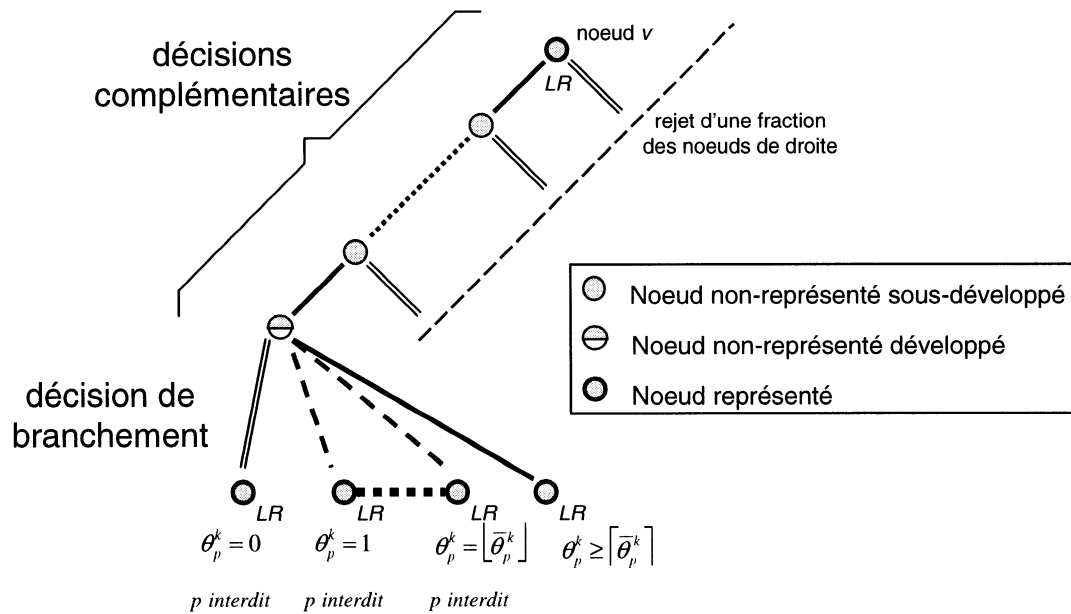


Figure 5.5: Branchement multiple

L'autre possibilité consiste à imposer une limite supérieure à ce flot en générant une colonne associée à ce chemin et en bornant cette variable dans le problème maître. On interdit aussi de générer ce chemin de nouveau dans le sous-problème correspondant. La figure 5.6 montre le branchement binaire résultant. Cette approche représente d'une plus large manière la restriction du flot par une borne supérieure.

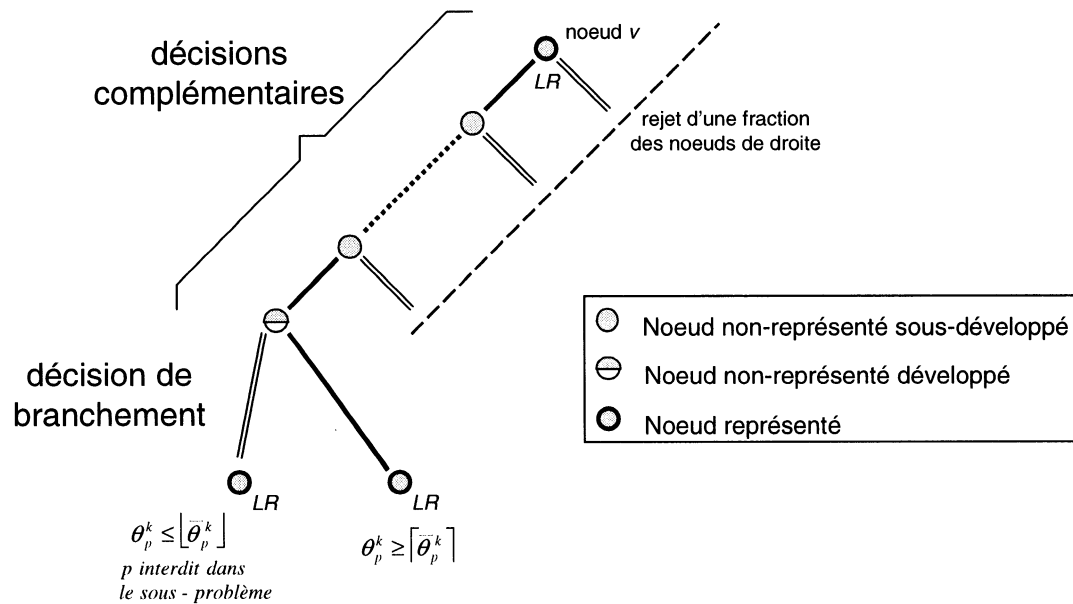


Figure 5.6: Branchement binaire

Le branchement multiple utilise un peu plus de mémoire que le branchement binaire. Toutefois, le branchement binaire implique d'ajouter des contraintes supplémentaires au problème maître pour représenter le noeud de gauche et donc de vérifier si le chemin a déjà été interdit. En outre, il faudrait stocker l'ancienne valeur du membre de droite de la contrainte pour pouvoir la restaurer correctement lors d'un retour-arrière sur plusieurs noeuds de gauche successifs. Le branchement multiple n'a qu'à permettre de nouveau le chemin puisqu'il ne peut pas être interdit par d'autres noeuds-aïeuls. Par soucis de simplicité d'implantation, nous avons retenu la version de branchement multiple.

5.3) RÉSULTATS NUMÉRIQUES

Pour cette nouvelle tactique de branchement, nous répétons l'analyse des trois scénarios décrits au chapitre 3. Le réseau et la base de données sont donc toujours les mêmes pour toutes ces résolutions. La stratégie d'exploration est celle qui nous a donné les meilleurs résultats au chapitre 4, soit la recherche par meilleures attentes en utilisant une mesure d'intégralité pondérée par les coûts. Pour fin d'analyse, nous utilisons les meilleures solutions entières après six heures de temps de calcul (CPU) car elles représentent bien la qualité relative des diverses recherches.

Le tableau 5.1 présente les valeurs de la fonction objectif pour les meilleures solutions entières obtenues. Le temps de calcul pour les résultats obtenus avec la méthode initiale de Ziarati et al. [64] était de l'ordre de quinze minutes. Pour les autres tactiques de branchement présentées, nous présentons les résultats après six heures de temps de calcul (CPU). Le paramètre d'accélération A est fixée à sa valeur optimale pour le branchement dynamique sans détérioration, soit $A=20$. Pour la nouvelle tactique de branchement avec détériorations, le paramètre d'accélération A a été testé successivement aux valeurs 20, 10, 2, 1 pour pouvoir juger de la progression de la meilleure solution entière en fonction du nombre de décisions entre les noeuds de l'arbre de recherche.

Tableau 5.1: Valeur de la fonction objectif pour la meilleure solution entière après six heures de temps de calcul (CPU)

	Scénario 1	Scénario 2	Scénario 3
Relaxation linéaire	14 710 664,2	141 122 318,0	136 374 840,5
Méthode initiale de Ziarati et al. [64]	14 742 646,6	147 336 400,6	149 133 490,2
Branchement dynamique sans détérioration et recherche par meilleures attentes avec $A = 20$ (6 h. de CPU)	14 719 470,1	145 791 083,2	145 741 119,6
Détériorations avec $A = 20$	14 723 999,1	149 584 113,1	152 549 487,9
Détériorations avec $A = 10$	14 722 809,4	148 333 680,0	151 164 906,5
Détériorations avec $A = 2$	14 718 806,6	147 209 866,6	147 882 128,9
Détériorations avec $A = 1$	14 715 599,9	147 143 618,2	148 612 548,4

Contrairement aux résultats obtenus au chapitre 4, la qualité des meilleures solutions entières a tendance à s'améliorer lorsque l'on diminue le nombre de décisions fixées entre deux noeuds de l'arbre de branchement, au lieu de plafonner vers $A=20$. Toutefois, on constate que même avec $A=1$, les résultats de la tactique de branchement ambivalente avec détériorations sont meilleurs que ceux sans détérioration uniquement pour le problème plus simple du scénario 1. Pour les scénarios 2 et 3, les solutions entières obtenues avec $A=1$ ou $A=2$ parviennent tout juste au niveau de la solution obtenue par la méthode de Ziarati et al. [64], mais n'atteignent pas la qualité de la meilleure solution entière obtenue au chapitre 4. Les solutions

entières obtenues avec des valeurs supérieures du paramètre d'accélération sont généralement pires. Le tableau 5.2 présente la même information en terme de saut d'intégrité.

Tableau 5.2: Saut d'intégrité pour la meilleure solution entière après six heures de temps de calcul (CPU)

	Scénario 1	Scénario 2	Scénario 3
Méthode initiale de Ziarati et al. [64]	0,216 %	4,40 %	9,36 %
Branchement dynamique sans détérioration et recherche par meilleures attentes avec $A = 20$ (6 h. de CPU)	0,059 9 %	3,31 %	6,87 %
Détériorations avec $A = 20$	0,090 6 %	6,00 %	11,9 %
Détériorations avec $A = 10$	0,082 6 %	5,11 %	10,8 %
Détériorations avec $A = 2$	0,055 4 %	4,31 %	8,44 %
Détériorations avec $A = 1$	0,033 6 %	4,27 %	8,97 %

Bien sûr, les effets "croisés" des décisions multiples amplifient les erreurs sur les prévisions de détérioration. Nous avons donc validé le calcul des détériorations lorsqu'elles sont prises une par une, avec $A=1$. Le tableau 5.3 résume la distribution des détériorations.

Tableau 5.3: Qualité des prédictions de détérioration pour la première solution entière

	Scénario 1	Scénario 2	Scénario 3	Moyenne
Nombre de décisions	582	688	776	100 %
Nombre de détériorations plus faibles que prévues	2	4	4	0,5 %
Nombre de détériorations prévues exactement	440	484	554	72,2%
Nombre de détériorations plus fortes que prévues	140	200	218	27,3%

Les rares détériorations plus faibles que prévues proviennent des approximations du coût de détour calculé à partir des coûts réduits pour acheminer une unité de flot supplémentaire au noeud j avec ou sans détour. Or, l'une de ces valeurs est utilisée pour évaluer la variation de coût lorsque le flot diminue d'une unité. Nous avons considéré que les détériorations sont exactes lorsque précises à l'unité près, soit à plus de trois chiffres significatifs en moyenne. Finalement, on constate que la proportion de détériorations plus fortes que prévues reflète bien la qualité relative de la meilleure solution entière obtenue avec $A=1$.

Le tableau 5.4 résume la même analyse pour les décisions portant sur des variables dont le flot est fixé à plus qu'une unité. La

proportion de détériorations plus fortes que prévues diminue, ce qui justifie leur emploi.

Tableau 5.4: Qualité des prédictions de détérioration des décisions à flot non-unitaire pour la première solution entière

	Scénario 1	Scénario 2	Scénario 3	Moyenne
Nombre de décisions	14	26	22	100 %
Nombre de détériorations prévues exactement	13	22	21	90,3%
Nombre de détériorations plus fortes que prévues	1	4	1	9,68%

Le tableau 5.3 montre que les prévisions sont légèrement plus fiables pour le premier scénario. Examinons comment sont distribuées les détériorations constatées. Le tableau 5.5 résume la nature des coûts supplémentaires imprévus en comparaison des détériorations prévues. Ceux-ci sont séparés en cinq catégories: les coûts d'exploitation de locomotives critiques comprenant principalement des pénalités de retard pour l'entretien, les coûts d'opérations des autres locomotives associés respectivement aux arcs d'entrée, de voyage à vide et de segment de train, et finalement les coûts de sous-recouvrement. En outre, le tableau contient aussi le total des détériorations imprévues.

Tableau 5.5: Différence entre détériorations prévues et constatées pour les trois scénarios

	Scénario 1	Scénario 2	Scénario 3
Détérioration prévue	591,0	550,5	459,0
Coûts supplémentaires de couverture des trains	1 237,2	835,7	676,2
Coûts supplémentaires sur les voyages à vide	27,7	11,3	33,9
Coûts fixes supplémentaires (entrée)	5 351,9	3 921,6	3 385,8
Coûts de retard des locomotives critiques	627,4	44 324,0	0
Coûts de sous-recouvrement	0	9 743 517,6	12 224 902,2
Détérioration totale imprévue constatée	7 244,2	9 792 607,3	12 228 998,1

Il semble que l'essentiel des détériorations imprévues proviennent de coûts de sous-recouvrement que les modèles simplifiés sont incapables de capturer, donc sur d'autres trains que celui sur lequel se fait l'analyse. Pour les scénarios 2 et 3, l'approximation des détériorations est insuffisante pour obtenir de meilleures décisions qu'avec les tactiques précédentes. L'information permettant d'évaluer correctement les détériorations dues aux coûts supplémentaires de sous-recouvrement des autres trains que celui étudié est disponible mais correspond à la résolution du noeud-fils. Le chapitre 6 présente un schéma de recherche prédicteur-correcteur de l'arbre d'énumération qui utilise cette méthode.

Tous les résultats précédents ont été obtenus avec la pondération des détériorations par le coût du chemin (5.36)-(5.43). D'autres comparaisons nous ont permis de valider ces critères pondérés en fonction du coût. Sans cette pondération, toutes les décisions de détérioration faible portant sur des voyages à vide sont prises en premier, laissant les dernières décisions portant sur les locomotives actives qui détériorent généralement le plus l'objectif sans grande liberté de choix. Même avec un raffinement maximal ($A=1$), les résultats sont moins bons qu'initialement, et pour des valeurs du paramètre d'accélération de 10 ou 20, les résultats sont nettement moins bons. Les erreurs semblent provenir de l'estimation des détériorations dues aux coûts de sous-recouvrement. Elles sont peut-être aggravées par l'ampleur des détériorations causées par les contraintes dynamiques dans ces deux scénarios — respectivement de 4 301 060,6 et 8 260 180,8 pour les scénarios 2 et 3 comparativement à seulement 692,9 pour le scénario 1. Peut-être que pour des problèmes ne permettant pas de sous-recouvrement, nous pourrions obtenir de meilleurs résultats. Il faudrait toutefois effectuer beaucoup plus de simulations pour valider cette hypothèse.

