

2411.2646.13

Université de Montréal

Mise au point et implantation d'algorithmes
pour l'allocation déterministe de conteneurs vides

par

Jawad Abrache

Département d'informatique et de recherche opérationnelle

Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures
en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en Informatique

Août, 1998

©Jawad Abrache, 1998



21 2026.1118

QA
76
U54
1998
V.027

Université de Montréal

Mise au point et finalisation d'algorithmes
pour l'abstraction d'événements de composants réels

1998

Jawad Alami

Département d'informatique et de génie logiciel

Faculté des arts et des sciences

Mémoire présenté à la Faculté des études supérieures

en vue de l'obtention du grade de

Maître en sciences (M.Sc.)

en informatique

1998

Jawad Alami, 1998



Université de Montréal
Faculté des études supérieures

Ce mémoire intitulé :

“Mise au point et implantation d’algorithmes
pour l’allocation déterministe de conteneurs vides”

présenté par :
Jawad Abrache

a été évalué par un jury composé des personnes suivantes :

Prof. Bernard Gendron	:	président rapporteur
Prof. Teodor Gabriel Crainic	:	directeur de recherche
Prof. Michel Gendreau	:	co-directeur de recherche
Prof. Sang Nguyen	:	membre du jury

Mémoire accepté le: 13.10.1998

Sommaire

L'allocation de conteneurs vides est un problème important rencontré par les compagnies de transport maritime dans le cadre de la gestion du transport terrestre de marchandises. C'est un problème caractérisé par une prise de décision, à court terme, concernant aussi bien l'envoi de conteneurs vides en réponse à des besoins des clients pour l'exportation de leurs produits, que la récupération des conteneurs par les entrepôts de la compagnie suite aux opérations d'importation.

L'objectif de ce mémoire est le développement d'algorithmes pour une allocation optimale des conteneurs vides. Ainsi, sur la base d'un modèle dynamique et multiproduits conçu pour le cas déterministe, nous identifions quelques caractéristiques importantes du problème, notamment la structure multipériodes du réseau espace-temps associé et l'existence, ainsi que la disposition particulière, d'arcs spéciaux représentant la propriété de substitution entre les différents produits qu'introduit le modèle.

Bien que la résolution du problème puisse très bien s'effectuer en faisant appel à des algorithmes généraux de flot à coût minimum dans les réseaux généralisés, dont nous effectuons une revue sommaire, nous proposons une nouvelle approche de décomposition primale fondée sur la restriction de la valeur du flot sur les arcs de substitution. L'algorithme générique qui en découle est à la base de plusieurs stratégies particulières de résolution.

Dans le but de mettre à l'épreuve notre approche de décomposition, nous réalisons l'implantation d'une adaptation efficace du simplexe aux réseaux avec gains, sur la base de laquelle nous effectuons une analyse comparative de l'efficacité des versions séquentielles et parallèles de trois de nos stratégies de décomposition. L'expérimentation est réalisée sur un réseau de stations de travail, en utilisant plusieurs échantillons de données générés grâce à une procédure que nous mettons au point. Ces tests numériques nous permettent principalement de tirer les conclusions suivantes :

- Les versions séquentielles de nos approches de décomposition du problème don-

ment des résultats extrêmement avantageux par rapport à une résolution brute du problème par le simplexe réseau.

- La densité des réseaux, et surtout l'intensité de la substitution sont des facteurs importants dans la complexité des problèmes d'allocation de conteneurs vides.
- L'une ou l'autre des approches de décomposition peut s'avérer particulièrement bien adaptée à un problème possédant une structure de substitution spéciale.
- Le parallélisme à gros grain dans un environnement distribué, mis en œuvre par les versions parallèles de nos approches de décomposition, apporte sa part d'accélération par rapport aux versions séquentielles. Cependant, le gain d'efficacité observé n'est pas proportionnel aux ressources mobilisées (nombre de stations utilisées), ce qui incite à explorer d'autres voies de développement, notamment les techniques de parallélisme à moindre granularité et la mise en œuvre des algorithmes sur d'autres plate-formes de traitement parallèle.

Mots-clés: Transport, gestion de la flotte terrestre, allocation multiproduits de conteneurs vides, substitution, réseaux généralisés, approche de décomposition, parallélisme, environnement distribué.

Table des matières

Sommaire	iii
Table des matières	v
Table des figures	viii
Liste des tableaux	ix
Introduction	1
1 Problématique	5
1.1 Le contexte général	6
1.2 Les caractéristiques du problème	7
1.2.1 Le réseau physique et les modes de transport	7
1.2.2 Les régimes de transport	8
1.2.3 L’homogénéité de la flotte	10
1.2.4 Les mouvements de conteneurs	11
1.3 Une approche méthodologique	13
1.4 La littérature sur l’allocation dynamique des véhicules	15
2 Un modèle d’allocation	18
2.1 Les éléments du problème	19
2.2 Le modèle mathématique	20
2.2.1 Notations et définitions	20
2.2.2 Les contraintes du modèle	23
2.2.3 La fonction économique	25
2.3 La formulation réseau	26

3	Méthodes de résolution	31
3.1	Résolution du flot à coût minimum dans les réseaux avec gains	32
3.1.1	Perspective historique	32
3.1.2	Définition et caractérisation du problème	33
3.1.3	Caractérisation des bases	35
3.1.4	Les structures de données pour la représentation de la base . .	36
3.1.5	L'algorithme du simplexe réseau avec gains	37
3.2	Une approche de résolution exploitant la structure multiproduits . . .	43
3.2.1	L'approche de restriction classique	44
3.2.2	Spécialisation au problème du flot à coût minimum dans la représentation réseau	46
3.2.3	Les avantages potentiels de l'algorithme	49
3.3	Prise en compte des propriétés dynamiques du réseau	50
3.3.1	Un algorithme progressif du type simplexe réseau	51
4	Implantation et exploitation des données	55
4.1	Organisation des données	56
4.1.1	Une classification des données du modèle mathématique . . .	57
4.1.2	La mise en œuvre des données	62
5	Résultats numériques	66
5.1	Objectifs	67
5.2	La génération aléatoire de problèmes test	67
5.2.1	Paramètres de la génération	68
5.3	Résultats des versions séquentielles	72
5.3.1	Conclusion	86
5.4	Versions parallèles	86
5.4.1	Un algorithme parallèle pour la première phase	87
5.4.2	Parallélisation de la seconde phase	87
5.4.3	Tests sur quelques problèmes	89
5.4.4	Conclusion	92
	Conclusion	93

	vii
Bibliographie	95
A Un exemple de fichier de données	i
B La représentation réseau correspondante	xi
Remerciements	xix

Table des figures

1.1	Configuration type d'un réseau physique de base	9
1.2	Représentation d'un processus de décision	15
2.1	Illustration de la formulation réseau du problème	29
3.1	Un exemple d'arbre augmenté	37
3.2	Mise à jour des variables duales sur un cycle	39
3.3	Le chemin de précedence dans le premier cas	40
3.4	Le chemin de précedence dans le second cas	41
4.1	Sections et attributs associés aux requêtes	59
4.2	Sections et attributs associés à l'allocation et au retour des conteneurs vides	60
4.3	Algorithme de construction de la représentation réseau	64
5.1	La seconde phase des deux premières stratégies de fusion	74
5.2	La seconde phase de la stratégie de fusion progressive binaire	79
5.3	Transformation des sous-problèmes issus de la décomposition duale en problèmes de flot à coût minimum	95
B.1	Le sous-réseau du produit 1	xii
B.2	Le sous-réseau du produit 2	xiii
B.3	Le sous-réseau du produit 3	xiv

Liste des tableaux

3.1	Représentations ATI et API d'un arbre augmenté	37
4.1	Sections et attributs généraux	58
4.2	Sections et attributs relatifs aux mouvements et opérations autres que ceux de l'allocation et de la récupération de conteneurs	61
5.1	Les variables aléatoires associées à la structure physique du réseau . .	69
5.2	Les variables aléatoires associées à l'activité des dépôts	70
5.3	Les variables aléatoires associées à l'allocation et au retour des conteneurs	71
5.4	La série S1 de problèmes de petite taille	75
5.5	Résultats comparatifs pour la première série de problèmes	76
5.6	Détails de la stratégie de fusion globale pour la première série	76
5.7	Détails de la stratégie de fusion progressive à une étape pour la première série	77
5.8	La série S2 de problèmes généraux de taille moyenne	78
5.9	Résultats comparatifs pour la seconde série de problèmes	80
5.10	La série S3 de problèmes généraux denses	81
5.11	La série S4 de problèmes à substitution intense	81
5.12	Les résultats des problèmes denses et à activité de substitution intense	81
5.13	Détails comparatifs de la fusion globale pour les problèmes S2R04 et S4R03	82
5.14	Les séries S5 et S6 de problèmes à structure particulière de substitution	84
5.15	Les résultats des problèmes à structure particulière de substitution . .	85
5.16	Les résultats comparatifs des versions séquentielles et parallèles des approches de fusion	91

À mes parents,
M'barek et Halima

Introduction

Le transport de biens a toujours été considéré comme une activité socio-économique d'une importance capitale, que ce soit de façon directe, à travers sa part dans le Produit national brut et dans le prix de revient des matières premières comme des produits finis et semi-finis, ou encore indirectement en faisant intervenir un large spectre d'acteurs (transporteurs, producteurs, instances gouvernementales, etc.), et en agissant comme moteur au développement d'activités commerciales et industrielles associées. Disposer d'infrastructures de transport étendues et diversifiées, ainsi que des moyens pour une gestion efficace de la flotte et des équipements devient donc un objectif prioritaire.

La diversité des intervenants, ainsi qu'une certaine tendance, observée durant les deux dernières décennies, à assouplir la réglementation du transport de biens, ont créé un environnement extrêmement compétitif où les compagnies recherchent aussi bien l'efficacité en termes de profits réalisés et de coûts encourus, qu'un niveau élevé de la qualité de service à la clientèle, notamment en ce qui a trait à la rapidité et à la fiabilité de l'acheminement des cargaisons.

Ces facteurs ont contribué à augmenter la complexité des processus de planification et de décision et des procédures opérationnelles des systèmes de transport. Dans des domaines aussi variés que la prise de décision interne des compagnies, la gestion des relations avec la clientèle et les partenaires commerciaux, la gestion des flottes et des ressources (véhicules, surfaces d'entreposage, etc.), cette complexité se manifeste en mobilisant une part considérable des moyens matériels et humains.

À ce niveau, l'apport des méthodes de la recherche opérationnelle, dans le

développement de modèles et d'algorithmes pour la planification et la gestion des opérations de transport, est déterminant. De plus en plus, les procédures développées sont intégrées à des outils interactifs de prise de décision, à interface graphique. Sont également mises à contribution les méthodes de gestion, ainsi que les techniques de simulation et les technologies modernes de l'information dans la mise en place de systèmes capables de la cueillette et du traitement des données en temps réel.

Dans le cadre particulier du transport terrestre de véhicules, les problèmes rencontrés en pratique sont de nature très variée. D'une part, sur le plan logistique, on retrouve principalement des modèles de localisation multiproduits simple de dépôts, ou de localisation multiproduits avec affectation des dépôts aux clients, ainsi que des problèmes plus généraux de conception de réseaux où le but est de choisir, parmi les liens possibles pour la formation d'un réseau de distribution, ceux qui permettent un transport au moindre coût. D'autre part, les modèles opérationnels les plus importants sont des formulations dynamiques et stochastiques associées à l'allocation et au repositionnement des ressources nécessaires au transport, à l'agencement des opérations et au routage des véhicules chargés et vides.

Tout au long de ce mémoire, nous nous intéresserons en particulier à l'allocation déterministe, dynamique et multiproduits de conteneurs vides. Il s'agit d'un problème d'un grand intérêt, et ce à plusieurs égards : pour plusieurs types de biens, le transport par conteneurs est la solution la plus avantageuse, sur les plans économique et sécuritaire ; en outre, et parmi tous les mouvements observés de conteneurs, l'importance relative des mouvements à vide est considérable (jusqu'à 40%, selon une étude menée dans le contexte européen (Crainic, Gendreau et Dejax [20])). D'autre part, l'allocation est un sous-problème fondamental qui, comme nous allons le voir plus loin, se situe au cœur du processus de planification global. Bien entendu, l'hypothèse de base des modèles déterministes que nous utilisons, à savoir que le comportement des agents du système est connu avec certitude, est très peu réaliste ; toutefois, il est très instructif de pouvoir comparer les résultats des approches déterministes et ceux des méthodes stochastiques, afin par exemple de constater une amélioration relative, ou de tirer des conclusions sur la véracité des hypothèses déterministes dans des cas

de figures particuliers. Enfin, ces approches de résolution ont l'avantage d'être abordables par un large éventail de méthodes classiques et par rapport aux algorithmes stochastiques, nécessitent un moindre recours par les modèles à des hypothèses simplificatrices.

Notre travail vise les objectifs suivants :

- À partir d'un modèle mathématique conçu spécialement pour l'allocation déterministe, dynamique et multiproduits de conteneurs vides et d'une formulation réseau associée (Crainic, Gendreau et Dejax [20]), la conception de plusieurs approches de résolution primales. Celles-ci seront basées sur une implantation du simplexe réseau généralisé que nous nous proposons de mettre au point. Notre but est, dans un premier temps, d'appliquer cette implantation à une résolution directe du problème ; ensuite, nous adoptons une démarche structurelle de restriction qui tire parti des propriétés multiproduits de notre problème afin de développer une classe générique de stratégies de résolution séquentielles et parallèles efficaces.
- Le développement, pour le problème de l'allocation des conteneurs vides, d'un modèle pour les données susceptible de s'intégrer au sein d'un système d'information global.

Le présent mémoire s'organise comme suit : le premier chapitre sera consacré à donner une description du problème en identifiant son contexte général et ses caractéristiques les plus importantes, à présenter une approche méthodologique qui permet de distinguer le problème de l'allocation de conteneurs vides dans le processus général de décision et à établir une revue de la littérature de l'allocation des véhicules dans le domaine du transport terrestre de marchandises. Nous présentons ensuite, dans un second volet, les formulations mathématique et en réseau d'un modèle déterministe, dynamique et multiproduits proposé par Crainic, Gendreau et Dejax [20]. Par la suite, nous verrons dans un troisième chapitre les méthodes de résolution de base du problème de flot identifié, et en particulier une adaptation du simplexe primal aux réseaux généralisés, ainsi qu'une nouvelle approche

de décomposition exploitant les propriétés multiproduits propres à notre application. Dans un quatrième chapitre consacré à la mise en œuvre des algorithmes, nous présentons un format de fichier pour la représentation des données du problème, ainsi que les principaux traitements visant l'exploitation de ces données. Nous réalisons dans un cinquième chapitre une analyse comparative détaillée des résultats des tests d'implantations séquentielles et parallèles de nos stratégies de résolution. Nous concluons finalement par une revue des contributions apportées et de quelques perspectives de recherche et de développement futurs.

Chapitre 1

Problématique

Nous examinons dans ce chapitre la problématique associée à l'allocation déterministe de conteneurs vides en identifiant le contexte du problème, ainsi que ses principales caractéristiques. Nous mettons l'accent ensuite sur l'aspect méthodologique du processus de modélisation et de prise de décision mis en œuvre par l'allocation et par les problèmes connexes. Nous établissons enfin une revue sommaire des principaux travaux qui ont été consacrés à l'allocation de véhicules dans le cadre du transport terrestre de biens.

Les développements du chapitre se réfèrent, de façon générale, à l'étude de modélisation de l'allocation déterministe et stochastique de conteneurs vides réalisée par Crainic, Gendreau et Dejax [20], sur la base d'une application de distribution de conteneurs de la Compagnie Générale Maritime (C.G.M.).

1.1 Le contexte général

Le problème auquel nous nous intéressons est celui rencontré par des compagnies de transport maritime prenant en charge, à une échelle nationale ou continentale, le transport terrestre de conteneurs vers des clients situés à l'intérieur du continent. La situation est la suivante : pour les besoins de leurs activités industrielles ou commerciales, les clients ont des demandes de biens importés, typiquement des matières premières. Ces produits arrivent aux ports chargés dans des conteneurs, et la compagnie de transport doit acheminer ces derniers, par voie terrestre, vers les clients, où ils sont déchargés et renvoyés au port d'origine, ou encore dirigés vers un des entrepôts de la compagnie, pour les besoins futurs. D'autre part, il arrive que les clients aient besoin d'exporter des biens, qui sont en général des produits finis et semi-finis. Ils requièrent par conséquent des conteneurs vides, qui doivent dans ce cas être envoyés d'un dépôt, chargés et acheminés vers un port, où l'embarquement s'effectue vers le destinataire.

Étant donné qu'il est à l'origine de la plupart des mouvements de conteneurs vides, le repositionnement des conteneurs suite à une importation de produits est parmi les opérations les plus importantes réalisées par la compagnie. Notons d'autre part que le besoin de transporter des marchandises de nature diversifiée impose le recours à des catégories de conteneurs de tailles et de fonctionnalités différentes. Ce caractère multiproduits est d'autant plus important que la compagnie, afin d'accommoder le plus possible les besoins des clients en fonction des disponibilités, peut réaliser des opérations de *substitution* entre les différents types de conteneurs (un exemple typique est la substitution, quand cela est possible, de deux conteneurs de longueur 20 pieds à un seul conteneur de longueur 40 pieds, à fonctionnalité similaire).

Cette façon de présenter le problème, certes, met l'accent sur la composante terrestre du transport des conteneurs. Elle n'occulte cependant pas le fait qu'il existe un lien étroit entre cette composante et la gestion du transport maritime correspondante. En effet, l'acheminement des conteneurs chargés d'un certain bien à un client importateur, par exemple, est conditionné par l'arrivée d'une cargaison associée à un port.

En outre, les activités des différentes régions en matière d'exportation de marchandises et les besoins fluctuants en conteneurs qui en découlent peuvent nécessiter des opérations d'import/export de conteneurs vides, destinés à assurer la disponibilité de ces derniers, opérations qui sont également très dépendantes de la gestion des lignes de transport maritime. Le problème peut donc être formulé dans un contexte plus général où, sur une échelle mondiale, la compagnie gère globalement les transports maritimes et terrestres. Toutefois, la composante terrestre est de plus en plus reconnue comme un problème en soi, et non plus comme une partie annexe d'une activité principale de gestion du transport maritime. Cette reconnaissance est avant tout la conséquence de certaines études qui ont mis l'accent sur la contribution importante du transport terrestre au coût global des opérations, mais se justifie également par le fait que la réglementation en vigueur dans le domaine du transport maritime diffère, de façon radicale, de celle en usage sur la voie terrestre.

D'un autre point de vue, il est important de remarquer que l'allocation de conteneurs vides s'associe très étroitement à d'autres problèmes, notamment à l'affectation des véhicules (principalement des camions) et des équipages aux tâches du transport. Il convient donc de relier l'allocation des conteneurs vides à un problème plus général d'allocation de la flotte, terme que l'on utilise ici dans un sens générique désignant toutes les ressources nécessaires au transport des conteneurs.

1.2 Les caractéristiques du problème

1.2.1 Le réseau physique et les modes de transport

Afin d'opérer, la compagnie exploite une partie des infrastructures de transport existantes, en l'occurrence :

- les terminaux portuaires, qui jouent le rôle de points d'entrée et de sortie des conteneurs par rapport au système de distribution terrestre mais remplissent aussi la fonctionnalité de surfaces d'entreposage de conteneurs vides ;
- les dépôts terrestres, qui s'associent aux ports pour permettre une plus grande proximité par rapport aux clients. En général, les dépôts terrestres sont également

des aires louées pour le stockage des conteneurs et se situent le plus souvent sur les principaux axes routiers et aux stations de chemin de fer.

Avec les ports et les dépôts terrestres, les clients de la compagnie forment les points physiques du réseau de distribution.

La seconde partie du réseau est constituée des liens reliant les clients, les ports et les dépôts. Dans le contexte du transport par voie terrestre, ces liens, illustrés par la figure 1.1, peuvent être des routes, des voies ferrées, des cours d'eau fluviaux navigables, ou des lignes de transport maritime régional.

Ces liens s'apparentent très étroitement à la notion de *mode* de transport, qu'il est possible de définir comme étant un moyen particulier de transport, possédant des caractéristiques propres. Un mode peut évidemment caractériser le transport par le biais d'une infrastructure unique, par exemple, le transport routier par camions, le transport ferroviaire, le transport fluvial ou le cabotage (navigation près des côtes, s'opérant localement entre un certain nombre de ports), mais permet aussi de décrire une combinaison de plusieurs de ces moyens (le transport de conteneurs par camions vers une station de chemin de fer, suivi du transfert aux wagons et de l'acheminement par rail aux clients, par exemple). Dans ce dernier cas, nous parlons de transport *multimodal*. C'est une notion d'une réelle importance qui apporte énormément de flexibilité aux opérations en permettant une meilleure utilisation de l'infrastructure de transport disponible, et qui, en tenant compte d'autres considérations telles que la sécurité accrue de transport et une plus grande facilité de stockage et de maintenance, peut expliquer le recours de plus en plus courant aux conteneurs comme support au transport terrestre de plusieurs types de biens.

1.2.2 Les régimes de transport

Le type d'infrastructure ne suffit pas à caractériser un mode, pris dans son sens large. En effet, les pratiques commerciales régissant les rapports entre les trois acteurs principaux (compagnie, clients et transporteurs), ayant des implications directes sur les coûts des opérations et la nature des mouvements, ont une importance significative.

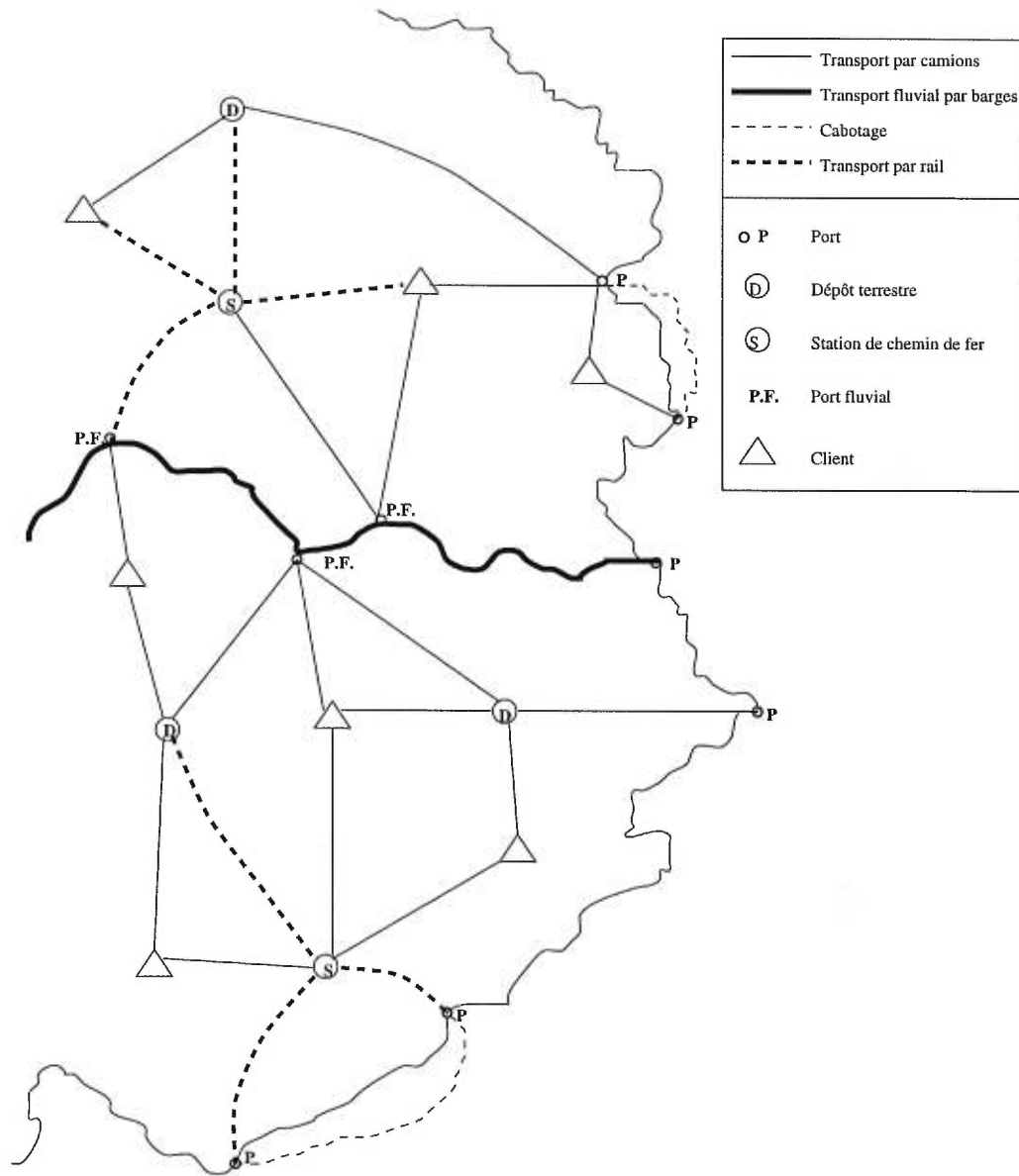


FIG. 1.1 – Configuration type d'un réseau physique de base

Nous préférons alors parler de *régimes de transport* pour qualifier ces rapports. Dans le cas d'une compagnie de transport maritime opérant en Europe, en l'occurrence la Compagnie Générale Maritime (C.G.M.), trois régimes sont en place :

- la compagnie prend en charge le coût de transport des conteneurs, qu'elle réalise soit par ses propres moyens, soit par l'intermédiaire d'un transporteur ;
- le coût intégral et l'organisation du transport incombe au client ;
- le régime mixte, où la compagnie prend uniquement en charge l'organisation et les coûts du transport des conteneurs vides.

1.2.3 L'homogénéité de la flotte

Nous parlons de flotte *homogène* ou *hétérogène* selon qu'il est question de gérer le transport d'un seul, ou de plusieurs types de conteneurs. Dans la plupart des cas, ce sont surtout la diversité des produits, les particularités de certaines lignes maritimes ou voies terrestres (notamment les conditions climatiques) et la réglementation liée à la sécurité du transport qui imposent le recours à une flotte hétérogène de conteneurs. La C.G.M, à titre d'exemple, utilise les combinaisons taille-fonctionnalité suivantes :

- conteneurs à usage général, de longueur 20 pieds (hauteurs variées et hauteur spéciale de 8'6") ou de longueur 40 pieds ;
- conteneurs à toit ouvert de 20 pieds ou de 40 pieds ;
- conteneurs à côté ouvert de 20 pieds ;
- conteneurs plats de 20 pieds ou de 40 pieds ;
- conteneurs isothermes normaux de 20 pieds ou de 40 pieds, ou spécifiques aux lignes maritimes des Antilles françaises ;
- conteneurs frigorifiques de 20 pieds ou de 40 pieds ;
- conteneurs citernes de 20 pieds ou de 40 pieds ;
- remorque ACL ;

- conteneurs pour le transport en vrac normaux ou spéciaux de longueur 20 pieds ;
- châssis nu pour le transport occasionnel par camions, sans conteneurs, de fret domestique.

1.2.4 Les mouvements de conteneurs

De façon générale, il est possible de faire la distinction entre deux types de mouvements : les mouvements commerciaux de conteneurs chargés et les mouvements techniques de conteneurs vides.

La première catégorie couvre les mouvements résultant d'une demande effective d'importation ou d'exportation de biens émanant des clients et générant un profit direct, ainsi que, le cas échéant, les mouvements d'acheminement et de retour de conteneurs vides correspondants. Plus explicitement, il s'agit des séquences de mouvements suivantes :

- série de mouvements à l'import, constituée d'un mouvement d'un conteneur plein d'un port ou d'un dépôt terrestre vers le client, suivi d'un mouvement de retour du conteneur, une fois déchargé, au dépôt d'origine ou à un autre dépôt de la compagnie ;
- série de mouvements à l'export, formée de façon très similaire d'un mouvement d'un conteneur vide d'un dépôt vers le client, qui charge le produit et initie un mouvement de conteneurs chargés vers un port d'embarquement ;
- mouvements intermédiaires de conteneurs vides et chargés entre les dépôts.

