



Pilotage de production à moyen et à court terme : contribution aux problématiques d'optimisation globale vs locale et à l'ordonnancement dans les raffineries

Georgios Saharidis

► To cite this version:

Georgios Saharidis. Pilotage de production à moyen et à court terme : contribution aux problématiques d'optimisation globale vs locale et à l'ordonnancement dans les raffineries. Sciences de l'ingénieur [physics]. Ecole Centrale Paris, 2006. Français. NNT : . tel-00182465

HAL Id: tel-00182465

<https://theses.hal.science/tel-00182465>

Submitted on 26 Oct 2007

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Thèse de doctorat de l'École Centrale des Arts et Manufactures
Spécialité : Génie Industriel

Présentée par
Georgios K. Saharidis

Le 10 novembre 2006
Pour obtenir le grade de Docteur de l'École Centrale des Arts et Manufactures

Sujet :
Pilotage de production à moyen et à court terme : contribution aux
problématiques d'optimisation globale vs locale et à l'ordonnancement dans les
raffineries

Jury :

Rapporteurs :	Alexandre DOLGUI - Professeur, École Nationale Supérieure des Mines de Saint-Etienne
	Jean-Claude HENNET - Professeur, LAAS-CNRS, Marseille
Examineurs :	Vassilis KOUIKOGLU, Professeur, Technical University of Crete, Greece
	Leslie TROTTER - Professeur, Cornell University, New York
Directeur de thèse :	Yves DALLERY - Professeur, École Centrale Paris
Co-directeur de thèse :	Michel MINOUX - Professeur, Université Paris VI

Laboratoire Génie Industriel
École Centrale Paris
Grande Voie des Vignes 92925
Châtenay-Malabry Cedex

Table des matières

Introduction	v
I Pilotage de production à moyen terme : optimisation locale vs globale	1
1 Positionnement du problème	3
1.1 Introduction	3
1.2 Vue d'ensemble	4
1.2.1 Modèles déterministes	5
1.2.2 Modèles stochastiques	7
1.2.3 Modèles de simulation	7
1.2.4 Modèles hybrides	8
1.3 Objectifs de l'étude (optimisation locale vs globale)	9
1.3.1 Introduction	9
1.3.2 Approche proposée par rapport à la littérature	10
1.4 Problématique étudiée	11
1.5 Paramètres et hypothèses	13
2 Optimisation locale vs globale : cas déterministe	15
2.1 Introduction	15
2.2 Politiques obtenues	19
2.2.1 Optimisation locale	19
2.2.2 Optimisation globale	20
2.3 Modélisation	23
2.3.1 Formulation de notre problématique	23
2.3.2 Les programmes linéaires	25
2.4 Exemple numérique	27
2.5 Résultats	28
2.5.1 Résultats qualitatifs	28
2.5.2 Les résultats numériques et la différence relative entre les deux modèles	29

2.6	Généralisation au cas de participations croisés des entreprises dans les deux usines	31
3	Optimisation locale vs globale : cas stochastique	35
3.1	Introduction	35
3.2	Politiques de contrôle	36
3.3	Modélisation	39
3.4	Algorithme	43
3.5	Résultats	46
3.5.1	Valeurs des paramètres	46
3.5.2	Résultats qualitatifs	47
3.5.3	Variation des coûts	48
3.5.4	Variation des paramètres μ_1 et μ_2	54
	Conclusion-Perspectives	57
	ANNEXES	59
	Annexe A.1	61
1.1	Modélisation	62
1.2	Propriétés 1.1/1.2/1.3 :	63
1.3	Propriété 1.4 :	63
1.4	Propriété 1.7 :	65
1.5	Propriétés 1.8-1.9-1.10-1.11 :	65
	Annexe A.2	67
2.1	Propriété 2.5	67
	Annexe A.3	69
II	Pilotage la production à court terme : ordonnancement dans les raffineries	77
4	Positionnement du problème	79
4.1	Introduction	79
4.2	Vue d'ensemble	83
4.2.1	MILP/MINLP à temps discret	84
4.2.2	MILP/MINLP à temps continu	87
4.2.3	Approches heuristiques	88
4.3	Objectifs de l'étude (ordonnancement du pétrole brut)	89
4.3.1	Introduction	89
4.3.2	Approche proposée par rapport à la littérature	91

4.4	Problématique étudiée	92
4.5	Exemple numérique de référence	95
5	Modélisation proposée	97
5.1	Modèle de base	97
5.1.1	Préparation des mélanges à l'aide des pipelines	101
5.1.2	Préparation des mélanges dans les réservoirs	102
5.1.3	Préparation des mélanges à l'aide des pipelines et dans les réservoirs . . .	102
5.2	Résultats numériques comparatifs	103
5.3	Décomposition de l'horizon temporel du plan de production par partition sur événement	105
5.4	Décomposition de Benders	108
5.4.1	Rappels Mathématiques	108
5.4.2	Algorithme de Benders	111
5.4.3	Décomposition du modèle développé	113
6	Etude algorithmique et mise en oeuvre efficace	119
6.1	Inégalités valides pour l'initialisation du programme maître	119
6.1.1	Inégalités valides générées à partir des données	120
6.1.2	Inégalités valides générées à partir des contraintes de fonctionnement . . .	121
6.1.3	Résultats	124
6.2	Génération multiple des coupes	125
6.3	Génération des coupes significatives	127
6.4	Benders avec génération multiple de coupes valides	129
6.5	Résultats numériques	131
6.5.1	Comparaison générale des modèles	132
6.5.2	Comparaison entre l'optimisation exacte et l'optimisation existante	133
6.5.3	Comparaison entre algorithme de Benders classique et Benders avec génération multiple de coupes valides	134
	Conclusion-Perspectives	139
	Annexe B.1	143

Introduction

L'environnement actuel des entreprises est caractérisé par des marchés soumis à une forte concurrence et sur lesquels les exigences et les attentes des clients deviennent de plus en plus fortes en terme de quantités, coûts et délais de mise à disposition. Cette évolution se renforce par le développement rapide des nouvelles technologies de l'information et de la communication qui permettent une relation directe entre les entreprises, entre les entreprises et leurs clients et entre les entreprises et leurs fournisseurs. Dans un tel contexte, la performance de l'entreprise se construit selon deux principales dimensions :

- une dimension technologique, qui vise à développer les performances intrinsèques des produits mis sur le marché afin de satisfaire aux exigences et aux besoins des marchés rapidement et efficacement et de réduire le coût de possession des produits. L'innovation technologique joue ici un rôle important et peut constituer un élément différenciateur déterminant pour la pénétration et le développement d'un marché,
- une dimension organisationnelle, qui vise à développer la performance en terme de durée des cycles de fabrication, respect des dates de livraison prévues, gestion de stocks et des en-cours, adaptation et réactivité face aux variations du carnet commercial. Il faut donc disposer de méthodes et d'outils de plus en plus performants pour l'organisation et la conduite de la production.

Cette organisation de la production doit être vue non seulement au niveau de l'entreprise elle-même, mais aussi au niveau de sa position au sein d'une chaîne logistique dont elle constitue l'un des maillons. Ceci conduit ainsi à une entreprise globale virtuelle qui doit être orientée vers la satisfaction du besoin des clients aux meilleures conditions. Pour atteindre ces objectifs, l'organisation repose en général sur la mise en oeuvre d'un certain nombre de fonctions parmi lesquelles la fonction d'optimisation à moyen et à court terme.

Ces fonctions visent en effet à organiser l'utilisation des ressources technologiques et humaines présentes dans les ateliers de l'entreprise pour satisfaire directement les demandes des clients. Le pilotage optimal de production à moyen terme est appelée parfois plan agrégé. L'agrégation des décisions permet de simplifier considérablement la formulation, la résolution

et l'interprétation des modèles. Par ailleurs, elle améliore la qualité des prévisions de la demande ainsi que l'estimation d'autres paramètres. Au contraire, le plan de production à court terme appelé aussi ordonnancement de la production, peut être vu comme une version détaillée et désagrégée du plan de production à moyen terme. Il désagrège les familles de produits en articles individuels et il planifie la production sur un horizon plus court avec un découpage temporel plus fin.

Dans cette thèse, nous nous intéressons aux problèmes d'optimisation de la production à moyen et à court terme. Deux parties sont à distinguer dans ce rapport. Dans la première partie, nous étudions le problème du pilotage optimal de production à moyen terme, considérant une optimisation centralisée et une optimisation décentralisée. La deuxième partie présente le problème d'optimisation à court terme, appelé ordonnancement de la production, appliqué à un cas d'étude d'une raffinerie pétrolière.

Dans la première partie, nous nous intéressons au problème de planification de la production à moyen terme et de pilotage de flux. Le système que nous considérons fait référence à deux usines principales, à deux stocks associés à ces deux usines et à deux usines externes appelées sous-traitants. L'objectif est d'obtenir la manière optimale de piloter la production des ces deux usines (que ce soit le plan de production ou les décisions de pilotage de flux). Il y a un seul stock entre la première usine et la deuxième qui est possédé par la première usine et mis à disposition de la deuxième. De plus, nous considérons que chaque usine est représentée en terme de contraintes de capacité par une ressource unique. Les deux usines externes fournissent immédiatement des produits aux stocks des deux usines principales durant la période où les deux usines en commandent. Ces deux usines externes fournissent la quantité voulue mais à un coût plus élevé que le coût unitaire de production de l'usine correspondante.

De nos jours, les gens s'intéressent de plus en plus à avoir un plan de production optimal pour chaque partie de la chaîne logistique mais aussi pour la chaîne entière. Dans la réalité, les entreprises cherchent le pilotage optimal pour chacune de leurs entités, sans tenir compte des attributs de tout le système, en faisant une optimisation locale. Cette approche donne des plans réalisables mais pas optimaux pour la chaîne logistique entière. La problématique étudiée dans cette première partie de thèse est de savoir quelle est la perte engendrée par une optimisation locale par rapport à une optimisation globale et quelles sont les politiques de production adoptées à chaque type d'optimisation. Afin de répondre à cette question, nous analysons le système de deux étages présenté brièvement au paragraphe précédent et l'optimisons de deux façons : localement et globalement. Pour l'optimisation locale nous avons une suite d'optimisations locales de l'aval vers l'amont et pour l'optimisation globale, nous considérons notre

système comme un système global et nous l'optimisons globalement. Nous étudions dès lors le profit et le comportement de notre système pour deux types de demande (déterministe et stochastique) dans le cas des deux techniques d'optimisation (locale et globale).

L'évaluation des performances des systèmes étudiés passe nécessairement par une phase de modélisation. Pour le cas déterministe nous avons développé deux programmes linéaires afin de résoudre le problème de planification et comparer l'optimisation globale et locale. Nous avons réussi à utiliser les deux modèles afin de mettre en évidence un certain nombre de comportements qualitatifs. Pour le cas stochastique, nous avons opté pour la modélisation par le formalisme des files d'attente afin de bénéficier des outils mathématiques puissants qui lui sont associés et qui nous ont permis d'aboutir aux résultats analytiques et numériques.

Les résultats obtenus par cette étude sont intéressants et nous montrent bien quel est le profit pour les entreprises avec une optimisation globale et quelle perte peut engendrer pour les entreprises une optimisation locale en fonction des conditions de sa réalisation. Généralement, nous montrons que l'optimisation globale donne toujours une meilleure solution même si elle pénalise parfois les sous-systèmes. Nos résultats montrent au cas déterministe une différence relative importante entre les deux types d'optimisation. De plus, dans certains cas, la demande stochastique donne aussi un gain important si l'on applique l'optimisation globale.

La première partie est structurée comme suit :

- dans le chapitre 1, nous donnons une vue d'ensemble détaillée du problème d'optimisation de la production à moyen terme, présentant les catégories existantes, selon l'approche de modélisation. Puis nous définissons le système étudié et les paramètres et hypothèses pris ainsi que les deux types de demande considérés,
- dans le chapitre 2, nous présentons le cas déterministe. Avant de présenter la modélisation développée, nous donnons les trois types de politiques obtenus pour les deux types d'optimisation. Après avoir présenté les programmes linéaires associés à notre système, nous donnons nos résultats qualitatifs et nous terminons avec quelques exemples numériques et une généralisation au cas de participations croisés des entreprises dans les deux usines,
- dans le chapitre 3, nous présentons le cas stochastique. Avant de présenter la modélisation développée, nous donnons les politiques de contrôle adoptées. Puis nous présentons les chaînes de Markov associées à ces politiques de pilotage de flux et l'algorithme développé pour les deux types d'optimisation. Cet algorithme définit efficacement les probabilités d'état de chaque chaîne et les stocks optimaux correspondant à la politique d'optimisation choisie. Nous terminons en donnant nos résultats qualitatifs, puis numériques,
- dans la conclusion, nous résumons notre contribution dans la compréhension et l'analyse

des deux types d'optimisation et nous présentons quelques perspectives de recherche.

Dans la deuxième partie de ce rapport de thèse, nous étudions le problème d'optimisation de la production à court terme. Plus précisément, nous étudions le problème d'ordonnancement dans une raffinerie. Le problème présenté correspond au problème d'ordonnancement des activités de chargement/déchargement (**OCD**) du pétrole brut dans les réservoirs de stockage intermédiaires, entre les navires et les unités de distillation du pétrole brut d'une raffinerie. Globalement, une raffinerie est un système constitué de ports, de pipelines, d'une série de réservoirs pour stocker le pétrole brut, d'Unités de Distillation du Pétrole Brut (**UDPB**), d'unités de raffinage et de réservoirs pour faire les mélanges et pour stocker les produits finis et les matières premières. Il y a deux niveaux de décision dans une raffinerie :

- la planification qui détermine les volumes des différentes matières de base à acheter ainsi que les mélanges et les quantités à produire sur plusieurs périodes,
- l'ordonnancement qui traite la synchronisation des opérations à réaliser dans le cadre d'une planification connue en prenant en compte les contraintes opérationnelles du problème.

Une fois que les quantités et les types de pétrole brut à recevoir et que les demandes des UDPB (Unité de **D**istillation du **P**étrole **B**rut) sont donnés, les gestionnaires doivent ordonnancer les chargements et les déchargements des réservoirs. D'une façon générale, la problématique est d'ordonnancer les transferts du pétrole brut des ports vers les réservoirs et des réservoirs vers les UDPB.

L'objectif de notre projet de recherche est de développer un modèle générique qui résout le problème de l'OCD pour tous les types de configuration d'une raffinerie. L'objectif à optimiser est le coût de reconfiguration des réservoirs nécessaires au chargement par les navires et au déchargement vers les unités de distillation. Le coût de reconfiguration est important parce que la reconfiguration d'un réservoir implique une série d'opérations coûteuses.

Cette deuxième partie est structurée comme suit :

- dans le chapitre 4, après la présentation d'un état de l'art sur le problème d'ordonnancement du pétrole brut et de notre approche proposée par rapport à la littérature, nous présentons la problématique étudiée et nous donnons un exemple numérique réel,
- dans le chapitre 5, nous proposons une modélisation générique, on la compare avec celle de la situation courante et nous donnons nos premiers résultats. La comparaison entre la méthode d'optimisation adoptée actuellement et notre méthode d'optimisation exacte montre un gain important en fonction du coût de reconfiguration des réservoirs. Le seul problème présenté est la grande différence en temps de résolution entre notre modèle et

la méthode actuelle. Pour améliorer l'efficacité de notre modèle générique, nous proposons une nouvelle répartition de l'horizon de temps et nous appliquons la méthode de décomposition de Benders,

- dans le chapitre 6, afin d'améliorer encore les résultats obtenus aux chapitres précédents, nous proposons une série d'inégalités valides générées par les données et les contraintes de fonctionnement de notre système. Nous terminons ce dernier chapitre en présentant une méthode originale. Cette méthode est appelée la génération multiple des coupes significatives et pourrait être appliquée dans tous les cas où la méthode de décomposition de Benders a été choisie. Finalement, nous donnons des résultats numériques pour comparer tous les modèles et toutes les méthodes développés,
- dans la conclusion, nous résumons notre contribution dans le domaine d'ordonnancement des activités de chargement/déchargement (**OCD**) du pétrole brut, ainsi que dans le domaine des méthodes de décomposition. Nous terminons avec nos perspectives pour les cas non linéaires du modèle que nous avons développé et pour l'amélioration de la méthode de génération multiple des coupes significatives.

Première partie

Pilotage de production à moyen terme : optimisation locale vs globale

Chapitre 1

Positionnement du problème

1.1 Introduction

Le pilotage optimal de production à moyen terme est parfois appelée '*plan agrégé*'. L'agrégation des décisions permet de simplifier considérablement la formulation, la résolution et l'interprétation des modèles (moins de données à collecter, de calculs à effectuer, de résultats à analyser). Par ailleurs, elle améliore également la qualité des prévisions de demande, ainsi que l'estimation d'autres paramètres.

Les variables contrôlées par le planificateur sont essentiellement de deux types. En premier lieu, la demande de chaque période, qui peut être satisfaite de différentes façons : la production effectuée au cours de cette période, la production des périodes antérieures (livraisons sur stocks) ou encore la sous-traitance à la période actuelle sont trois possibilités de satisfaire cette demande. En combinant ces diverses possibilités, l'entreprise peut niveler son niveau d'activité et donc stabiliser la quantité de ressources utilisées. Le nombre d'unités de chaque article produites, stockées, sous-traitées ou dont la livraison est différée, correspond donc à un premier type de variable de décision. En second lieu, le planificateur doit encore déterminer les ressources de production auxquelles il va faire appel. La ressource la plus flexible sur l'horizon considéré est celle constituée par le personnel de production (certaines décisions d'acquisition d'équipement ou de sous-traitance peuvent parfois être incluses dans le plan agrégé de production). Les décisions à prendre concernent donc principalement les volumes d'embauche, de licenciement, de prestation d'heures supplémentaires, etc. Chacune de ces décisions peut a priori avoir sa portée limitée par des considérations stratégiques ou d'autres contraintes (ex. une politique de maintien de l'emploi, des conventions sectorielles, etc.).

Un modèle classique de planification agrégée a été développé dans les années cinquante par Holt, Modigliani, Muth et Simon (1960). Les variables considérées dans le modèle *HMMS* décrivent les niveaux de production, de stocks, de personnel et d'heures supplémentaires. L'ob-

jectif de minimisation des coûts est traduit dans un modèle d'optimisation quadratique sans contraintes. Les conditions d'optimalité du premier ordre donnent lieu à des règles de décision linéaires qui définissent les niveaux de production et d'emploi optimaux à chaque période sous forme de fonctions linéaires de ces niveaux aux périodes précédentes et de la demande prévisionnelle.

Le modèle *HMMS* permet l'optimisation des coûts, nécessite peu de calculs et conduit à des règles d'utilisation faciles. Ces avantages ont contribué à sa popularité initiale. En revanche, il est peu flexible dans la mesure où il engendre l'addition de contraintes. Plus précisément, le modèle d'optimisation quadratique sous contraintes qui en résulte est beaucoup plus difficile à résoudre que le modèle *HMMS* original. Cette constatation limite considérablement l'applicabilité du modèle.

Les modèles de programmation linéaire, à l'inverse, fournissent une bonne approche qui peut être plus facilement enrichie par la considération de contraintes secondaires. Par exemple, les limitations de capacité de certains centres de production peuvent être naturellement intégrées dans le modèle sous forme de contraintes d'inégalité. Ceci autorise un ajustement charge-capacité (*rough-cut capacity planning*) qui, dans le modèle *HMMS* ou dans les approches manuelles, doit être réalisé a posteriori. Cette souplesse de modélisation, alliée aux autres avantages bien connus de la programmation linéaire, fait de la programmation linéaire un instrument de choix pour l'optimisation de la production agrégée efficace et un outil flexible pour calculer des plans agrégés de production. Plusieurs modèles ont d'ailleurs été développés pour effectuer ce genre de calcul. Enfin, la programmation linéaire permet d'effectuer des analyses de sensibilité sur les solutions proposées.

1.2 Vue d'ensemble

Dans cette section, nous allons présenter un bref éventail des modèles à moyen terme. Ces modèles, dont nous nous servons pour l'analyse et la conception de la chaîne logistique, peuvent être divisés en quatre catégories, selon l'approche de modélisation. Les quatre catégories sont les suivantes :

- des modèles déterministes analytiques, dans lesquels les variables sont connues et bien déterminées,
- des modèles stochastiques analytiques, où au moins une des variables est inconnue et est modélisée par une distribution de probabilité donnée,
- des modèles de simulation,
- des modèles hybrides.

Dans ce qui suit, nous donnons une vue d'ensemble détaillée du problème d'optimisation de la production à moyen terme. Puis nous présentons notre motivation et la problématique de notre recherche. Finalement, nous définissons notre système étudié et les différents cas traités.

1.2.1 Modèles déterministes

[Ishii et al., 1988] développent un modèle déterministe afin de déterminer les niveaux de stocks et les délais d'exécution associés à la solution du coût minimal, pour une chaîne logistique intégrée, sur un horizon fini. Les niveaux de stock et les délais d'exécution sont déterminés de façon à empêcher des ruptures de stock et à minimiser la quantité de stock non utilisé. Dans leur modèle, ils utilisent un système de commande de *type pull*, qui est conduit par des processus de demande linéaires.

[Nahmias, 1996] propose un modèle linéaire permettant de trouver la solution du problème de la planification de la production. La fonction objectif de son modèle permet de minimiser l'ensemble des coûts, particulièrement ceux associés à l'embauche et au licenciement, au maintien des stocks, à la fabrication et à la sous-traitance. Cette fonction est soumise aux contraintes de balancement de flux de la main d'oeuvre, des produits et aux limites de capacités.

Un modèle de planification de la production multi-produits a été développé par Hax et présenté par [Thomas and McClain, 1993]. Ce modèle permet de générer un plan maître de production pour un certain nombre de périodes, avec un niveau d'employés stable et une certaine allocation en temps supplémentaire permise. Le modèle tient compte du coût de maintien en stock, des items produits à l'avance et génère un plan de production en minimisant les frais d'entreposage. Les auteurs citent que le modèle présenté peut être complété, en ajoutant certaines variables et contraintes permettant, dans une certaine mesure, l'embauche et la mise à pied de personnel ainsi que la rupture de stock.

Il est pourtant possible d'incorporer au niveau de la planification agrégée, un peu plus de détails sur les produits et les équipements. Le prix à payer pour l'augmentation de la précision est l'augmentation du nombre de variables et donc du temps de résolution. Un modèle intéressant est celui de [Bilington et al., 1983]. Il s'agit d'un modèle qui inclut la structure des produits et les temps de mise en oeuvre.

[Cohen and Lee, 1989] présentent un modèle déterministe, non linéaire, de programmation mathématique en nombres entiers, basé sur des techniques de quantités économiques de commande (*economic order quantity techniques*) afin de développer ce que les auteurs appellent une politique de déploiement global de ressources. Plus spécifiquement, la fonction objectif employée

dans leur modèle maximise le gain total dans les installations industrielles et les centres de distribution. Cette fonction objectif est soumise à quelques contraintes, incluant des contraintes managériales (contraintes concernant les ressources et la production) et des contraintes logiques de cohérence (faisabilité, disponibilité, limites de demande et non négativité de variables). Les résultats de leur modèle incluent :

- des allocations : des produits finis et des pièces détachées par installation de fabrication, des vendeurs par centre de distribution et des centres de distribution par région de marché,
- des quantités de composants : des pièces détachées et des produits finaux à expédier aux vendeurs, aux installations industrielles et aux centres de distribution,
- des quantités de composants : des pièces détachées et des produits finaux à fabriquer dans les installations industrielles,
- de plus, ce modèle développe des exigences en matière première (material requirements) et des allocations pour tous les produits, en maximisant les gains.

Le modèle de [Hax and Candea, 1984] permet de générer un plan maître de production pour un certain nombre de périodes, avec un niveau d'employés stable et une certaine allocation en temps supplémentaire permis. Le modèle prend en compte le coût de maintien en stock des articles produits à l'avance et génère un plan de production minimisant les coûts du temps supplémentaire et les frais d'entreposage. Ce modèle est complété en ajoutant certaines variables et contraintes permettant, dans une certaine mesure, l'embauche et la mise à pied de personnel ainsi que la rupture de stock.

Le modèle développé par [Young and Sook, 2002] comporte deux sous-modèles, le modèle de production et le modèle de distribution, mais l'optimisation a lieu de façon centralisée. Ce modèle n'a pas la forme typique d'un modèle multi-étages (multi-usines) mais ses principes sont les mêmes. Dans ce cas, nous avons deux tâches qui produisent ensemble le produit final et considérons que chaque tâche a son propre stock. Ensuite, le produit fini est stocké au dépôt principal (*Stack Buffer*) qui fournit les p dépôts satellites qui existent. Finalement, les produits sont distribués à partir de ces dépôts aux q commerçants. L'objectif de ce modèle est de minimiser le coût total des deux étages : le coût de production, le coût de distribution et le coût de stockage.

[Williams, 1981] a développé sept algorithmes heuristiques pour le problème de l'ordonnement et de la distribution. De plus, il a développé [Williams, 1983] un algorithme de programmation dynamique pour l'optimisation de la production. [Morton et al., 1990] proposent quant à eux un modèle qui utilise la sous-traitance pour satisfaire la demande supérieure à la capacité de production de la chaîne logistique. Finalement, [Cohen and Moon, 1990] ont développé l'idée

de [Cohen and Lee, 1989] en élaborant un modèle d'optimisation. Ce modèle étudie les effets des divers coûts de fonctionnement de la chaîne logistique et considère le problème de placement des nouvelles usines et des dépôts.

1.2.2 Modèles stochastiques

[Pyke and Cohen, 1993] développent un modèle de programmation mathématique pour une chaîne logistique intégrée, en utilisant des sous-modèles stochastiques pour calculer les valeurs des variables aléatoires incluses dans le programme mathématique. Ils considèrent une chaîne logistique mono-produit à trois niveaux : une installation industrielle, une installation d'entrepôts et un détaillant. Le modèle minimise le coût total, est soumis à une contrainte au niveau de service et maintient constants les temps de configuration (*setup*), de traitement et de remplissage. Le modèle estime approximativement l'intervalle économique du coût minimal pour commander de nouveau, des quantités de remplissage pour un réseau de production particulier.

[Lee et al., 1997] ont développé des modèles stochastiques mathématiques en décrivant l'effet de *bullwhip*, qui est défini comme le phénomène qui se traduit par l'amplification et la déformation de la fluctuation de la demande de l'acheteur à chaque échelon de la chaîne logistique. En d'autres termes, la variance actuelle et l'ampleur des commandes à chaque échelon deviennent au fur et à mesure plus hautes que la variance et l'ampleur des ventes, ce phénomène ayant comme conséquence la propagation en amont dans la chaîne. Dans cette recherche, les auteurs développent des modèles stochastiques analytiques décrivant les causes de l'effet de *bullwhip* qui sont le traitement du signal de demande et la variation des prix. Ils exposent comment ces causes contribuent à cet effet.

[Lee and Feitzinger, 1995] ont afin développé un modèle analytique qui analyse la configuration de la production avec une demande stochastique. Les auteurs ont pris pour hypothèse de départ une production qui se compose de I pas. Les I pas peuvent avoir lieu soit à l'entreprise, soit aux M centres de distribution. Le problème est de déterminer un pas P tel que les pas entre 1 et P auront lieu à l'entreprise et que les pas de $P + 1$ jusque I auront lieu aux centres de distribution. Ils déterminent le problème optimal qui minimise le coût de la configuration de la production comme la somme du coût du stockage, du coût de configuration et du coût de la fabrication des produits.

1.2.3 Modèles de simulation

[Tzafestas and Kapsiotis, 1994] ont utilisé une approche déterministe de programmation mathématique afin d'optimiser une chaîne logistique et ont employé des techniques de simulation pour analyser un exemple numérique de leur modèle d'optimisation. Dans ce travail, les auteurs

ont déroulé l'optimisation sous deux scénarios différents :

- optimisation d'installations industrielles : sous ce scénario, l'objectif est de minimiser le coût total d'installation industrielle,
- optimisation globale de la chaîne logistique : ce scénario présuppose un rapport coopératif entre toutes les étapes de la chaîne logistique et réduit donc au minimum les coûts opérationnels totaux de la chaîne.

Les auteurs observent que pour l'exemple choisi, les écarts parmi les coûts des deux scénarios ne diffèrent pas significativement.

[Towill et al., 1992] ont employé quant à eux des techniques de simulation pour étudier les effets d'une amplification de la demande sur diverses stratégies dans la chaîne logistique. Les stratégies examinées sont les suivantes :

- éliminer l'échelon de distribution de la chaîne logistique par l'inclusion de la fonction de distribution dans l'échelon industriel,
- intégrer le flux d'information partout dans la chaîne,
- mettre en oeuvre une stratégie d'inventaire de stock (*JIT*) pour réduire les délais,
- améliorer le mouvement des produits et des recours en modifiant les procédures de quantité de commandes,
- modifier les paramètres des procédures actuelles concernant les quantités de commandes.

L'objectif du modèle de simulation était de déterminer quelles stratégies sont les plus efficaces pour l'élimination des variations de la demande. Les auteurs ont montré que la stratégie juste-à-temps et la stratégie de déplacement d'échelon sont les plus efficaces.

Enfin, [Towill and Vecchio, 1994] utilisent le *filter theory* et des techniques de simulation pour l'analyse d'une chaîne logistique. Les techniques de simulation sont utilisées afin de déterminer le stock de sécurité minimal qui satisfait le niveau du service.

1.2.4 Modèles hybrides

La combinaison des modèles analytiques et des modèles de simulation donne les modèles hybrides. La première taxinomie des ces modèles fut présentée par [Shanthikumar and Sargent, 1983]. Des études plus récentes ont été faites par [Ozdamar and Birbil, 1998], [Byrne and Bakir, 1999] et [Mummolo and Iavagnilio, 1999]. Nous donnons ci-dessous deux exemples des modèles hybrides.

[Riane and Iassinovski, 1999] ont développé un modèle pour un *Hybrid flowshop* qui prend en compte des sous-traitants pour optimiser la chaîne logistique. *Hybrid flowshop* est une combinaison de tâches en série et de tâches en parallèle. Pour donner la forme du modèle de

programmation mathématique d'un *Hybrid flowshop* à N produits et W tâches, il faut définir d'abord un horizon du temps H qui est séparé en I sous-périodes : pendant la période t , tous les centres G de chaque tâche w ont une capacité d'unité du temps $Capa(w, g, t)$.

Le modèle développé par [Lee and Kim, 2000] comporte deux sous-modèles, le modèle de production et le modèle de distribution, mais l'optimisation a lieu de façon centralisée (globale). Ce modèle a les principes d'un modèle multi-usines. Dans ce cas, on a deux tâches qui produisent ensemble le produit final (chaque tâche dispose de son propre stock). Ensuite, le produit fini est stocké dans l'inventaire qui fournit les dépôts existants. Finalement, les produits sont distribués à partir de ces dépôts aux détaillants. La méthode employée fut une approche hybride, analytique et de simulation. Les auteurs prétendent que si une de ces approches était employée séparément de l'autre, le résultat ne serait pas si performant.

Suite à cet état de l'art, nous présentons à présent notre motivation, notre objectif de recherche et notre système étudié.

1.3 Objectifs de l'étude (optimisation locale vs globale)

1.3.1 Introduction

De nos jours, les gens s'intéressent de plus en plus à une vision globale de la chaîne logistique qui se compose maintenant d'un grand nombre des sites répartis partout dans le monde. Quelques entreprises se sont divisées en de plus petites structures chacune ayant des fournisseurs, des clients et des sous-traitants. L'optimisation de la production est en conséquence devenue plus compliquée et plus difficile. Il est alors nécessaire d'avoir un plan de production optimal pour chaque membre de la chaîne logistique considéré individuellement, mais aussi pour la chaîne entière.

Dans la réalité, les entreprises cherchent un plan optimal pour chacune de ses entités, sans tenir compte des attributs de tout le système, en faisant une optimisation locale. Cette approche donne des plans réalisables mais pas optimaux pour la chaîne logistique entière. Les raisons pour lesquelles les chefs des entreprises préfèrent faire une optimisation locale au lieu d'une optimisation globale sont nombreuses. Les plus critiques sont :

- la concurrence entre ces entités,
- la difficulté d'avoir une distribution des informations entre les membres de la chaîne logistique à cause de l'indisponibilité ou du délai de transmission.

Notre problème initial est de savoir quelle est la perte engendrée par une optimisation locale par rapport à une optimisation globale et quelles sont les politiques de production adoptées par chaque type d'optimisation ? Afin de répondre à cette question, nous allons analyser une chaîne logistique simple et l'optimisons de deux façons : localement et globalement. Nous étudions dès lors le profit et le comportement de notre système pour deux types de demande (déterministe et stochastique) dans le cas des deux techniques d'optimisation.

1.3.2 Approche proposée par rapport à la littérature

La recherche bibliographique nous permet de voir que peu d'articles traitent le sujet précédent. A l'heure de la rédaction de la thèse ne semble exister que seulement deux articles : l'un traite ce problème avec une demande déterministe et l'autre avec une demande stochastique.

Le cas déterministe est traité par [Jen Ming and Tsung Hui, 2005]. Les auteurs étudient le problème d'une chaîne logistique à deux étages qui vend un grand nombre de produits sur un marché international. La demande de chaque produit est déterministe et constante, pendant toute la période de vente. Les auteurs ont considéré deux politiques de décision, la politique centralisée et la politique décentralisée. Ils examinent le problème de coordination et de réduction du coût total de fonctionnement en fonction de deux politiques adoptées. L'objectif est de minimiser le coût de reconfiguration pour satisfaire une demande. Chaque fois qu'il y a une commande, les entreprises paient un coût fixe de reconfiguration qui est induit par exemple par la distance que le camion va parcourir et qui ne varie pas en fonction de la quantité commandée. Des résultats numériques sont présentés pour examiner la performance mathématique des deux modèles proposés et pour montrer le gain du système en adoptant la politique centralisée, qui devient en l'occurrence très intéressante quand le coût de reconfiguration est important. L'article conclut en proposant aux entreprises d'adopter la politique centralisée, en donnant en même temps une méthode pour partager le gain obtenu. L'approche que nous proposons de cette thèse se différencie dans ce travail en raison des points suivants : premièrement, la demande choisie est une demande déterministe mais pas constante. Nous avons choisi une demande saisonnière pour que l'on puisse aussi étudier ce qui se passe au niveau des politiques de production adoptées (produire par anticipation, sous-traiter, etc.). Dans notre étude, la politique de production n'est pas a priori choisie mais elle résulte d'une optimisation. Avec cette possibilité, il peut satisfaire sa demande en produisant un stock d'anticipation mais aussi, si c'est économiquement plus attractif, de sous-traiter. Nous voulions étudier quelles sont les politiques adoptées par une optimisation locale et quels sont les changements au niveau des politiques si nous pratiquons l'optimisation globale (cf. section 2.2). Enfin, une autre différence entre notre modélisation et la modélisation proposée par [Jen Ming and Tsung Hui, 2005] est que notre modèle est un modèle linéaire complet avec des contraintes de capacité et de production où l'on cherche à minimi-

ser un coût total (production, sous-traitance et stockage) et pas à minimiser seulement le coût de reconfiguration. Cette différence est une conséquence de l'hypothèse de demande saisonnière.

Le cas stochastique est traité par [Gnonia et al., 2003]. Les auteurs présentent une étude de cas dans l'industrie automobile et traitent le problème de *Lot Sizing and Scheduling Problem* (LSSP) d'un système multi-sites avec des contraintes de capacité, multi-produits et multi-périodes avec une demande stochastique qui suit une distribution de Weibull. Le LSSP est résolu par un modèle hybride constitué d'un modèle linéaire en nombres entiers et d'un modèle de simulation. Ce modèle hybride est développé pour faire l'optimisation locale et globale. L'article étudie comment la demande stochastique influence les performances économiques du système dans les deux cas d'optimisation. Deux situations ont été comparées. La première considère le système comme une série des sous-systèmes et optimise chacun des ses membres séparément. La deuxième considère le système comme un système entier et fait l'optimisation globale. Les résultats obtenus par l'optimisation locale ont été comparés avec la simulation de la situation courante et la comparaison a montré une réduction importante du coût de production. Ensuite, l'optimisation globale a été comparée avec l'optimisation locale et la comparaison a également montré une réduction supplémentaire et une différence relative importante entre les deux coûts de production. L'approche que nous proposons dans cette thèse se différencie de ce travail en raison des points suivants : premièrement, la demande suit une distribution Poissonienne, une distribution qui nous semble être plus générique que la distribution de Weibull. De plus, nous avons choisi de maximiser le profit de notre système au lieu de minimiser le coût de production pour que l'on puisse étudier plus profondément le comportement de la première usine de notre système et son influence par rapport à la deuxième usine. Finalement, la plus grande différence entre les deux études est la modélisation adoptée. Alors que [Gnonia et al., 2003] ont développé un modèle hybride, notre modèle est un modèle stochastique qui utilise les chaînes de Markov.

1.4 Problématique étudiée

Le système que nous étudions dans notre travail fait référence à deux usines principales (usine₁, usine₂), à deux stocks qui sont fournis par ces deux usines et à deux usines externes appelées sous-traitants. Dans ce travail, nous voulons obtenir la manière optimale de piloter la production de ces deux usines. Il y a un seul stock entre l'usine₁ et l'usine₂ et ce stock est possédé par l'usine₁ et est mis à disposition de l'usine₂. De plus, nous considérons que chaque usine est représentée en terme de contraintes de capacité par une ressource unique. Les deux usines externes fournissent des produits aux stocks des deux usines principales immédiatement durant la période pendant laquelle les deux usines les commandent. Ces deux usines externes fournissent la quantité voulue mais à un coût unitaire plus élevé que le coût unitaire de production de l'usine correspondante. Notre système peut avoir deux variantes : soit le sous-traitant

prend des produits du stock amont de l'usine qui sous-traite (cf. figure 4.1), soit d'un stock extérieur (cf. figure 4.2).

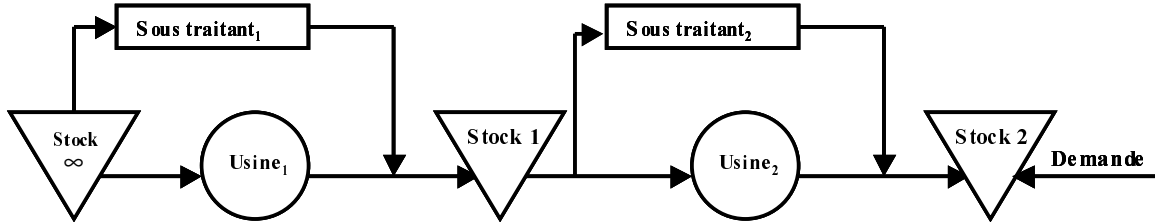


Fig. 1.1 – Système Étudié 1

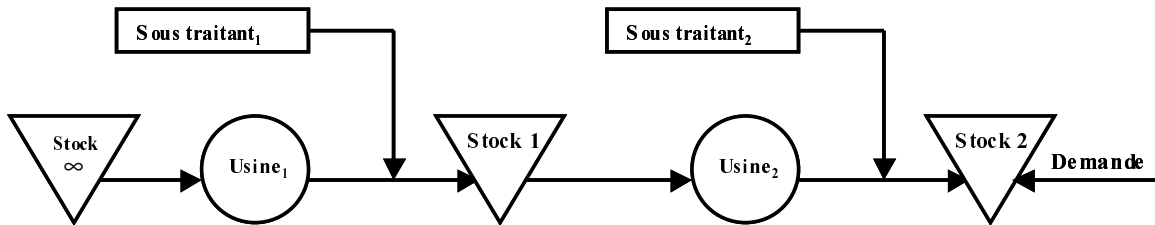


Fig. 1.2 – Système Étudié 2

Dans l'approche classique, nous avons une suite d'optimisations locales de l'aval vers l'amont. La démarche est contraire pour l'optimisation globale : nous considérons notre système comme un système global et nous l'optimisons globalement en tenant compte de toutes les particularités des deux usines que sont les coûts de production, de sous-traitance et de stockage, ainsi que les capacités de production. Notre problème initial est de savoir d'une part quel est le bénéfice d'une optimisation globale de cette chaîne logistique par rapport à l'optimisation locale et d'autre part si le bénéfice engendré est suffisant. Cette situation est présentée dans les figures 1.3 et 1.4 (les lignes en pointillé représentent le cas où les sous-traitants prennent des pièces du stock amont de l'usine qui sous-traite).

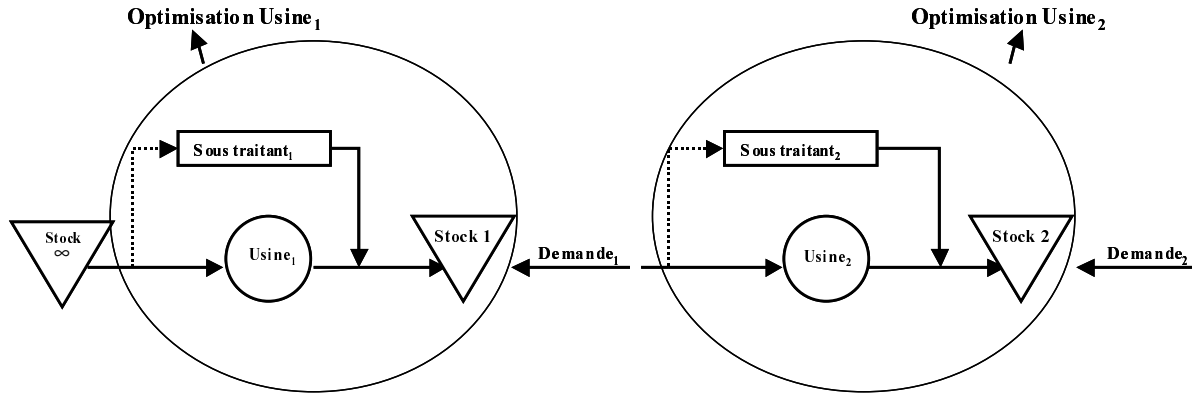


Fig. 1.3 – Optimisation locale

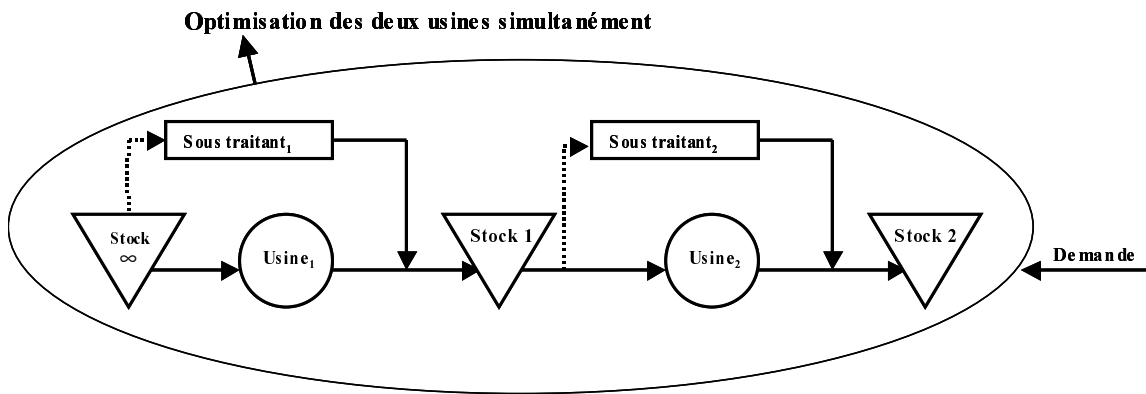


Fig. 1.4 – Optimisation globale

Notre travail a pour objectif de développer deux modèles, le premier fondé sur l'optimisation globale et l'autre sur l'optimisation locale. La fonction objectif de ces modèles est de maximiser le profit total, sous les contraintes et les hypothèses de notre système. Dans la section suivante, nous présentons les variables du système, les coûts, les contraintes et les hypothèses prises.

1.5 Paramètres et hypothèses

Les variables et les coûts de notre système sont liés avec la production, la sous-traitance et le niveau de stock. Nous distinguons les variables et les coûts suivants :

- une variable qui correspond à la production de l'usine_i,
- une variable qui correspond à la production du sous-traitant_i,
- une variable qui correspond au niveau de stock à l'usine_i.

Au niveau des contraintes, nous prenons les hypothèses suivantes :

- nous produisons un seul produit,

- les sous-traitants ont une capacité de production infinie,
- les coûts unitaires de production et de sous-traitance sont fixes et proportionnels à la quantité produite et sous-traitée réciproquement (coût linéaire),
- le stock qui alimente l'usine₁ n'est jamais vide,
- les coûts de stockage sont linéaires,
- le coût de stockage pour la deuxième usine est toujours plus élevé que celui de la première usine (la deuxième usine apporte de la valeur ajoutée au produit).

Concernant la demande du produit, nous avons étudié, pour les deux types d'optimisation (globale et locale), deux types de demande. Une demande déterministe et saisonnière (cf. figure 1.5) et une demande stochastique stationnaire non saisonnière (cf. figure 1.6).

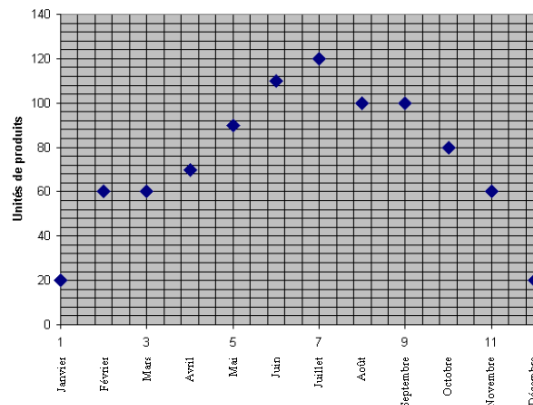


FIG. 1.5 – Demande Déterministe



FIG. 1.6 – Demande Stochastique

Dans les deux prochains chapitres, nous présentons le comportement de notre système face aux deux méthodes d'optimisation et aux deux types de demande. Nous illustrons notre modélisation et nous donnons nos résultats numériques et qualitatifs.