Nous ne recommandons pas l'approche du chapitre 5 pour obtenir de meilleurs résultats. En effet, elle ne fonctionne bien qu'avec un raffinement important, donc est relativement lente, et ce uniquement pour des problèmes "simples". Toutefois, même une petite amélioration ΔZ_c^* de la valeur de l'objectif de la meilleure solution entière peut permettre d'élaguer plusieurs noeuds.

CHAPITRE 6

UN BRANCHEMENT PRÉDICTEUR-CORRECTEUR

Le chapitre précédent nous a permis d'établir une première approximation de la détérioration associée à chaque décision élémentaire lors du passage du noeud courant v de l'arbre de recherche à son noeud-fils. Pour des problèmes simples — de taille relativement restreinte, cette approximation permet d'obtenir de meilleurs résultats. Toutefois, pour des problèmes d'une taille plus importante, la détérioration est mal estimée entre les relaxations au noeud courant et celles des noeuds-fils. Cette fois-ci, nous proposons de résoudre la relaxation linéaire au premier noeud-fils que nous appellerons noeud-prédicteur. En effet, toutes les décisions élémentaires prises au noeud courant sont prises sur des variables θ_p^k de base, donc des variables de coût réduit nul. Par contre, ces variables peuvent posséder de nouveaux coûts réduits au noeud-prédicteur. Ces nouveaux coûts réduits ainsi que la détérioration exacte associée à la décision multiple M_1^v devraient permettre de mieux estimer la détérioration associée à chaque décision élémentaire entre le noeud courant et le noeud-prédicteur.

En utilisant cette nouvelle information, nous analysons les décisions élémentaires pour les réordonner afin d'isoler celles qui semblent de piètre qualité. Le réordonnancement des décisions élémentaires menant au noeud-prédicteur n'affecte en rien le noeud lui-même. Par contre, après cette révision, il est plus facile de séparer

les décisions utiles des décisions nuisibles ou de celles qui sont ambivalentes. Dans le contexte de l'exploration heuristique d'un arbre de recherche par une méthode de séparation et évaluation progressive par génération de colonnes, il serait toutefois dommage d'attendre un nouveau coup de sonde pour corriger les erreurs identifiées par cette analyse du noeud-prédicteur. Nous proposons donc de continuer l'exploration de l'arbre à partir d'un noeud dont les décisions élémentaires sont corrigées: le noeud-correcteur. On obtient ainsi un algorithme de recherche prédicteur-correcteur similaire à la méthode employée pour calculer des approximations numériques, à la différence qu'il produit un noeud et non une estimation. En théorie, plusieurs noeuds-correcteurs pourraient être générés et être placés en compétition avec le prédicteur pour poursuivre le coup de sonde vers une feuille de l'arbre de recherche L . En pratique toutefois, la création d'un seul noeud-correcteur garantit une variété entre les différents coups de sonde. Localement, cette technique rejoint le principe d'alternance entre profondeur-d'abord et meilleur-d'abord déjà utilisé avec succès par l'approche en coups de sonde du chapitre 4 en l'appliquant à une étape de résolution contenant plusieurs décisions. Ce chapitre comporte donc quatre sections: tout d'abord une analyse de la qualité des décisions élémentaires D_p^k — *c.f.* équation (3.1) — *a posteriori*, puis une description de leur réordonnancement, ensuite une révision du branchement pour créer le(s) noeud(s)-correcteur(s) et enfin une section présentant les résultats numériques obtenus avec ou sans retour-arrière.

6.1) ANALYSE DES DÉCISIONS ÉLÉMENTAIRES A POSTERIORI

La génération du problème au noeud-prédicteur est effectuée à partir du noeud courant avec les tactiques de branchement habituelles décrites au chapitre 3. Toutefois dans un premier temps, on ne génère que le noeud de gauche à titre de noeud-prédicteur et non le noeud de droite. Par la suite, on procède à une révision des décisions *a posteriori* en évaluant les détériorations associées à chaque décision élémentaire après coup. Il s'agit encore d'une estimation lorsque plusieurs décisions élémentaires sont prises entre deux noeuds, mais contrairement au chapitre 5, la détérioration Δ_p^k pour un chemin p de la commodité k est toujours exacte lorsque l'on prend une seule décision à la fois.

6.1.1) Révision des détériorations

Les variables fixées au noeud courant étant des variables de base, leur coût réduit est nul et la détérioration prévue est nulle aussi lorsque la solution est extrapolée linéairement en conservant la même base. Toutefois, en général plusieurs changements de base surviennent lors de la résolution de la relaxation linéaire au noeud-prédicteur et la détérioration totale Δ_{tot} entre le noeud courant v de l'arbre de recherche et le noeud-prédicteur est non-nulle. Comment redistribuer l'erreur entre les décisions élémentaires lorsque la détérioration totale Δ_{tot} est non-nulle? Nous proposons de le faire en comparant les coûts réduits $\bar{c}_p^k$ après décisions. Tout d'abord, lorsque

le coût réduit reste nul, les décisions élémentaires correspondantes n'ont aucune responsabilité dans la détérioration imprévue. Lorsque le coût réduit associé à une décision élémentaire devient non-nul après la prise de décision, c'est de là que provient la détérioration imprévue Δ_{imp} . La figure 6.1 illustre les différentes pentes permettant d'évaluer la détérioration Δ_p^k . L'axe des abscisses représente la distance fractionnaire associée à la décision D_p^k : $\theta_p^k \geq 1$. Les détériorations Δ_p^k maximale et Δ_p^k réelle sur l'axe des ordonnées représentent respectivement la détérioration maximale théoriquement possible avec le coût réduit $\bar{c}_p^k$ au noeud-prédicteur et la détérioration réelle après la décision D_p^k .

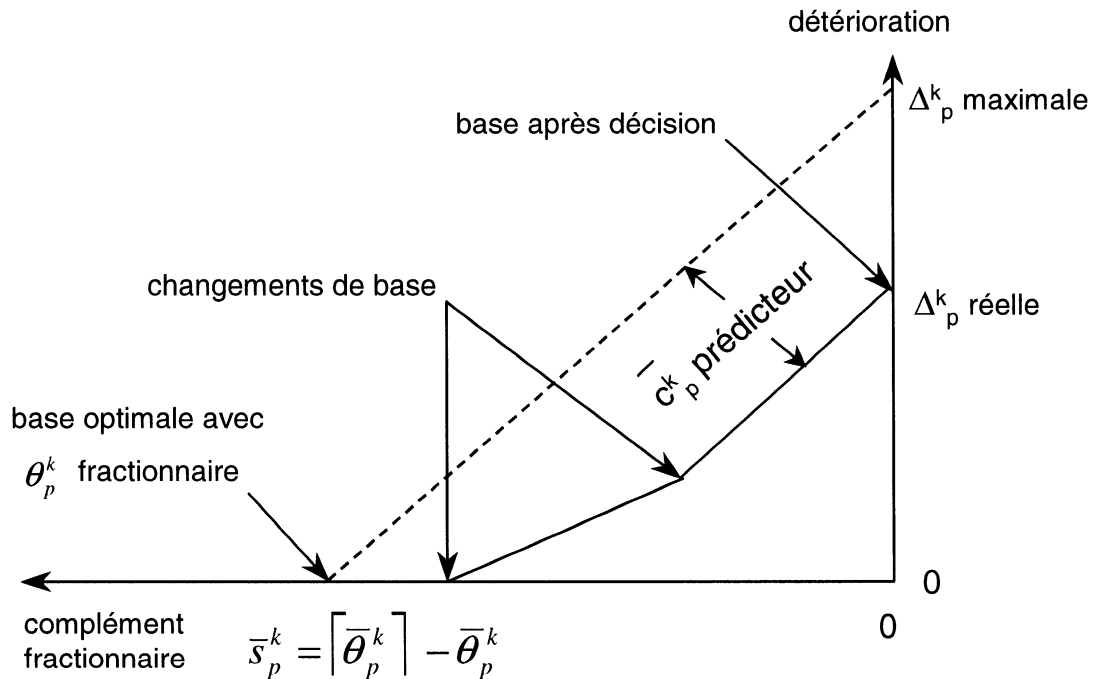


Figure 6.1: Détériorations maximale et réelle associées à une décision élémentaire

On calcule donc les détériorations minimales et maximales dues à une décision élémentaire contribuant à générer le noeud-prédicteur:

$$\Delta_p^k \min = 0 \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v, \quad (6.1)$$

$$\Delta_p^k \max = \bar{c}_p^k \cdot \bar{s}_p^k \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v, \quad (6.2)$$

où $\bar{c}_p^k$ et $\bar{s}_p^k$ représentent respectivement le coût réduit au noeud-prédicteur et le complément fractionnaire associés à une décision élémentaire D_p^k .

Mais en prenant plusieurs décisions à la fois entre deux noeuds de calcul des relaxations linéaires, on n'obtient que la détérioration réelle totale Δ_{tot} :

$$\sum_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} \Delta_p^k = Z_{prédicteur} - Z_v = \Delta_{tot} \quad (6.3)$$

où Z_v et $Z_{prédicteur}$ représentent les valeurs des solutions respectives des relaxations linéaires au noeud courant et au noeud-prédicteur. Comment redistribuer cette détérioration totale Δ_{tot} entre les décisions élémentaires? Tout d'abord, il est parfois possible de préciser les détériorations minimales. En effet, la détérioration totale Δ_{tot} est composée d'une partie connue et d'une partie imprévue Δ_{imp} :

$$\Delta_{tot} = \sum_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} \Delta_p^k \min + \Delta_{imp}, \quad (6.4)$$

où $\sum_{(k,p)} \Delta_p^k \min$ est nulle initialement. Toutefois, la détérioration minimale d'une décision élémentaire doit absorber tout résidu laissé par les autres décisions élémentaires en supposant qu'elles provoquent

des détériorations maximales. L'équation suivante résume ces précisions:

$$\Delta_p^k \min = \max \left(0, \Delta_{tot} - \sum_{(k',p') \neq (k,p)} \Delta_{p'}^{k'} \max \right) \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v. \quad (6.5)$$

Ces modifications ne sont pas récursives, une détérioration minimale ne pouvant être modifiée qu'une seule fois. Toutefois, il faut redéfinir la valeur résiduelle de la détérioration imprévue en fonction des nouvelles détériorations minimales qui en ont absorbées une partie:

$$\Delta_{imp} = \Delta_{tot} - \sum_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} \Delta_p^k \min. \quad (6.6)$$

Une fois les détériorations minimales précisées, la détérioration imprévue est attribuée en proportion équivalente à toutes les décisions élémentaires. Cette approximation suppose une distribution uniforme des probabilités d'occurrence du dernier changement de base en fonction de la distance fractionnaire, ce qui semble être le meilleur modèle disponible. Le poids de chaque décision élémentaire est donc:

$$\frac{\Delta_p^k \max - \Delta_p^k \min}{\sum_{(k',p') \in (K, \Omega^{k'}) \mid D_{p'}^{k'} \in M_1^v} \Delta_{p'}^{k'} \max - \Delta_{p'}^{k'} \min} \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v, \quad (6.7)$$

$$\text{et } \Delta_p^k \approx \Delta_p^k \min + \frac{\Delta_p^k \max - \Delta_p^k \min}{\sum_{(k',p') \in (K, \Omega^{k'}) \mid D_{p'}^{k'} \in M_1^v} \Delta_{p'}^{k'} \max - \Delta_{p'}^{k'} \min} \Delta_{imp} \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v. \quad (6.8)$$

Bien sûr, on retrouve la détérioration exacte lorsqu'une seule décision élémentaire est considérée.

6.1.2) Critères de réordonnancement

A partir des estimations de détériorations, il est facile de classer les décisions en commençant par les meilleures et en finissant par les pires. Toutefois, il faut encore tenir compte du travail relatif accompli par ces décisions élémentaires en plus de leur détérioration respective. Comme dans le chapitre 5, nous proposons d'employer le coût c_p^k comme mesure du travail effectué par une décision élémentaire D_p^k , ce qui nous ramène à des critères identiques à ceux des équations (5.36) - (5.39) pour identifier la décision la plus fiable:

$$\arg \min_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} r_p^k, \quad (6.9)$$

$$\text{où} \quad r_p^k = \frac{\Delta_p^k}{c_p^k} \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v, \quad (6.10)$$

$$c_p^k = \sum_{(i,j) \in p} I_{ij}^k \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v, \quad (6.11)$$

$$I_{ij}^k = \begin{cases} d_{ij}^k & \text{si } a_{ij} \in F \\ c_{ij}^k & \text{si } a_{ij} \in A^k \setminus F \end{cases} \quad \forall k \in K, \forall p \in \Omega^k, \forall (i,j) \in p. \quad (6.12)$$

Après le calcul des détériorations relatives r_p^k pour chaque décision élémentaire menant au noeud-prédicteur, les décisions élémentaires correspondant à chaque paire commodité-chemin sont donc réordonnées en ordre croissant des critères r_p^k , avant d'être conservées ou remises en question au noeud-correcteur. Tel que mentionné plus tôt, ce réordonnancement n'affecte en rien le noeud-prédicteur lui-même.