Ces mouvements peuvent faire partie d'opérations simples faisant intervenir uniquement deux agents par opération, ou d'opérations plus compliquées, spécialement dans le cas du camionnage, où il est possible d'avoir des :

- opérations à mouvement triangulaire caractérisé par le transport d'un seul conteneur par camion chez un premier client, où il est déchargé ; le conteneur vide est alors transporté sur le même camion vers un second client qui le charge et le retourne au dépôt d'origine ;

- opérations à chargements multiples où, sur un même châssis, sont transportés des conteneurs pleins destinés à différents clients. À chaque arrêt chez un client, un certain nombre de conteneurs peuvent être déchargés, les conteneurs vides résultants sont alors soit transportés sur le châssis pour le reste du voyage, soit utilisés par le client pour une exportation de produits sur le même châssis ou sur un autre camion ; d'autre part, le camion peut également recevoir des conteneurs vides ou chargés résultant d'autres mouvements.

Les mouvements techniques de conteneurs vides, quant à eux, permettent surtout de se procurer de nouveaux conteneurs, d'en louer ou d'acheminer des conteneurs avariés vers les aires de réparation. Cependant, ils peuvent aussi servir occasionnellement à rééquilibrer les différences qui tendent à apparaître au sein d'une région donnée, entre les quantités de conteneurs vides disponibles aux dépôts et la demande de conteneurs nécessaires aux opérations d'export, à l'aide de mouvements périodiques de conteneurs vides entre les dépôts excédentaires et ceux ayant un déficit. Ainsi, bien que ne générant pas directement de profit, ils permettent à la compagnie d'assurer le déroulement normal de ses activités : en partie grâce à eux (sont également envisagés, à la même fin, des mouvements maritimes d'import/export de conteneurs vides), la compagnie espère prévenir une pénurie de conteneurs, qui rendrait nécessaire le recours à des solutions plus coûteuses (location ou achat de conteneurs à des partenaires, par exemple), d'autant plus que ce trafic d'équilibrage, qui fait intervenir un grand nombre de conteneurs et est généralement effectué via le rail, autorise la compagnie à négocier des tarifs de transport plus avantageux que ceux des mouvements réguliers de conteneurs.

Le fait que ces mouvements à vide, tout en étant incontournables, ne génèrent pas de profit direct à la compagnie et occupent une bonne partie des ressources disponibles, leur confère une importance particulière. Par ailleurs, quantitativement, en termes de volume de trafic et de coût, ils ont un poids considérable (Dejax, Crainic et Delorme [26]) ; à défaut donc de pouvoir éliminer cette composante, il est primordial de bien la gérer.

1.3 Une approche méthodologique

Le problème du transport terrestre de la flotte est d'une grande complexité. Les difficultés qui y sont reliées ont surtout trait au nombre important de sous-problèmes difficiles qu'il fait intervenir et aux relations de dépendance qui existent entre ces derniers. En effet, le processus de planification semble obéir à une séquence hiérarchique de décision où les données d'un sous-problème (l'allocation de conteneurs vides par exemple) peuvent être le résultat d'un plan établi par un autre sous-problème tel la localisation de dépôts. De plus, le fait que ces sous-problèmes nécessitent des ressources différentes au niveau de la planification et de la gestion des opérations et se présentent différemment dans le temps exclut la possibilité de développer un modèle pour une résolution globale et simultanée de tous les problèmes.

Dans cette perspective, la compagnie est amenée à intégrer à son système d'information les moyens requis pour une identification des flux d'information entre les divers sous-problèmes, une cueillette systématique des données et un échange d'information entre les différentes composantes du problème.

Une approche de planification pour les problèmes de gestion de la flotte développée par Crainic, Dejax et Gendreau [18] identifie trois niveaux de décision :

- La planification stratégique, caractérisée à la fois par la nature des décisions prises (politique générale) et par un horizon de planification de longue durée (une année ou plus).
- La planification tactique, qui s'occupe principalement de concrétiser, à moyen terme, les décisions prises au niveau stratégique, tout en assurant de les mettre à jour en fonction des variations importantes des données du système, par exemple suite à des changements saisonniers.
- Le niveau opérationnel, où, disposant des décisions prises dans les niveaux supérieurs, on s'intéresse à la réalisation des opérations. Vu qu'elles couvrent des horizons de planification plus courts de l'ordre d'une à plusieurs semaines, les décisions prises à ce niveau ont un degré de détail plus élevé et tiennent

compte d'avantage de la dynamique du système.

La figure 1.2 illustre cette approche de décision dans le cadre particulier de la gestion de la flotte. Le problème le plus important résolu au niveau stratégique est la localisation des dépôts avec affectation des clients aux dépôts. Pour ce problème, Crainic, Dejax et Delorme [17] proposent un modèle multiproduits, qui tient compte du trafic d'équilibrage entre les dépôts et pour la résolution duquel une méthode d'augmentation duale (Crainic et Delorme [19]) et un algorithme de séparation et évaluation progressive (Gendron et Crainic [31]) ont notamment été proposés. Le problème de localisation des dépôts, avec la définition d'une politique de tarification à moyen et long terme, forment l'essentiel du *plan tactique*, qui fournit les données nécessaires au problème de l'allocation dynamique des conteneurs vides, intervenant au niveau opérationnel afin d'établir une liste des ordres de mouvement de ces derniers. Sur la base de ces ordres et du plan tactique, les modèles de routage et d'ordonnancement établissent enfin les itinéraires détaillés des conteneurs chargés et vides, l'affectation des véhicules et les horaires des voyages.

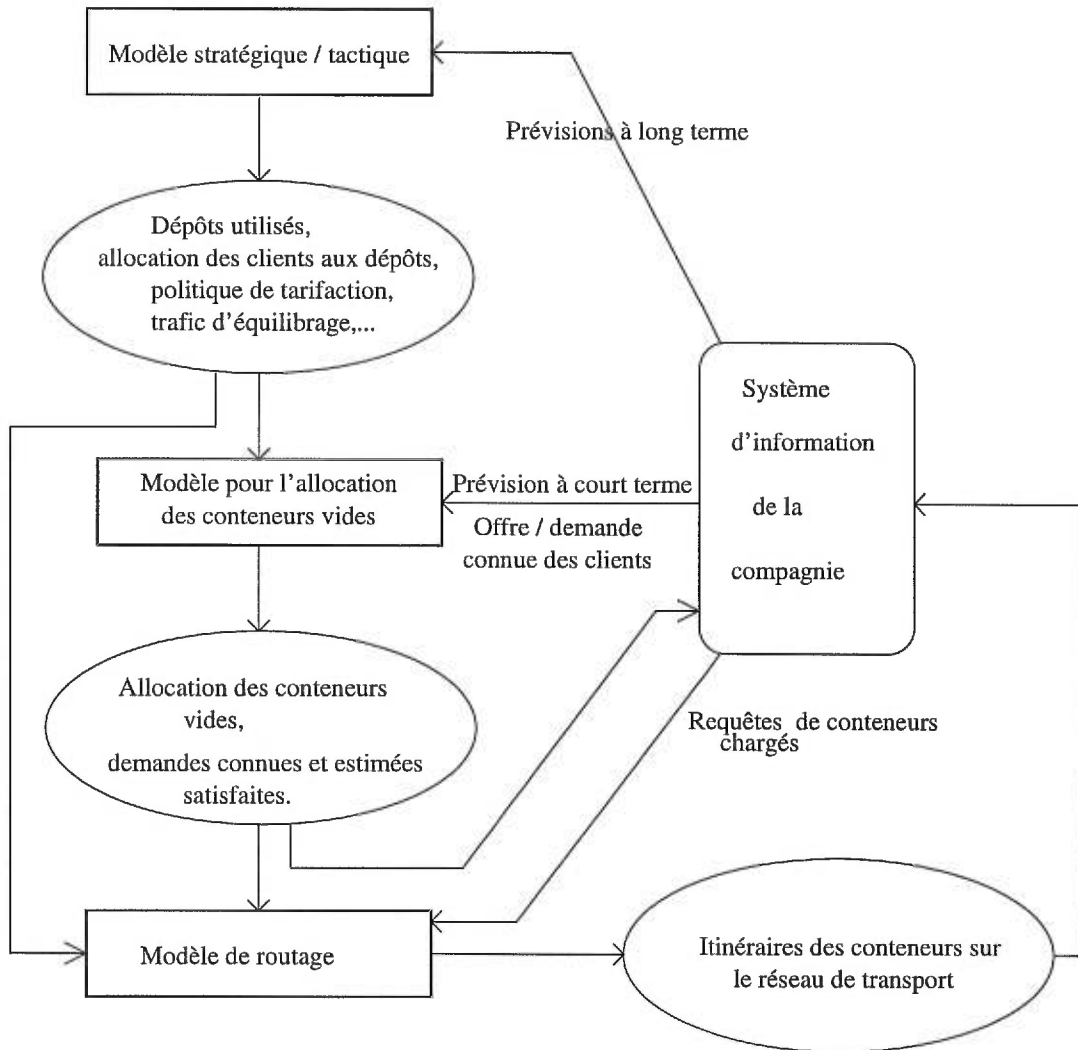


FIG. 1.2 – Représentation d'un processus de décision

1.4 La littérature sur l'allocation dynamique des véhicules

L'allocation des conteneurs vides entre dans le cadre plus général de l'allocation dynamique des véhicules. Le but de cette section est de passer en revue, de façon très sommaire, les principaux efforts orientés vers la modélisation et la résolution de cette catégorie de problèmes, tout en référant le lecteur intéressé à plus de détails aux revues des modèles de planification et de gestion de la flotte pour le transport terrestre de marchandises de Dejax et Crainic [25] et de Crainic et Laporte [21], ainsi qu'à celle de Powell, Jaillet et Odoni [57], consacrée aux modèles et algorithmes déterministes

et stochastiques pour les réseaux de transport dynamiques.

Les tous premiers travaux se sont intéressés presque exclusivement à l'allocation des wagons vides dans le domaine du transport ferroviaire. Dans ce cadre, les modèles de Leddon et Wrathall [45] et Misra [46] sont des formulations déterministes et statiques du problème sous forme de programme linéaire, résolu par la méthode du simplexe ou par un algorithme de type arcs non conformes (“Out of Kilter”). Pour leur part, White et Bomberault [63] et Ouimet [50] introduisent pour la première fois la notion de réseau espace-temps et développent des approches de résolution exploitant la structure multipériodes du problème. Les travaux de Jordan et Turnquist [41], quant à eux, sont un autre pas en avant dans la mesure où la formulation stochastique qu'ils proposent intègre, en plus du caractère dynamique, l'incertitude dans l'offre et la demande de wagons. Enfin, le modèle proposé par Haghani [37] se distingue des travaux précédents en intégrant la gestion du flux de retour des wagons vides à un problème de routage des trains entre les stations d'un réseau de chemin de fer.

Les travaux de Powell font référence dans le domaine du transport terrestre de marchandises par camions. La première contribution significative à ce champ est un modèle dynamique et stochastique pour l'allocation de véhicules (Powell, Sheffi et Thiriez [58]), où l'objectif est de maximiser le profit espéré, en faisant face à des demandes prévues incertaines de camions. Ce modèle de base a par la suite été étendu (Powell [52, 53]) pour permettre l'entreposage des camions comme alternative à l'envoi de vides et pour tenir compte des mouvements de camions chargés. On doit noter également les récentes contributions de Frantzeskakis et Powell [30] et de Cheung et Powell [14], qui proposent des approches heuristiques pour l'approximation linéaire et convexe de la fonction du profit espéré du modèle d'allocation dynamique et stochastique de Powell, en tenant compte de l'incertitude sur la capacité des arcs.

Très récemment, Powell, Carvalho, Godfrey et Simaō [56] ont développé une nouvelle méthodologie connue sous le nom de réseaux logistiques à files d'attente (“Logistics Queueing Networks”), dont l'application à l'allocation de véhicules à grande échelle (Powell et Carvalho [55]), qui fait intervenir une décomposition spatio-

temporelle du problème, semble fournir des résultats très prometteurs.

Par rapport à ces travaux spécifiques au rail et au camionnage, le transport multimodal de conteneurs reste un sujet relativement peu traité. À notre connaissance, les formulations déterministe et stochastique de Crainic, Gendreau et Dejax [20] constituent la contribution la plus importante. Les auteurs proposent ainsi un premier modèle déterministe pour l'allocation multiproduits qui intègre la possibilité de substitution entre les produits et suggèrent les méthodes de décomposition comme moyen d'exploiter la structure particulière du modèle. Cette composante du modèle est d'une importance particulière car elle est à la base des approches de résolution que nous proposons. Ils traitent ensuite, en se limitant au cas d'un seul produit, l'incertitude dans les valeurs de l'offre et de la demande en développant un modèle stochastique basé sur un *programme stochastique à deux étapes avec recours complet*, qui comporte une première étape où un programme déterministe fournit les décisions relatives à la période courante, ainsi que les autres décisions associées aux offres et aux demandes certaines, et une seconde phase qui suit les fluctuations des niveaux d'inventaire des dépôts en examinant les différents scénarios stochastiques.

Il paraît opportun de citer, bien que n'ayant pas de relation directe avec l'allocation, deux travaux reliés aux données nécessaires à la planification opérationnelle. Le premier, qui concerne l'estimation de l'offre et de la demande de conteneurs, a été réalisé par Steffan [61] et vise, à travers l'observation des mouvements vides et chargés de conteneurs d'une compagnie maritime, à déterminer des fonctions de distribution associées. Le second (Vasseur [62]) est une étude similaire pour les coûts.

Chapitre 2

Un modèle d'allocation déterministe de conteneurs

Dans ce chapitre, nous examinons la composante déterministe du modèle de Craïnic, Gendreau et Dejax [20] pour l'allocation multiproduits de conteneurs vides. Après avoir identifié les éléments de décision du problème d'allocation, nous présentons le modèle mathématique et une représentation réseau associée.

2.1 Les éléments du problème

Sur un horizon de planification, subdivisé en n périodes de temps, les modèles stratégiques et tactiques spécifient une configuration du réseau de distribution, constituée de l'ensemble des dépôts actifs, des associations entre les clients et les dépôts et des mouvements d'équilibrage inter-dépôts. Sont également précisées les demandes connues et estimées des clients en conteneurs vides de différents types, ainsi que la disponibilité, après le déchargement des marchandises, de conteneurs vides qui doivent être redirigés vers les dépôts. Sur la base de ces données, le problème de l'allocation de conteneurs vides se résume principalement à produire une liste d'ordres de mouvement de conteneurs vides permettant la satisfaction des requêtes de demande et le retour des conteneurs vides.

En plus de la possibilité d'avoir des mouvements d'équilibrage inter-régionaux, deux autres caractéristiques contribuent à rendre le problème plus souple :

- l'ouverture du réseau de distribution terrestre vers l'extérieur, via les ports, qui jouent le rôle de points d'entrée/sortie à travers l'importation et l'exportation de conteneurs, mais aussi grâce à la possibilité d'acheter, de louer et d'échanger des conteneurs à l'extérieur, notamment à des partenaires commerciaux ;
- la substitution d'un type de conteneurs à un autre, qui s'effectue en s'appuyant sur un ensemble de règles précises définies par les décideurs.

Ainsi, en plus des ordres de mouvement associés aux requêtes, des décisions annexes sont à prendre, et concernent :

- les niveaux de stock aux dépôts ;
- les nombres et les types de conteneurs à acheter, à louer ou à échanger aux partenaires ;
- les substitutions entre les différents types de conteneurs.

Notons l'importance de la dimension temporelle du problème. En effet, en ce qui concerne la clientèle, la distinction doit être clairement faite entre un client physique, et un client considéré comme "point de demande et d'offre" et donc caractérisé

également par les requêtes qu'il effectue, et qui peuvent impliquer, pour des périodes de temps différentes, des nombres de conteneurs distincts, d'un même type, ou de types différents. Ce caractère dynamique conditionne aussi, de la même façon, tous les autres aspects du problème (changements temporels des niveaux d'inventaire aux dépôts, politiques tarifaires et règles de substitution changeantes, etc.).

2.2 Le modèle mathématique

2.2.1 Notations et définitions

Soit

T : la longueur (nombre de périodes) de l'horizon de planification ;

P : l'ensemble des produits (types de conteneurs) ;

H : l'ensemble des ports ;

D : l'ensemble des dépôts terrestres ;

τ_{ij} : le temps de transit entre une source i et une destination j , i et j pouvant être un port, un dépôt terrestre ou un client ;

I^{pt} : l'ensemble des clients effectuant des requêtes de demande en conteneurs vides du type p , pour la période t , $p \in P$, $t = 1, 2, \dots, T$;

Δ_i : la fenêtre temporelle pour une livraison de conteneurs correspondant au client $i \in I^{pt}$, qui indique que la livraison devra se faire durant une des périodes $t - \Delta_i, t - \Delta_i + 1, \dots, t$;

S^{pt} : l'ensemble des clients ayant un certain nombre de conteneurs vides du type p , durant la période t , $p \in P$, $t = 1, 2, \dots, T$, qui doivent être repositionnés aux dépôts ;

X_i^{pt} : la demande en conteneurs vides de type p correspondant au client $i \in I^{pt}$;

Y_s^{pt} : l'offre en conteneurs vides de type p associée au client $s \in S^{pt}$;

X_j^{pt} : la demande d'export en conteneurs vides de type p , du port j , durant la période t , $t = 1, 2, \dots, T$;

Y_j^{pt} : le volume importé de conteneurs vides de type p , devenu disponible au port j durant la période t , $t = 1, 2, \dots, T$;

Nous définissons également les ensembles suivants représentant les associations entre les clients et les dépôts :

I_j^{pt} : l'ensemble des requêtes de demande, concernant le produit p , pouvant être satisfaites par un mouvement de conteneurs partant du dépôt j durant la période t et arrivant à destination durant $t + \tau_{ji}$, $j \in D \cup H$, $t = 1, 2, \dots, T$;

S_j^{pt} : l'ensemble des requêtes d'offre, concernant le produit p , pouvant être satisfaites par un mouvement de conteneurs arrivant au dépôt j durant la période t , $j \in D \cup H$, $t = 1, 2, \dots, T$;

$J_i^{pt'}$: l'ensemble des dépôts $j \in I^{pt}$ pouvant être utilisés pour satisfaire une requête de demande i , grâce à un mouvement de conteneurs de type p initié durant la période t' et arrivant à destination à $t' + \tau_{ji}$, $t' = 1, 2, \dots, T$, de façon à avoir $t - \Delta_i \leq t' + \tau_{ji} \leq t$;

J_s^{pt} : l'ensemble des dépôts j pouvant être utilisés pour collecter l'offre en conteneurs vides d'un client $s \in S^{pt}$, $t = 1, \dots, T$.

Les variables de décision du modèle concernent les volumes de conteneurs associés à l'allocation, aux mouvements d'équilibrage, aux ordres de substitution, aux niveaux de stock dans les dépôts, aux mouvements de nouveaux conteneurs introduits dans le système et à la demande d'export insatisfaite. Cependant, avant de les énumérer, notons que la propriété de substitution implique que l'allocation de conteneurs à une requête de demande réalisée par un client pour un type de conteneurs p donné, peut se faire en conteneurs du même type, comme elle peut être réalisée par l'envoi d'un certain nombre de conteneurs *équivalents*, d'un autre type r pouvant se substituer à p . Afin de contourner cette difficulté, le modèle propose de séparer les deux fonctionnalités d'un dépôt, à savoir le stockage des conteneurs, et leur allocation à des requêtes de demande, ce qui permet de différencier l'allocation de conteneurs de type "physique" p , de l'allocation de substituts d'autres types.

Soit donc :

v_j^{pt} : le volume de conteneurs de type p devant être alloués durant la période t par le dépôt j , $j \in D \cup H$;

v_{ji}^{pt} : le volume de conteneurs de type p , ou d'autres types substitués à p , alloués

durant la période t par le dépôt j au client i , $j \in D \cup H$, $i \in I_j^{pt}$;

v_{sj}^{pt} : le volume de conteneurs de type p retournés à la période t au dépôt j par le client s , $j \in D \cup H$, $s \in S_j^{pt}$;

w_{jj}^{pt} : le volume de conteneurs vides du type p en inventaire au dépôt j , à la fin de la période t , $j \in D \cup H$;

v_{jk}^{pt} : le volume d'un mouvement d'équilibrage en conteneurs vides de type p initié à la période t du dépôt j vers le dépôt k , $j \in D \cup H$, $k \in D \cup H$;

w_j^{prt} : le volume de conteneurs de type p substitués à des conteneurs de type r au dépôt j durant la période t , $j \in D \cup H$;

b_j^{pt} : le volume de nouveaux conteneurs de type p introduits dans le système par le dépôt j durant la période t , $j \in D \cup H$;

e_j^{pt} : la demande d'export insatisfaite en conteneurs de type p par le port j durant la période t , $j \in H$.

En nous basant sur des hypothèses de linéarité des fonctions de coût associées aux variables de décisions et d'indépendance de chacune de ces fonctions par rapport au reste du système (Crainic, Gendreau et Dejax [20]), nous définissons les coûts unitaires suivants :

c_{ji}^{pt} : le coût unitaire de transport de conteneurs de type p , durant la période t , du dépôt j vers le client i , $j \in D \cup H$, $i \in I_j^{pt}$, $t' = t + \tau_{ji}$, $t = 1, 2, \dots, T$;

c_{sj}^{pt} : le coût unitaire du transport de conteneurs de type p , durant la période t , du client s vers le dépôt j , $j \in D \cup H$, $s \in S_j^{pt}$, $t = 1, 2, \dots, T$;

c_{jk}^{pt} : le coût unitaire associé au transport de conteneurs de type p , durant la période t , du dépôt j vers le dépôt k , $j \in D \cup H$, $k \in D \cup H$, $t = 1, 2, \dots, T$;

c_j^{pt} : le coût unitaire associé au maintien en stock de conteneurs du type p au dépôt j , durant la période t , $j \in D \cup H$, $t = 1, 2, \dots, T$;

c_j^{prt} : le coût unitaire de la substitution de conteneurs de type p à des conteneurs de type r , au dépôt j durant la période t , $j \in D \cup H$, $t = 1, 2, \dots, T$;

\bar{c}_j^{pt} : le coût unitaire de l'introduction dans le système, par le dépôt j et durant la

période t , de nouveaux conteneurs de type p , $j \in D \cup H, t = 1, 2, \dots, T$;

\tilde{c}_j^{pt} : le coût unitaire pour faire face à une demande insatisfaite d'export de conteneurs de type p au port j durant la période t , $j \in H, t = 1, 2, \dots, T$;

2.2.2 Les contraintes du modèle

Le modèle tient compte des catégories de contraintes suivantes :

- La demande des clients :

La demande d'un client $i \in I^{pt}, t = 1, 2, \dots, T$ peut être satisfaite par tous les dépôts $j \in J_i^{pt'}, t' \leq t$, reliés à i de façon à ce qu'un mouvement de conteneurs initié par j durant la période t' puisse arriver à l'intérieur de l'intervalle temporel $[t - \Delta_i, t]$ de livraison du client i .

Si on note K_i^{pt} le volume de conteneurs envoyés au client $i \in I^{pt}, t = 1, 2, \dots, T$ avant le début de l'horizon de planification, la satisfaction de la demande du client i se traduit alors par :

$$\sum_{t' \leq t} \sum_{j \in J_i^{pt'}} v_{ji}^{pt'} = X_i^{pt} - K_i^{pt}; \quad \forall p \in P, \forall i \in I^{pt}, t = 1, 2, \dots, T. \quad (1)$$

- L'offre des clients :

Les dépôts de l'ensemble J_s^{pt} peuvent tous être sollicités pour recueillir le volume de conteneurs vides devant être retournés par le client $s \in S^{pt}$:

$$\sum_{j \in J_s^{pt}} v_{sj}^{pt} = Y_s^{pt}; \quad \forall p \in P, \forall s \in S^{pt}, t = 1, 2, \dots, T. \quad (2)$$

- Les stocks aux dépôts non portuaires :

Comme les clients, les dépôts peuvent aussi recevoir des conteneurs vides envoyés avant le début de l'horizon de planification. Nous noterons ce volume de conteneurs K_j^{pt} . Désignons également, conformément à la notation utilisée, le stock initial du

produit p , à un dépôt $j \in D \cup H$, disponible au début de l'horizon de planification, par w_{jj}^{p0} .

À la fin de la période t , le niveau d'inventaire est la différence des volumes de conteneurs vides reçus (inventaire à la fin de la période précédente, volumes de conteneurs reçus des clients, mouvements d'équilibrage reçus des autres dépôts, nouveaux conteneurs et volume reçu avant le début de l'horizon) et des volumes de conteneurs vides alloués, qui sont physiquement utilisés comme conteneurs de type p ou qui se substituent à d'autres types.

Ceci est exprimé par :

$$\begin{aligned}
w_{jj}^{pt} &= w_{jj}^{p(t-1)} + \sum_{t' \leq t} \left(\sum_{k \in D \cup H | \tau_{kj} = t-t'} w_{kj}^{pt'} + \sum_{s \in S_j^{pt'}} v_{sj}^{pt'} \right) \\
&\quad + b_j^{pt} + K_j^{pt} - \sum_{k \in D \cup H} w_{jk}^{pt} - v_j^{pt} - \sum_{r \in P} v_j^{prt}; \\
&\quad \forall j \in D, \forall p \in P, t = 1, 2, \dots, T. \quad (3)
\end{aligned}$$

- Les stocks aux dépôts portuaires :

La contrainte correspondante est très semblable à celle des dépôts terrestres, à la seule différence qu'elle tient compte aussi de la demande d'export X_j^{pt} et du volume de conteneurs vides importés Y_j^{pt} , ainsi que de la possibilité de ne pas satisfaire une partie de la demande d'export :

$$\begin{aligned}
w_{jj}^{pt} &= w_{jj}^{p(t-1)} + \sum_{t' \leq t} \left(\sum_{k \in D \cup H | \tau_{kj} = t-t'} w_{kj}^{pt'} + \sum_{s \in S_j^{pt'}} v_{sj}^{pt'} \right) \\
&\quad + b_j^{pt} + K_j^{pt} - \sum_{k \in D \cup H} w_{jk}^{pt} - v_j^{pt} - \sum_{r \in P} v_j^{prt} \\
&\quad - X_j^{pt} + Y_j^{pt} + e_j^{pt}; \\
&\quad \forall j \in H, \forall p \in P, t = 1, 2, \dots, T. \quad (4)
\end{aligned}$$

- La disponibilité de conteneurs pour l'allocation :

Cette contrainte traduit tout simplement le fait que le volume de conteneurs vides de type p alloués par le dépôt j peut être composé de conteneurs qui sont physiquement de type p et de conteneurs d'autres types substitués à p .

Ainsi, si a_{rp} désigne le facteur de substitution de conteneurs de type r à des conteneurs de type p , nous avons :

$$v_j^{pt} + \sum_{r \in P} a_{rp} w_j^{rpt} - \sum_{i \in I_j^{pt}} v_{ji}^{pt} = 0 ;$$

$$\forall j \in D \cup H, \forall p \in P, t = 1, 2, \dots, T. \quad (5)$$

- Les flots d'équilibrage :

Des bornes inférieures l_{jk}^{pt} et supérieures de capacité u_{jk}^{pt} , résultant d'accords avec les transporteurs, règlent les volumes des flots d'équilibrage inter-dépôts.

$$l_{jk}^{pt} \leq w_{jk}^{pt} \leq u_{jk}^{pt} ; \quad \forall j \in D \cup H, \forall k \in D \cup H, t = 1, 2, \dots, T. \quad (6)$$

- Les capacités des arcs de substitution :

Des bornes supérieures B_j^{prt} existent sur le volume de conteneurs de type p substitués au dépôt j à des conteneurs de type r , durant la période t :

$$w_j^{prt} \leq B_j^{prt} ; \quad \forall j \in D \cup H, \forall (p, r) \in P^2, t = 1, 2, \dots, T. \quad (7)$$

2.2.3 La fonction économique

Le modèle déterministe, dynamique et multiproduits pour l'allocation de conteneurs vides consiste donc à minimiser le coût total des opérations :

$$\begin{aligned}
\sum_{t=1, \dots, T} \sum_{p \in P} [& \sum_{j \in DUH} (\sum_{i \in I_j^{pt}} c_{ji}^{pt} v_{ji}^{pt} + \sum_{k \in DUH} c_{jk}^{pt} v_{jk}^{pt} \\
& + c_j^{pt} w_{jj}^{pt} + \tilde{c}_j^{pt} b_j^{pt} + \sum_{r \in P} c_j^{prt} w_{jj}^{prt}) \\
& + \sum_{j \in H} \tilde{c}_j^{pt} e_j^{pt} + \sum_{s \in S^{pt}} \sum_{j \in J_s^{pt}} c_{sj}^{pt} v_{sj}^{pt}] \quad (8)
\end{aligned}$$

en étant sujet aux contraintes (1) à (7), ainsi qu'aux contraintes de non-négativité des valeurs de flots.