Chapitre 2

Optimisation locale vs globale : cas déterministe

2.1 Introduction

Concernant l'optimisation de la production à moyen terme avec une demande déterministe, les gestionnaires font la planification de la production. La planification de la production au sein d'une chaîne logistique constitue un enjeu majeur pour les entreprises. Ce processus a pour rôle de déterminer le meilleur équilibre entre la demande et les ressources mises en oeuvre pour satisfaire cette demande. Ce processus porte le plus souvent sur un horizon de plusieurs mois. Dans le cas déterministe, nous avons utilisé le système présenté au chapitre 1.4 (cf. figure 2.1). Dans cette version de notre système, les sous-traitants prennent des pièces du stock amont de l'usine qui sous-traite. De plus ce système correspond à l'étude de cas de l'entreprise ANALKO développé dans la section 2.4.

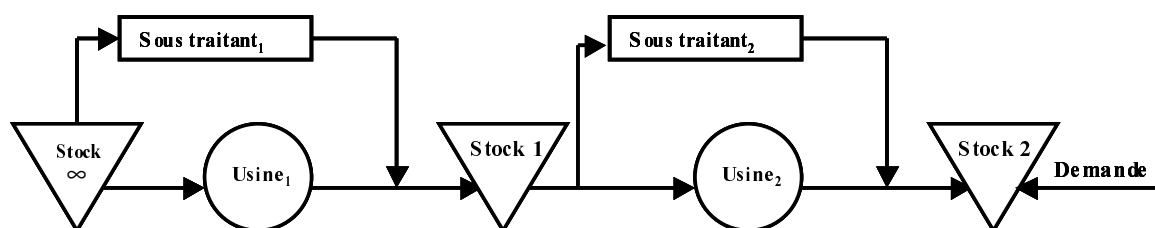


Fig. 2.1 – Système Etudié-Cas Déterministe

Dans le cas déterministe, nous prenons en compte quelques hypothèses supplémentaires. Premièrement, nous ne permettons pas de demande en attente (pas de retard dans la satisfaction des demandes). En conséquence la demande de chaque période qui excède la capacité de production doit être satisfaite, soit par une production d'anticipation, soit par la sous-traitance, soit par une combinaison des deux. Tout en gardant la généralité, nous supposons que le stock

initial et le stock final sont vides et qu'il n'y a pas de demande pendant la première période. Deuxièmement, nous supposons ne pas avoir de contraintes de capacité de stock des usines et de pouvoir toujours satisfaire toute la demande supplémentaire en produisant par anticipation. Finalement, nous prenons un délai d'exécution (*leadtime*) entre les deux usines qui est égal à une période. Pour un horizon de planification de T périodes, les variables de décision sont :

- $P_{i,t}$: la production à l'usine $_i$, durant la période t ,
- $I_{i,t}$: le niveau de stock à l'usine $_i$, durant la période t (*Inventory*),
- $SC_{i,t}$: les produits sous-traités par le sous-traitant $_i$, durant la période t (*SubContracting*).

Les coûts sont définis dans la liste suivante :

- p_i : le prix de vente d'une pièce par l'usine $_i$,
- cp_i : le coût de production d'une pièce à l'usine $_i$,
- h_i : le coût de stockage d'une pièce, durant une période à l'usine $_i$,
- csc_i : le coût d'une pièce sous-traitée par le sous-traitant $_i$.

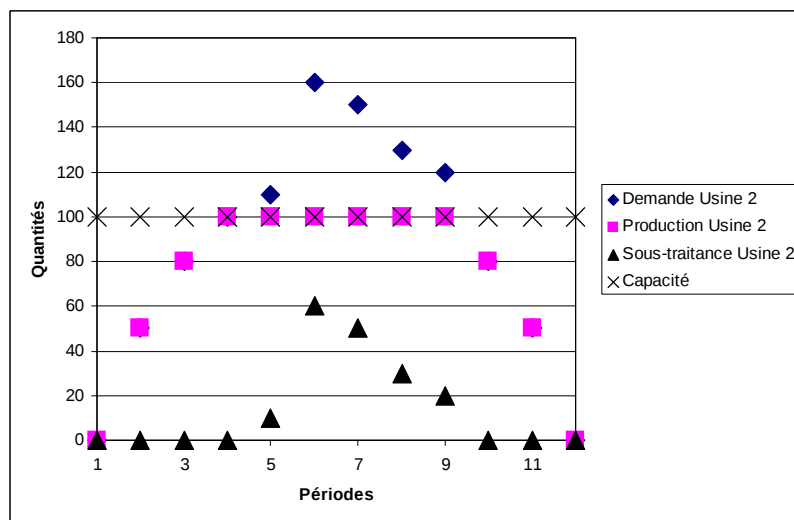
Nous avons choisi, pour étudier le comportement de notre modèle de planification de la production de ces deux usines, une demande saisonnière. Cette demande permet aux deux usines, en tenant compte de leur propre coût, d'avoir plusieurs politiques pour produire de façon à satisfaire la demande de leurs clients. Les deux usines peuvent prendre trois politiques pour satisfaire la demande supplémentaire :

- soit elles produisent par anticipation,
- soit elles sous-traitent,
- soit elles obtiennent une politique mixte qui correspond à une combinaison des deux politiques précédentes.

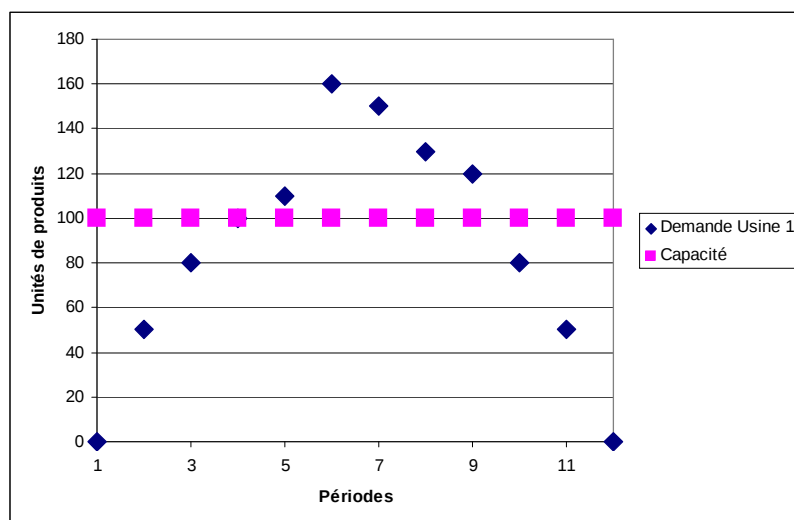
Nous nous intéressons dans cette section à caractériser aussi la demande de l'usine $_1$. Nous notons par D_t la demande à satisfaire par l'usine $_2$ et D'_t par l'usine $_1$, pendant la période t . La demande D'_t est la somme de la production et de la sous-traitance de l'usine $_2$ durant la période t parce que le sous-traitant $_2$ prend des pièces par le stock de l'usine $_1$.

Nous pouvons distinguer trois cas généraux, concernant la demande D'_t de l'usine $_1$ qui dépend du plan de production de l'usine $_2$:

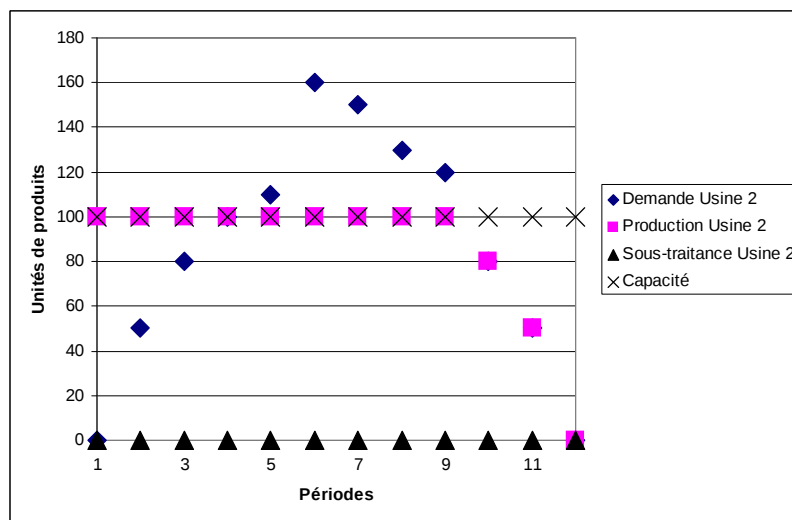
- Le pire cas pour l'usine $_1$ est quand l'usine $_2$ satisfait sa demande en adoptant la politique de sous-traitance (figure 2.2).

Fig. 2.2 – Les plans de l'usine₂

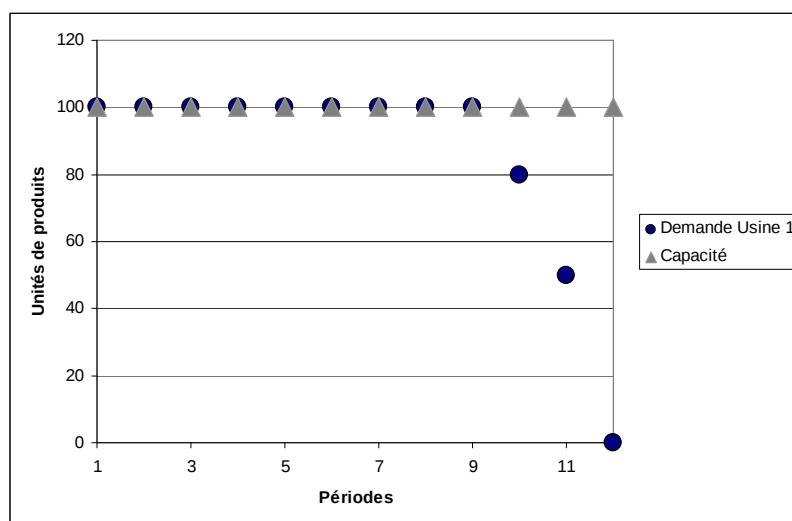
Dans ce cas, la demande de l'usine₁ est exactement la même que celle de l'usine₂ (figure 2.3). Dans ce cas, l'usine₁ est obligée de satisfaire la plus grande quantité possible de demande supplémentaire.

Fig. 2.3 – La demande de l'usine₁

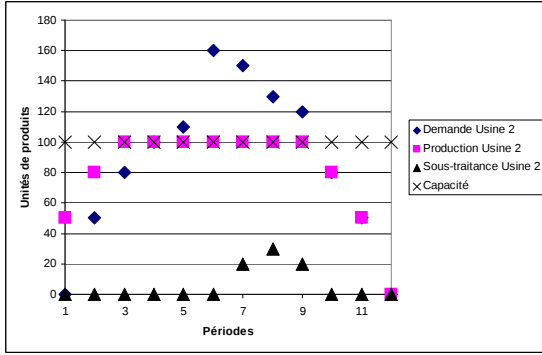
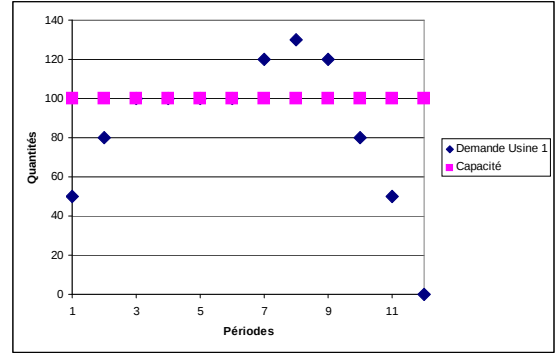
- Le meilleur cas pour l'usine₁ est quand l'usine₂ satisfait sa demande en adoptant la politique de production par anticipation (figure 2.4).

Fig. 2.4 – Les plans de l'usine₂

Dans ce cas, la demande supplémentaire de la capacité de production de l'usine₁ est la plus petite possible (figure 2.5).

Fig. 2.5 – La demande de l'usine₁

- Le cas intermédiaire est quand l'usine₂ satisfait sa demande en adoptant une politique mixte. Tant qu'elle produit par anticipation, le coût de production de l'usine₁ est inférieur. Dans les figures 2.6 et 2.7, nous présentons un exemple qui illustre ce cas.

FIG. 2.6 – Les plans de l'usine₂FIG. 2.7 – La demande de l'usine₁

2.2 Politiques obtenues

Ce qui nous intéresse aussi est d'examiner les différentes politiques obtenues par les deux usines dans les deux types d'optimisation et de savoir quels sont les changements en fonction de la politique choisie. Dans ce chapitre, nous donnons les politiques possibles que peut adopter notre système dans le cadre d'une optimisation locale. Ensuite, nous examinerons les différences entre l'optimisation locale et l'optimisation globale en ce qui concerne les politiques optimales obtenues par chaque usine.

2.2.1 Optimisation locale

La politique adoptée dépend aussi de la capacité de production des deux usines (C_i la capacité de production de l'usine _{i}). Il y a trois possibilités :

- soit la capacité de production des deux usines est égale, $C_1 = C_2$,
- soit l'usine₁ a une capacité de production supérieure à la capacité de l'usine₂, $C_1 > C_2$,
- soit l'usine₂ a une capacité de production supérieure à la capacité de l'usine₁, $C_1 < C_2$.

Avant de présenter, dans le tableau 2.1, toutes les politiques possibles (optimisation locale), nous définissons les notations utilisés :

- Pour chaque politique nous utilisons les notations suivantes :
 - $\emptyset$: aucune politique adoptée, car la demande ne dépasse pas la capacité de production,
 - I : produire par anticipation,
 - SC : sous-traiter,
 - I-SC : politique mixte (i.e. produire par anticipation et sous-traiter en même temps).
- Pour présenter les politiques adoptées par les deux usines, nous avons défini le symbolisme suivant : avant la flèche, nous présentons la politique adoptée par l'usine₂ et après la flèche la politique adoptée par l'usine₁. L'exemple $SC \Rightarrow I$ signifie que l'usine₂ sous-traite toute la demande supplémentaire et l'usine₁ produit par anticipation,

- Pour préciser le type d'optimisation adopté, nous utilisons le symbolisme suivant : ($OL : SC \Rightarrow I$) signifie que pour l'optimisation locale (OG à l'optimisation globale) nous avons la politique $SC \Rightarrow I$.

Dans tous les cas de l'optimisation locale, les usines peuvent adopter toutes les combinaisons possibles des politiques. La seule exception est dans certains cas où l'usine₂ adopte la politique de produire par anticipation. Dans ce cas, nous n'observons pas de période où la demande de l'usine₁ est supérieure à sa capacité de production. Cette situation donne la possibilité à l'usine₁ de satisfaire toute sa demande avec sa propre production sans créer un stock et sans sous-traiter. Dans le tableau suivant (tabl.2.1), nous présentons toutes les politiques possibles pour les deux usines en fonction de la capacité de production.

$C_1 = C_2$	$C_1 > C_2$	$C_1 < C_2$
$I \Rightarrow \emptyset$	$I \Rightarrow \emptyset$	$I \Rightarrow I$
$I \& SC \Rightarrow I$	$I \& SC \Rightarrow \emptyset_{si \exists SC_2, t \leq C_1 - C_2}$	$I \Rightarrow I \& SC$
$I \& SC \Rightarrow I \& SC$	$I \& SC \Rightarrow I$	$I \Rightarrow SC$
$I \& SC \Rightarrow SC$	$I \& SC \Rightarrow SC$	$I \& \Rightarrow I$
$SC \Rightarrow I$	$I \& SC \Rightarrow I \& SC$	$I \& SC \Rightarrow I \& SC$
$SC \Rightarrow I \& SC$	$SC \Rightarrow \emptyset_{si \exists SC_2, t \leq C_1 - C_2}$	$I \& SC \Rightarrow SC$
$SC \Rightarrow SC$	$SC \Rightarrow I$	$SC \Rightarrow I$
	$SC \Rightarrow SC$	$SC \Rightarrow I \& SC$
	$SC \Rightarrow I \& SC$	$SC \Rightarrow SC$

TAB. 2.1 – Optimisation locale

2.2.2 Optimisation globale

Pour que nous puissions examiner toutes les politiques que nous pouvons obtenir à l'optimisation globale (OG) à la suite d'une optimisation locale (OL), nous présentons les trois principaux attributs de notre système, concernant le coût de production :

- Coût d'Optimisation Globale du système entier (**COG**) $\leq$ Coût d'Optimisation Locale du système entier (**COL**) (propr. 1.1, annexe A.1),
- Coût d'Optimisation Locale en usine₂ (**COLU2**) $\leq$ Coût d'Optimisation Globale en usine₂ (**COGU2**) (propr. 1.2, annexe A.1),
- Coût d'Optimisation Locale en usine₁ (**COLU1**) $\geq$ Coût d'Optimisation Globale en usine₁ (**COGU1**) (propr. 1.3, Annexe A.1).

Quand les capacités de production des deux usines sont égales, nous pouvons obtenir toutes les combinaisons présentées aux tableaux 2.2 et 2.3. Le terme "POS" est utilisé pour tous les combinaisons possibles et le terme "IMP" pour les combinaisons impossibles.

$C_1 = C_2$	Globale				
Politiques	Locale	$I \Rightarrow \emptyset$	$I \& SC \Rightarrow I$	$I \& SC \Rightarrow I \& SC$	$I \& SC \Rightarrow SC$
Pol.1	$I \Rightarrow \emptyset$	POS	IMP	IMP	IMP
Pol.2	$I \& SC \Rightarrow I$	POS	POS	POS	POS
Pol.3	$I \& SC \Rightarrow I \& SC$	POS	POS	POS	POS
Pol.4	$I \& SC \Rightarrow SC$	POS	POS	POS	POS
Pol.5	$SC \Rightarrow I$	POS	POS	POS	POS
Pol.6	$SC \Rightarrow I \& SC$	POS	POS	POS	POS
Pol.7	$SC \Rightarrow SC$	POS	POS	POS	POS

TAB. 2.2 – Optimisation locale vs globale ($C_1 = C_2$)

$C_1 = C_2$	Globale			
Politiques	Locale	$SC \Rightarrow I$	$SC \Rightarrow I \& SC$	$SC \Rightarrow SC$
Pol.1	$I \Rightarrow \emptyset$	IMP	IMP	IMP
Pol.2	$I \& SC \Rightarrow I$	IMP	IMP	IMP
Pol.3	$I \& SC \Rightarrow I \& SC$	IMP	IMP	IMP
Pol.4	$I \& SC \Rightarrow SC$	IMP	IMP	IMP
Pol.5	$SC \Rightarrow I$	POS	IMP	IMP
Pol.6	$SC \Rightarrow I \& SC$	IMP	POS	IMP
Pol.7	$SC \Rightarrow SC$	IMP	IMP	POS

TAB. 2.3 – (*suite*) Optimisation local vs globale ($C_1 = C_2$)

- Quand l'OL donne ($L : I \Rightarrow \emptyset$), l'OG donne la même politique et exactement le même plan de production ($G : I \Rightarrow \emptyset$). Nous ne pouvons pas avoir une autre politique sinon le $COL1 \geq COG1$ ne sera pas valable (propr.1.3),
- Quand l'OL donne ($L : I \& SC \Rightarrow I$), l'OG donne toutes les politiques possibles sauf les trois dernières. Si nous prenons pour l'OL comme politique optimale pour l'usine₂ $I \& SC$, l'OG ne peut pas donner la politique SC , parce que dans ce cas nous aurions $COL1 < COG1$, qui n'est pas valable (propr.1.3). Pour la même raison, les trois dernières politiques ne sont pas valables pour les cas ($L : I \& SC \Rightarrow I \& SC$) et ($L : I \& SC \Rightarrow SC$),
- Quand l'OL donne ($L : SC \Rightarrow I$), l'OG donne les politiques : ($G : I \& SC \Rightarrow I$), ($G : I \& SC \Rightarrow I \& SC$) et ($G : I \& SC \Rightarrow SC$) parce qu'en produisant par anticipation à l'usine₂, l'usine₁ doit satisfaire une demande supplémentaire. Cette demande est plus petite qu'à l'OL où l'usine₂ sous-traite pour satisfaire la demande supplémentaire du produit fini. Les cas : ($G : SC \Rightarrow I \& SC$) et ($G : SC \Rightarrow I \& SC$) ne se produisent pas parce que l'usine₁ suit la même courbe de la demande pour les deux optimisations et, avec la même demande, donne la même solution optimale : ($G : SC \Rightarrow I$). Pour les mêmes raisons, pour les optimisations locales ($L : SC \Rightarrow I \& SC$) et ($L : SC \Rightarrow SC$) nous pouvons prendre toutes les politiques sauf ($G : SC \Rightarrow I$), ($G : SC \Rightarrow SC$) et ($G : SC \Rightarrow I$), ($G : SC \Rightarrow I \& SC$) respectivement.

Dans les tableaux 2.4 et 2.5, nous présentons toutes les combinaisons quand la capacité de production de l'usine₁ est supérieure à la capacité de production de l'usine₂.

$C_1 > C_2$	Globale				
Politiques	Locale	$I \Rightarrow \emptyset$	$I \& SC \Rightarrow \emptyset$	$I \& SC \Rightarrow I$	$I \& SC \Rightarrow I \& SC$
Pol.1	$I \Rightarrow \emptyset$	POS	POS	IMP	IMP
Pol.2	$I \& SC \Rightarrow \emptyset$	POS	POS	IMP	IMP
Pol.3	$I \& SC \Rightarrow I$	POS	IMP	POS	POS
Pol.4	$I \& SC \Rightarrow I \& SC$	POS	IMP	POS	POS
Pol.5	$I \& SC \Rightarrow SC$	POS	IMP	POS	POS
Pol.6	$SC \Rightarrow \emptyset$	POS	POS	IMP	IMP
Pol.7	$SC \Rightarrow I$	POS	POS	POS	POS
Pol.8	$SC \Rightarrow I \& SC$	POS	POS	POS	POS
Pol.9	$SC \Rightarrow SC$	POS	POS	POS	POS

TAB. 2.4 – Optimisation local vs globale ($C_1 > C_2$)

$C_1 > C_2$	Globale					
Politique	Locale	$I \& SC \Rightarrow SC$	$SC \Rightarrow \emptyset$	$SC \Rightarrow I$	$SC \Rightarrow I \& SC$	$SC \Rightarrow SC$
Pol.1	$I \Rightarrow \emptyset$	IMP	POS	IMP	IMP	IMP
Pol.2	$I \& SC \Rightarrow \emptyset$	IMP	POS	IMP	IMP	IMP
Pol.3	$I \& SC \Rightarrow I$	POS	IMP	IMP	IMP	IMP
Pol.4	$I \& SC \Rightarrow I \& SC$	POS	IMP	IMP	IMP	IMP
Pol.5	$I \& SC \Rightarrow SC$	POS	IMP	IMP	IMP	IMP
Pol.6	$SC \Rightarrow \emptyset$	NON	POS	IMP	IMP	IMP
Pol.7	$SC \Rightarrow I$	POS	IMP	POS	IMP	IMP
Pol.8	$SC \Rightarrow I \& SC$	POS	IMP	IMP	IMP	POS
Pol.9	$SC \Rightarrow SC$	POS	IMP	IMP	POS	IMP

TAB. 2.5 – (suite) Optimisation local vs globale ($C_1 > C_2$)

- Quand l'OL donne (L : $I \& SC \Rightarrow I$), au moins pour un t , nous avons : $SC_{2,t} > C_1 - C_2$, à l'OG nous ne pouvons pas avoir la politique (G : $I \& SC \Rightarrow \emptyset$) parce qu'elle n'est pas valable $SC_{2,t} \leq C_1 - C_2 \forall t \in [1, T]$,
- Quand l'OL donne (L : $I \& SC \Rightarrow I$) ou (L : $I \& SC \Rightarrow I \& SC$) ou (L : $I \& SC \Rightarrow SC$), à l'OG nous ne pouvons pas obtenir une solution optimale avec les trois dernières politiques. La raison est que si nous avons à l'OL la politique $I \& SC$ pour l'usine₂ et qu'à l'OG nous avons une politique SC , alors la demande supplémentaire à la capacité de production de l'usine₁ est plus élevée. D'une part, pour que nous puissions avoir la 3ième, 4ième, 5ième, 7ième, 8ième et 9ième politique, il faut que $SC_{2,t} > C_1 - C_2$ soit valable et d'autre part pour que nous puissions avoir la 2ième et la 6ième politique, il faut que $SC_{2,t} \leq C_1 - C_2 \forall t$ soit valable.

Dans les tableaux 2.6 et 2.7, le cas $C_1 < C_2$ est présenté.

$C_1 < C_2$	Globale					
Politique	Locale	$I \Rightarrow I$	$I \Rightarrow I \& SC$	$I \Rightarrow SC$	$I \& SC \Rightarrow I$	$I \& SC \Rightarrow I \& SC$
Pol.1	$I \Rightarrow I$	POS	IMP	IMP	IMP	IMP
Pol.2	$I \Rightarrow I \& SC$	POS	POS	POS	IMP	IMP
Pol.3	$I \Rightarrow SC$	POS	POS	POS	IMP	IMP
Pol.4	$I \& SC \Rightarrow I$	POS	POS	POS	POS	POS
Pol.5	$I \& SC \Rightarrow I \& SC$	POS	POS	POS	POS	POS
Pol.6	$I \& SC \Rightarrow SC$	POS	POS	POS	POS	POS
Pol.7	$SC \Rightarrow I$	POS	POS	POS	POS	POS
Pol.8	$SC \Rightarrow I \& SC$	POS	POS	POS	POS	POS
Pol.9	$SC \Rightarrow SC$	POS	POS	POS	POS	POS

TAB. 2.6 – Optimisation local vs globale ($C_1 < C_2$)

$C_1 < C_2$	Globale				
Politique	Locale	$I \& SC \Rightarrow SC$	$SC \Rightarrow I$	$SC \Rightarrow I \& SC$	$SC \Rightarrow SC$
Pol.1	$I \Rightarrow I$	IMP	IMP	IMP	IMP
Pol.2	$I \Rightarrow I \& SC$	IMP	IMP	IMP	IMP
Pol.3	$I \Rightarrow SC$	IMP	IMP	IMP	IMP
Pol.4	$I \& SC \Rightarrow I$	POS	IMP	IMP	IMP
Pol.5	$I \& SC \Rightarrow I \& SC$	POS	IMP	IMP	IMP
Pol.6	$I \& SC \Rightarrow SC$	POS	IMP	IMP	IMP
Pol.7	$SC \Rightarrow I$	POS	POS	IMP	IMP
Pol.8	$SC \Rightarrow I \& SC$	POS	IMP	POS	IMP
Pol.9	$SC \Rightarrow SC$	POS	IMP	IMP	POS

TAB. 2.7 – (*suite*) Optimisation locale vs globale ($C_1 < C_2$)

- Quand l'OL donne ($L : I \Rightarrow I$), l'OG donne exactement la même politique optimale,
- Généralement, quand l'OL donne une politique qui n'est pas mixte ($I \& SC$), si l'OG donne exactement la même politique optimale à l'usine₂, la politique optimale à l'OG de l'usine₁ est égale à celle de l'OL.

2.3 Modélisation

2.3.1 Formulation de notre problématique

Nous rappelons qu'avec l'approche classique, nous avons une suite d'optimisations locales de l'aval vers l'amont (optimisation de l'usine₂ puis optimisation de l'usine₁). Au contraire à l'optimisation globale, nous considérons notre système comme un système global et nous l'optimisons globalement en tenant compte de toutes les particularités des deux usines que sont les coûts de production, de sous-traitance, de stockage ainsi que la capacité de production. Notre question initiale est de savoir quel est le bénéfice d'une optimisation globale de cette chaîne logistique par rapport aux optimisations locales successives et de savoir si le bénéfice du système d'une optimisation globale est suffisant et satisfait les besoins des deux usines. L'optimisation

locale et l'optimisation globale de notre système sont représentées dans les figures 2.8 et 2.9.

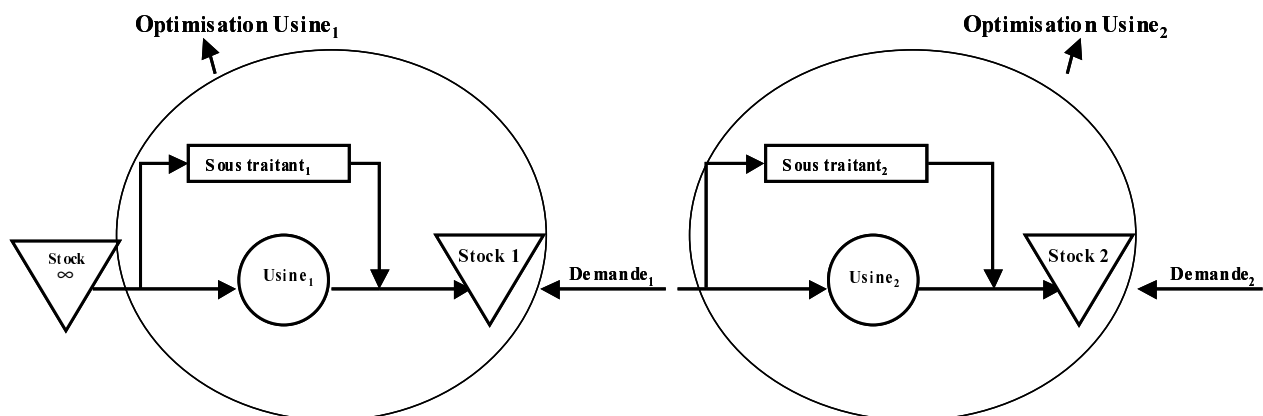


Fig. 2.8 – Optimisation Locale (OL)

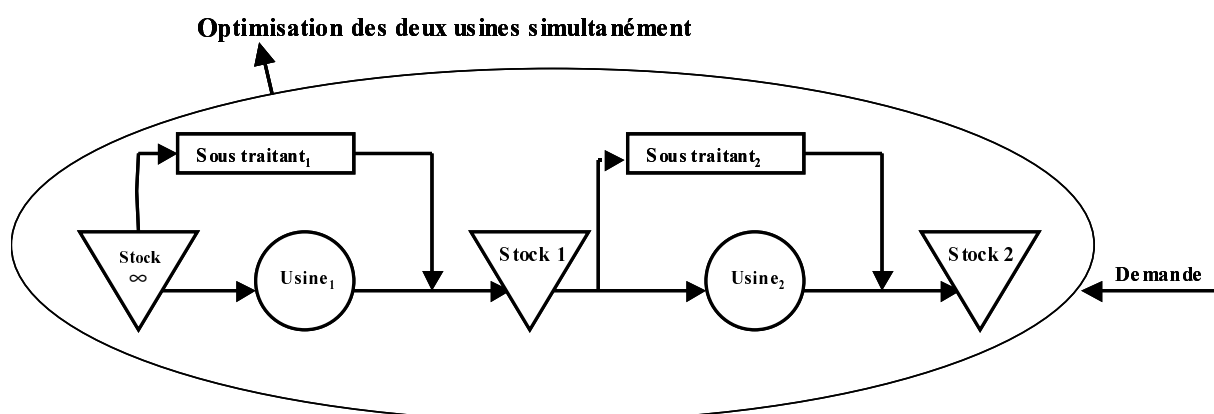


Fig. 2.9 – Optimisation Globale (OG)

Notre travail a comme objectif de développer deux modèles, le premier fondé sur l'optimisation globale (OG) et l'autre sur l'optimisation locale (OL). Les deux modèles conduisent à une optimisation de type programmation linéaire et sont présentés dans le paragraphe suivant. La fonction objectif est de maximiser le profit total qui est le profit de vente des pièces moins la somme des coûts de production, de stockage et de sous-traitance. Les programmes linéaires ont deux variables de décision :

- le taux de production de chaque usine,
- les produits commandés à la sous-traitance.

Au niveau des contraintes nous avons :

- la satisfaction de la demande durant toutes les périodes,
- les capacités de production de chaque usine,
- les contraintes initiales,
- les équations d'équilibre.

Par la suite, nous présentons les formules de deux modèles. Nous optimisons le plan de production de l'usine₂ par le programme linéaire ci-dessous et ensuite, en utilisant ces résultats, nous optimisons le plan de production de l'usine₁ avec le deuxième programme linéaire qui suit.

2.3.2 Les programmes linéaires

Notons qu'au cas déterministe et avec un système où les sous-traitants prennent des pièces du stock amont de l'usine qui sous-traite et avec la contrainte de satisfaire toute la demande, la valeur du profit de vente des pièces est constante. Dès lors, la maximisation du profit total est équivalente à la minimisation du coût de production, de stockage et de sous-traitance. Par la suite, nous donnons les formules de deux modèles avec une fonction objectif qui minimise la somme de coûts.

Modèle d'optimisation globale

Fonction objectif :

$$\min Z = cp_1 \sum_{t=1}^T P_{1,t} + h_1 \sum_{t=1}^T I_{1,t} + csc_1 \sum_{t=1}^T SC_{1,t} + cp_2 \sum_{t=1}^T P_{2,t} + h_2 \sum_{t=1}^T I_{2,t} + csc_2 \sum_{t=1}^T SC_{2,t} \quad (2.1)$$

Sous les contraintes :

Equations d'équilibre :

$$I_{1,t} = I_{1,t-1} + P_{1,t} + SC_{1,t} - P_{2,t} - SC_{2,t} \quad \forall t \in [1, T-1] \quad (2.2)$$

$$I_{2,t} = I_{2,t-1} + P_{2,t} + SC_{2,t} - D_t \quad \forall t \in [2, T] \quad (2.3)$$

Capacité de production :

$$P_{1,t} \leq C_1 \quad \forall t \quad (2.4)$$

$$P_{2,t} \leq C_2 \quad \forall t \quad (2.5)$$

Modèle d'optimisation locale**Modèle d'optimisation locale pour l'usine₁ :****Fonction objectif :**

$$\min Z_1 = cp_1 \sum_{t=1}^T P_{1,t} + h_1 \sum_{t=1}^T I_{1,t} + csc_1 \sum_{t=1}^T SC_{1,t} \quad (2.6)$$

Sous les contraintes :

Equations d'équilibre :

$$I_{1,t} = I_{1,t-1} + P_{1,t} + SC_{1,t} - P_{2,t} - SC_{2,t} \quad \forall t \in [1, T-1] \quad (2.7)$$

Capacité de production :

$$P_{1,t} \leq C_1 \quad \forall t \quad (2.8)$$

Modèle d'optimisation locale pour l'usine₂ :**Fonction objectif**

$$\min Z_2 = cp_2 \sum_{t=1}^T P_{2,t} + h_2 \sum_{t=1}^T I_{2,t} + csc_2 \sum_{t=1}^T SC_{2,t} \quad (2.9)$$

Sous les contraintes :

Equations d'équilibre :

$$I_{2,t} = I_{2,t-1} + P_{2,t} + SC_{2,t} - D_t \quad \forall t \in [2, T1] \quad (2.10)$$

Capacité de production :

$$P_{2,t} \leq C_2 \quad \forall t \quad (2.11)$$

Nous avons utilisé les deux modèles afin de mettre en évidence un certain nombre de comportements qualitatifs et nous avons réussi à les démontrer. Plusieurs analyses ont été réalisées pour pouvoir définir le comportement de chaque modèle afin de pouvoir les comparer et nous avons constaté comment se comportent les deux modèles de manière générale. Notre idée initiale va être illustrée dans le prochain paragraphe par un exemple numérique.

2.4 Exemple numérique

Dans cette section, nous présentons un exemple numérique réel d'une entreprise qui produit des portes et des fenêtres en aluminium. Le nom de cette entreprise est ANALKO et elle se trouve en Grèce. Cette entreprise possède plusieurs usines. Nous nous focalisons sur deux entreprises principales pour lesquelles nous présentons un exemple numérique et pour lesquelles nous appliquons les deux types d'optimisation. La première entreprise produit les composants principaux métalliques pour la production des portes et des fenêtres et la deuxième fabrique les produits finaux et les distribue aux clients.

L'exemple numérique, que nous présentons, correspond à la demande d'une porte nommée type A. Ce produit est le produit qui a la demande la plus importante pendant l'année. La demande utilisée est la demande de cette porte en Europe, pendant une année, multipliée par un nombre positif pour des raisons de confidentialité. Nous suivons la même procédure pour toutes les valeurs utilisées.

La demande est saisonnière et atteint son maximum durant le printemps et son minimum durant l'hiver. Pour qu'elle puisse satisfaire tous ses clients pendant les périodes de grande demande, l'entreprise a trois possibilités pour satisfaire la demande supplémentaire de la capacité de production, soit par la sous-traitance, soit par la consommation des portes produites les mois précédents. Enfin, la troisième possibilité est une combinaison des deux premières possibilités. Les sous-traitants peuvent fabriquer une porte entière ou peuvent faire une des opérations que l'entreprise ne peut réaliser, par exemple la coloration du produit ou le placement des parties vitrées. Les capacités de production sont presque les mêmes pour les deux usines et nous allons les considérer égales et le coût de stockage est calculé de façon à être égal à 5% de coût de production.

En partant de l'approche classique, nous avons une suite d'optimisations de l'aval vers l'amont. L'optimisation locale donne le résultat suivant : en ce qui concerne l'usine₂, elle donne une politique de sous-traitance pour satisfaire la demande et l'usine₁ adopte une politique consistant à garder un stock d'anticipation. Le coût total est de 143115 (61335 pour l'usine₁ et 81780 pour l'usine₂).

	Coût usine ₁	Coût usine ₂	Coût total
Optimisation locale	61335	81780	143115
Optimisation globale	37482.5	88595	126077.5
Différence relative	39%	-8%	12%

TAB. 2.8 – Exemple numérique

Au contraire, dans le cas de l'optimisation globale, tenant compte de tous les coûts du système global, le coût total est 126077.5 et les politiques prises ont changé. L'usine₂ constitue un stock d'anticipation et l'usine₁ suit la courbe de la demande. La différence entre les deux types d'optimisation est égale à 17037.5 soit une réduction de 12%.

2.5 Résultats

Dans cette section, nous présentons les résultats de notre analyse du modèle présenté à la section 2.3.2. Dans un premier temps, nous donnons nos résultats qualitatifs et puis quelques résultats numériques.

2.5.1 Résultats qualitatifs

Nous avons utilisé les deux modèles afin de mettre en évidence un certain nombre de comportements qualitatifs et nous avons réussi à les démontrer (annexe A.1). Premièrement, nous avons démontré qu'à l'optimisation globale, le coût de production pour le système entier est toujours inférieur ou égal au coût de production lors d'une optimisation locale (propr. 1.1). Concernant le plan de production optimal des deux usines, pour l'usine₂ nous l'obtenons à l'optimisation locale (propr. 1.2) et pour l'usine₁ à l'optimisation globale (propr. 1.3).

Propriété 1.4 :

Quand la différence entre le coût de production et le coût de sous-traitance reste la même, les plans qui en résultent sont les mêmes et la différence des solutions optimales (en fonction du coût) entre les deux modèles reste constante. Dans la réalité, le sous-traitant demande souvent un profit standard par l'usine qui n'est pas lié avec le coût de production et ne dépend pas de la tendance du marché. Quand les prix au marché changent, les coûts de production changent mais pas les gains fixes du sous-traitant. Etant donnée cette propriété, nous ne sommes pas obligés de changer le plan de production de notre usine quand il y a un changement au marché.

Propriété 1.5 :

Dans certains cas, nous pouvons éviter une des deux analyses. Quand l'optimisation globale donne comme solution optimale pour l'usine₂ de ne faire que de la sous-traitance et n'importe quelle politique pour l'usine₁, l'optimisation locale donne exactement la même politique optimale. Pour faciliter alors la coordination de notre système, il suffit de faire une seule fois l'optimisation globale, de constater ce type de politique optimale et continuer avec des optimisations locales. Ainsi, nous prenons le plan optimal sans être obligé de transférer un grand nombre d'informations aux autres membres de la chaîne.

Cette propriété est valable pour les raisons suivantes. Si pour l'optimisation globale l'usine₂ ne fait que de la sous-traitance, l'usine₁ obtient la même courbe de la demande (cf. section 2.1). A l'optimisation locale où nous obtenons la meilleure politique pour l'usine₂ (cf. propr. 1.2), dans le pire des cas, nous aurons la même politique obtenue ou une politique mixte. Bien évidemment la satisfaction de la première courbe (optimisation globale) est plus coûteuse que celle de la deuxième (optimisation locale) parce que la quantité supplémentaire à satisfaire est supérieure. Dès lors le coût de production de l'usine₁ pour l'optimisation locale est inférieur ou égal au coût de production pour l'optimisation globale (i.e. $COLU1 \leq COGU1$). Mais par la propriété 1.3, nous avons $COLU1 \geq COGU1$ en conséquence, les deux coûts sont égaux ($COLU1 = COGU1$). Finalement, par les propriétés 1.1 et 1.2 nous montrons que ($COLU2 = COGU2$).

Propriété 1.6 :

La situation est analogue pour le cas où l'optimisation locale pour l'usine₂ donne la politique de satisfaction de la demande supplémentaire avec un stock d'anticipation, l'optimisation globale donne exactement la même politique optimale.

Si pour l'optimisation locale l'usine₂ ne fait que du stock, l'usine₁ obtient la meilleur courbe de demande qu'elle peut obtenir. Pour l'optimisation globale, au meilleur cas pour l'usine₁, nous obtenons la même politique ou une politique mixte pour l'usine₂. Si nous obtenons le cas de la politique mixte, le coût d'optimisation globale de l'usine₁ est inférieur ou égal au coût d'optimisation locale (i.e. $COGU1 \geq COLU1$). Mais par la propriété 1.3, $COGU1 \leq COLU1$. En conséquence, les deux coûts sont égaux ($COGU1 = COLU1$). Finalement, par les propriétés 1.1 et 1.2, nous montrons que le coût d'optimisation locale de l'usine₂ est égal au coût d'optimisation globale ($COLU2 = COGU2$).

2.5.2 Les résultats numériques et la différence relative entre les deux modèles

Dans cette section, nous décrivons des exemples numériques. Ce que nous avons constaté est l'augmentation de la différence relative quand nous augmentons le coût unitaire de la sous-traitance pour la deuxième usine. Au début, en gardant tous les coûts constants, nous avons commencé par un coût csc_2 égal à zéro, pour trouver les espaces numériques dans lesquels la politique obtenue optimale pour l'usine₂ serait constante. Nous avons trouvé les espaces ci-dessous :

- $csc_2 \leq c_1$ nous avons une politique de sous-traitance (dans notre exemple $c_1 = 150$),
- $c_1 + 1 \leq csc_2 \leq c_2$ nous avons une politique mixte (dans notre exemple $c_2 = 549$),
- $c_2 + 1 \leq csc_2$ nous avons une politique de production par anticipation.

Dans le premier espace, le coût de la sous-traitance est très faible et en conséquence, l'usine₂ obtient toujours la même politique optimale (pour les deux types d'optimisation). Cette politique est la politique de satisfaction de la demande supplémentaire seulement en sous-traitant. Avec cette politique adoptée, l'usine₁ suit toujours la même courbe de demande qui est celle du produit final. Gardant les coûts de l'usine₁ constants, le plan optimal de l'usine₁ reste toujours le même. Dans le deuxième espace, le csc_2 devient plus cher que le coût de production. Pour cette raison, la politique change et devient une politique mixte. Dans ce deuxième espace, nous pouvons avoir toutes les combinaisons des politiques. Cette situation est mise en évidence par les fluctuations de la courbe présentée à la figure 2.10. Finalement, à partir de c_2 , le coût de la sous-traitance devient important et le plan optimal est de satisfaire la demande par un stock d'anticipation. Dans ce cas, les deux types d'optimisation n'ont pas de différences (*propr.1.6*). De plus, nous avons constaté, d'après nos exemples, que quand la différence entre le coût de

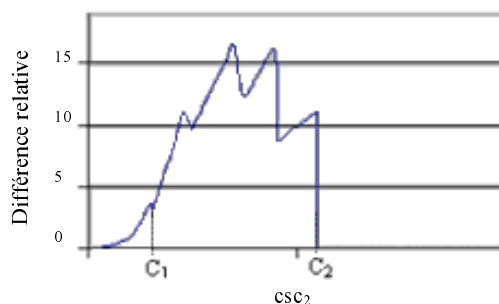


Fig. 2.10 – Variation de csc_2

production et le coût de la sous-traitance reste la même, nous avons comme résultat le même plan de production (une chose qui arrive pour les deux méthodes d'optimisation) : par ailleurs, la différence entre les deux modèles reste identique (*propr.1.4*). Nous présentons tout d'abord nos exemples numériques et dans l'annexe A.1 la démonstration. Les coûts initiaux utilisés sont présentés à la première ligne du tableau ci dessous. Puis sur les lignes, nous présenterons d'autres exemples où nous avons gardé la différence entre le coût de production et le coût de la sous-traitance constante.

h_1	cp_1	csc_1	Δ	h_2	cp_2	csc_2	Δ
5	90	190	100	10	100	210	110
5	100	200	100	10	150	260	110
5	90	190	100	10	120	230	110
5	1000	1100	100	10	100	210	110
5	90	190	100	10	1000	1110	110

TAB. 2.9 – Le coûts

Pour tous ces coûts, nous obtenons le même plan de production optimal donné dans le tableau suivant :

périodes	Stock :		Production :		Sous-traitance :	
	Usine ₁	Usine ₂	Usine ₁	Usine ₂	Usine ₁	Usine ₂
1	0	0	60	0	0	0
2	40	0	100	60	0	0
3	40	0	100	60	0	0
4	40	30	100	100	0	0
5	40	40	100	100	0	0
6	40	30	100	100	0	0
7	20	10	100	100	0	0
8	0	0	100	100	0	20
9	0	0	100	100	0	20
10	0	0	80	100	0	0
11	0	0	60	80	0	0
12	0	0	0	60	0	0

TAB. 2.10 – Le plan de production optimal

2.6 Généralisation au cas de participations croisés des entreprises dans les deux usines

Généralement l'optimisation globale donne toujours une meilleure solution même si elle pénalise parfois les sous-systèmes. Dans la réalité, seulement un pourcentage des actions de chaque usine appartient à une entreprise. Dans l'exemple de l'entreprise ANALKO, l'entreprise détient un pourcentage des actions de chaque usine. Le pourcentage des actions possédé par l'entreprise est entre 10% et 100%. Les gestionnaires peuvent alors décider du type d'optimisation, choisissant le coût de production le plus élevé pour l'usine ou elle détient le moins d'actions. Si par exemple elle détient moins d'actions pour l'usine₁, ils peuvent décider de faire une optimisation locale, qui est plus favorable pour l'usine₂, même si cette optimisation donne de moins bons résultats pour le système entier. En effet, l'optimisation locale peut être plus avantageuse dans ce cas pour l'entreprise dans la mesure où le coût de production de l'usine₂ minimisé à l'optimisation locale est suffisamment bas compte tenu des pourcentages d'actions détenues par l'entreprise.