Enfin, nous définissons des détériorations relatives minimales et maximales correspondant aux détériorations minimales et maximales (6.1) et (6.2) :

$$r_p^k \min = \frac{\Delta_p^k \min}{c_p^k} \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v \quad (6.13)$$

et

$$r_p^k \max = \frac{\Delta_p^k \max}{c_p^k} \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v . \quad (6.14)$$

6.2) REGROUPEMENT DES DÉCISIONS ÉLÉMENTAIRES

Une fois que les détériorations relatives r_p^k ont permis d'ordonner les décisions élémentaires menant au noeud-prédicteur, il est possible de regrouper ces décisions pour les classer dans l'une des trois catégories suivantes: 1) "Maintenue", 2) "Branchée", 3) "Rejetée". Cette classification permet de développer un autre noeud-fils à partir du noeud courant v . On peut ainsi réouvrir un branchement multiple pour en séparer les décisions nuisibles et faire apparaître le noeud ou les noeuds-correcteurs qui nous intéressent. Plusieurs auteurs dont Tseng et al. [59] ont montré que les décisions multiples sont sous-optimales en ce qui concerne la qualité de la solution. Même en permettant un retour-arrière, on ne peut pas atteindre une meilleure solution si elle n'est pas représentée. Cette technique permet de corriger la représentation. La première étape consiste à déterminer dans quelle catégorie placer chaque décision élémentaire.

Plusieurs techniques permettent de définir les frontières délimitant les catégories recherchées: certaines fixent la cardinalité des ensembles, d'autres leurs limites. Dans le but de regrouper des décisions élémentaires dont les détériorations sont de même ordre de grandeur, nous utilisons des valeurs limites pour les critères r_p^k . Parmi les diverses techniques pour déterminer les limites entre les catégories de décisions, nous en retenons trois: la répartition proportionnelle, la séparation par référence et les plus grands écarts.

La première attribue une fraction du domaine total à chaque groupe. Par exemple, un professeur essayant de regrouper ces élèves à partir de leur note pour leur attribuer des cotes (A,B,C,D,F) pourrait alors fixer arbitrairement les limites des groupes: F [0%,35%[, D [35%,55%[, C [55%,75%[, B [75%,90%[et A [90%,100%] par exemple. Cela correspondrait à attribuer les fractions suivantes du domaine à chaque note: F (7/20), D (1/5), C (1/5), B (3/20) et A (1/10), le tout couvrant l'ensemble du domaine des notes possibles. De la même façon, nous pourrions considérer que la meilleure moitié des décisions doivent être maintenues, le quart suivant branchées, et le dernier quart rejetées.

Ces fractions qui sont déterminées de manière arbitraire peuvent l'être plus naturellement à partir de la répartition des détériorations associées aux décisions. La seconde technique suppose une distribution prévue, ou du moins certaines caractéristiques de l'échantillon de détériorations pour servir de références. On débute en déterminant les

limites minimales et maximales des critères r_p^k ainsi qu'un certain degré de confiance r_{conf} . Puis, à partir du degré de confiance, on fixe les limites des décisions élémentaires entre chaque groupe. Dans l'exemple précédent, ce serait le cas du professeur qui classe ses élèves en supposant une distribution gaussienne des notes. Il détermine la moyenne μ et l'écart-type σ de la classe et attribue les cotes en conséquence: F $[0\%, \mu - 3\sigma[$, D $[\mu - 3\sigma, \mu - \sigma[$, C $[\mu - \sigma, \mu + \sigma[$, B $[\mu + \sigma, \mu + 3\sigma[$ et A $[\mu + 3\sigma, 100\%]$. Par contre, il est plus difficile de prévoir la forme de la distribution des critères r_p^k . Par inspection des détériorations estimées au chapitre 5, une distribution exponentielle — de Poisson — ou une distribution de Fisher semblent les candidates les mieux adaptées, bien que souvent la distribution n'ait pas de forme particulière. Le meilleur critère permettant de décrire la qualité globale des décisions de façon relativement fiable est donc le degré de confiance r_{conf} . Il peut être estimé par le rapport entre la détérioration imprévue résiduelle et la détérioration imprévue résiduelle maximale:

$$r_{conf} = \frac{\Delta_{imp}}{\sum_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} \Delta_p^k \max - \Delta_p^k \min} . \quad (6.15)$$

Toutefois, r_{conf} ne permet pas d'établir une valeur de référence pour définir les limites de séparation des trois groupes de décisions, car sans distribution particulière, c'est un indicateur assez limité quant à la qualité d'une décision élémentaire. En outre, il contribue déjà par définition aux valeurs estimées des détériorations de chaque décision élémentaire (6.8):

$$\Delta_p^k \approx \Delta_p^k \min + r_{conf} (\Delta_p^k \max - \Delta_p^k \min) \quad \forall k \in K, \forall p \in \Omega^k \mid D_p^k \in M_1^v. \quad (6.16)$$

Il est donc nécessaire de déterminer une valeur référence sans tenir compte d'une forme de distribution quelconque. Puisque les temps de calcul et l'efficacité des chemins interdits sont respectivement plus grands et plus faible en comparaison de la prise de décision, nous désirons prendre toujours au moins une décision élémentaire. Lorsque l'on désire ne produire qu'un seul noeud-correcteur, cette décision élémentaire de référence est suffisante pour former les trois catégories recherchées. Par exemple avec la tactique de branchement dynamique, la détérioration relative attribuée à la décision associée à la variable de flot dont la partie fractionnaire est la plus proche de 0,5 constitue notre valeur de référence. Toutes les décisions dont la détérioration relative est moindre appartiennent à la catégorie "Maintenue", toutes celles dont la détérioration relative est supérieure à celle de la décision de référence sont regroupées sous la catégorie "Rejetée". La décision ambivalente constitue à elle seule la catégorie "Branchée". Pour la tactique de branchement sur le dernier meilleur chemin, la valeur maximale de la détérioration relative de la meilleure décision élémentaire disponible semble donc une référence acceptable pour déterminer l'ordre de grandeur des décisions à prendre de façon simultanée. La séparation s'effectue donc autour de la première décision dont la détérioration relative est supérieure à la détérioration relative maximale de la meilleure décision élémentaire.

Pour produire plusieurs noeuds-correcteurs, nous définissons l'ensemble des décisions maintenues tel que:

$$r_p^k \max \leq r_{p'}^{k'} \max \mid (k', p') = \arg \min_{(k, p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} (r_p^k). \quad (6.17)$$

L'ensemble des décisions rejetées est défini en utilisant la même référence:

$$r_p^k \min \geq r_{p'}^{k'} \max \mid (k', p') = \arg \min_{(k, p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} (r_p^k). \quad (6.18)$$

Finalement, l'ensemble des décisions élémentaires pouvant admettre $r_{p'}^{k'} \max$ comme valeur de détérioration sont placées initialement dans le groupe des décisions à brancher. Par soucis de monotonie, il serait toutefois utile de placer dans les groupes extrêmes toute décision dont la détérioration relative estimée est plus faible que celle de la pire décision maintenue et toute décision dont la détérioration relative estimée est plus grande que celle de la meilleure décision rejetée. La figure 6.2 illustre le classement ainsi obtenu pour un exemple donné contenant huit décisions élémentaires. Les limites définies en (6.17) et (6.18) isolent deux décisions maintenues et deux décisions rejetées de l'ensemble initial. Les corrections font apparaître une décision supplémentaire dans chacun de ces groupes pour obtenir une répartition finale 3-2-3, monotone selon les détériorations relatives estimées en ordre croissant.

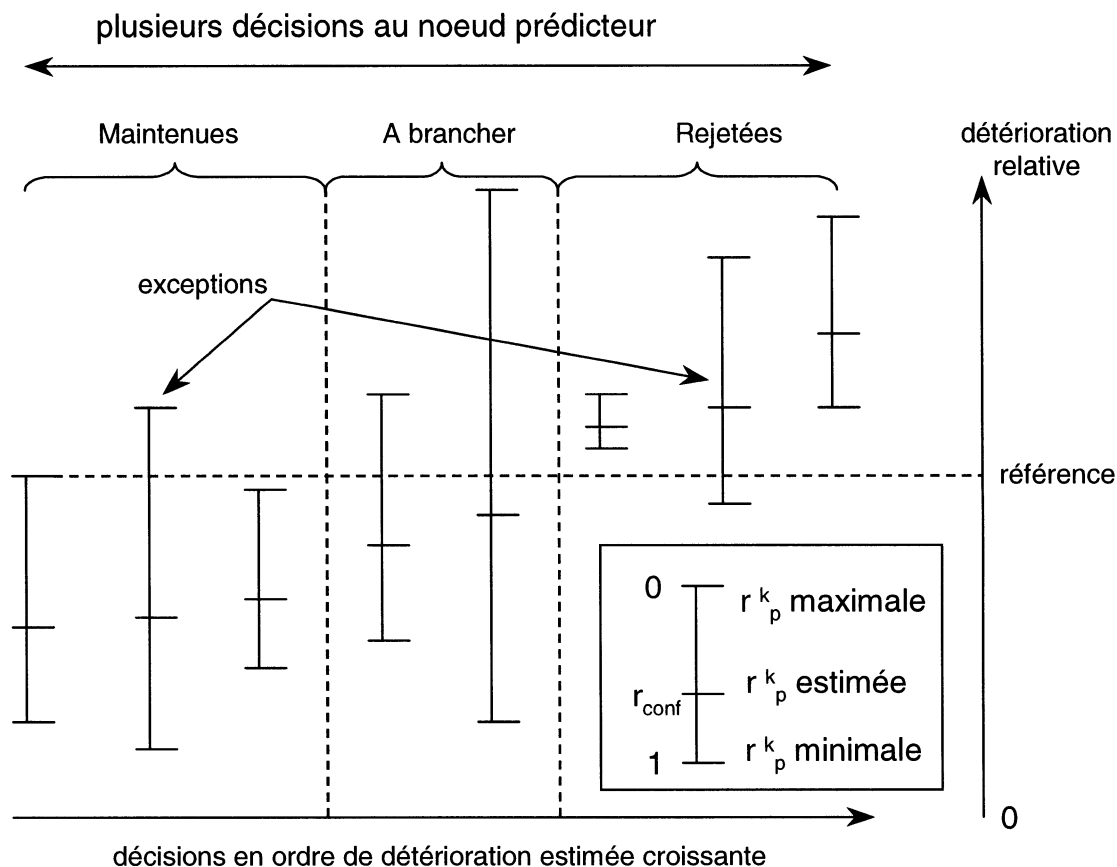


Figure 6.2: Détermination des limites de séparation par référence

La dernière technique de classement des décisions repose sur les écarts entre les détériorations relatives r_p^k . On suppose qu'il y a T critères r_p^k ou T décisions élémentaires prises entre le noeud courant et le noeud-prédicteur. Notons δ_i la différence entre les critères r_i et r_{i+1} classés en ordre croissant. Pour obtenir trois catégories, il suffit d'identifier les deux plus grandes différences. Toutefois, ce procédé implique que les trois groupes ne sont pas vides. Cette condition est peut-être souhaitable pour les décisions élémentaires que l'on désire maintenir, mais pas forcément pour les deux autres

catégories. Pour permettre des catégories vides pour les groupes extrêmes — "Maintenues" et "Rejetées", il suffit de considérer des différences virtuelles avec les limites minimales et maximales. En appliquant cette logique pour séparer les décisions élémentaires en trois groupes, il faut tout d'abord identifier les valeurs extrêmes $r_{\min}$ et $r_{\max}$ des critères de sélection r_p^k :

$$r_{\min} = 0 \quad (6.19)$$

$$r_{\max} = \frac{\max_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} \Delta_p^k}{\min_{(k,p) \in (K, \Omega^k) \mid D_p^k \in M_1^v} C_p^k} . \quad (6.20)$$

Les différences virtuelles δ_0 et δ_T sont donc définies ainsi :

$$\delta_0 = r_1 - r_{\min} = r_1 \quad (6.21)$$

$$\delta_T = r_{\max} - r_T . \quad (6.22)$$

Pour permettre une catégorie vide au milieu du classement, il suffit de considérer qu'une différence peut être elle-même divisée en deux. Ces deux moitiés peuvent alors représenter le début et la fin d'une catégorie vide. En ne considérant que la différence virtuelle supérieure δ_T et non l'autre δ_0 , nous obtenons un classement garantissant qu'au moins une décision sera prise. Les trois catégories ont ainsi les caractéristiques souhaitées. Cette méthode a l'avantage de minimiser les risques d'erreur, un peu comme si le professeur de nos exemples précédents voulait éviter toute forme de contestation. Toutefois, les groupes ainsi obtenus peuvent varier fortement en taille entre deux noeuds consécutifs et sont fortement instables lorsque l'on ajoute ou retire une décision élémentaire. La figure 6.3 montre le

classement que l'on obtiendrait avec le même exemple contenant huit décisions élémentaires.