2.3 La formulation réseau

Dans cette section, nous présentons une formulation réseau équivalente au modèle mathématique proposé plus haut. Cette formulation permet de représenter le problème de l'allocation sous la forme d'un problème de flot à coût minimum dans un réseau G à plusieurs sources et plusieurs puits.

La démarche poursuivie pour l'obtention de la formulation réseau est très classique et exploite l'hypothèse que la configuration du réseau ne change pas durant une période donnée de l'horizon de planification, pour un type donné de conteneurs. Le réseau $G = (N, A)$ correspondant est ainsi une structure tridimensionnelle suivant trois axes :

- l'axe x du temps, subdivisé en unités représentant les périodes temporelles de l'horizon de planification ;
- l'axe y désignant l'espace physique ;
- l'axe z des produits, ou types de conteneurs.

L'ensemble N des nœuds du réseau espace-temps G correspond donc à la duplication, pour chaque période de temps et chaque type de conteneurs, de la configuration physique du réseau. Par exemple, un client i ayant, durant la période t , une demande

de conteneurs vides du type p , sera représenté dans N par un nœud i^{pt} .

De façon plus formelle, $N = \{i^{pt}, s^{pt}, j^{pt}, j'^{pt}, h^{pt}, h'^{pt}, \alpha, \alpha'\}$ où :

- i^{pt} et s^{pt} sont les nœuds associés respectivement à une demande de conteneurs vides de type p durant la période t , par le client i , et à une disponibilité, chez le client s , de conteneurs de type p à récupérer durant la période t ;
- h^{pt} et j^{pt} sont les nœuds résultant de la duplication, respectivement, du port physique h et du dépôt terrestre physique j , et qui correspondent à la fonctionnalité “stockage de conteneurs vides” des dépôts;
- h'^{pt} et j'^{pt} sont les nœuds représentant la source, respectivement au port h et au dépôt terrestre j , du volume alloué de conteneurs vides de type p durant la période t . Ces nœuds correspondent, quant à eux, à la composante “allocation de conteneurs vides” des dépôts;
- α représente l'origine des nouveaux conteneurs introduits dans le système;
- α' est un nœud fictif qui récupère les demandes d'export non satisfaites par les ports.

L'ensemble d'arcs A se subdivise alors en plusieurs catégories, supportant chacune un type de mouvement physique ou logique de conteneurs :

- les mouvements de nouveaux conteneurs, représentés par les arcs (α, j^{pt}) et (α, h^{pt}) ;
- les mouvements de compensation de la demande d'export non satisfaite (α', h^{pt}) ;
- les mouvements d'allocation $(j'^{pt}, i^{p(t+\tau_{ji})})$ ou $(h'^{pt}, i^{p(t+\tau_{hi})})$ et de récupération $(s^{pt}, j^{p(t+\tau_{sj})})$ ou $(s^{pt}, h^{p(t+\tau_{sh})})$ de conteneurs. Les arcs correspondants traversent le réseau espace-temps par des mouvements s'étalant généralement sur plusieurs périodes pour relier des points géographiquement distincts;
- l'inventaire des dépôts à la fin d'une période t , modélisé par un flot $(j^{pt}, j^{p(t+1)})$ et $(h^{pt}, h^{p(t+1)})$ de transfert temporel local aux dépôts;

- le volume de conteneurs d'un type p destinés à l'allocation par un dépôt, (j^{pt}, j'^{pt}) ou (h^{pt}, h'^{pt}) , modélisé sous la forme d'un flux reliant la composante "stockage" à la composante "allocation" des dépôts;
- le volume de conteneurs d'un type r substitués à des conteneurs de type p , représenté par les liens (j^{rt}, j'^{pt}) ou (h^{rt}, h'^{pt}) ;
- les mouvements d'équilibrage inter-dépôts $(j^{pt}, k^{p(t+\tau_{jk})})$.

Cette structure est illustrée par la figure 2.1:

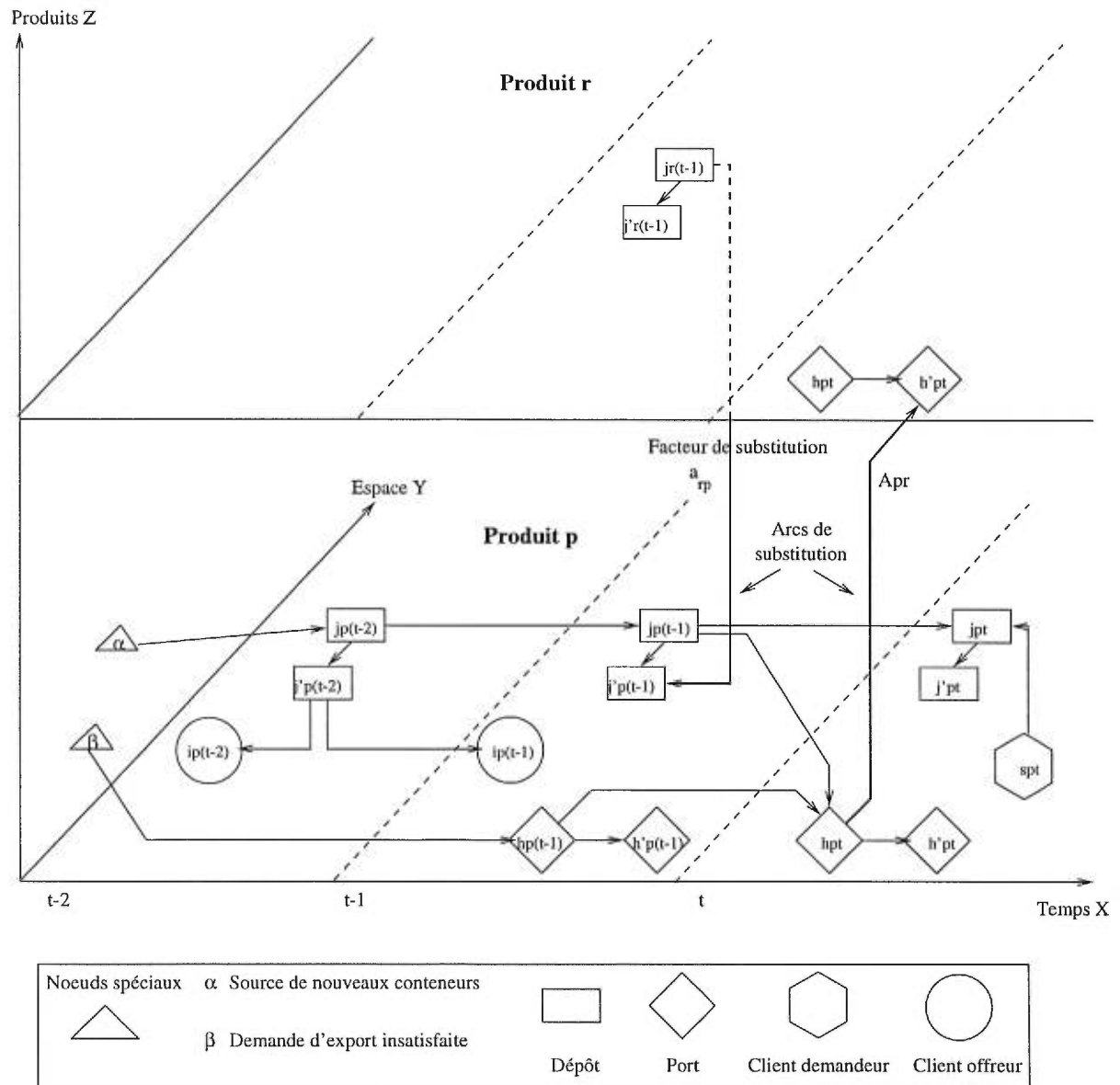


FIG. 2.1 – Illustration de la formulation réseau du problème

L'intérêt d'une telle formulation, outre la simplicité obtenue grâce à une représentation graphique où, par exemple, les contraintes de conservation de flot deviennent sous-jacentes, réside tout d'abord dans le fait qu'elle nous permet, au lieu de considérer le problème comme un programme linéaire général, tel que le suggère a priori le modèle mathématique, de prendre en considération la structure de réseau qui lui est sous-jacente et d'envisager plutôt l'utilisation d'algorithmes plus efficaces d'optimisation dans les réseaux. L'autre avantage de cette formulation est qu'à travers une représentation graphique du problème, les caractéristiques structurelles du réseau, dont nous allons tirer profit dans l'élaboration de nos stratégies de résolution, sont davantage mises en valeur.

Chapitre 3

Méthodes de résolution

Nous entamons ce chapitre par l'examen des algorithmes de flot à coût minimum dans les réseaux avec gains, susceptibles de fournir un moyen général pour résoudre le problème de l'allocation de conteneurs vides, tout en mettant un accent particulier, pour les raisons que nous verrons plus loin, sur l'adaptation du simplexe primal aux réseaux. Des stratégies de résolution plus élaborées, qui tendent à exploiter la structure dynamique et multiproduits du problème, sont par la suite proposées.

3.1 Résolution du flot à coût minimum dans les réseaux avec gains

3.1.1 Perspective historique

Les premiers efforts d'adaptation de la méthode du simplexe en programmation linéaire aux réseaux avec gains sont dûs à Dantzig [24]. L'intérêt de ces travaux était surtout d'identifier la structure particulière des bases et de proposer une alternative intéressante aux méthodes primales-duales (l'algorithme primal-dual de Jewell [40], par exemple) qui étaient d'une grande popularité à l'époque. Dans ce contexte de travaux précurseurs, une approche duale d'augmentation de flot (Jensen et Bhau-mik [39]) a également été proposée.

Plusieurs implantations du simplexe réseau ont mis par la suite l'accent sur l'importance des structures de données utilisées notamment pour la représentation des bases. Parmi ces travaux, nous citons en particulier ceux de Glover, Klingman et Stutz [32], Barr, Glover et Klingman [6] et Brown et McBride [12], ainsi que les récents efforts de parallélisation de Peters [51] et Barr et Hickman [5].

Par ailleurs, Bertsekas et Tseng [9] ont mis au point la méthode de relaxation pour les problèmes de flot à coût minimum dans les réseaux purs et avec multiplicateurs, et dont les résultats numériques d'implantation ont montré une certaine supériorité pour quelques classes de problèmes, par rapport aux implantations basées sur le simplexe primal.

À l'heure actuelle, la tendance est surtout à l'adaptation des méthodes de points intérieurs à la structure de réseau (Resende et Veiga [59]) et aux nouveaux algorithmes polynomiaux développés dans le cadre particulier du problème de flot maximum dans les réseaux généralisés (Goldberg, Plotkin et Tardos [33], à titre d'exemple), et dont les principes directeurs semblent être prometteurs pour le problème général.

Notre choix de mettre l'accent sur l'adaptation du simplexe primal aux réseaux avec gains n'est pas fortuit. En effet, des études récentes (notamment Lebeuf [44])

montrent que le simplexe, au niveau des performances, peut se comparer très avantageusement aux meilleurs algorithmes disponibles, notamment aux implantations de la méthode de relaxation, pourvu qu'un choix judicieux des structures de données de l'implantation et un emploi des heuristiques de choix de pivots les plus adaptées soient effectués. De plus, l'énorme quantité de travaux théoriques consacrés au simplexe, ainsi qu'un grand nombre d'implantations robustes, nous fournissent une assise et un point de référence très appréciables, en vue éventuellement d'une analyse comparative avec notre propre implantation.

3.1.2 Définition et caractérisation du problème

Le problème de réseau généralisé consiste à trouver un vecteur x^* à n composantes tel que :

$$cx^* = \min\{cx : x \in X\} \quad \text{où } X = \{\mathcal{A}x = b, 0 \leq x \leq u\} \quad (3.1)$$

et où u est un vecteur à n composantes toutes positives et \mathcal{A} est une matrice $m \times n$ de plein rang et possédant la propriété que chaque colonne de \mathcal{A} est composée d'au plus deux éléments non nuls .

Notons que ce problème général inclut, comme cas particuliers importants, le problème de flot à coût minimum dans les réseaux standard, ainsi que le problème de flot à coût minimum dans les réseaux avec gains.

D'autre part, soit $G = [N, A]$ un réseau orienté à m nœuds et n arcs. Nous associons à chaque arc $a \in A$:

- un flot x_a ,
- un coût unitaire c_a ,
- une capacité maximale de flot u_a ,
- un multiplicateur α_a^+ associé à la source de l'arc a ,
- un multiplicateur α_a^- associé au puits de l'arc a .

À chaque nœud i correspond une valeur d'offre/demande b_i et la contrainte de conservation avec transformation de flot suivante :

$$\sum_{a \in A_i^+} \alpha_a^+ x_a + \sum_{a \in A_i^-} \alpha_a^- x_a = b_i \quad (3.2)$$

où A_i^+ désigne l'ensemble d'arcs émanant du nœud i et A_i^- l'ensemble d'arcs entrant dans i .

Nous établissons le lien avec la forme analytique (3.1) en formant la matrice \mathcal{A} , dont les lignes correspondent aux nœuds de N et les colonnes aux arcs de A , de la façon suivante :

$$\mathcal{A}_{ij} = \begin{cases} \alpha_a^+ & \text{si } i \in I_a^+, \\ \alpha_a^- & \text{si } i \in I_a^-, \\ 0 & \text{sinon;} \end{cases}$$

où I_a^+ (resp. I_a^-) est l'indice de la ligne correspondant à la source (resp. au puits) de l'arc a .

Avec cette formulation, le problème (3.1) correspond à la recherche, dans G , d'un flot à coût minimum satisfaisant les contraintes (3.2) de conservation de flot aux nœuds et celles associées aux capacités des arcs.

Un cas particulier intéressant du problème général est le problème de réseau généralisé *avec gains positifs*, où α_a^+ et α_a^- sont de signes opposés. Dans ce cas, et sans perte de généralité, nous pouvons supposer que $\alpha_a^+ = +1$ et $\alpha_a^- = -\theta_a$ où θ_a est le *multiplicateur* positif de l'arc a et obtenir le *problème de flot à coût minimum dans un réseau avec gains* suivant :

$$\min \sum_{a \in A} c_a x_a$$

sujet à :

$$\sum_{a \in A_i^+} x_a - \sum_{a \in A_i^-} \theta_a x_a = b(i); \quad \forall i \in N \quad (3.3)$$

$$0 \leq x_a \leq u_a; \quad \forall a \in A$$

3.1.3 Caractérisation des bases

Soit \mathcal{B} une base du problème de réseau généralisé (3.1), extraite de \mathcal{A} . Un résultat bien connu permet de dégager la structure particulière de \mathcal{B} :

Théorème 1 (Dantzig[24]) *En réarrangeant les lignes et les colonnes de \mathcal{B} , il est possible d'obtenir une matrice $\tilde{\mathcal{B}}$, diagonale par blocs, ayant la forme suivante :*

$$\tilde{\mathcal{B}} = \begin{pmatrix} \tilde{\mathcal{B}}_1 & & & \\ & \tilde{\mathcal{B}}_2 & & \\ & & \ddots & \\ & & & \tilde{\mathcal{B}}_l \end{pmatrix}$$

où $\{\tilde{\mathcal{B}}_i\}_{1 \leq i \leq l}$ sont des matrices carrées possédant chacune la propriété d'être ou bien triangulaire supérieure ou "presque" triangulaire supérieure, avec un seul élément non nul en dessous de la diagonale.

Afin de donner à ce résultat une signification plus graphique, nous rappelons la notion d'arbre augmenté :

Définition 1 *Un arbre augmenté est un graphe connexe contenant exactement un cycle.*

Notons le cas particulier où le cycle est réduit à une boucle, qui correspond à une variable artificielle.

Le théorème précédent peut donc être énoncé sous la forme équivalente suivante : une base du problème de réseau généralisé est un ensemble d'arbres augmentés disjoints.

Cette propriété que possèdent les bases du problème de réseau généralisé s'avère fondamentale et permet, comme nous le verrons plus loin, d'adapter le simplexe primal de façon très efficace, en exploitant notamment la facilité de mettre à jour, à chaque itération de l'algorithme :

- les variables primales (les valeurs de flot), par un processus similaire à la résolution par substitution des systèmes linéaires triangulaires ;

- les variables duales par substitution inverse ;
- la structure de la base suite à un pivot.

3.1.4 Les structures de données pour la représentation de la base

Les efforts d’implantation du simplexe primal pour les réseaux purs et les réseaux avec gains, au cours des années 70, ont permis de se rendre compte que le choix des structures de données les plus appropriées pour la représentation de la base est un facteur important de l’efficacité des algorithmes. Les facteurs communs des structures proposées sont :

- Une fonction de précédence $P()$, définie sur l’ensemble des nœuds, et rendant possible, à partir de n’importe quel nœud, la remontée vers le cycle unique de l’arbre augmenté couvrant le nœud en question, et si nécessaire, le parcours de ce cycle.

Étant donné que l’orientation conventionnelle de l’arbre n’est pas celle du graphe original, l’information supplémentaire sur cette dernière doit aussi être disponible.

- Une fonction $T()$ de parcours de l’arbre, selon un ordre de succession des nœuds donné (par exemple, le préordre) ou simplement une fonction de succession $S()$. Selon le choix de ces fonctions, nous classons les structures de données connues en deux grandes catégories :

- Les structures de données de type ATI (“Augmented Threaded Index”), basées sur un parcours en préordre de l’arbre (fonction $\text{Préordre}()$) avec une information supplémentaire qui peut être la profondeur du nœud dans l’arbre, ou sa distance au cycle (Brown et McBride [12]), ou encore le nombre de ses successeurs (Bradley, Brown et Graves [11]).
- Les structures de type API (“Augmented Predecessor Index”), avec ce que l’on nomme un *père* dans le rôle du prédécesseur, un *fil*s représentant l’un des successeurs et des *frères* partageant le même père.

Les deux types de structures sont illustrées dans la figure 3.1 sur un arbre augmenté à 11 nœuds et 11 arcs, avec un cycle unique (1-2-3-4-1).

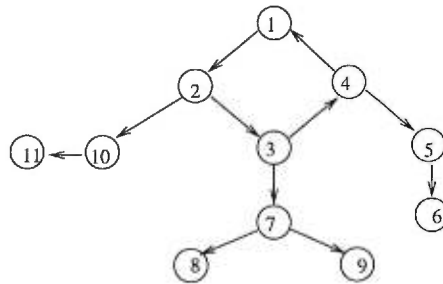


FIG. 3.1 – Un exemple d'arbre augmenté

Nœud	Représentation de type ATI			Représentation de type API		
	Prédécesseur	Préordre	Profondeur	Père	Fils	Frère
1	4	2	0	4	2	5
2	1	10	0	1	3	-1
3	2	7	0	2	4	10
4	3	5	0	3	1	7
5	4	6	1	4	6	-1
6	5	1	2	5	-1	-1
7	3	8	1	3	8	-1
8	7	9	2	7	-1	9
9	7	4	2	7	-1	-1
10	2	11	1	2	11	-1
11	10	3	2	10	-1	-1

TAB. 3.1 – Représentations ATI et API d'un arbre augmenté

3.1.5 L'algorithme du simplexe réseau avec gains

Les conditions d'optimalité

Soit x^* une solution de base réalisable du problème (3.3). La condition d'optimalité de x^* , traduisant les écarts complémentaires pour (3.3) et son dual est l'existence d'un vecteur y^* , à m composantes, tel que le couple (x^*, y^*) vérifie :

$$\begin{cases} 0 < x_a^* < u_a & \Rightarrow \bar{c}_a = c_a - y_i^* + \theta_a y_j^* = 0, \text{ où } a = (i, j); \\ x_a^* = 0 & \Rightarrow \bar{c}_a \geq 0; \\ x_a^* = u_a & \Rightarrow \bar{c}_a \leq 0 \end{cases}$$

L'énoncé de l'algorithme - Ahuja, Magnanti et Orlin [1]

1- Initialisation

Construction d'une solution de base réalisable initiale x_0 par l'ajout de variables artificielles (boucles aux nœuds) à très grand coût unitaire.

Calcul des variables duales associées.

2- Choix d'une variable d'entrée

Les arcs admissibles à l'entrée dans la base sont les arcs hors-base vérifiant ($x_a = 0$ et $\bar{c}_a < 0$) ou ($x_a = u_a$ et $\bar{c}_a > 0$). Si aucun arc n'est admissible et que la base ne contient pas d'arcs artificiels, la solution de base courante est optimale; si des arcs artificiels existent dans la base, alors le problème n'est pas réalisable.

3- Détermination d'une variable de sortie

Un arc bloquant pour l'augmentation de flot sur l'arc d'entrée est choisi pour quitter la base.

4- Pivotage

Mise à jour des structures de base et des variables primales et duales.

Retour à l'étape 2.

La mise à jour des variables duales

Soit \mathcal{F} l'ensemble d'arbres augmentés associés à une solution réalisable de base x . En considérant une orientation donnée des arcs, nous définissons le multiplicateur du chemin $[s, p]$ de source s et de puits p par $\Gamma_{[s,p]} = \prod_{a \in [s,p]} \theta_a$. En particulier, lorsque $s = p$, et que $\mathcal{C} = [s, p]$ est un cycle, nous parlons de multiplicateur du cycle \mathcal{C} , $\Gamma_{\mathcal{C}}$. Nous faisons l'hypothèse que l'algorithme du simplexe maintient des cycles strictement générateurs de flot, ou strictement absorbants; en d'autres termes, que $\Gamma_{\mathcal{C}} > 1$ ou $\Gamma_{\mathcal{C}} < 1$.

La mise à jour des variables duales $(y_i), i \in N$, repose sur la condition de réalisabilité duale :

$$\bar{c}_a = c_a - y_i + \theta_a y_j = 0; \quad \forall a = (i, j) \in \mathcal{F}. \quad (3.4)$$

Il est intéressant à ce propos de remarquer que seules les variables duales correspondant aux nœuds associés aux arbres augmentés sollicités par un pivot devront être mises à jour, à chaque itération ; d'autre part, étant donné qu'un arbre augmenté est constitué d'un cycle \mathcal{C} et d'un ensemble de sous-arbres "attachés" aux nœuds $i \in I_{\mathcal{C}}$ du cycle, notons que le calcul des variables duales se réduit à celui d'une seule variable $y_i, i \in I_{\mathcal{C}}$, les autres se déduisant directement, par récursivité, grâce à (3.4). Par exemple, et comme le montre la figure 3.2, nous avons successivement $y_1 = \frac{y_0 - c_{(0,1)}}{\theta_{(0,1)}}$, $y_2 = \frac{y_1 - c_{(1,2)}}{\theta_{(1,2)}}$, et ainsi de suite.

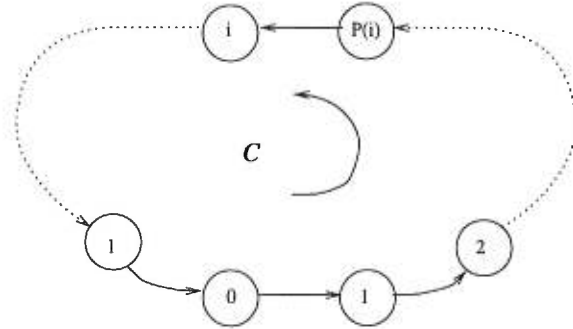


FIG. 3.2 – Mise à jour des variables duales sur un cycle

En général, il est possible de montrer que :

$$c_a - y_i + \theta_a y_j = 0; \quad \forall a = (i, j) \in \mathcal{C}. \quad (3.5)$$

détermine un système linéaire de $l + 1$ équations à $l + 1$ inconnues, possédant une solution unique $\{y_i\}_{i \in I_{\mathcal{C}}}$ où :

$$y_0 = \frac{\sum_{a \in \mathcal{C}} c_a \Gamma_{[0,j]}}{\Gamma(\mathcal{C}) - 1}$$

lorsque l'hypothèse $\Gamma(\mathcal{C}) \neq 1$ est vérifiée.

La détermination de l'arc de sortie et la mise à jour des valeurs de flot

Soit (p, q) l'arc choisi pour l'entrée dans la base. La question est de déterminer la répercussion sur les valeurs de flot des arcs de la base, d'un défaut d'une unité et d'un excès de $\theta_{(p,q)}$ unités, créés par l'envoi d'une unité de flot sur (p, q) , de manière

à préserver la réalisabilité de la solution.

Les figures 3.3 et 3.4 représentent des sous-arbres augmentés obtenus en traçant le chemin de précédence \mathcal{P} à partir de la source p de l'arc d'entrée, et en y ajoutant le cycle unique \mathcal{C} de l'arbre augmenté correspondant. Ces sous-réseaux, avec ceux qui correspondent de façon tout à fait symétrique à la destination q de l'arc d'entrée, sont les seules portions de l'arbre augmenté qui verront la valeur de leur flot changer suite au pivot courant. Nous allons donc illustrer le calcul des valeurs de flot uniquement sur ces arbres augmentés. Par commodité, nous posons $p = 0$, et nous définissons :

- \tilde{x}_i , l'augmentation de la valeur de flot sur l'arc $(P(i), i), i \in \mathcal{P} \cup \mathcal{C}$, résultant du défaut d'une unité en p ;
- $\tilde{\theta}_i = \theta_{(P(i), i)}$ le multiplicateur de $(P(i), i)$.

Deux cas sont alors à considérer :

1. \mathcal{C} est un cycle à plus d'un nœud :

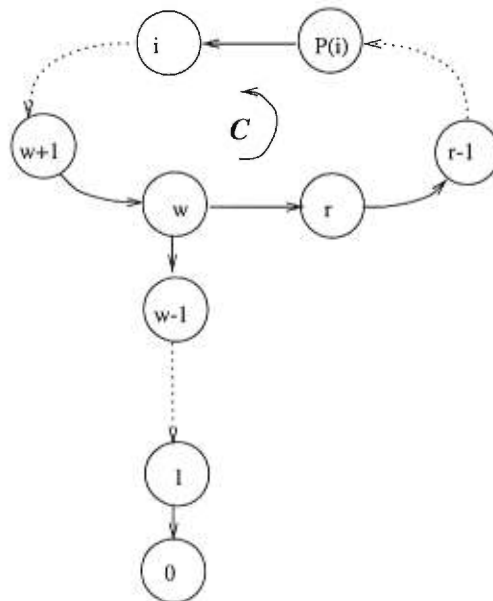


FIG. 3.3 – Le chemin de précédence dans le premier cas

À un défaut de flot au nœud 0 (voir 3.3), conséquence de l'envoi d'une unité de flot sur l'arc d'entrée, correspond l'envoi de $\theta_{(0,1)}$ unités de flot sur l'arc $(0, 1)$,

$$\begin{bmatrix} \tilde{\theta}_0 & & & & & \\ -1 & \tilde{\theta}_1 & & & & \\ & & \ddots & & & \\ & & & -1 & \tilde{\theta}_{w-1} & \\ & & & & -1 & \tilde{\theta}_w - 1 \end{bmatrix} \begin{bmatrix} \tilde{x}_0 \\ \tilde{x}_1 \\ \vdots \\ \tilde{x}_{w-1} \\ \tilde{x}_w \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad (3.7)$$

dont la solution unique, obtenue par substitution, est :

$$\begin{cases} \tilde{x}_k = \frac{1}{\prod_{i=0}^k \tilde{\theta}_i}, & k = 0, 1, \dots, w-1; \\ \tilde{x}_w = \frac{\tilde{x}_{w-1}}{\tilde{\theta}_w - 1}. \end{cases}$$

La convergence de l'algorithme

La présence de plusieurs arcs candidats à la sortie de la base, à chaque itération, conduit au phénomène des itérations dégénérées, bien connu en programmation linéaire, où aucune amélioration n'est apportée à la fonction objectif, et a la conséquence que la convergence de l'algorithme n'est plus assurée. Le fait additionnel que, dans le contexte des réseaux en particulier, les itérations dégénérées sont extrêmement fréquentes donne beaucoup d'importance à la prévention du comportement cyclique du simplexe réseau.