Exemple : Afin d'illustrer ces propos, nous présentons maintenant un exemple numérique simple. Nous supposons être en présence de deux entreprises E1 et E2. L'entreprise E1 possède 50% de l'usine₁ et n'a pas de parts de l'usine₂. L'entreprise E2 possède quant à elle 100% de l'usine₂ et les 50% restants de l'usine₁. Nous supposons enfin, étant donnée cette distribution, que l'entreprise E2 décide de l'optimisation (locale ou globale) qui sera menée. Nous appelons ici COLU1, COLU2, COGU1, COGU2 coûts de production des usines 1 et 2 respectivement à l'optimisation locale et à l'optimisation globale. Nous supposons que COLU1=200, COLU2=30, COGU1=100, COGU2=120. Quelle stratégie va adopter l'entreprise E2 qui décide de l'optimisation qui sera réalisée ? Nous avons d'après toutes ces données le tableau suivant des coûts de production (les données du problème sont en italique) où COLE2, COGE2 sont les coûts à payer par l'entreprise₂ à l'optimisation locale et à l'optimisation globale respectivement :

	Usine ₁	Usine ₂	Coût total
OG	$COGU1 = 100$	$COGU2 = 120$	$COG = 220$
OL	$COLU1 = 200$	$COLU2 = 30$	$COL = 230$
E2 à l'OG	Coût à payer = 50	Coût à payer = 120	$COGE2 = 170$
E2 à l'OL	Coût à payer = 100	Coût à payer = 30	$COLE2 = 130$

TAB. 2.11 – Exemple numérique

L'optimisation globale est en effet plus performante pour le système entier que l'optimisation locale : le coût global de production à l'optimisation globale est de 220 et celui à l'optimisation locale est de 230. En revanche, étant données les parts de l'entreprise E2 dans chacune des usines, le coût global de production pour E2 est de 170 à l'optimisation globale et de 130 à l'optimisation locale : en conséquence, si l'on regarde du point de vue de E2, l'optimisation locale donne de meilleurs résultats pour elle. L'entreprise E2 qui détermine le type d'optimisation choisi est donc susceptible de vouloir une optimisation locale, plus avantageuse pour elle, quoique moins avantageuse pour le système global.

Mais si les parts dans l'usine₁ avaient été de 95% pour E2 et 5% pour E1, le coût global pour E2 aurait été à l'optimisation globale de 215 et à l'optimisation locale de 220 : E2 aurait donc cette fois-ci choisi l'optimisation globale, meilleure solution pour elle et pour le système entier. Ceci nous laisse à penser qu'il existe des conditions limites sur les parts de E2 dans chacune des entreprises (dépendant a priori des coûts de production COLU1, COLU2, COGU1, COGU2) au-delà desquelles E2 choisira l'optimisation locale plutôt que l'optimisation globale.

Pour généraliser l'idée présentée dans les paragraphes précédents, nous considérons deux entreprises, l'entreprise E1 et l'entreprise E2. E1 détient le $(1-\alpha)\%$ de l'usine₁ et $\beta\%$ de l'usine₂. Au contraire, E2 détient le $\alpha\%$ de l'usine₁ et $(1-\beta)\%$ de l'usine₂. Les coûts de production pour les deux usines sont donnés par les fonctions suivantes :

- $COLE1 = \beta \times COLU2 + (1-\alpha) \times COLU1$,
- $COGE1 = \beta \times COGU2 + (1-\alpha) \times COGU1$,
- $COLE2 = (1-\beta) \times COLU2 + \alpha \times COLU1$,
- $COGE2 = (1-\beta) \times COGU2 + \alpha \times COGU1$.

Ensuite, nous donnons trois exemples : les plus réalistes et plus représentatifs. Premièrement, si $\alpha = \beta = 0$, l'E1 possède le 100% de l'usine₁ et l'E2 le 100% de l'usine₂. Dans ce cas, l'optimisation locale est préférable pour l'E2 (*propr.* 1.2) et l'optimisation globale pour l'E1 (*propr.* 1.3). Ensuite, nous prenons un exemple typique ou une entreprise (ex. E2) possède le 100% d'une usine (ex. $\beta = 0$) et le $\alpha\%$ d'une autre usine, nous avons des cas où il sera mieux pour l'entreprise de faire une optimisation locale au lieu de faire une optimisation globale, et d'autre

cas où le contraire arrive. Pour que l'optimisation globale soit préférable pour l'E2, il faut que :

$$0 \leq \alpha < \left(\frac{COGU2 - COLU2}{COLU1 - COGU1} \right) \quad \text{propr. 1.7}$$

En utilisant la propr. 1.7 (annexe A.1) et pour l'exemple numérique présenté à la paragraphe 2.4, nous prenons le graphe 2.11. Nous constatons que pour tout α inférieur à 30%, l'optimisation locale doit être recommandée par rapport à l'optimisation globale. Après cette valeur de α , l'E2 adopte l'optimisation globale.

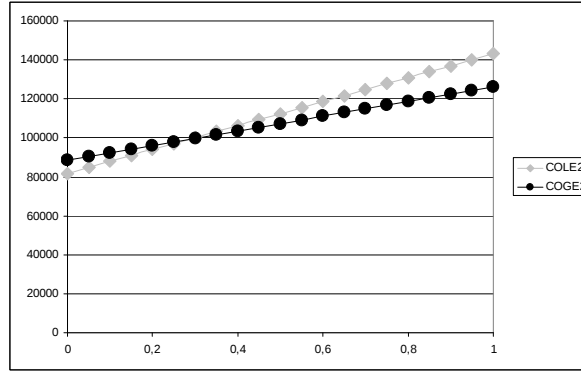


Fig. 2.11 – Optimisation locale vs globale

Finalement, si nous avons deux entreprises E1 et E2 avec $\alpha \neq 0$ et $\beta \neq 0$ nous avons les limites suivantes :

- Pour E2 l'OL est préférable à l'OG si $\frac{\alpha}{(1-\beta)} < \frac{(COGU2-COLU2)}{(COLU1-COGU1)}$ (propr. 1.8, annexe A.1),
- Pour E2 l'OG est préférable à l'OL si $\frac{\alpha}{(1-\beta)} > \frac{(COGU2-COLU2)}{(COLU1-COGU1)}$ (propr. 1.9, annexe A.1),
- Pour E1 l'OL est préférable à l'OG si $\frac{(1-\alpha)}{\beta} < \frac{(COGU2-COLU2)}{(COLU1-COGU1)}$ (propr. 1.10, annexe A.1),
- Pour E1 l'OG est préférable à l'OL si $\frac{(1-\alpha)}{\beta} > \frac{(COGU2-COLU2)}{(COLU1-COGU1)}$ (propr. 1.11, annexe A.1).

Nous comprenons que généralement l'optimisation globale donne un coût inférieur ou égal à l'optimisation locale pour le système, mais la décision du choix du type d'optimisation est aussi lié à plusieurs sujets financiers. L'entreprise qui a la majorité des actions est en meilleure position. Cette entreprise peut obtenir l'optimisation la plus profitable pour elle. Enfin, elle peut prévoir sa perte probable par une optimisation globale et mieux négocier avec l'autre entreprise.

Chapitre 3

Optimisation locale vs globale : cas stochastique

3.1 Introduction

Dans le cas stochastique et pour l'optimisation de la production, les gestionnaires pilotent le flux des matières et des produits. Le pilotage de flux est l'ensemble des règles de gestion des flux physiques de la chaîne logistique. Ainsi, piloter les flux consiste à prendre des décisions, à chaque étape de la chaîne et pour chaque entité, afin de déterminer quand et dans quelle mesure lancer une activité.

A partir de cette définition, la représentation adoptée consiste en effet à des noeuds de prise de décision indépendamment de la nature de l'opération à effectuer. Ainsi, une activité de la chaîne logistique peut être une succession d'activités élémentaires de production, de transport, tant que ces activités se succèdent sans la nécessité d'une décision.

Généralement, le nombre de points de décision augmente la complexité du pilotage mais aussi la qualité du contrôle. Il est donc primordial de distinguer les points de la chaîne où le contrôle de flux est le plus important, notamment en terme d'impact sur le coût du système (activité à forte valeur ajoutée) ou en terme de contraintes techniques (activité à forte différenciation, ex. espace de stockage limité).

La mise en place d'un système de pilotage de flux consiste donc, d'une part, à déterminer la localisation des points de contrôle du flux et, d'autre part, à choisir les règles de pilotage, c'est-à-dire les conditions de passage de l'entité à l'étape suivante. Dans notre étude, les points de contrôle correspondant au lancement en production ou en sous-traitance pour chaque usine.

Dans le cas stochastique, nous avons utilisé le système présenté dans la figure 3.1. Les hypothèses supplémentaires pour ce deuxième cas sont premièrement la permission d'avoir des

demandes en attente à l'usine₂. Deuxièmement, si le stock est vide et s'il y a une demande, le sous-traitant peut fournir immédiatement le stock de l'usine. Pour avoir un modèle suffisamment simple nous représentons de manière simplifiée chaque usine par une ressource unique. Le processus d'arrivée des clients est un processus de Poisson et le temps de traitement dans chaque usine suit une distribution exponentielle. Finalement, une hypothèse simplificatrice pour faciliter la modélisation présentée dans les chapitres qui suivent, est que la pièce en traitement à l'usine₂ est toujours considérée comme appartenant au stock de l'usine₁.

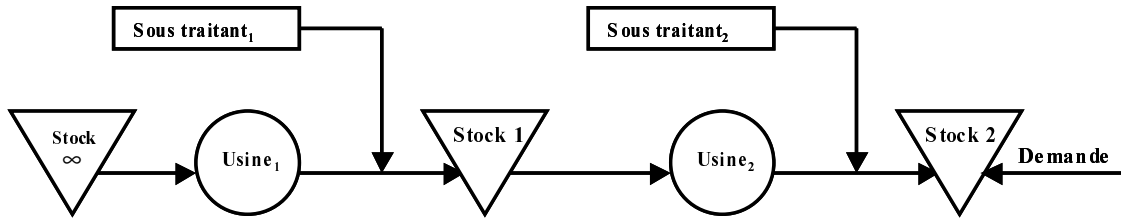


Fig. 3.1 – Système Étudie-Cas Stochastique

Les paramètres et les coûts de notre système, sont comme suit :

Les paramètres :

- μ_i : le taux de production à l'usine_{*i*},
- λ : le taux des arrivées des nouveaux clients (processus d'arrivée Poissonnier),
- μ_{SCi} : le taux de production du sous-traitant_{*i*} qui est toujours pris infini.

Les coûts :

- p_0 : le prix d'achat des matières premières pour l'usine₁,
- p_i : le prix de vente d'une pièce par l'usine_{*i*},
- cp_i : le coût de production d'une pièce à l'usine_{*i*},
- h_i : le coût de stockage d'une pièce à l'usine_{*i*},
- cb : le coût d'une commande en attente d'une pièce, à l'usine₂,
- csc_i : le coût d'une pièce sous-traitée par le sous-traitant_{*i*}.

3.2 Politiques de contrôle

Pour le cas stochastique les politiques considérées sont de type : capacité finie (Finite Buffer, FB), stock nominal (Base Stock, BS), installation (Installation Stock, IS) et échelon (Echelon Stock, ES). La combinaison de ces quatre cas donnent suite à quatre politiques de pilotage de flux :

- la politique FBIS (Finite Buffer Installation Stock),
- la politique FBES (Finite Buffer Echelon Stock),
- la politique BSIS (Base Stock Installation Stock),

- la politique BSES (Base Stock Echelon Stock).

Il existe plusieurs politiques de pilotage de flux dans la littérature. L'objectif de notre recherche n'est pas de comparer ces différentes politiques mais d'examiner chaque fois quel est le gain d'une optimisation globale par rapport à une optimisation locale avec la politique adoptée. Avant de présenter ces politiques, nous donnons la définition d'un terme que nous avons introduit et utilisé dans notre recherche. Ce terme est le terme **D**emande en **A**ttente **P**artielle (**DAP**). Ce terme est défini comme la politique où nous permettons des demandes en attente mais jusqu'à un niveau au delà duquel l'usine commence à sous-traiter. L'idée est de chercher quel est le nombre maximal de demandes en attente après lequel il est désavantageux de permettre d'avoir d'autres demandes en attente.

Généralement nous avons deux politiques possibles concernant les demandes en attente :

- soit on n'autorise pas les demandes en attente,
- soit on autorise les demandes en attente.

Dans le cas où les demandes en attente ne sont pas permises, si une demande arrive et le stock de l'usine est vide, cette dernière soit est perdue (ce qui donne lieu à une pénalité), soit est satisfaite immédiatement par le sous-traitant. Au contraire, dans le cas de la politique dont les demandes en attente sont permises, nous pouvons avoir deux possibilités différentes : soit avoir de demandes en attente infinies, soit avoir des demandes en attente partielles. Concernant la première possibilité, si le stock est vide et des nouvelles demandes arrivent, ces dernières sont mises en attente. Dans ce cas, l'usine ne sous-traite pas. Par rapport à la deuxième possibilité, si de nouvelles demandes arrivent alors que le stock est vide, ces dernières sont mises en attente jusqu'à un niveau au delà duquel si une nouvelle demande arrive, soit le sous-traitant fournit immédiatement une pièce pour satisfaire la première demande en attente, soit cette demande est perdue (ce qui donne lieu à une pénalité). Cette politique est la politique des demandes en attente partielle (DAP).

Dans notre système, nous avons adopté la politique de DAP concernant les demandes en attente. Nous trouvons que cette politique est plus générale et plus réaliste que les politiques qui permettent d'avoir des demandes en attente infinies ou qui ne permettent pas d'accepter des demandes en attente. Dans l'hypothèse de satisfaire toute la demande, si une nouvelle demande arrive et le nombre de demandes en attente est égal au nombre maximal de demandes en attente permis, le sous-traitant fournit une pièce pour satisfaire la première demande en attente.

Dans toutes les politiques de pilotage de flux adoptées, les variables de décision et les variables d'état sont les suivantes :

Les variables de décision :

- B_i : le niveau maximum du stock à l'usine $_i$,
- DAP : le niveau maximum des demandes en attente à l'usine $_2$.

Les variables d'état :

- I_i : le niveau de stock à l'usine $_i$,
- BL : le niveau de demandes en attente à l'usine $_2$.

Généralement, la politique adoptée n'influence pas la règle de production à l'usine $_2$. L'usine $_2$ produit si son stock est inférieur à B_2 et permet des demandes en attente supplémentaires si $BL \leq DAP$. Les différences parmi les politiques se trouvent au niveau de l'usine $_1$. La politique la plus simple de pilotage de flux est la politique de FBIS (Finite Buffer Installation Stock). Pour cette politique l'usine $_1$ produit si $I_1 \leq B_1$. Pour la politique BSIS, la seule différence est que l'usine $_1$ produit si le niveau de son stock est inférieur à $B_1 - DAP$.

La grande différence entre la politique installation et la politique échelon est la considération ou pas du stock de l'usine $_2$ dans la décision de pilotage de l'usine $_1$. Si nous considérons que le stock de l'usine $_1$ ne possède que son stock local, nous avons adopté une politique du type installation. Au contraire, si nous considérons comme stock la somme du stock local de l'usine $_1$ et du stock qui existe à l'usine $_2$, nous avons adopté une politique du type échelon. La politique de FBES est alors une politique où nous avons considéré comme stock $_1$ le stock réel de l'usine $_1$ plus le stock réel de l'usine $_2$. L'usine produit si la somme de ces stocks est inférieure à $B_1 + B_2$. Enfin, nous avons la politique BSES où l'usine $_1$ produit si et seulement si la somme de ces stocks est inférieure à $B_1 + B_2 - DAP$. Nous remarquons que si nous ne permettons pas de demandes en attente ($DAP=0$), les politiques FBIS et BSIS sont identiques, de même que les politiques FBES et BSES.

Dans les chapitres suivants, nous nous sommes concentrés sur deux de ces quatre politiques : la politique FBIS et la politique BSES, l'une du type capacité finie et l'autre au contraire de type échelon stock. Pour davantage d'explications sur les politiques de pilotage de flux, nous pouvons nous référer aux articles [Liberopoulos and Dallery, 2000] et [Liberopoulos and Dallery, 2003] où toutes les politiques de pilotage de flux sont présentées et où plusieurs études et comparaisons ont été faites.

3.3 Modélisation

Pour toutes les politiques adoptées, nous nous intéressons à définir le niveau de stock optimal B_1 , B_2 et le nombre de DAP (**Demandes en Attende Partielles**) de l'usine₂, tout en maximisant le profit de notre chaîne logistique. Afin de définir le plan de production optimal, le système étudié est modélisé en utilisant les chaînes de Markov.

Dans le cas le plus simple où les deux usines adoptent la politique de FBIS sans demande en attente, équivalente à la politique BSIS, le système est modélisé de la façon suivante : chaque état est décrit par un couple (I_1, I_2) où $I_i = 1, 2, \dots, B_i$ ($\forall i = 1, 2$) représente le niveau de stock de l'usine _{i} . Plus spécifiquement, I_1 décrit les pièces existant dans le stock₁ et I_2 décrit les pièces existant dans le stock des produits finis de l'usine₂. Les deux usines continuent leur production jusqu'à ce que $I_i = B_i$ ($\forall i = 1, 2$). Les transitions entre les états de la chaîne de Markov sont décrites par les taux de production μ_1 , μ_2 de chaque usine et le taux de la demande du produit fini λ .

Dans la figure suivante, (cf. figure 3.2) nous présentons la chaîne de Markov complète qui correspond au système étudié où la politique FBIS est adoptée. La transition de l'état I_1 vers l'état $I_1 + 1$ a lieu avec un taux μ_1 , de l'état (I_1, I_2) vers l'état $(I_1 - 1, I_2 + 1)$ a lieu avec un taux μ_2 et de I_2 vers $I_2 - 1$, λ (si c'est possible). L'état $(0, B_2)$ est un cas particulier. Nous pouvons aller à cet état seulement à partir de l'état $(1, B_2 - 1)$, c'est à dire quand l'usine₂ produit sa dernière pièce avant de remplir son stock nominal. Dans ce cas, le stock de l'usine₂ qui est égal à I_2 devient B_2 et l'usine₁ n'a plus besoin d'une pièce immédiatement par le sous-traitant₁, c'est pourquoi elle commence sa production avec un taux μ_1 . De l'état $(0, B_2)$ le système peut effectuer la transition vers l'état $(1, B_2)$ avec un taux égal à μ_1 ou vers $(1, B_2 - 1)$ si l'usine₂ a besoin d'une pièce et l'usine₁ achète une pièce de son sous-traitant₁. Enfin, la transition entre l'état $(1, I_2)$ vers $(1, I_2 + 1)$ a lieu avec un taux égal à μ_2 et avec un coût égal à csc_1 . L'usine₁ sous-traite immédiatement pour qu'elle puisse avoir une pièce disponible pour l'usine₂ qui n'est pas encore arrivée à son stock maximal. Finalement, quand I_2 est égal à zéro et qu'il y a un nouveau client (λ), l'usine₂ sous-traite pour qu'elle puisse satisfaire immédiatement cette nouvelle demande.

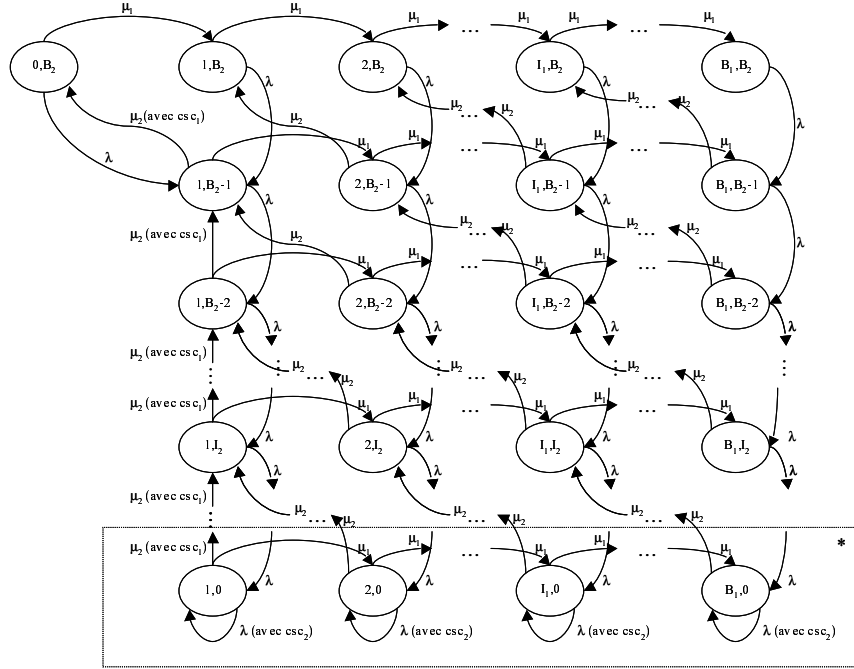


Fig. 3.2 – FBIS sans DAP

Si, au lieu de la politique FBIS sans demandes en attente partielles, nous adoptons la politique FBIS avec DAP, la partie encadrée (*) est remplacée par (cf. figure 3.3) :

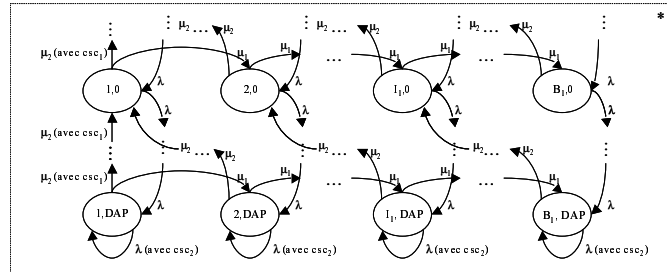


Fig. 3.3 – Modification de FBIS avec DAP

La chaîne complète après avoir ajouté la possibilité d'avoir aussi DAP est décrite par le schéma suivant (cf. figure 3.4) :

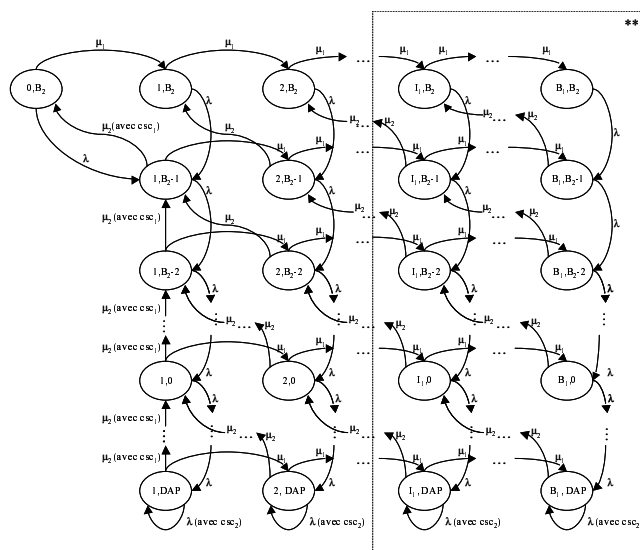


Fig. 3.4 – FBIS avec DAP

Au contraire, pour la politique BSES, la chaîne de Markov est complètement différente. La partie encadrée (**) de la figure précédente (cf. figure 3.4) est remplacée par la chaîne suivante :

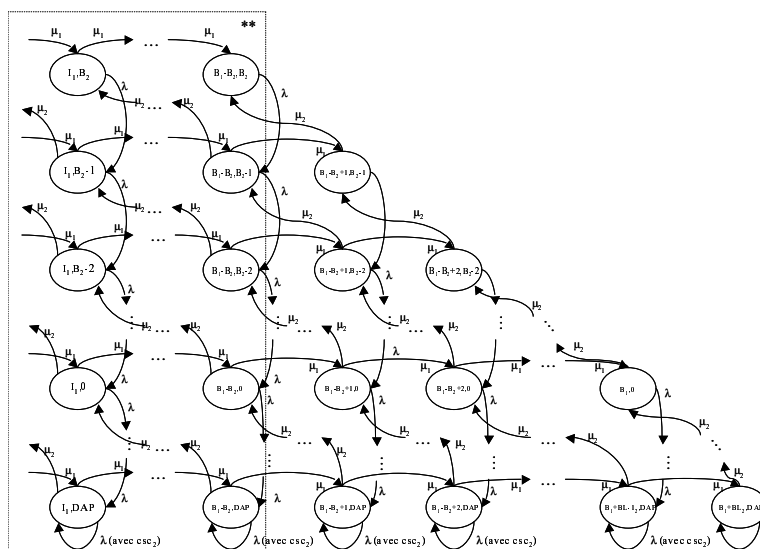


Fig. 3.5 – Modification de BSES avec DAP

Avant de présenter l'algorithme développé pour l'obtention de B_1 , B_2 et DAP optimaux, nous éclaircissons notre modélisation, à travers un exemple numérique. Nous présentons la chaîne avec $B_1=3$, $B_2=3$ et DAP=1 pour les deux politiques que nous venons de présenter (FBIS, BSES avec DAP, cf. figures 3.6 et 3.7).

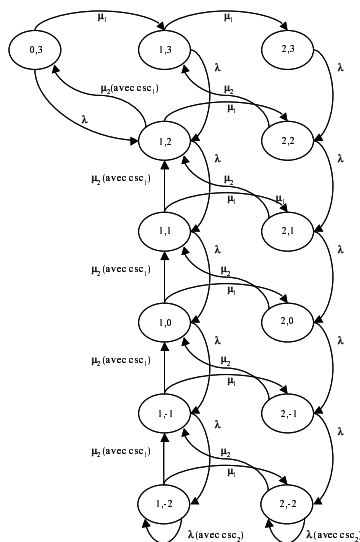


Fig. 3.6 – FBIS, $B_1=2$, $B_2=3$, DAP=2

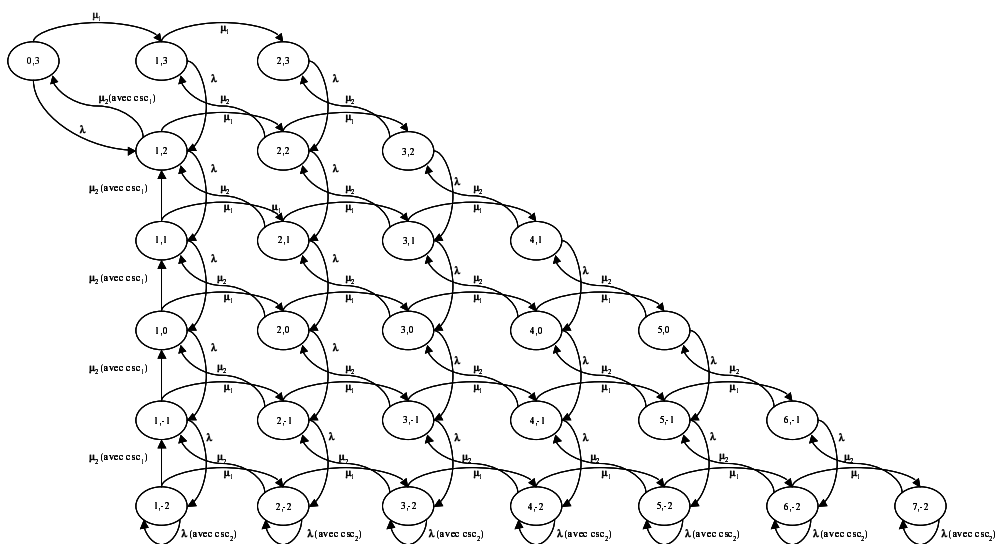


Fig. 3.7 – ESBS, $B_1=2$, $B_2=3$, DAP=2

3.4 Algorithme

Nous avons développé l'algorithme suivant en adoptant la méthode de [Yeralan and Muth, 1987], lequel définit efficacement le B_1, B_2 et DAP. Nous définissons $P(I_1, I_2)$ comme étant la probabilité pour que notre système se trouve à l'état (I_1, I_2) et la fonction $\delta_{(x)} = 1$ si x est vrai, sinon $\delta_{(x)} = 0$. La probabilité est calculée en résolvant l'équation de Chapman-Kolmogorov :

$$\begin{aligned}
 P(I_1, I_2) \times [\mu_1 \times \delta_{(1 < I_1 < B_1)} + \mu_2 \times \delta_{(I_2 < B_2)} + \lambda \times \delta_{(I_2 > 0)}] = \\
 P(I_1 - 1, I_2) \times [\mu_1 \times \delta_{(I_1 > 1)}] \\
 + P(I_1 + 1, I_2 - 1) \times [\mu_2 \times \delta_{(1 < I_1 < B_1), (I_2 > 0)}] \\
 + P(1, I_2 - 1) \times [\mu_2 \times \delta_{(I_1 = 1), (0 < I_2 < B_2)}] \\
 + P(I_1, I_2 + 1) \times [\mu_2 \times \lambda \times \delta_{(0 < I_2 < B_2)}]
 \end{aligned} \tag{3.1}$$

et pour l'état extrême $(0, B_2)$:

$$P(0, B_2) \times (\lambda + \mu_1) = P(1, B_2 - 1) \times (\mu_2) \tag{3.2}$$

Nous définissons aussi le vecteur suivant :

$$P(I_2) = \begin{bmatrix} P(1, I_2) \\ P(2, I_2) \\ \vdots \\ P(I_1, I_2) \end{bmatrix} \tag{3.3}$$

Les équations présentées peuvent s'exprimer sous la forme matricielle suivante. Les matrices A_0, A_1, A, C_0, C, B_1 et B sont des matrices décrivant les taux des transitions à travers les états du système, donnant un I_2 fixe :

$$P(0) \times A_0 = P(1) \times C_0 \tag{3.4}$$

$$P(I_2) \times A = P(I_2 - 1) \times B + P(I_2 + 1) \times C, \quad \forall I_2 = 1, \dots, B_2 - 1 \tag{3.5}$$

$$P(B_2) \times A_1 = P(B_2 - 1) \times B_1 \tag{3.6}$$

De plus, nous avons l'équation supplémentaire suivante :

$$\sum_{I_2=1}^{B_2} \sum_{I_1=1}^{B_1} P(I_1, I_2) = 1 \tag{3.7}$$

Notre algorithme passe par trois étapes pour trouver tous les $P(I_2)$. Premièrement nous résolvons l'équation 3.4 et exprimons ensuite toutes les équations en fonction de $P(0)$. Après, nous faisons la somme des équations 3.4-3.6 et nous utilisons l'équation 3.7 pour trouver $P(0)$. Finalement, nous utilisons la valeur obtenue de $P(0)$ pour trouver tous les $P(I_2)$.

La mesure de performance de notre système est le profit total de la chaîne logistique. Nous avons choisi le profit total parce que de cette manière, on peut étudier profondément le comportement de l'usine₁ face au deux types d'optimisation et l'influence de l'usine₂ sur elle. Le profit dépend des ventes, des coûts de stockage, des coûts de production et de coûts de sous-traitance (cf. section 3.1). Nous définissons le TH_i comme le débit de production des pièces à l'usine_i et $SCTH_i$ le débit de sous-traitance_i. Nous considérons alors que $\lambda = TH_2 + SCTH_2$, grâce à la satisfaction de toute la demande, soit par la production, soit par la sous-traitance. De même, $TH_2 = TH_1 + SCTH_1$ dans la mesure où la production d'une pièce à l'usine₂ nécessite une pièce de l'usine₁ ou venant de la sous-traitance₁. Les quantités $TH_i, SCTH_i$ et aussi le coût de stockage $H_i \forall i$, peuvent être calculés par les probabilités d'état. Par exemple, $SCTH_2$ est le coût de fonctionnement durant la période pendant laquelle le stock de l'usine₂ est vide. Le $SCTH_2$ peut être calculé par l'équation suivante :

$$SCTH_2 = \lambda \times P(I_2 = 0) \quad (3.8)$$

et le taux de production est :

$$TH_2 = \lambda - SCTH_2 \quad (3.9)$$

De même, pour l'usine₁ :

$$TH_1 = \mu_1 \times P(I_1 < b_1) \quad (3.10)$$

et le taux de la sous-traitance est :

$$SCTH_1 = TH_2 - TH_1 \quad (3.11)$$

Les coûts de stockage sont calculés par les deux équations suivantes :

$$H_1 = \sum_{I_2=1}^{B_2} \sum_{I_1=1}^{B_1} I_1 \times P(I_1, I_2) \quad (3.12)$$

$$H_2 = \sum_{I_2=1}^{B_2} \sum_{I_1=1}^{B_1} I_2 \times P(I_1, I_2) \quad (3.13)$$

Le profit total est donné par l'équation suivante :

$$J(b_1, b_2) = \lambda \times p_2 - \sum_{i=1}^2 TH_i \times c_i - \sum_{i=1}^2 SCTH_i \times csc_i - \sum_{i=1}^2 H_i \times h_i \quad (3.14)$$

Le couple optimal (b_1, b_2) est trouvé par la recherche exhaustive de toutes les combinaisons $0, 1, \dots, K_{1max} \times 0, 1, \dots, K_{2max}$ où K_{1max}, K_{2max} sont les bornes supérieures de notre recherche. Cette fonction nous donne le profit d'optimisation globale.

Nous supposons maintenant que les deux usines adoptent une politique d'optimisation locale. L'usine₂ définit spécifiquement son stock B_2 optimal qui lui permet d'avoir le maximum de profit. Après son optimisation locale, elle donne l'information de son stock nominal à l'usine₁ qui ensuite définit son stock nominal optimal B_1 . Cette stratégie appelée optimisation locale est une politique décentralisée.

Le profit individuel de chaque usine est calculé de la façon suivante. En supposant qu'il y a toujours des pièces disponibles pour l'usine₂, nous pouvons modéliser le système comme une chaîne $M/M/1/B_2$. Cette modélisation est valable parce que le processus d'arrivée des clients est un processus de Poisson de taux λ et le temps de service dans l'usine suit une distribution exponentielle de paramètre μ_2 . Nous définissons :

$$\rho = \mu_2 / \lambda \quad (3.15)$$

et la probabilité est donnée par l'équation suivante :

$$P(I_2) = (1 - \rho) \times \rho^{I_2} / (1 - \rho^{1+B_2}) \forall I_2 = 0, 1, \dots, B_2 \quad (3.16)$$

L'optimum local B_2 est le stock qui maximise le profit et minimise le coût de production, de stockage et de sous-traitance. Ce profit est donné par l'équation suivante :

$$J_2(b_2) = \lambda \times p_2 - \lambda \times P(0) \times csc_1 - \lambda \times (1 - P(0)) \times (p_1 + c_1) - \sum_{I_2=1}^{B_2} I_2 \times P(I_2) \times h_2 \quad (3.17)$$

La fonction (3.17) est une fonction concave [Susan and Shanthikumar, 2001] et elle a un seul optimum qui peut être défini facilement avec *trial-and-error*. Ensuite, à partir de cette optimisation où nous obtenons la valeur B_2 , nous pouvons trouver l'optimum du stock de l'usine₁ en maximisant son propre profit :

$$J_1(b_1) = TH_2 \times p_1 - TH_1 \times c_1 - SCTH_1 \times csc_1 - H_1 \times h_1 \quad (3.18)$$

où $TH_1, TH_2, SCTH_1$ et H_1 sont exprimés en fonction des probabilités $P(I_1, I_2)$ qui sont

issus du calcul effectué pour l'optimisation globale, en fixant le B_2 à sa valeur optimale (obtenue par l'équation 3.17). Finalement, le profit de la chaîne pour une optimisation locale est donné par l'équation (3.14) en utilisant les optima locaux B_1 et B_2 de cette optimisation. Enfin, nous remarquons que la maximisation du profit tient aussi compte du terme TH_2 , ce qui implique l'influence de l'usine₂ sur l'usine₁. De plus, le sous-traitant₂ ne prend pas de pièces par le stock de l'usine₁. Dès lors, la simplification obtenue en minimisant le coût de production, de stockage et de sous-traitance adoptée au cas déterministe n'est plus valable.

3.5 Résultats

3.5.1 Valeurs des paramètres

Dans les différentes analyses que nous effectuons, les paramètres de notre système seront définis par le vecteur $\vec{u} = (cp_1, cp_2, p_0, p_1, p_2, csc_1, csc_2, h_1, h_2, cb_2, \lambda, \mu_1, \mu_2)$.

Paramètres de coût :

Le tableau 3.1 fournit les valeurs nominales ainsi que les intervalles de variation des paramètres du système considéré. Les valeurs nominales choisies sont similaires à des valeurs de paramètres de coût qu'on rencontre dans la littérature. Dans nos analyses numériques, pour évaluer l'impact d'un paramètre sur le comportement du système, nous ferons varier un paramètre donné dans son intervalle de variation alors que les autres paramètres seront égaux à leurs valeurs nominales.

Paramètre	Valeurs nominales	Intervalle de variation
cp_1	50	$cp_1 \in [25, 26, 27, \dots, 70]$
cp_2	50	$cp_2 \in [30, 31, 32, \dots, 70]$
csc_1	125	$csc_1 \in [110, 112, 114, \dots, 175]$
csc_2	225	$csc_2 \in [210, 212, 214, \dots, 270]$
p_0	50	valeur constante
p_1	150	$p_1 \in [100, 102, 104, \dots, 170]$
p_2	250	$p_2 \in [190, 192, 194, \dots, 270]$
h_1	$5\% \times (cp_1 + p_0)$	
h_2	$5\% \times (cp_2 + p_1)$	
cb_2	30	valeur constante

TAB. 3.1 – Valeurs des paramètres

Remarque sur les valeurs nominales des paramètres de coût :

Pour être fidèle à la réalité, nous avons à considérer les relations suivantes entre les différents paramètres :

- $h_1 \leq h_2$: le coût unitaire de stockage à l'usine₁ ne peut pas être supérieur à celui de l'usine₂,
- $h_2 < cb_2$: le coût unitaire de demande en attente est supérieur au coût unitaire de stockage à l'usine₂,
- $p_{i-1} + cp_i < csc_i, \forall i \in \{1, 2\}$: le coût unitaire total de production (le prix de vente d'une

pièce par l'usine précédente + le coût de production) doit être strictement inférieur au coût unitaire de la sous-traitance,

- $p_{i-1} + cp_i < p_i, \forall i \in \{1, 2\}$: le coût unitaire total de production doit être strictement inférieur au prix de vente unitaire de produit.

Paramètres de capacité et de demande :

Sans perte de généralité, nous considérons les valeurs suivantes :

- $\mu_1 = \mu_2 = 10$
- $\lambda < \{\mu_1, \mu_2\}$, pour permettre au système de satisfaire toute la demande sans forcément faire appel à la sous-traitance. Dans nos analyses, λ prendra la valeur $\lambda = 9$.

Finalement notons par :

- PU1G : le profit de l'usine₁ à l'optimisation globale,
- PU2G : le profit de l'usine₂ à l'optimisation globale,
- P1-2G : le profit du système à l'optimisation globale,
- B1G : le stock nominal B₁ à l'optimisation globale,
- B2G : le stock nominal B₂ à l'optimisation globale,
- DAPG : le niveau de demandes en attente partielles (DAP) à l'optimisation globale,
- PU1L : le profit de l'usine₁ à l'optimisation locale,
- PU2L : le profit de l'usine₂ à l'optimisation locale,
- P1-2L : le profit du système à l'optimisation locale,
- B1L : le stock nominal B₁ à l'optimisation locale,
- B2L : le stock nominal B₂ à l'optimisation locale,
- DAPL : le niveau de demandes en attente partielles (DAP) à l'optimisation locale.

3.5.2 Résultats qualitatifs

Après avoir étudié notre système à travers des exemples numériques, nous en avons déduit certaines propriétés. Premièrement, le profit total de notre système à l'optimisation globale est supérieur ou égal au profit total de l'optimisation locale. L'optimisation globale considère toutes les combinaisons des valeurs B₁, B₂ et DAP or l'optimisation locale consiste à optimiser B₁ étant donnés B₂ et DAP (chapitre 3.4). En conséquence, la somme de profits locaux est inférieure ou égale au profit total de l'optimisation globale : $PU1G + PU2G \geq PU1L + PU2L$ (*propr. 2.1*).

Pour chaque usine prise en compte individuellement et pour l'usine₂ en particulier, à l'optimisation locale, la marge de liberté de l'usine est plus grande qu'à l'optimisation globale où l'on cherche à optimiser le profit simultané des deux usines, ce qui restreint le nombre de choix possibles pour B₁ : $PU2G \leq PU2L$ (*propr. 2.2*). Comme conséquence directe des propriétés 2.1 et 2.2. nous avons : $PU1G \geq PU1L$ (*propr. 2.3*).

Finalement, concernant la relation des valeurs B_2 et DAP, nous avons constaté qu'il suffit de garder la différence (M) entre B_2 et DAP constante pour qu'on puisse obtenir la même chaîne de Markov avec les mêmes probabilités d'état. En particulier, les dimensions de notre chaîne de Markov dépendent des valeurs de B_1 et de la différence (M) entre B_2 et DAP. Pour l'exemple présenté dans la figure 3.6, les dimensions de notre chaîne de Markov sont $B_1=2$ et $M=6$ ($B_2=3$, $DAP=2$). Si nous gardons B_1 et M constants, nous obtenons exactement la même chaîne de Markov (*propr. 2.4*, annexe A.2), à ceci près que là où nous avons B_2 , nous allons avoir B'_2 (ex. $B'_2=1$) et là où nous avons DAP, nous avons $DAP'=M-B'_2$ ($DAP'=4$).

3.5.3 Variation des coûts

Avant de présenter nos résultats numériques nous remarquons que l'algorithme présenté dans la section 3.4 a été programmé en langage Visual Basic 6.0. Les expérimentations présentées ont été réalisées sur un ordinateur Pentium (R) 4, CPU 2,40 GHz et RAM 1 GB, sous Windows 2000 professionnel.

Dans cette section, nous donnons les résultats numériques de notre modèle présenté à la section 3.1 en faisant varier tous les paramètres de coût du système. Dans un premier temps nous présentons le comportement de notre système face à l'optimisation locale et globale et face à la variation du coût unitaire de production des deux usines. Par la suite, nous faisons varier les coûts unitaires des sous-traitants et nous terminons avec la variation du prix de vente des produits de l'usine₁ à l'usine₂ et de l'usine₂ aux clients.

Les premières valeurs que nous faisons varier sont les valeurs des coûts unitaires de production des usines. Naturellement, l'augmentation du coût unitaire de production pour une usine induit une diminution directe du profit de cette usine. Plus précisément, lorsque les politiques optimales restent constantes pour les deux types d'optimisation les probabilités d'état ne changent pas et le profit décroît car le coût unitaire de production augmente (cf. annexe A.3 : $40 \leq cp_1 \leq 45$, tableau 3.1, $30 \leq cp_2 \leq 38$, tableau 3.2). Nous constatons que cette augmentation fait décroître les valeurs de B_1 , B_2 et DAP et rend la politique de sous-traitance plus attractive. Ceci est lié à la diminution de la différence entre la valeur du coût unitaire de production et de sous-traitance et à l'augmentation du coût unitaire de stockage.

Concernant le coût unitaire de production à l'usine₁ (cp_1), son augmentation n'influence pas l'optimisation locale de l'usine₂. L'optimisation locale donne toujours le même plan d'optimisation pour l'usine₂ et le même profit (PU2L) (cf. annexe A.3, tableau 3.1). De plus, le profit PU2L étant constant et le profit PU1L étant décroissant, le profit total P1-2L diminue (cf. figure 3.8). Concernant la différence relative entre les deux types d'optimisation, nous constatons une

diminution quand le coût cp_1 augmente. Plus cp_1 augmente, plus les politiques optimales se rapprochent car la probabilité de sous-traiter et la quantité sous-traitée augmentent. Le maximum de différence entre les optimisations locale et globale, égal à 2.8%, est apparu avec un coût de production très petit (cf. annexe A.3, tableau 3.1). Cette différence n'est pas très grande mais elle peut être intéressante pour certaines applications industrielles où l'application d'une optimisation globale n'est pas coûteuse. Quand cp_1 augmente, le profit de l'usine₁ diminue et le rôle relatif que joue l'usine₁ dans la chaîne logistique s'atténue car elle fait souvent appel à la sous-traitance. L'usine₂ se comporte donc comme le seul acteur de la chaîne et plus cp_1 croît, plus le profit de l'usine₂ à l'optimisation globale converge vers celui de l'optimisation locale qui est sa borne supérieure (propriété 2.2). Finalement, pour la même raison, la somme P1-2G diminue (cf. figure 3.8).