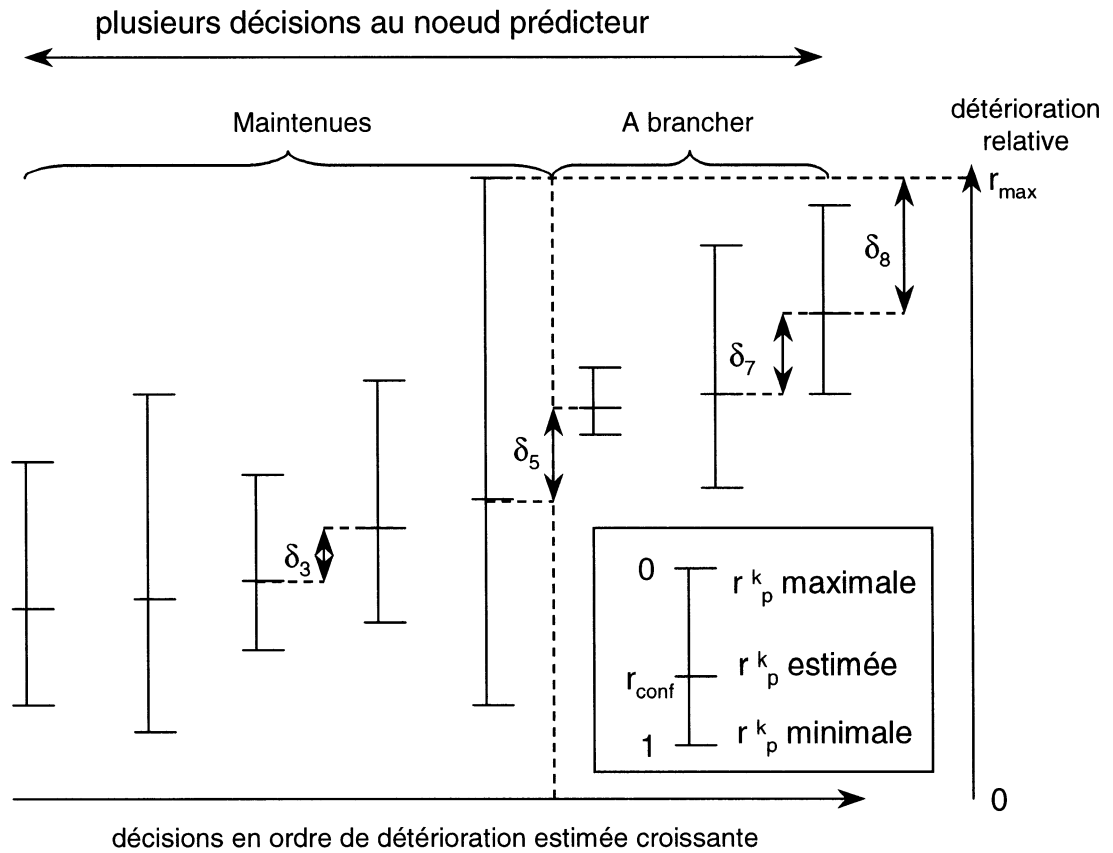


Figure 6.3: Détermination des limites de séparation par différences

Pour simplifier, on suppose que la cinquième décision élémentaire est celle de détérioration maximale la plus grande et aussi celle de coût minimal. Seules les plus grandes différences sont représentées:

$$\delta_8 > \delta_5 > \delta_7 > \frac{\delta_8}{2} > \delta_3. \quad (6.23)$$

Les deux limites entre les groupes seraient donc placées après les décisions cinq et huit, donc aucune décision ne serait rejetée et les cinq premières seraient maintenues. Cette dernière technique pourrait être employée pour réparer des regroupements qui séparent arbitrairement des décisions proches relativement aux autres en utilisant le degré de confiance r_{conf} pour en restreindre la portée ou pour augmenter un nombre trop faible de décisions maintenues. La figure 6.4 montre simplement des exemples de cas pathologiques.

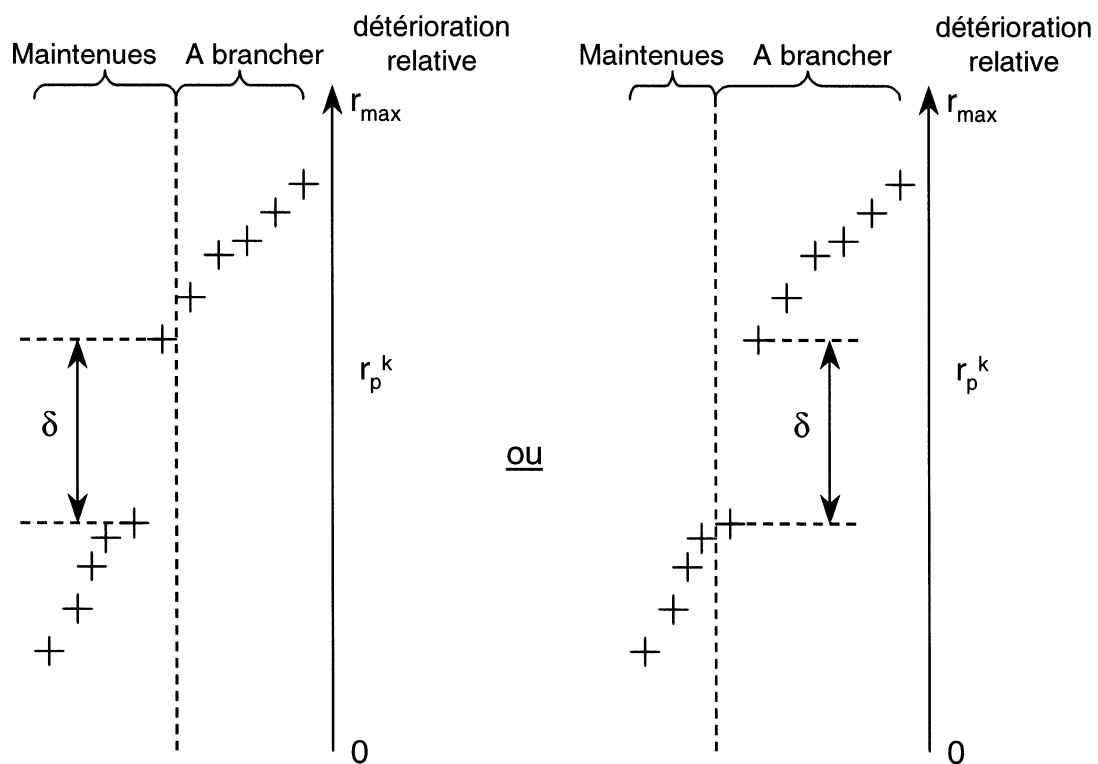


Figure 6.4: Exemples de regroupements à corriger

La technique de classement des décisions élémentaires par grands écarts peut donc être employée pour améliorer les séparations obtenues par

Tout d'abord, dans le second cas de branchement ambivalent, aucune analyse n'est effectuée sur la variable associée à la décision interdite: sans générer la colonne correspondante au chemin interdit, nous pouvons difficilement connaître le coût réduit *a posteriori*. Aucune décision concernant la variable associée au chemin interdit ne sera donc prise aux noeuds-correcteurs: la variable demeure libre.

Dans tous les cas, après classement des décisions élémentaires, il suffirait de conserver les décisions maintenues, d'interdire toutes les décisions rejetées et finalement de considérer toutes les combinaisons possibles de décisions à brancher. La figure 6.6 illustre le branchement théoriquement obtenu. Malheureusement, ce branchement est très lourd en nombre de chemins interdits, ce qui retarde grandement la résolution des noeuds suivants.

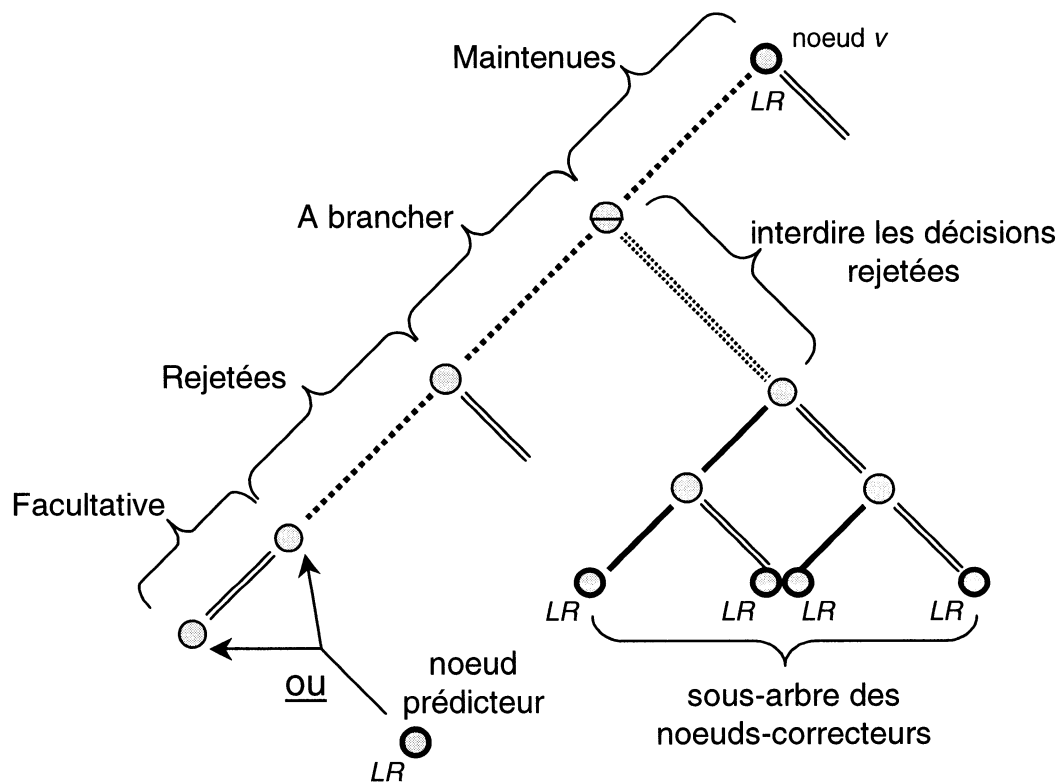


Figure 6.6: Branchement corrigé théorique

Nous utiliserons donc un branchement corrigé légèrement modifié pour bénéficier de l'approche systématique introduite par les chemins interdits sans toutefois surcharger la machine de longs calculs. En effet, il est plus rapide et peu nuisible de considérer le plus possible de variables associées à des décisions interdites comme des variables libres, en s'assurant toutefois de générer une nouvelle portion de l'arbre de recherche à chaque noeud-fils. Nous économisons ainsi beaucoup de travail de prolongation d'étiquettes interdites en ne rajoutant qu'un seul chemin interdit entre deux relaxations linéaires. La figure 6.7 montre le résultat obtenu pour générer les noeuds-correcteurs.

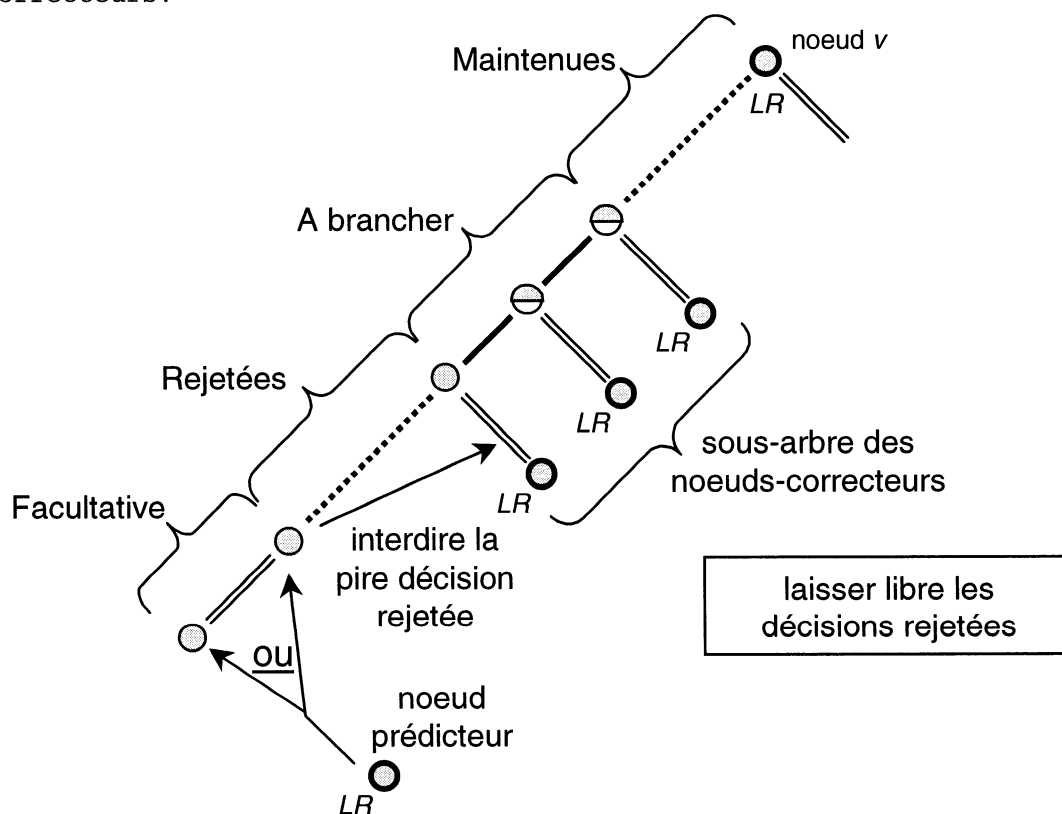


Figure 6.7: Branchement corrigé pratique

Finalement, il reste à déterminer les nouvelles valeurs espérées de la meilleure solution entière du sous-arbre de recherche à un noeud donné. Pour le prédicteur, il suffit de corriger sa valeur après la résolution de sa propre relaxation linéaire comme on le ferait pour l'attribuer aux noeuds-fils (4.4). Si le noeud-correcteur est unique, la même technique est employée et le coup de sonde se poursuit avec le plus prometteur des deux noeuds-frères. Lorsqu'il y a plusieurs noeuds-correcteurs, on ne résout la relaxation linéaire que du correcteur le plus profond. Ceux dont on ne résout pas la relaxation linéaire ne peuvent conserver la valeur extrapolée à partir du noeud-père et d'un noeud-aïeul. En effet, on doit leur attribuer des valeurs légèrement différentes pour orienter le prochain coup de sonde à partir du meilleur d'entre eux. Nous proposons donc une simple interpolation linéaire en fonction de la profondeur entre l'objectif espéré au noeud-correcteur résolu et cette valeur initiale.