Deux techniques classiques peuvent être adaptées. La première est la perturbation qui consiste à substituer au vecteur d'offre/demande b le vecteur $b' = b + \xi$ où $\xi = (\epsilon_1, \epsilon_2, \dots, \epsilon_m)$ et les $\{\epsilon_i\}_{i=1, \dots, m}$ sont des constantes suffisamment petites. La seconde technique, connue sous le nom de règle lexicographique, est une implantation implicite de la perturbation, plus robuste dans la mesure où elle ne nécessite pas de spécifier des valeurs adéquates pour les constantes de perturbation.

Dans le contexte spécifique des réseaux généralisés, Elam, Glover et Klingman [27], développent une règle de choix de l'arc de sortie qui s'appuie sur ce que les auteurs dénotent par *bases fortement convergentes*, possédant les deux propriétés additionnelles que :

- le flot sur chaque arc de la base, ne correspondant pas à une variable artificielle,

n'est pas à sa borne supérieure de capacité ;

- Chaque cycle \mathcal{C} , qui n'est pas une boucle, est absorbant, c'est à dire que $\Gamma(\mathcal{C}) < 1$.

Elam, Glover et Klingman montrent qu'avec un choix approprié de l'arc de sortie, l'algorithme maintient en permanence des bases fortement convergentes ; de plus, ils établissent la convergence en montrant qu'il ne peut y avoir qu'un nombre fini d'itérations dégénérées successives.

Pour conclure, notons l'article de Orlin [49], qui établit le lien entre le règle de choix de l'arc de sortie de Elam, Glover et Klingman et la règle lexicographique.

3.2 Une approche de résolution exploitant la structure multiproduits

Jusque là, l'approche préconisée consistait à résoudre le problème d'allocation par une application directe, à la représentation réseau correspondante, d'un algorithme de flot à coût minimum. Le défaut d'une telle démarche est principalement qu'elle ne tient pas compte des caractéristiques du réseau associé à notre application qui, rappelons le, possède une structure à plusieurs couches correspondant chacune à un type de conteneurs, et qui sont reliées par des arcs de substitution dont l'omission décompose le réseau en plusieurs sous-réseaux disjoints. Il est également important de noter que ces derniers sont des réseaux dynamiques purs.

Dans cette section, nous nous proposons de remédier à cette limitation en présentant dans un premier temps une approche de décomposition primale du problème basée sur la technique de restriction que nous appliquons aux variables de flot associées aux arcs de substitution. L'algorithme générique résultant sera alors à la base des stratégies particulières que nous développons par la suite. Nous examinons également la possibilité d'appliquer des algorithmes dynamiques généraux pour une meilleure prise en compte de la structure multipériodes du réseau.

3.2.1 L'approche de restriction classique

Nous rappelons, dans un premier temps, le principe de l'approche générale de restriction. Les principes directeurs de cette approche seront par la suite spécialisés à notre problème pour produire un algorithme générique de décomposition.

Considérons le problème de programmation mathématique (P) suivant :

$$\begin{aligned} \text{Min } & f(x) \\ & g_i(x) \leq 0, \quad i \in M \\ & x \in X \end{aligned}$$

où M est un ensemble d'indices de contraintes $\{1, 2, \dots, m\}$, X un ensemble convexe et f, g_1, g_2, \dots, g_m sont des fonctions de classe C^1 sur l'ensemble X .

Soit $R \subset M$ un sous-ensemble d'indices de contraintes en fonction duquel nous définissons le problème restreint (P^R):

$$\begin{aligned} \text{Min } & f(x) \\ & g_i(x) \leq 0, \quad i \in M - R \\ & g_i(x) = 0, \quad i \in R \\ & x \in X \end{aligned}$$

Avant de présenter l'algorithme de restriction, nous faisons les hypothèses de régularité du problème restreint suivantes :

- Quand (P^R) est réalisable et borné inférieurement, une solution optimale x^* pour (P^R) existe.
- Chaque minimum local de (P^R) est un point régulier pour les contraintes, ce qui signifie, en d'autres termes, que les gradients $\nabla g_i(x^*)$, $\forall i \in R \cup J$, sont linéairement indépendants, J étant l'ensemble des indices j des contraintes d'inégalité actives en x^* (telles que $g_j(x^*) = 0$). Cette hypothèse permet de conclure à l'existence d'un vecteur optimal de multiplicateurs de Lagrange tel que les conditions d'optimalité de Kuhn-Tucker soient vérifiées en x^* .

L'approche de restriction consiste alors à :

Étape 0 - Initialisation

Choisir un sous-ensemble initial R et prendre $opt = \infty$.

Étape 1

Résoudre (P^R) .

Si (P^R) est non borné inférieurement, alors (P) l'est aussi. Arrêt de l'algorithme.

Si (P^R) est non réalisable et $R \neq \emptyset$, faire $R = R - V$ où V est un sous-ensemble de R et reprendre à l'étape 1. Si $R = \emptyset$, $(P) = (P^R)$ est non réalisable et l'algorithme s'arrête.

Étape 2

Identifier une solution optimale de (P^R) , x^R et un vecteur optimal de multiplicateurs de Lagrange associé $\{\xi_i^R\}_{i \in M}$.

Vérifier si (x^R, ξ^R) satisfait les conditions d'optimalité de Kuhn-Tucker pour le problème (P) : $\xi_i^R \geq 0, \forall i \in R$.

Étape 3

Si x^R n'est pas une solution optimale pour (P) , soit $V \subseteq R$, contenant au moins un indice de contrainte i telle que $\xi_i^R < 0$ et $D \subseteq J$ un sous-ensemble de contraintes d'inégalité actives en x^R .

Si $f(x^R) = opt$, faire $S = S - V$ et retourner à l'étape 1.

Si $f(x^R) < opt$, faire $S = S \cup D - V$, $opt = f(x^R)$ et retourner à l'étape 1.

La convergence de la procédure de restriction est assurée par le résultat suivant :

Théorème 1 (Lasdon [43]) *La suite des valeurs de la fonction objectif $f(x^R)$ produite par la procédure de restriction est décroissante et l'algorithme, au bout d'un nombre fini d'itérations, termine soit en déterminant une solution vérifiant les conditions d'optimalité pour le problème (P) , soit en concluant que (P) est non réalisable ou non borné inférieurement.*

3.2.2 Spécialisation au problème du flot à coût minimum dans la représentation réseau

Une extension très commune de la procédure classique de restriction présentée consiste à augmenter l'ensemble des contraintes $g_i(x) \leq 0, i \in M$, en y intégrant les contraintes sur les variables qui forment généralement l'ensemble convexe de définition X du problème (P). Dans le cas du modèle d'allocation déterministe que nous avons présenté, l'ensemble de définition est constitué des contraintes $x_a \geq 0, a \in A$, où A est l'ensemble des arcs de la formulation réseau ; et parmi ces contraintes, celles concernant la non-négativité du flot sur les arcs de substitution sont particulièrement intéressantes dans la mesure où la restriction sur ces contraintes revient à annuler la valeur du flot correspondant. Ceci est, intuitivement, une perspective très attrayante étant donné qu'elle correspond à supprimer la substitution et donc à décomposer le problème.

Examinons à présent plus en détail l'adaptation de la procédure de restriction à notre problème. (P) est le problème de flot à coût minimum ($PFCM$) suivant :

$$\begin{aligned} \text{Min } & cx \\ \mathcal{A}x &= b \\ x_a &\geq 0, \quad a \in \mathcal{S} \\ x &\in X \end{aligned}$$

où $\mathcal{S} \subset A$ est l'ensemble des indices des arcs de substitution, et ($PFCM^R$) est le problème restreint où certaines valeurs de flot de substitution sont astreintes à être nulles :

$$\begin{aligned} \text{Min } & cx \\ \mathcal{A}x &= b \\ x_a &\geq 0, \quad a \in \mathcal{S} - R \\ x_a &= 0, \quad a \in R \\ x &\in X \end{aligned}$$

Nous choisissons de préciser l'ensemble initial R , dont le choix par la procédure générale de restriction est arbitraire. La solution la plus naturelle consiste à prendre $R = \mathcal{S}$, ce qui assure une résolution de $|\mathcal{S}|$ sous-problèmes indépendants de flot à coût minimum à la première itération de l'algorithme.

La propriété de réseau *acyclique* que possède la formulation réseau de notre problème d'allocation dynamique intervient pour simplifier la procédure de restriction. En effet, le problème $(PFCM)$ (et a fortiori les problèmes $(PFCM^R)$), ne peut être non borné inférieurement ; ceci rend le contrôle correspondant, effectué à l'étape 1 de l'algorithme général, superflu.

Finalement, l'utilisation des variables artificielles nous permet également d'unifier les cas de figures ((P^R) est non réalisable et $R \neq \emptyset$) et (x^R une solution optimale de (P^R)) vers une conclusion unique $R = R - V$, dans la mesure où une solution non-réalisable pour un sous-problème P^R , c'est à dire comportant des variables artificielles non nulles dans la base correspondante, peut être utilisée, au même titre qu'une solution réalisable optimale de P^R , comme point de départ à l'itération suivante. Et enfin, le choix de $D = \emptyset$ a pour conséquence de fusionner les deux cas $f(x^R) = opt$ et $f(x^R) < opt$ lors de l'étape 3.

Le nouvel algorithme se présente donc comme suit :

Étape 0 - Initialisation

$R = \mathcal{S}$ est initialement formé de tous les indices des contraintes de non-négativité des valeurs du flot sur les arcs de substitution.

Étape 1 - Boucle principale

Étape 1-1: Résoudre $(PFCM^R)$. Si $(PFCM^R)$ est non réalisable, aller à l'étape 1-2, sinon, aller à 1-3.

Étape 1-2: Si $R = \emptyset$, alors $(PFCM^R) = (PFCM)$ est non réalisable. Arrêt de l'algorithme.

Sinon, aller à l'étape 1-4.

Étape 1-3: Identifier un flot à coût minimum x^R pour le problème restreint, ainsi que des prix optimaux y^R pour le dual associé.

Si les conditions d'optimalité (les écarts complémentaires) du problème $(PFCM)$ sont satisfaites par le couple (x^R, y^R) , arrêt de l'algorithme.

Étape 1-4: Choisir $V \subseteq R$, faire $R = R - V$ et retourner à l'étape 1-1.

Remarques

1. Nous distinguons clairement dans l'algorithme précédent l'étape d'initialisation qui consiste à résoudre $|\mathcal{S}|$ sous-problèmes de flots à coût minimum, du reste du processus. Dans les analyses qui suivent, nous désignerons cette étape préliminaire par *phase 1* de l'algorithme et le reste par *phase 2*.
2. À chaque itération, l'étape 1-4 consiste en une relaxation d'un certain nombre de contraintes $x_a = 0$, $a \in R$, ce qui revient à tenir compte, à l'itération qui suit, de certaines variables de flot, dont les indices font partie de l'ensemble R . Si nous faisons appel à la formulation réseau du modèle d'allocation, chaque élément $a \in V$ est un indice d'un arc de substitution. Si G_a^- et G_a^+ sont respectivement les sous-réseaux (correspondant à un ou plusieurs types de conteneurs) associés à l'origine et à la destination de l'arc a , l'opération $R = R - V$ peut être vue, de façon schématique, comme faisant partie d'une fusion des deux sous-réseaux G_a^- et G_a^+ .
3. En pratique, après une relaxation $R' = R - V$, la résolution du problème

($PFCM^R$) à l'étape 1.1 utilisera les solutions optimales du problème restreint de l'étape précédente ($PFCM^R$), comme de bonnes solutions réalisables de départ, en faisant ensuite appel à un processus de réoptimisation.

4. L'algorithme est générique dans ce sens que le choix de l'ensemble V des indices de variables de substitution utilisé pour relâcher la restriction à l'étape 1-4 de l'algorithme est arbitraire. En pratique toutefois, un choix bien défini de cet ensemble donne lieu à des stratégies particulières, dont les plus importantes sont :

- La stratégie de *fusion globale*, où à la première itération, toutes les variables de flot restreintes sont relaxées ($V = R$).
- La stratégie de *fusion progressive*, avec $V = A_i \subset R$ où $A_i \neq R$ est un ensemble bien défini et généralement restreint d'arcs à intégrer à une itération i de l'algorithme. Cette stratégie, de par le choix libre du sous-ensemble A_i , possède plusieurs variantes (fusion progressive à une étape, fusion progressive binaire, etc.) que nous allons voir en détail plus loin.

3.2.3 Les avantages potentiels de l'algorithme

Par rapport à l'application directe de l'algorithme standard de flot à coût minimum à la formulation réseau du modèle, le nouvel algorithme, basé sur la restriction du flot sur les arcs de substitution, possède trois gros avantages :

- La première phase d'initialisation consiste en une résolution de sous-problèmes de flot à coût minimum dans des sous-réseaux. Ces derniers ont la particularité d'être des réseaux *purs*, étant donné que tous les flots de substitution sont astreints à être nuls. Ceci permet notamment une résolution plus rapide en utilisant des algorithmes plus simples de flot à coût minimum dans les réseaux purs.
- La résolution des sous-problèmes ($PFCM^R$) fait intervenir des structures de données moins volumineuses (notamment les structures de représentation de la base et les listes d'arcs candidats à l'entrée dans la base) qu'une résolution

globale. Nous avons donc des raisons de penser que l'implantation des stratégies précédentes donne lieu à des gains significatifs de performance.

- Les stratégies particulières de l'algorithme générique ont toutes, plus ou moins, un potentiel intrinsèque de parallélisation, durant les deux phases de l'algorithme, par :
 - la résolution initiale du flot à coût minimum sur les sous-réseaux ;
 - le choix de plusieurs ensembles disjoints V_1, V_2, \dots, V_p durant la phase de relaxation de la restriction, tels que l'identification de sous-problèmes indépendants soit possible, et la résolution en parallèle de ces derniers à l'itération suivante.

3.3 Prise en compte des propriétés dynamiques du réseau

La structure multipériodes sous-jacente au réseau de transport associé au problème d'allocation est l'autre caractéristique structurelle significative. En effet, si en théorie l'application de méthodes classiques de résolution sur la représentation réseau reste envisageable, des limitations existent sur la taille des problèmes à résoudre et suggèrent le recours à des algorithmes dynamiques plus efficaces.

Sur un plan méthodologique, les algorithmes progressifs (“forward algorithms”), dont le principe général consiste en la résolution par étapes, sur des horizons de plus en plus grands, de sous-problèmes du problème global, font partie des approches les mieux adaptées aux réseaux multipériodes. Dans cette section, nous présentons les grandes lignes d'un algorithme progressif du type simplexe réseau, dû à Aronson et Chen [3] et destiné à des structures multipériodes très générales. Bien que cet aspect sorte du cadre de notre recherche, dont l'objectif principal est, rappelons-le, le développement d'approches de décomposition basées sur le caractère multiproduits du problème, il nous a paru nécessaire de souligner l'intérêt d'un algorithme dynamique et de noter la possibilité de l'adapter afin d'exploiter également la structure multipériodes du modèle d'allocation de conteneurs vides.

3.3.1 Un algorithme progressif du type simplexe réseau

Soit $G = (N, A)$ un réseau de transport multipériodes, défini sur un horizon de planification formé de T périodes, où $N = \bigcup_{t=1}^T N(t)$ et $A = \bigcup_{t=1}^T A(t)$. L'ensemble $N(t)$ est formé de couples (i, t) , $i \in I(t)$, désignant les nœuds actifs durant la période t , et $A(t)$ est l'ensemble d'arcs $((i, t), (j, t'))$ qui ont leur source à la période t . Afin de simplifier le développement de l'algorithme, nous supposons que les temps de transit $\tau = t' - t$ ne peuvent être que 1, -1 ou 0. C'est une hypothèse qui, toutefois, n'occasionne aucune perte de généralité car il est possible, en ajoutant des nœuds et des arcs artificiels, de transformer un problème à temps de transit quelconques afin de le rendre conforme à cette hypothèse.

Le problème de flot à coût minimum dans le réseau multipériodes (*PFCMMP*) se formule alors comme suit :

$$\min \sum_{t=1}^T \sum_{a \in A(t)} c_a x_a$$

sujet à :

$$\sum_{a \in A_{(i,t)}^+} x_a - \sum_{a \in A_{(i,t)}^-} x_a = b(i, t); \quad \forall i \in N(t) \quad (3.8)$$

$$0 \leq x_a \leq u_a; \quad \forall a \in A(t)$$

où $A_{(i,t)}^-$ et $A_{(i,t)}^+$ sont les ensembles d'arcs adjacents intérieurement et extérieurement au nœud (i, t) , $b(i, t)$ est la demande du nœud, et x_a , u_a et c_a sont respectivement la valeur du flot, la borne supérieure de capacité et le coût unitaire de l'arc $a \in A(t)$.

Le principe du simplexe réseau progressif consiste à résoudre une série de sous-problèmes formés à partir d'une restriction du problème (*PFCMMP*) en ne considérant que les nœuds et les arcs de l'intervalle $[1, t]$. Le passage du sous-problème de l'intervalle $[1, t]$ à celui de $[1, t + 1]$ s'effectue en construisant une solution réalisable à

partir de la solution optimale du premier sous-problème et des variables relaxées par la prise en considération de l'intervalle $[t, t + 1]$. La procédure continue ainsi jusqu'à ce qu'un critère d'arrêt soit satisfait (qui, en pratique, peut être la fin de l'horizon de planification ($t = T$), l'atteinte d'un niveau satisfaisant pour la valeur optimale de la fonction objectif des sous-problèmes, ou l'obtention d'un niveau suffisamment faible pour les fluctuations de cette valeur, pour un nombre donné de périodes successives).

Plus précisément, l'algorithme proposé par Aronson et Chen procède comme suit :

Étape 0 - Initialisation

$t = 1$ et $Z_t = 1$.

Étape 1 - Augmentation à une solution réalisable du sous-problème de la période t

Ajouter à la représentation du problème les nœuds et les arcs de la période t et augmenter la solution optimale courante par :

- l'ajout de variables artificielles associées à l'offre/demande des nœuds de la période t , ou,
- l'ajout des valeurs optimales de flot obtenues en considérant localement le sous-réseau de la période t exclusivement.

Étape 2 - Choix d'un arc admissible à l'entrée dans la base

Choisir (à l'aide d'une heuristique de choix de l'arc d'entrée) un arc admissible, parmi les arcs des sous-ensembles $A(\tau), \tau = Z_t, \dots, t$, en restreignant les arcs $((i, t), (j, t + 1))$ à être à leur borne inférieure de capacité. Si aucun arc admissible n'est trouvé de cette façon, aller à l'étape 4 ; sinon continuer à l'étape 3.

Étape 3 - Pivotage, mise à jour des variables primales et duales et de l'indice Z_t

Réaliser un pivot du simplexe, et mettre à jour les valeurs de flots aux arcs et des variables duales aux nœuds.

Soit Y_t la première période (dans l'axe du temps) durant laquelle la variable duale associée à un nœud $(i, \tau), \tau = Z_t, \dots, t$ a été mise à jour par le pivot courant.

$$Z_t = \min\{Z_t, \max\{Y_t - 1, 1\}\}.$$

Retourner à l'étape 2.

Étape 4 - Test d'optimalité

Si un critère d'arrêt est satisfait (par exemple, si $t = T$), la solution est optimale. Sinon mettre $Z_t = t, t = t + 1$ et retourner à l'étape 1.

L'introduction de l'indice Z_t , justifiée par le résultat ci-dessous, est la clé de l'efficacité de l'algorithme. En effet, le fait que la recherche des arcs candidats à l'entrée dans la base se limite, à chaque itération, aux arcs de l'intervalle $[Z_t, t]$ réduit les besoins en mémoire pour les listes de candidats, ainsi que les temps de calcul nécessaires

à la mise à jour des coûts réduits et au parcours de la liste.

Théorème 1 (Aronson et Chen [3]) *Supposons que nous résolvons la restriction du problème (PFCMMP) à l'intervalle $[1, t]$. Alors aucun des arcs $a \in A(\tau)$, $\tau < Z_t$ n'est admissible à l'entrée dans la base, lors de l'étape de choix de l'arc d'entrée.*

Chapitre 4

Implantation et exploitation des données

Dans ce chapitre, nous nous intéressons à l'implantation des données du problème de l'allocation de conteneurs vides. Ainsi, nous proposons tout d'abord un format de fichier pour le modèle mathématique. Nous examinons ensuite les principaux traitements visant l'exploitation des données par les différentes approches de résolution, notamment le processus de construction de la représentation réseau correspondante.

4.1 Organisation des données

Comme nous avons pu le constater en présentant le modèle mathématique, les données du problème de l'allocation déterministe, dynamique et multiproduits de conteneurs vides sont d'une grande complexité. Une attention particulière doit par conséquent être accordée au support informatique de ces données (par le biais d'un système de fichiers ou de bases de données), ainsi qu'à leur exploitation, tout spécialement par le processus de construction de la représentation réseau correspondante.

À cette fin, notre choix s'est porté sur le format NFF (Network File Format), développé au Laboratoire sur les systèmes intelligents de transport du Centre de recherche sur les transports. NFF nous a paru être un excellent choix pour les données du modèle d'allocation déterministe de conteneurs vides, étant donné qu'il a été conçu à l'origine pour le support des données des modèles de réseaux, pour lesquels il est extrêmement bien adapté. D'autres facteurs tels une syntaxe simple et la disponibilité d'une librairie de développement pour le langage C++ ont également motivé notre décision.

Sans donner une description détaillée du format NFF, pour laquelle nous référons le lecteur au site WWW consacré à NFF (http://www.crt.umontreal.ca/~lab_sit/DOC-NFF), nous énonçons ses éléments de base, qui sont :

- l'objet NFF, qui est une entité indépendante d'information faisant partie du système étudié. Dans le contexte des réseaux, un objet *réseau* est un exemple typique d'objet NFF.
- la section, qui désigne un ensemble de composantes de l'objet NFF qui ont en commun une même structure. À titre d'exemple, *nœuds* et *arcs* peuvent être des sections de l'objet *réseau*.
- les attributs d'une section, qui sont les champs d'information des sections. Dans un problème de flot, *coût unitaire* peut être l'attribut d'une section *arcs*, elle même faisant partie d'un objet *réseau*.

Pour les attributs, NFF permet l'utilisation de types de données élémentaires (entiers, réels, chaînes de caractères), de listes et d'un type particulier appelé *référence*, qui agit comme un pointeur vers les sections et permet d'établir des liens entre ces dernières.

4.1.1 Une classification des données du modèle mathématique

Une subdivision possible des données du modèle mathématique est la suivante :

Données générales

Cette catégorie comprend :

- T : la longueur de l'horizon de planification ;
- P, C, H, D : les ensembles respectifs de produits, de clients, de ports et de dépôts terrestres ;
- $\tau_{\eta\mu}$: le temps de transit d'une paire de nœuds origine/destination (η, μ) , qui peuvent être des clients, des ports et des dépôts terrestres.

Ce sont les données les plus simples étant donné que leur correspondance avec les sections associées, illustrée par le tableau 4.1, est directe et qu'elles ne requièrent qu'une information minimale pour leur identification.

Données associées aux requêtes des clients

Le modèle mathématique définit l'ensemble des *requêtes identifiées de clients* comme étant les requêtes d'offre et de demande d'un produit spécifique, pour une période de temps donnée. La catégorie de données reliées directement aux requêtes comprend :

- $I^{pt}(S^{pt})$: les ensembles de requêtes identifiées de demande (d'offre) des clients ;
- $X_i^{pt}(Y_s^{pt})$: les valeurs de demande (d'offre) de la requête d'un client i (s) ;
- Δ_i : la valeur de la fenêtre temporelle de livraison pour une requête de demande i ;

Section	Attributs	Description
LengthHorizon	Value	longueur de l'horizon de planification
Commodities	Id	identificateur du produit
Customers	Id	identificateur du client
Ports	Id	identificateur du port
Depots	Id	identificateur du dépôt terrestre
TransitTimes	TypeOfOrigin, TypeOfDestination, ReferenceToOrigin, ReferenceToDestination, Value	catégorie de l'origine (client, port ou dépôt) catégorie de la destination référence à l'origine référence à la destination valeur en nombre de périodes de temps du transit

TAB. 4.1 – Sections et attributs généraux

- K_i^{pt} : le volume de conteneurs de type p envoyés au client demandeur i avant le début de l'horizon de planification, et qui arrivent à destination à la période t .

La figure 4.1 illustre les sections correspondantes et les relations qui existent entre elles : une section *RequestsByPeriodAndCommodity* correspond aux ensembles I^{pt} et S^{pt} , et regroupe l'information sur les requêtes formulées pour chaque période et pour chaque type de conteneurs. Cette information se présente sous forme de listes de références à des sections *DemandCustomersRequests* et *SupplyCustomersRequests* contenant les données X_i^{pt} , Y_s^{pt} , Δ_i , K_i^{pt} spécifiques à chaque requête.

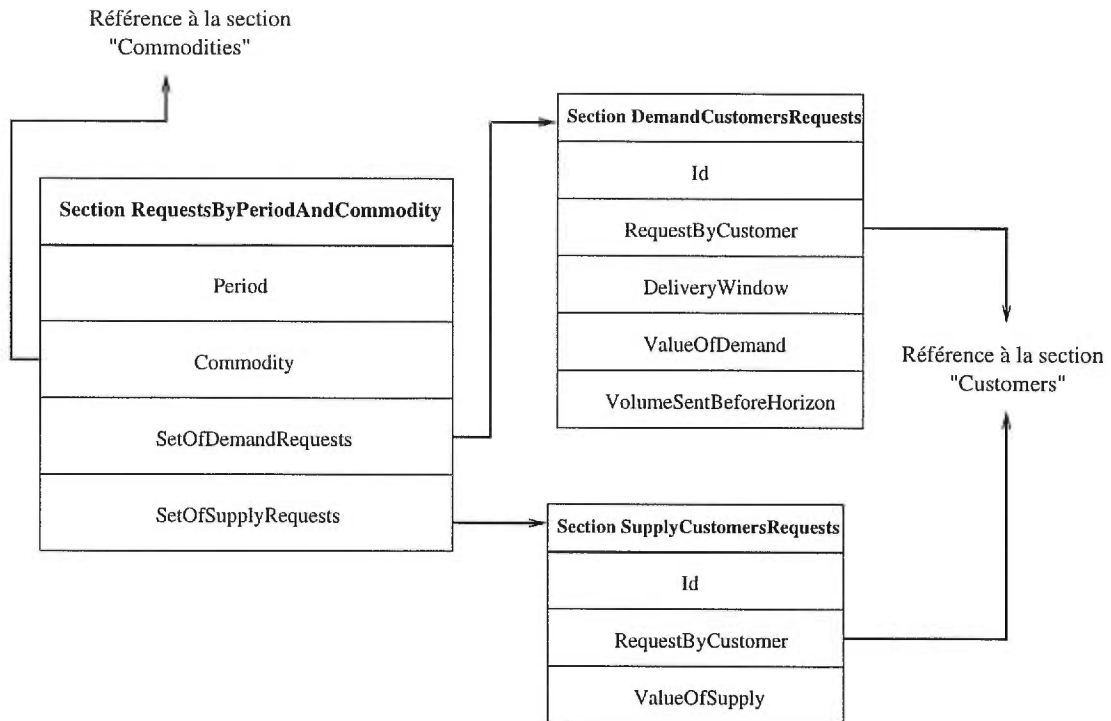


FIG. 4.1 – Sections et attributs associés aux requêtes

Données reliées à l'allocation et au retour des conteneurs vides

Nous identifions dans cette catégorie les éléments suivants :

- données établissant les associations de service entre les dépôts et les requêtes $J_i^{pt'}(J_s^{pt})$ et $I_j^{pt}(S_j^{pt})$;
- données associées spécifiquement aux dépôts et relatives à leur activité d'allocation et de récupération de conteneurs:
 - $X_j^{pt}(Y_j^{pt})$: demandes d'export (offres importées) du dépôt j en produit p durant la période t ;
 - K_j^{pt} : volumes de conteneurs envoyés au dépôt j avant le début de l'horizon de planification ;
 - w_{jj}^{p0} le stock de conteneurs disponible au dépôt j au début de l'horizon.