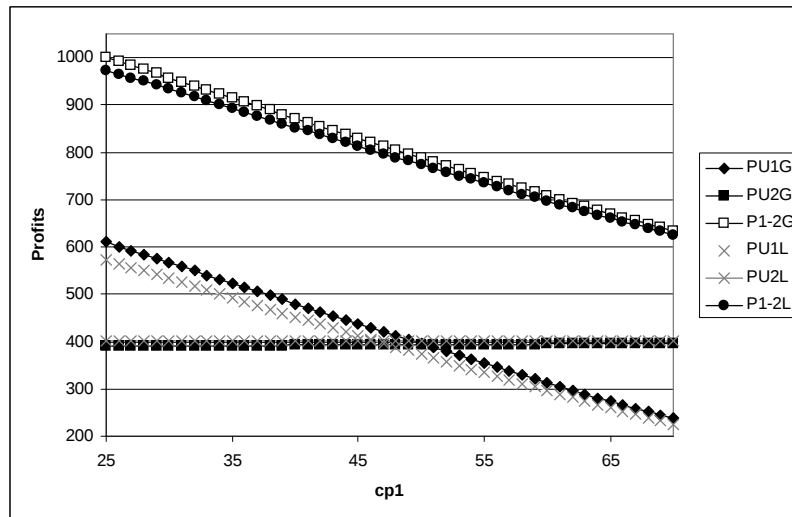


Fig. 3.8 – Les profits du système en fonction de cp_1

Quant au coût unitaire de production à l'usine₂ (cp_2), son augmentation diminue les profits PU1G et PU1L (cf. figure 3.9) car l'usine₁ vend moins de pièces à l'usine₂ qui fait plus souvent appel à la sous-traitance. Plus cp_2 augmente, plus la sous-traitance devient économiquement attractive, ce qui permet de satisfaire plus rapidement la demande et donc de diminuer le niveau de demandes en attente DAP. De plus, à l'optimisation locale, l'usine₂ est relativement plus sensible à l'augmentation du coût cp_2 , son stock nominal (B2L) diminue de manière plus forte que celui de l'optimisation globale (B2G), ce qui implique une augmentation de la différence relative (cf. annexe A.3, tableau 3.2). Enfin une autre raison pour laquelle DAP diminue est qu'en augmentant le coût de production à l'usine₂, on a aussi une augmentation du coût unitaire de stockage (h_2) qui converge vers le coût unitaire d'une demande en attente (cb). Eu égard à

la propriété 2.4, comme le niveau du stock nominal B_2 diminue et que h_2 converge vers cb , la différence entre B_2 et DAP diminue, ce qui implique aussi la diminution de DAP. D'après les résultats présentés au tableau 3.2 de l'annexe A.3, l'augmentation de cp_2 donne une différence relative entre les deux types d'optimisation jusque l'ordre de 7.42% quand cp_2 obtient une valeur proche à csc_2 . Cette différence nous montre bien l'intérêt de faire une optimisation globale au lieu d'une optimisation locale aussi dans le cas stochastique.

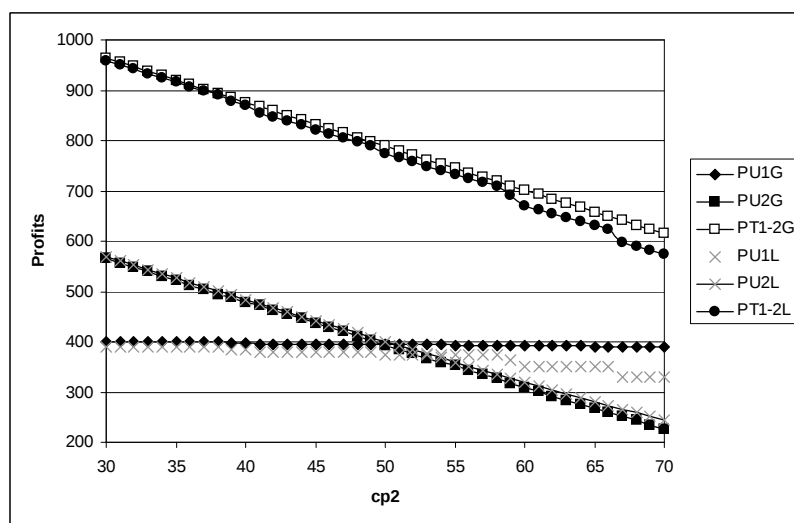


Fig. 3.9 – Les profits du système en fonction de cp_2

Ensuite, nous avons fait varier les coûts unitaires de la sous-traitance. Comme nous l'avons aussi constaté dans le cas de l'augmentation du coût unitaire de production, l'augmentation du coût unitaire de la sous-traitance pour une usine a pour conséquence la diminution du profit de cette usine. De plus, cette augmentation implique l'augmentation des valeurs de B_1 , B_2 , DAP car la production par anticipation est plus attractive. Ceci est lié à l'augmentation de la différence entre la valeur du coût unitaire de production et du coût unitaire de la sous-traitance et au fait que le coût unitaire de stockage reste constant. Si le coût unitaire de la sous-traitance est élevé, les usines sous-traitent rarement en satisfaisant leurs clients avec un stock nominal plus grand et le rôle de la sous-traitance s'atténue. Cela implique en moyenne une petite différence relative de l'ordre de 2-3% parce que le système devient moins flexible (cf. annexe A.3, tableau 3.3 et 3.4).

Concernant le coût unitaire de la sous-traitance de l'usine₁ (csc_1), comme nous l'avons présenté pour les autres paramètres de coût de l'usine₁, son augmentation n'influence pas l'optimisation locale de l'usine₂. De plus, le profit PU1L étant décroissant et le profit PU2L étant constant, la somme P1-2L diminue (cf. figure 3.10). Plus le coût csc_1 augmente, plus les po-

litiques optimales se rapprochent car la probabilité de sous-traiter et la quantité sous-traité diminuent.

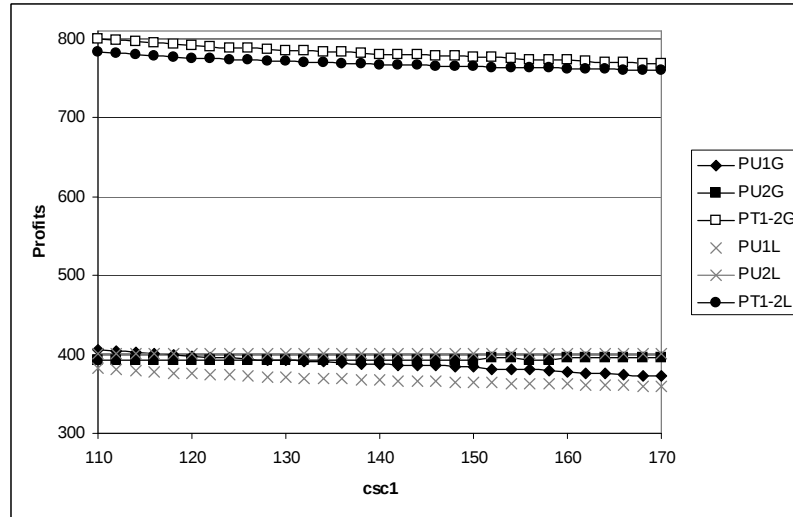


Fig. 3.10 – Les profits du système en fonction de csc_1

Enfin, quand le coût csc_1 augmente, l'usine₁ a une seule solution pour satisfaire sa demande supplémentaire à cause du coût unitaire de la sous-traitance très élevé : l'usine₁ construit un stock d'anticipation au lieu de sous-traiter et le rôle relatif qu'il joue dans la chaîne logistique s'atténue. En conséquence, le profit de l'usine₂ à l'optimisation globale converge vers celui de l'optimisation locale qui est sa borne maximale (propr. 2.2) et le profit total (P1-2G) diminue.

Quant au coût de sous-traitance (csc_2), son augmentation augmente le stock nominal B_2 et le niveau de demande en attente DAP aux deux types d'optimisation parce que la production et le stockage des produits sont économiquement plus attractifs que la sous-traitance. L'augmentation de la quantité des pièces produites a comme résultat l'augmentation du nombre de pièces demandées par l'usine₁, ce qui implique l'augmentation de son profit (PU1G et PU1L) (cf. figure 3.11) et son niveau de stock nominal ($B1G$, $B1L$) (cf. annexe A.3, tableau 3.4).

Tant que csc_2 augmente, l'usine₂ sous-traite moins et en conséquence, la différence relative entre les deux types d'optimisation diminue globalement. Une différence plus importante est apparue avec un coût csc_2 très petit ($csc_2=210$, $DR=3.5\%$) car l'usine₂ fait plus souvent appel à la sous-traitance₂ et achète moins de produits par l'usine₁.

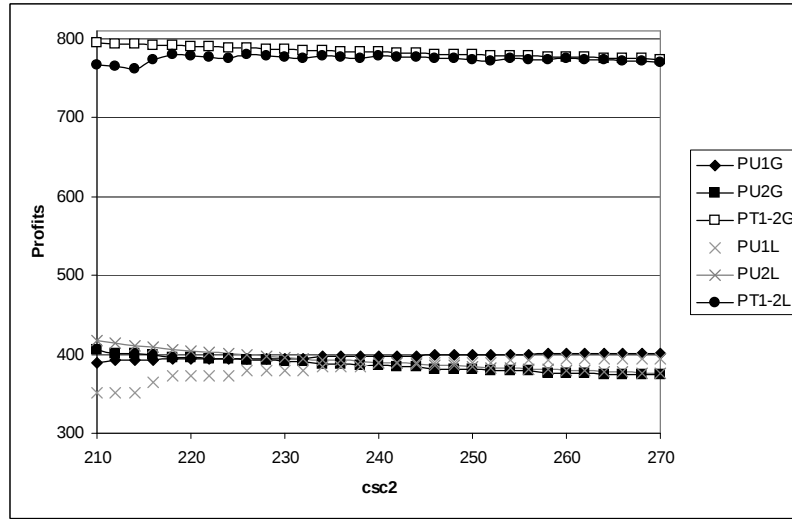


Fig. 3.11 – Les profits du système en fonction de csc_2

Les deux dernières valeurs que nous avons fait varier sont les prix de vente des deux usines (p_1 , p_2). La variation du prix p_1 , influence aussi le coût de stockage à l'usine₂ ($h_2 = 5\% \times (p_1 + cp_2)$) et donne la plus grande différence relative entre l'optimisation locale et globale. L'augmentation de p_1 implique l'augmentation du coût h_2 et l'augmentation du coût unitaire total de production de l'usine₂. Le coût unitaire total de production converge vers le coût unitaire de la sous-traitance, ce qui implique la diminution du niveau de stock nominal B2 et du niveau de demande en attente DAP car la politique de sous-traitance est plus attractive (cf. annexe A.3, tableau 3.5).

De plus, si p_1 croît, le profit de l'usine₁ croît. Le PU1G est négatif au début parce que la différence entre $p_0 + cp_1$ et p_1 est petite et ne peut pas couvrir le coût de stockage et de la sous-traitance. D'une part, quand le prix de vente de l'usine₁ est inférieur au coût unitaire de la sous-traitance, la probabilité de sous-traiter et la quantité sous-traitée sont très petites. D'autre part, quand la différence entre p_1 et csc_1 croît ($p_1 > csc_1$), l'usine₁ fait appel à la sous-traitance plus souvent et B₁ diminue. La plus grande différence relative entre l'optimisation locale et globale est obtenue au cas où le prix de vente de l'usine₁ est très grande ($p_1=170$, DR=9.5%). Dans ce cas l'usine₂ sous-traite plus et le rôle relatif de l'usine₁ dans la chaîne logistique s'atténue (cf. annexe A.3, tableau 3.5).

L'augmentation du prix p_2 n'influence pas les valeurs optimales de B₁, B₂ et DAP. L'augmentation de p_2 n'influence pas la différence entre l'optimisation globale et l'optimisation locale mais augmente le profit total du système à l'optimisation locale, ce qui implique la diminution

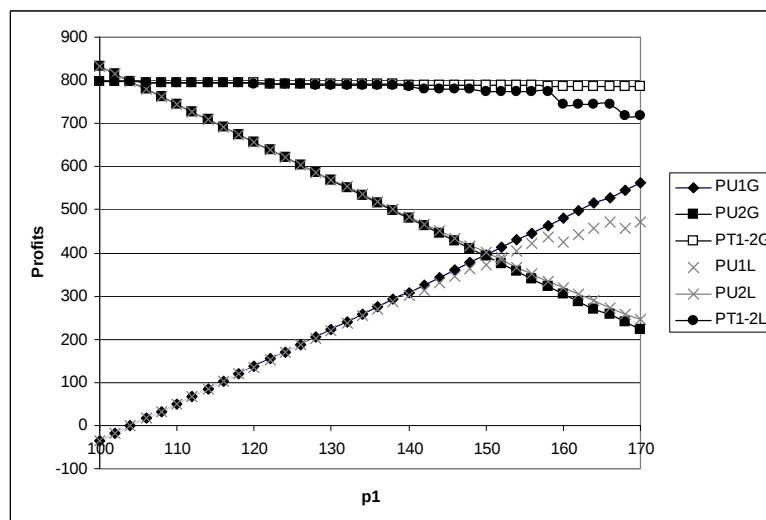


Fig. 3.12 – Les profits du système en fonction de p_1

de la différence relative entre les deux types d'optimisation (cf. annexe A.3, tableau 3.6).

Enfin, si le niveau nominal du stock B_1 et la différence entre B_2 et DAP restent constantes, le profit de notre système est une fonction concave de B_2 ou DAP (cf. annexe A.2, propriété 2.4). Cette propriété est liée au niveau de service de notre chaîne logistique. Si nous gardons B_1 constant et si nous gardons en même temps constante la somme de produits stockés et des produits en attente, le profit de notre système est une fonction concave, ce qui implique la présence d'un seul optimum (cf. figure 3.13).

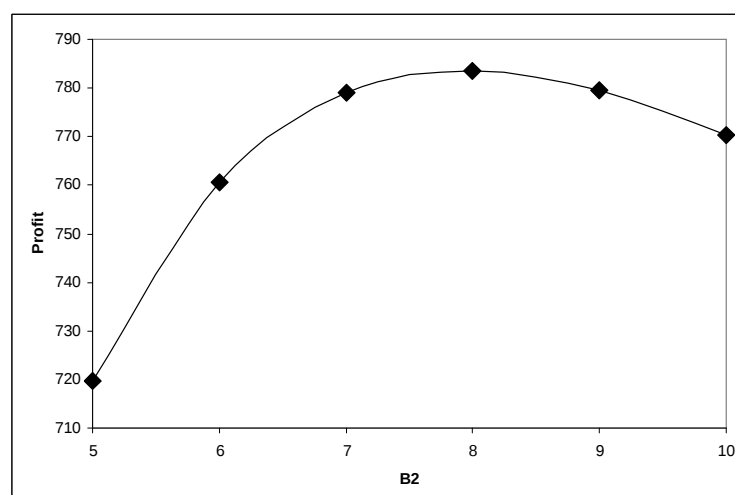


Fig. 3.13 – Le profit du système en fonction de B_2 (Différence entre B_2 et DAP constante)

3.5.4 Variation des paramètres μ_1 et μ_2

Nos résultats numériques trouvés en variant les coûts, présentent une image assez complète concernant le comportement de notre système. Généralement si le processus d'arrivée des clients est un processus de Poisson de taux λ et le temps de traitement suit dans les deux usines une distribution exponentielle de paramètre μ_1 et μ_2 , la différence relative entre les deux types d'optimisation peut être parfois importante. Nous remarquons que cette variation dépend des valeurs choisies des coûts.

Dans la section précédente, nous avons choisi les paramètres μ_1 et μ_2 égaux et supérieurs mais pas très loin de λ , car cette situation représente bien la réalité. Dans cette section, nous nous intéressons à l'étude du comportement de notre système face à la variation des paramètres μ_1 et μ_2 , en gardant tous les coûts constants. Si nous gardons le paramètre λ fixe et nous faisons varier les paramètres μ_1 et μ_2 , nous pouvons obtenir des résultats qui nous montrent une plus grande différence relative. Dans le tableau 3.2, nous présentons la différence relative entre l'optimisation locale et l'optimisation globale en fonction de μ_1 et μ_2 . Notons que les valeurs de μ_1 et μ_2 présentées dans le tableau suivant représentent aussi bien la réalité que les valeurs nominales (en gras).

μ_1	μ_2	Différence relative
10	13	14.99%
10	11	8.845%
10	10	9.55%
10	9	18.33%
10	7	16.29%
13	10	9.93%
11	10	9.73%
10	10	9.55%
9	10	9.28%
7	10	8.17%

TAB. 3.2 – Variation de μ_1 , μ_2

D'après ces exemples numériques nous constatons une augmentation de la différence relative (par rapport aux valeurs nominales) jusqu'à 50% quand le temps de traitement de l'usine₂ suit une distribution exponentielle de paramètre μ_2 égal au taux λ de processus d'arrivée des clients. Pour les exemples présentés dans l'annexe A.3, nous avons testé ces valeurs des paramètres μ_1 et μ_2 . Nous avons trouvé une augmentation de 30% jusqu'à 50% concernant la différence relative. Ceci implique que la différence relative entre l'optimisation locale et l'optimisation globale, qui arrive jusqu'à 18%, n'est pas seulement liée aux différents coûts mais aussi aux paramètres μ_1 et μ_2 . Ces paramètres impliquent souvent une augmentation ou une diminution de la différence relative entre l'optimisation locale et globale plus importante que la variation des coûts.

Conclusion-Perspectives

De nos jours, les gens s'intéressent de plus en plus à avoir un plan de production optimal pour chaque partie de la chaîne logistique mais aussi pour la chaîne toute entière. Dans la réalité, les entreprises cherchent un plan optimal pour chacune de leurs entités, sans tenir compte des attributs de tout le système, tout en faisant une optimisation locale. Cette approche donne des plans réalisables mais pas optimaux pour la chaîne logistique entière.

Dans la première partie de la thèse, nous avons étudié la perte engendrée par une optimisation locale par rapport à une optimisation globale. Le système étudié fait référence à deux usines principales, à deux stocks qui sont associés à ces deux usines et à deux usines externes appelées sous-traitants. Ces deux usines externes fournissent immédiatement des produits aux stocks des deux usines principales durant la période où les deux usines en commandent.

Notre travail a comme objectif de développer deux modèles, le premier fondé sur l'optimisation globale et l'autre sur l'optimisation locale. A l'optimisation locale nous avons une suite d'optimisations locales de l'aval vers l'amont et à l'optimisation globale, nous considérons notre système comme un système global et nous l'optimisons globalement. La fonction objectif de ces modèles est la maximisation du profit total, sous les contraintes et les hypothèses de notre système. Nous avons étudié, pour les deux types d'optimisation (globale et locale), deux types de demande : une demande déterministe et saisonnière et une demande stochastique stationnaire non saisonnière

Dans les chapitres 2 et 3, nous avons présenté le comportement de notre système face aux deux méthodes d'optimisation et aux deux types de demande. Nous avons réussi à utiliser les modèles développés afin de mettre en évidence un certain nombre de comportements qualitatifs.

En outre, nous avons cherché en particulier à déterminer dans quelles conditions les deux modèles donnent des résultats significativement différents et nous avons pu trouver les cas où les deux modèles développés donnent exactement le même plan de production. Les résultats obtenus sont intéressants et nous montrent bien quel est le profit pour les entreprises avec une

optimisation globale et quelle perte peut engendrer pour les entreprises une optimisation locale en fonction de ses conditions de réalisation.

Généralement, nous avons montré que l'optimisation globale donne toujours une meilleure solution même si elle pénalise parfois les sous-systèmes. Nos résultats montrent que dans le cas déterministe, une différence relative importante existe entre les deux types d'optimisation. Finalement, dans certains cas, la demande stochastique assure un gain important en appliquant l'optimisation globale.

Les extensions possibles de nos travaux sont nombreuses. En effet, dans l'objectif d'obtenir des résultats analytiques sans avoir recours à des hypothèses limitatives, nous avons étudié des modèles simples de chaînes logistique. Cette étude nous a servi pour comprendre le comportement du système en fonction des différents paramètres qui entrent en jeu. Ainsi, il est intéressant d'étendre cette étude à des modèle plus complexes et variés. Nous proposons ci-dessous quelque idées :

- pour la suite de ce travail, nous pouvons proposer l'étude d'une chaîne logistique multi-produits. Le cas multi-produits sera une étude assez intéressante car il s'agit d'un cas réel qui apparaît dans la vie industrielle. De plus, une étude de cas avec plusieurs fournisseurs et plusieurs détaillants serait intéressant car dans la réalité, l'organisation de la chaîne logistique n'est pas souvent linéaire mais a une structure en réseau,
- nous proposons aussi l'étude d'autres politiques de pilotage de flux : Dans les articles [Liberopoulos and Dallery, 2000] et [Liberopoulos and Dallery, 2003], les auteurs présentent plusieurs politiques de pilotage de flux et plusieurs études et comparaisons. Les auteurs montrent que la politique Kanban généralisée donne des performances très proches de la politique optimale. Il serait intéressant d'étudier la différence entre le deux méthodes d'optimisation sur les performances de cette politique dans le cas multi-étages. De plus, l'étude d'une demande à la fois stochastique et saisonnier peut être une autre perspectives,
- finalement, comme nous le remarquons, l'optimisation globale donne toujours une meilleure solution mais elle pénalise parfois les sous-systèmes. L'usine₂ n'a pas directement intérêt à pratiquer une optimisation globale. Pour que l'optimisation globale devienne plus attractive pour les parties d'une chaîne logistique, il faudrait qu'une partie de gain engendré par l'usine₁ soit redonné à l'usine₂. Une méthode de répartition du gain d'une optimisation globale dans le même esprit que celle présentée par [Jen Ming and Tsung Hui, 2005] pourrait s'inscrire comme une autre extension de nos travaux de recherche. De cette manière, les deux usines bénéficieraient de l'optimisation globale.

ANNEXES

Annexe A.1

Notons par :

- COG : le Coût d'Optimisation Globale du système entier,
- COL : le Coût d'Optimisation Locale du système entier,
- COGU1 : le Coût d'Optimisation Globale de l'usine₁,
- COGU2 : le Coût d'Optimisation Globale de l'usine₂,
- COLU1 : le Coût d'Optimisation Locale de l'usine₁,
- COLU2 : le Coût d'Optimisation Locale de l'usine₂,
- D_t : la demande à satisfaire par l'usine₂ durant la période t ,
- D'_t : la demande à satisfaire par l'usine₁ durant la période t ,
- $P_{i,t}$: la production à l'usine _{i} , durant la période t ,
- $I_{i,t}$: le niveau de stock à l'usine _{i} , durant la période t ,
- $SC_{i,t}$: les produits sous-traités par le sous-traitant _{i} , durant la période t ,
- cp_i : le coût de production d'une pièce à l'usine _{i} ,
- h_i : le coût de stockage d'une pièce, durant une période à l'usine _{i} ,
- csc_i : le coût d'une pièce sous-traité par le sous-traitant _{i} ,
- C_i : la capacité de production de l'usine _{i} ,
- Δ_i : la différence entre le cp_i et csc_i .

1.1 Modélisation

Modèle d'optimisation globale

Fonction objectif :

$$\begin{aligned} \min Z = & cp_1 \sum_{t=1}^T P_{1,t} + h_1 \sum_{t=1}^T I_{1,t} + csc_1 \sum_{t=1}^T SC_{1,t} \\ & + cp_2 \sum_{t=1}^T P_{2,t} + h_2 \sum_{t=1}^T I_{2,t} + csc_2 \sum_{t=1}^T SC_{2,t} \end{aligned} \quad (1.19)$$

Sous les contraintes :

Equations d'équilibre :

$$I_{1,t} = I_{1,t-1} + P_{1,t} + SC_{1,t} - P_{2,t} - SC_{2,t} \quad \forall t \in [1, T-1] \quad (1.20)$$

$$I_{2,t} = I_{2,t-1} + P_{2,t} + SC_{2,t} - D_t \quad \forall t \in [2, T] \quad (1.21)$$

Capacité de production :

$$P_{1,t} \leq C_1 \quad \forall t \quad (1.22)$$

$$P_{2,t} \leq C_2 \quad \forall t \quad (1.23)$$

Modèle d'optimisation locale

Modèle d'optimisation local pour l'usine 1 :

Fonction objectif :

$$\min Z_1 = cp_1 \sum_{t=1}^T P_{1,t} + h_1 \sum_{t=1}^T I_{1,t} + csc_1 \sum_{t=1}^T SC_{1,t} \quad (1.24)$$

Sous les contraintes :

Equations d'équilibre :

$$I_{1,t} = I_{1,t-1} + P_{1,t} + SC_{1,t} - P_{2,t} - SC_{2,t} \quad \forall t \in [1, T-1] \quad (1.25)$$

Capacité de production :

$$P_{1,t} \leq C_1 \quad \forall t \quad (1.26)$$

Modèle d'optimisation local pour l'usine 2 :

Fonction objectif

$$\min Z_2 = cp_2 \sum_{t=1}^T P_{2,t} + h_2 \sum_{t=1}^T I_{2,t} + csc_2 \sum_{t=1}^T SC_{2,t} \quad (1.27)$$

Sous les contraintes :

Equations d'équilibre :

$$I_{2,t} = I_{2,t-1} + P_{2,t} + SC_{2,t} - D_t \quad \forall t \in [2, T] \quad (1.28)$$

Capacité de production :

$$P_{2,t} \leq C_2 \quad \forall t \quad (1.29)$$

1.2 Propriétés 1.1/1.2/1.3 :

- Propriété 1.1 : le Coût d'Optimisation Globale est inférieur ou égal au Coût d'Optimisation Locale (i.e. $COG \leq COL$).

La solution du modèle d'optimisation locale est une solution réalisable pour le modèle d'optimisation globale mais pas forcément l'optimale,

- Propriété 1.2 : le Coût d'Optimisation Locale de l'usine₂ est inférieur ou égal au Coût d'Optimisation Globale (i.e. $COLU_2 \leq COGU_2$).

La solution du modèle d'optimisation globale est une solution réalisable pour le modèle d'optimisation locale mais pas forcément l'optimale,

- Propriété 1.3 : le Coût d'Optimisation Locale de l'usine₁ est supérieur ou égal au Coût d'Optimisation Globale (i.e. $COLU_1 \geq COGU_1$).

Conséquence directe des propriétés 1.1 et 1.2.

1.3 Propriété 1.4 :

Quand la différence entre le coût de production cp_i et le coût de la sous-traitance csc_i reste la même, les plans qui en résultent sont les mêmes et la différence des solutions optimales (en fonction du coût), entre les deux modèles, reste constante. Prenons l'exemple suivant : soient

cp_2 , csc_2 ayant une différence Δ_2 et un autre couple cp'_2 et csc'_2 avec la même différence Δ_2 . Il suffit de démontrer que les deux fonctions objectifs sont identiques et donnent le même plan. En démontrant cela pour la fonction objectif de l'usine₂, (modèle décentralisé), nous pouvons le démontrer de la même façon pour l'usine₁ (modèle décentralisé) et pour la fonction objectif du modèle centralisé.

Pour cp'_2 , csc'_2 nous avons la fonction objectif suivante :

$$\min Z = cp'_2 \sum_{t=1}^T P_{2,t} + h_2 \sum_{t=1}^T I_{2,t} + csc'_2 \sum_{t=1}^T SC_{2,t} \quad (1.30)$$

Sous les contraintes :

Equation d'équilibre :

$$I_{2,t} = I_{2,t-1} + P_{2,t} + SC_{2,t} - D_t \quad \forall t \in [2, T] \quad (1.31)$$

Capacité de production :

$$P_{2,t} \leq C_2 \quad \forall t \quad (1.32)$$

Il est aussi vrai que :

$$\sum_{t=1}^T P_{2,t} + \sum_{t=1}^T SC_{2,t} = \sum_{t=1}^T D_t \quad (1.33)$$

$$cp'_2 - csc'_2 = \Delta_2 \quad (1.34)$$

Donc, la fonction objectif devient :

$$\min Z = cp'_2 \sum_{t=1}^T P_{2,t} + h_2 \sum_{t=1}^T I_{2,t} + csc'_2 \sum_{t=1}^T SC_{2,t} \Rightarrow \quad (1.35)$$

$$\min Z = cp'_2 \left(\sum_{t=1}^T D_t - \sum_{t=1}^T SC_{2,t} \right) + h_2 \sum_{t=1}^T I_{2,t} + csc'_2 \sum_{t=1}^T SC_{2,t} \Rightarrow \quad (1.36)$$

$$\min Z = cp'_2 \sum_{t=1}^T D_t + h_2 \sum_{t=1}^T I_{2,t} + (csc'_2 - cp'_2) \sum_{t=1}^T SC_{2,t} \Rightarrow (csc'_2 - cp'_2 = \Delta_2) \quad (1.37)$$

$$\min Z = cp'_2 \sum_{t=1}^T D_t + h_2 \sum_{t=1}^T I_{2,t} + \Delta_2 \sum_{t=1}^T SC_{2,t} \quad (1.38)$$

La fonction objectif avec les coûts initiaux devient de la même façon :

$$\min Z = cp_2 \sum_{t=1}^T D_t + h_2 \sum_{t=1}^T I_{2,t} + \Delta_2 \sum_{t=1}^T SC_{2,t} \quad (1.39)$$

Les fonctions objectif 1.38 et 1.39 en présence des mêmes contraintes donnent les mêmes plans de production parce que les variables de décision ont les mêmes multiplicateurs. Dans la fonction objectif, il y a une seule différence dans les membres fixes. Ce changement n'influence pas l'optimisation parce qu'il ne contient pas de variables de décision. En conséquence, les différences relatives entre les deux méthodes d'optimisation vont rester les mêmes.

1.4 Propriété 1.7 :

Notons par $COLE2$ le coût de l'optimisation locale de l'entreprise₂ (E2), et $COGE2$ de l'optimisation globale. Pour que la politique d'optimisation locale soit recommandable, pour l'E2 (entreprise₂) qui possède 100% des actions de l'usine₂ et le $\alpha\%$ de l'usine₁ il faut que :

$$COLE2 < COGE2 \Rightarrow$$

$$\bullet COLE2 = \alpha \times COLU1 + COLU2$$

$$\bullet COGE2 = \alpha \times COGU1 + COGU2$$

$$\Rightarrow \alpha \times COLU1 + COLU2 < \alpha \times COGU1 + COGU2 \Rightarrow \alpha \times (COLU1 - COGU1) < (COGU2 - COLU2) \Rightarrow \alpha < (COL1 - COG1)/(COG2 - COL2).$$

1.5 Propriétés 1.8-1.9-1.10-1.11 :

Pour généraliser l'idée présentée dans la section précédente, nous supposons deux entreprises, l'entreprise₁ (E1) et l'entreprise₂ (E2). E1 détient le $(1 - \alpha)\%$ de l'usine₁ et $\beta\%$ de l'usine₂ avec $\alpha, \beta \in [0, 1]$. Au contraire E2 détient le $\alpha\%$ de l'usine₁ et $(1 - \beta)\%$ de l'usine₂. Notons par $COLE1$, $COLE2$ les coûts de l'optimisation locale de l'E1, E2 et respectivement $COGE1$, $COGE2$ de l'optimisation globale. Les coûts de production pour les deux usines sont donnés par les fonctions suivantes :

$$\bullet COLE1 = \beta \times COLU2 + (1 - \alpha) \times COLU1,$$

$$\bullet COGE1 = \beta \times COGU2 + (1 - \alpha) \times COGU1,$$

$$\bullet COLE2 = (1 - \beta) \times COLU2 + \alpha \times COLU1,$$

$$\bullet COGE2 = (1 - \beta) \times COGU2 + \alpha \times COGU1.$$

En poursuivant la procédure présentée à la démonstration de la propriété 1.7 nous montrons les propriétés 1.8-1.9-1.10-1.11.

Propriété 1.8 : (L'OL est préférable à l'OG pour l'E2) :

$$COLE2 < COGE2 \Rightarrow$$

$$\begin{aligned}
(1 - \beta) \times COLU2 + \alpha \times COLU1 &< (1 - \beta) \times COGU2 + \alpha \times COGU1 \Rightarrow \\
\alpha \times [COLU1 - COGU1] &< (1 - \beta) \times [COGU2 - COLU2] \Rightarrow \\
\frac{\alpha}{(1-\beta)} &< \frac{(COGU2-COLU2)}{(COLU1-COGU1)} \quad (\beta \neq 1)
\end{aligned}$$

Propriété 1.9 : (L'OG est préférable à l'OL pour l'E2)

$$\begin{aligned}
COLE2 > COGE2 &\Rightarrow \dots \Rightarrow \\
\frac{\alpha}{(1-\beta)} &> \frac{(COGU2-COLU2)}{(COLU1-COGU1)} \quad (\beta \neq 1)
\end{aligned}$$

Propriété 1.10 : (L'OL est préférable à l'OG pour l'E1)

$$\begin{aligned}
COLE1 < COGE1 &\Rightarrow \\
\beta \times COLU2 + (1 - \alpha) \times COLU1 &< \beta \times COGU2 + (1 - \alpha) \times COGU1 \Rightarrow \\
(1 - \alpha) \times [COLU1 - COGU1] &< \beta \times [COGU2 - COLU2] \Rightarrow \\
\frac{(1-\alpha)}{\beta} &< \frac{(COGU2-COLU2)}{(COLU1-COGU1)} \quad (\beta \neq 0)
\end{aligned}$$

Propriété 1.11 : (L'OG est préférable à l'OL pour l'E1)

$$\begin{aligned}
COLE1 > COGE1 &\Rightarrow \dots \Rightarrow \\
\frac{(1-\alpha)}{\beta} &> \frac{(COGU2-COLU2)}{(COLU1-COGU1)} \quad (\beta \neq 0)
\end{aligned}$$

Annexe A.2

2.1 Propriété 2.5

En ayant B_1 et la différence (M) entre B_2 et DAP constants, la chaîne de Markov obtenue ne change pas en ce qui concernent les valeurs des probabilités d'état. La seule différence est au niveau de calcul du coût de stockage et du coût de demande en attente. Dès lors, pour montrer que le profit de notre système est une fonction concave de B_2 ou de DAP quand B_1 et M sont constants, il suffit de montrer que la fonction suivante est une fonction concave car les autres membres de la fonction objective (maximisation du profit) restent constantes :

$$\phi = - \sum_{I_1=1}^{B_1} \sum_{I_2=1}^{B_2} I_2 \times P(I_1, I_2) \times h_2 - \sum_{I_1=1}^{B_1} \sum_{BL=1}^{DAP} BL \times P(I_1, BL) \times cb$$

Pour montrer que ϕ est une fonction concave il suffit de montrer que $\phi' = -\phi$ est une fonction convexe

$$\phi' = \sum_{I_1=1}^{B_1} \sum_{I_2=1}^{B_2} I_2 \times P(I_1, I_2) \times h_2 + \sum_{I_1=1}^{B_1} \sum_{BL=1}^{DAP} BL \times P(I_1, BL) \times cb$$

pour montrer que la fonction est une fonction convexe il faut montrer que :

$$2 \times \phi'(DAP) \leq \phi'(DAP - 1) + \phi'(DAP + 1)$$

Démonstration : En ayant B_1 et la différence M entre B_2 et DAP constants les probabilités suivantes sont égales :

$P(I_1, M-DAP)$	$=$	$P(I_1, M-DAP-1)$	$=$	$P(I_1, M-DAP+1)$
$\vdots$	$=$	$\vdots$	$=$	$\vdots$
$\vdots$	$=$	$\vdots$	$=$	$\vdots$
$P(I_1, 1)$	$=$	$P(I_1, 0)$	$=$	$P(I_1, 2)$
$P(I_1, 0)$	$=$	$P(I_1, -1)$	$=$	$P(I_1, 1)$
$P(I_1, -1)$	$=$	$P(I_1, -2)$	$=$	$P(I_1, 0)$
$\vdots$	$=$	$\vdots$	$=$	$\vdots$
$\vdots$	$=$	$\vdots$	$=$	$\vdots$
$P(I_1, DAP)$	$=$	$P(I_1, DAP+1)$	$=$	$P(I_1, DAP-1)$

$$\begin{aligned}
\phi'(DAP) &= cb \times (DAP) \times P(I_1, DAP) + \\
&\quad \dots + cb \times 1 \times P(I_1, -1) + \\
&\quad \dots + h_2 \times 1 \times P(I_1, 1) + \\
&\quad \dots + h_2 \times (M - DAP) \times P(I_1, M - DAP)
\end{aligned}$$

$$\begin{aligned}
\phi'(DAP - 1) &= cb \times (DAP - 1) \times P(I_1, DAP) + \\
&\quad \dots + cb \times 1 \times P(I_1, -2) + \\
&\quad \dots + h_2 \times 2 \times P(I_1, 1) + \\
&\quad \dots + h_2 \times (M - DAP + 1) \times P(I_1, M - DAP)
\end{aligned}$$

$$\begin{aligned}
\phi'(DAP + 1) &= cb \times (DAP) \times P(I_1, DAP) + \\
&\quad \dots + cb \times 1 \times P(I_1, 0) + \\
&\quad \dots + h_2 \times 0 \times P(I_1, 1) + \\
&\quad \dots + h_2 \times (M - DAP - 1) \times P(I_1, M - DAP)
\end{aligned}$$

Si on fait la somme de fonctions précédentes nous obtenons le résultat suivant :

$$\phi'(DAP - 1) + \phi'(DAP + 1) = 2 \times \phi'(DAP) + (h_2 + cb) \times P(I_1, DAP) > 2 \times \phi'(DAP)$$

Ce qui implique que la fonction ϕ' est une fonction convexe et ϕ une fonction concave.

Annexe A.3

cp_1	Dif.Rel.	PU1G	PU2G	P1-2G	BU1G	BU2G	DAPG	PU1L	PU2L	P1-2L	BU1L	BU2L	DAPL
25	2,770	609,99	390,792	1000,782	10	7	5	573,58	400,22	973,8	9	4	2
26	2,746	601,32	390,792	992,12	10	7	5	565,38	400,22	965,6	9	4	2
27	2,715	592,66	390,792	983,452	10	7	5	557,23	400,22	957,45	8	4	2
28	2,678	583,991	390,792	974,783	10	7	5	549,13	400,22	949,35	8	4	2
29	2,641	575,32	390,792	966,112	10	7	5	541,03	400,22	941,25	8	4	2
30	2,602	566,653	390,792	957,445	10	7	5	532,941	400,22	933,161	8	4	2
31	2,563	557,985	390,792	948,777	10	7	5	524,844	400,22	925,064	8	4	2
32	2,522	549,31	390,79	940,1	10	7	5	516,746	400,22	916,966	8	4	2
33	2,492	540,730	390,79	931,520	9	7	5	508,649	400,22	908,869	8	4	2
34	2,463	532,169	390,79	922,959	9	7	5	500,552	400,22	900,772	8	4	2
35	2,433	523,6	390,79	914,39	9	7	5	492,45	400,22	892,67	8	4	2
36	2,402	515,04	390,79	905,83	9	7	5	484,357	400,22	884,577	8	4	2
37	2,371	506,48	390,792	897,272	9	7	5	476,27	400,22	876,49	7	4	2
38	2,324	497,927	390,792	888,719	9	7	5	468,311	400,22	868,531	7	4	2
39	2,276	489,36	390,792	880,152	9	7	5	460,34	400,22	860,56	7	4	2
40	2,246	478,36	393,38	871,74	8	7	4	452,37	400,22	852,59	7	4	2
41	2,215	469,95	393,38	863,33	8	7	4	444,4	400,22	844,62	7	4	2
42	2,183	461,55	393,38	854,93	8	7	4	436,44	400,22	836,66	7	4	2
43	2,151	453,14	393,38	846,52	8	7	4	428,472	400,22	828,692	7	4	2
44	2,119	444,736	393,38	838,116	8	7	4	420,5	400,22	820,72	7	4	2
45	2,086	436,329	393,38	829,709	8	7	4	412,53	400,22	812,75	7	4	2
46	2,055	427,955	393,38	821,335	7	7	4	404,569	400,22	804,789	7	4	2
47	2,026	419,69	393,38	813,07	7	7	4	396,7	400,22	796,92	6	4	2
48	1,989	411,443	393,38	804,823	7	7	4	388,9	400,22	789,12	6	4	2
49	1,951	403,186	393,38	796,566	7	7	4	381,1	400,22	781,32	6	4	2
50	1,912	394,93	393,38	788,31	7	7	4	373,298	400,22	773,518	6	4	2
51	1,871	386,67	393,38	780,05	7	7	4	365,496	400,22	765,716	6	4	2
52	1,831	378,41	393,38	771,79	7	7	4	357,69	400,22	757,91	6	4	2
53	1,808	370,300	393,38	763,680	6	7	4	349,892	400,22	750,112	6	4	2
54	1,791	362,230	393,38	755,610	6	7	4	342,090	400,22	742,310	6	4	2
55	1,761	354,161	393,38	747,541	6	7	4	334,379	400,22	734,599	5	4	2
56	1,712	346,091	393,38	739,471	6	7	4	326,799	400,22	727,019	5	4	2
57	1,663	338,022	393,382	731,404	6	7	4	319,218	400,22	719,438	5	4	2
58	1,612	329,952	393,382	723,335	6	7	4	311,637	400,22	711,857	5	4	2
59	1,594	322,122	393,382	715,505	5	7	4	304,057	400,22	704,277	5	4	2
60	1,578	312,144	395,549	707,693	5	6	4	296,476	400,22	696,696	5	4	2
61	1,564	304,345	395,549	699,894	5	6	4	288,896	400,22	689,116	5	4	2
62	1,511	296,546	395,549	692,095	5	6	4	281,567	400,22	681,787	4	4	2
63	1,459	288,810	395,549	684,360	4	6	4	274,297	400,22	674,517	4	4	2
64	1,444	281,339	395,549	676,888	4	6	4	267,027	400,22	667,247	4	4	2
65	1,430	273,868	395,549	669,417	4	6	4	259,758	400,22	659,978	4	4	2
66	1,415	266,397	395,549	661,946	4	6	4	252,488	400,22	652,708	4	4	2
67	1,344	258,926	395,549	654,475	4	6	4	245,575	400,22	645,795	3	4	2
68	1,328	251,927	395,549	647,476	3	6	4	238,770	400,22	638,990	3	4	2
69	1,313	244,937	395,549	640,486	3	6	4	231,964	400,22	632,184	3	4	2
70	1,293	237,947	395,549	633,496	3	6	4	225,188	400,22	625,408	2	4	2

TAB. 3.1 – Variation de cp_1

cp_2	Dif.Rel.	PU1G	PU2G	PT1-2G	BU1G	BU2G	DAPG	PU1L	PU2L	PT1-2L	BU1L	BU2L	DAPL
30	0,618	398,879	565,890	964,770	7	8	5	389,085	569,750	958,836	7	6	3
31	0,594	398,879	557,001	955,880	7	8	5	389,085	561,149	950,234	7	6	3
32	0,5682	398,879	548,111	946,991	7	8	5	389,085	552,547	941,633	7	6	3
33	0,543	398,879	539,221	938,101	7	8	5	389,085	543,946	933,031	7	6	3
34	0,517	398,879	530,332	929,211	7	8	5	389,085	535,344	924,430	7	6	3
35	0,490	398,879	521,442	920,322	7	8	5	389,085	526,742	915,828	7	6	3
36	0,463	398,879	512,552	911,432	7	8	5	389,085	518,141	907,227	7	6	3
37	0,435	398,879	503,662	902,542	7	8	5	389,085	509,539	898,625	7	6	3
38	0,407	398,879	494,773	893,652	7	8	5	389,085	500,938	890,024	7	6	3
39	0,851	397,083	487,743	884,826	7	7	5	385,015	492,343	877,359	7	5	3
40	0,820	397,083	478,929	876,012	7	7	5	385,015	483,868	868,884	7	5	3
41	1,381	394,930	472,285	867,215	7	7	4	379,976	475,423	855,400	6	5	2
42	1,346	394,930	463,518	858,448	7	7	4	379,976	467,063	847,040	6	5	2
43	1,311	394,930	454,751	849,681	7	7	4	379,976	458,704	838,681	6	5	2
44	1,275	394,930	445,984	840,914	7	7	4	379,976	450,344	830,321	6	5	2
45	1,239	394,930	437,217	832,147	7	7	4	379,976	441,985	821,962	6	5	2
46	1,201	394,930	428,450	823,380	7	7	4	379,976	433,625	813,602	6	5	2
47	1,163	394,930	419,683	814,613	7	7	4	379,976	425,266	805,242	6	5	2
48	1,124	394,930	410,916	805,846	7	7	4	379,976	416,906	796,883	6	5	2
49	1,085	394,930	402,149	797,079	7	7	4	379,976	408,547	788,523	6	5	2
50	1,911	394,930	393,382	788,312	7	7	4	373,298	400,227	773,526	6	4	2
51	1,853	394,930	384,615	779,545	7	7	4	373,298	392,057	765,356	6	4	2
52	1,795	394,930	375,848	770,778	7	7	4	373,298	383,887	757,186	6	4	2
53	1,734	394,930	367,081	762,011	7	7	4	373,298	375,717	749,016	6	4	2
54	1,673	394,930	358,314	753,244	7	7	4	373,298	367,547	740,846	6	4	2
55	1,613	392,314	352,186	744,501	7	6	4	373,298	359,377	732,676	6	4	2
56	1,562	392,314	343,513	735,828	7	6	4	373,298	351,207	724,506	6	4	2
57	1,510	392,314	334,841	727,155	7	6	4	373,298	343,037	716,336	6	4	2
58	1,456	392,314	326,168	718,483	7	6	4	373,298	334,867	708,166	6	4	2
59	2,738	392,314	317,495	709,810	7	6	4	363,997	326,890	690,887	6	4	1
60	4,673	392,314	308,823	701,138	7	6	4	350,653	319,180	669,833	5	3	1
61	4,569	392,314	300,150	692,465	7	6	4	350,653	311,553	662,206	5	3	1
62	4,462	392,314	291,478	683,792	7	6	4	350,653	303,927	654,580	5	3	1
63	4,353	392,314	282,805	675,120	7	6	4	350,653	296,300	646,953	5	3	1
64	4,242	392,314	274,132	666,447	7	6	4	350,653	288,673	639,327	5	3	1
65	4,128	389,085	268,696	657,782	7	6	3	350,653	281,047	631,700	5	3	1
66	4,023	389,085	260,095	649,181	7	6	3	350,653	273,420	624,074	5	3	1
67	7,420	389,085	251,493	640,579	7	6	3	330,098	266,228	596,327	5	2	1
68	7,261	389,085	242,892	631,978	7	6	3	330,098	259,094	589,192	5	2	1
69	7,098	389,085	234,290	623,376	7	6	3	330,098	251,959	582,058	5	2	1
70	6,931	389,085	225,689	614,775	7	6	3	330,098	244,825	574,924	5	2	1