6.4) RÉSULTATS NUMÉRIQUES

Pour ce nouveau type de branchement, nous analysons encore les résultats obtenus pour les trois scénarios décrits au chapitre 3. Le réseau et la base de données sont donc toujours les mêmes pour toutes ces résolutions. Pour regrouper les décisions menant au noeud-prédicteur, nous utilisons les techniques de séparation par référence et par plus grands écarts. On obtient ainsi deux suggestions par limite entre les catégories. Les limites permettant de prendre le plus

grand nombre de décisions-élémentaires au noeud-correcteur sont utilisées.

6.4.1) Résultats sans retour-arrière

Une première série de résultats, obtenue sans effectuer de retour-arrière dans l'arbre de recherche, permet de vérifier que le schéma prédicteur-correcteur produit de meilleures solutions entières dès la première descente. Le tableau 6.1 contient tout d'abord des références: la valeur de la relaxation linéaire de chaque problème initial, les résultats obtenus par la méthode initiale de Ziarati et al. [64] et les résultats obtenus après la première descente en utilisant la meilleure combinaison de tactique de branchement et de stratégie de recherche du chapitre 4, soit la tactique dynamique ambivalente avec une valeur de paramètre $A=20$. Puis le tableau présente la valeur de la première solution entière obtenue en appliquant le schéma prédicteur-correcteur pour chacun des trois scénarios décrits plus tôt. Les résultats présentés correspondent à huit combinaisons de schéma prédicteur-correcteur, de tactique de branchement et de paramètre. Les valeurs des sauts d'intégrité relatifs sont indiquées entre parenthèses — 100% pour la méthode initiale de Ziarati et al. [64]. Les solutions entières de coût inférieur à celui obtenu avec la meilleure combinaison de tactique de branchement et de stratégie d'exploration du chapitre 3 apparaissent en caractères gras.

Tableau 6.1: Valeur de la première solution entière

	Scénario 1	Scénario 2	Scénario 3
Relaxation linéaire	14 710 664,2	141 122 318,0	136 374 840,5
Méthode initiale de Ziarati et al. [64]	14 742 646,6 (100%)	147 336 400,6 (100%)	149 133 490,2 (100%)
Branchement dynamique avec A = 20	14 719 973,0 (29,1%)	146 330 398,5 (83,8%)	147 772 198,4 (89,3%)
Schéma prédicteur-correcteur générant un seul noeud-correcteur			
Branchement sur dernier meilleur chemin avec 40 décisions	14 722 510,4 (37%)	148 631 136,0 (120,8%)	149 282 777,4 (101,2%)
Branchement sur dernier meilleur chemin avec 20 décisions	14 721 881,7 (35%)	146 922 768,3 (93,3%)	147 027 634,5 (83,5%)
Branchement dynamique avec A = 40	14 719 829,5 (28,7%)	146 354 272,4 (84,2%)	145 695 579,2 (73%)
Branchement dynamique avec A = 20	14 719 214,0 (26,7%)	147 174 355,7 (97,4%)	146 138 156,6 (76,5%)
Schéma prédicteur-correcteur générant plusieurs noeuds-correcteurs			
Branchement sur dernier meilleur chemin avec 40 décisions	14 722 510,4 (37%)	146 678 438,9 (89,4%)	148 274 336,8 (93,3%)
Branchement sur dernier meilleur chemin avec 20 décisions	14 722 117,6 (35,8%)	147 032 121,7 (95,1%)	148 303 638,8 (93,5%)
Branchement dynamique avec A = 40	14 719 342,2 (27,1%)	147 083 040,0 (95,9%)	147 684 457,0 (88,6%)
Branchement dynamique avec A = 20	14 719 994,2 (29,2%)	145 875 329,7 (76,5%)	146 083 597,5 (76,1%)

Avec les schémas prédicteur-correcteur, les premières solutions entières obtenues sont nettement meilleures en comparaison avec la méthode initiale de Ziarati et al. [64], à quelques exceptions près. Il est plus difficile de voir la tendance par rapport aux meilleures solutions obtenues dès la première descente avec la meilleure combinaison obtenue au chapitre 3. En moyenne, on constate des améliorations comparables à celles obtenues par la meilleure tactique de branchement dynamique ambivalent, soit avec $A=20$. En effet, cette méthode permettait d'obtenir une réduction de 32,5% des sauts d'intégrité obtenus par la méthode initiale de Ziarati et al. [64] pour la moyenne des scénarios. En comparant les meilleures valeurs du paramètre A , le schéma prédicteur-correcteur générant plusieurs noeuds-correcteurs produit des résultats légèrement meilleurs — réduction relative de 39,5% — lors de la descente initiale que ceux générés avec un seul noeud-correcteur — réduction relative de 38%. Pour la moyenne des scénarios, cette tendance est conservée: la qualité des meilleures solutions entières a tendance à s'améliorer avec les schémas prédicteur-correcteur, quelles que soient les tactiques de branchement et valeurs de paramètre. Mais pour un seul scénario, une seule descente ne peut être garante d'un meilleur résultat, surtout lorsque l'on ne génère qu'un seul noeud-correcteur.

6.4.2) Résultats avec retour-arrière

Il est donc nécessaire d'employer plusieurs coups de sonde pour essayer d'obtenir un meilleur résultat quel que soit le scénario étudié. La stratégie d'exploration est donc celle qui nous a donné les meilleurs résultats, soit la recherche par meilleures attentes avec mesure de la distance fractionnaire pondérée par les coûts. Les résultats présentés sont ceux obtenus pour le schéma prédicteur-correcteur générant un seul noeud-correcteur. Les figures suivantes comparent l'évolution de la meilleure solution entière sur vingt-quatre heures de temps de calcul *CPU* pour les trois scénarios considérés.

La figure 6.8 illustre six courbes représentant l'évolution de la meilleure solution entière pour le scénario 1. Les six combinaisons de tactique de branchement et de schéma de recherche sont les suivantes. Tout d'abord, deux tactiques de branchement sans schéma prédicteur-correcteur servent de références : il s'agit des courbes obtenues avec les tactiques de branchement sur le dernier meilleur chemin avec 40 décisions (*Bdern_meilleurD40*) et de branchement dynamique ambivalent avec $A=20$ (*Bdyn_ambivA20*). Elles représentent les résultats extrêmes considérés jusqu'ici — *c.f.* figure 4.16. En outre, les quatre courbes représentant les combinaisons de tactiques de branchement et de valeurs de paramètres avec schéma prédicteur-correcteur (*PredBdern_meilleurD40*, *PredBdern_meilleurD20*, *PredBdyn_ambivA40*, *PredBdyn_ambivA20*) sont aussi tracées. En d'autres termes, il s'agit de comparer les résultats de

référence obtenus au chapitre 4 avec ceux obtenus par un schéma prédicteur-correcteur.

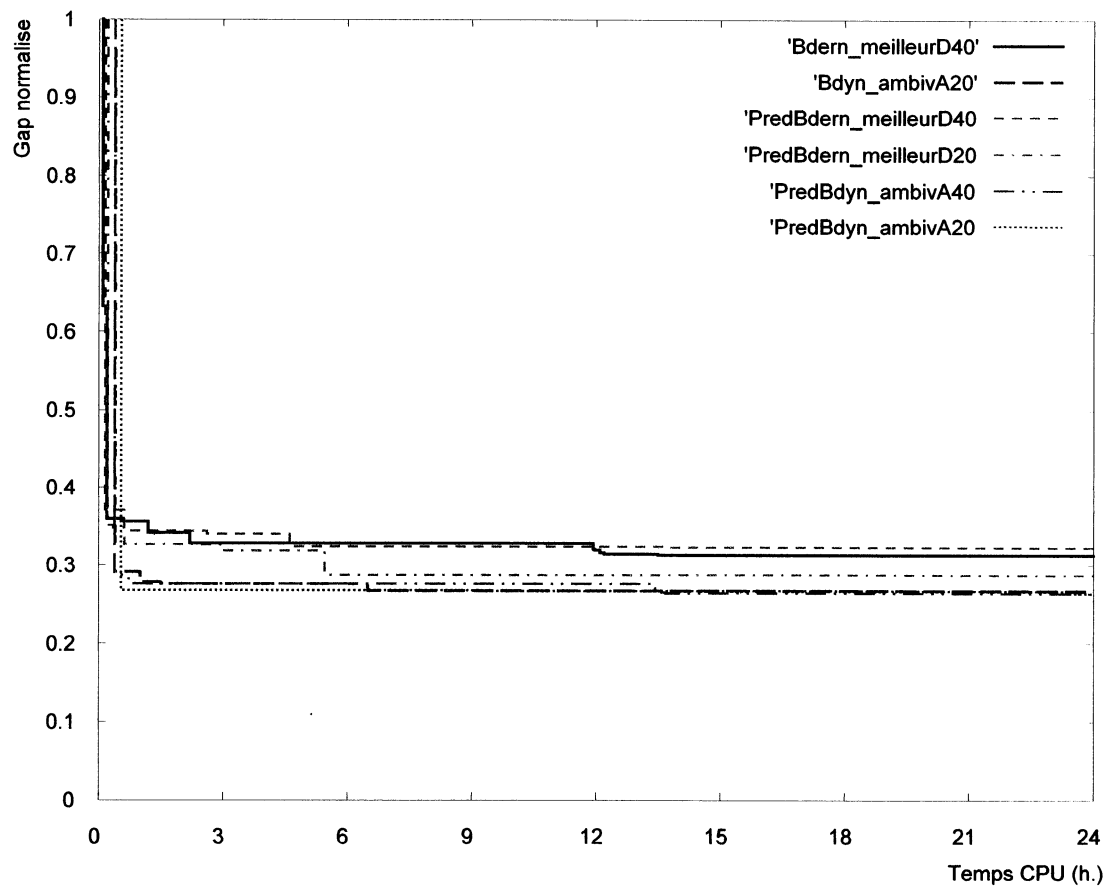


Figure 6.8: Comparaison du coût de la meilleure solution entière pour le scénario 1 avec et sans schéma prédicteur-correcteur

La figure 6.8 montre que pour le scénario le plus facile, les résultats sont comparables avec ou sans schéma prédicteur-correcteur. On remarque toutefois que la tactique de branchement dynamique ambivalente reste supérieure en qualité au branchement sur le dernier

meilleur chemin. Les figures 6.9 et 6.10 présentent les mêmes courbes pour les deux autres scénarios.

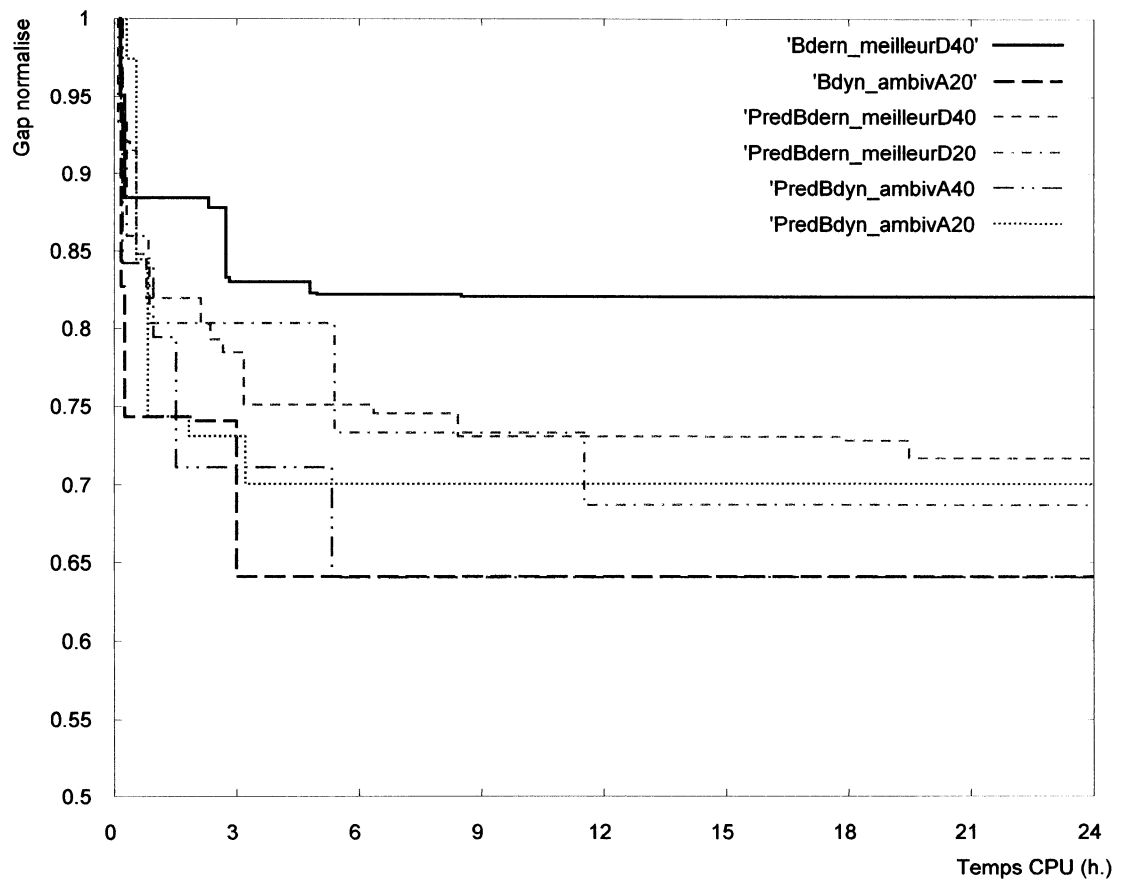


Figure 6.9: Comparaison du coût de la meilleure solution entière pour le scénario 2 avec et sans schéma prédicteur-correcteur

Pour le second scénario, il est clair que les résultats obtenus avec n'importe quel schéma prédicteur sont meilleurs que ceux obtenus par simple extension de la méthode initiale de Ziarati *et al.* [64] avec une stratégie d'exploration par meilleures attentes. Toutefois, la

tactique de branchement dynamique ambivalente semble toujours tenir tête aux versions avec schéma prédicteur-correcteur.