La figure 4.2 présente les sections *SetsOfPotentiallySatisfiedRequests* décrivant les associations de service dépôts/requêtes, les sections *PortsInService* et *DepotsInService*

contenant l'information sur les dépôts en activité et enfin, les sections *AllocationFlowLinks* et *PickUpFlowLinks* associées respectivement aux mouvements d'allocation de la demande et de récupération de l'offre.

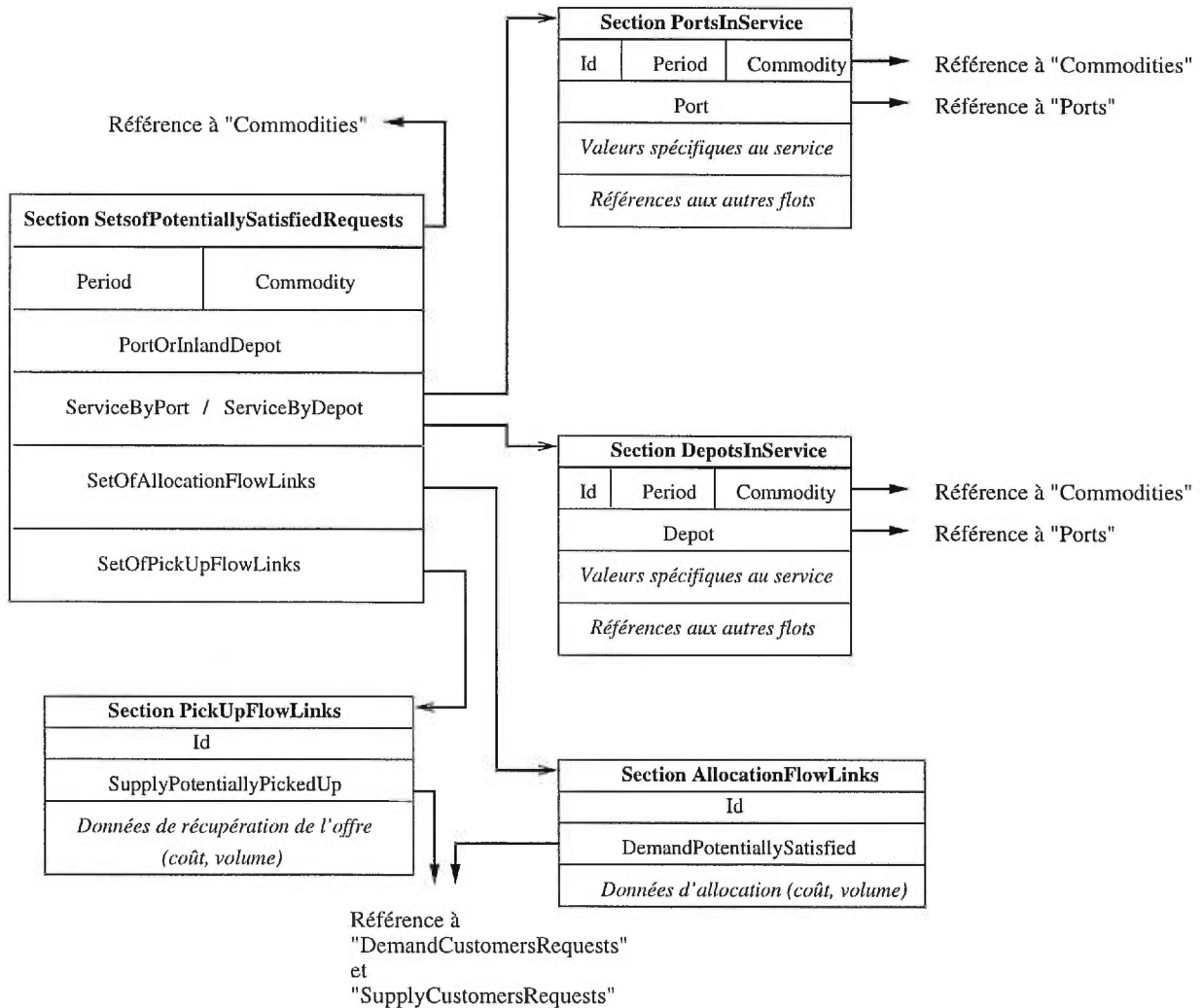


FIG. 4.2 – Sections et attributs associés à l'allocation et au retour des conteneurs vides

Données reliées aux mouvements et opérations autres que d'allocation et de retour

Il reste à présenter les sections et attributs supportant les données relatives aux demandes d'export non satisfaites par les ports, aux mouvements d'équilibrage inter-dépôts, au flux de nouveaux conteneurs introduits dans le système, ainsi qu'à l'inventaire et la substitution qui s'opèrent au niveau des dépôts. En général, l'information requise, résumée dans le tableau 4.2, concerne le volume de conteneurs sur le lien, le coût unitaire, et éventuellement pour chaque type de liens des informations spécifiques telles que des bornes supérieures ou inférieures de capacité, des facteurs de substitution, etc.

Section	Attributs	Description
InventoryFlowLinks	Id, UnitCost, Volume	identificateur du flot d'inventaire coût unitaire et volume du flot
NewFlowLinks	Id, UnitCost, Volume	identificateur du flot de nouveaux conteneurs coût unitaire et volume du flot
InsatisfiedExport- FlowLinks	Id, UnitCost, Volume	identificateur de la demande d'export insatisfaite coût unitaire et volume de la demande
EquivalentFlowLinks	Id, Volume	identificateur du flot de conteneurs de type "physique" alloués par le dépôt volume du flot
BalancingFlowLinks	Id, Port, Depot, LowerBound, UpperBound, UnitCost, Volume	identificateur du flot d'équilibrage port ou dépôt destinataire bornes inférieure et supérieure de capacité coût unitaire et volume du flot
SubstitutionFlow- Links	Id, CommodityOfThe- Destination, SubstitutionFactor, UpperBound, UnitCost, Volume	identificateur du flot de substitution produit correspondant à la destination facteur de substitution borne supérieure de capacité coût unitaire et volume du flot

TAB. 4.2 – Sections et attributs relatifs aux mouvements et opérations autres que ceux de l'allocation et de la récupération de conteneurs

4.1.2 La mise en œuvre des données

La construction de la représentation réseau du problème est sans doute le processus de traitement le plus important des fichiers de données du modèle mathématique. En effet, rappelons que la formulation réseau est de façon générale, la mieux adaptée aux algorithmes d'optimisation sur les réseaux, et permet, en particulier, de distinguer clairement la structure "en couches" du réseau multiproduits ainsi que la disposition particulière des arcs de substitution, éléments qui motivent nos stratégies de résolution.

Nous choisissons de diviser le processus de construction en deux étapes, la première consistant à créer les nœuds, et la seconde les arcs. Cette manière de procéder, très naturelle en somme, possède l'avantage de limiter les besoins en mémoire, qui, à cause de la complexité du modèle, peuvent être très importants.

La question se pose toutefois de savoir comment établir de façon efficace, durant la construction des arcs, le lien entre la désignation logique des nœuds origine et destination et leur adresse physique obtenue lors de la phase de construction des nœuds. Par exemple, supposons que nous voulions créer un arc représentant le flot d'inventaire du dépôt j , pour le produit p , à la fin de la période t . Des références internes à des nœuds représentant le dépôt en question dans le réseau espace-temps durant les périodes t et $t + 1$ sont établies par la première phase de création des nœuds, et doivent être rendues accessibles à la seconde. Si nous désignons en général par C une catégorie de nœuds physiques (clients, ports, dépôts, nœuds fictifs, etc.), par i l'indice physique d'un nœud, p un produit donné et t une période temporelle, le problème revient à établir une association entre les quadruplés (C, i, p, t) désignant le représentant dans le réseau espace-temps et les références correspondantes $ref_C^{pt}(j)$, qui est nécessaire à l'établissement du lien entre les arcs et leurs nœuds source et puits.

Sur le plan de l'implantation informatique, il est possible d'utiliser un simple tableau établissant le lien entre tous les quadruplés (C, i, p, t) possibles et les références associées. C'est une solution qui accède directement à la référence pour un nœud donné mais qui occasionne une perte importante d'espace mémoire étant donné qu'unique-

ment un sous-ensemble parmi les quadruplés (C, i, p, t) possibles est pertinent. Une autre solution, consistant à utiliser une liste chaînée d'enregistrements associant les quadruplés (C, i, p, t) aux références correspondantes, donne la priorité à l'espace mémoire, mais présente d'autre part l'inconvénient que les temps de recherche des références sont linéaires par rapport au nombre de nœuds du réseau espace-temps. Nous choisissons donc un compromis entre les besoins en mémoire et l'efficacité d'extraction de l'information en mettant au point un mécanisme à base de *tables de correspondance*, qui utilisent la structure de données suivante :

```
// the basic structure for the map
struct MapItem {

    short category;
    long physicalIndex, commodity, period; // the quadruplet

    long Ref; // the internal reference to the node
};
```

Les champs “category”, “physicalIndex”, “commodity” et “period” de la structure **MapItem** désignent respectivement les éléments de (C, i, p, t) , tandis que “Ref” est la référence interne $ref_C^{pt}(j)$ du nœud correspondant.

Une table :

```
// the map array
MapItem* MapArray;
```

est créée par le processus et alimentée d'un enregistrement à chaque création de nœud durant la première phase. À la fin de celle-ci, la table est triée, suivant un ordre quelconque choisi pour les quadruplés (par exemple, l'ordre lexicographique des catégories, des indices physiques des nœuds, des indices de produits et des périodes temporelles). Tout en assurant l'utilisation d'un espace mémoire minimal correspondant au nombre exact de nœuds du réseau espace-temps, l'extraction des références durant la seconde phase se fait de façon efficace par une recherche binaire au coût

moyen de $\Theta(\log n)$, où n est le nombre de nœuds du réseau.

L'algorithme de construction de la représentation réseau, dont la figure 4.3 donne une représentation schématique, consiste donc en une première phase où, sur la base d'un fichier de données, représentant le modèle mathématique, sont créés et sauvegardés dans un fichier intermédiaire les nœuds du réseau espace-temps correspondant ; une table de correspondance est également formée et sauvegardée dans un fichier. Les fichiers intermédiaires de la première phase, ainsi que les données pertinentes du modèle mathématique, sont consultés par la seconde phase qui crée les arcs et fournit un fichier résultat contenant une représentation réseau complète du problème.

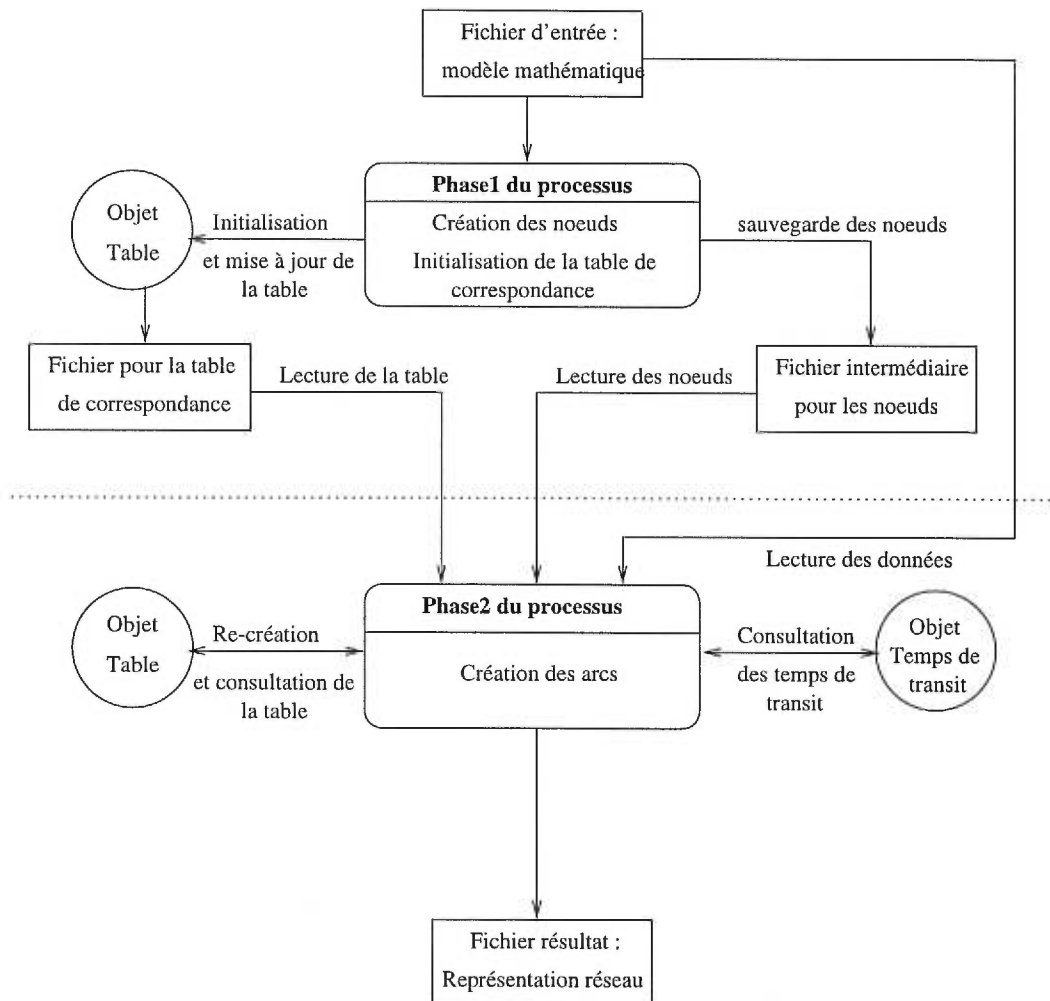


FIG. 4.3 – *Algorithme de construction de la représentation réseau*

Nous donnons en annexe, en guise d'illustration, un exemple sous forme d'un petit problème d'allocation sur un horizon de planification composé de 4 périodes et faisant

intervenir 4 ports, 3 dépôts terrestres, 5 clients et 3 produits différents, pour lequel nous présentons les fichiers NFF représentant respectivement les données du modèle mathématique et celles de la formulation réseau correspondante.

Chapitre 5

Résultats numériques

Ce dernier chapitre est consacré à la présentation des implantations séquentielles et parallèles de certaines de nos stratégies de décomposition, ainsi qu'à des tests numériques conduits sur la base d'échantillons de problèmes générés aléatoirement. Les résultats obtenus sont ensuite analysés et permettent de tirer des conclusions sur l'efficacité des approches proposées.

5.1 Objectifs

L'objectif premier de l'expérimentation que nous avons réalisée est de prouver l'intérêt de notre approche de décomposition par rapport à une application directe d'un algorithme de flot à coût minimum à la représentation réseau du problème, qui servira de point de référence. Nous espérons par ailleurs que l'analyse des résultats numériques obtenus puisse nous inciter à reconnaître l'importance fondamentale de la propriété de substitution et son impact sur la complexité du problème.

Dans cette optique, nous allons essayer à travers ces tests de répondre principalement aux questions suivantes :

- Quelle est l'origine du gain de temps CPU réalisé grâce à des implantations purement séquentielles des approches de restriction, si gain il y a ? D'autre part, quel impact a un changement de la structure générale des problèmes sur l'efficacité des stratégies ?
- Y a-t-il des stratégies plus efficaces que d'autres pour des problèmes particuliers ; notamment que se passe-t-il pour des problèmes à structure de substitution spéciale, par exemple quand les substitutions entre certains produits sont privilégiées par rapport à d'autres ?
- Quel est le potentiel de parallélisation des différentes approches ? Comment se comparent entre elles les versions parallèles ?

5.2 La génération aléatoire de problèmes test

Pour y arriver, nous aurons besoin de produire des échantillons variés de problèmes test, de façon à cerner le plus grand nombre de situations possibles. Bien qu'à cette fin il soit tout à fait possible d'utiliser un générateur standard de réseaux, il nous a paru préférable de mettre au point notre propre générateur de problèmes d'allocation de conteneurs. Deux considérations penchent en faveur de ce choix : d'abord, il n'est pas évident de produire directement, à l'aide d'un générateur à cadre général, des réseaux ayant la structure de notre modèle spécifique, et d'autre part, il est souhaitable d'avoir le contrôle précis des paramètres afin de diversifier la nature des réseaux

produits.

Le principe général du générateur de problèmes test que nous proposons est de modéliser, à l'aide de variables aléatoires très simples, la structure du réseau physique de distribution, ainsi que les activités d'allocation, de retour des conteneurs vides, d'inventaire aux dépôts, d'équilibrage inter-dépôts et de substitution. Notre but n'est pas de suppléer, par l'intermédiaire de ce générateur, au plan stratégique et tactique, ni même à un simulateur de scénarios plus complet. Cependant, une génération élémentaire nous semble tout à fait suffisante pour tester convenablement, dans le contexte déterministe qui est le nôtre, les approches de résolution que nous avons développées.

En ce qui concerne l'utilisation du générateur pour la production des séries de problèmes test de notre expérimentation, elle obéit à trois impératifs : varier les paramètres généraux associés au réseau physique de distribution (longueur de l'horizon de planification, nombre de produits, nombres de clients, de ports et de dépôts) et qui sont directement reliés à la taille des problèmes, contrôler la complexité des problèmes et la densité des réseaux représentatifs en agissant surtout sur l'intensité des activités d'allocation et de repositionnement de conteneurs vides et des opérations de substitution, et enfin, pouvoir générer des problèmes avec des structures de substitution spéciales.

5.2.1 Paramètres de la génération

Nous regroupons les paramètres de la génération aléatoire de problèmes test, selon leur fonctionnalité, en quatre catégories principales :

Paramètres généraux associés au réseau physique

Dans cette classe, nous intégrons les variables aléatoires représentant la longueur de l'horizon de planification, les nombres de clients, de ports, de dépôts terrestres, des types de conteneurs utilisés, ainsi que la matrice des temps de transit. Toutes ces variables, résumées dans le tableau 5.1, ont des distributions discrètes uniformes, pour lesquelles nous spécifions un intervalle de valeurs possibles.

Var. aléatoire	Description	Fonction de distribution
\tilde{T}	longueur de l'horizon de planification	discrète uniforme sur $[T_{min}, T_{max}]$
\tilde{C}	nombre de clients	discrète uniforme sur $[C_{min}, C_{max}]$
\tilde{H}	nombre de ports	discrète uniforme sur $[H_{min}, H_{max}]$
\tilde{D}	nombre de dépôts terrestres	discrète uniforme sur $[D_{min}, D_{max}]$
\tilde{P}	nombre de produits	discrète uniforme sur $[P_{min}, P_{max}]$
$\tilde{\tau}$	matrice des temps de transit	chaque élément $\tilde{\tau}_{ij}$ a une distribution discrète uniforme sur $[\tau_{min}, \tau_{max}]$

TAB. 5.1 – Les variables aléatoires associées à la structure physique du réseau

Paramètres reliés à l'activité de ports et des dépôts terrestres

Nous qualifions un dépôt d'*actif* à une période et pour un type de conteneurs donnés si, à cette période et pour ce type de conteneurs, le dépôt en question peut réaliser une opération (allocation ou récupération de conteneurs vides, import/export ou implication dans un mouvement d'équilibrage).

Nous définissons donc, en premier lieu, deux variables aléatoires \tilde{H}^{pt} et \tilde{D}^{pt} désignant respectivement les ensembles de ports et de dépôts terrestres actifs pour le type de conteneurs p , durant la période t . Nous construisons ces ensembles de la manière la plus simple possible consistant à générer à chaque période et pour chaque type de conteneurs les nombres de dépôts actifs et à choisir ensuite, de façon tout à fait arbitraire, ces dépôts parmi les éléments de H et de D .

Une fois que les dépôts actifs seront choisis, nous avons besoin de décrire leurs activités. À part les opérations d'allocation et de récupération de conteneurs vides, qui sont associées aux clients et que nous verrons plus loin, un dépôt peut :

- avoir un stock initial, et maintenir en inventaire un certain nombre de conteneurs. Nous définissons donc la quantité de conteneurs disponibles au début de l'horizon \tilde{w}_j^0 à chaque dépôt j et à chaque dépôt actif d , nous associons un coût unitaire d'inventaire \tilde{c}_{inv}^d ;
- être à l'origine, ou recevoir un flux d'équilibrage. Nous décidons par conséquent, pour tout dépôt actif d , à l'aide d'une variable aléatoire binaire \tilde{B}_{equ}^d , s'il y aura

un flux d'équilibrage en partance de d et nous définissons un ensemble de dépôts actifs destinataires \tilde{D}_{equ}^d ; pour chaque lien ainsi construit entre d et $d' \in \tilde{D}_{equ}^h$, nous générons une valeur $\tilde{c}_{equ}^{dd'}$ pour le coût unitaire correspondant.

Quand il s'agit de dépôts portuaires, nous avons également la possibilité d'avoir de l'import/export de conteneurs, activités pour lesquelles nous définissons, pour chaque port actif h , deux variables binaires \tilde{B}_{imp}^h et \tilde{B}_{exp}^h décidant s'il y a import/export et deux variables aléatoires uniformes \tilde{X}_{imp}^h et \tilde{X}_{exp}^h des nombres de conteneurs impliqués. Il existe enfin des paramètres additionnels sous forme de variables aléatoires décrivant la possibilité, pour un dépôt actif donné, d'avoir une demande d'export non satisfaite, ou encore de se procurer de nouveaux conteneurs.

Le tableau 5.2 résume les paramètres le plus importants de l'activité des dépôts :

Var. aléatoire	Description	Fonction de distribution
\tilde{H}^{pt}	ensemble de ports actifs	sous-ensemble arbitraire de H
\tilde{D}^{pt}	ensemble de dépôts actifs	sous-ensemble arbitraire de D
\tilde{w}_j^0	stock initial au dépôt j	discrète uniforme sur $[w_{min}^0, w_{max}^0]$
\tilde{c}_{inv}^d	coût unitaire d'inventaire au dépôt d	uniforme sur $[c_{min}^{inv}, c_{max}^{inv}]$
\tilde{B}_{equ}^d	d est-il l'origine d'un flux d'équilibrage?	Bernoulli de paramètre $P(\tilde{B}_{equ}^d)$
\tilde{D}_{equ}^d	ensemble des destinations du flux d'équilibrage	sous-ensemble arbitraire de $\tilde{H}^{pt} \cup \tilde{D}^{pt}$
$\tilde{c}_{equ}^{dd'}$	coût unitaire d'équilibrage	uniforme sur $[c_{min}^{equ}, c_{max}^{equ}]$
\tilde{B}_{imp}^h	h réalise t-il une opération d'import?	Bernoulli de paramètre $p(\tilde{B}_{imp}^h)$
\tilde{X}_{imp}^h	volume de conteneurs importés	discrète uniforme sur $[X_{min}^{imp}, X_{max}^{imp}]$
\tilde{B}_{exp}^h	h réalise t-il une opération d'export?	Bernoulli de paramètre $p(\tilde{B}_{exp}^h)$
\tilde{X}_{exp}^h	demande d'export	discrète uniforme sur $[X_{min}^{exp}, X_{max}^{exp}]$

TAB. 5.2 – Les variables aléatoires associées à l'activité des dépôts

Paramètres reliés à l'allocation et la récupération des conteneurs vides

De façon similaire aux dépôts, nous définissons un client *actif* comme étant un client ayant une demande ou initiant un mouvement de retour de conteneurs vides,

durant une période et pour un type de conteneurs donnés et nous introduisons naturellement une variable aléatoire \tilde{I}^{pt} désignant l'ensemble des clients actifs pour le produit p et la période t .

Un client actif i réalisant un ensemble \tilde{R}^i de requêtes d'offre ou de demande, nous devons établir la correspondance entre chaque requête $r \in \tilde{R}^i$ et un ensemble de dépôts actifs \tilde{D}^r , susceptibles de satisfaire la demande de conteneurs vides ou de recueillir les conteneurs vides disponibles (avec éventuellement une fenêtre temporelle pour la livraison $\tilde{\Delta}^r$ lorsqu'il s'agit de requêtes de demande) et deux variables aléatoires \tilde{X}_{off}^r et \tilde{X}_{dem}^r qui représentent les valeurs de l'offre ou de la demande correspondantes. Enfin, \tilde{c}_{all}^d et \tilde{c}_{ret}^d représentent respectivement les coûts unitaires des flux d'allocation et de retour faisant intervenir un dépôt $d \in \tilde{D}^r$ en réponse à la requête r .

Var. aléatoire	Description	Fonction de distribution
\tilde{I}^{pt}	ensemble de clients actifs	sous-ensemble arbitraire de C
\tilde{r}^i \tilde{R}^i	nombre de requêtes du client actif i ensemble de requêtes du client actif i	discrète uniforme sur $[r_{min}, r_{max}]$ $\tilde{R}^i = \{1, 2, \dots, \tilde{r}^i\}$
\tilde{X}_{off}^r \tilde{X}_{dem}^r $\tilde{\Delta}^r$	valeur de l'offre de la requête r valeur de la demande de la requête r fenêtre temporelle de livraison associée à la requête de demande r	discrète uniforme sur $[X_{min}^{off}, X_{max}^{off}]$ discrète uniforme sur $[X_{min}^{dem}, X_{max}^{dem}]$ discrète uniforme sur $[\Delta_{min}, \Delta_{max}]$
\tilde{D}^r	ensemble de dépôts actifs pouvant servir la requête r	ensemble arbitraire de dépôts actifs
\tilde{c}_{all}^d \tilde{c}_{ret}^d	coût unitaire du flux d'allocation coût unitaire du flux de retour	uniforme sur $[c_{min}^{all}, c_{max}^{all}]$ uniforme sur $[c_{min}^{ret}, c_{max}^{ret}]$

TAB. 5.3 – Les variables aléatoires associées à l'allocation et au retour des conteneurs

Paramètres de la substitution

En premier lieu, nous générons une matrice aléatoire des facteurs de substitution entre les paires de produits. La procédure de génération doit être capable de produire non seulement des matrices purement aléatoires pour les problèmes généraux sans aucune caractéristique spéciale de substitution, mais aussi, pour les besoins de certains tests que nous allons conduire, des matrices où sont favorisées des substitutions particulières entre certains types de conteneurs.

Il est également nécessaire d'intervenir localement, au niveau de chaque dépôt actif $d \in \tilde{H}^{pt} \cup \tilde{D}^{pt}$, afin de spécifier s'il y aura substitution, et quels seront dans ce cas les produits impliqués. Ceci est réalisé grâce à une variable binaire \tilde{B}_{sub}^d , suivant une distribution de Bernoulli, de paramètre $p(\tilde{B}_{sub}^d)$ et, au cas où il y aurait substitution, un ensemble arbitraire \tilde{P}^d de produits pouvant se substituer à p et un coût unitaire \tilde{c}_{sub}^{pr} pour chaque produit $r \in \tilde{P}^d$.

5.3 Résultats des versions séquentielles

Dans un premier temps, nous nous limiterons à deux approches de restriction séquentielles qui sont des instances particulières de l'algorithme générique proposé au chapitre précédent. Ces deux approches partagent par conséquent une première phase commune où l'on résout, l'un à la suite de l'autre, les sous-problèmes interdépendants obtenus en opérant une restriction sur tous les arcs de substitution. En revanche, et durant la seconde phase, la manière de relaxer ces restrictions diffère d'une approche à l'autre. La première stratégie, que nous désignons par *fusion globale*, opère simplement une relaxation de toutes les contraintes additionnelles du problème restreint, en une seule itération, puis résout le problème global. La seconde, quant à elle, a une seconde phase plus élaborée qui consiste à :

Initialisation

Définir un ensemble S de produits. $S = \{p_0\}$ où $p_0 \in P$ est un produit quelconque.

Boucle principale de la seconde phase

Tant que $S \neq P$ faire

$S = S \cup p$ où $p \in P - S$ est un produit quelconque non encore choisi.

Former le réseau $G^S = [N^S, A^S]$ où $N^S \subset N$ désigne le sous-ensemble des nœuds associés aux produits de S et $A^S = \{a = (i, j) : i \in N^S, j \in N^S\}$.