TAB. 3.2 – Variation de cp_1

csc_1	Dif.Rel.	PU1G	PU2G	PT1-2G	BU1G	BU2G	DAPG	PU1L	PU2L	PT1-2L	BU1L	BU2L	DAPL
110	2,168	406,636	393,382	800,018	4	7	4	382,808	400,227	783,035	4	4	2
112	2,151	404,596	393,382	797,978	5	7	4	380,941	400,227	781,168	4	4	2
114	2,102	402,746	393,382	796,128	5	7	4	379,510	400,227	779,738	5	4	2
116	2,044	400,952	393,382	794,335	6	7	4	378,196	400,227	778,423	5	4	2
118	2,032	399,520	393,382	792,903	6	7	4	376,882	400,227	777,109	5	4	2
120	2,008	398,088	393,382	791,471	6	7	4	375,657	400,227	775,885	6	4	2
122	1,948	396,656	393,382	790,039	6	7	4	374,714	400,227	774,941	6	4	2
124	1,922	395,493	393,382	788,875	7	7	4	373,770	400,227	773,997	6	4	2
126	1,900	394,366	393,382	787,749	7	7	4	372,826	400,227	773,054	6	4	2
128	1,879	393,240	393,382	786,623	7	7	4	371,883	400,227	772,110	6	4	2
130	1,853	392,114	393,382	785,496	7	7	4	370,978	400,227	771,205	7	4	2
132	1,820	391,159	393,382	784,542	8	7	4	370,290	400,227	770,517	7	4	2
134	1,794	390,264	393,382	783,646	8	7	4	369,601	400,227	769,829	7	4	2
136	1,769	389,369	393,382	782,751	8	7	4	368,913	400,227	769,141	7	4	2
138	1,744	388,473	393,382	781,856	8	7	4	368,225	400,227	768,452	7	4	2
140	1,718	387,578	393,382	780,960	8	7	4	367,537	400,227	767,764	7	4	2
142	1,705	386,775	393,382	780,157	9	7	4	366,849	400,227	767,076	7	4	2
144	1,702	386,057	393,382	779,439	9	7	4	366,168	400,227	766,395	8	4	2
146	1,675	385,339	393,382	778,722	9	7	4	365,660	400,227	765,887	8	4	2
148	1,649	384,622	393,382	778,004	9	7	4	365,152	400,227	765,379	8	4	2
150	1,623	383,904	393,382	777,286	9	7	4	364,644	400,227	764,871	8	4	2
152	1,599	381,037	395,549	776,586	9	6	4	364,136	400,227	764,363	8	4	2
154	1,579	380,370	395,549	775,919	9	6	4	363,628	400,227	763,855	8	4	2
156	1,430	380,882	393,382	774,264	10	6	4	363,120	400,227	763,347	8	4	2
158	1,421	380,303	393,382	773,685	10	6	4	362,612	400,227	762,839	8	4	2
160	1,397	377,438	395,549	772,987	10	6	4	362,104	400,227	762,331	8	4	2
162	1,395	376,905	395,549	772,454	10	6	4	361,596	400,227	761,824	8	4	2
164	1,260	375,404	395,549	770,953	10	6	4	361,128	400,227	761,355	9	4	2
166	1,166	374,304	395,549	769,853	10	6	4	360,750	400,227	760,977	9	4	2
168	1,051	373,048	395,549	768,598	10	6	4	360,371	400,227	760,599	9	4	2
170	1,004	372,304	395,549	767,853	10	6	4	359,993	400,227	760,220	9	4	2

TAB. 3.3 – Variation de csc_1

csc_2	Dif.Rel.	PU1G	PU2G	PT1-2G	BU1G	BU2G	DAPG	PU1L	PU2L	PT1-2L	BU1L	BU2L	DAPL
210	3,530	389,085	405,750	794,835	7	6	3	350,653	417,074	767,728	5	3	1
212	3,787	392,314	401,494	793,809	7	6	4	350,653	414,190	764,844	5	3	1
214	4,059	392,314	400,579	792,894	7	6	4	350,653	411,307	761,960	5	3	1
216	2,487	392,314	399,665	791,979	7	6	4	363,997	408,759	772,757	6	4	1
218	1,426	394,930	396,137	791,067	7	4	2	373,298	406,644	779,943	6	4	2
220	1,564	394,930	395,350	790,280	7	7	4	373,298	404,811	778,110	6	4	2
222	1,702	394,930	394,563	789,493	7	7	4	373,298	402,977	776,276	6	4	2
224	1,841	394,930	393,775	788,705	7	7	4	373,298	401,144	774,442	6	4	2
226	1,091	394,930	392,988	787,918	7	7	4	379,976	399,431	779,408	6	5	2
228	1,187	394,930	392,201	787,131	7	7	4	379,976	397,919	777,896	6	5	2
230	1,282	394,930	391,414	786,344	7	7	4	379,976	396,4083171	776,385	5	5	2
232	1,378	394,930	390,627	785,557	7	7	4	379,976	394,896	774,873	6	5	2
234	0,817	397,083	387,725	784,808	7	7	5	385,015	393,428	778,443	7	5	3
236	0,894	397,083	387,043	784,126	7	7	5	385,015	392,163	777,178	7	5	3
238	0,970	397,083	386,362	783,445	7	7	5	385,015	390,898	775,914	7	5	3
240	0,512	397,083	385,680	782,763	7	7	5	389,085	389,689	778,7757	7	6	3
242	0,562	397,083	384,998	782,082	7	7	5	389,085	388,619	777,705	7	6	3
244	0,613	397,083	384,317	781,400	7	7	5	389,085	387,548	776,634	7	6	3
246	0,668	398,879	381,867	780,747	7	8	5	389,085	386,477	775,563	7	6	3
248	0,730	398,879	381,274	780,153	7	8	5	389,085	385,407	774,493	7	6	3
250	0,793	398,879	380,681	779,560	7	8	5	389,085	384,336	773,4224	7	6	3
252	0,856	398,879	380,087	778,967	7	8	5	389,085	383,265	772,351	7	6	3
254	0,487	398,879	379,494	778,374	7	8	5	392,314	382,286	774,601	7	6	4
256	0,529	398,879	378,901	777,780	7	8	5	392,314	381,372	773,686	7	6	4
258	0,579	400,394	376,858	777,253	7	8	6	392,314	380,457	772,772	7	6	4
260	0,283	400,394	376,340	776,734	7	8	6	394,930	379,606	774,536	7	7	4
262	0,318	400,394	375,821	776,216	7	8	6	394,930	378,819	773,749	7	7	4
264	0,353	400,394	375,303	775,697	7	8	6	394,930	378,032	772,962	7	7	4
266	0,389	400,394	374,784	775,178	7	8	6	394,930	377,245	772,175	7	7	4
268	0,424	400,394	374,265	774,660	7	8	6	394,930	376,457	771,387	7	7	4
270	0,459	400,394	373,747	774,141	7	8	6	394,930	375,670	770,600	7	7	4

TAB. 3.4 – Variation de csc_2

$price_1$	Dif.Rel.	PU1G	PU2G	PT1-2G	BU1G	BU2G	DAPG	PU1L	PU2L	PT1-2L	BU1L	BU2L	DAPL
100	0	-35,875	832,840	796,964	7	8	4	-35,875	832,840	796,964	7	8	4
102	0	-18,557	815,136	796,579	7	8	4	-18,557	815,136	796,579	7	8	4
104	0	-1,239	797,4332	796,194	7	8	4	-1,239	797,433	796,1940	7	8	4
106	0	16,079	779,729	795,808	7	8	4	16,079	779,729	795,808	7	8	4
108	0	33,397	762,025	795,423	7	8	4	33,397	762,025	795,423	7	8	4
110	0	50,715	744,321	795,037	7	8	4	50,715	744,321	795,037	7	8	4
112	0	68,034	726,618	794,652	7	8	4	68,034	726,618	794,652	7	8	4
114	0,021	85,352	708,914	794,267	7	8	4	85,099	708,994	794,093	7	7	4
116	0,013	102,671	691,210	793,881	7	8	4	102,312	691,460	793,772	7	7	4
118	0,005	119,989	673,507	793,496	7	8	4	119,524	673,926	793,451	7	7	4
120	0,080	136,737	656,392	793,129	7	7	4	136,034	656,455	792,490	7	7	3
122	0,082	153,950	638,858	792,808	7	7	4	153,119	639,034	792,154	7	7	3
124	0,084	171,163	621,324	792,487	7	7	4	170,205	621,613	791,818	7	7	3
126	0,086	188,376	603,790	792,166	7	7	4	187,290	604,192	791,482	7	7	3
128	0,256	205,589	586,256	791,845	7	7	4	202,863	586,953	789,817	7	6	3
130	0,250	222,801	568,722	791,524	7	7	4	219,792	569,750	789,543	7	6	3
132	0,244	240,014	551,188	791,202	7	7	4	236,722	552,547	789,269	7	6	3
134	0,239	257,227	533,654	790,881	7	7	4	253,651	535,344	788,995	7	6	3
136	0,233	274,440	516,120	790,560	7	7	4	270,580	518,141	788,722	7	6	3
138	0,227	291,653	498,586	790,239	7	7	4	287,510	500,938	788,448	7	6	3
140	0,599	308,865	481,052	789,918	7	7	4	301,340	483,868	785,208	7	5	3
142	1,089	326,078	463,518	789,596	7	7	4	314,023	467,063	781,087	6	5	2
144	1,078	343,291	445,984	789,275	7	7	4	330,511	450,344	780,856	6	5	2
146	1,066	360,504	428,450	788,954	7	7	4	347,000	433,625	780,625	6	5	2
148	1,055	377,717	410,916	788,633	7	7	4	363,488	416,906	780,394	6	5	2
150	1,911	394,930	393,382	788,312	7	7	4	373,298	400,227	773,526	6	4	2
152	1,892	412,142	375,848	787,991	7	7	4	389,465	383,887	773,352	6	4	2
154	1,874	429,355	358,314	787,669	7	7	4	405,631	367,547	773,179	6	4	2
156	1,855	446,568	340,780	787,348	7	7	4	421,797	351,207	773,005	6	4	2
158	1,836	463,781	323,246	787,027	7	7	4	437,964	334,867	772,832	6	4	2
160	5,539	480,994	305,712	786,706	7	7	4	426,233	319,180	745,414	5	3	1
162	5,515	498,206	288,178	786,385	7	7	4	441,350	303,927	745,277	5	3	1
164	5,492	515,419	270,644	786,064	7	7	4	456,466	288,673	745,140	5	3	1
166	5,474	528,997	256,787	785,785	7	6	4	471,582	273,420	745,003	5	3	1
168	9,579	546,083	239,442	785,525	7	6	4	457,757	259,094	716,852	5	2	1
170	9,556	563,168	222,097	785,265	7	6	4	471,942	244,825	716,767	5	2	1

TAB. 3.5 – Variation de $price_1$

<i>price</i> ₂	Dif.Rel.	PU1G	PU2G	PT1-2G	BU1G	BU2G	DAPG	PU1L	PU2L	PT1-2L	BU1L	BU2L	DAPL
190	6,331	394,930	-146,617	248,312	7	7	4	373,298	-139,772	233,526	6	4	2
192,3	5,878	394,930	-128,617	266,312	7	7	4	373,298	-121,772	251,526	6	4	2
194	5,485	394,930	-110,617	284,312	7	7	4	373,298	-103,772	269,526	6	4	2
196	5,142	394,930	-92,617	302,312	7	7	4	373,298	-85,772	287,526	6	4	2
198	4,839	394,930	-74,617	320,312	7	7	4	373,298	-67,772	305,526	6	4	2
200	4,570	394,930	-56,617	338,312	7	7	4	373,298	-49,772	323,526	6	4	2
202	4,329	394,930	-38,617	356,312	7	7	4	373,298	-31,772	341,526	6	4	2
204	4,112	394,930	-20,617	374,312	7	7	4	373,298	-13,772	359,526	6	4	2
206	3,916	394,930	-2,617	392,312	7	7	4	373,298	4,227	377,526	6	4	2
208	3,738	394,930	15,382	410,312	7	7	4	373,298	22,227	395,526	6	4	2
210	3,575	394,930	33,382	428,312	7	7	4	373,298	40,227	413,526	6	4	2
212	3,426	394,930	51,382	446,312	7	7	4	373,298	58,227	431,526	6	4	2
214	3,289	394,930	69,382	464,312	7	7	4	373,298	76,227	449,526	6	4	2
216	3,162	394,930	87,382	482,312	7	7	4	373,298	94,227	467,526	6	4	2
218	3,045	394,930	105,382	500,312	7	7	4	373,298	112,227	485,526	6	4	2
220	2,936	394,930	123,382	518,312	7	7	4	373,298	130,227	503,526	6	4	2
222	2,835	394,930	141,382	536,312	7	7	4	373,298	148,227	521,526	6	4	2
224	2,740	394,930	159,382	554,312	7	7	4	373,298	166,227	539,526	6	4	2
226	2,652	394,930	177,382	572,312	7	7	4	373,298	184,227	557,526	6	4	2
228	2,569	394,930	195,382	590,312	7	7	4	373,298	202,227	575,526	6	4	2
230	2,491	394,930	213,382	608,312	7	7	4	373,298	220,227	593,526	6	4	2
232	2,417	394,930	231,382	626,312	7	7	4	373,298	238,227	611,526	6	4	2
234	2,348	394,930	249,382	644,312	7	7	4	373,298	256,227	629,526	6	4	2
236	2,283	394,930	267,382	662,312	7	7	4	373,298	274,227	647,526	6	4	2
238	2,221	394,930	285,382	680,312	7	7	4	373,298	292,227	665,526	6	4	2
240	2,163	394,930	303,382	698,312	7	7	4	373,298	310,227	683,526	6	4	2
242	2,107	394,930	321,382	716,312	7	7	4	373,298	328,227	701,526	6	4	2
244	2,054	394,930	339,382	734,312	7	7	4	373,298	346,227	719,526	6	4	2
246	2,004	394,930	357,382	752,312	7	7	4	373,298	364,227	737,526	6	4	2
248	1,957	394,930	375,382	770,312	7	7	4	373,298	382,227	755,526	6	4	2
250	1,911	394,930	393,382	788,312	7	7	4	373,298	400,227	773,526	6	4	2
252	1,868	394,930	411,382	806,312	7	7	4	373,298	418,227	791,526	6	4	2
254	1,826	394,930	429,382	824,312	7	7	4	373,298	436,227	809,526	6	4	2
256	1,786	394,930	447,382	842,312	7	7	4	373,298	454,227	827,526	6	4	2
258	1,748	394,930	465,382	860,312	7	7	4	373,298	472,227	845,526	6	4	2
260	1,712	394,930	483,382	878,312	7	7	4	373,298	490,227	863,526	6	4	2
262	1,677	394,930	501,382	896,312	7	7	4	373,298	508,227	881,526	6	4	2
264	1,643	394,930	519,382	914,312	7	7	4	373,298	526,227	899,526	6	4	2
266	1,611	394,930	537,382	932,312	7	7	4	373,298	544,227	917,526	6	4	2
268	1,580	394,930	555,382	950,312	7	7	4	373,298	562,227	935,526	6	4	2
270	1,550	394,930	573,382	968,312	7	7	4	373,298	580,227	953,526	6	4	2

TAB. 3.6 – Variation de *price*₂

Deuxième partie

Pilotage la production à court terme : ordonnancement dans les raffineries

Chapitre 4

Positionnement du problème

4.1 Introduction

Le plan de production à court terme, ou ordonnancement de la production, peut être vu comme une version détaillée et désagrégée du plan de production à moyen terme : il désagrège les familles de produits en articles individuels et il planifie la production sur un horizon plus court avec un découpage temporel plus fin.

L'étude présentée ici porte sur l'ordonnancement des activités de chargement/déchargement du pétrole brut dans les réservoirs de stockage intermédiaires, entre les navires et les unités de distillation du pétrole brut d'une raffinerie. Globalement, une raffinerie est un système constitué de ports, de pipelines, d'une série de réservoirs pour stocker le pétrole brut, d'Unités de Distillation de Pétrole Brut (**UDPB**), d'unités de raffinage et de réservoirs pour faire les mélanges et pour stocker les produits finis et les matières premières (cf. figure 4.1). Il y a deux niveaux de décision dans une raffinerie :

- la planification qui détermine les volumes des différentes matières de base à acheter ainsi que les mélanges et les quantités à produire sur plusieurs périodes,
- l'ordonnancement qui traite de la synchronisation des opérations à réaliser dans le cadre d'une planification connue en prenant en compte les contraintes opérationnelles du problème.

Une fois les quantités et les types de pétrole brut à recevoir et les demandes des UDPB (Unité de **D**istillation du **P**étrole **B**rut) sont donnés, les gestionnaires doivent ordonnancer les chargements et déchargements des réservoirs. De façon générale, la problématique est d'ordonnancer les transferts du pétrole brut des ports vers les réservoirs et des réservoirs vers les UDPB.

Le pétrole brut arrive à la raffinerie de plusieurs manières : par oléoduc, par navire pétrolier ou par la combinaison de ces deux moyens. Il est chargé dans les réservoirs où il séjourne quelques heures afin d'y être séparé de l'eau de mer. Ces réservoirs ont une capacité d'une

centaine de milliers de mètres cubes. Le pétrole brut doit y être stocké avant d'être distillé dans les UDPB. Finalement, il est déchargé vers les UDPB ou vers d'autres réservoirs par oléoduc. Il est très important que le pétrole brut soit déchargé vers les UDPB en continu à cause de pertes et d'un coût de lancement d'un déchargement appelé coût de reconfiguration.

Le système qui nous intéresse est un sous-système (cf. figure 4.2) du système présenté à la figure 4.1. Ce système est utilisé pour le stockage du pétrole brut arrivé à la raffinerie, la préparation des mélanges et l'alimentation des UDPB. Ce système se compose d'une série de réservoirs de grand stockage, de ports qui accueillent les bateaux apportant le pétrole brut et d'UDPB où les mélanges de pétrole brut sont distillés. De plus ce système se compose d'un grand réseau de pipelines qui connecte les ports avec les réservoirs et les réservoirs avec les UDPB. Finalement, dans ce système existent des pompes électriques et des mixeurs respectivement pour le chargement et le déchargement du pétrole brut et pour la préparation des mélanges demandés par les UDPB. L'objectif est d'ordonnancer les chargements et les déchargements du pétrole brut dans les réservoirs en minimisant le coût de reconfiguration des réservoirs disponibles. L'ordonnancement doit tenir compte de contraintes temporelles et de contraintes portant sur l'utilisation et la disponibilité des ressources. Il doit aussi prendre en compte les dimensions du parc de ressources et la capacité de chaque ressource ainsi que les modes et les conditions de fonctionnement.

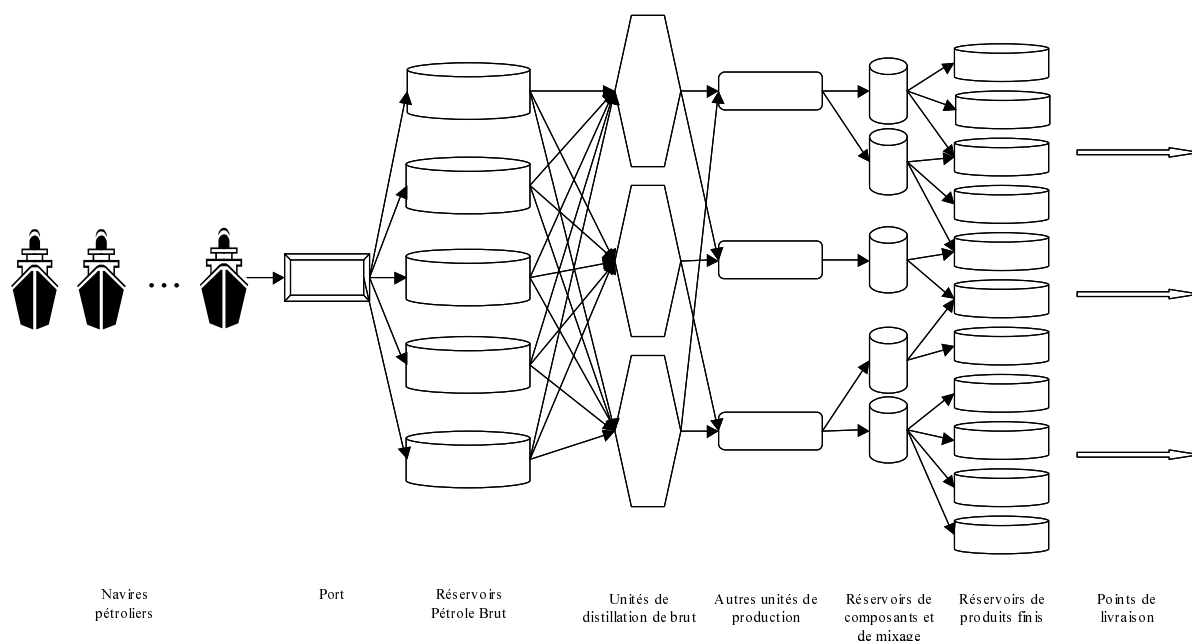


Fig. 4.1 – Système Générique de Raffinerie

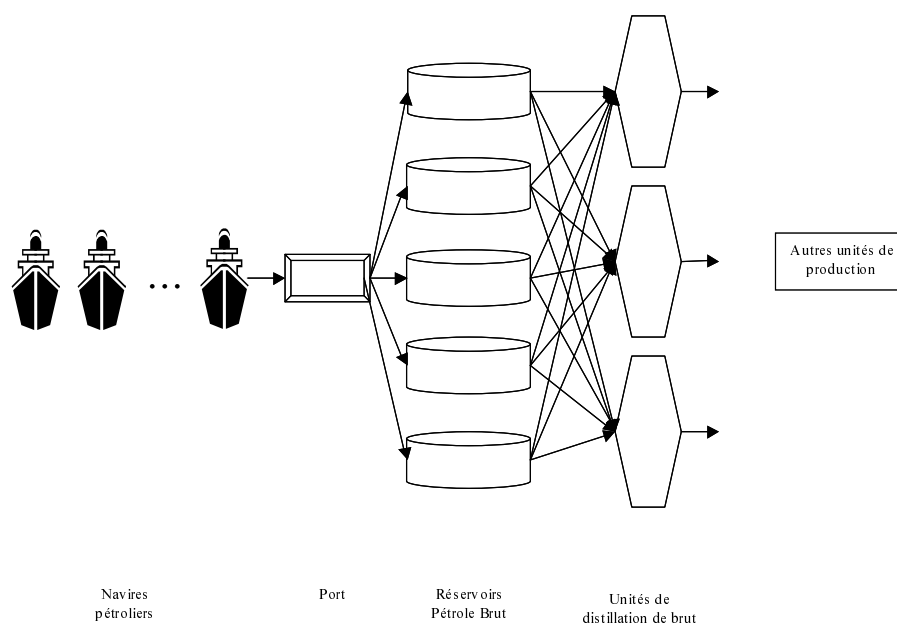


Fig. 4.2 – Système Etudié

Il faut donc décider pour tout l’horizon de planification quels réservoirs vont être chargés par les bateaux qui vont arriver au port et quels réservoirs vont être déchargés vers les Unités de Distillation de Pétrole Brut (UDPB), avec pour objectif de minimiser le coût de reconfiguration des réservoirs et du réseau des pipelines. Il existe plusieurs modes de préparation des mélanges et différentes options de distillation ce qui rend le problème plus complexe. La modélisation de ces deux aspects nécessite l’introduction d’un grand nombre de variables de décision et de contraintes dans le problème, engendrant une forte complexité combinatoire.

Dans une raffinerie typique il existe environ 6-10 réservoirs pour le stockage du pétrole brut. La capacité de stockage de chaque réservoir varie de 80000 m^3 à 150000 m^3 . Concernant les débits des pipelines, nous trouvons deux différents types. Le premier représente le débit entre les ports et les réservoirs correspondant au débit de chargement. Ce débit dépend de la capacité de déchargement des bateaux et peut varier de $1000 \text{ m}^3/h$ à $5000 \text{ m}^3/h$. Le deuxième type de débit est celui du déchargement des réservoirs vers les UDPB. Ce débit correspond à celui de la distillation de chaque UDPB. Chaque UDPB a un débit de distillation qui varie de $200 \text{ m}^3/h$ à $2000 \text{ m}^3/h$. Généralement dans une raffinerie typique on peut trouver 3 à 4 UDPB qui ont un débit de distillation de l’ordre de $2000 \text{ m}^3/h$ au total. Finalement une raffinerie a besoin de deux ports au maximum car les arrivées des bateaux ne sont pas assez fréquentes (3-4 au maximum par mois).

Un objectif important est de minimiser le coût de reconfiguration sur l'horizon considéré. Le coût de reconfiguration correspond au coût de reconfiguration des réservoirs chargés par les navires arrivés aux ports et déchargés vers les UDPB ainsi qu'à celui du réseau des pipelines. Ce coût est l'un des coûts les plus importants dans l'ordonnancement du pétrole brut d'une raffinerie. Une reconfiguration d'un réservoir, quelle que soit son utilisation (chargement-déchargement), nécessite en effet une série d'opérations relativement longue et coûteuses (cf. section 4.4).

Une fois le système modélisé, les données et les paramètres d'entrée de ce problème sont : les dimensions du parc de ressources, le programme d'arrivée des bateaux résultant de la planification à moyen terme, la demande en pétrole brut par les unités de distillation, les conditions initiales du système (quantité et composition dans chaque réservoir entre autres) et les détails concernant le fonctionnement (temps opératoires, débits limités). De plus, le nombre de réservoirs disponibles pour le stockage du pétrole brut et leur capacité de stockage sont connus. Finalement, le débit de distillation de chaque UDPB est fixé a priori ainsi que l'option de distillation et le type de préparation des mélanges choisis.

La modélisation de ce problème nécessite un ensemble de variables de décision et de contraintes. Les variables de décision continues correspondent principalement aux flux port $\Rightarrow$ réservoir, aux flux réservoirs $\Rightarrow$ unités de distillation et aux quantités stockées dans les réservoirs. On utilise des variables de décision binaires pour les décisions liées à l'établissement d'une connexion port $\Rightarrow$ réservoir, à l'établissement d'une connexion réservoir $\Rightarrow$ unité de distillation, à la disponibilité des réservoirs, et à la reconfiguration des réservoirs. Les contraintes restreignant l'ensemble des solutions réalisables correspondent :

- aux contraintes de fonctionnement,
- aux équilibres et bilans de matière,
- au respect des capacités de stockage,
- aux propriétés des mélanges,
- à l'établissement des connexions entre les ports et les réservoirs,
- à l'établissement des connexions entre les réservoirs et les UDPB,
- aux reconfigurations des réservoirs.

Enfin, l'ordonnancement en raffinerie est un problème complexe qui requiert simultanément une solution pour les flux de pétrole brut, l'allocation de navires aux réservoirs, l'allocation de réservoirs aux UDPB (Unité de Distillation de Pétrole Brut) et le calcul des compositions de pétrole brut. Cet ordonnancement se caractérise en outre par des décisions discrètes ainsi que par diverses relations non linéaires relatives aux mélanges. Plusieurs auteurs ont tenté de

développer des modèles et des techniques pour l'ordonnancement en raffinerie et en particulier l'ordonnancement des activités de chargement et déchargement du pétrole brut.

Dans ce qui suit, nous présentons une vue d'ensemble de la littérature sur le problème d'ordonnancement du pétrole brut en proposant une classification qui nous a semblé pertinente. Nous évoquons les différents types des modèles développés. Ensuite, nous faisons une étude approfondie du problème et présentons les principes d'un modèle générique qui tient compte de tous les modes de préparation de mélanges et des différentes options de distillation. Pour une meilleure compréhension du problème, nous illustrons notre démarche sur un exemple numérique réel d'une raffinerie grecque. Après la présentation des modèles, nous faisons une comparaison entre la solution obtenue par la modélisation proposée et la solution sous-optimale obtenue par l'implémentation de la méthode actuellement utilisée par les décideurs de la raffinerie. Dans les chapitres suivants, nous donnons les différentes méthodes développées pour améliorer l'efficacité de la résolution et nous présentons une stratégie de décomposition du modèle initial. Nous comparons de plus les différentes méthodes en terme de critères de qualité de la solution obtenue et du temps de résolution. Finalement, dans le chapitre 6, nous présentons une série d'inégalités valides et une extension de la méthode de décomposition de Benders implémentée pour améliorer l'efficacité du modèle décomposé proposé au chapitre 5. Nous concluons en indiquant quelques perspectives d'avenir pour ce travail de recherche.

4.2 Vue d'ensemble

Historiquement, les problèmes de planification de raffinerie ont été abordés dès l'introduction de la programmation linéaire dans les années 1950. En effet, [Symonds, 1955] et [Manne, 1956] appliquent les techniques de la programmation linéaire à la planification des approvisionnements et de la production dans les unités de traitement de pétrole brut. Par la suite, des logiciels pour la planification de la production en raffinerie fondés sur la programmation linéaire ont vu le jour. Nous pouvons citer celui de [Bonner and Moore, 1979]. Ces travaux ont permis beaucoup de progrès au niveau de la planification globale d'une raffinerie. Notons que le développement et la maturité de la recherche sur la planification de production à moyen terme en raffinerie sont bien supérieurs à l'état de la recherche relatif à l'ordonnancement dans ce même secteur.

Cette partie présente l'ensemble des travaux menés sur l'ordonnancement de raffinerie au cours des deux dernières décennies. La classification qui nous a semblé pertinente est la suivante. Nous avons d'une part regroupé les approches utilisant des méthodes de programmation mathématique linéaire (MILP) ou non linéaire (MINLP). Nous avons classé ces approches selon la nature de la représentation temporelle employée : temps discret ou temps continu. D'autre part, nous avons distingué les approches utilisant des méthodes heuristiques explicites.

4.2.1 MILP/MINLP à temps discret

Chronologiquement, les premières approches par la programmation mathématique du problème d'Ordonnancement des activités de **C**hargement/**D**échargement du pétrole brut (**OCD**) ont utilisé une formulation en temps discret.

L'un des principes de la représentation en temps discret est de subdiviser l'horizon de planification en un ensemble d'intervalles de taille égale. Si nous décidons de subdiviser l'horizon en N périodes de temps alors, pour chaque décision, on devra définir N variables binaires. Chacune précisera si l'action est oui ou non en cours à la période considérée. Par conséquent, entre deux pas successifs, si la valeur de la variable binaire passe de 0 à 1 ou de 1 à 0, cela implique que l'action a commencé ou s'est achevée en début de période.

La première modélisation rigoureuse de ce problème a été faite par [Shah, 1996]. A cause de la complexité et de la non-linéarité du modèle initial, l'auteur utilise une décomposition du problème en deux sous-problèmes de type MILP (Mixed Integer Linear Program). Par la suite, des formulations en temps discret utilisant un modèle unique pour résoudre le problème OCD ont vu le jour. Nous citons [Lee et al., 1996], [Li et al., 2002] ainsi que [Reddy et al., 2004a]. Ces derniers ont abouti à des MINLP (Mixed Integer Non Linear Program) qu'ils ont résolu par l'introduction de reformulations ou de divers algorithmes de résolution successifs impliquant des MILP. Dans la suite de cette partie, nous présentons ces approches de formulation et de résolution successives.

L'auteur de [Shah, 1996] présente un modèle à temps discret pour l'ordonnancement du pétrole brut conduisant à la résolution d'un programme linéaire en variables mixtes (MILP : Mixed Integer Linear Problem). Dans cette formulation, le problème a été décomposé en un sous-problème amont, considérant le déchargement et le stockage du brut dans les réservoirs du port, et un sous-problème aval, impliquant le chargement des réservoirs et les opérations de distillation de pétrole. Nous pouvons qualifier cette approche d'approximation raisonnable car la solution obtenue est une solution réalisable mais pas forcément optimale. L'objectif est de minimiser la quantité de matière, nommée *talon*, laissée dans un réservoir après que celui-ci ait été utilisé pour alimenter une UDPB. Dans [Shah, 1996], l'auteur propose un modèle qui résout le problème d'**OCD** (Ordonnancement des activités de **C**hargement/**D**échargement du pétrole brut) pour un système où les réservoirs disponibles ne peuvent alimenter qu'une seule UDPB à la fois et où une UDPB ne peut être alimentée que par un seul réservoir à la fois. De plus, l'auteur subdivise l'horizon de planification en intervalles de durées égales. Chaque activité doit commencer et se terminer aux bornes de ces intervalles. Enfin, le système étudié ne prend en compte qu'un seul port mais l'extension du modèle à un système permettant d'accueillir plu-

sieurs bateaux à la fois nous semble facile.

Une autre approche pour la résolution du problème est présentée par [Lee et al., 1996]. Les auteurs ont proposé un modèle MILP pour la minimisation des coûts opératoires liés au déchargement du pétrole brut, à son stockage et à son chargement. Les termes bi-linéaires d'une catégorie de contraintes résultant des opérations de mélange de bruts sont linéarisés. Cependant, et comme cela a été noté par [Li et al., 2002] ainsi que par [Reddy et al., 2004a], cette linéarisation mène à une anomalie des compositions de brut dans le chargement des UDPB. La formulation proposée pour aboutir à la linéarisation des contraintes bi-linéaires tient compte des règles opératoires générales existant dans toutes les raffineries mais elle est limitée par les contraintes suivantes :

- chaque réservoir peut alimenter au plus une UDPB sur un intervalle de temps,
- chaque UDPB ne peut être chargée que par un seul réservoir sur un intervalle de temps considéré.

Dans [Li et al., 2002], les auteurs proposent un algorithme de combinaison itérative d'un MILP et d'un NLP, afin de résoudre l'anomalie de composition constatée dans le précédent modèle. Ils tentent de réduire le nombre de variables binaires en agrégeant les variables binaires tri-indexées en variables bi-indexées. Ils ont aussi incorporé de nouvelles dispositions comme des ports multiples mais ils conservent la contrainte qu'une UDPB ne peut être alimentée que par au plus deux réservoirs sur une période donnée de l'horizon. Leur algorithme nécessite la résolution d'un NLP (Problème Non Linéaire ou Non Linear Problem) à chaque itération et il ne permet pas de garantir l'obtention d'une solution réalisable.

Les auteurs de [Reddy et al., 2004a] ont proposé un modèle MILP itératif, dit hybride et en temps discret, traité aussi dans [Reddy et al., 2004b]. Le nombre de variables binaires d'allocation a été réduit de façon importante. De plus, des variables continues définies sur $[0, 1]$ ont été introduites afin de faciliter la résolution de problèmes plus grands. Un aspect intéressant de leur modèle réside dans le fait qu'il se rapproche d'une représentation continue du temps en permettant que plusieurs chargements aient lieu au cours d'une période. Le modèle proposé tient compte de nouvelles possibilités en terme de configuration du système, à savoir l'emploi de jetées multiples. Dans [Reddy et al., 2004a], les auteurs proposent une formulation qu'ils qualifient d'hybride discrète-continue. L'objectif est ici fondé sur une maximisation d'une fonction profit plus réaliste, incluant des pénalités sur les stocks de sécurité et une pénalité due en cas d'attente d'un bateau en mer à partir d'une date échue. Les auteurs ne tiennent pas compte du coût de reconfiguration des réservoirs car les pénalités payées par la mauvaise planification à moyen terme sont prioritaires. L'algorithme proposé pour la résolution est une procédure

itérative basée sur une résolution de MILP, évitant le recours supplémentaire à la résolution de NLP comme cela était le cas dans [Li et al., 2002].

Les auteurs de [Pinto et al., 2000] ont développé deux modèles : un modèle pour la planification et un autre pour l'ordonnancement. Concernant le modèle d'ordonnancement du pétrole brut, ils ont développé un modèle linéaire en nombre entier (MILP) pour le problème d'Ordonnancement des activités de **C**hargement/**D**échargement du pétrole brut (**OCD**) entre les navires et les Unités de Distillation du Pétrole Brut (UDPB) dans une raffinerie au Brésil. La modélisation proposée a pour objectif de minimiser le nombre des réservoirs utilisés pour le chargement et le déchargement du pétrole brut. L'horizon de temps utilisé est d'une semaine et le modèle proposé tient compte de toutes les contraintes générales de fonctionnement mais pas de tous les types de préparation de mélanges ni de toutes les options de distillation.

Les auteurs de [Neiro and Pinto, 2002] proposent une extension du modèle proposé par [Pinto et al., 2000]. Les auteurs modélisent de façon assez complète une raffinerie au Brésil en prenant en compte un grand nombre d'opérations qui ont lieu dans cette raffinerie. Avant de présenter l'étude de cas et le MILNP de grande taille proposé, les auteurs donnent une description détaillée du problème de planification d'une raffinerie telle qu'il a été traité jusqu'aux années 90. Plusieurs modèles ont été développés par les auteurs avant d'aboutir au modèle complet d'optimisation du système entier. Le premier modèle correspond à toutes les unités d'opérations, le deuxième modèle correspond aux opérations effectuées sur les réservoirs et finalement le troisième concerne les transferts du pétrole brut, des matières premières et des produits finis entre les réservoirs et les unités d'opération. Le modèle complet est présenté après l'illustration des trois modèles précédents. En conclusion, les auteurs proposent la linéarisation des contraintes non-linéaires en appliquant des méthodes de décomposition qui permettent en même temps de résoudre plusieurs scénarii en parallèle.

Finalement, l'article de [Persson and Gothe-Lundgren, 2005] présente un autre travail dans le domaine de la planification et de l'ordonnancement en raffinerie. Le problème étudié consiste à ordonnancer les opérations de raffinage dans les UDPB et dans les autres unités de raffinage dans une raffinerie en Suède. L'horizon de temps utilisé est égal à un mois. Les auteurs considèrent des intervalles de temps de durée égale et cherchent à ordonnancer l'utilisation des unités des raffinage. La demande en produits finis est donnée et l'objectif est de minimiser le coût total de production. De plus, les auteurs proposent des extensions à leur modèle notamment dans le cas où les dates d'arrivées des bateaux sont soumises à des aléas et où on connaît seulement une fenêtre de temps où l'arrivée du bateau est probable.

4.2.2 MILP/MINLP à temps continu

Ces dernières années, une autre famille de modèles d'ordonnancement utilisant une représentation continue du temps est apparue. La nécessité de prédéfinir des intervalles de temps de longueurs égales pose en effet des problèmes dans certains cas. Dans les modèles à temps continu, l'instant de début, de même que l'instant de fin, de chaque activité est une variable que la solution du modèle détermine. La modélisation en temps continu convient particulièrement à l'ordonnancement du pétrole brut car les différentes activités d'une raffinerie ont des durées très variables : de quelques minutes à plusieurs heures [Joly et al., 2002].

Un modèle à temps discret nécessiterait un grand nombre d'intervalles de temps augmentant parfois excessivement la complexité combinatoire du problème. Ainsi, le principal avantage d'une représentation continue du temps est la possibilité de réduire la complexité par l'introduction d'un nombre réduit de variables binaires dans le modèle. En outre, un modèle en temps continu présente l'avantage d'une meilleure précision au niveau de la solution obtenue. Cependant, une formulation en temps continu a aussi des inconvénients dont le plus critique est l'apparition d'un grand nombre de contraintes non linéaires.

Dans [Joly et al., 2002], une formulation en temps continu du problème d'ordonnancement en raffinerie est proposé. L'article décrit la modélisation du problème d'ordonnancement au coeur de la raffinerie mais sans donner tous les détails du modèle relatif au problème de l'ordonnancement du pétrole brut. Dans la partie de l'article [Joly et al., 2002] réservée au problème **OCD** une étude de cas réel d'une raffinerie recevant différents types de bruts approvisionnés par oléoduc est présentée. On remarque qu'il y a un seul type de réservoirs qui sont à la fois des réservoirs de stockage et de mélange. Les temps de transfert peuvent là aussi varier de quelques minutes à quelques heures. Cette disparité des temps de transfert donne un argument supplémentaire pour justifier l'intérêt de recourir à cette modélisation en temps continu. La solution obtenue est celle résultant de la linéarisation des contraintes non linéaires du modèle développé. Cette linéarisation augmente la taille du problème mais assure la faisabilité de la solution obtenue.

Dans [Jia et al., 2003] puis dans [Jia and Ierapetritou, 2004] est présenté un modèle d'ordonnancement des opérations en raffinerie fondé sur une formulation en temps continu. Les auteurs ont divisé les opérations de raffinerie en trois sous-problèmes. Le premier comporte les opérations relatives au pétrole brut (OCD), le second concerne les procédés de raffinage et les réservoirs intermédiaires et le troisième est relatif aux fins de processus et aux opérations de mélanges finaux. L'article a traité seulement du premier sous-problème dans lequel est utilisé la contrainte d'équilibre des composants, introduit par [Lee et al., 1996]. Le coût de reconfigu-

ration des réservoirs n'a pas été pris en compte par ce modèle. Comme nous l'avons indiqué précédemment, le chargement et le déchargement du pétrole brut ainsi que la reconfiguration des réservoirs sont des activités opérationnelles importantes dans une raffinerie pouvant donner lieu à des pertes élevées. Le modèle proposé ne permet pas une grande variété de configurations comme la possibilité d'alimentation d'une UDPB par plusieurs réservoirs ou encore la possibilité pour un réservoir d'alimenter plusieurs UDPB car l'application industrielle ne l'a pas rendue nécessaire. Finalement les deux articles introduisent une nouvelle idée de modélisation. Les auteurs proposent de formaliser la décomposition structurelle du problème d'ordonnancement en raffinerie en plusieurs sous-problèmes afin de pouvoir les résoudre en un temps raisonnable. Les solutions obtenues sont des solutions exactes pour chaque sous-problème mais pas optimale pour le système entier.

Les auteurs de [Reddy et al., 2004b] et de [Reddy et al., 2004a] proposent également un modèle en temps continu pour l'ordonnancement du pétrole brut dans une raffinerie. Les auteurs disent que la formulation proposée est la première formulation complète pour le problème OCD en MILP appliquée en industrie pétrolière. Le modèle proposé est un modèle assez complet mais il ne tient pas compte de tous les types de préparation de mélange et de toutes les options de distillation car il correspond à une application industrielle. De plus, les auteurs font une comparaison entre leur modèle en temps continu et leur modèle en temps discret pour montrer la nécessité d'une modélisation en temps continu pour leur application.

4.2.3 Approches heuristiques

Pour plusieurs problèmes de taille industrielle, résoudre le MILP en utilisant les outils commerciaux basés sur les méthodes de Branch & Bound ou les méthodes de Branch & Cut peut être difficile et, dans certains cas, sans intérêt du fait de la complexité associée (grand nombre de variables et des contraintes non linéaires). Au lieu de cela, plusieurs chercheurs ont conçu des algorithmes heuristiques permettant de résoudre des problèmes de grande taille et de complexité importante. L'inconvénient d'une approche heuristique est que ni l'optimalité globale, ni même la faisabilité globale, ne peuvent être garanties. En pratique, ces méthodes ont fait leurs preuves sur des problèmes parfois trop grands pour pouvoir être résolus par les méthodes conventionnelles seules [Kelly and Mann, 2002].

Dans [Kelly and Mann, 2002] les auteurs présentent une heuristique dite de décomposition chronologique (CDH : *Chronological Decomposition Heuristic*). CDH est une stratégie simple de type « diviser pour régner » (*divide-and-conquer*) fondée sur un fractionnement de l'horizon, pour trouver rapidement une solution réalisable à des problèmes d'ordonnancement de grande taille. CDH a été spécialement conçue pour les problèmes d'ordonnancement de la production

en industrie pétrolière et pétrochimique. Le principe de base de CDH est de découper l'horizon d'ordonnancement en morceaux de temps (*time-chunks*) qui sont des multiples du pas de temps de base. Les morceaux de temps peuvent être de longueurs non uniformes selon la catégorie du problème et le cas traité. Le problème d'ordonnancement sur chaque fraction de l'horizon est résolu en utilisant les techniques de la programmation mixte linéaire entière (MILP) en commençant par le premier morceau de temps (connaissant la situation initiale) et puis en avançant dans le temps. Finalement, si nécessaire, la technique du *backtracking* (retour en arrière) chronologique est utilisée. L'efficacité de l'heuristique s'explique par le fait qu'elle décompose le problème global portant sur tout l'horizon de planification en sous-problèmes portant sur une fraction restreinte de l'horizon. Ces sous-problèmes de petite taille sont résolus successivement et fournissent finalement une solution pour le problème global. L'idée de base d'une telle stratégie de décomposition a été présentée partiellement dans l'article [Bassett et al., 1996]. Enfin, CDH doit être considérée comme une étape aidant le responsable de l'ordonnancement à trouver rapidement des solutions faisables de qualité raisonnable.

[Kelly, 2003] décrit une heuristique primale simple et efficace permettant la diminution du nombre de variables binaires avant qu'une heuristique de recherche énumérative implicite ne soit appliquée pour trouver une solution entière. Elle s'applique de manière générale aux problèmes complexes d'ordonnancement. Le principe de base de cette technique est d'employer des fonctions de lissage (*smoothing functions*) bien connues utilisées pour résoudre des problèmes de complémentarité au problème d'optimisation locale qui consiste à minimiser la somme pondérée des variables binaires multipliées par leurs compléments. L'algorithme accélérateur de lissage et d'exploration en profondeur (SDA : *Smooth & Dive Accelerator*) consiste à résoudre des relaxations successives du programme linéaire avec les fonctions de lissage ajoutées à l'objectif du problème courant puis à utiliser, si nécessaire, des variables binaires dites de plongée (*diving*) ou d'exploration en profondeur. Si les termes de la fonction de lissage ne s'annulent pas, un algorithme de Branch & Bound ou de Branch & Cut est appelé pour clore la procédure et trouver au moins une solution primale entière faisable. L'efficacité de l'heuristique est illustrée dans l'article par son application au problème du OCD. En outre, une série de fonctions de lissage connues est proposée.