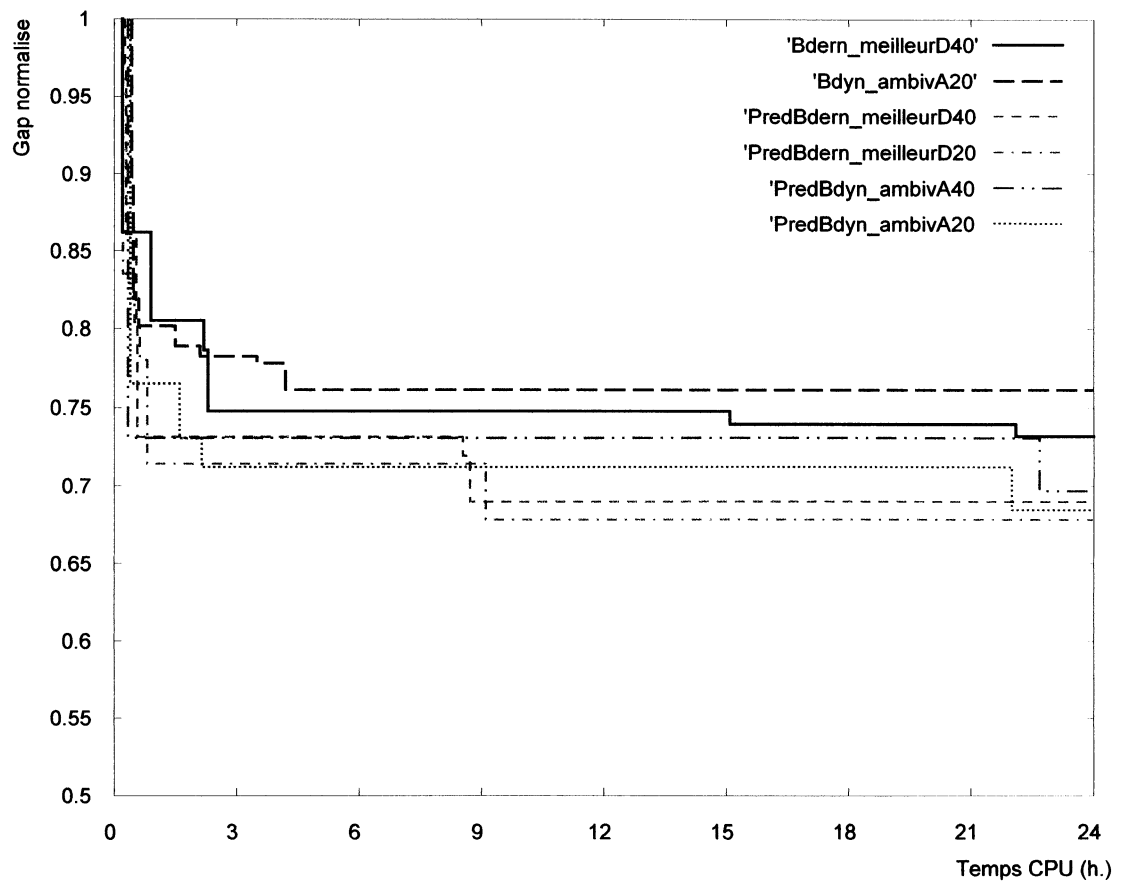


Figure 6.10: Comparaison du coût de la meilleure solution entière pour le scénario 3 avec et sans schéma prédicteur-correcteur

Finalement, pour le scénario le plus difficile, la supériorité des schémas prédicteur-correcteur apparaît plus clairement, alors qu'aucune nouvelle solution entière n'est trouvée par la tactique ambivalente sans schéma prédicteur-correcteur pendant ses dernières vingt heures de calcul en temps *CPU*.

Pour résumer, la figure 6.11 présente les mêmes courbes pour la moyenne des trois scénarios. On constate que le schéma prédicteur-correcteur générant un seul noeud-correcteur produit très rapidement de

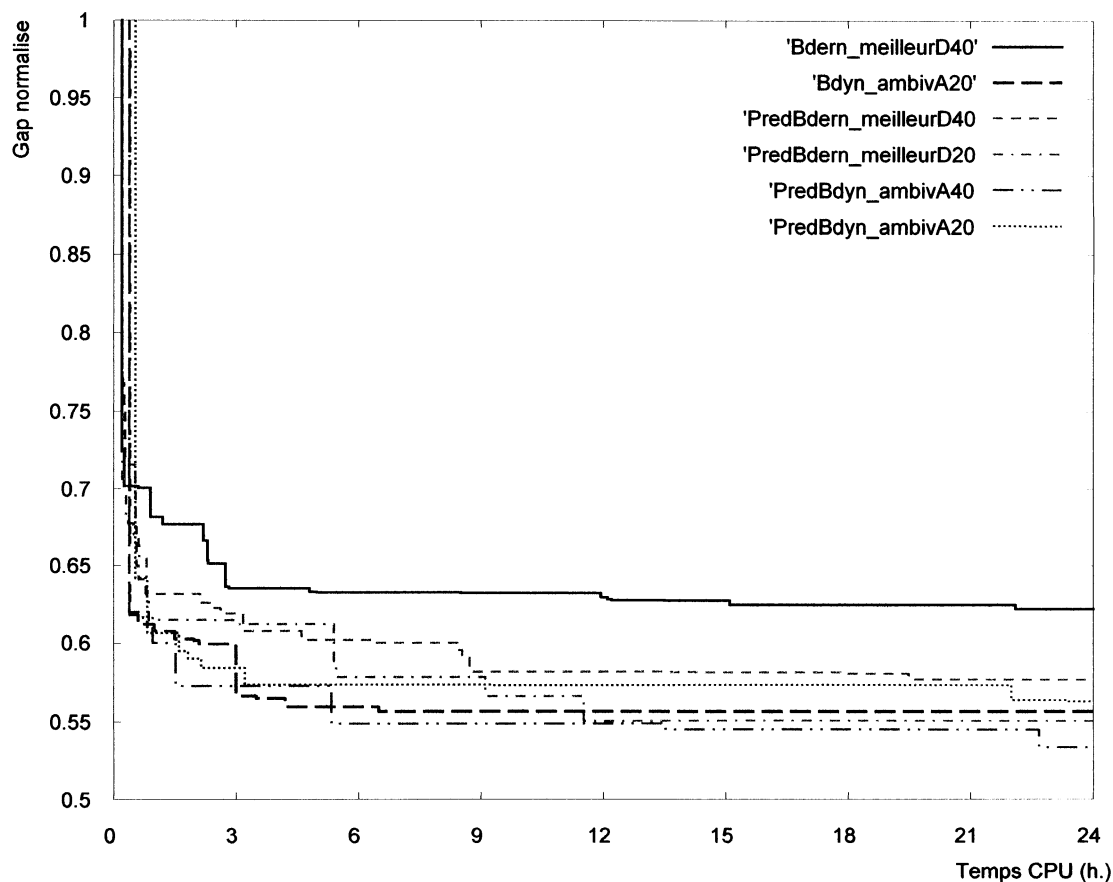


Figure 6.11: Comparaison du coût moyen de la meilleure solution entière pour tous les scénarios avec et sans schéma prédicteur-correcteur

nouvelles solutions entières intéressantes, comparables en moyenne avec la meilleure combinaison précédente de tactique de branchement et de valeur de paramètre. Après vingt-quatre heures de temps de calcul, les gains varient entre 5% et -0,5% pour un gain moyen de 1,5% sur les

quatre combinaisons de tactique de branchement et de paramètre. Ramenée en terme d'économies en utilisant le scénario comme référence typique d'exploitation, cela représente près de 300 000\$ annuellement pour une compagnie comme le CN. Notez toutefois qu'il faut employer une valeur de paramètre de $A=40$ pour améliorer la meilleure combinaison utilisée jusqu'ici de 2,5% de saut d'intégrité relatif (équivalent à 450 000\$ pour le scénario 2).

Après six ou vingt-quatre heures de calcul en temps CPU, le schéma prédicteur-correcteur générant plusieurs noeuds-correcteurs mène à des solutions comparables avec la tactique de branchement sur le dernier meilleur chemin et moins bonnes avec le branchement dynamique ambivalent. En résumé, le schéma prédicteur-correcteur générant plusieurs noeuds-correcteurs produit des résultats légèrement meilleurs lors de la descente initiale. Toutefois les nombreux noeuds-frères ont tendance à initier plusieurs coups de sonde consécutifs similaires, sacrifiant de la variété dans les solutions considérées. Il mène donc plus lentement à de bonnes solutions que le schéma générant un seul noeud-correcteur lorsque l'on permet un retour-arrière.

Nous recommandons donc d'employer un schéma de recherche prédicteur-correcteur générant un seul noeud-correcteur, appliqué au branchement dynamique ambivalent avec paramètre $A=40$ et combiné à la stratégie d'exploration par meilleures attentes avec distance à l'intégralité pondérée par les coûts.

6.4.3) Résultats pour un scénario d'une semaine

Pour valider la qualité des outils développés plus tôt, nous avons appliqué cette dernière recommandation au même scénario employé par Ziarati et al. [64]. Ce scénario d'une semaine est traité en décomposant l'horizon de calcul — une semaine — en plusieurs fenêtres d'optimisation qui se chevauchent. La figure 6.12 illustre comment combiner plusieurs — quatre — fenêtres d'optimisation pour couvrir la totalité de l'horizon de temps étudié. Il est plus difficile d'évaluer la qualité des réductions de coûts réalisées en utilisant les relaxations linéaires comme référence. En effet, puisque certaines fenêtres d'optimisation se chevauchent, celles-ci seraient considérées deux fois. Pour ce genre de problèmes, le coût de la solution reconstituée finalement est un bon indicateur générique. Le nombre de trains couverts par rapport au nombre de locomotives employées permet aussi de représenter la qualité de la solution de façon plus concrète.

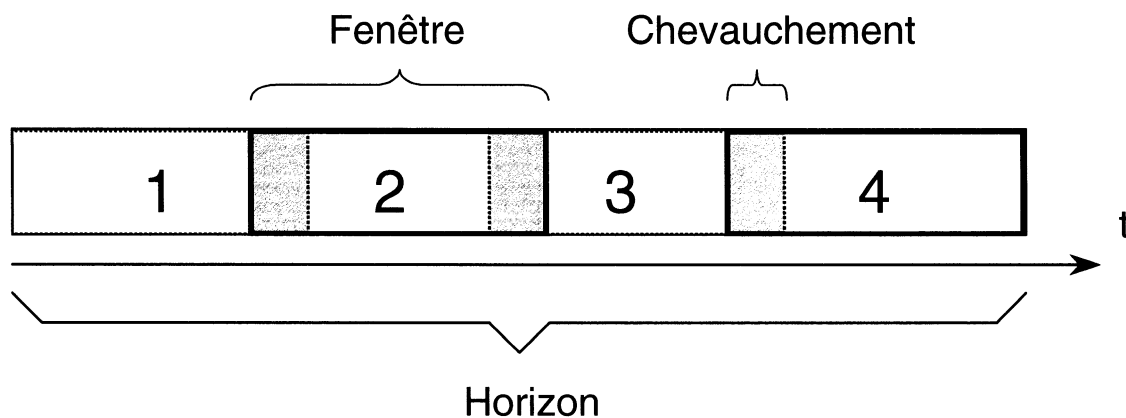


Figure 6.12: Décomposition de l'horizon de temps étudié en fenêtres d'optimisation

Pour le cas étudié, nous avons utilisé trois fenêtres de trois jours avec une journée de chevauchement, ce qui produit une solution valide sur une semaine. Nous avons employé le même temps *CPU* pour obtenir au moins une solution entière à toutes les fenêtres de temps, soit 30 minutes. Le tableau 6.2 résume en terme de locomotives et de puissance effectivement employée les gains obtenus en comparaison avec la méthode initiale de Ziarati et al. [64]. Les variations en terme de puissance déployée sont très faibles. Mais malgré un nombre moindre de locomotives, leur utilisation est plus efficace. En effet, on remarque que l'approche en coups de sonde et branchement dynamique ambivalent permet de trouver une solution qui nécessite moins de locomotives pour couvrir plus de trains. De plus, le schéma prédicteur-correcteur apporte un gain supplémentaire (moins de locomotives utilisées et plus de trains couverts). Sachant que l'amortissement de l'achat d'une locomotive et son entretien coûte environ 400 000 dollars (\$) annuellement, les gains obtenus sont d'au moins 5 600 000 dollars (\$) par année, sans oublier les économies additionnelles représentées par la réduction des pénalités de sous-recouvrement en puissance (*HP*) et tonnage (*DP*).