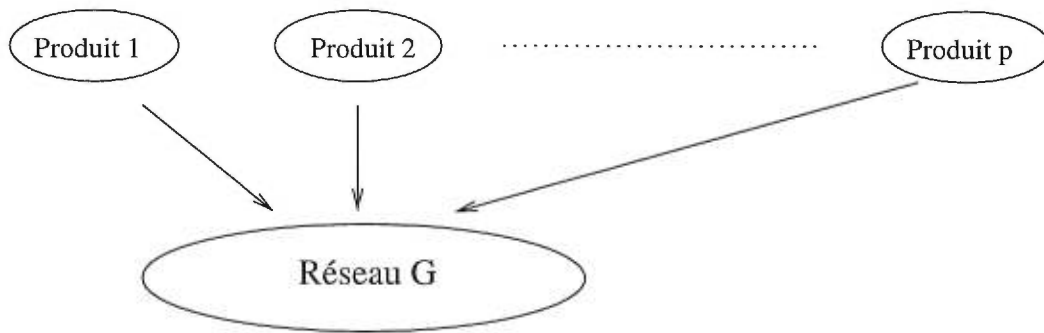
Résoudre le problème de flot à coût minimum sur G^S .

Fin

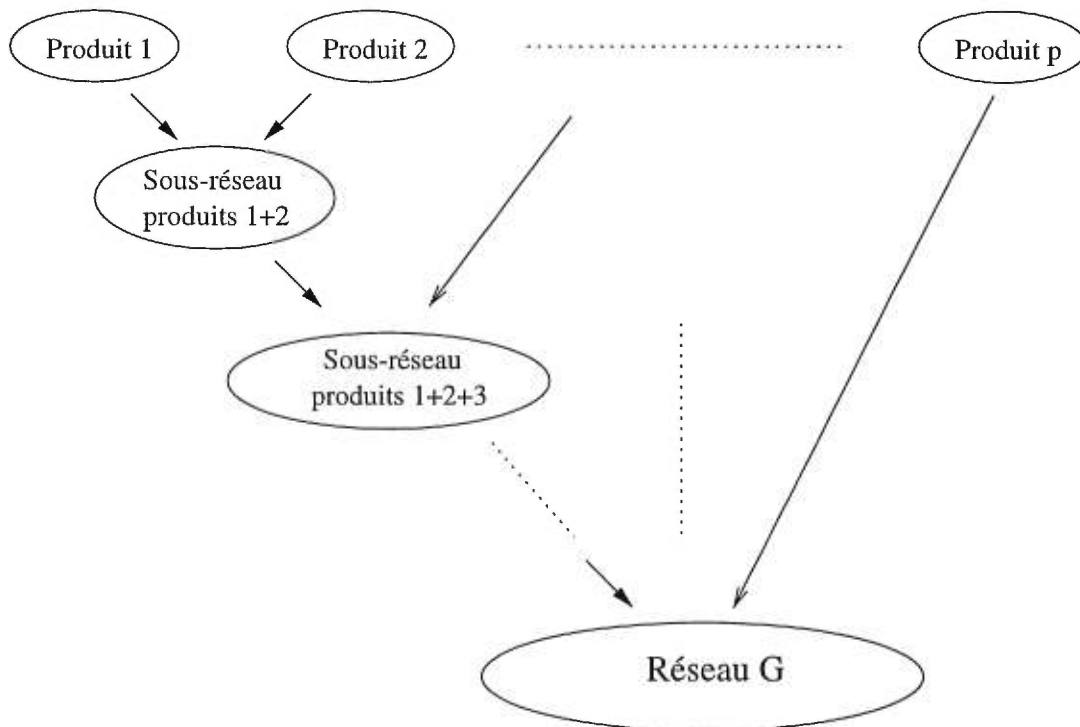
Intuitivement, cette approche semble, à chaque itération de la boucle principale

de sa seconde phase, “connecter” un sous-réseau correspondant à un produit donné au réseau intermédiaire G^S ; et au bout des $|P| - 1$ itérations de cette boucle, G^S n’est autre que le réseau G . Pour cette raison, nous appellerons cette stratégie la *fusion progressive à une étape*, et les itérations de la boucle *étapes* de la seconde phase.

La figure 5.1 illustre schématiquement la manière de procéder des deux approches de fusion.



Stratégie de fusion globale



Stratégie de fusion progressive à une étape

FIG. 5.1 – La seconde phase des deux premières stratégies de fusion

Nous avons réalisé l'implantation de ces deux stratégies en C++, sur des stations de travail Sun SPARC Ultra avec 32MB de mémoire vive, en utilisant le compilateur de Sun. L'algorithme de base de flot à coût minimum dans les réseaux généralisés utilisé est une implantation que nous avons réalisée de l'algorithme du simplexe réseau de Elam, Glover et Klingman [27] et qui fait appel à la règle simple de Dantzig pour le choix des pivots.

Les premiers tests ont été effectués sur un échantillon de problèmes de petite taille (série S1 du tableau 5.4). Bien que trop petits pour permettre des conclusions significatives, ces tests nous aideront à tirer quelques enseignements élémentaires sur le comportement des deux approches de décomposition par rapport à la stratégie de base.

Problème	Nombre de périodes	Nombre de produits	Nombres totaux de (ports, dépôts, clients)	Nombres totaux de (nœuds, arcs)	Nombre total d'arcs de substitution
S1R00	7	3	(5, 5, 10)	(436, 1000)	8
S1R01	7	3	(5, 10, 20)	(488, 1390)	9
S1R02	7	3	(10, 10, 20)	(556, 1518)	8
S1R03	7	3	(10, 20, 20)	(695, 1693)	12
S1R04	7	3	(15, 20, 40)	(910, 1921)	12
S1R05	7	3	(20, 30, 70)	(1097, 2164)	19

TAB. 5.4 – La série S1 de problèmes de petite taille

Sur la base de ce premier échantillon, nous testons d'une part une application brute du simplexe réseau généralisé à la représentation réseau des problèmes, qui sera désignée par *stratégie de base*, et d'autre part, les approches de fusion globale et de fusion progressive à une étape.

Les résultats obtenus sont résumés dans le tableau 5.5 :

Problème	Stratégie de base		Fusion globale		Fusion progressive à une étape	
	Nbr. pivots	CPU (s)	Nbr. pivots	CPU (s)	Nbr. pivots	CPU (s)
S1R00	372	1.32	365	0.54	365	0.54
S1R01	514	2.55	500	1.11	500	1.12
S1R02	552	3.03	539	1.06	539	1.06
S1R03	605	3.57	602	1.39	602	1.36
S1R04	956	6.39	944	2.72	944	2.59
S1R05	992	7.26	981	2.81	981	2.73

TAB. 5.5 – Résultats comparatifs pour la première série de problèmes

Dans le tableau 5.5, les nombres de pivots effectués et les temps CPU des deux phases ont été agrégés afin de pouvoir les comparer à ceux de la stratégie de base. Il est important toutefois, pour raffiner notre analyse des résultats, de donner les détails des calculs durant chaque phase.

Pour les deux approches de fusion, nous reportons (voir les tableaux 5.6 et 5.7) les nombres de pivots effectués et les temps CPU du simplexe sur les sous-réseaux correspondant aux différents produits. En seconde phase, nous notons, pour la stratégie de fusion globale, les résultats du simplexe sur le réseau global issu de la fusion, et pour la fusion progressive à une étape les résultats partiels du simplexe sur chaque réseau intermédiaire de la seconde phase.

		S1R00		S1R01		S1R02	
		Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
Phase 1		108	0.10	118	0.13	162	0.32
		101	0.12	190	0.39	187	0.33
		113	0.14	130	0.23	164	0.26
Phase 2	fusion	43	0.18	62	0.36	26	0.15

		S1R03		S1R04		S1R05	
		Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
Phase 1		184	0.41	302	0.65	253	0.52
		172	0.30	243	0.52	326	0.77
		205	0.40	273	0.63	334	0.97
Phase 2	fusion	41	0.28	126	0.92	68	0.55

TAB. 5.6 – Détails de la stratégie de fusion globale pour la première série

		S1R00		S1R01		S1R02	
		Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
Phase 1		108	0.10	118	0.13	162	0.32
		101	0.12	190	0.39	187	0.33
		113	0.14	130	0.23	164	0.26
Phase 2	étape 1	0	0.00	0	0.00	11	0.40
	étape 2	43	0.17	62	0.36	15	0.10

		S1R03		S1R04		S1R05	
		Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
Phase 1		184	0.41	302	0.65	253	0.52
		172	0.30	243	0.52	326	0.77
		205	0.40	273	0.63	334	0.97
Phase 2	étape 1	0	0.00	60	0.28	25	0.12
	étape 2	41	0.27	66	0.49	43	0.36

TAB. 5.7 – Détails de la stratégie de fusion progressive à une étape pour la première série

Les résultats du tableau 5.5 indiquent une nette supériorité, en termes de temps CPU, des approches de fusion par rapport à la stratégie de base. Le rapport d'efficacité

$$r = \frac{\text{Temps CPU de la stratégie de base}}{\text{Temps CPU des stratégies de fusion}} \quad (5.1)$$

a une valeur moyenne de 2.5 et son évolution semble promettre un gain de performance encore plus marqué sur des problèmes de plus grande taille. Nous ne relevons néanmoins aucune différence significative dans les nombres totaux de pivots effectués. La question qui vient à l'esprit est donc de savoir quelle est l'origine du gain en temps de calcul réalisé.

L'examen des détails des calculs reportés dans les tableaux 5.6 et 5.7 fournit une indication. Si nous prenons le cas du problème S1R00 comme exemple, nous observons que 108 pivots ont été réalisés sur le sous-réseau correspondant au premier produit, au coût de 0.1 secondes alors qu'après la fusion, 0.18 secondes ont été nécessaires pour effectuer 43 pivots! L'origine du gain provient donc de la décomposition faite durant la première phase où les pivots nécessitent moins de temps de calcul, ce qui est le résultat de l'application du simplexe sur des structures de données (notamment les listes des arcs candidats à l'entrée) de moindre taille.

Cette remarque présume que les méthodes de fusion progressive seraient plus efficaces que la fusion globale. Bien que sur la base des problèmes de la première série, ce fait soit presque toujours confirmé (à l'exception du problème S1R03), nous ne pouvons tirer des conclusions à ce sujet sur la base des résultats de la première série et nous attendons des tests sur des problèmes de plus grande taille.

Forts de cette constatation, nous allons enrichir notre analyse, pour les besoins des tests qui vont venir, en introduisant une nouvelle stratégie progressive de fusion, que nous appelons *fusion progressive binaire*. Sans donner de présentation formelle de cette stratégie, nous énonçons le principe de la seconde phase qui lui est associée et qui est de réaliser des fusions séparées faisant intervenir exactement deux réseaux intermédiaires. Ce principe est illustré par la figure 5.2, dans le cas d'un nombre pair de produits.

Il convient de noter que le choix de faire la fusion d'exactly deux sous-réseaux n'est pas obligé et qu'en général, il est possible de varier les stratégies en réalisant des fusions de $s \in \{2, \dots, p\}$ sous-réseaux. À cet égard, les méthodes de fusion globale et de fusion progressive binaire sont des cas extrêmes, avec respectivement $s = p$ et $s = 2$. Ces deux stratégies sont aussi des stratégies dites "*symétriques*", par rapport aux sous-réseaux impliqués dans les fusions, par contraste à la fusion progressive à une étape où le réseau intermédiaire G^R joue un rôle spécial.

Nous produisons une seconde série de problèmes à structure générale, de taille moyenne (tableau 5.8):

Problème	Nombre de périodes	Nombre de produits	Nombres totaux de (ports, dépôts, clients)	Nombres totaux de (nœuds, arcs)	Nombre total d'arcs de substitution
S2R00	7	8	(20, 50, 100)	(3674, 6288)	56
S2R01	7	8	(20, 50, 150)	(3630, 6418)	75
S2R02	7	8	(20, 50, 200)	(3902, 7162)	81
S2R03	7	8	(40, 50, 200)	(5463, 8270)	114
S2R04	7	8	(50, 100, 300)	(7635, 10430)	148

TAB. 5.8 – La série S2 de problèmes généraux de taille moyenne

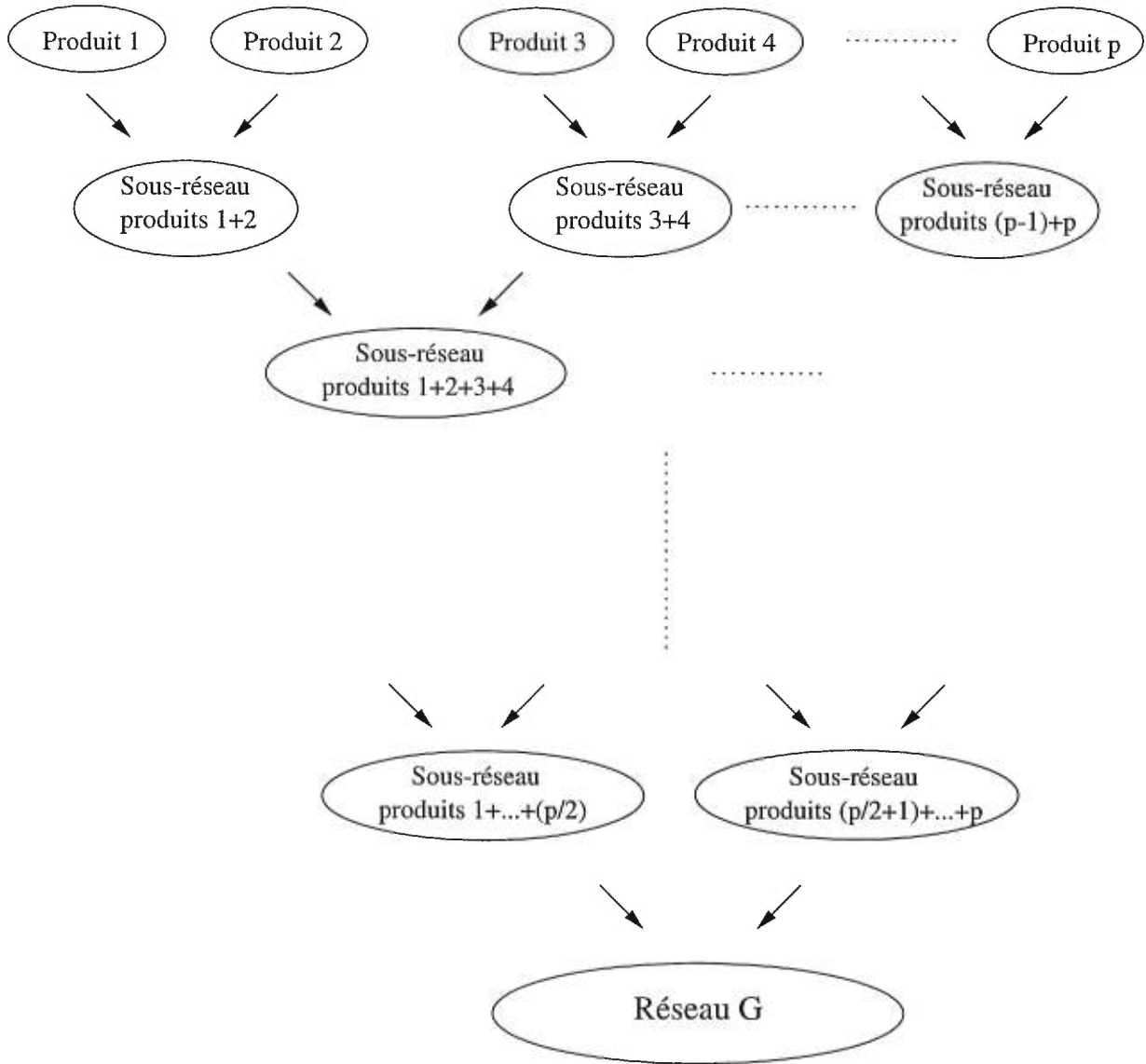


FIG. 5.2 – La seconde phase de la stratégie de fusion progressive binaire

Pour cette série, nous obtenons les résultats suivants (tableau 5.9) :

Problème	Stratégie de base		Fusion globale		Fusion progressive à une étape		Fusion progressive binaire	
	Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
S2R00	3360	71.13	3328	11.13	3328	10.87	3328	10.90
S2R01	3515	76.82	3486	15.45	3487	14.27	3482	12.87
S2R02	3885	95.19	3763	18.84	3763	17.84	3789	16.91
S2R03	5121	143.14	5083	28.49	5082	26.31	5068	24.61
S2R04	6717	238.79	6677	34.62	6677	33.40	6681	33.12

TAB. 5.9 – Résultats comparatifs pour la seconde série de problèmes

Les résultats obtenus confirment la tendance générale du rapport d'efficacité r à augmenter lorsque la taille des problèmes croît. En effet, la moyenne de ce rapport pour la fusion globale est de 5.67 sur les cinq problèmes de la série, et cette moyenne est encore plus élevée dans le cas des fusions progressives. Nous retenons également que les résultats obtenus avec la fusion progressive binaire sont comparables à ceux de la fusion progressive à une étape et que toutes deux dépassent toujours en efficacité la fusion globale. En fait, sur presque la totalité des problèmes, exception faite de S2R00, la fusion binaire surpasse, quoique de façon très peu significative, la fusion progressive à une étape.

Il serait utile, maintenant que l'intérêt général de nos approches de décomposition par rapport à la stratégie de base a été établi, de voir le comportement de ces approches vis-à-vis des caractéristiques du problème. À cet égard, la propriété de substitution nous paraît être de la plus grande importance. Ainsi, et dans le but de mettre en évidence le rôle particulier des arcs de substitution, nous générons deux nouvelles séries de problèmes. La première (tableau 5.10) est constituée de problèmes à structure générale, mais avec une activité accrue des dépôts et une présence plus marquée des clients, ce qui a principalement pour résultat d'augmenter, de façon significative, la densité des réseaux associés, par rapport à celle des problèmes de la série S2. La seconde série (tableau 5.11) est un ensemble de problèmes avec une activité normale des dépôts et des clients, mais avec des possibilités de substitution très supérieures, effet que l'on obtient en augmentant le paramètre $p(\tilde{B}_{sub}^d)$ de la procédure de génération aléatoire des problèmes.

Problème	Nombre de périodes	Nombre de produits	Nombres totaux de (ports, dépôts, clients)	Nombres totaux de (nœuds, arcs)	Nombre total d'arcs de substitution
S3R00	7	8	(20, 50, 200)	(5618, 17447)	81
S3R01	7	8	(40, 50, 200)	(7550, 20790)	114
S3R02	7	8	(40, 50, 200)	(10767, 40083)	114
S3R03	7	8	(40, 50, 300)	(15630, 69994)	118

TAB. 5.10 – La série S3 de problèmes généraux denses

Problème	Nombre de périodes	Nombre de produits	Nombres totaux de (ports, dépôts, clients)	Nombres totaux de (nœuds, arcs)	Nombre total d'arcs de substitution
S4R00	7	8	(40, 50, 200)	(5349, 8406)	346
S4R01	7	8	(50, 100, 300)	(7607, 10659)	414
S4R02	7	8	(50, 100, 300)	(7788, 11243)	620
S4R03	7	8	(50, 100, 300)	(7836, 11869)	931

TAB. 5.11 – La série S4 de problèmes à substitution intense

Les résultats obtenus sont comme suit (tableau 5.12) :

Problème	Stratégie de base		Fusion globale		Fusion progressive à une étape		Fusion progressive binaire	
	Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
S3R00	6451	423.51	6094	106.20	6126	104.34	6117	83.71
S3R01	9119	717.97	9098	240.55	9128	198.67	9118	209.50
S3R02	13538	2131.04	13310	791.56	13254	691.29	13576	692.77
S3R03	16198	+1 heure	15406	887.20	15440	731.95	15418	763.14
S4R00	4954	141.19	4842	28.79	4841	26.42	4841	26.45
S4R01	7580	277.02	7523	62.90	7519	54.16	7531	53.51
S4R02	8065	314.49	7914	79.24	8000	69.22	8058	67.06
S4R03	8808	366.25	8788	114.74	8500	84.86	8642	86.05

TAB. 5.12 – Les résultats des problèmes denses et à activité de substitution intense

L'examen du rapport de performance des méthodes de fusion par rapport à la stratégie de base, pour les deux catégories de problèmes, nous permet de tirer un certain nombre d'enseignements. La moyenne de ce rapport, pour la fusion globale sur les trois premiers problèmes "denses" de la série S3, est de 3.2, valeur qui est très inférieure à celle obtenue pour les problèmes moins denses de la série S2 (5.67). Ceci infirme la remarque précédente que l'efficacité relative des approches de fusion augmente toujours avec la taille des problèmes, et ce, en imposant la densité comme autre facteur important à tenir en considération.

En ce qui concerne ce rapport d'efficacité relative pour les problèmes de la série S4, pour laquelle la seule différence avec les problèmes de la série S2 réside dans la présence d'une activité de substitution plus intense, la moyenne relevée (4.12) est également plus basse que la normale et tend à diminuer avec l'augmentation relative du nombre d'arcs de substitution. Afin de mieux comprendre ce phénomène, nous donnons dans le tableau 5.13 les détails de la fusion globale pour les problèmes S2R04 et S4R03. Remarquons en premier lieu que le nombre de pivots qui s'effectuent durant la seconde phase, pour le problème S4R03 (2086 pivots) est beaucoup plus élevé que ce nombre pour le problème S2R04 (222 pivots), bien que les premières phases dans les deux problèmes soient très comparables. En fait, pour le problème S4R03, 23.74% des pivots sont effectués après la fusion, et coûtent approximativement 74.43% du temps CPU de résolution total, à comparer avec seulement 3.30% des pivots en 23.17% du temps CPU durant la seconde phase du problème S2R04.

		S2R04		S4R03	
		Pivots	CPU (s)	Pivots	CPU (s)
Phase 1		884	3.86	931	4.54
		781	3.01	819	3.81
		842	3.88	871	3.72
		933	4.20	937	4.70
		806	3.27	772	3.36
		767	3.05	815	3.14
		690	2.37	743	2.76
		752	2.96	814	3.31
Phase 2	fusion	222	8.02	2086	85.40

TAB. 5.13 – *Détails comparatifs de la fusion globale pour les problèmes S2R04 et S4R03*

Cette remarque indique qu'avec la taille des problèmes et leur densité, l'intensité de la substitution est également un paramètre qui conditionne de manière très significative la complexité des problèmes et l'efficacité relative des approches de décomposition proposées.

Finalement, nous examinons le comportement des approches de fusion dans le cas de structures de substitution très particulières. Nous choisissons, à cet effet, d'intervenir au niveau de la génération aléatoire de la matrice de substitution afin de favoriser certaines associations entre des produits spécifiques.

Dans un premier temps, nous produisons une série S5 de problèmes où la substitution entre les couples de produits $\{(2l + 1, 2l + 2) : l \geq 0, 2l + 2 \leq p\}$ est très favorisée. Par exemple, la matrice de substitution

$$S_1 = \begin{pmatrix} -1 & 0.33 & -1 & -1 & -1 & -1 & -1 & -1 \\ 3.00 & -1 & -1 & -1 & -1 & -1 & 0.87 & -1 \\ -1 & -1 & -1 & 0.50 & -1 & -1 & -1 & -1 \\ -1 & -1 & 2.01 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & 0.18 & -1 & -1 \\ -1 & -1 & -1 & -1 & 5.54 & -1 & -1 & -1 \\ -1 & 1.15 & -1 & -1 & -1 & -1 & -1 & 0.83 \\ -1 & -1 & -1 & -1 & -1 & -1 & 1.20 & -1 \end{pmatrix}$$

sera utilisée pour produire les problèmes S5R00 et S5R01 présentés dans le tableau 5.14, alors que la matrice

$$S_2 = \begin{pmatrix} -1 & 0.33 & -1 & -1 & -1 & -1 & -1 & -1 \\ 3.00 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 0.50 & -1 & -1 & -1 & -1 \\ -1 & -1 & 2.01 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & 0.18 & -1 & -1 \\ -1 & -1 & -1 & -1 & 5.54 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0.83 \\ -1 & -1 & -1 & -1 & -1 & -1 & 1.20 & -1 \end{pmatrix}$$

est utilisée lors de la génération de S5R02 et S5R03. Notons que la matrice S_2 a une structure de substitution $\{(2l + 1, 2l + 2) : l \geq 0, 2l + 2 \leq p\}$ *pure*, c'est à dire qu'aucune autre substitution, en dehors de ces associations, n'est permise. Remarquons également que les problèmes ainsi formulés sont naturellement décomposables en $\lceil \frac{p}{2} \rceil$ sous-problèmes indépendants.

Le même principe est mis en œuvre pour générer un autre échantillon de problèmes à structure de substitution différente. Cette fois, ce sont tout simplement les associations de substitution faisant intervenir *les trois premiers produits* qui sont privilégiées. Ainsi, la matrice S_3 est utilisée pour la génération des problèmes S6R00 et S6R01 et la matrice S_4 , à structure "pure", pour les problèmes S6R02 et S6R03.

$$S_3 = \begin{pmatrix} -1 & 1.29 & 5.51 & -1 & -1 & 8.79 & -1 & -1 \\ 0.77 & -1 & 2.75 & -1 & -1 & -1 & -1 & -1 \\ 0.18 & 0.36 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 0.11 & -1 & -1 & -1 & -1 & -1 & -1 & 0.29 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix}$$

et

$$S_4 = \begin{pmatrix} -1 & 1.29 & 5.51 & -1 & -1 & -1 & -1 & -1 \\ 0.77 & -1 & 2.75 & -1 & -1 & -1 & -1 & -1 \\ 0.18 & 0.36 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix}$$

Les deux séries de problèmes produits sont les suivantes (tableau 5.14) :

Problème	Nombre de périodes	Nombre de produits	Nombres totaux de (ports, dépôts, clients)	Nombres totaux de (nœuds, arcs)	Nombre total d'arcs de substitution
S5R00	7	8	(50, 100, 300)	(7703, 10743)	150
S5R01	7	8	(50, 100, 300)	(9831, 23511)	150
S5R02	7	8	(50, 100, 300)	(7843, 11030)	200
S5R03	7	8	(50, 100, 300)	(9831, 23490)	134
S6R00	7	8	(50, 100, 300)	(7933, 11196)	182
S6R01	7	8	(50, 100, 300)	(9588, 24037)	202
S6R02	7	8	(50, 100, 300)	(7559, 10456)	113
S6R03	7	8	(50, 100, 300)	(9763, 24381)	131

TAB. 5.14 – Les séries S5 et S6 de problèmes à structure particulière de substitution

Et nous obtenons les résultats suivants (tableau 5.15) :

Problème	Stratégie de base		Fusion globale		Fusion progressive à une étape		Fusion progressive binaire	
	Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)	Pivots	CPU (s)
S5R00	6953	251.87	6841	39.54	6841	37.39	6841	31.68
S5R01	10370	920.14	10287	234.58	10298	193.28	10298	156.16
S5R02	7512	283.79	7247	48.87	7247	41.54	7247	34.01
S5R03	9783	1070.64	9714	180.17	9714	152.75	9714	110.91
S6R00	7371	278.66	7187	42.52	7195	37.76	7195	36.49
S6R01	9901	884.80	9594	179.10	9759	147.48	9735	140.13
S6R02	6865	240.59	6715	34.92	6715	29.87	6711	30.85
S6R03	9720	869.60	9546	183.88	9849	129.99	9846	140.44

TAB. 5.15 – *Les résultats des problèmes à structure particulière de substitution*

Les résultats relatifs aux problèmes S5R00 et S5R01 montrent un avantage numérique plus marqué que d’habitude, en faveur de la fusion progressive binaire. À titre d’exemple, la fusion binaire est 18% plus efficace que la fusion progressive pour S5R00 et 23% plus efficace pour S5R01. Pour les réseaux à structure de substitution “pure”, ces chiffres atteignent 22% pour S5R02 et 37% pour S5R03. Ces gains très substantiels trouvent leur justification dans la façon avec laquelle la fusion est réalisée par l’approche de fusion binaire : en effet, il n’est pas difficile de voir que dans le cas des problèmes à structure particulière pure de substitution, la fusion binaire intègre tous les arcs de substitution durant la première étape de sa seconde phase, où justement on fusionne les réseaux correspondant aux produits $2l + 1$ et $2l + 2$ où l vérifie $l \geq 0, 2l + 2 \leq p$. Le reste de la seconde phase est alors complètement superflu puisque l’optimum a été trouvé à la fin de cette étape. En d’autres termes, la fusion binaire “voit” et exploite la “décomposabilité” des problèmes résultant de la structure spéciale de la substitution, ce que n’arrivent pas à faire les autres stratégies.