4.3 Objectifs de l'étude (ordonnancement du pétrole brut)

4.3.1 Introduction

Les responsables de l'ordonnancement suivent continuellement les mouvements du pétrole brut et les associent à une demande fluctuante sur les produits finis. Dans la majorité des cas, soumis à des contraintes de délais très fortes et face à un système de production peu flexible, le décideur compte largement sur sa propre expérience et choisit souvent la première solution

réalisable résultant de critères heuristiques qu'il a lui-même développées.

Notre projet de recherche porte sur le développement d'une modélisation générique qui nous permette de trouver l'ordonnancement optimal des chargements et des déchargements des réservoirs dans une raffinerie pendant une période de temps allant jusqu'à 720 heures, ce qui correspond à 30 jours de fonctionnement.

Les opportunités de gain financier et opérationnel sont énormes, car un outil avancé d'ordonnancement permettrait :

- d'optimiser les ruptures ou manques de pétrole brut,
- d'évaluer la meilleure manière de mettre en oeuvre les plans de production périodiques,
- de fournir des directives aux opérations,
- de rendre possible la ré-optimisation en présence d'aléas,
- de comparer les résultats opérationnels actuels aux objectifs d'ordonnancement et de planification.

L'opération d'ordonnancement la plus critique est le chargement/déchargement des réservoirs de pétrole brut. Une meilleure analyse de cette opération d'ordonnancement permet de bénéficier d'une capacité supérieure, grâce à une meilleure utilisation des ressources, ainsi que d'une amélioration de la visibilité globale et du contrôle sur la chaîne logistique. Le besoin d'une méthodologie systématique et informatisée pour l'optimisation de cette opération en raffinerie se justifie et s'exprime donc clairement.

Jusqu'ici, une modélisation générique n'avait pas encore été présentée. Les raisons sont multiples mais les plus importantes sont les suivantes. Premièrement, les besoins spécifiques de chaque raffinerie n'avaient pas permis aux spécialistes de développer une modélisation générique car leur objectif était de développer un modèle qui correspond exactement au cas étudié. De plus, les problèmes de non-linéarité résultant d'une modélisation générique n'étaient pas résolus et le grand nombre de variables de décision et de contraintes restait un inconvénient pour le développement d'une formulation générique. Nos travaux de recherche ont visé à éliminer ces inconvénients en explorant les pistes suivantes :

- en remplaçant chaque fois que possible des contraintes initialement non linéaires par des contraintes linéaires,
- en développant des méthodes pour la diminution du nombre de variables de décision et de contraintes (grâce à une nouvelle façon de partitionner l'horizon de temps en périodes),
- en développant des inégalités valides pour diminuer l'espace des solutions,
- en appliquant des méthodes de décomposition classiques et en proposant des améliorations

algorithmiques de ces méthodes.

4.3.2 Approche proposée par rapport à la littérature

Aucune des approches présentées dans la littérature n'est à la fois assez générique et exacte. Ceci est à l'origine de notre motivation pour développer une nouvelle approche exacte. Cette dernière fournit une solution meilleure que les solutions fournies par les approches heuristiques et prend en compte tous les types de configuration possibles d'une raffinerie. Notre décision de développer un modèle exact en temps discret et non en temps continu découle de deux raisons principales. La première raison est que, si nous modélisons le système avec une formulation en temps continu, nous aurons toujours des contraintes non linéaires, ce qui implique un modèle plus compliqué et plus difficile à résoudre. En revanche, la modélisation en temps discret nous permet d'avoir le plus possible de contraintes linéaires. La deuxième raison pour laquelle nous n'avons pas fait une modélisation en temps continu est l'absence de besoin d'une grande précision temporelle parce que la solution optimale n'est pas sensible à un décalage de quelques minutes. La précision de l'ordre d'une minute obtenue par une modélisation en temps continu est inutile. Il suffit de savoir dans quel intervalle de temps (de l'ordre d'une à quelques heures) le chargement/déchargement d'un réservoir va démarrer.

L'un des principaux avantages de la formulation discrète est qu'elle fournit une grille temporelle de référence pour toutes les opérations en concurrence pour une ressource partagée. Cela peut rendre l'écriture des contraintes du problème d'ordonnancement plus simple. Le recours à la formulation en temps continu reste cependant une alternative possible dans certains cas, notamment lors qu'il y a une grande disparité dans les temps opératoires. De même, les approches heuristiques présentent un intérêt lorsque la taille du problème rend impossible l'application des méthodes exactes du fait de la complexité associée. Elle peuvent également servir pour obtenir une bonne solution réalisable initiale et ainsi accélérer la convergence d'une méthode exacte.

Nous remarquons aussi que l'ensemble des approches développées traite de la modélisation et de la résolution sur des cas particuliers sous différentes configurations. Ces dernières sont notablement enrichies dans les articles les plus récents. Toutefois, nous observons l'absence d'une modélisation générique capable d'englober tous les cas possibles. Les chercheurs préfèrent se concentrer sur l'étude de la complexité liée à la modélisation d'un cas particulier, le développement d'un modèle générique est donc nécessaire. Notre modèle est plus générique car il laisse la liberté de charger plusieurs réservoirs en même temps par bateau, limites imposées par [Lee et al., 1996], de décharger plusieurs réservoirs vers une UDPB ou d'alimenter plusieurs UDPB par un seul réservoir, limite imposée par [Li et al., 2002] et [Neiro and Pinto, 2002]. De plus, notre modélisation se différencie des travaux précédents par la fonction objectif choisie : la

minimisation du coût de reconfiguration des réservoirs qui est un enjeu économique important (cf. section 4.4).

Les travaux de [Pinto et al., 2000] concernent un problème proche de celui qui est étudié ici. L'approche que nous proposons s'en distingue néanmoins sur les points suivants :

- le nombre de réservoirs chargés en même temps : [Pinto et al., 2000] le limitent à un, notre modèle n'impose pas de contraintes sur ce nombre,
- la préparation des mélanges : [Pinto et al., 2000] ne l'autorisent que dans les réservoirs, nous laissons la possibilité de faire cette préparation dans les réservoirs, les pipelines ou avec une combinaison des deux,
- la distillation : [Pinto et al., 2000], la distillation se fait librement, sans contrainte de respect des compositions. Cette hypothèse, adaptée à leur cas industriel, n'est cependant pas réaliste dans la majorité des cas. Notre modèle peut prendre en compte toutes les options de distillation,
- l'horizon de la planification : [Pinto et al., 2000] se limitent à une semaine pour ne pas trop augmenter les temps de calcul. Nous proposons une nouvelle partition de l'horizon de temps où on subdivise l'horizon de planification en intervalles de durées qui ne sont pas égales (partition sur événement, cf. section 5.3). Cette partition de l'horizon de temps nous permet d'étendre l'horizon à un mois.

Un modèle générique plus global tel que celui que nous avons développé implique inévitablement une complexité plus importante. Les méthodes de décomposition mathématique nous semblent une piste intéressante et pratiquement jamais exploitée, à notre connaissance, pour la résolution de ce type d'application. On peut cependant noter une exception : les méthodes de décomposition structurelle du système, utilisées par [Shah, 1996] et [Jia and Ierapetritou, 2004]. Ces méthodes impliquent une approximation et ne permettent donc pas une résolution exacte. En revanche, la méthode de décomposition mathématique, présentée au chapitre 5.4, nous fournit des modèles donnant une solution exacte et optimale pour le problème OCD.

4.4 Problématique étudiée

L'objectif de notre recherche est de développer un modèle générique qui résout le problème de l'OCD (**O**rdonnancement des activités de **C**hargement/**D**échargement du pétrole brut) pour tous les types de configurations d'une raffinerie. Après avoir visité plusieurs raffineries, nous avons constaté qu'il existe plusieurs types de préparation de mélanges et plusieurs options de distillation. Nous pouvons citer trois types différents de préparation de mélanges :

- les mélanges sont préparés à l'aide des pipelines juste en amont des UDPB,
- les mélanges sont préparés dans les réservoirs,

- les mélanges sont préparés par une combinaison des deux modes précédents.

Pour le premier type, le mélange est fait juste en amont des UDPB avec les pipelines. Nous remarquons que juste en amont des UDPB, il y a un système, appelé *manifold*, qui mélange instantanément les différents types de pétrole brut déchargés par les réservoirs, quelques minutes avant leur distillation. Dans ce cas, un seul type de pétrole brut est stocké dans chaque réservoir à la fois. La deuxième possibilité est de préparer les mélanges, demandés par les UDPB, dans les réservoirs. Dans ce cas, une quantité d'un type donné de pétrole brut, déjà chargée dans un réservoir, est réservée en attente d'une autre quantité d'un autre type de pétrole brut, afin de produire le mélange demandé. Finalement, un troisième type de préparation de mélange, qui implique plusieurs configurations du système d'une raffinerie, est constitué d'une combinaison des deux types précédents.

On peut distinguer deux options de distillation différentes :

- distillation exacte,
- distillation flexible.

Pour la première option de distillation, le mélange demandé doit satisfaire une composition exacte avant d'être distillé, quelle que soit la façon dont il est préparé. Par exemple, le mélange demandé par une UDPB doit contenir 20% du type A et 80% du type B pour une quantité égale à αm^3 . La deuxième option de distillation est une relaxation de la première. Pour cette option, les composants du mélange demandé doivent satisfaire des bornes supérieures et inférieures. Par exemple, une UDPB demande une quantité αm^3 d'un mélange, où le type A peut constituer au moins 20% de la quantité totale et au plus 50%. Il existe évidemment d'autres bornes à satisfaire pour les autres types de pétrole brut.

Avant de présenter l'objectif à minimiser pour les modèles développés, décrivons les règles générales de fonctionnement à suivre dans une raffinerie :

- Règle 1 : les limitations de capacité des équipements (capacité des réservoirs, taux ou flux de pompage),
- Règle 2 : si un réservoir donné est en train d'alimenter une UDPB, celui-ci ne peut être chargé en pétrole brut, et vice versa,
- Règle 3 : respect des limitations imposées par le type de préparation des mélanges choisis,
- Règle 4 : respect des limitations des mélanges demandés, imposées par l'option de distillation,
- Règle 5 : respect des contraintes de satisfaction de la demande par les UDPB.

L'objectif de notre modèle, indiqué précédemment, est la minimisation du coût de reconfiguration des réservoirs. La reconfiguration des réservoirs, pour charger les quantités arrivées

aux ports et pour décharger les mélanges de pétrole brut et/ou un seul type de pétrole brut vers les UDPB, impose une série d'opérations long et coûteuses pour la raffinerie. Les opérations les plus critiques et les plus coûteuses sont les suivantes :

Avant le chargement/déchargement :

- la configuration du réseau de pipelines pour l'ouverture de vannes,
- le remplissage des pipelines avec du pétrole brut,
- l'échantillonnage du pétrole brut (pour des analyses chimiques),
- la mesure du pétrole brut qui existe dans le réservoir avant le chargement/déchargement,
- le démarrage du chargement/déchargement,
- la fin du chargement/déchargement.

Après le chargement/déchargement :

- la configuration du réseau de pipelines pour la fermeture de vannes,
- la purge des pipelines,
- la mesure du pétrole brut chargé/déchargé dans les réservoirs.

Cela implique alors d'ordonnancer les activités de chargement/déchargement de façon à ce que le nombre de reconfigurations devienne minimal.

En effet, dans une raffinerie, il existe plusieurs types de réservoirs. En général, les plus grands sont utilisés pour le stockage du pétrole brut. De plus, les problèmes de stockage dans les raffineries apparaissent de plus en plus souvent et l'utilisation multiple des réservoirs devient nécessaire. En conséquence, la diminution du nombre de reconfigurations des réservoirs devient critique car elle correspond à la minimisation du nombre de réservoirs utilisés ce qui implique une augmentation du nombre de réservoirs disponibles pour une autre utilisation (ex. stockage des produits finis). En minimisant ce nombre, nous pouvons obtenir aussi les informations suivantes :

- la réalisabilité d'une solution avec un nombre de réservoirs plus petit,
- la perte due à cette diminution en fonction du coût de reconfiguration,
- le profit créé par la décision d'utiliser quelques réservoirs pour d'autres opérations,
- la valeur du coût de reconfiguration supplémentaire à payer.

Toutes ces informations nous permettent de mieux ordonnancer toutes les activités qui se déroulent dans une raffinerie et de gérer les situations extrêmes. Avant de présenter notre modèle générique et afin d'être plus clair, nous donnons ensuite un exemple numérique réel d'une raffinerie qui se trouve en Grèce, que nous utilisons dans tout le reste de cette thèse.

4.5 Exemple numérique de référence

L'exemple de référence qui est utilisé dans la suite pour illustrer les méthodes proposés correspond à un système de raffinerie situé en Grèce. Ce système est constitué d'un port, de six réservoirs et de deux unités de distillation. Dans les six réservoirs, le pétrole brut est déchargé vers les deux unités de distillation et est chargé par le port.

La configuration de notre système respecte les règles générales présentées dans la section 4.4 ainsi que les règles suivantes :

- le temps de déchargement d'un bateau est égal à 36 heures,
- les débits de distillation pour les deux UDPB sont :
 - UDPB 0 = 270-350 m^3 /heure,
 - UDPB 1 = 300-1660 m^3 /heure.
- les mélanges sont préparés à l'aide des pipelines juste en amont des UDPB.

Les données de notre système sont :

- la période de planification est égale à 30 jours (720 heures),
- le plan d'arrivée des trois bateaux (quand, quel type de pétrole brut, quelle quantité de pétrole brut),
- une douzaine de mélanges demandés (constitués au maximum par trois types de pétrole brut) par les deux unités de distillation (quand, quel type de mélange, quelle quantité),
- l'état de chacun des réservoirs en début de période et leurs capacité de stockage égale à 100000 m^3 .

Dans la figure 4.3, un plan détaillé est présenté pour l'horizon de temps choisi (30 jours).

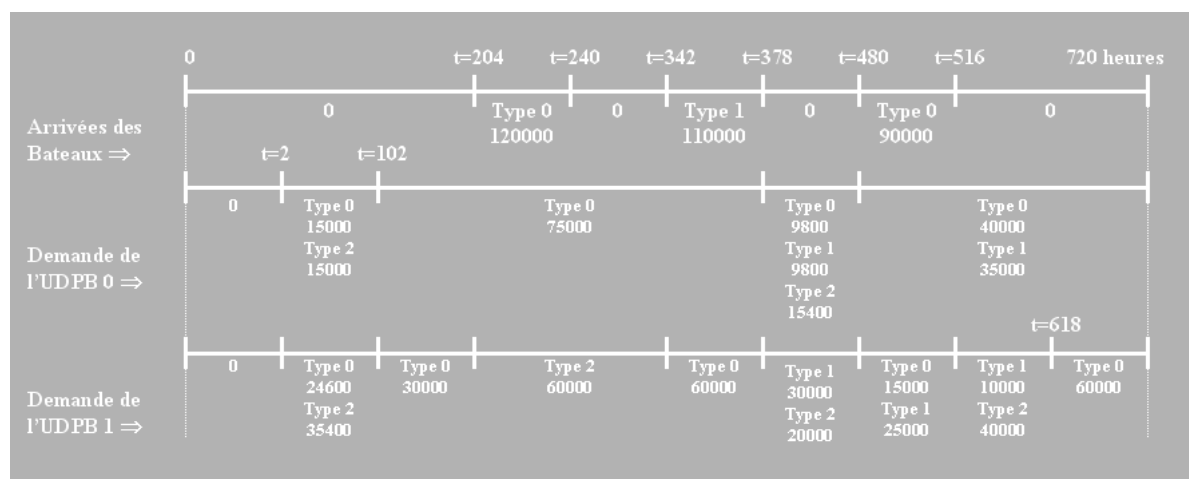


Fig. 4.3 – Le plan détaillé

A l'instant $t=204$ heures est planifiée l'arrivée du premier bateau et son départ est prévu pour l'instant $t=240$ heures, c'est-à-dire après 36 heures. Ce bateau apporte une quantité égale à 120000 m^3 du pétrole brut de type 0. A noter que, d'après le plan de la figure 4.3, quatre mélanges sont demandés par l'UDPB 0 et huit par l'UDPB 1. Les mélanges demandés sont constitués par 2 ou 3 types de pétrole brut, par exemple, entre les instants $t=378$ et $t=480$, l'UDPB 0 demande un mélange de 35000 m^3 dont la composition est : 28% du type 0, 28% du type 1 et 44% du type 2.

L'objectif est de minimiser le coût de reconfiguration des six réservoirs qui sont chargés par les trois bateaux, qui arrivent à des heures précises, tout en préparant les douze mélanges demandés et les déchargeant vers les UDPB aux moments prévus.

Chapitre 5

Modélisation proposée

Généralement, l'optimisation n'est pas utilisée dans l'ordonnancement des raffineries Grecques et aucun modèle mathématique n'est utilisé. L'optimisation se fait grâce à l'expérience des cadres techniciens par calculs manuels et sur l'optimisation locale de chaque période prise en compte individuellement en satisfaisant toutes les contraintes et en minimisant le nombre de réservoirs utilisés. Ce type d'optimisation nous permet d'avoir un plan d'ordonnancement des chargements et des déchargements réalisable mais non optimal. Les résultats numériques montrent que les solutions obtenues de cette façon sont généralement assez éloignées de l'optimum, typiquement de 15-20%. Le besoin de développer un modèle d'optimisation globale pour l'horizon de temps entier apparaît donc clairement.

Dans le paragraphe qui suit, nous donnons notre modélisation pour le problème d'ordonnancement des chargements et des déchargements pour tous les types de préparation des mélanges et pour toutes les options de distillation. Ensuite, nous précisons pour chaque cas toutes les données, toutes les variables et toutes les contraintes supplémentaires qui correspondent à chaque type de préparation de mélange et à chaque option de distillation. Notons que ce modèle est appelé modèle optimal (**MO**). Nous allons utiliser la notation suivante $MOij$, avec $i \in \{1, 2, 3\}$ et $j \in \{1, 2\}$. L'indice i désigne soit la préparation dans les pipelines ($i = 1$), soit dans les réservoirs ($i = 2$), soit la combinaison des deux modes ($i = 3$). L'indice $j = 1$ (respectivement 2) correspond à l'option de distillation exacte (respectivement distillation flexible).

5.1 Modèle de base

Dans cette section, nous précisons les données du système étudié, les variables de décision et les contraintes utilisées dans tous les cas de préparation de mélanges et dans toutes les options de distillation. Généralement, les données de notre système sont les quantités de mélanges arrivées aux ports et les quantités demandées par les UDPB (Unités de Distillation de Pétrole Brut) ainsi que la nature et les quantités de pétrole brut présentés dans les réservoirs en début

de période. Les gestionnaires connaissent a priori le nombre de bateaux qui vont arriver, les dates de leur arrivée, les quantités et les types de pétrole brut transportés. De plus, pour chaque option de distillation choisie, sont connues, soit la composition exacte des mélanges demandés par les UDPB (distillation exacte), soit les bornes supérieures et inférieures que leurs composants doivent satisfaire (distillation flexible).

Pour les données du problème nous utilisons les notations suivantes :

- $E_{i,t}$: la quantité globale disponible, au port i , pendant la période t (entrée),
- $ae_{i,j,t}$: le pourcentage du pétrole de type j par rapport à la quantité globale disponible, au port i , pendant la période t ($E_{i,t}$),
- $S_{k,t}$: la demande globale de l'UDPB k , pendant la période t (sortie),
- le mélange demandé par l'UDPB qui dépend de l'option de distillation adoptée :
 - * Données relative à la distillation exacte :
 - $\Rightarrow as_{k,j,t}$: le pourcentage du type j , demandé par l'UDPB k , pendant la période t .
 - * Données relative à la distillation flexible :
 - $\Rightarrow a_{k,j,t}$: est égal à 1 si l'UDPB k demande le pétrole brut de type j , pendant la période t et égal à zéro sinon,
 - $\Rightarrow amin_{k,j,t}$: le pourcentage minimal acceptable, de type j , contenu dans le mélange déchargé vers l'UDPB k , pendant la période t ,
 - $\Rightarrow amax_{k,j,t}$: le pourcentage maximal acceptable, de type j , contenu dans le mélange déchargé vers l'UDPB k , pendant la période t .
- NR : le nombre de réservoirs,
- NP : le nombre de ports,
- $NUDPB$: le nombre d'unités de distillation du pétrole brut.

Nous considérons deux types de variables de décisions. D'une part, nous avons des variables continues qui mesurent les flux entre les ports et les réservoirs, les flux entre les réservoirs et les UDPB et les quantités stockées des différents types de pétrole brut. D'autre part, nous avons des variables de décision binaires qui représentent les prises ou non de certaines décisions. Ces dernières peuvent être le chargement, le déchargement et la reconfiguration d'un réservoir (qui est nécessaire soit avant le chargement de la quantité disponible au port, soit avant le déchargement vers les UDPB).

Les variables de décision utilisées sont les suivantes :

- $X_{i,z,j,t}$: une variable continue qui correspond à la quantité du pétrole brut du type j , chargée par le port i , dans le réservoir z , pendant la période t (flux port $\Rightarrow$ réservoir),

- $Y_{z,k,j,t}$: une variable continue qui correspond à la quantité du pétrole brut du type j , déchargée par le réservoir z , à l'UDPB k , pendant la période t (flux réservoir $\Rightarrow$ UDPB),
- $C_{i,z,t}$: une variable binaire (0-1) qui est égale à 1 si la connexion est établie entre le port i et le réservoir z , pendant la période t et égale à zéro sinon,
- $D_{z,k,t}$: une variable binaire (0-1) qui est égale à 1 si la connexion est établie entre le réservoir z et l' UDPB k , pendant la période t et égale à zéro sinon,
- $SC_{i,z,t}$: une variable binaire (0-1) qui est égale à 1 si on établit la connexion entre le port i et le réservoir z , au début de la période t et égale à zéro sinon,
- $SD_{z,k,t}$: une variable binaire (0-1) qui est égale à 1 si on établit la connexion entre le réservoir z et l' UDPB k , au début de la période t et égale à zéro sinon,

Les variables d'état sont les suivantes :

- $I_{z,j,t}$: une variable continue qui correspond à la quantité du pétrole brut type j , qui existe, à la fin de la période t , dans le réservoir z . Nous précisons que les $I_{z,j,0}$ sont des données correspondant aux quantités stockées dans les réservoirs au début de la période d'ordonnancement.

Au niveau des contraintes, nous en avons quatre groupes qui garantissent certaines conditions :

- les contraintes qui garantissent que les quantités chargées dans les réservoirs sont égales aux quantités disponibles aux ports,
- les contraintes exprimant que les quantités déchargées vers les UDPB doivent être égales aux quantités demandées par ces dernières,
- les contraintes d'équilibre exprimant que les quantités qui existent dans un réservoir sont égales aux quantités qui existaient à la période précédente plus les quantités chargées moins les quantités déchargées
- enfin, nous avons des contraintes qui garantissent que la quantité stockée dans un réservoir ne dépasse pas la capacité de stockage de ce dernier.

Donnons l'expression détaillée de chacun de ces types de contraintes :

Les contraintes de chargement :

- la somme des quantités chargées dans tous les réservoirs, par le port i , pendant la période t , est égale à la quantité disponible au port i :
$$\sum_{z=1}^{z_{max}} X_{i,z,j,t} = E_{i,t} \times ae_{i,j,t} \quad \forall t, i, j.$$

Les contraintes de déchargement :

- la somme des quantités déchargées par tous les réservoirs, vers l'UDPB k , pendant la période t , est égale à la quantité demandée par l'UDPB k :
$$\sum_{z=1}^{z_{max}} \sum_{j=1}^{j_{max}} Y_{z,k,j,t} = S_{k,t} \quad \forall k, t.$$

Les équations d'équilibre sont les suivantes :

- La quantité du pétrole brut, de type j , qui existe dans le réservoir z , à la période t , est égale à la quantité qui existait à la période $t - 1$, plus la somme des quantités du même type qui sont chargées pendant cette période par tous les ports i , moins la somme des quantités déchargées par ce réservoir vers toutes les UDPB :

$$I_{z,j,t} = I_{z,j,t-1} + \sum_{i=1}^{i_{max}} X_{i,z,j,t} - \sum_{k=1}^{k_{max}} Y_{z,k,j,t} \quad \forall z, j, t.$$

La somme des quantités des différents types de pétrole brut ne dépasse pas la capacité de stockage :

- $0 \leq \sum_{j=1}^{j_{max}} I_{z,j,t} \leq \text{Capacité de stockage} \quad \forall z, t.$

Une autre série de contraintes dites de fonctionnement expriment le fait qu'il doit y avoir une connexion s'il y a un flux entre un réservoir z et un port i ou une UDPB k :

- chargement : $\sum_{j=1}^{j_{max}} X_{i,z,j,t} \leq M \times C_{i,z,t}, \sum_{j=1}^{j_{max}} X_{i,z,j,t} \geq C_{i,z,t} \quad \forall i, z, t,$
- déchargement : $\sum_{j=1}^{j_{max}} Y_{z,k,j,t} \leq M \times D_{z,k,t}, \sum_{j=1}^{j_{max}} Y_{z,k,j,t} \geq D_{z,k,t} \quad \forall z, k, t.$

(Notons que ci-dessus la grande constante M peut prendre la valeur de la capacité de stockage des réservoirs).

Un réservoir ne peut pas être déchargé et être chargé simultanément :

- chargement/déchargement : $\sum_{i=1}^{i_{max}} C_{i,j,t} + D_{z,k,t} \leq 1 \quad \forall z, k, t.$

Les contraintes de configuration : Au début de chaque chargement ou de chaque déchargement, on est obligé de reconfigurer le réservoir z si pendant la période précédente la connexion n'était pas établie. Dans ce cas si à la période t la connexion est établie, un coût de reconfiguration est payé pendant la période t :

- chargement : $C_{i,z,t-1} + SC_{i,z,t} \geq C_{i,z,t} \quad \forall i, z, t,$
- déchargement : $D_{z,k,t-1} + SD_{z,k,t} \geq D_{z,k,t} \quad \forall z, k, t.$

La fonction objectif : minimiser le coût total de reconfiguration des réservoirs nécessaires pour les chargements et les déchargements :

$$\bullet \text{ MinZ} = \sum_{i=1}^{i_{max}} \sum_{z=1}^{z_{max}} \sum_{t=1}^{t_{max}} SC_{i,z,t} + \sum_{z=1}^{z_{max}} \sum_{k=1}^{k_{max}} \sum_{t=1}^{t_{max}} SD_{z,k,t}.$$

Toutes les variables de décision et toutes les contraintes précédemment présentées sont utilisées pour tous les modèles $MO_{i,j}$ c'est à dire pour tous les types de préparation de mélange

et pour toutes les options de distillation. Dans les sections qui suivent, nous présentons, pour chaque cas de préparation de mélange et pour chaque option de distillation, les données, les variables de décision et les contraintes supplémentaires par rapport au modèle de base.

5.1.1 Préparation des mélanges à l'aide des pipelines

Les variables de décision et les contraintes supplémentaires dans le cas de la préparation des mélanges à l'aide des pipelines sont les suivantes :

Les variables de décision supplémentaires :

- $f_{z,j,t}$: une variable binaire (0-1) qui est égale à 1 si le réservoir z contient le type j , pendant la période t et égale à zéro sinon.

Les contraintes du mélange : nous ne pouvons pas avoir de mélange dans les réservoirs. Un seul type j de pétrole brut peut exister dans un réservoir z , pendant la période t :

- $$\sum_{j=1}^{j_{max}} f_{z,j,t} \leq 1 \quad \forall z, t.$$

Les contraintes suivantes assurent la cohérence entre les variables de décision $I_{z,j,t}$ et $f_{z,j,t}$:

- $I_{z,j,t} \leq M \times f_{z,j,t} \quad \forall z, j, t,$
- $I_{z,j,t} \geq f_{z,j,t} \quad \forall z, j, t$

Préparation du mélange à l'aide des pipelines-Distillation exacte/MO11

Si l'option de distillation choisie est la distillation exacte, les contraintes supplémentaires sont les suivantes : la somme des quantités déchargées par tous les réservoirs du type j , pendant la période t , vers l'UDPB k est égale à la quantité demandée de ce type de pétrole brut par cette UDPB :

- $$\sum_{z=1}^{z_{max}} Y_{z,k,j,t} = S_{k,t} \times as_{k,j,t} \quad \forall k, j, t.$$

Préparation du mélange à l'aide des pipelines-Distillation flexible/MO12

Si l'option de distillation choisie est la distillation flexible, les contraintes supplémentaires sont les suivantes : la somme des quantités déchargées par tous les réservoirs du type j , pendant la période t , vers l'UDPB k doit satisfaire les bornes supérieures et inférieures de la quantité demandée de ce type par cette UDPB :

- $$amin_{k,j,t} \times a_{k,j,t} \times S_{k,t} \leq \sum_{z=1}^{z_{max}} Y_{z,k,j,t} \leq amax_{k,j,t} \times a_{k,j,t} \times S_{k,t} \quad \forall k, j, t.$$

5.1.2 Préparation des mélanges dans les réservoirs

Dans le cas de la préparation des mélanges dans les réservoirs, nous avons introduit un nouveau terme : le mélange acceptable. Ce terme est introduit pour linéariser des contraintes qui sont au début non linéaires. Un mélange est dit acceptable si et seulement si il satisfait les contraintes, résultantes de l'option de distillation sur les pourcentages des composants de chaque mélange demandé. Pour qu'un réservoir puisse décharger vers une UDPB, le mélange qui existe dedans doit être un mélange acceptable par l'UDPB. Cette situation est représentée par les deux groupes des contraintes présentées ci-dessous.

Préparation du mélange dans les réservoirs-Distillation exacte/MO21

Si l'option de distillation choisie est la distillation exacte, les contraintes supplémentaires sont les suivantes : pour qu'un réservoir z puisse décharger vers l'UDPB k , à la période t , la quantité du type j doit être égale à la quantité totale qui existe dans le réservoir multipliée par le taux $as_{k,j,t}$ demandé par l'UDPB k . C'est-à-dire le composant j doit être dans le réservoir z à un pourcentage égale à $as_{k,j,t}$. Cela implique les deux contraintes suivantes :

$$\begin{aligned} \bullet & [I_{z,j,t} - \sum_{j'=1}^{j_{max}} I_{z,j',t} \times as_{k,j,t}] \leq M \times [1 - D_{z,k,t}] \quad \forall k, z, j, t, \\ \bullet & [I_{z,j,t} - \sum_{j'=1}^{j_{max}} I_{z,j',t} \times as_{k,j,t}] \geq -M \times [1 - D_{z,k,t}] \quad \forall k, z, j, t. \end{aligned}$$

Préparation du mélange dans les réservoirs-Distillation flexible/MO22

Si l'option de distillation choisie est la distillation flexible, les contraintes supplémentaires sont les suivantes : pour qu'un réservoir z puisse décharger vers l'UDPB k , à la période t , la quantité du type j doit satisfaire les bornes supérieures et inférieures du mélange demandé par l'UDPB k . C'est-à-dire le composant j doit satisfaire les bornes $amin_{k,j,t} \times as_{k,j,t}$ et $amax_{k,j,t} \times as_{k,j,t}$. Cela implique les contraintes suivantes :

$$\begin{aligned} \bullet & [I_{z,j,t} - \sum_{j'=1}^{j_{max}} I_{z,j',t} \times amax_{k,j,t} \times as_{k,j,t}] \leq M \times [1 - D_{z,k,t}] \quad \forall k, z, j, t, \\ \bullet & [I_{z,j,t} - \sum_{j'=1}^{j_{max}} I_{z,j',t} \times amin_{k,j,t} \times as_{k,j,t}] \geq -M \times [1 - D_{z,k,t}] \quad \forall k, z, j, t. \end{aligned}$$

5.1.3 Préparation des mélanges à l'aide des pipelines et dans les réservoirs

Les variables de décision et les contraintes supplémentaires dans le cas de la préparation des mélanges dans les pipelines et dans les réservoirs sont les suivantes :

Variable de décision supplémentaire :

- $\gamma_{z,k,t}$: le pourcentage de la quantité stockée dans un réservoir z et qui va être déchargée vers l'UDPB k , pendant la période t .

Contraintes supplémentaires : la quantité déchargée par un réservoir z , pendant la période t , du type j vers l'UDPB doit satisfaire les contraintes non linéaires du mélange produit :

- $Y_{z,k,j,t} = I_{z,j,t} \times \gamma_{z,k,t} \quad \forall z, k, t$,
- $\sum_{k'=1}^{k_{max}} \gamma_{z,k',t} \leq 1 \quad \forall z, t$.

La raison pour laquelle le MO31 et MO32 sont des modèles non-linéaires est qu'ils présentent deux degrés de liberté au niveau de la décision. Le premier degré est relatif aux réservoirs où on peut prendre la décision de préparer des mélanges. Le deuxième degré est relatif à la partie qui se trouve juste en amont des UDPB où on peut prendre à nouveau la décision de préparer un nouveau mélange à l'aide des pipelines.

Préparation de mélange à l'aide des pipelines et aussi dans les réservoirs-Distillation exacte/MO31

Si l'option de distillation choisie est la distillation exacte, les contraintes supplémentaires sont les suivantes : la somme des quantités déchargées par tous les réservoirs du type j pendant la période t vers l'UDPB k est égale à la quantité demandée de ce type par cette UDPB :

- $\sum_{z=1}^{z_{max}} Y_{z,k,j,t} = S_{k,t} \times as_{k,j,t} \quad \forall k, j, t$.

Préparation de mélange à l'aide des pipelines et aussi dans les réservoirs-Distillation flexible/MO32

Si l'option de distillation choisie est la distillation flexible, les contraintes supplémentaires sont les suivantes : la somme des quantités déchargées par tous les réservoirs du type j pendant la période t vers l'UDPB k doit satisfaire les bornes supérieures et inférieures de la quantité demandée de ce type par cette UDPB :

- $amin_{k,j,t} \times a_{k,j,t} \times S_{k,t} \leq \sum_{z=1}^{z_{max}} Y_{z,k,j,t} \leq amax_{k,j,t} \times a_{k,j,t} \times S_{k,t} \quad \forall k, t$.

5.2 Résultats numériques comparatifs

Rappelons, qu'en pratique, l'optimisation se fait en se fondant sur l'expérience des praticiens par calculs manuels et optimisation locale, de chaque période étant prise en compte individuellement vis à vis du respect de toutes les contraintes et en minimisant le nombre des réservoirs utilisés. Ce type d'optimisation donne pour l'exemple présenté à la section 4.5 (cf. figure 4.3, exemple de référence) une solution sous-optimale qui est égale à 25 reconfigurations pour char-

ger les quantités du pétrole brut apportées par les trois bateaux et pour satisfaire la demande de douze mélanges par les deux UDPB. On a obtenu ce résultat en implémentant la façon de faire des décideurs de la raffinerie. Appelons **A** cette première Méthode de résolution (MA11, la même notation présentée à l'introduction de ce chapitre est utilisée). Si nous utilisons le modèle **MO11**, la solution obtenue est égale à 21 reconfigurations de 6 réservoirs pour charger les mêmes bateaux et pour satisfaire la même demande par les deux UDPB.

En résumant, nous obtenons les résultats suivants :

- Méthode A (**MA11**) : solution sous-optimale égale à 25 reconfigurations de 6 réservoirs
- temps de résolution 10 secondes,
- Méthode exacte, **MO11** : solution optimale égale à 21 reconfigurations de 6 réservoirs
- temps de résolution 16 heures.

Pour l'exemple de référence, le gain par une optimisation globale par rapport à une optimisation heuristique, fondée sur l'expérience des praticiens et sur les calculs manuels est égal à 16%. En constatant que le gain est important, nous avons essayé de trouver une méthodologie pour diminuer le temps de résolution du modèle optimal, en essayant de garder l'optimalité de la solution. Dans les sections suivantes, nous présentons notre méthodologie pour l'amélioration de notre modèle optimal.

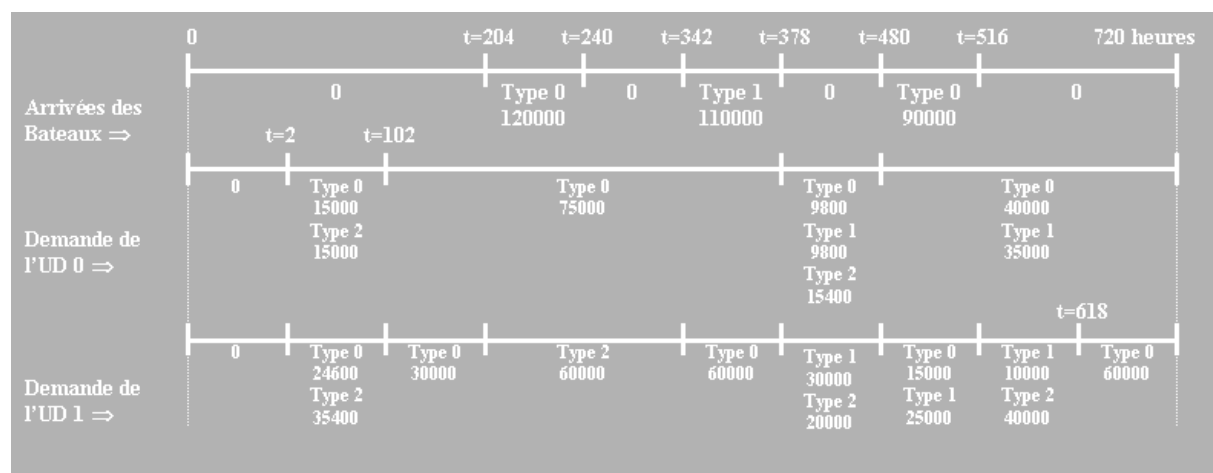
Les modèles ont été développés en utilisant le langage C++ et en faisant appel aux logiciels commerciaux CPLEX 8.1. Les expérimentations présentées ont été réalisées sur un ordinateur Pentium (R) 4, CPU 2,40 GHz et RAM 1 GB, sous Windows 2000 professionnel. Rappelons que l'exemple numérique présenté à la section 4.5 sera notre exemple de référence. Cet exemple réel correspond à une raffinerie en Grèce où les mélanges sont préparés à l'aide des pipelines et l'option de distillation choisie est la distillation exacte. Dans le cadre de cette thèse, nous avons étudié numériquement seulement les cas linéaires (MO11, MO12, MO21, MO22). Pour les deux autres cas (MO31, MO32), nous donnons seulement une modélisation qui nous paraît pertinente mais qui reste non linéaire. De plus, c'est pour le modèle MO11 que nous présentons le plus de résultats numériques car il correspond à la raffinerie de référence.

Un modèle à temps discret nécessiterait un nombre d'intervalles de temps assez important, augmentant parfois excessivement la complexité de calcul du problème. Considérant l'exemple de référence de la section 4.5, un horizon d'un mois implique un nombre de périodes égal à 720. Pour pallier cet inconvénient que présente notre modélisation, nous avons suivi une nouvelle méthodologie. Notons qu'une grande précision pour une formulation en temps discret impose parfois, un pas de temps suffisamment petit. En effet, certaines solutions faisables peuvent être infaisables quand le pas de temps n'est pas adapté. Par ailleurs, un pas de temps petit implique

un grand nombre de pas de temps pour représenter le modèle sur l'horizon considéré. Cela a pour conséquence la difficulté de la mise en oeuvre du modèle vu sa taille excessive dans certains cas, et cela est accentué lorsque l'horizon temporel devient long. Dès lors, le choix de la taille du pas de temps joue un rôle important et les décideurs doivent trouver l'équilibre entre un pas suffisamment petit et un nombre de variables de décision raisonnable. Pour que notre modèle puisse être efficace, il a fallu trouver une procédure pour diminuer le nombre de variables et de contraintes et restreindre son espace des solutions. Pour ce faire, nous introduisons le principe de partition de l'horizon de temps sur événement et nous appliquons la méthode de décomposition de Benders. Ensuite, nous considérons également l'impact de l'ajout d'une série d'inégalités valides et finalement, nous donnons une mise en oeuvre efficace de la méthode de Benders avec génération multiple des coupes valides.

5.3 Décomposition de l'horizon temporel du plan de production par partition sur événement

Dans notre modélisation, au lieu de faire une partition de l'horizon de temps en périodes d'une heure, nous avons fait une partition sur événement : à chaque événement correspond une période. Les événements peuvent être les arrivées des bateaux ou le changement de la constitution du mélange demandé par une UDPB. En prenant l'exemple de référence présenté par la suite (et aussi présenté au chapitre 4.5) et en faisant une partition sur événement, le plan de



Exemple de référence

production prend la forme présentée dans la figure 5.1. Pour cet exemple, le nombre de périodes devient égal à 10 au lieu de 720.

En faisant la partition sur événement, toutes les périodes (d'une heure), pendant lesquelles aucun nouvel événement ne se produit, sont regroupées en une seule période. Pour l'exemple

de référence, entre l'instant t égal à 2 et l'instant t égal à 102, les mêmes événements se produisent : la demande d'un certain mélange par l'UDPB 0 (mélange du type 0 et du type 2) et la demande d'un certain mélange par l'UDPB 1 (mélange du type 0 et du type 2). Cela implique le regroupement de toutes les périodes situées entre ces deux instants en une seule période pendant laquelle la composition des mélanges demandés par les UDPB reste constante. Ensuite, un autre événement se produira entre l'instant t égal à 102 et l'instant t égal à 204 : La composition des deux mélanges demandés par les UDPB change. Cela implique aussi le regroupement de toutes les périodes correspondantes en une seule période. Puis, entre l'instant t égal à 204 et t égal à 240, et même si la composition des mélanges demandés par les UDPB reste constante, l'arrivée du premier bateau aura lieu, ce qui amène à considérer l'existence d'une troisième période. Cette procédure est utilisée aussi pour la définition des autres périodes.

	0	2	102	204	240	342	378	480	516	618	720	
	0	1	2	3	4	5	6	7	8	9		10 périodes
Arrivées des Bateaux $\Rightarrow$	0	0	0	Type 0 120000	0	Type 1 110000	0	Type 0 90000	0	0		
Demande de l'UDPB 0 $\Rightarrow$	0	Type 0 15000 Type 2 15000	Type 0 15000	Type 0 20000	Type 0 10000	Type 0 30000	Type 0 9800 Type 1 9800 Type 2 15400	Type 0 10000 Type 1 10000	Type 0 15000 Type 1 10000	Type 0 15000	Type 0 15000	
Demande de l'UDPB 1 $\Rightarrow$	0	Type 0 24600 Type 2 35400	Type 0 30000	Type 2 40000	Type 2 20000	Type 0 60000		Type 0 15000 Type 1 25000	Type 1 10000 Type 2 40000	Type 0 60000		

Fig. 5.1 – Partition sur événement

Cette grande diminution du nombre de périodes implique une grande diminution du nombre de variables de décision et des contraintes sans affecter la précision désirable. Dans le tableau suivant, (tabl.5.1) et pour l'exemple de référence nous faisons la comparaison entre les deux types de partition de l'horizon du temps pour le modèle optimal (MO11) (cf. section 5.1.1) qui correspond au cas où le mélange est préparé à l'aide des pipelines juste en amont des UDPB et l'option de distillation choisie est la distillation exacte. Après la partition de l'horizon de temps sur événement, nous constatons une grande diminution du temps de résolution. Pour l'exemple de référence, le temps de résolution avec une partition en périodes d'une heure est égal à 16

heures. Après la partition sur événement et grâce à la diminution du nombre de variables et de contraintes, nous obtenons la solution optimale après 1 heure et 4 minutes.

MO11	Partition sur heure	Partition sur événement
Nombre de périodes	720	10
Nombre de variables de décision	112320	1560
Nombre de contraintes	116640	1620
Temps de résolution	16 heures	1 heure 4 minutes
Solution optimale	21	21

TAB. 5.1 – Comparaison entre partition sur heure et partition sur événement

Nous remarquons que la précision du modèle obtenu par une partition sur événement est de bonne qualité. Pour les décideurs, il n'est pas nécessaire de savoir exactement à quelle heure tel réservoir va être chargé par un bateau qui se trouve au port. L'information importante est de savoir quels réservoirs vont être chargés pendant les 36 heures que le bateau va passer au port. En plus, l'obtention d'une solution qui ne précise pas exactement l'heure de chargement d'un réservoir donne aux décideurs une flexibilité plus grande permettant la prise d'une décision finale en temps réel, en tenant compte des situations courantes.

Cette répartition du temps peut nous donner des exemples où certains des échantillons obtenus par une répartition sur événement sont assez longs. Pour donner la flexibilité nécessaire à notre modèle pour changer si besoin sa décision pendant ces périodes, nous proposons d'ajouter des périodes supplémentaires sur ces échantillons. Ces périodes supplémentaires ne correspondent pas à un nouvel événement mais correspondent à des instants où le changement de décision peut être effectué. Le nombre de périodes ajoutées dépend de la longueur de la période qui va être répartie. Nous avons vérifié expérimentalement, en mesurant l'impact sur la fonction objectif que la solution obtenue pour notre modèle avec la répartition sur événement était égale à celle obtenue avec la partition sur heure (cf. exemple de référence, tableau 5.1 et annexe B1, tableau 1.6).