Tableau 6.2: Résultats pour un scénario d'une semaine

	Nombre de Locomotives	Puissance (Hp x 10 ⁷)	Nombre de trains couverts	Temps CPU (minutes)
Méthode initiale de Ziarati et al. [64]	1 134	1,029	1 595	25
Coups de sonde avec branchement dynamique	1 124	1,028	1 606	90
Coups de sonde avec branchement dynamique et schéma prédicteur- correcteur	1 120	1,030	1 611	90

CONCLUSION

Dans cette thèse, nous avons proposé plusieurs outils permettant d'améliorer les temps de calculs et la qualité des solutions entières obtenues pour des problèmes d'affectation décomposables en problème multiflot avec contraintes liantes. Ces techniques sont fonctionnelles pour résoudre tout problème de séparation et évaluation progressive par génération de colonnes, que ce soit un problème d'affectation simple ou de recouvrement généralisé tel que le problème d'affectation de locomotives pour les trains de marchandises. La taille d'un problème industriel de ce genre est telle que l'on ne peut pas énumérer l'ensemble de l'arbre de recherche: on doit se contenter d'explorer une fraction des solutions. Nos outils ont donc été développés dans ce cadre: une recherche systématique incomplète par une méthode heuristique d'énumération implicite par génération de colonnes. Plus spécifiquement, les chapitres 3, 5 et 6 cherchent à perfectionner les tactiques de branchement aux noeuds de l'arbre de recherche et le chapitre 4 à accélérer l'exploration de ce même arbre de recherche. Chacun des quatre chapitres d'application représente une étape supplémentaire pour essayer d'isoler l'élément ayant le plus d'impact sur la qualité des solutions. Même si le milieu industriel est plus intéressé par des temps de calcul de l'ordre de deux heures, nous utilisons les résultats obtenus après vingt-quatre heures de temps de calcul (CPU) pour mesurer les améliorations apportées par nos différents outils. Ainsi, on obtient des comparaisons stables malgré le petit nombre de scénarios étudiés. De plus, les temps de vingt-

quatre heures employés sur une machine de 400 MHz pourront être réduit à quelques heures avec les machines de 2 ou 3 GHz.

Au chapitre 3, nous avons étudié les tactiques de branchement dans le contexte de sous-problèmes de plus court chemin avec chemins interdits. Ce principe permet une recherche systématique de l'arbre des solutions. La résolution bénéficie ainsi de la force brute de l'ordinateur en explorant à chaque noeud une solution différente. L'interdiction de chemins est effectuée en utilisant la méthode des déviations de Villeneuve et Desaulniers [61]. Les résultats numériques obtenus avec les trois tactiques de branchement pour trois scénarios différents avec la stratégie d'exploration classique en profondeur-d'abord montre une réduction moyenne du saut d'intégrité relatif de l'ordre de 38%. Particulièrement, l'extension de la méthode de Ziarati et al. [64] par une stratégie d'exploration en profondeur-d'abord produit aussi une réduction de 38% du saut d'intégrité relativement à la solution initiale pour la moyenne des scénarios étudiés. Toutefois, il apparaît clairement que la stratégie d'exploration en profondeur-d'abord convient mal à la tactique de branchement dynamique ambivalente.

Dans le chapitre 4, nous avons exposé une autre stratégie d'exploration appelée exploration par meilleures attentes. En utilisant que les flots fractionnaires des variables représentant un chemin pour mesurer la distance à l'intégralité d'une solution du problème relaxé, la réduction relative du saut d'intégrité est de 4% en

moyenne, bien que nulle pour la tactique de branchement de référence. Par la suite, en tenant compte de l'impact des coûts des variables du modèle sur la mesure d'intégralité, la réduction de saut d'intégrité relatif toujours estimée après vingt-quatre heures de temps de calcul (CPU) reste identique, mais les premières bonnes solutions sont obtenues plus tôt. En outre, cela met en évidence la supériorité de la tactique de branchement dynamique ambivalente. Celle-ci permet un gain supplémentaire de 5% de saut d'intégrité relatif. Finalement, l'amélioration de la précision sur des objectifs entiers espérés obtenue par l'extrapolation adaptée ne se traduisant pas par de meilleures solutions entières, les travaux suivants sont donc concentrés sur des révisions des tactiques de branchement.

Au chapitre 5, nous avons essayé de prédire de nouvelles estimations des détériorations causées par une décision élémentaire grâce à des problèmes d'optimisation simplifiés pour chaque arc du réseau associé à une commodité. Malheureusement, cela n'a pas produit de meilleurs résultats que les techniques précédentes à l'exception du scénario 1 lorsque le paramètre d'accélération est réduit à sa valeur minimale ($A=1$). Nous ne retenons donc pas l'approche d'estimation des détériorations associées aux décisions élémentaires grâce à des problèmes linéaires simplifiés. Notons toutefois qu'il devient efficace de fixer des variables de chemin à des valeurs entières non-unitaires lorsque l'on emploie une approche par estimations des détériorations.

Dans le chapitre 6, nous avons décrit les étapes permettant de construire un schéma de branchement prédicteur-correcteur de recherche dans l'arbre des solutions. Ce dernier chapitre a confirmé la supériorité du branchement dynamique ambivalent lorsque combiné à un schéma prédicteur-correcteur. Les résultats obtenus sans retour-arrière pour un branchement prédicteur-correcteur appliqué à nos scénarios de référence sont les suivants: 39,5% de réduction moyenne de saut d'intégrité relatif pour une première solution entière avec plusieurs noeuds-correcteurs et 38% avec un seul noeud-correcteur. Toutefois, avec retour-arrière, le schéma prédicteur-correcteur générant un seul noeud-correcteur produit de meilleurs résultats, soit un gain de 4% en terme de saut d'intégrité relatif sur vingt-quatre heures de temps de calcul (CPU). Avec les méthodes précédentes, il faudrait plusieurs heures de temps de calcul additionnel pour obtenir la même réduction.

La combinaison des quatre composantes citées plus haut constitue, après 24 heures de temps de calcul, une réduction de saut d'intégrité de 47% relativement aux solutions obtenues par la méthode initiale de Ziarati et al. [64] pour la moyenne des scénarios étudiés. La figure suivante permet d'apprécier l'apport successif de chacune de ces composantes. Nous recommandons donc d'employer un schéma de recherche prédicteur-correcteur générant un seul noeud-correcteur, appliqué au branchement dynamique ambivalent avec paramètre $A=40$ et combiné à la stratégie d'exploration par meilleures attentes avec distance à

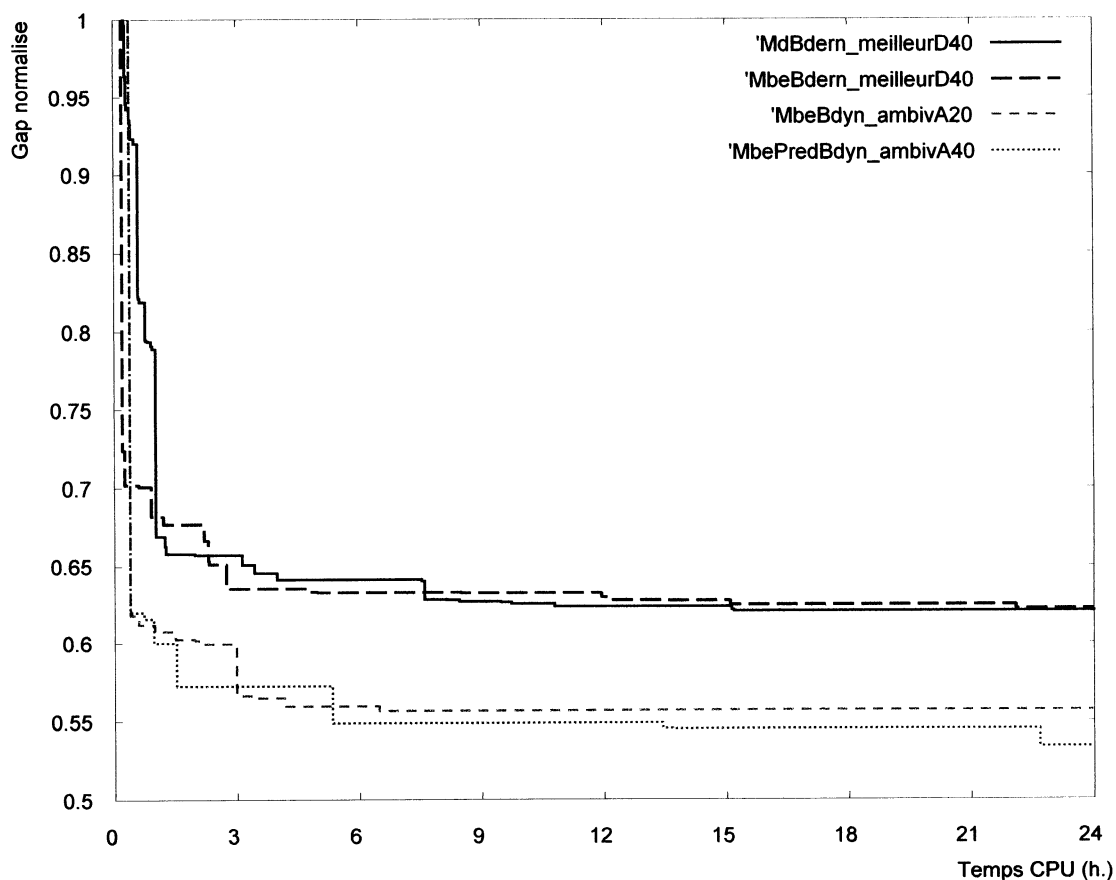


Figure 7.1: Comparaison du coût de la meilleure solution entière pour tous les scénarios en ajoutant chaque composante

l'intégralité pondérée par les coûts. En utilisant le scénario 2 comme référence, ces économies peuvent être estimées à plus de six millions de dollars (6 340 000 \$) annuellement pour un transporteur de marchandises comme le Canadian National. Cette estimation est confirmée par les résultats obtenus pour le scénario d'une semaine. En outre, nous arrivons clairement à couvrir plus de trains avec moins de locomotives. Ces économies s'additionnent à celles déjà effectuées par Ziarati et al. [64] et estimées à 24 millions de dollars.

En plus de l'élément financier important, cette thèse peut donner lieu à de nombreux développements futurs. Tout d'abord, au chapitre 5, nous n'avons pas réussi à exploiter la structure de coûts particulière du problème d'affectation des locomotives pour obtenir de meilleurs guides vers une solution entière sans recalculer de relaxation linéaire. A défaut de pouvoir mieux identifier les locomotives susceptibles de participer à une solution entière optimale, l'utilisation des facettes 3-*consist* devrait être considérée comme étape suivante pour calculer une borne inférieure plus serrée. Deuxièmement, nous ajoutons actuellement les contraintes dynamiques non-respectées sans considérer aucun autre type de décisions. Or, les contraintes dynamiques violées sont généralement peu nombreuses à un noeud. Nous pourrions simultanément continuer de prendre des décisions de branchement sans lien direct avec le(s) train(s) concerné(s) par la(es) contrainte(s) violée(s). En évitant de réoptimiser ces noeuds intermédiaires des travaux précédents ont mené à une réduction pouvant atteindre 10% du temps de calcul pour trouver une première solution entière. Toutefois, cette stratégie peut parfois mener à des solutions non-réalisables. Mieux qu'une technique de retour-arrière ou de *backmarking*, la réouverture du branchement par un schéma prédicteur-correcteur approprié permettrait de garantir une solution réalisable sur l'une des branches lorsque nécessaire. Troisièmement, puisque pour une seule instance d'un problème donné, on a constaté que d'autres combinaisons de tactiques de branchement et de stratégies d'exploration sont susceptibles de trouver plus tôt de meilleures solutions entières,

nous pourrions placer d'autres combinaisons en compétition en parallélisant différentes recherche en coups de sonde.

Enfin, les outils proposés sont aussi fonctionnels pour tout problème multiflot à contraintes liantes comportant des sous-problèmes de plus court chemin avec ou sans ressource. D'autres économies sont donc envisageables pour d'autres problèmes de même type, avec un minimum d'adaptation.

BIBLIOGRAPHIE

- [1] AHO, A.V. et CORASICK, M.J. (1975). *Efficient string matching: an aid to bibliographic search*, Journal of the ACM, Vol.18, n.6, p.333-340, Prentice Hall.

- [2] AHUJA, R.K., LIU, J., ORLIN, J.B., SHARMA, D. et SHUGHART, L.A. (Avril 2002). *Solving real-life locomotive scheduling problems*, Sloan Working Paper n.4389-02, MIT, Cambridge.

- [3] AHUJA, R.K., MAGNANTI, T.L. et ORLIN, J.B. (1993). Network Flows: Theory, Algorithms, and Applications, p.93-120, Prentice Hall, Englewood Cliffs (NJ).

- [4] AL-KHAYYAL, F.A. (Septembre 1987). *An implicit enumeration procedure for the general linear complementarity problem*, Mathematical Programming Study, Vol.31, p.1-20.

- [5] ALJ, A. et FAURE, R. (1986). Guide de la recherche opérationnelle, Tome 1, p.232-240, Masson, Paris.

- [6] ALJ, A. et FAURE, R. (1990). Guide de la recherche opérationnelle, Tome 2, p.327-344, Masson, Paris.

- [7] ANANTHARAMAN, T., CAMPBELL, M.S. et HSU, F.H. (1990). *Singular extensions: adding selectivity to brute-force searching*, Artificial Intelligence, Vol.43, p.99-109, North-Holland.

- [8] ASSAD, A.A. (1982). *A class of train scheduling problems*, Transportation Science, Vol.16, p.281-310.
- [9] BAGCHI, A. et MAHANTI, A. (1983). *Search algorithms under different kinds of heuristics - A comparative study*, Journal of the Association for Computing Machinery, Vol.30, n.1, p.1-21.
- [10] BALAS, E. (1965). *An additive algorithm for solving linear programs with 0-1 variables*, Operations Research, Vol.13, p.517-549.
- [11] BARNHART, C., JOHNSON, E.L., NEMHAUSER, G.L., SAVELSBERGH, M.W.P. et VANCE, P.H. (1998). *Branch-and-price: column generation for solving huge integer programs*, Operations Research, Vol.46, n.3, p.316-329.
- [12] BARTLETT, T.E. et CHARNES, A. (1957). *Cyclic scheduling and combinatorial topology: assignment and routing of motive power to meet scheduling and maintenance requirements. Part II. Generalization and analysis*, Naval Research Logistic Quarterly, Vol.4, p.207-220.
- [13] BIXBY, R.E. et LEE, K. (1998). *Solving a truck dispatching scheduling problem using branch-and-cut*, Operations Research, Vol.46, n.3, p.355-367.
- [14] CHARLAND, Y. (Août 1996). Reconstruction d'itinéraires optimaux dans un réseau multimodes-multiproduits, mémoire de Maîtrise, Université de Montréal.