Une situation similaire, pour les mêmes raisons invoquées, se produit dans le cas des problèmes de la série S6, mais cette fois en faveur de la fusion progressive à une étape. Néanmoins, dans le cas des problèmes “presque” décomposables S6R00 et S6R01, les différences sont faibles et il arrive même, comme pour S6R00, que la fusion binaire soit supérieure. La raison que nous invoquons, et que confirment les détails des calculs de seconde phase, est que la fusion progressive à une étape est plus sensible à la présence des arcs de substitution durant les dernières étapes de sa seconde phase,

ce qui peut très bien se produire, comme c'est le cas avec la matrice de substitution S_3 .

5.3.1 Conclusion

Les implantations séquentielles des approches de décomposition du problème que nous avons proposées se sont avérées d'une grande efficacité par rapport à la stratégie de base, sur tous les problèmes tests utilisés lors de l'expérimentation. Nous avons également pu conclure que la taille, la densité des problèmes, ainsi que la propriété de substitution étaient des paramètres déterminants et pouvaient influencer grandement les performances. Enfin, nous avons montré que l'une ou l'autre des deux stratégies de fusion progressive pouvait être plus adaptée à des problèmes possédant des structures de substitution spéciales.

L'analyse détaillée que nous avons faite, par moments, des deux phases des stratégies de fusion et de leur poids relatif, nous semble extrêmement intéressante, en prévision du développement de versions parallèles. En effet, la résolution des sous-problèmes indépendants de flot à coût minimum qui intervient durant la première phase (qui, rappelons-le, est la même pour toutes les stratégies) rend sa parallélisation parfaitement triviale ; en revanche, la parallélisation de la seconde phase est moins évidente et le potentiel des différentes stratégies à cet égard, n'est pas le même.

5.4 Versions parallèles

Nous développons, dans cette partie, deux approches de parallélisation élémentaires, respectivement pour la fusion globale et pour la fusion progressive binaire. En les qualifiant d'élémentaires, nous signifions que ces approches seront basées exclusivement sur un parallélisme à *gros grain*. Autrement dit, nous nous limiterons à une exécution simultanée des traitements uniquement lorsqu'une identification nette de sous-problèmes indépendants est possible. En pratique, les algorithmes parallèles que nous proposons mettent en œuvre un mécanisme hiérarchique maître-esclaves, à base d'échanges de messages dans un environnement distribué.

5.4.1 Un algorithme parallèle pour la première phase

Le processus maître s'occupe uniquement de l'initialisation des esclaves, et de la réception synchrone, à la fin de la phase, des sous-structures optimales ; ainsi, si nous avons p produits différents, la partie maître du processus de la première phase correspond à :

Première phase - processus maître

Début

Initialiser p processus et allouer les tâches.

Boucle principale d'attente synchrone des résultats

Répéter

 Réception d'une sous-structure optimale d'un esclave.

Jusqu'à réception de p structures.

Fin

Chaque esclave construit simplement le sous-problème correspondant à sa tâche et le résout :

Première phase - processus esclave l , $l = 1, \dots, p$

Début

Construction du sous-problème correspondant au produit l .

Résolution du sous-problème.

Envoi de la solution optimale au maître.

Fin

5.4.2 Parallélisation de la seconde phase

En ce qui concerne la stratégie de fusion globale, la seconde phase est purement séquentielle et consiste en une fusion des sous-structures optimales obtenues de la part des esclaves, et l'utilisation de la solution réalisable qui en découle comme solution

de départ pour la résolution du problème global.

Pour ce qui est de la fusion progressive binaire, la seconde phase comprend, pour un problème à p produits, où $p > 2$, $\lceil \log_2(p) \rceil$ étapes (voir la figure 5.2). Si nous désignons par N_i le nombre de sous-problèmes au début d'une étape i , nous voyons facilement que $N_1 = p$ et que $\{N_i\}_{i=1, \dots, \lceil \log_2(p) \rceil}$ vérifie la récurrence $N_{i+1} = \lceil (\frac{N_i}{2}) \rceil$. Autrement dit, une étape i consiste en $\lfloor (\frac{N_i}{2}) \rfloor$ fusions et autant de résolutions de sous-problèmes.

L'algorithme est donc le suivant :

Fusion binaire, seconde phase - processus maître

Début

Pour chaque étape i , $i = 1, \dots, \lceil \log_2(p) \rceil$, faire
 Initialiser N_i processus esclaves;
 Envoi des sous-structures optimales aux esclaves;
 Boucle d'attente des résultats intermédiaires;
 Répéter
 Réception d'une sous-structure optimale d'un esclave.
 Jusqu'à réception de N_i structures.
 Fin de l'étape i .

Fin

Le rôle du maître est plus important que dans la première phase puisqu'il se charge également d'envoyer les sous-structures optimales aux processus subordonnés. D'autre part, les fins d'étape constituent des points de synchronisation de l'algorithme, où une attente de la réception des résultats des esclaves est nécessaire avant de passer à l'étape suivante.

Les esclaves se comportent sensiblement de la même façon que dans la première phase :

Fusion binaire, seconde phase - processus esclave l , $l = 1, \dots, N_i$

Début

Fusion des structures reçues et construction d'un sous-problème intermédiaire.

Résolution du sous-problème.

Envoi des solutions optimales au maître.

Fin

5.4.3 Tests sur quelques problèmes

Nous avons conduit quelques tests sur la base d'une implantation réalisée des deux algorithmes parallèles que nous venons de voir. L'objectif poursuivi est surtout de pouvoir comparer l'efficacité de ces algorithmes par rapport aux versions séquentielles correspondantes.

L'environnement matériel qui a servi de support à ces tests est un réseau de stations de travail Sun SPARC Ultra. Étant donné que tous les problèmes tests que nous avons sélectionnés étaient constitués de 8 produits, nous avons eu recours à un maximum de 9 stations de travail, une pour le processus maître et les autres pour les esclaves. Enfin, le logiciel PVM a servi à construire, à partir de ces stations, une machine parallèle virtuelle.

Dans le contexte parallèle, il est important de discuter de la mesure du temps de calcul adoptée. Ce problème ne se pose pas pour les versions séquentielles des algorithmes où le temps CPU est presque toujours utilisé. Par contre, il est difficile, surtout lorsque la synchronisation n'est pas triviale, ou encore que l'algorithme est asynchrone, de tirer des conclusions à base de ce temps dans une implantation parallèle où des processus s'activent simultanément, et où chacun d'eux a son propre temps CPU. En outre, dans un contexte de calcul distribué, les temps de communication sont significatifs, et doivent intervenir dans l'analyse. Bien qu'il soit possible de recourir au temps réel d'exécution, ce dernier ne fournit généralement pas une bonne indication car il comprend également les temps associés au service des programmes par le système d'exploitation (accès à la mémoire, entrées-sorties, etc.). Ceci nous

amène à utiliser plutôt une évaluation du temps basée uniquement sur les temps CPU et les délais de communication inter-processus.

Notre tâche est simplifiée par la synchronisation très simple mise en œuvre par nos algorithmes parallèles. En général, ces algorithmes se présentent, au niveau du maître, sous la forme d'une séquence $\{SEQ_i\}_{i=1,\dots,s}$ d'opérations à caractère séquentiel (initialisations, formation de structures, communication avec les esclaves, etc.), ou à caractère parallèle (attente de résolutions simultanées de sous-problèmes par les esclaves). Si SEQ_{i_0} est une opération parallèle donnée, composée par exemple de l sous-opérations simultanées $\{PAR_k^{i_0}\}, k = 1, \dots, l$, et $T(OP)$ le temps d'exécution réel d'une opération OP , alors $T(SEQ_{i_0}) = \max_{k=1,\dots,l} T(PAR_k^{i_0})$, et $T(Algorithme) = \sum_{i=1}^s T(SEQ_i)$. Ces formules indiquent simplement que le temps d'exécution total de l'algorithme est la somme des temps d'exécution des séquences, et qu'à chaque fois qu'une séquence se compose d'opérations simultanées, son temps d'exécution est le plus grand parmi les temps d'exécution de ces opérations.

Dans notre cas particulier, nous avons décidé de ne garder, dans notre évaluation, que les temps CPU des résolutions, et les temps de communication inter-processus. Ainsi, pour la fusion globale par exemple, nous avons l'approximation suivante du temps d'exécution total de l'algorithme

$$T(Algorithme) \approx \max_{l=1,\dots,p} T^{CPU}(SR_l) + T^{CPU}(global) + T(communication) \quad (5.2)$$

où T^{CPU} désigne le temps CPU et SR_l le processus de résolution par l'esclave correspondant au produit l , durant la première phase.

Nous utilisons cette approximation pour évaluer l'accélération obtenue grâce aux algorithmes parallèles. La définition de l'accélération (Bertsekas et Tsitsiklis [10]), pour n processeurs utilisés, est :

$$S(n) = \frac{T}{T_n} \quad (5.3)$$

où T désigne le temps d'exécution du meilleur algorithme séquentiel pour un problème donné et T_n le temps requis par l'algorithme parallèle. Afin de pouvoir évaluer le potentiel de parallélisation des deux approches de fusion choisies, T sera pour nous le temps requis pour la version séquentielle correspondante, plutôt que celui du meilleur algorithme séquentiel (d'ailleurs, nous ne connaissons pas cet algorithme!).

Les résultats comparatifs, ainsi que les accélérations correspondantes, sont données dans le tableau 5.16 :

Problème	Fusion Globale			Fusion progressive binaire		
	Version séquentielle	Version parallèle	Accélération	Version séquentielle	Version parallèle	Accélération
S2R00	11.13	4.44	2.51	10.90	4.58	2.38
S2R01	15.45	8.86	1.74	12.87	7.44	1.72
S2R02	18.84	10.55	1.79	16.91	9.88	1.71
S2R03	28.49	15.95	1.78	24.61	13.62	1.80
S2R04	34.62	12.97	2.67	33.12	13.99	2.36
S3R00	106.20	72.79	1.45	83.71	52.39	1.59
S3R01	240.55	187.73	1.28	209.50	159.11	1.32
S3R02	791.56	651.77	1.21	692.77	539.30	1.28
S3R03	887.20	515.13	1.72	763.14	388.30	1.96
S4R00	28.79	14.82	1.94	26.45	16.46	1.60
S4R01	62.90	41.05	1.53	53.51	32.08	1.66
S4R02	79.24	56.06	1.41	67.06	41.73	1.61
S4R03	114.74	90.80	1.26	86.05	59.44	1.44
S5R00	39.54	17.60	2.24	31.68	11.67	2.71
S5R01	234.58	160.84	1.45	156.16	72.80	2.15
S5R02	48.87	25.19	1.94	34.01	10.12	3.36
S5R03	180.17	106.55	1.69	110.91	39.20	2.83

TAB. 5.16 – Les résultats comparatifs des versions séquentielles et parallèles des approches de fusion

Sur l'ensemble de problèmes sans structure spéciale de substitution, constitué des séries S2, S3 et S4, les deux stratégies parallèles apportent les mêmes ordres d'accélération vis à vis les stratégies séquentielles correspondantes. La moyenne de 1.7, sur les 13 problèmes, est du reste très faible, dans la mesure où nous avons utilisé 9 stations de travail. Ceci n'est cependant pas surprenant, étant donné que les rapports d'importance en temps de calcul entre les deux phases, sur des problèmes à structure de substitution générale, penchent toujours en faveur de la seconde, qui

justement est purement séquentielle dans l'implantation parallèle de la fusion globale, et ne fait intervenir qu'un parallélisme très limité pour la fusion progressive binaire. Par ailleurs, le fait que, malgré son potentiel de parallélisation relativement supérieur, la fusion binaire n'apporte pas de meilleure accélération ne peut s'expliquer que par les communications plus coûteuses qu'elle nécessite.

La situation est différente pour la série de problèmes à structure spéciale S5, pour laquelle nous avons pu noter que, déjà, les versions séquentielles de la fusion binaire étaient favorisées. Ce constat est confirmé en ce qui concerne l'accélération, avec une moyenne qui se démarque aisément de celle de la fusion globale.

5.4.4 Conclusion

Les résultats des tests des algorithmes parallèles montrent les limitations de l'approche simpliste qui tente d'exploiter les propriétés du problème d'allocation afin de développer un parallélisme élémentaire à gros grain. Sans vouloir diminuer l'intérêt de ces approches, car avant tout, elles nous ont permis de mieux comprendre les propriétés numériques du problème et de mieux cerner ses difficultés, elles ne sont pas, à notre avis, à la hauteur des moyens matériels mis en œuvre. Toutefois, il est permis de penser que des efforts plus poussés dans le sens d'améliorer leur degré de sophistication puissent donner de meilleurs résultats à l'avenir.

Conclusion

Le but de ce mémoire était de concevoir des approches de résolution du problème de l'allocation déterministe, dynamique et multiproduits de conteneurs vides basées sur un principe de décomposition primale du problème. L'approche suggérée met en œuvre un processus classique de restriction en exploitant la structure multiproduits du problème et l'existence d'arcs de substitution.

Des implantations séquentielles et parallèles de trois variantes de notre algorithme générique de décomposition multiproduits ont été réalisées et des tests comparatifs ont pu être effectués sur un jeu de données du problème de l'allocation de conteneurs vides, soigneusement construit afin de couvrir un certain nombre de configurations différentes portant sur la taille des problèmes, la densité des réseaux espace-temps associés et l'intensité de la substitution. Les résultats de cette expérimentation démontrent le grand intérêt de notre approche de décomposition vis-à-vis d'une approche générale qui ne tient pas compte de la structure multiproduits. Il nous a été permis en outre de constater le rôle important que joue la propriété de substitution dans la complexité du problème et de prendre note que l'une ou l'autre des stratégies de résolution que nous avons proposées peut être spécialement adaptée à une structure de substitution particulière.

Les résultats des implantations parallèles que nous avons réalisées nous ont également permis de tirer un certain nombre d'enseignements utiles. Nous avons notamment retenu que les stratégies de parallélisation à forte granularité que nous mettons en œuvre apportaient certes un gain de performance par rapport aux versions séquentielles correspondantes, mais que l'utilisation des stations de travail était loin d'être optimale, notamment lors de la seconde phase des stratégies de décomposition

où le potentiel de parallélisation des traitements était inexistant ou du moins très faible. Nous avons suggéré les approches de parallélisme à granularité moyenne ou faible comme moyens de gagner en efficacité. Pour le simplexe réseau, ces approches concernent principalement l'exécution simultanée de plusieurs pivots (Chang, Engquist, Finkel et Meyer [13]) ainsi que l'exécution en parallèle des opérations d'un pivot, à savoir l'évaluation des coûts marginaux d'une part et d'autre part la mise à jour de la base et des variables primales et duales (Peters [51], Barr et Hickmann [5]).

Suite à ce travail, plusieurs avenues de recherche, d'ordre théorique et pratique, peuvent être envisagées. Dans ce sens, notons tout d'abord que nous avons considéré jusque-là des horizons de planification fixes, de l'ordre d'une semaine. Ceci est une approche simpliste qui, d'une part, limite sérieusement la taille des problèmes qu'il est possible de résoudre dans des temps raisonnables, et d'autre part, est irréaliste, car elle ne tient pas compte du fait que l'activité des compagnies de transport se poursuit en théorie indéfiniment, en tout cas bien au-delà d'un horizon de planification, aussi long qu'il puisse être. Cette remarque nous amène à la conclusion qu'il est indispensable soit d'intégrer dans le futur notre algorithme à un processus de planification à horizon fuyant, caractérisé par une fenêtre temporelle de planification, composée d'un certain nombre de périodes et qui se déplace dans le temps, et par des décisions qui sont matérialisées au jour le jour, soit de prendre en considération les effets de ce que l'on appelle la *fin du temps*, ou la fin de l'horizon de planification. En ce qui concerne cette seconde alternative, les techniques connues pour mettre un terme à la planification sur des horizons infinis se classent selon deux catégories :

- Les techniques fondées sur la restriction du modèle à une période temps limitée tout en imposant soit une pénalité, soit des contraintes additionnelles de capacité sur les effets de fin de l'horizon, ou en choisissant de façon empirique un horizon de planification suffisamment long de manière à assurer un niveau acceptable pour ces effets (Morton [47]).
- Les techniques basées sur l'approximation de l'état du système de distribution de la compagnie par un état d'équilibre stationnaire (Grinold [35]). La distinction est alors effectuée entre deux phases du système : une phase transitoire composée

de plusieurs périodes, suivie d'une phase stationnaire où le comportement du système semble régénérer sur une base périodique. Il est alors possible, grâce à cette approximation, de ramener la planification sur un horizon infini à une planification sur un horizon fini composé de la phase transitoire et d'une seule période représentative de la phase stationnaire.

Le développement d'une approche de décomposition duale, par exemple en relaxant les contraintes de conservation de flot aux sources et aux puits des arcs de substitution, semble également être une approche très intéressante au problème. Il n'est pas difficile de voir que les solutions optimales du problème relaxé ont des valeurs de flot de substitution nulles, et donc que le problème relaxé est décomposable en p sous-problèmes indépendants, où p est le nombre de produits. Étant donné que les contraintes de conservation de flots aux extrémités des arcs de substitution ont été relaxées, les sous-problèmes obtenus ne sont pas des problèmes de flots à coût minimum. Toutefois, et par une transformation très simple, illustrée par la figure 5.3, qui consiste à remplacer chaque nœud pour lequel la conservation de flot n'est pas requise par deux nœuds de transbordement, reliés par un arc à capacité infinie sur lequel l'excès ou le défaut de flot au nœud peut circuler à un coût unitaire nul; et il n'y a aucune difficulté à établir l'équivalence des deux problèmes. Nous croyons que l'implantation de cette approche duale, soit à travers la relaxation classique ou la relaxation lagrangienne, peut servir à notre approche de décomposition primale, à la fois de complément naturel et de base pour la comparaison des résultats.

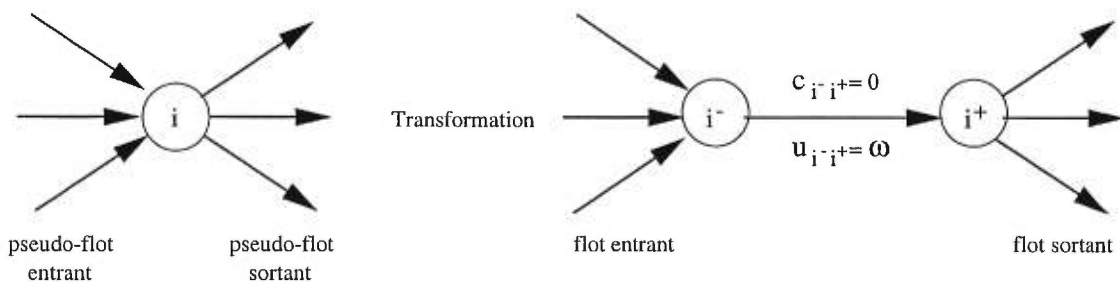


FIG. 5.3 – Transformation des sous-problèmes issus de la décomposition duale en problèmes de flot à coût minimum

Bibliographie

- [1] R.K. Ahuja, T.L. Magnanti et J.B. Orlin. *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, 1993.
- [2] J.E. Aronson. A Survey of Dynamic Network Flows. *Annals of Operations Research*, 20:1–66, 1989.
- [3] J.E. Aronson et B.D. Chen. A Forward Network Simplex Algorithm for Solving Multiperiod Network Flow Problems. *Naval Research Logistics Quarterly*, 33:445–467, 1986.
- [4] A.A. Assad. Models for Rail Transportation. *Transportation Research*, 14A:205–220, 1980.
- [5] R.S. Barr et B.L. Hickman. Parallel Simplex for Large Pure Network Problems : Computational Testing and Sources of Speedup. *Operations Research*, 42:65–80, 1994.
- [6] R.S. Barr, F. Glover et D. Klingman. Enhancements of Spanning Tree Labelling Procedures for Network Optimization. *INFOR*, 17:16–34, 1979.
- [7] M. Bazaraa, J.J. Jarvis et J.J. Sherali. *Linear Programming and Network Flows*. John Wiley & Sons, New York, 1990.
- [8] D.P. Bertsekas. *Linear Network Optimization: Algorithms and Codes*. MIT Press, Cambridge, 1991.
- [9] D.P. Bertsekas et P. Tseng. Relaxation Methods for Minimum Cost Ordinary and Generalized Network Flow Problems. *Operations Research*, 36:93–114, 1988.

- [10] D.P. Bertsekas et J.N. Tsitsiklis. *Parallel and Distributed Computation*. Prentice Hall, 1989.
- [11] G. Bradley, G. Brown et G. Graves. Design and Implementation of a Large Scale Primal Transshipment Algorithm. *Management Science*, 24:1–34, 1977.
- [12] G. Brown et R. McBride. Solving Generalized Networks. *Management Science*, 30:1497–1523, 1984.
- [13] M.D. Chang, Engquist M., R. Finkel et R.R. Meyer. A Parallel Algorithm for Generalized Networks. *Annals of Operations Research*, 14:125–145, 1988.
- [14] R.K.-M. Cheung et W.B. Powell. An Algorithm for Multistage Dynamic Networks with Random Arc Capacities, with an Application to Dynamic Fleet Management. *Operations Research*, 44, 1996.
- [15] K. C.-K. Chih. *A Real Time Dynamic Optimal Freight Car Management Simulation Model of the Multiple Railroad, Multicommodity Temporal Spatial Network Flow Problem*. Thèse de doctorat, Princeton University, 1986.
- [16] T.G. Crainic. *Un modèle de planification tactique pour le transport ferroviaire de marchandises*. Thèse de doctorat, Université de Montréal, 1982.
- [17] T.G. Crainic, P. Dejax et L. Delorme. Models for Multimode Multicommodity Location Problems with Interdepot Balancing Requirements. *Annals of Operations Research*, 18:279–302, 1989.
- [18] T.G. Crainic, P. Dejax et M. Gendreau. *Transport Terrestre Multimodal de Conteneurs : Problématiques et Modèles*, 1989. Publication du CRT # 656.
- [19] T.G. Crainic et L. Delorme. Dual-Ascent Procedures for Multicommodity Location Allocation Problems with Balancing Requirements. *Transportation Science*, 27B:90–101, 1993.
- [20] T.G. Crainic, M. Gendreau et P. Dejax. Dynamic and Stochastic Models for the Allocation of Empty Containers. *Operations Research*, 41:102–126, 1993.

- [21] T.G. Crainic et G. Laporte. Planning Models for Freight Transportation. *European Journal of Operations Research*, 409–438, 1997.
- [22] T.G. Crainic et J. Roy. O.R. Tools for Tactical Freight Transportation Planning. *European Journal of Operational Research*, 33:290–297, 1988.
- [23] W.H. Cunningham. Theoretical Properties of the Network Simplex Method. *Mathematics of Operations Research*, 4:196–208, 1979.
- [24] G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1962.
- [25] P. Dejax et T.G. Crainic. A Review of Empty Flows and Fleet Management Models in Freight Transportation. *Transportation Science*, 21:227–247, 1987.
- [26] P. Dejax, T.G. Crainic et L. Delorme. Strategic/Tactical Planning of Container Land Transportation Systems. TIMS XXVIII/EURO IX Joint Meeting, Paris, Juillet 1988.
- [27] J. Elam, F. Glover et D. Klingman. A Strongly Convergent Algorithm for Generalized Networks. *Mathematics of Operations Research*, 4:34–59, 1979.
- [28] J.M. Farvolden et W.B. Powell. A Dynamic Network Model for Less-Than-Truckload Motor Carrier Operations, 1991. Working Paper 90-05, Department of Industrial Engineering, University of Toronto.
- [29] M. Florian. An Introduction to Network Models Used in Transportation Planning. In M. Florian, editor, *Transportation Planning Models*, pages 137–152. North-Holland, Amsterdam, 1984.
- [30] L. Frantzeskakis et W.B. Powell. A Successive Linear Approximation Procedure for Stochastic Dynamic Vehicle Allocation Problems. *Transportation Science*, 24:40–57, 1990.
- [31] B. Gendron et T.G. Crainic. A Branch-and-Bound Algorithm for Depot Location and Container Fleet Management. *Location Science*, 3A:39–53, 1995.

- [32] F. Glover, D. Klingman et J. Stutz. Augmented Threaded Index Method for Network Optimization. *INFOR*, 2:293–298, 1974.
- [33] A.V. Goldberg, S.A. Plotkin et É. Tardos. Combinatorial Algorithms for the Generalized Circulation Problem. *Mathematics of Operations Research*, 16:351–381, 1991.
- [34] M.D. Grigoriadis. An Efficient Implementation of the Network Simplex Method. *Mathematical Programming Studies*, 26:83–111, 1986.
- [35] R. Grinold. Model Building Techniques for the Correction of End Effects in Multistage Convex Programs. *Operations Research*, 31:407–431, 1983.
- [36] J. Guélat, M. Florian, et T.G. Crainic. A Multimode Multiproduct Network Assignment Model For Strategic Planning of Freight Flows. *Transportation Science*, 24:25–39, 1990.
- [37] A.E. Haghani. Formulation and Solution of Combined Train Routing and Makeup, and Empty Car Distribution Model. *Transportation Research*, 23B:433–452, 1989.
- [38] R.V. Helgason, J.L. Kennington et H.A. Zaki. A Parallelization of the Simplex Method. *Annals of Operations Research*, 14:17–40, 1988.
- [39] P. Jensen et G. Bhaumik. A Flow Augmentation Approach to the Network with Gains Minimum Cost Flow Problem. *Management Science*, 23:631–643, 1977.
- [40] W.S. Jewell. Optimal Flows through Networks with Gains. *Operations Research*, 10:496–499, 1962.
- [41] W.C. Jordan et M.A. Turnquist. A Stochastic Dynamic Network Model for Railroad Car Distribution. *Transportation Science*, 17:123–145, 1983.
- [42] J.L. Kennington et R.V. Helgason. *Algorithms for Network Programming*. John Wiley & Sons, New York, 1980.

- [43] L.S. Lasdon. *Optimization Theory for Large Systems*. Macmillan Series in Operations Research, 1972.
- [44] D. Lebeuf. *Contributions au Simplexe primal adapté au réseau et implantations séquentielles et parallèles efficaces*. Thèse de doctorat, Département d'Informatique et de Recherche Opérationnelle - Université de Montréal, 1996.
- [45] C. Leddon et E. Wrathall. Scheduling Empty Car Fleets on the Louisville and Nashville Railroad. In *Second International Symposium on the Use of Cybernetics on the Railroads*, pages 1–6, Octobre 1967.
- [46] C.S. Misra. Linear Programming of Empty Wagon Disposition. *Rail International*, 3:151–158, 1972.
- [47] T. Morton. Infinite Horizon Dynamic Programming Models: A Planning Horizon Formulation. *Operations Research*, 27:730–742, 1979.
- [48] K.G. Murty. *Network Programming*. Prentice Hall, 1992.
- [49] J.B. Orlin. On the Simplex Algorithm for Networks and Generalized Networks. *Mathematical Programming Study*, 24:166–178, 1985.
- [50] G.P. Ouimet. Empty Freight Car Distribution. Mémoire de maîtrise, Queen's University, 1972.
- [51] J. Peters. The Network Simplex Method on a Multiprocessor. *Networks*, 20:845–859, 1990.
- [52] W.B. Powell. A Stochastic Model of the Dynamic Vehicle Allocation Problem. *Transportation Science*, 20:117–129, 1986.
- [53] W.B. Powell. An Operational Planning Model for the Dynamic Vehicle Allocation Problem. *Transportation Research*, 21B:217–232, 1987.
- [54] W.B. Powell. A Comparative Review of Alternative Algorithms for the Dynamic Vehicle Allocation Problem. In B.L. Golden and A.A. Assad, editors, *Vehicle Routing: Methods and Studies*, pages 249–292. North-Holland, Amsterdam, 1988.