Le besoin d'un plan d'ordonnancement des chargements et des déchargement des réservoirs pour plusieurs scénarii est important. Les décideurs doivent pouvoir prendre des décisions en temps réel en réaction à des événements imprévisibles. Un événement de ce type peut être une panne du système de mixage ou la décision de saisir une offre intéressante sur une quantité de pétrole brut qui se trouve dans la région de la raffinerie. Dans ces situations, plusieurs scénarii doivent être testés et le besoin d'un temps de résolution réduit devient important. Le temps idéal doit être de l'ordre de quelques minutes. Pour que l'on puisse rendre notre modélisation plus efficace avec un temps de résolution de l'ordre de quelques minutes, nous avons appliqué la méthode

de décomposition de Benders [Benders, 1962]. Cette méthode présentée au cours de la section suivante est une méthode de décomposition mathématique et non pas structurelle du système étudié, comme celles utilisées par [Shah, 1996], [Jia et al., 2003] et [Jia and Ierapetritou, 2004]. La méthode de décomposition mathématique garantit la validité d'un modèle exact donnant la solution optimale. De plus, l'exploration de cette méthode est intéressante car elle n'a jamais été appliquée jusqu'ici à des problème de gestion de raffineries.

5.4 Décomposition de Benders

La méthode de décomposition de Benders [Benders, 1962] (appelée aussi méthode de décomposition par partitionnement des variables) est une méthode très connue et a fait ses preuves dans de très nombreuses applications industrielles. La taille des applications industrielles est souvent grande et l'utilisation d'une méthode de décomposition paraît être une solution pour réduire les temps de calcul. Dans la suite, nous donnons un rappel mathématique concernant la méthode de décomposition de Benders avant de présenter l'exemple de son application au modèle MO11.

5.4.1 Rappels Mathématiques

La présentation ci-dessous est inspirée de celle donné dans [M.Minoux, 1983]. Un programme linéaire en nombres entiers et continus en toute généralité, est de la forme suivante :

$$P \begin{cases} \text{Min} z = c \times x + f \times y \\ \text{sous les contraintes} : \\ D \times x + F \times y = d \\ x \geq 0, \quad y \in Y \subset R^m. \end{cases}$$

La matrice

$$D = \begin{bmatrix} D_1 & & & 0 \\ & \ddots & & \\ & & \ddots & \\ 0 & & & D_K \end{bmatrix}$$

est bloc-diagonale. Elle induit des partitions en blocs du vecteur $x = (x_1, x_2, \dots, x_K)$, du vecteur $c = (c_1, c_2, \dots, c_K)$, de la matrice F et du second membre d

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_K \end{bmatrix}, \quad d = \begin{bmatrix} d_1 \\ \vdots \\ d_K \end{bmatrix}$$

ainsi que de l'ensemble Y des vecteurs y qui est l'ensemble R^m tout entier. En distinguant les différents blocs de la matrice des contraintes, le programme devient :

$$\left\{ \begin{array}{l} \text{Min} z = \sum_{k=1}^K c_k \times x_k + f \times y \\ \text{sous les contraintes :} \\ D_1 \times x_1 + F_1 \times y = d_1 \\ D_2 \times x_2 + F_2 \times y = d_2 \\ \vdots \\ D_K \times x_K + F_K \times y = d_K \\ x_1, x_2, \dots, x_K \geq 0, \quad y \in Y. \end{array} \right.$$

Dès l'instant où la valeur des variables y est fixée, les équations :

$$D_K \times x_K = d_K - F_K \times y$$

ont toutes une solution $x_k \geq 0$ et la solution du problème restreint aux variables x se décompose en K programmes linéaires indépendants (**P**rogrammes **E**slaves : **PE**). C'est cette observation qui est à l'origine des techniques de décomposition par partitionnement des variables que nous allons explorer par la suite. En gardant à l'esprit que chaque fois que l'on doit résoudre, pour une y fixée, le programme suivant :

$$Q(y) \left\{ \begin{array}{l} \text{Min} z = c \times x \\ \text{sous les contraintes :} \\ D \times x = d - F \times \bar{y} \\ D_2 \times x_2 + F_2 \times y = d_2 \\ x \geq 0. \end{array} \right.$$

(ou son dual), on est conduit à une décomposition en K sous-programmes indépendants. Pour pouvoir appliquer la technique de partitionnement suggérée, on ne peut pas choisir n'importe comment les variables y dans Y . Il faut au moins que le programme $Q(y)$ ait un ensemble de solution non vide. Pour traduire cette condition, nous utilisons le théorème de *Farkas* et *Minkowski*. Associons à chaque contrainte i de $Q(y)$ une variable duale u_i (non contrainte en signe) et notons par u le vecteur-ligne des variables duales (de dimension égale au nombre de contraintes de $Q(y)$). Alors le théorème de *Farkas* et *Minkowski* s'énonce :

$Q(y)$ a une solution, $x \geq 0$ si et seulement si : $u \times (d - F \times y) \leq 0$ pour tout u vérifiant $u \times D \leq 0$.

Comme le cône $\vartheta = \{u / u \times D \leq 0\}$ est polyédrique, il y a un nombre fini de générateurs qui seront notés $u^1, u^2, \dots, u^p$ (tout $u \in \vartheta$ est une combinaison linéaire à coefficients positifs ou nuls des $u^i, i = 1, \dots, p$). La condition nécessaire et suffisante du théorème de *Farkas* et *Minkowski*

est alors équivalente au système d'inégalités suivant :

$$(I) \left\{ \begin{array}{l} u^1 \times (d - F \times y) \leq 0 \\ u^2 \times (d - F \times y) \leq 0 \\ \vdots \\ u^P \times (d - F \times y) \leq 0 \end{array} \right.$$

Généralement, ce système comporte un nombre énorme d'inéquations égal au nombre de générateurs du cône polyédrique ϑ . Si (I) n'a pas de solution en y , cela signifie, par construction, qu'il n'existe pas un $y \in R^m$ tel que $Q(y)$ ait une solution. On conclut que le problème (P) lui-même n'a pas de solution.

Nous supposons dans la suite que (I) a une solution. Soit R l'ensemble de vecteurs $y \in Y$ satisfaisant (I) (nécessairement $R \neq \emptyset$ si (P) a une solution), le problème (P) est alors équivalent à :

$$\min_{y \in R} \{f \times y + \min_x \{c \times x / D \times x = d - F \times y; \ x \geq 0\}\}$$

Ceci correspond bien à l'idée initiale de fixer y , de résoudre un programme linéaire $Q(y)$ (décomposition en K sous-programmes indépendants), de choisir une meilleure valeur de y , etc. Pour $y \in R$ fixé, le dual du problème $Q(y)$ s'écrit, en notant u le vecteur des variables duales associées aux contraintes de $Q(y)$ de la façon suivante :

$$(Q^*(y)) \left\{ \begin{array}{l} \text{Max } u \times (d - F \times y) \\ \text{sous les contraintes :} \\ u \times D \leq c, \ u \text{ de signe quelconque} \end{array} \right.$$

Le polytope des contraintes de $Q^*(y)$, $V = \{u / u \times D \leq c\}$ ne dépend pas de y , et les $(u^1, u^2, \dots, u^q)$ définis plus haut sont ses rayons extrémaux. Si V est vide, alors, d'après le théorème de la dualité on obtient un de ces deux cas suivants :

- $Q(y)$ n'a pas de solution,
- $Q(y)$ est non borné.

Mais par définition, $y \in R$ ce qui implique que $Q(y)$ a une solution. Par suite, si V est vide, alors $Q(y)$ est non borné pour toutes les valeurs de $y \in R$. Donc, dans ce cas, le problème (P) est lui-même non borné. En attribuant la valeur $-\infty$ au maximum de $Q^*(y)$ lorsque $Q^*(y)$ n'a pas de solution, et en utilisant le théorème de la dualité, on peut récrire (P) sous la forme suivante :

$$\min_{y \in R} \{f \times y + \max \{u \times (d - F \times y) / u \times D \leq c\}\}$$

Le maximum de $Q^*(y)$ ne peut pas être non borné $(+\infty)$ pour $y \in R$. Il est donc atteint en un point extrême du polytope : $V = \{u / u \times D \leq c\}$. En supposant V non vide, notons $u^1, u^2, \dots, u^q$

les points extrêmes (en nombre fini) du polytope V . (P) peut alors s'écrire comme suit :

$$\min_{y \in R} \{f \times y + \text{Max}\{u^i \times (d - F \times y)\}\}$$

La dernière forme de (P) apparaît être équivalente au programme linéaire suivant :

$$\left\{ \begin{array}{l} \text{Min } z \\ \text{sous les contraintes :} \\ z \geq f \times y + u^1 \times (d - F \times y) \\ z \geq f \times y + u^2 \times (d - F \times y) \\ \vdots \\ z \geq f \times y + u^q \times (d - F \times y) \\ y \in R. \end{array} \right.$$

En exprimant la condition $y \in R$ par le système d'inéquations (I), on déduit l'équivalence entre (P) et entre le programme linéaire (Programme Maître : PM) suivant :

$$(PM) \left\{ \begin{array}{l} \text{Min } z \\ \text{sous les contraintes :} \\ f \times y + u^1 \times (d - F \times y) - z \leq 0 \\ \dots \\ f \times y + u^q \times (d - F \times y) - z \leq 0 \\ u^1 \times (d - F \times y) - z \leq 0 \\ \dots \\ u^p \times (d - F \times y) - z \leq 0 \\ y \in R. \end{array} \right.$$

Le nombre de contraintes de ce problème, qui est égal au nombre de points extrêmes et de rayons extrémaux de V , est généralement énorme. A la section suivante, nous présentons l'algorithme de Benders où un programme maître restreint converge vers la solution optimale.

5.4.2 Algorithme de Benders

Supposons qu'à une étape quelconque, seules quelques contraintes de PM sont connues explicitement. Il en résulte un **Programme Maître Restreint (PMR)** formé à partir des sous-ensembles $I \subset \{1, 2, \dots, p\}$ et $J \subset \{1, 2, \dots, q\}$ des contraintes du PM :

$$(PMR) \left\{ \begin{array}{l} \text{Min } z \\ \text{sous les contraintes :} \\ f \times y + u^j \times (d - F \times y) - z \leq 0, (\forall j \in J) \\ u^i \times (d - F \times y) - z \leq 0 (\forall i \in I) \\ y \in R. \end{array} \right.$$

Pour $Y = R^m$, PMR est un programme linéaire qui peut être résolu par l'algorithme du simplexe. Soit $(\bar{y}, \bar{z})$ une solution optimale du PMR. Comme PMR est formé à partir de PM par

relaxation d'un certain nombre de contraintes, $\bar{z}$ est un minorant de z^* , valeur optimale de PM et de $P : \bar{z} \leq z^*$.

Une condition nécessaire et suffisante pour que $(\bar{y}, \bar{z})$ constitue une solution optimale du programme PM, donc de P , est que $(\bar{y}, \bar{z})$ satisfasse toutes les contraintes de PM non explicitées dans le PMR. Pour vérifier cette condition, il suffit de résoudre le problème $Q^*(\bar{y})$. Mise à part la situation où $Q^*(\bar{y})$ n'a pas de solution, trois cas peuvent se présenter :

- **Cas 1** : la valeur optimale de $Q^*(\bar{y})$ n'est pas bornée ($: +\infty$). L'algorithme du simplexe appliqué à $Q^*(\bar{y})$ fournit donc un rayon extrême $\bar{u}$ de V tel que :

$$\bar{u} \times (d - F \times \bar{y}) > 0 \quad \text{et} \quad \bar{u} \times D \leq 0.$$

La contrainte : $\bar{u} \times (d - F \times y) \leq 0$ n'est donc pas vérifiée par la solution courante $\bar{y}$ du PMR. $(\bar{y}, \bar{z})$ n'est donc pas une solution du PM. La contrainte $\bar{u} \times (d - F \times y) \leq 0$ doit alors être rajoutée au PMR pour former un nouveau programme restreint augmenté.

- **Cas 2** : L'optimum de $Q^*(\bar{u})$ est de valeur finie (donc atteint en un point extrême $\bar{u}$ de V) et l'on a : $f \times \bar{y} + \bar{u} \times (d - F \times \bar{y}) - \bar{z} \leq 0$. On peut alors écrire : $f \times \bar{y} + u^j \times (d - F \times \bar{y}) - \bar{z} \leq 0$ pour tout $j = 1, \dots, q$ (puisque $\bar{u}$ est un point extrême de V qui minimise $u \times (d - F \times \bar{y})$ sur l'ensemble des points extrêmes). D'autre part, on a : $u^i \times (d - F \times \bar{y}) \leq 0$ pour tous les rayons extrémaux u^i , alors l'optimum de $Q^*(\bar{y})$ serait non borné ($: +\infty$). On en déduit que $(\bar{y}, \bar{z})$ est une solution optimale de PM, donc $\bar{y}$ est une solution optimale de P et l'algorithme se termine.
- **Cas 3** : L'optimum de $Q^*(\bar{y})$ est de valeur finie et atteint en un point extrême $\bar{u}$ de V , mais, contrairement au cas 2, on a : $f \times \bar{y} + \bar{u} \times (d - F \times \bar{y}) - \bar{z} > 0$. Ceci montre que la contrainte $f \times y + \bar{u} \times (d - F \times y) - z \leq 0$ n'est pas vérifiée par la solution courante $(\bar{y}, \bar{z})$ du PMR. La contrainte $f \times y + \bar{u} \times (d - F \times y) - z \leq 0$ doit donc être rajoutée au PMR pour former un nouveau programme restreint augmenté.

Tant que le test d'optimalité n'est pas satisfait (cas 2), on peut ajouter au programme maître restreint (PMR) des contraintes non satisfaites par la solution courante $(\bar{y}, \bar{z})$. Le nouveau PMR est alors résolu en (y, z) et ainsi de suite. Si, à une étape quelconque, le problème restreint n'a pas de solution, alors le problème P n'a pas de solution et l'algorithme s'arrête. Enfin, si V est vide (ce que l'on constate dès la première itération de l'algorithme, lorsqu'on résout $Q^*(y)$ pour la première fois) et si P a une solution, alors on peut conclure que P est non borné, et l'algorithme s'achève.

5.4.3 Décomposition du modèle développé

Dans la suite, nous présentons le programme maître restreint (**PMR**) et les programmes esclaves (**PE**) en considérant le modèle MO11 qui correspond au cas de la préparation des mélanges à l'aide des pipelines juste en amont des UDPB et à une distillation exacte. En étudiant les contraintes de reconfiguration des réservoirs, on peut remarquer que les variables de décision $SC_{i,z,t}$ et $SD_{k,z,t}$, prennent soit une valeur égale à 1, soit une valeur supérieure ou égale à zéro. Notons que l'objectif est de minimiser la somme des variables $SC_{i,z,t}$ et $SD_{k,z,t} \forall i, z, k, t$ et si $SC_{i',z',t'} \geq 0$ pour un i', j' et t' , $SC_{i',z',t'}$ sera égal à zéro (également pour $SD_{k,z,t}$) grâce à la fonction objectif. Cela implique que si l'on relaxe les variables binaires $SC_{i,z,t}$ et $SD_{k,z,t}$ en variables continues $[0, 1]$, notre modèle reste valable (en effet l'intégralité des variables $C_{i,z,t}$ et $D_{k,z,t}$ garantir à l'optimum l'intégralité des variables $SC_{i,z,t}$ et $SD_{k,z,t}$). Après cette relaxation des variables binaires $SC_{i,z,t}$ et $SD_{k,z,t}$, la décomposition du MO11 se représente par les deux modèles suivants. Notons que nous poursuivons la même procédure de décomposition pour les autres cas de préparation des mélanges et d'option de distillation (MO12, MO21, MO22).

Programme Maître Restreint :

Le programme maître restreint (PMR) se compose de toutes les variables binaires du MO11 et de toutes les contraintes du MO11 qui ne contiennent que des variables de décision binaires. Les variables de décision binaires sont les suivantes :

- $C_{i,z,t}$: une variable binaire (0-1) qui est égale à 1 si la connexion est établie entre le port i et le réservoir z , pendant la période t et égale à zéro sinon,
- $D_{z,k,t}$: une variable binaire (0-1) qui est égale à 1 si la connexion est établie entre le réservoir z et l'UDPB k , pendant la période t et égale à zéro sinon,
- $f_{z,j,t}$: une variable binaire (0-1) qui est égale à 1 si le réservoir z contient le type j , pendant la période t et égale à zéro sinon.

Les contraintes initiales du PMR sont :

Un réservoir ne peut pas être déchargé et être chargé simultanément :

- chargement/déchargement : $\sum_{i=1}^{i_{max}} C_{i,j,t} + D_{z,k,t} \leq 1 \forall z, k, t.$

Nous ne pouvons pas avoir de mélange dans les réservoirs. Un seul type j de pétrole brut peut exister dans un réservoir z , pendant la période t :

- $\sum_{j=1}^{j_{max}} f_{z,j,t} \leq 1 \forall z, t$

Le modèle maître restreint n'a pas de variables de décision du MO11 à la fonction objectif, grâce à la relaxation en variables continues des variables $SC_{i,z,t}$ et $SD_{k,z,t}$. Cela implique la fonction objectif suivante :

- $\text{MinZ} = \varphi$

Modèles Esclaves : Les données du programme :

- $E_{i,t}$: la quantité globale disponible, au port i , pendant la période t (entrée),
- $ae_{i,j,t}$: le pourcentage du PB type j disponible, au port i , pendant la période t ,
- $S_{k,t}$: la demande globale de l'UDPB k , pendant la période t (sortie),
- $as_{k,j,t}$: le pourcentage du type j , demandé par l'UDPB k , pendant la période t .

Soit $(\bar{C}_{i,z,t} \ \bar{D}_{z,k,t} \ \bar{f}_{z,j,t})$ la solution optimale du programme maître restreint (PMR) ; le programme esclave aura alors, la forme suivante :

Les variables de décision sont les suivantes :

- $X_{i,z,j,t}$: une variable continue qui correspond à la quantité du pétrole brut du type j , chargée par le port i , dans le réservoir z , pendant la période t (flux port $\Rightarrow$ réservoir),
- $Y_{z,k,j,t}$: une variable continue qui correspond à la quantité du pétrole brut du type j , déchargée par le réservoir z , à l'UDPB k , pendant la période t (flux réservoir $\Rightarrow$ UDPB),
- $SC_{i,z,t}$: une variable binaire relaxée en variable continue $[0,1]$ qui est égale à 1 si on établit la connexion entre le port i et le réservoir z , au début de la période t et égale à zéro sinon,
- $SD_{z,k,t}$: une variable binaire relaxée en variable continue $[0,1]$ qui est égale à 1 si on établit la connexion entre le réservoir z et l'UDPB k , au début de la période t et égale à zéro sinon.

Les variables d'état sont les suivantes :

- $I_{z,j,t}$: une variable continue qui correspond à la quantité du pétrole brut type j , qui existe, à la fin de la période t , dans le réservoir z .

Les contraintes de chargement :

- la somme des quantités chargées dans les réservoirs z , par le port i , pendant la période t , est égale à la quantité disponible au port i :
$$\sum_{z=1}^{z_{max}} X_{i,z,j,t} = E_{i,t} \times ae_{i,j,t} \quad \forall t, i, j$$

Les contraintes de déchargement :

- la somme des quantités déchargées par les réservoirs z , vers l'UDPB k , pendant la période t , est égale à la quantité demandée par l'UDPB k :
$$\sum_{z=1}^{z_{max}} \sum_{j=1}^{j_{max}} Y_{z,k,j,t} = S_{k,t} \quad \forall k, t$$

Les équations d'équilibre :

- La quantité du pétrole brut de type j , qui existe dans le réservoir z pendant la période t est égale à la quantité qui existait à la période $t - 1$ plus la somme des quantités du même type qui sont chargées pendant cette période par tous les ports i , moins la somme des quantités déchargées par ce réservoir vers toutes les UDPB :

$$I_{z,j,t} = I_{z,j,t-1} + \sum_{i=1}^{i_{max}} X_{i,z,j,t} - \sum_{k=1}^{k_{max}} Y_{z,k,j,t} \quad \forall z, j, t$$

La somme des quantités des différents types de pétrole brut ne dépasse pas la capacité de stockage :

- $0 \leq \sum_{j=1}^{j_{max}} I_{z,j,t} \leq \text{Capacité de stockage } \forall z, t$

Distillation Exacte : la somme des quantités déchargées par tous les réservoirs du type j pendant la période t vers l'UDPB k est égale à la quantité demandée de ce type de pétrole brut par cette UDPB :

- $\sum_{z=1}^{z_{max}} Y_{z,k,j,t} = S_{k,t} \times as_{k,j,t} \quad \forall k, j, t.$

En obtenant la solution $(\bar{C}_{i,z,t} \quad \bar{D}_{z,k,t} \quad \bar{f}_{z,j,t})$ par le programme maître restreint (PMR), les autres contraintes des programmes esclaves prennent la forme suivante :

- Les connexions entre les réservoirs et les ports et les réservoirs et les UDPB sont données par le PMR, nous obtenons alors les contraintes suivantes :

- chargement : $\sum_{j=1}^{j_{max}} X_{i,z,j,t} \leq M \times \bar{C}_{i,z,t}, \sum_{j=1}^{j_{max}} X_{i,z,j,t} \geq \bar{C}_{i,z,t} \quad \forall i, z, t,$
- déchargement : $\sum_{j=1}^{j_{max}} Y_{z,k,j,t} \leq M \times \bar{D}_{z,k,t}, \sum_{j=1}^{j_{max}} Y_{z,k,j,t} \geq \bar{D}_{z,k,t} \quad \forall z, k, t,$
- chargement : $\bar{C}_{i,z,t-1} + SC_{i,z,t} \geq \bar{C}_{i,z,t} \quad \forall i, z, t,$
- déchargement : $\bar{D}_{z,k,t-1} + SD_{z,k,t} \geq \bar{D}_{z,k,t} \quad \forall z, k, t.$

- Etant donné le type j du pétrole brut stocké dans les réservoirs, pour toute période t , nous obtenons les contraintes des mélanges suivantes : si $F_{z',j',t'} = 1$, le type j' existe dans un réservoir z' , pendant la période t' :

- $\bar{f}_{z',j',t'} < I_{z',j',t'} \leq M \times \bar{f}_{z',j',t'}$

La fonction objectif est la minimisation du coût total de reconfiguration des réservoirs pour être chargé par les ports ou pour décharger vers les UDPB :

- $\text{MinZ} = \sum_{i=1}^{i_{max}} \sum_{z=1}^{z_{max}} \sum_{t=1}^{t_{max}} SC_{i,z,t} + \sum_{z=1}^{z_{max}} \sum_{k=1}^{k_{max}} \sum_{t=1}^{t_{max}} SD_{z,k,t}.$

Le programme maître restreint (PMR) n'a pas une fonction objectif car la fonction objectif du MO11 ne contient que des variables de décision binaires qui étaient relaxées en variables continues. Donc nous cherchons à chaque itération de notre algorithme une solution réalisable. Dans la figure 5.2, nous présentons l'organigramme de l'algorithme de décomposition.

Après avoir appliqué la méthode de décomposition de Benders classique, nous avons obtenu des résultats décevants. Dans le tableau qui suit, nous présentons les résultats obtenus pour l'exemple de référence :

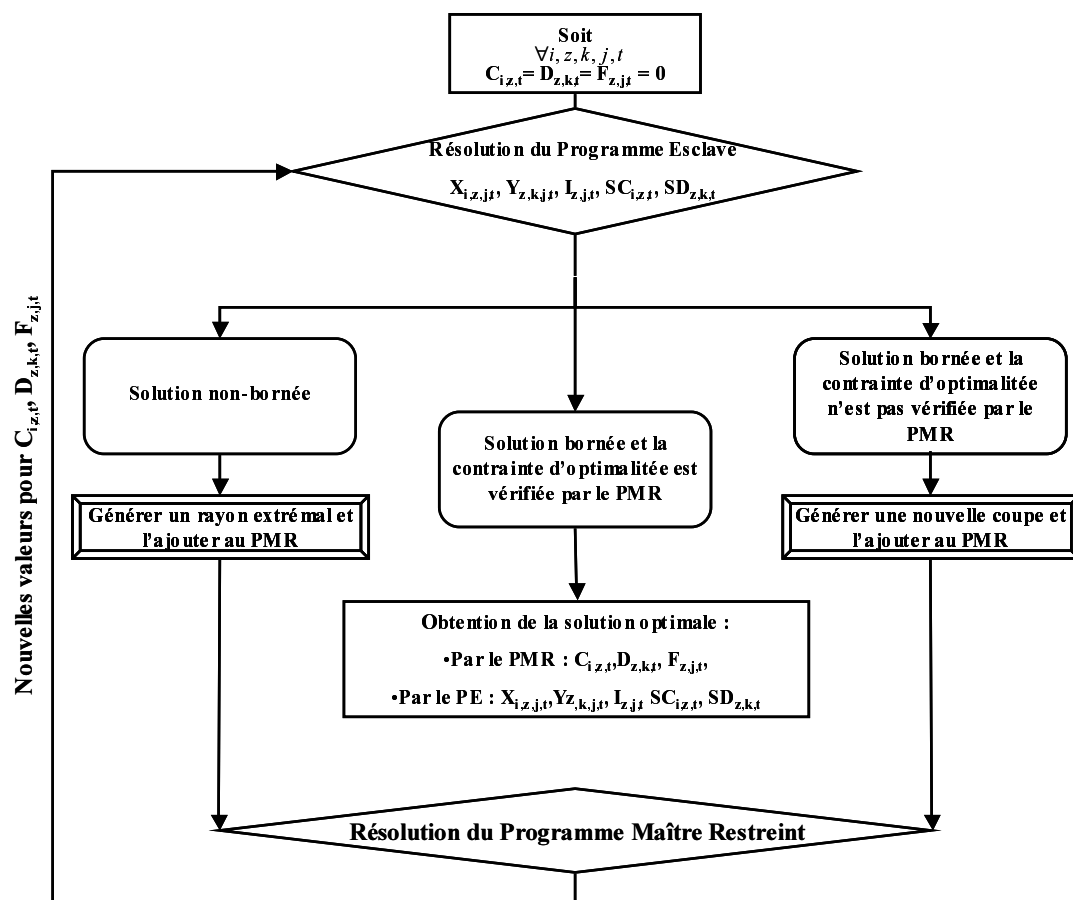


Fig. 5.2 – Organigramme de l'algorithme

Modèle	MO11	MO11 Décomposé (Benders Classique)
Temps de résolution	1 heure et 4 minutes	plus de 24 heures

TAB. 5.2 – Modèle non-décomposé/Modèle décomposé

Après l'obtention de nos résultats numériques, nous avons commencé à étudier les raisons pour lesquelles l'application de la méthode de décomposition de Benders donne un temps de résolution assez important, très supérieur à celui de la résolution directe du MO11. Nous avons identifié deux raisons principales :

- la solution initiale obtenue par le PMR est très loin de la solution optimale du MO11. Pour l'exemple de référence, la solution du MO11 a une valeur égale à 21 et la solution initiale du PMR a une valeur égale à 0,
- les PE deviennent souvent infaisables, à cause des solutions obtenues par le PMR.

La première raison rend nécessaire une meilleure initialisation du PMR et la deuxième raison implique l'élimination systématique des solutions obtenues par PMR, rendant les PE

infaisables. Pour ce faire, nous étudions la possibilité d'initialiser le PMR grâce à l'utilisation d'inégalités valides de différents types. Ces inégalités font d'une côté converger la valeur initiale des PMR vers la solution optimale et éliminent d'une autre côté une grande partie des solutions qui rendent infaisables les PE. Au chapitre suivant, nous présentons toutes les inégalités valides développées pour le cas de préparation des mélanges dans les pipelines juste en amont des UDPB et pour les deux types de distillation (MO11 et MO12). Toutes ces inégalités sont fondées sur les données et les règles de fonctionnement de notre système. Comme nous le verrons certaines de ces inégalités sont également valides pour les deux autres cas de préparation de mélanges.

Chapitre 6

Etude algorithmique et mise en oeuvre efficace

6.1 Inégalités valides pour l'initialisation du programme maître

Afin de montrer les inégalités valides développées dans ce chapitre (pour le cas où les mélanges demandés sont préparés dans les pipelines juste en amont de l'UDPB), nous commençons par analyser quelques situations typiques où le PMR (programme maître restreint) envoie une solution aux PE (programmes esclaves), les rendant infaisables :

- le PMR donne à la variable de décision $C_{i',z',t'}$ une valeur égale à 1 et soit il n'y a aucun bateau au port i' pendant la période t' , soit le bateau qui se trouve au port i' pendant la période t' apporte du pétrole brut de type j' et le réservoir z' contient du pétrole brut de type j'' ,
- le PMR donne à la variable de décision $D_{z',k',t'}$ une valeur égale à 1 alors que le réservoir z' est soit vide pendant la période t' , soit contient du pétrole brut type j' qui n'est pas un des composant du mélange demandé par l'UDPB k' pendant la période t' ,
- la variable de décision $F_{z',i',t'}$ est égale à 1 pendant la période t' alors que le réservoir z' est soit vide, soit contient un autre type de pétrole brut pendant la période t' .

Pour remédier à ces cas d'infaisabilité effectivement rencontrés lors de l'application de la méthode de Benders classique, nous avons développé une série d'inégalités valides qui vont être présentées par la suite et qui les éliminent. Ci-dessous est la définition des variables supplémentaires utilisées dans les inégalités valides générées par les données :

- $W1_{i,j,t}$: une variable binaire (0-1) qui est égale à 1 si un bateau se trouve au port i , apportant du pétrole brut de type j , pendant la période t et égale à zéro sinon,
- $W2_{k,j,t}$: une variable binaire (0-1) qui est égale à 1 si l'UDPB k demande du pétrole brut de type j , pendant la période t et égale à zéro sinon,
- $S2_t$: le nombre maximal de différents types de pétrole brut demandés par les UDPB, pendant la période t ,

- $L_{j,t}$: une variable binaire (0-1) qui est égale à 1 si le pétrole brut de type j est demandé par une UDPB, pendant la période t , et égale à zéro sinon.

6.1.1 Inégalités valides générées à partir des données

En utilisant les données de notre système, nous avons généré trois groupes d'inégalités valides. Le premier groupe correspond aux données relatives aux arrivées des bateaux et aux chargements des réservoirs, le deuxième groupe correspond à celles relatives aux mélanges demandés par les UDPB et aux déchargements des réservoirs et le troisième à la constitution d'un réservoir.

Chargement : Le nombre des réservoirs en cours de chargement est :

- inférieur ou égal au nombre des réservoirs de la raffinerie moins le nombre des différents types demandés par les UDPB ($S2_t$). Pour l'exemple de référence (cf. section 4.5) et pour la période $t=7$, $S2_7 = 2$ et le nombre des réservoirs est égal à 6, ce qui implique que la borne supérieure du nombre des réservoirs, en cours de chargement, est égale à 4,
- supérieur ou égal au nombre minimal des réservoirs pour décharger la quantité du pétrole brut arrivée au port i , pendant la période t . Pour l'exemple de référence (cf. section 4.5) et pour la période $t=7$, la quantité arrivée est égale à $90000 m^3$. Notons que la capacité de stockage des réservoirs est égale à $100000 m^3$, la borne inférieure est égale à 1.

Cela implique la contrainte suivante :

$$\sum_{j=1}^J W1_{i,j,t} \leq \sum_{z=1}^Z C_{i,z,t} \leq (NR - S2_t) \quad \forall i, t \quad (6.1)$$

En plus, si aucun bateau n'arrive pendant la période t , le nombre des réservoirs chargés est égal à zéro :

$$\sum_{z=1}^Z C_{i,z,t} \leq M \times \sum_{j=1}^J W1_{i,j,t} \quad \forall i, t \quad (6.2)$$

Déchargement : Le nombre des réservoirs en cours de déchargement est :

- inférieur ou égal au nombre des réservoirs de la raffinerie moins la somme de tous les bateaux qui sont disponibles dans tous les ports pendant la période t . Pour l'exemple de référence et pour la période $t = 7$, le nombre de réservoirs est égale à 6 et la somme de tous les bateaux qui sont disponibles dans tous les ports est égale à 1, ce qui implique que la borne supérieure du nombre des réservoirs en cours de déchargement est égale à 5,
- supérieur ou égal à la somme de tous les types de pétrole brut demandé par les UDPB.

Cela implique la contrainte suivante :

$$\sum_{j=1}^J W2_{k,j,t} \leq \sum_{z=1}^Z D_{z,k,t} \leq (NR - \sum_{i=1}^I \sum_{j=1}^J W1_{i,j,t}) \quad \forall k, t \quad (6.3)$$

De plus, nous remarquons que les bornes supérieures de ces deux inégalités (6.1, 6.3) ont une forme différente parce qu'un bateau apporte un seul type de pétrole brut mais un réservoir peut décharger vers plusieurs UDPB. Ceci implique le terme $S2_t$ à la l'inégalité 6.1 et le terme

$$\sum_{i=1}^I \sum_{j=1}^J W1_{i,j,t} \text{ à l'inégalité 6.3.}$$

Constitution d'un réservoir : La somme des variable de décision $F_{z,j,t}$ sur tous les réservoirs est :

- inférieure ou égale au nombre des réservoirs moins le nombre maximal des différents types demandés par tous les UDPB pendant la période t , plus 1, car parmi les types demandés par les UDPB, il y en a un qui se trouve éventuellement dans un réservoir z ,
- supérieure ou égale à 1 si le type j est demandé par une UDPB pendant la période t .

Cela implique la contrainte suivante :

$$L_{j,t} \leq \sum_{z=1}^Z F_{z,j,t} \leq (NR - S2_t + 1) \quad \forall j, t. \quad (6.4)$$

Notons que cette inégalité (6.4) est aussi valide pour les deux options de distillation et dans le cas ou les mélanges sont préparés dans les pipelines juste en amont des UDPB et pas aux deux autres types de préparation des mélanges.

6.1.2 Inégalités valides générées à partir des contraintes de fonctionnement

La deuxième catégorie d'inégalités valides est générée à partir des contraintes de fonctionnement. Dans cette catégorie, il y a quatre groupes d'inégalités valides. Les trois premiers tiennent compte des règles de fonctionnement de la raffinerie pour une certaine période t et le quatrième lie la période t à la période $t + 1$.

Chargement pendant la période t : Pour qu'on puisse charger un réservoir z avec du pétrole brut de type j pendant la période t , il faut que le réservoir z soit vide ou qu'il contienne le même type de pétrole brut, pendant cette période. De plus, un réservoir qui contient du pétrole brut de type j' pendant la période t , ne peut pas être chargé par une quantité de pétrole brut de type j'' pendant cette même période. Cette contrainte de fonctionnement implique l'inégalité valide suivante :

$$(1 - W1_{i,j,t}) + (1 - (\sum_{jj=1}^J F_{z,jj,t} - F_{z,j,t})) \geq C_{i,z,t} \quad \forall i, j, t, z \quad (6.5)$$

Notons que cette inégalité (6.5) est valide pour les deux options de distillation et au cas où les mélanges sont préparés dans les pipelines juste en amont des UDPB et pas pour les deux autres types de préparation des mélanges.

Déchargement pendant la période t : Si un réservoir z est vide pendant la période t , aucun déchargement ne peut avoir lieu par ce réservoir. Cela implique la contrainte suivante :

$$NUDPB \times \sum_{j=1}^J F_{z,j,t} \geq \sum_{k=1}^K D_{z,k,t} \quad \forall z, t \quad (6.6)$$

De plus, si pendant la période t aucune UDPB ne demande du pétrole brut de type j d'un réservoir z , mais d'autres types de pétrole brut sont demandés (j', j''), le réservoir z ne décharge pas pendant la période t . Cela implique la contrainte suivante :

$$W2_{k,j,t} + \left[\sum_{jj=1}^J F_{z,j,t} - F_{z,j,t} \right] \geq D_{z,k,t} \quad \forall z, k, j, t \quad (6.7)$$

Etat de réservoirs : Une autre série d'inégalités valides restreint le nombre de réservoirs non vides. Si pendant la période t , des bateaux arrivent au port, avec une somme de quantités de pétrole brut égale à $\sum_{i=1}^I E_{i,t}$, la somme $\sum_{z=1}^Z \sum_{j=1}^J F_{z,j,t}$ doit être supérieure ou égale à $S2_t$ et inférieure ou égale au nombre total de réservoirs moins le nombre minimal de réservoirs nécessaire pour $\sum_{i=1}^I E_{i,t}$. Cela implique l'inégalité suivante :

$$S2_t \leq \sum_{z=1}^Z \sum_{j=1}^J F_{z,j,t} \leq NR - \sum_{i=1}^I E_{i,t} \quad \forall t \quad (6.8)$$

Liaison des périodes t et $t + 1$:

- Le premier groupe d'inégalités valides représente l'état d'un réservoir z à la période $t + 1$ connaissant son état à la période t est présenté par la suite. Avec cette inégalité valide nous garantissons que si rien ne se passe pendant la période $t + 1$ pour un réservoir z , ce réservoir conserve la situation qu'il avait pendant la période t :

$$\sum_{k=1}^K (-D_{z,k,t} \times W2_{z,k,t}) + F_{i,z,t} \leq F_{i,z,t+1} \leq F_{i,z,t} + \sum_{i=1}^I (C_{i,z,t+1} \times W1_{i,z,t+1}) \quad \forall z, j, t \quad (6.9)$$

Considérons seulement l'inégalité de droite de 6.9. Cette inégalité est équivalente à :

$$-1 - NP \leq \sum_{i=1}^I (-C_{i,z,t+1} \times W1_{i,z,t+1}) + F_{i,z,t+1} - F_{i,z,t} \leq 0 \quad \forall z, j, t \quad (6.10)$$

L'inégalité de gauche de 6.9 est équivalente à :

$$-1 - NUDPB \leq \sum_{k=1}^K (-D_{z,k,t} \times W2_{z,k,t}) + F_{i,z,t} - F_{i,z,t+1} \leq 0 \quad \forall z, j, t \quad (6.11)$$

Les bornes inférieures qui apparaissent sont les valeurs minimales obtenues par la somme des variables de décision.

Exemple explicatif : Si par exemple aucun bateau apportant du pétrole brut de type j n'arrive à un port de la raffinerie, pendant la période $t + 1$ (c'est à dire $\sum_{i=1}^I W1_{i,j,t+1} = 0 \implies \sum_{i=1}^I W1_{i,j,t+1} \times C_{i,z,t} = 0$) et si dans le réservoir z , il n'y avait pas du pétrole brut de type j pendant la période t , alors à la période $t + 1$, la variable $F_{z,j,t+1}$ sera égale à zéro.

Remarquons que l'ajout du terme $C_{i,z,t+1}$ est fait pour que nous puissions aussi tenir compte du cas où un bateau arrive avec du pétrole brut de type j mais qui ne charge pas dans le réservoir z . Le terme $F_{z,j,t}$ garantit que si rien ne se passe ($\sum_{i=1}^I C_{i,j,t+1} = \sum_{k=1}^K D_{z,k,t} = 0$), le réservoir reste au même état pendant la période $t + 1$. Finalement, nous ajoutons le terme $\sum_{k=1}^K (D_{z,k,t} \times W2_{k,j,t})$ et s'il y a un déchargement du réservoir z (où il y a le type j) pendant la période t la variable de décision $F_{z,j,t+1}$ est libre de prendre les valeurs 0 ou 1.

- Un deuxième groupe d'inégalités valides qui lient la période t à la période $t + 1$ est constitué d'inégalités qui garantissent que si à la période t , $F_{z,j,t} = 1$ et à la période $t + 1$, un bateau arrive avec du pétrole brut de type j' ($\sum_{i=1}^I W1_{i,j',t+1} \neq 0$) la variable de décision $C_{i,z,t+1}$ prend la valeur zéro. Si pendant la période t , le réservoir z est vide ($\sum_{i=1}^I F_{z,i,t} = 0$), et pendant la période $t + 1$ un bateau arrive avec du pétrole brut de type j , le réservoir z peut être chargé avec ce type de pétrole brut, c'est à dire :

$$\begin{aligned}
1 - \sum_{j=1}^J F_{z,j,t} + F_{z,j,t} + \sum_{k=1}^K D_{z,k,t} &\geq \sum_{i=1}^I (C_{i,z,t+1} \times W1_{i,j,t+1}) \quad \forall z, j, i, t \Rightarrow \\
-1 &\leq \sum_{j=1}^J F_{z,j,t} + F_{z,j,t} + \sum_{k=1}^K D_{z,k,t} - \sum_{i=1}^I (C_{i,z,t+1} \times W1_{i,j,t+1}) \leq 2 \quad \forall z, j, i, t
\end{aligned} \tag{6.12}$$

- En utilisant la règle de fonctionnement qui impose la contrainte qu'un réservoir ne peut pas être déchargé et être chargé simultanément pendant la période t , nous obtenons la troisième catégorie d'inégalités valides :

$$\begin{aligned}
D_{k,j,t+1} &\leq 1 - W2_{k,j,t+1} + \sum_{j=1}^J F_{z,j,t} + F_{z,j,t} \quad \forall z, k, t \Rightarrow \\
-2 &\leq D_{k,j,t+1} - \sum_{j=1}^J F_{z,j,t} - F_{z,j,t} \leq 1 - W2_{k,j,t+1} \quad \forall z, k, t
\end{aligned} \tag{6.13}$$

Si pendant la période t , le réservoir z est vide et si pendant la période $t + 1$ une UDPB k demande du pétrole brut de type j' alors $D_{z,k,t+1} = 0$.

6.1.3 Résultats

Après avoir ajouté les inégalités valides au PMR initial (pour l'exemple de référence le nombre total d'inégalités valides ajouté est égal à 1400 sur un nombre total de contraintes égal à 3100), on constate que la solution obtenue est beaucoup plus proche de la solution optimale du problème. Pour l'exemple de référence, la solution initiale du PMR est passée de zéro à 18 (rappelons que la solution optimale du MO11 est égale à 21). Cette nouvelle initialisation du PMR a impliqué une diminution du temps de résolution assez importante. Cependant, ce temps total de résolution reste supérieur au temps du modèle non décomposé (MO11). Les résultats sont résumés dans le tableau suivant :

Modèle	Temps de résolution
MO11	1h 04mm
Benders sans les inégalités valides	plus de 24h
Benders avec les inégalités valides	14h

TAB. 6.1 – L'effet des inégalités valides

On constate cependant, à partir de ces résultats, que même avec des inégalités valides, le temps de calcul reste important. Dans la section suivante, nous présentons notre étude sur les coupes générées par l'application de la méthode de décomposition de *Benders* classique et les problèmes qui se présentent. Pour résoudre ces problèmes, nous sommes amenés à proposer une procédure originale de génération multiples de coupes.

6.2 Génération multiple des coupes

Après avoir ajouté les inégalités valides au programme maître restreint (**PMR**), on a remarqué que le temps de résolution devient d'une part plus court, mais continue d'autre part à être plus grand que le temps de résolution du modèle non décomposé (cf. tableau 6.1). Ceci rend nécessaire le développement d'une méthodologie plus efficace qui nous permet de diminuer davantage le temps de résolution du modèle décomposé. En étudiant les coupes générées par l'algorithme de décomposition de *Benders* classique, nous avons constaté que les coupes générées sont des coupes peu denses. Une coupe est dite peu dense si seulement une petite partie des variables de décision y est prise en compte. Si, par exemple, à chaque coupe générée, seulement 10% des variables de décision sont prises en compte, la coupe peut être considérée comme une coupe de faible densité. Pour l'exemple de référence le nombre de variables de décision est égale à 420 et le nombre moyen de variables qui est présentés à chaque coupe est égale à 30-35, c'est-à-dire moins de 10%. L'idée étudiée ici est alors de générer des coupes non pas une par une mais par paquets, de façon à faire intervenir le plus grand nombre possible de variables de décision du programme maître restreint (**PMR**). De la sorte, l'espace de solution sera plus rapidement restreint et l'algorithme de *Benders* convergera à la solution optimale plus rapidement.

De plus, la plus grande partie du temps de résolution correspond à la résolution du **PMR** car il s'agit d'un programme en nombres entiers. Au contraire, les programmes esclaves (**PE**) sont des programmes linéaires en nombres continus et leurs temps de résolution ne sont pas importants. Pour profiter de cet avantage des **PE** et pour éliminer l'inconvénient du **PMR**, nous avons développé une nouvelle méthodologie. Cette méthodologie est une extension d'une idée présentée dans l'article [Gabrel et al., 1999]. Cette méthodologie implique la génération de plusieurs coupes valides à chaque itération, résolvant plusieurs fois successives un programme auxiliaire (**PA**), en utilisant les mêmes valeurs envoyées par le **PMR**. Ce programme est un programme en nombres continus et il a été développé en se fondant sur le programme esclave (**PE**) à l'étape considérée.

La conception de la génération multiple implique la génération simultanée de plusieurs coupes à chaque itération de notre algorithme. Pour diminuer le temps de convergence de notre algorithme vers la solution optimale, des coupes plus significatives doivent d'une part

être générées et le nombre de résolution du PMR doit d'autre part être diminué. Pour la première exigence, des coupes sont générées où le maximum de variables de décision du PMR sont présentes. Pour la deuxième exigence, une génération multiple des coupes significatives correspondantes à la même solution obtenue par le PMR est exercée.