- [15] CHIH, K.C., HORNUNG, M.A., ROTHENBERG, M.S., KORNHAUSER, A.L., et WORKMAN, J.D. (1990). *Implementation of a real time locomotive distribution system*, Computer Applications in Railway Planning and Management, p.39-49, T.K.S. Murthy et al. (eds), Computational Mechanics Publications, Southampton.
- [16] CORDEAU, J.F., DESAULNIERS, G., LINGAYA, N., SOUMIS, F. et DESROSIERS, J. (2001). *Simultaneous locomotive and car assignment at VIA Rail Canada*, Transportation Research Part B, Vol.35, p.767-787.
- [17] CORDEAU, J.F., SOUMIS, F. et DESROSIERS, J. (2000). *A Benders decomposition approach for the locomotive and car assignment problem*, Transportation Science, Vol.34, p.133-149.
- [18] CORDEAU, J.F., SOUMIS, F. et DESROSIERS, J. (2001). *Simultaneous assignment of locomotives and cars to passenger trains*, Operations Research, Vol.49, p.531-548.
- [19] CORDEAU, J.F., TOTH, P. et VIGO, D. (1998). *A survey of optimization models for train routing and scheduling*, Transportation Science, Vol.32, p.380-404.
- [20] DANTZIG, G.B. et WOLFE, P. (1960). *Decomposition principle for linear programs*, Operations Research, Vol.8, p.101-111.
- [21] DESAULNIERS, G., DESROSIERS, J., GAMACHE, M. et SOUMIS, F. (1998). *Crew scheduling in air transportation*, Fleet Management and Logistics, p.169-185, T.G. Crainic et G. Laporte (eds), Kluwer, Boston (MA).

- [22] DESAULNIERS, G., DESROSIERS, J., IOACHIM, I., SOLOMON, M.M., SOUMIS, F. et VILLENEUVE, D. (1998). *A unified framework for deterministic time constrained vehicle routing and crew scheduling problems*, Fleet Management and Logistics, p.57-93, T.G. Crainic et G. Laporte (eds), Kluwer, Boston (MA).

- [23] DESROSIERS, J., DUMAS, Y., SOLOMON, M.M. et SOUMIS, F. (1995). *Time constrained routing and scheduling*, Handbooks in Operations Research and Management Science, Vol.8, p.35-139, Network Routing, M.O. Ball et al. (eds), Elsevier Science B.V., Amsterdam.

- [24] EPPSTEIN, D. (1998). *Finding the k shortest paths*, SIAM Journal on Computing, Vol.28, n.2, p.652-673.

- [25] FLORIAN, M., BUSCHELL, G., FERLAND, J., GUÉRIN, G. et NASTANSKY, L. (1976). *The engine scheduling problem in a railway network*, INFOR, Vol.14, p.121-138.

- [26] GAMACHE, M., SOUMIS, F., MARQUIS, G. et DESROSIERS, J. (Mars 1999). *A column generation approach for large-scale aircrew rostering problems*, Operations Research, Vol.47, n.2, p.247-263.

- [27] GASCHNIG, J. (1979). Performance measurement and analysis of certain search algorithms, thèse de Doctorat, University of Carnegie-Mellon.

- [28] GAUVIN, J. (1995). Leçons de programmation mathématique, Éditions de l'École Polytechnique de Montréal.

- [29] GOMORY, R.E. (1960). An algorithm for the mixed integer problem, The Rand Corporation, rapport technique **RM-2597**.
- [30] GONDRAN, M. et MINOUX, M. (1979). Graphes et Algorithmes, p.496-531, Eyrolles, Paris.
- [31] HANSEN, P., JAUMARD, B. et POGGI de ARAGAO, M. (1991). Un algorithme primal de programmation linéaire généralisée pour les programmes mixtes, Compte-rendus de l'Académie des Sciences de Paris, Tome **313**, Série **1**, p.557-560.
- [32] HANSEN, P., JAUMARD, B. et POGGI de ARAGAO, M. (1998). *Mixed-integer column generation algorithms and the probabilistic maximum satisfiability problem*, European Journal of Operational Research, Vol.**108**, p.671-683.
- [33] HANSEN, P., MINOUX, M. et LABBÉ, M. (1987). Extension de la programmation linéaire généralisée au cas des programmes mixtes, Compte-rendus de l'Académie des Sciences de Paris, Tome **305**, Série **1**, p.569-572.
- [34] HARALICK, R.M. et ELLIOT, G.L. (1980). *Increasing tree searching efficiency for constraint satisfaction problems*, Artificial Intelligence, Vol.**14**, p.263-313.
- [35] HARRIS, L.R. (1974). *The heuristic search under conditions of error*, Artificial Intelligence, Vol.**5**, n.**3**, p.217-234.

- [36] HJORRING, C. et HANSEN, J. (2000). *Column generation with a rule modelling language for airline crew pairing*, rapport technique, Carmen Systems AB, Gothenburg.
- [37] HOFFMAN, K.L. et PADBERG, M. (1993). *Solving airline crew scheduling problems by branch-and-cut*, Management Science, Vol.39, p.657-682.
- [38] JAUMARD, B., MEYER, C. et VOVOR, T. (Février 1999). *How to combine a column and row generation method with a column or row elimination procedure - Application to a channel assignment problem*, Les Cahiers du GERAD G-99-18, HEC, Montréal.
- [39] LAND, A.H. et DOIG, A.G. (Juillet 1960). *An automatic method of solving discrete programming problems*, Econometrica, Vol.28, n.3, p.497-520.
- [40] LASDON, L.S. (1970). Optimization Theory for Large Systems, p.144-234, Mac Millan, New-York.
- [41] LAWLER, E.L. et WOOD, D.W. (1966). *Branch-and-bound methods: a survey*, Operations Research, Vol.14, p.699-719.
- [42] LITTLE, J.D.C., MURTY, K.G., SWEENEY, D.W. et KAREL, C. (1963). *An algorithm for the traveling-salesman problem*, Operations Research, Vol.11, p.972-989.
- [43] LÜBBECKE, M.E. et ZIMMERMANN, U.T. (Mai 2003). *Engine routing and scheduling at industrial in-plant railroads*, Transportation Science, Vol.37, n.2, p.183-197.

- [44] LUENBERGER, D.G. (1989). Linear and Nonlinear Programming (2nd edition), p.30-103, Addison-Wesley, Reading.
- [45] MANS, B. et ROUCAIROL, C. (1996). *Performances of parallel branch and bound algorithms with best-first search*, Discrete Applied Mathematics, Vol.**66**, p.57-74.
- [46] McMILLAN Jr., C. (1970). Mathematical Programming: an introduction to the design and application of optimal decision machines, p.271-362, Wiley, Toronto.
- [47] MIAOU, S.P. et CHIN, S.M. (1991). *Computing k-shortest path for nuclear spent fuel highway transportation*, European Journal of Operational Research, Vol.**53**, p.64-80, North-Holland.
- [48] NEMHAUSER, G.L. et SIGISMONDI, G. (1992). *A strong cutting plane/branch-and-bound algorithm for node packing*, Journal of the Operational Research Society, Vol.**43**, n.5, p.443-457.
- [49] NEMHAUSER, G.L. et WOLSEY, L.A. (1988). Integer and Combinatorial Optimization, p.30-545, Wiley, New-York.
- [50] NEWTON, H.N. (1996). Network design under budget constraints with application to the railroad blocking problem, thèse de Doctorat, Auburn University.
- [51] NOU, A., DESROSIERS, J. et SOUMIS, F. (1997). *Weekly locomotive scheduling at swedish state railways*, Les Cahiers du GERAD G-97-35, HEC, Montréal.

- [52] PEARL, J. (1984). Heuristics: intelligent search strategies for computer problem solving, Addison Wesley, Series in artificial intelligence, p.1-212, Reading (MA).
- [53] POHL, I. (1977). *Practical and theoretical considerations in heuristic search algorithms*, Machine Intelligence, Vol.8, p.55-72, Wiley, New-York.
- [54] POSPICHAL, J. et KVASNICKA, V. (1996). *Pruning the search tree in the constructive enumeration of molecular graphs*, Discrete Applied Mathematics, Vol.67, p.189-207, Bratislava.
- [55] ROSING, K.E. et REVELLE, C.S. (1997). *Heuristic concentration: two stage solution construction*, European Journal of Operational Research, Vol.97, p.75-86.
- [56] SMITH, S. et SHEFFI, Y. (1988). *Locomotive scheduling under uncertain demand*, Transportation Research Record, n.1251, p.45-53.
- [57] TAHA, H.A. (1975). Integer Programming: theory, applications and computations, p.52-173, Academy Press, New-York.
- [58] TOMLIN, J.A. (1969). *An improved branch-and-bound method for integer programming*, p.1070-1075, notes techniques, Scientific Control Systems, London.
- [59] TSENG, C.H., WANG, L.W. et LING, S.F. (Mai 1995). *Enhancing branch-and-bound methods for structural optimization*, Journal of Structural Engineering, Vol.121, n.5, p.831-837.

- [60] VILLENEUVE, B. (Juillet 2000). Répartition et routage d'un système automatique de chariots à l'aide d'une méthode de génération de colonnes, mémoire de Maîtrise, École Polytechnique de Montréal.
- [61] VILLENEUVE, D. et DESAULNIERS, G. (Août 2000). *The shortest path problem with forbidden paths*, Les Cahiers du GERAD G-2002-41, HEC, Montréal.
- [62] YU, B. et YUAN, B. (1993). *A more efficient branch and bound algorithm for feature selection*, Pattern Recognition, Vol.26, p.883-889.
- [63] ZIARATI, K. (Mars 1997). Affectation des locomotives aux trains, thèse de Doctorat, École Polytechnique de Montréal.
- [64] ZIARATI, K., SOUMIS, F., DESROSIERS, J., GÉLINAS, S. et SAINTONGE, A. (1997). *Locomotive assignment with heterogeneous consists at CN North America*, European Journal of Operational Research, Vol.97, p.281-292.
- [65] ZIARATI, K., SOUMIS, F., DESROSIERS, J. et SOLOMON, M.M. (1999). *A branch-first, cut-second approach for locomotive assignment*, Management Science, Vol.45, p.1156-1168.

ANNEXE

Dans un souci de simplicité et de continuité, plusieurs notations sont identiques à celles utilisées par Ziarati [63].

N : ensemble des noeuds

A : ensemble de tous les arcs

$G = (N, A)$: réseau espace-temps permettant de modéliser le problème

A^k : ensemble des arcs composant le réseau associé à une commodité k

$(i, j) \in A^k$: un arc du réseau précédent reliant les noeuds i et j

X_{ij}^k : variable entière non-négative de flot sur l'arc $(i, j) \in A^k$

F_{ij}^k : nombre entier de *deadhead* de classe k sur l'arc (i, j)

c_{ij}^k : coût d'une locomotive active de classe k passant sur l'arc (i, j)

d_{ij}^k : coût de *deadhead* pour une locomotive de classe k sur l'arc (i, j)

S : ensemble des indices des variables supplémentaires

$s \in S$: indice désignant une variable supplémentaire

S^n : ensemble des variables supplémentaires associées aux contraintes de recouvrement en nombre de locomotives

S^p : ensemble des variables supplémentaires associées aux contraintes de recouvrement en puissance

S^q : ensemble des variables supplémentaires associées aux contraintes de recouvrement en tonnage

Y_s : variable supplémentaire d'indice $s \in S$

c_s : coût correspondant à la variable supplémentaire Y_s

W^n : ensemble des segments de train dont la demande est exprimée en terme du nombre minimum de locomotives

$w \in W$: indice désignant une tâche

$a_{w,ij}^k$: coefficient binaire de couverture de la tâche w par la variable de flot X_{ij}^k

- a_{ws} : coefficient de couverture de la tâche w par la variable Y_s
 n_w : nombre minimum de locomotives pour couvrir la tâche $w \in W^n$
 W^p : ensemble des segments de train dont la demande de puissance est exprimée en chevaux-vapeur (hp)
 p^k : puissance opérationnelle des locomotives de classe k
 p_w : demande de puissance en hp pour le train $w \in W^p$
 W^q : ensemble des trains dont la demande de traction/poids est exprimée en tonnes en considérant les élévations le long du parcours
 q^k : poids total qui peut être tiré par une locomotive de classe k
 q_w : tonnage du train $w \in W^q$ qui doit être tiré par les locomotives
 D : ensemble de périodes cycliques (exemple: jours de la semaine)
 $d \in D$: période quelconque du cycle des disponibilités et besoins
 M : ensemble des ateliers
 $m \in M$: un atelier de cet ensemble
 $g_{m,ij}^{kd}$: coefficient binaire de disponibilité de l'atelier $m \in M$ au jour $d \in D$ pour la variable de flot X_{ij}^k
 n_m^d : capacité maximum de l'atelier $m \in M$ le jour $d \in D$
 O : ensemble des postes locaux
 $o \in O$: un poste local de cet ensemble
 $g_{o,ij}^{kd}$: coefficient binaire de la variable de flot X_{ij}^k , 1 si l'arc (i,j) est un arc de poste local o au jour d , 0 sinon
 n_o^d : demande en nombre de locomotives au poste local o le jour d
 H : ensemble des points de changement de puissance (PCP)
 $h \in H$: un PCP de cet ensemble
 $g_{h,ij}^k$: coefficient binaire de la variable de flot X_{ij}^k , 1 si l'arc (i,j) est un arc de cible pour le PCP h , 0 sinon
 p_h : puissance demandée cycliquement au PCP h
 $K^c \subset K$: ensemble des locomotives critiques
 t_{ij}^k : consommation de la ressource (temps) entre les noeuds i et j