- [55] W.B. Powell et T.A. Carvalho. Dynamic Control of Logistics Queueing Networks for Large Scale Fleet Management. Rapport technique SOR-96-01, Department of Civil Engineering and Operations Research - Princeton University, Janvier 1996.
- [56] W.B. Powell, T.A. Carvalho, G.A. Godfrey et H.P. Simaõ. Optimal Control Methods for Logistics Queueing Networks. In *TRISTAN II, TRIennial Symposium on Transportation ANalysis*, pages 185–199, 1994. préimprimé.
- [57] W.B. Powell, P. Jaillet et A. Odoni. Stochastic and Dynamic Networks and Routing. In M. Ball, T.L. Magnanti, C.L. Monma, and G.L. Nemhauser, editors, *Handbooks in Operations Research and Management Science - Network Routing*, volume 8, pages 141–295. North-Holland, Amsterdam, 1995.
- [58] W.B. Powell, Y. Sheffi et S. Thiriez. The Dynamic Vehicule Allocation Problem with Incertain Demands. In *Ninth International Symposium on Transportation and Traffic Theory*, pages 357–374, 1984.
- [59] M. Resende et G. Veiga. An Efficient Implementation of a Network Interior Point Method. Rapport technique, AT & T Bell Laboratories, 1992.
- [60] J. Roy. *Un modèle de planification globale pour le transport routier de marchandises*. Thèse de doctorat, Ecole des Hautes Etudes Commerciales, Université de Montréal, 1984.
- [61] H. Steffan. Étude de la demande de conteneurs vides à court terme, Octobre 1989. Publication du CRT # 651.
- [62] C. Vasseur. Les coûts du transport ferroviaire par conteneurs en France : étude de cas, Octobre 1989. Publication du CRT # 650.
- [63] W. White et A. Bomberault. A Network Algorithm for Empty Freight Car Allocation. *IBM Systems Journal*, 8:147–171, 1969.

Annexe A

Un exemple de fichier de données pour le modèle mathématique

```
<title> Data </title>
```

```
<section LengthHorizon>
```

```
  <structure>  
    long Value;  
  </structure>
```

```
  <data>  
    4;  
  </data>
```

```
</section>
```

```
<section Commodities>
```

```
  <structure>  
    long Id;  
  </structure>
```

```
  <data>  
    1;  
    2;  
    3;  
  </data>
```

```
</section>
```

```
<section Ports>
```

```
  <structure>  
    long Id;  
  </structure>
```

```
  <data>
```

```
    1;
    2;
    3;
    4;
</data>
</section>

<section Depots>
  <structure>
    long Id;
  </structure>
  <data>
    1;
    2;
    3;
  </data>
</section>

<section Customers>
  <structure>
    long Id;
  </structure>
  <data>
    1;
    2;
    3;
    4;
    5;
  </data>
</section>

<section RequestsByPeriodAndCommodity>
  <structure>
    long Period;
    ref Commodities:Id Commodity;
    (ref) DemandCustomersRequests:Id SetOfDemandRequests;
    (ref) SupplyCustomersRequests:Id SetOfSupplyRequests;
  </structure>
  <data>
    1, 1, (1,2), (1);
    1, 3, (), (2,3);
    2, 2, (3), (4);
```

```
    3, 1, (4), ();
    3, 2, (5,6), ();
    4, 2, (7), (5);
</data>
</section>

<section DemandCustomersRequests>
  <structure>
    long Id;
    ref Customers:Id RequestByCustomer;
    long DeliveryWindow;
    float ValueOfDemand;
    float VolumeSentBeforeHorizon;
  </structure>
  <data>
    1, 2, 0, 2.0, 0.0;
    2, 1, 0, 5.0, 0.0;
    3, 5, 1, 2.0, 1.0;
    4, 1, 0, 3.0, 0.0;
    5, 3, 0, 5.0, 1.0;
    6, 4, 0, 3.0, 0.0;
    7, 2, 0, 5.0, 0.0;
  </data>
</section>

<section SupplyCustomersRequests>
  <structure>
    long Id;
    ref Customers:Id RequestByCustomer;
    float ValueOfSupply;
  </structure>
  <data>
    1, 1, 5.0;
    2, 2, 3.0;
    3, 1, 1.0;
    4, 5, 10.0;
    5, 4, 5.0;
  </data>
</section>

<section SetsOfPotentiallySatisfiedRequests>
  <structure>
    long Period;
```

```

    ref Commodities:Id Commodity;
    short PortOrInlandDepot;
    ref PortsInService:Id ServiceByPort;
    ref DepotsInService:Id ServiceByDepot;
    (ref) AllocationFlowLinks:Id SetOfAllocationFlowLinks;
    (ref) PickupFlowLinks:Id SetOfPickupFlowLinks;
</structure>
<data>
    1, 1, 0, 1, 1, (1), ();
    1, 1, 0, 5, 1, (2), (1);
    2, 1, 0, 3, 1, (), (2);
    2, 1, 0, 6, 1, (3), ();
    1, 2, 1, 1, 1, (4,5), ();
    2, 2, 1, 1, 2, (), (3);
    3, 2, 1, 1, 3, (6,7), ();
    4, 2, 1, 1, 4, (), (4);
    1, 3, 0, 9, 1, (), (5,6);
</data>
</section>

<section PortsInService>
    <structure>
        long Id;
        long Period;
        ref Commodities:Id Commodity;
        ref Ports:Id Port;
        float ExportDemand;
        float ImportSupply;
        float StockAvailableAtTheBeginning;
        float VolumeSentBeforeHorizon;
        ref InventoryFlowLinks:Id InventoryFlowLink;
        ref NewFlowLinks:Id NewFlowLink;
        ref UnsatisfiedExportFlowLinks:Id UnsatisfiedExportFlowLink;
        ref EquivalentFlowLinks:Id EquivalentFlowLink;
        (ref) BalancingFlowLinks:Id SetOfBalancingFlowLinks;
        (ref) SubstitutionFlowLinks:Id SetOfSubstitutionFlowLinks;
    </structure>
    <data>
        1, 1, 1, 1, 0.0, 1.0, 3.0, 2.0, 1, 1, 0, 1, (), ();
        2, 2, 1, 1, 4.0, 0.0, 3.0, 2.0, 2, 2, 0, 2, (), ();
        3, 3, 1, 1, 0.0, 1.0, 3.0, 2.0, 3, 0, 1, 3, (2), ();
        4, 4, 1, 1, 3.0, 0.0, 3.0, 2.0, 0, 0, 2, 4, (), (1);
    </data>
</section>

```



```

5, 1, 1, 2, 0.0, 1.0, 3.0, 0.0, 4, 0, 0, 5, (1), ();
6, 2, 1, 2, 1.0, 0.0, 3.0, 0.0, 5, 0, 0, 6, (), ();
7, 3, 1, 2, 0.0, 1.0, 3.0, 0.0, 6, 3, 0, 7, (), ();
8, 4, 1, 2, 0.0, 1.0, 3.0, 0.0, 0, 0, 0, 8, (), ();
9, 1, 3, 1, 0.0, 1.0, 5.0, 2.0, 10, 0, 0, 17, (), ();
10, 2, 3, 1, 4.0, 0.0, 5.0, 2.0, 11, 0, 0, 18, (), ();
11, 3, 3, 1, 0.0, 1.0, 5.0, 2.0, 12, 0, 0, 19, (), ();
12, 4, 3, 1, 3.0, 0.0, 5.0, 2.0, 0, 0, 0, 20, (), ();
13, 1, 3, 4, 0.0, 1.0, 5.0, 2.0, 7, 4, 0, 9, (), ();
14, 2, 3, 4, 4.0, 0.0, 5.0, 2.0, 8, 0, 3, 10, (), ();
15, 3, 3, 4, 0.0, 1.0, 5.0, 2.0, 9, 0, 0, 11, (), ();
16, 4, 3, 4, 3.0, 0.0, 5.0, 2.0, 0, 0, 4, 12, (), ();
</data>
</section>

<section DepotsInService>
  <structure>
    long Id;
    long Period;
    ref Commodities:Id Commodity;
    ref Depots:Id Depot;
    float StockAvailableAtTheBeginning;
    float VolumeSentBeforeHorizon;
    ref InventoryFlowLinks:Id InventoryFlowLink;
    ref NewFlowLinks:Id NewFlowLink;
    ref EquivalentFlowLinks:Id EquivalentFlowLink;
    (ref) BalancingFlowLinks:Id SetOfBalancingFlowLinks;
    (ref) SubstitutionFlowLinks:Id SetOfSubstitutionFlowLinks;
  </structure>
  <data>
    1, 1, 2, 3, 3.0, 2.0, 10, 0, 13, (), ();
    2, 2, 2, 3, 3.0, 2.0, 11, 5, 14, (), ();
    3, 3, 2, 3, 3.0, 2.0, 12, 6, 15, (), ();
    4, 4, 2, 3, 3.0, 2.0, 0, 0, 16, (), ();
  </data>
</section>

<section AllocationFlowLinks>
  <structure>
    long Id;
    ref DemandCustomersRequests:Id DemandPotentiallySatisfied;
    float UnitCost;

```

```
    float Volume;
</structure>
<data>
  1, 2, 1.0, 0.0;
  2, 1, 1.0, 0.0;
  3, 4, 1.0, 0.0;
  4, 3, 1.0, 0.0;
  5, 7, 1.0, 0.0;
  6, 5, 1.0, 0.0;
  7, 6, 1.0, 0.0;
</data>
</section>

<section PickupFlowLinks>
  <structure>
    long Id;
    ref DemandCustomersRequests:Id SupplyPotentiallyPickedUp;
    float UnitCost;
    float Volume;
  </structure>
  <data>
    1, 1, 1.0, 0.0;
    2, 1, 1.0, 0.0;
    3, 4, 1.0, 0.0;
    4, 5, 1.0, 0.0;
    5, 3, 1.0, 0.0;
    6, 2, 1.0, 0.0;
  </data>
</section>

<section InventoryFlowLinks>
  <structure>
    long Id;
    float UnitCost;
    float Volume;
  </structure>
  <data>
    0, 0.0, 0.0;
    1, 1.0, 0.0;
    2, 1.0, 0.0;
    3, 1.0, 0.0;
    4, 1.0, 0.0;
    5, 1.0, 0.0;
    6, 1.0, 0.0;
```

```
    7, 1.0, 0.0;
    8, 1.0, 0.0;
    9, 1.0, 0.0;
    10, 1.0, 0.0;
    11, 1.0, 0.0;
    12, 1.0, 0.0;
    13, 1.0, 0.0;
    14, 1.0, 0.0;
    15, 1.0, 0.0;
</data>
</section>

<section NewFlowLinks>
  <structure>
    long Id;
    float UnitCost;
    float Volume;
  </structure>
  <data>
    0, 0.0, 0.0;
    1, 1.0, 0.0;
    2, 1.0, 0.0;
    3, 1.0, 0.0;
    4, 1.0, 0.0;
    5, 1.0, 0.0;
    6, 1.0, 0.0;
  </data>
</section>

<section InsatisfiedExportFlowLinks>
  <structure>
    long Id;
    float UnitCost;
    float Volume;
  </structure>
  <data>
    0, 0.0, 0.0;
    1, 1.0, 0.0;
    2, 1.0, 0.0;
    3, 1.0, 0.0;
    4, 1.0, 0.0;
  </data>
</section>
```

```
<section EquivalentFlowLinks>
  <structure>
    long Id;
    float Volume;
  </structure>
  <data>
    1, 0.0;
    2, 0.0;
    3, 0.0;
    4, 0.0;
    5, 0.0;
    6, 0.0;
    7, 0.0;
    8, 0.0;
    9, 0.0;
    10, 0.0;
    11, 0.0;
    12, 0.0;
    13, 0.0;
    14, 0.0;
    15, 0.0;
    16, 0.0;
    17, 0.0;
    18, 0.0;
    19, 0.0;
    20, 0.0;
  </data>
</section>

<section BalancingFlowLinks>
  <structure>
    long Id;
    short PortOrInlandDepot;
    ref Ports:Id Port;
    ref Depots:Id Depot;
    float UpperBound;
    float LowerBound;
    float UnitCost;
    float Volume;
  </structure>
  <data>
    1, 0, 1, 1, 5.0, 0.0, 1.0, 0.0;
    2, 0, 2, 1, 5.0, 0.0, 1.0, 0.0;
```

```

    </data>
</section>

<section SubstitutionFlowLinks>

    <structure>
        long Id;
        ref Commodities:Id CommodityOfTheDestination;
        float SubstitutionFactor;
        float UpperBound;
        float UnitCost;
        float Volume;
    </structure>

    <data>
        1, 3, 2.0, 4.0, 1.0, 0.0;
    </data>
</section>

<section TransitTimes>

    <structure>
        short TypeOfOrigin;
        short TypeOfDestination;
        ref Customers:Id CustomerOrigin;
        ref Ports:Id PortOrigin;
        ref Depots:Id DepotOrigin;
        ref Customers:Id CustomerDestination;
        ref Ports:Id PortDestination;
        ref Depots:Id DepotDestination;
        long Value;
    </structure>

    <data>
        0, 1, 1, 1, 1, 1, 1, 1, 1;
        0, 1, 1, 1, 1, 1, 2, 1, 0;
        0, 1, 1, 1, 1, 1, 4, 1, 0;
        0, 1, 2, 1, 1, 1, 4, 1, 0;
        0, 2, 4, 1, 1, 1, 1, 3, 0;
        0, 2, 5, 1, 1, 1, 1, 3, 0;
        1, 0, 1, 1, 1, 1, 1, 1, 0;
        1, 0, 1, 2, 1, 1, 1, 1, 1;
        1, 0, 1, 2, 1, 2, 1, 1, 0;
        1, 1, 1, 1, 1, 1, 2, 1, 1;
        1, 1, 1, 2, 1, 1, 1, 1, 2;
        2, 0, 1, 1, 3, 2, 1, 1, 3;
        2, 0, 1, 1, 3, 3, 1, 1, 0;

```

```
    2, 0, 1, 1, 3, 4, 1, 1, 0;  
    2, 0, 1, 1, 3, 5, 1, 1, 1;  
  </data>  
</section>  
  
</nff>
```

Annexe B

La représentation réseau correspondante

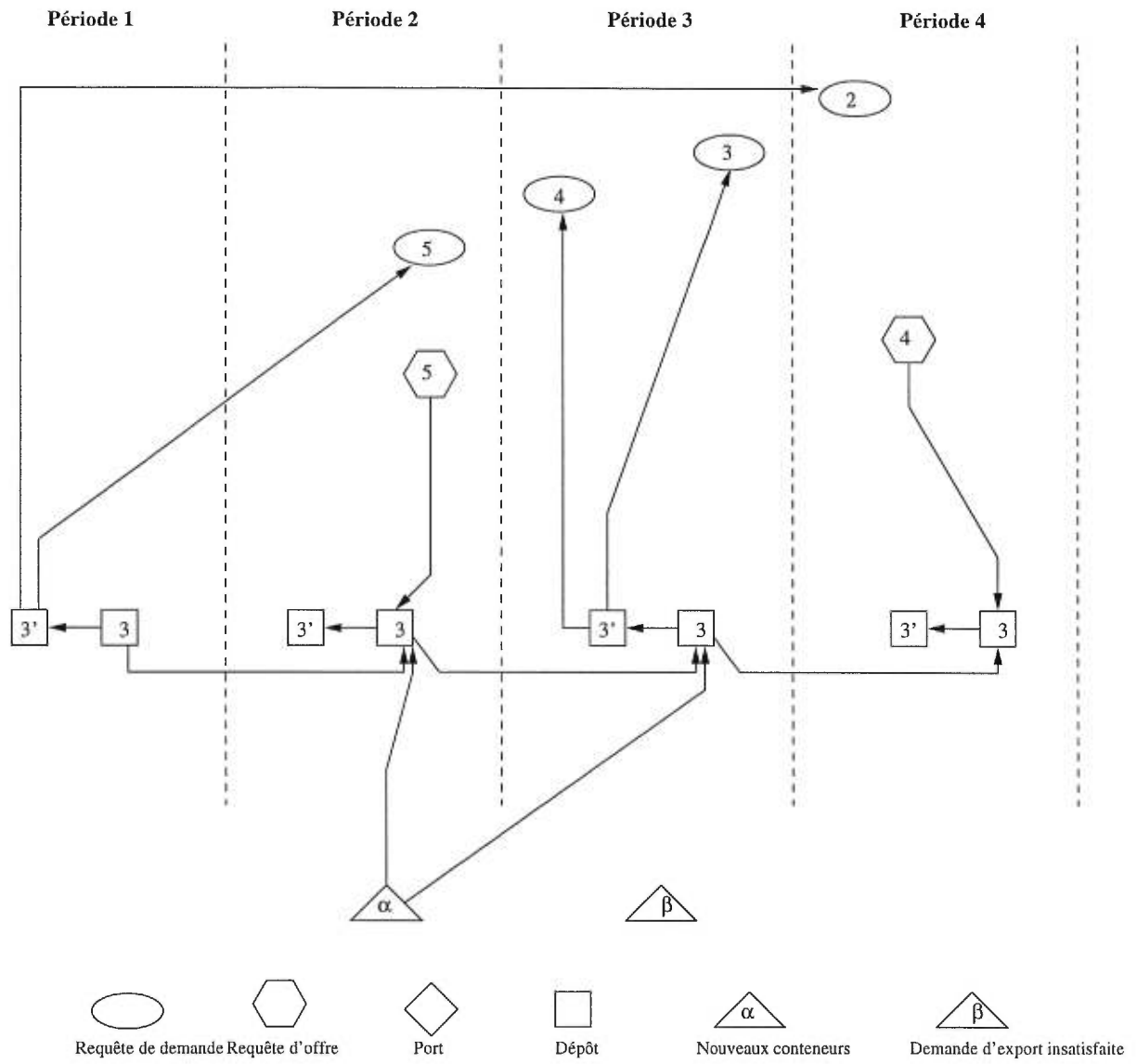


FIG. B.2 – Le sous-réseau du produit 2

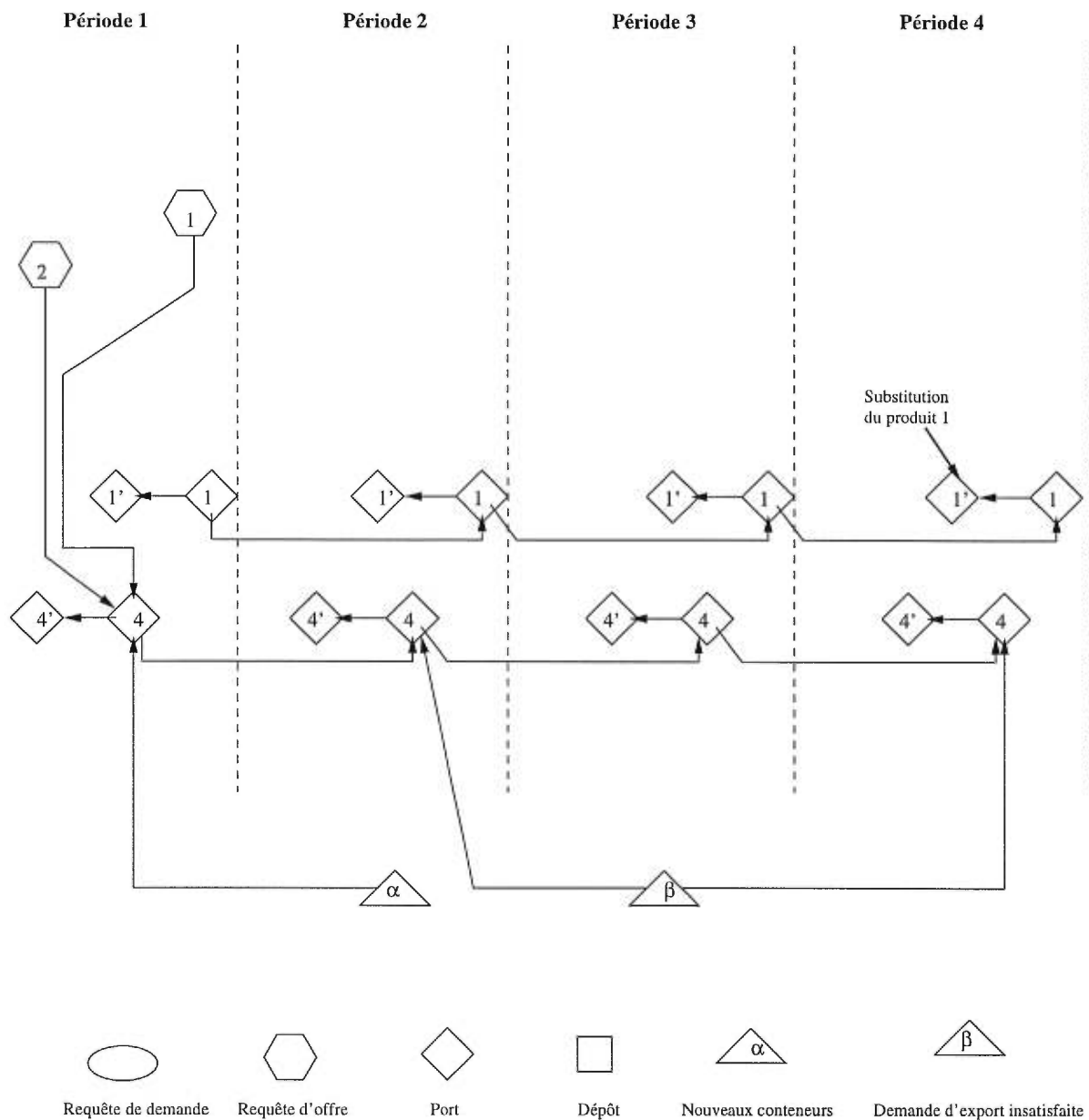


FIG. B.3 – *Le sous-réseau du produit 3*

Et finalement le fichier NFF correspondant :

```
<nff 3.0>
<section Commodities>
<structure>
  long Id;
</structure>
<data 3>
1;
2;
3;
</data>
</section>
<section Nodes>
<structure>
  long Id;
  ref Commodities:Id Commodity;
  float SupplyDemand;
</structure>
<data 58>
0,1,-2;
1,1,-5;
2,1,5;
3,3,3;
4,3,1;
5,2,-1;
6,2,10;
7,1,-3;
8,2,-4;
9,2,-3;
10,2,-5;
11,2,5;
12,1,6;
13,1,0;
14,1,-2;
15,1,0;
16,1,3;
17,1,0;
18,1,-1;
19,1,0;
20,1,4;
21,1,0;
22,1,-1;
23,1,0;
24,1,1;
25,1,0;
26,1,1;
```

```
27,1,0;
28,3,8;
29,3,0;
30,3,-2;
31,3,0;
32,3,3;
33,3,0;
34,3,-1;
35,3,0;
36,3,8;
37,3,0;
38,3,-2;
39,3,0;
40,3,3;
41,3,0;
42,3,-1;
43,3,0;
44,2,5;
45,2,0;
46,2,2;
47,2,0;
48,2,2;
49,2,0;
50,2,2;
51,2,0;
52,1,1e+07;
53,1,1e+07;
54,2,1e+07;
55,2,1e+07;
56,3,1e+07;
57,3,1e+07;
</data>
</section>
<section Arcs>
<structure>
  long Id;
  short Substitution;
  ref Commodities:Id CommodityOrigin;
  ref Nodes:Id Origin;
  ref Commodities:Id CommodityDestination;
  ref Nodes:Id Destination;
  float Capacity;
  float Cost;
  float Multiplier;
</structure>
<data 61>
```

0,0,1,13,1,1,1e+07,1,1;
1,0,1,21,1,0,1e+07,1,1;
2,0,1,2,1,20,1e+07,1,1;
3,0,1,2,1,14,1e+07,1,1;
4,0,1,23,1,7,1e+07,1,1;
5,0,2,45,2,5,1e+07,1,1;
6,0,2,45,2,10,1e+07,1,1;
7,0,2,6,2,46,1e+07,1,1;
8,0,2,49,2,8,1e+07,1,1;
9,0,2,49,2,9,1e+07,1,1;
10,0,2,11,2,50,1e+07,1,1;
11,0,3,4,3,36,1e+07,1,1;
12,0,3,3,3,36,1e+07,1,1;
13,0,1,12,1,14,1e+07,1,1;
14,0,1,14,1,16,1e+07,1,1;
15,0,1,16,1,18,1e+07,1,1;
16,0,1,20,1,22,1e+07,1,1;
17,0,1,22,1,24,1e+07,1,1;
18,0,1,24,1,26,1e+07,1,1;
19,0,3,28,3,30,1e+07,1,1;
20,0,3,30,3,32,1e+07,1,1;
21,0,3,32,3,34,1e+07,1,1;
22,0,3,36,3,38,1e+07,1,1;
23,0,3,38,3,40,1e+07,1,1;
24,0,3,40,3,42,1e+07,1,1;
25,0,2,44,2,46,1e+07,1,1;
26,0,2,46,2,48,1e+07,1,1;
27,0,2,48,2,50,1e+07,1,1;
28,0,1,52,1,12,1e+07,1,1;
29,0,1,52,1,14,1e+07,1,1;
30,0,1,52,1,24,1e+07,1,1;
31,0,3,56,3,36,1e+07,1,1;
32,0,2,54,2,46,1e+07,1,1;
33,0,2,54,2,48,1e+07,1,1;
34,0,1,53,1,16,1e+07,1,1;
35,0,1,53,1,18,1e+07,1,1;
36,0,3,57,3,38,1e+07,1,1;
37,0,3,57,3,42,1e+07,1,1;
38,0,1,12,1,13,1e+07,0,1;
39,0,1,14,1,15,1e+07,0,1;
40,0,1,16,1,17,1e+07,0,1;
41,0,1,18,1,19,1e+07,0,1;
42,0,1,20,1,21,1e+07,0,1;
43,0,1,22,1,23,1e+07,0,1;
44,0,1,24,1,25,1e+07,0,1;
45,0,1,26,1,27,1e+07,0,1;

```
46,0,3,28,3,29,1e+07,0,1;  
47,0,3,30,3,31,1e+07,0,1;  
48,0,3,32,3,33,1e+07,0,1;  
49,0,3,34,3,35,1e+07,0,1;  
50,0,3,36,3,37,1e+07,0,1;  
51,0,3,38,3,39,1e+07,0,1;  
52,0,3,40,3,41,1e+07,0,1;  
53,0,3,42,3,43,1e+07,0,1;  
54,0,2,44,2,45,1e+07,0,1;  
55,0,2,46,2,47,1e+07,0,1;  
56,0,2,48,2,49,1e+07,0,1;  
57,0,2,50,2,51,1e+07,0,1;  
58,0,1,16,1,26,5,1,1;  
59,0,1,20,1,16,5,1,1;  
60,1,1,18,3,35,4,1,2;  
</data>  
</section>  
</nff>
```

Remerciements

Je tiens à exprimer ma profonde gratitude à mon directeur de recherche, le Professeur Teodor Gabriel Crainic, pour m'avoir suggéré ce sujet de recherche. Sa confiance, sa grande disponibilité ainsi que ses conseils et ses remarques toujours pertinentes m'ont permis, tout au long de ces deux années, de mener à bien mon projet de maîtrise et d'acquérir une expérience appréciable dans le domaine de la recherche opérationnelle appliquée au transport.

Je tiens aussi à remercier mon co-directeur de recherche, le Professeur Michel Gendreau, pour son support et ses encouragements constants, ainsi que pour les échanges d'idées que nous avons eus.

Je souligne que ce projet de maîtrise a été réalisé grâce au support financier du programme Canadien des bourses de la francophonie. J'en profite pour exprimer ma gratitude à tous les responsables du programme, en particulier à mon conseiller pédagogique, Monsieur Ugo-Mercier Gouin.

Je remercie également tous les membres du corps professoral du DIRO, que j'ai côtoyés durant ces deux années de maîtrise, et grâce à qui j'ai énormément appris, notamment les professeurs Jacques Ferland, Sang Nguyen, Felisa Vázquez-Abad, Patrice Marcotte, Claude Crépeau et Bernard Gendron.

Finalement, je rends un grand hommage à ma famille pour son encouragement et ses sacrifices durant mes longues années d'étude.