En toute généralité et pour l'exemple de référence, un rayon extrême est généré par les programmes esclaves suivants :

$$(PE) \begin{cases} \text{Max } -z \\ \text{sous les contraintes :} \\ A \times x - z \times e \leq b - B \times \bar{y} \\ x \geq 0. \end{cases}$$

et les duals sont :

$$(Dual \ PE) \begin{cases} \text{Min } u^T \times (b - B \times \bar{y}) \\ \text{sous les contraintes :} \\ u^T \times A \geq 0 \\ -u^T \times e = -1 \\ u \geq 0 \end{cases}$$

En toute généralité, les coupes générées par les programmes esclaves sont de la forme suivante :

$$u^T \times (b - B \times y) \geq 0 \Rightarrow$$

$$(u^T \times B) \times y \leq u^T \times b$$

La procédure que nous suivons dans notre méthodologie est de générer plusieurs coupes à chaque itération, au lieu d'une seule qui est générée par l'algorithme de Benders classique. De plus, les variables de décision du PMR interviennent dans notre méthodologie de façon significative. Avant de présenter la procédure de la génération multiple de coupes significative, nous donnons les deux définitions suivantes :

- *Définition 1* : une variable y_{j_0} intervient d'une façon significative dans la contrainte générée si son coefficient n'est pas égal à zéro,
- *Définition 2* : une variable y_{j_0} est réputée « couverte » si son coefficient est supérieur ou égal à $10^{-3} \times \eta$ en valeur absolue,
- *Remarque* : η ainsi que les autres paramètres de configuration de notre algorithme vont être présentés par la suite. Notons que η est la borne des coefficients des variables de décision $y_j \forall j \neq j_0$.

6.3 Génération des coupes significatives

A chaque itération de l'algorithme de Benders classique, compte tenu de la solution optimale du PMR à l'itération considérée ($\bar{y}$), nous résolvons le programme esclave (PE) correspondant et ensuite nous générons la coupe suivante :

$$(u^T \times B) \times y \leq u^T \times b \quad (6.14)$$

où u est la solution optimale duale du PE à l'étape considérée. Le coefficient de y_j dans la coupe est $u^T \times B^j$. Dans notre application on constate expérimentalement qu'en moyenne 90% de ces coefficients sont égaux à zéro, ce qui implique la génération d'une coupe peu dense avec seulement 10% des variables de décision qui participent à la définition de la coupe.

L'idée étudiée ici est de générer des ensembles de coupes faisant globalement intervenir le plus grand nombre possible de variables de décision. Dans le problème générique qu'on veut résoudre, on ajoute des contraintes de borne sur les coefficients de chaque variable de décision y_j du programme maître restreint (PMR), afin de générer de nouvelles coupes valides. Les contraintes de borne sur les coefficients sont du type :

$$\alpha_j \leq (u^T \times B^j) \leq \beta_j \quad (6.15)$$

où α_j est la borne inférieure et β_j est la borne supérieure qu'on souhaite imposer au coefficient de la variable y_j .

Ces contraintes de borne sur les coefficients de chaque variable de décision y_j nous conduisent à résoudre des programmes esclaves duals « augmentés » (avec des contraintes supplémentaires) où le nombre de contraintes ajoutées est égal au nombre de variables y_j multiplié par 2 (à chaque variable correspond une contrainte de borne inférieur et de borne supérieur). Ces programmes sont appelés programmes auxiliaires duals (**PAD**) et prennent la forme suivante :

$$(\text{PAD}) \left\{ \begin{array}{l} \text{Min } u^T \times (b - B \times \bar{y}) \\ \text{sous les contraintes :} \\ u^T \times A \geq 0 \\ \text{ainsi que les deux contraintes de borne :} \\ -u^T \times B^j \geq -\beta_j \quad \forall j \\ u^T \times B^j \geq \alpha_j \quad \forall j \\ u \geq 0. \end{array} \right.$$

(Ils seront à résoudre pour différents combinaisons possibles des paramètres α_j et β_j comme expliqué plus loin). Appelons x , ϑ_j et μ_j les variables correspondant aux contraintes du pro-

gramme dual auxiliaire. Le programme primal (programme auxiliaire, PA) correspondant est :

$$(\mathbf{PA}) \left\{ \begin{array}{l} \text{Max} - z - \sum_j \beta_j \times \vartheta_j + \sum_j \alpha_j \times \mu_j \\ \text{sous les contraintes :} \\ A \times x - \sum_j B^j \times \vartheta_j + \sum_j B^j \times \mu_j \leq (b - B \times \bar{y}) \\ x, \theta, \mu \geq 0, \quad z \text{ de signe quelconque.} \end{array} \right.$$

Nous introduisons ce programme auxiliaire car, ayant les valeurs optimales $\bar{y}$ par le PMR de l'itération actuelle, ce programme est le programme esclave « augmenté » avec deux nouveaux groupes de variables ϑ_j et μ_j . Pour avoir une procédure générique et efficace, c'est plus facile de résoudre ce programme PA. Généralement, on a le choix entre résoudre PA ou PAD mais c'est plus commode sous la forme PA. En effet on va devoir résoudre l'un de ces programmes pour beaucoup de combinaisons α_j et β_j différentes. Pour PA, il suffit de changer les coefficients de la fonction objectif en gardant constant l'espace des solutions. Ceci rend la résolution de PA plus facile car pour l'itération $n+1$, on part d'une base réalisable obtenue par l'itération n .

Pour qu'on puisse générer à l'itération actuelle une coupe de façon à faire intervenir une variable de décision y_{j_0} , nous résolvons le programme auxiliaire (PA) qui vient d'être présenté, en obtenant les valeurs $\bar{y}$ par le PMR de l'étape considérée et en fixant les paramètres suivants :

$$\alpha_{j_0} = \beta_{j_0} = +1$$

$$\alpha_j = -\eta, \quad \beta_j = +\eta \quad \forall j \neq j_0$$

En fixant ces paramètres nous garantissons que la variable de décision y_{j_0} intervient dans la nouvelle coupe de façon significative en forçant son coefficient à prendre une valeur égale à 1. La validité de cette nouvelle coupe générée par le programme auxiliaire, ayant les valeurs de $\bar{y}$ par le PMR de l'itération actuelle, résulte de la validité des PAD. Une solution optimale duale est demandée par le PE dual de l'itération actuelle, dans un espace restreint de son espace de solutions. Nous rappelons que le PAD est obtenue par le PE dual en ajoutant les deux contraintes de borne au coefficient de chaque variable de décision du PMR.

Remarquons qu'avec cette procédure nous garantissons la présence de la variable de décision y_{j_0} dans la nouvelle coupe générée sans interdire aux autres coefficients de prendre une valeur non nulle. Par contre, d'après nos exemples numériques, nous avons constaté que le nombre minimum de variables de décision avec un coefficient non nul dans la nouvelle coupe est, au pire des cas égal à celui de la coupe générée par le PE.

6.4 Benders avec génération multiple de coupes valides

Comme nous avons dit dans la section 6.2, en étudiant les coupes générées par l'algorithme de décomposition de Benders classique, nous avons constaté que ces coupes sont des coupes peu denses. Dans la section précédente, nous avons présenté une procédure pour générer pendant une itération une coupe dans laquelle en fixant des valeurs aux paramètres α_j et β_j du PA de l'étape considérée, la présence dans la coupe d'une variable y_{j_0} est garantie. Cette procédure est nécessaire mais pas suffisante pour restreindre l'espace de solutions du PMR de façon optimale. L'idée est alors de générer des coupes non pas une par une, mais par paquet pendant chaque itération, de façon à faire intervenir le plus grand nombre possible de variables de décision. Cette procédure est appelée la génération multiple de coupes.

La procédure poursuivie pour la génération multiple de nouvelles coupes à chaque itération de notre algorithme de Benders est la suivante : au lieu de résoudre seulement le programme esclave correspondant aux valeurs optimales $\bar{y}$ du programme PMR à l'itération considérée (cf. section 5.4), nous effectuons des résolutions successives au programme auxiliaire (PA) correspondant à ces mêmes valeurs. A chaque résolution du PA, nous fixons une autre paire de paramètres (à une valeur égalé à 1) en donnant à nouveau les valeurs η et $-\eta$ aux bornes inférieures et supérieures du coefficient précédent.

Procédure complète pour la génération multiple des coupes

Pour générer une inégalité où y_{j_0} intervient de façon significative, nous résolvons le PA à l'itération considérée, tout en fixant les paramètres suivants :

$$\alpha_{j_0} = \beta_{j_0} = +1, \quad \alpha_j = -\eta, \quad \beta_j = +\eta \quad \forall j \neq j_0$$

Après avoir résolu le PA avec les paramètres précédents, nous ajoutons la coupe générée au programme maître restreint et nous ré-initialisons les valeurs des α_j et β_j . Pour une autre variable non couverte j'_0 , nous donnerons la valeur +1 aux $\alpha_{j'_0}$ et $\beta_{j'_0}$, et à nouveau les valeurs η et $-\eta$ respectivement aux bornes inférieures et supérieures du coefficient de y_{j_0} . Nous générons une deuxième coupe et la procédure continue jusqu'au moment où le nombre de coupes générées à l'itération courante est satisfait. Une fois la procédure arrêtée, on a, soit un nombre de coupes générées égal à une borne ξ choisie a priori, soit toutes les variables de décision du PMR couvertes. L'organigramme complet de l'algorithme de Benders avec la génération multiple des coupes significatives est présenté dans la figure 6.1.

Nous remarquons qu'avant de générer les coupes par le programme auxiliaire (PA) l'algorithme génère une coupe par la procédure classique de l'algorithme de Benders (par le programme esclave). Ensuite un test est utilisé pour vérifier quelles variables sont réputées cou-

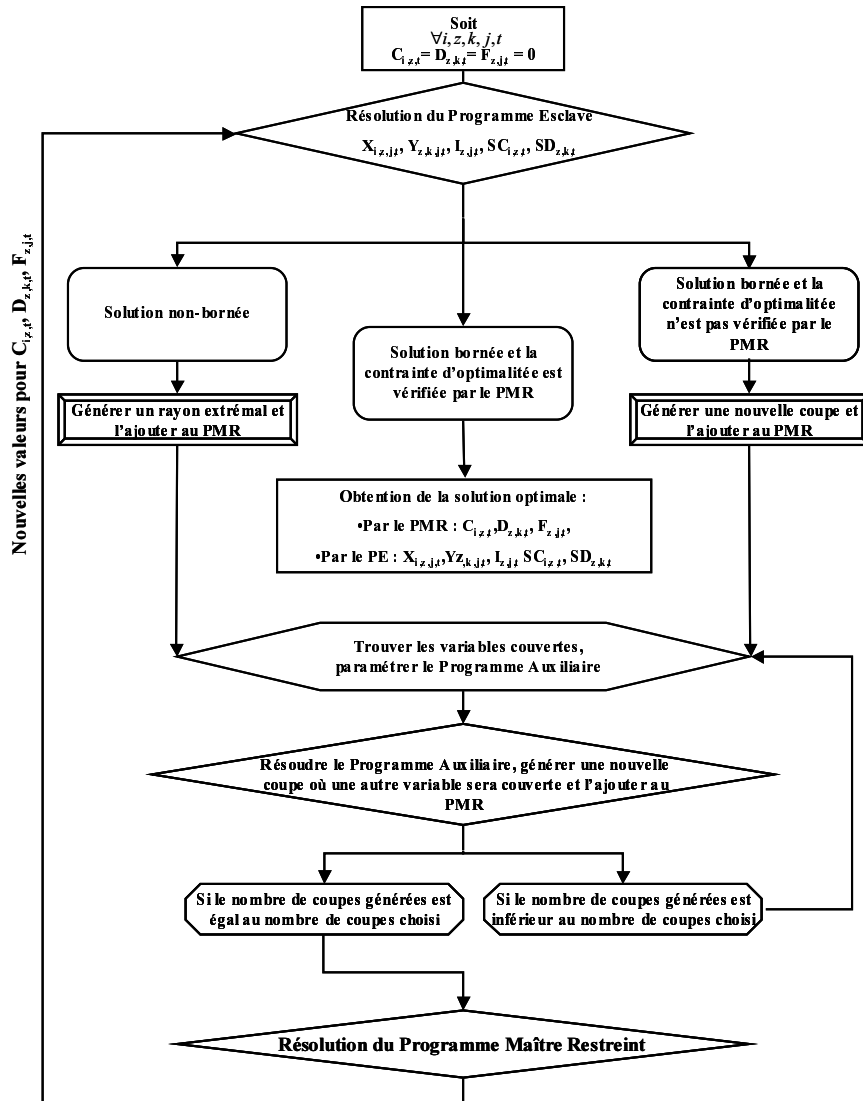


Fig. 6.1 – Organigramme de Benders/Multi-génération des coupes

vertes par la coupe faite. Après avoir trouvé toutes les variables de décision non couvertes, le programme auxiliaire est résolu plusieurs fois pour générer le nombre désiré de coupes à chaque fois en prenant comme indice j_0 , l'indice d'une variable non encore couverte par les coupes générées jusqu'ici.

La génération multiple des coupes est considérée comme une procédure qui, en fixant les valeurs des variables de décision du PMR, génère des coupes vers toutes les directions. De cette façon, un nombre maximal de coupes est ajouté au PMR, lui donnant un espace de solutions restreint de façon optimale et réduisant le nombre d'itérations de l'algorithme.

6.5 Résultats numériques

Après avoir effectué notre analyse numérique, nous avons constaté que la méthode de décomposition de Benders est vraiment moins efficace, en fonction du temps de résolution, que la méthode de résolution directe du modèle non décomposé. Nous avons constaté que pour des exemples de petite taille (nombre de variables $\simeq 200$ -250, nombre de contraintes $\simeq 250$ -300) et pour la génération multiple des coupes, le temps de résolution du modèle décomposé est à peu près égal au temps de résolution du modèle non décomposé. En augmentant la taille du problème, la résolution directe du modèle non décomposé continue, jusqu'à maintenant, à être plus efficace.

Les résultats présentés ensuite correspondent à des exemples réels de trois raffineries qui se trouvent en Grèce dont la planification a eu lieu dans les années précédentes. Une comparaison est faite entre les résultats obtenus par l'application de notre modèle et le plan de production que les gestionnaires avaient choisi. Dans l'annexe B.1, nous présentons en détails certains des exemples traités. Remarquons que nous ne pouvons pas donner les détails de tous les exemples pour raison de confidentialité. De plus, nous remarquons que ces exemples correspondent à des années passées. Ceci implique un changement probable du type de préparation du mélange ou/et de l'option de distillation choisie ou/et du nombre de réservoirs disponibles pour le stockage du pétrole brut dans les raffineries.

Notons par :

- MA11 : méthode d'optimisation actuelle dans la raffinerie (préparation de mélange juste en amont des UDPB, distillation exacte),
- MO11 : modèle optimal (préparation du mélange juste en amont des UDPB, distillation exacte),
- IV : inégalités valides,
- DCB : décomposition classique de Benders,
- DBGGM : décomposition de Benders avec génération multiple de coupes valides,
- MO11 DCB : décomposition classique de Benders appliquée au modèle optimal (MO11),
- MO11 DCB avec IV : décomposition classique de Benders appliquée au modèle optimal (MO11) avec les inégalités valides (IV),
- MO11 DBGGM avec IV : décomposition de Benders avec génération multiple appliquée au modèle optimal (MO11), avec les inégalités valides (IV),
- MO11 avec IV : le modèle optimal (MO11), (non décomposé) avec les inégalités valides (IV).

6.5.1 Comparaison générale des modèles

Nous avons développé plusieurs modélisations différentes pour résoudre le problème d'Ordonancement des activités de **C**hargement/**D**échargement du pétrole brut (**OCD**). Au début, nous avons appliqué la méthode d'optimisation actuelle. La solution sous-optimale obtenue par cette méthode était comparée à la solution optimale obtenue par notre modèle exact. Nous avons constaté pour l'exemple de référence (cf. section 4.5) une diminution de 16% du nombre de reconfigurations des 6 réservoirs de la raffinerie.

D'une part, nous avons constaté une diminution importante du nombre de reconfigurations des réservoirs mais d'autre part une augmentation du temps de résolution. En constatant que la réduction du nombre de reconfigurations des réservoirs est intéressante, nous avons développé des méthodes pour diminuer le temps de résolution du modèle exact qui était assez important. Pour l'exemple de référence les deux temps de résolution sont très éloignés. Pour l'optimisation actuelle, quelques secondes sont suffisantes pour trouver la solution et pour le modèle exact, il nous faut 16 heures.

Type de partition	Modèle	Temps de résolution	Solution
Sur événement	MA	10 sec	25
Sur heure	MO11	$\simeq 16h$	21
Sur événement	MO11	1h 04min	21
Sur événement	MO11 DCB	$\simeq 24h$	21
Sur événement	MO11 DCB avec IV	$\simeq 14h$	21
Sur événement	MO11 DBGGM avec IV	$\simeq 9h$	21
Sur événement	MO11 avec IV	13 min	21

TAB. 6.2 – Comparaison globale

La première approche que nous avons développée est constituée d'un autre type de partition de l'horizon de temps : la partition sur événement. Avec ce type de partition, nous avons constaté une grande diminution du temps de résolution du modèle exact de 16h à 1h et 04min grâce à la diminution du nombre de variables de décision et des contraintes. Notre recherche continue en appliquant la méthode de décomposition de Benders afin de diminuer davantage le temps de résolution. Dans un premier temps, nous avons appliqué l'algorithme de Benders classique (MO11 DCB) mais les résultats au niveau du temps de résolution n'étaient pas comparables à ceux du modèle exact (MO11). Le temps de résolution du MO11 DCB était de plus d'un jour. En étudiant les raisons pour lesquelles le MO11 DCB était assez lent, nous avons fréquemment constaté infaisabilité des PE (programmes esclaves). En ce sens nous avons développé une série d'inégalités valides et nous les avons ajoutées au programme PMR (pour mieux l'initialiser) du modèle MO11 DCB. Nous avons constaté une réduction importante de 24h à 14h = 42%, mais le temps de résolution est resté plus grand que celui du modèle MO11 (avec le nouveau

type de partition de l'horizon du temps). La génération des coupes denses par l'algorithme de Benders classique et la nécessité de diminuer davantage le temps de résolution du MO11 DCB nous a donné l'idée de la génération multiple des coupes valides et du développement du MO11 DBGm avec des inégalités valides. Le temps de résolution de 14h passe à 9h, ce qui représente une réduction assez importante de l'ordre de 35%. Au total, nous avons une réduction de 62.5% entre le temps du modèle MO11 BDC et le MO11 DBGm avec IV. Malgré cette grande diminution de temps de résolution, le modèle MO11 décomposé reste encore plus lent que le modèle MO11 non décomposé. Finalement, pour qu'on puisse obtenir un meilleur temps de résolution, nous avons utilisé un résultat obtenu par l'application de la méthode de décomposition. Nous avons ajouté toutes les inégalités valides, générées par l'étude du programme maître restreint (PMR) du modèle décomposé, au modèle MO11 non décomposé. Le résultat obtenu est d'avoir une nouvelle réduction du temps de résolution de l'ordre de 79% (cf. tableau 6.2).

6.5.2 Comparaison entre l'optimisation exacte et l'optimisation existante

La raison pour laquelle nous avons développé une modélisation générique, qui peut nous donner efficacement et rapidement une solution optimale, est que la méthode d'optimisation appliquée actuellement à la raffinerie donnait aux décideurs une solution sous-optimale éloignée de la solution optimale obtenue par un modèle exact. La solution obtenue par notre modèle montre une réduction importante pour des exemples réels présentés par la suite. Dans le tableau qui suit, nous présentons les résultats obtenus par les deux types d'optimisation et pour les trois premiers modèles.

Exemple	Modèle MA11	Modèle MO11	Différence relative
Exe.1	9 reconfig.	8 reconfig.	11.11%
Exe.2	8 reconfig.	7 reconfig.	12.5%
Exe.3	14 reconfig.	13 reconfig.	7.14%
Exe.4	22 reconfig.	19 reconfig.	13.63%
Exe.5	13 reconfig.	12 reconfig.	7.69%
	Modèle MA12	Modèle MO12	Différence relative
Exe.6	16 reconfig.	13 reconfig.	18.75%
Exe.7	13 reconfig.	12 reconfig.	7.69%
Exe.8	20 reconfig.	15 reconfig.	25%
Exe.9	23 reconfig.	18 reconfig.	21.73%
Exe.10	16 reconfig.	13 reconfig.	18.75%
	Modèle MA21	Modèle MO21	Différence relative
Exe.11	23 reconfig.	20 reconfig.	13.04%
Exe.12	18 reconfig.	15 reconfig.	16.66%
Exe.13	19 reconfig.	18 reconfig.	5.26%
Exe.14	35 reconfig.	30 reconfig.	14.28%
Exe.15	17 reconfig.	15 reconfig.	11.76%

TAB. 6.3 – Comparaison du MO et du MA

Nous pouvons constater une réduction de 10% jusqu'à 20% du nombre de reconfigurations des réservoirs de la raffinerie. Cette réduction est importante pour la raffinerie parce qu'elle lui permet d'avoir une meilleure planification sur les prochaines périodes (changer l'utilisation d'un réservoir, maintenir le système, utiliser un réservoir par une autre raffinerie) avec l'utilisation optimale des réservoirs (cf. section 4.4).

6.5.3 Comparaison entre algorithme de Benders classique et Benders avec génération multiple de coupes valides

Dans cette section, nous présentons l'effet de la génération multiple quand nous utilisons la méthode de décomposition de Benders. Notre objectif n'est pas de comparer la modélisation non décomposée et la modélisation décomposée. Le modèle non décomposé après avoir ajouté les inégalités valides est toujours plus efficace que le modèle décomposé sans ou avec la génération multiple des coupes valides. Même si l'application de la méthode de décomposition de Benders au cas d'une raffinerie n'est pas avantageuse, le modèle décomposé avec génération multiple prévoit son efficacité si on le compare avec la méthode de Benders classique.

Généralement, la génération multiple des coupes valides influence positivement l'efficacité du modèle décomposé à condition de choisir le bon nombre de coupes à générer à chaque itération et les meilleures paramètres pour les programmes auxiliaires (**PA**). Le temps de résolution du modèle DBGGM (décomposition de Benders avec génération multiple de coupes valides) est significativement plus court que le temps du modèle DCB (décomposition classique de Benders). Pour certains exemples, le temps de résolution décroît jusqu'à 98%. Dans le tableau qui suit, nous donnons cinq exemples pour montrer l'effet de la génération multiple au modèle décomposé. Pour qu'on puisse étudier au mieux cet effet nous avons fait un changement au programme maître restreint de notre modèle décomposé. Nous n'avons pas fait la relaxation des deux variables de décision ($SC_{i,z,t}$, $SD_{k,z,t}$) qui se trouvent au niveau de la fonction objectif du programme esclave. Ceci implique un programme esclave plus simple à résoudre car il n'a pas de fonction objectif. De plus, le programme maître restreint devient plus compliqué (et donc plus long à résoudre) car il a une fonction objectif et deux groupes de contraintes supplémentaires. Après avoir fait ce changement au modèle décomposé, les résultats obtenus sont les suivants :

D'après ces exemples, nous constatons que l'effet de la génération multiple est important. Une diminution importante du temps de résolution ainsi que du nombre d'itérations est apparue. En augmentant la complexité du programme maître restreint (PMR), la différence entre la résolution du modèle décomposé sans génération multiple de coupes et avec génération multiple devient plus importante grâce à la diminution du nombre d'itérations. Rappelons que le nombre d'itérations correspond au nombre de fois où nous résolvons le PMR. Le PMR est un

Exemple	Benders Classique		Benders génération multiple		Différence relative	
	Temps	Itérations	Temps	Itérations	Temps	Itérations
Exe.16	25 min.	814	20 min.	42	20%	94%
Exe.17	$\simeq$ 24 h.	3500	22 min.	44	98%	98%
Exe.18	28 min.	817	2 min.	65	92%	92%
Exe.19	$\simeq$ 12 h.	3317	5 min.	106	99%	96%
Exe.20	27 min.	228	5.5 min.	27	76%	88%

TAB. 6.4 – Comparaison entre DCB et DBGM

programme en nombres entiers donc plus difficile à résoudre que le programme esclave (PE) qui est un programme en nombres continus. Ceci a un effet plus grand quand le PMR est plus complexe (plus de variables de décision et plus de contraintes).

Ce que nous avons en outre constaté est que la génération multiple donne toujours un nombre d'itérations plus petit que le nombre d'itérations de l'algorithme de Benders classique (cf. tableau 6.5). Pour l'exemple exe.1 (présente à l'annexe B.1), l'algorithme de Benders classique trouve la solution optimale en **38** secondes en faisant **180** itérations. Si nous appliquons la génération multiple, le temps de résolution décroît jusqu'à 44% (nombre de coupes générées à chaque itération égal à 4) et le nombre d'itérations décroît jusqu'à 71% (nombre de coupes générées à chaque itération est égal à 32).

Nombre de coupes	Temps de résolution	Nombre d'itérations
1	25sec.	88
4	21sec.	64
7	33sec.	62
10	1min. 21sec.	62
13	2min.	71
16	1min. 24sec.	48
19	1min. 24sec.	54
22	7min. 16sec.	130
25	2min. 25 sec.	57
28	2min. 30sec.	55
32	2min. 40sec.	52

TAB. 6.5 – L'effet de la génération multiple des coupes

Nous avons constaté que l'augmentation du nombre de coupes diminue généralement de façon importante le nombre d'itérations. La réduction minimale apparue à l'exemple exe.1 est égale à 27% (de 180 itérations à 130 itérations). De plus, nous avons constaté qu'il existe des cas où la génération multiple d'un grand nombre de coupes influence certaines fois de façon négative l'algorithme de décomposition. Par exemple, si à chaque itération nous générons 22 coupes, le temps de résolution devient très grand. Remarquons que même si le temps de résolution en générant 22 coupes à chaque itérations est lent, le nombre d'itérations est toujours inférieur

à celui de l'algorithme de Benders classique. Cela implique la nécessité de trouver le nombre de coupes qui a un effet positif, pas seulement en nombre d'itérations, mais aussi en temps de résolution.

En outre, nous avons constaté que si nous changeons les paramètres du programme auxiliaire, nous obtenons des résultats différents. Rappelons que les paramètres utilisés par notre algorithme sont α_j qui est la borne inférieure et β_j qui est la borne supérieure du coefficient de la variable de décision y_j . Rappelons aussi que pour générer une inégalité où y_{j_0} intervient d'une façon significative, nous résolvons le programme auxiliaire (PA), tout en fixant les paramètres suivantes :

$$\alpha_{j_0} = \beta_{j_0} = +1$$

$$\alpha_j = -\eta, \quad \beta_j = +\eta \quad \forall j \neq j_0$$

Prenons l'exemple exe.1 : si nous générons 22 coupes à chaque itération en choisissant comme valeurs des paramètres $\eta = 1$ et $\alpha_{j_0} = \beta_{j_0} = +3$, nous obtenons les résultats suivants :

- temps de résolution égal à 7 min. et 16 sec.,
- nombre d'itérations égal à 130.

Si pour le même exemple, nous choisissons d'autres valeurs pour les paramètres $\eta = 1$ et $\alpha_{j_0} = \beta_{j_0} = +5$, les résultats obtenus sont différents :

- temps de résolution égal à 4 min.,
- nombre d'itérations égal à 90.

Ceci implique que le choix des paramètres est un choix critique pour l'efficacité de l'algorithme.

La nécessité de développer une procédure pour trouver le nombre de coupes idéales et en même temps les paramètres optimaux pour le programme auxiliaire est une procédure qui n'est pas optimisée dans le cadre de cette thèse, mais qui peut être une bonne perspective pour l'avenir.

Finalement, nous avons constaté que si l'on n'examine pas la première coupe générée par l'algorithme de Benders classique et nous générons successivement toutes les coupes possibles, une réduction du temps de résolution de 10% est faite. Nous pensons que cette caractéristique est valable pour notre application et pas dans toutes les applications où la méthode de décomposition de Benders est utilisée. Nous proposons alors de tester la génération multiple avec ou sans le test et de choisir après la meilleure stratégie.

Conclusion-Perspectives

Dans cette deuxième partie de thèse, nous avons étudié le problème d'ordonnancement des activités de chargement/déchargement (**OCD**) du pétrole brut dans les réservoirs de stockage intermédiaires, entre les navires et les unités de distillation du pétrole brut d'une raffinerie. Après avoir visité plusieurs raffineries, nous avons constaté qu'il existe plusieurs types de préparation de mélanges et plusieurs options de distillation. L'objectif de notre projet de recherche est de développer un modèle générique qui résout le problème de l'OCD pour tous les types de configuration d'une raffinerie. L'objectif à optimiser est le coût de reconfiguration des réservoirs nécessaire au chargement par les navires et au déchargement vers les unités de distillation. Le coût de reconfiguration est important parce que la reconfiguration d'un réservoir implique une série d'opérations coûteuses.

Dans cette partie de thèse, après la présentation d'un état de l'art sur le problème d'ordonnancement de pétrole brut, nous proposons une modélisation générique, laquelle est comparée avec celle de la situation courante. Initialement, le temps de résolution de notre modèle est assez long mais la solution optimale obtenue montre un gain important. Cela implique la nécessité de développer une procédure pour améliorer l'efficacité de la résolution de notre modélisation. Nous avons développé une nouvelle répartition de l'horizon de temps qui a engendré une grande diminution du temps de résolution. Ensuite, nous avons appliqué la méthode de décomposition de Benders classique afin d'améliorer encore l'efficacité de notre modèle générique. Malheureusement, les résultats obtenus étaient décevants, d'où l'idée de développer une série d'inégalités valides. Les inégalités valides ont bien initialisé le programme maître restreint (**PMR**) mais le modèle non décomposé restait encore plus efficace.

Afin d'améliorer l'efficacité du modèle décomposé, nous avons non seulement développé une nouvelle méthodologie pour améliorer l'efficacité de l'algorithme de Benders dans le cas de l'application de la raffinerie, mais aussi une procédure générale appliquée dans tous les cas où la méthode de Benders est utilisée. L'idée initiale est la génération des coupes significatives, c'est-à-dire des coupes où le plus grand nombre de variables de décision du programme maître restreint (**PMR**) apparaît. Ensuite, à l'aide d'un programme auxiliaire, nous produisons plu-

sieurs coupes significatives à la fois. A chaque itération, nous ajoutons un grand nombre de coupes significatives avant de résoudre le programme maître restreint.

Nous avons testé plusieurs exemples numériques et les résultats obtenus concernant la performance de cette nouvelle méthodologie sont intéressants. D'une part, nous avons comparé le modèle non décomposé et le modèle décomposé et d'autre part, nous avons comparé la méthode de décomposition de Benders classique avec la méthode de génération multiple des coupes significatives. Pour notre application, la génération multiple diminue fortement le temps de résolution mais malgré cette grande diminution, le modèle non décomposé reste plus efficace. La comparaison entre les deux méthodes de décomposition a donné les résultats les plus intéressants. Généralement, la génération multiple des coupes valides influence positivement l'efficacité du modèle décomposé à condition de choisir le bon nombre de coupes à générer à chaque itération et les meilleurs paramètres pour le programme auxiliaire qui génère ces coupes. Nous avons constaté en moyenne une réduction de 90% du nombre des itérations et du temps de résolution (facteur d'amélioration de 10).

Notre travail suggère évidemment de nombreuses questions dont l'étude ultérieure serait intéressante :

- D'après notre analyse, nous avons constaté que la variation des paramètres du modèle auxiliaire donne des résultats différents. La meilleure sélection de ces paramètres peut être une possibilité pour la suite de cette recherche. Rappelons que le nombre de coupes générées à chaque itération correspond au nombre de résolutions du modèle auxiliaire. Nos résultats nous ont montré qu'il existe une limite au delà de laquelle la génération des coupes supplémentaires influence négativement l'efficacité de l'algorithme. Après cette limite, d'une part, la génération des coupes supplémentaires augmente le temps total de résolution de l'algorithme (résoudre plus de fois le PA) et d'autre part, ces coupes ne restreignent plus (ou plus assez) l'espace des solutions du programme maître restreint (**PMR**). Le développement d'une procédure qui sélectionne les coupes les plus significatives parmi toutes les coupes et les ajoute au PMR peut s'inscrire comme une autre extension de nos travaux de recherche,
- Concernant la modélisation développée pour le problème d'ordonnancement des activités de chargement/déchargement du pétrole brut, l'analyse des cas non linéaires peut être une suite de ce projet de recherche. Il est à rappeler qu'en programmation mathématique, la RLT (reformulation linearisation technique for bilinear programming) est une technique de linéarisation des modèles bilinéaires. Introduite par [Sherali and Alameddine, 1992], et appliquée pour ce type de problème par [Quesada and Grossmann, 1995], celle-ci donne une borne inférieure du problème bilinéaire à l'aide d'un programme linéaire. Cette approche

peut s'inscrire comme une démarche intéressante, à utiliser néanmoins avec précaution. Son intérêt est la linéarité qui va réduire la complexité de l'algorithme de résolution. En outre, son utilisation systématique peut se faire pour le calcul d'une borne inférieure (supérieure) s'il s'agit de minimiser (maximiser) un coût (profit),

- Enfin, les approches heuristiques peuvent présenter un intérêt lorsque la taille du problème rend impossible l'application des méthodes exactes du fait de la complexité associée. Ils serait intéressant d'explorer la possibilité de les utiliser pour construire une bonne solution initiale pour les programmes esclaves de la méthode de décomposition de Benders afin d'accélérer la convergence des deux programmes (PE et PMR) de cette méthode.

Annexe B.1

	Réservoir	Type0	Type1	Type2	Type3
	R1	40000			
	R2	80000			
	R3			95000	
	R4			100000	
	R5				
	R6				
	Périodes	UDPB 1	UDPB 2		Arrivé
0		Type0 : 15000	Type0 : 24500		
	1	Type1 : -	Type1 : -		
	102 h	Type2 : 15000	Type2 : 35400		
102		Type3 : -	Type3 : -		
102		Type0 : 15000	Type0 : 30000		
	2	Type1 : -	Type1 : -		
	102 h	Type2 : -	Type2 : -		
204		Type3 : -	Type3 : -		
204		Type0 : 20000	Type0 : -		
	3	Type1 : -	Type1 : -		Type0 : 120000
	36 h	Type2 : -	Type2 : 40000		
240		Type3 : -	Type3 : -		
240		Type0 : 10000	Type0 : -		
	4	Type1 : -	Type1 : -		
	102 h	Type2 : -	Type2 : 20000		
342		Type3 : -	Type3 : -		

TAB. 1.1 – Exemple 1

	Réservoir	Type0	Type1	Type2	Type3
	R1	80000			
	R2			100000	
	R3	40000			
	R4				
	R5				
	Période/durée	UDPB 1	UDPB 2		Arrivé
0		Type0 : 15000	Type0 : 24500		
	1/90h	Type1 : -	Type1 : -		
		Type2 : 5000	Type2 : -		
90		Type3 : -	Type3 : -		
90		Type0 : 15000	Type0 : 30000		
	2/110h	Type1 : -	Type1 : -		
		Type2 : -	Type2 : -		
200		Type3 : -	Type3 : -		
200		Type0 : 20000	Type0 : -		
	3/36h	Type1 : -	Type1 : -		Type0 : 120000
		Type2 : -	Type2 : 10000		
236		Type3 : -	Type3 : -		
236		Type0 : 10000	Type0 : -		
	4/114h	Type1 : -	Type1 : -		
		Type2 : -	Type2 : 20000		
350		Type3 : -	Type3 : -		

TAB. 1.2 – Exemple 2

	Réservoir	Type0	Type1	Type2	Type3
	R1	40000			
	R2	40000			
	R3			95000	
	R4	40000			
	R5				
	R6				
	R7			60000	
	R8			50000	
	Période/durée	UDPB 1	UDPB 2		Arrivé
0		Type0 : 15000	Type0 : 24600		
	1/152h	Type1 : -	Type1 : -		
		Type2 : 15000	Type2 : 25400		
152		Type3 : -	Type3 : -		
152		Type0 : 20000	Type0 : 30000		
	2/140h	Type1 : -	Type1 : -		
		Type2 : -	Type2 : -		
292		Type3 : -	Type3 : -		
292		Type0 : 10000	Type0 : -		
	3/36h	Type1 : -	Type1 : -		Type0 : 130000
		Type2 : -	Type2 : 50000		
328		Type3 : -	Type3 : -		
328		Type0 : 15000	Type0 : -		
	4/160h	Type1 : -	Type1 : -		
		Type2 : -	Type2 : 20000		
488		Type3 : -	Type3 : -		
488		Type0 : 60000	Type0 : 30000		
	5/36h	Type1 : -	Type1 : -		Type1 : 110000
		Type2 : -	Type2 : -		
524		Type3 : -	Type3 : -		

TAB. 1.3 – Exemple 3

	Réservoir	Type0	Type1	Type2	Type3		
	R1			95000			
	R2	90000					
	R3			100000			
	R4	50000					
	R5						
	R6						
	Période/durée	UDPB 1	UDPB 2	UDPB 3	UDPB 4		Arrivé
0		Type0 : 7000	Type0 : 4000	Type0 : 7480	Type0 : 20000		
	1/140h	Type1 : -	Type1 : -	Type1 : -	Type1 : -		
		Type2 : -	Type2 : 16000	Type2 : 14520	Type2 : -		
140		Type3 : -	Type3 : -	Type3 : -	Type3 : -		
140		Type0 : 7800	Type0 : 10000	Type0 : 7000	Type0 : 10000		
	2/110h	Type1 : -	Type1 : -	Type1 : -	Type1 : -		
		Type2 : 5200	Type2 : -	Type2 : -	Type2 : -		
250		Type3 : -	Type3 : -	Type3 : -	Type3 : -		
250		Type0 : 10000	Type0 : -	Type0 : 10000	Type0 : 10000		
	3/36h	Type1 : -	Type1 : -	Type1 : -	Type1 : -		Type0 : 120000
		Type2 : -	Type2 : 30000	Type2 : -	Type2 : 10000		
286		Type3 : -	Type3 : -	Type3 : -	Type3 : -		
286		Type0 : 2000	Type0 : -	Type0 : 8000	Type0 : -		
	4/74h	Type1 : -	Type1 : -	Type1 : -	Type1 : -		
		Type2 : -	Type2 : 10000	Type2 : -	Type2 : 10000		
		Type3 : -	Type3 : -	Type3 : -	Type3 : -		
360		Type0 : 15000	Type0 : 20000	Type0 : 15000	Type0 : 20000		
360		Type1 : -	Type1 : -	Type1 : -	Type1 : -		Type1 : 110000
	5/36h	Type2 : -	Type2 : -	Type2 : 5000	Type2 : -		
		Type3 : -	Type3 : -	Type3 : -	Type3 : -		
396							

TAB. 1.4 – Exemple 4

	Réservoir	Type0	Type1	Type2	Type3
	R1				30000
	R2			95000	
	R3			100000	
	R4				
	R5				
	R6				
	Période/durée	UDPB	UDPB		Arrivé
0		Type0 : -	Type0 : -		
	1/36h	Type1 : -	Type1 : -		Type3 : 90000
		Type2 : 15000	Type2 : 35400		
36		Type3 : -	Type3 : -		
36		Type0 : -	Type0 : -		
	2/72h	Type1 : -	Type1 : -		
		Type2 : -	Type2 : -		
108		Type3 : 15000	Type3 : 30000		
108		Type0 : -	Type0 : -		
	3/36h	Type1 : -	Type1 : -		Type3 : 110000
		Type2 : -	Type2 : 40000		
144		Type3 : 20000	Type3 : 10000		
144		Type0 : -	Type0 : -		
	4/154h	Type1 : -	Type1 : -		
		Type2 : -	Type2 : -		
298		Type3 : 30000	Type3 : 50000		
298		Type0 : -	Type0 : -		
	5/36h	Type1 : -	Type1 : -		Type1 : 100000
		Type2 : -	Type2 : 20000		
334		Type3 : 10000	Type3 : -		

TAB. 1.5 – Exemple 5

Exemples	Partition sur heure	Partition sur événement
ex.1	8	8
ex.2	7	7
ex.3	13	13
ex.4	19	19
ex.5	12	12
ex.6	13	13
ex.7	12	12
ex.8	15	15
ex.9	18	18
ex.10	13	13
ex.11	20	20
ex.12	15	15
ex.13	18	18
ex.14	30	30
ex.15	15	15

TAB. 1.6 – Répartition sur heure/sur événement

Bibliographie

- Bassett, M., Pekny, J., and Reklaitis, G. (1996). Decomposition techniques for the solution of large-scale scheduling problems. *AIChE Journal*, 42 :3373–3387.
- Benders, J. F. (1962). Partitioning procedures for solving mixed variable programming problems. *Numerical Mathématique*, 4 :238–252.
- Bilington, P., McClain, J., and Thomas, L. (1983). Mathematical programming approaches to capacity-constrained mrp systems : Review, formulation and problem reduction. *Management Science*, 29 (10) :1126–1141.
- Bonner and Moore (1979). The refinery and petrochemical modeling system. *Management Science RPMS, Houston*.
- Byrne, M. and Bakir, M. (1999). Production planning using a hybrid simulation-analytical approach. *International Journal of Production Economic*, 59 :305–311.
- Cohen, M. and Lee, H. (1989). Resource deployment analysis of global manufacturing and distribution networks. *Journal of Manufacturing and Operations Management*, 2 :81–104.
- Cohen, M. and Moon, S. (1990). Impact of production scale economies, manufacturing complexity, and transportation costs on supply chain facility networks. *Journal of Manufacturing and Operations Management*, 3 :269–292.
- Gabrel, V., Knippel, A., and Minoux, M. (1999). Exact solution of multicommodity network optimization problems with general step cost functions. *Operations Research Letters*, 25 :15–23.
- Gnonia, M., Iavagnilio, R., Mossa, G., Mummolo, G., and DiLeva, A. (2003). Production planning of a multi-site manufacturing system by hybrid modelling : A case study from the automotive industry. *International Journal Production Economics*, 85 :251–262.
- Hax, A. and Candea, D. (1984). Production and inventory management. *Prentice Hall*.
- Holt, C., Modigliani, Muth, J., and Simon, H. (1960). Planning production, inventories and work force. *Englewood Cliffs, NJ : Prentice-Hall*.
- Ishii, K., Takahashi, K., and Muramatsu, R. (1988). Integrated production, inventory and distribution systems. *International Journal of Production Research*, 26(3) :473–482.

- Jen Ming, C. and Tsung Hui, C. (2005). The multi item replenishment problem in two echelon supply chain : the effect of centralization versus decentralization. *Computers & Operations Research*, 32 :3191–3207.
- Jia, Z. and Ierapetritou, M. (2004). Efficient short-term scheduling of refinery operation based on continuous time formulation. *Computer & Chemical Engineering*, 28 :1001–1019.
- Jia, Z., Ierapetritou, M., and Kelly, J. D. (2003). Refinery short-term scheduling using continuous time formulation : Crude-oil operations. *Industrial Engineering & Chemical Research*, 42 :3085–3097.
- Joly, M., Moro, L., and Pinto, J. (2002). Planning and scheduling for petroleum refineries using mathematical programming. *Brazilian Journal of Chemical Engineering*, 19 :207–228.
- Kelly, J. and Mann, J. (2002). Crude oil blend scheduling optimization : an application with multimillion dollar benefits. *Hydrocarbon Processing*, pages 47–53.
- Kelly, J. D. (2003). Smooth-and-dive accelerator, a pre-milp primal heuristic applied to scheduling. *Computer & Chemical Engineering*, 27 :873–832.
- Kouikoglou, V., Ioannidis, S., and Saharidis, G. K. (2005). Review of some queuing models for managing inventories, backorders, and quality jointly in stochastic manufacturing systems. *International Congress on Analysis of Manufacturing Systems, Samos - Greece*.
- Lee, H. and Feitzinger, E. (1995). Product configuration and postponement for supply chain efficiency. *Fourth Industrial Engineering Research Conference Proceeding, Institute of Industrial Engineers*, pages 43–48.
- Lee, H., Padmanabhan, V., and Whang, S. (1997). Information distortion in a supply chain : The bullwhip effect. *Management Science*, 43 (4) :546–548.
- Lee, H., Pinto, J., Grossmann, I., and Park, S. (1996). Mixed-integer programming model for refinery short term scheduling of crude oil unloading with inventory management. *Industrial and Engineering Chemistry Research*, 35 :1630–1641.
- Lee, Y. and Kim, S. (2000). Optimal production-distribution planning in supply chain management using a hybrid simulation-analytic approach. *Proceedings of the 2000 Simulation Conference*.
- Li, W., Hui, C.W. and Hua, B., and Zhongxuan, T. (2002). Scheduling crude oil unloading, storage and processing. *Industrial and Engineering Chemistry Research*, 40 :6723–6734.
- Liberopoulos, G. and Dallery, Y. (2000). A unified framework for pull control mechanisms in multi-stage manufacturing systems. *Annals of Operations Research*, 93 :325–355.
- Liberopoulos, G. and Dallery, Y. (2003). Comparative modelling of multi-stage production-inventory control policies with lot sizing. *International Journal of Production Research*, 41 :1273–1298.

- Manne, A. (1956). Scheduling of petroleum operations. *Harvard University Press*, , Cambridge.
- M.Minoux (1983). Programmation mathématique : Théorie et algorithmes. chapter 8. Collec-tion technique et scientifique des télécommunications DUNOD, Paris.
- Morton, I., Kamien, and Lode, L. (1990). Subcontracting, coordination, flexibility and pro-duction smoothing in aggregate planning. *Management Science*, 36(11).
- Mummolo, G. Ranaldo, S. and Iavagnilio, R. (1999). Integrating analytic and simulation models for fms loading and agv fleet sizing. *3rd International Conference on Engineering Desing and Automation*, pages 813–824.
- Nahmias, S. (1996). *Production and operations management*. Boston, MA : Irwin, 3rd edition.
- Neiro, S. and Pinto, J. (2002). A general modelling framework for the operational planning of petroleum supply chains. *Computers and Chemical Engineering*, 28 :871–896.
- Ozdamar, L. and Birbil, S. (1998). Hybrid heuristic for the capacited lot sizing and loading problem with setup times and overtime decisions. *European Journal of Operational Research*, 110 :525–547.
- Persson, J. and Gothe-Lundgren, M. (2005). Shipment planning at oil refineries using column generation and valid inequalities. *European Journal of Operational Research*, 163 :631–652.
- Pinto, J. M., Joly, M., and Moro, L. F. (2000). Planning and scheduling models for refinery operations. *Computers and Chemical Engineering*, 24 :2259–2276.
- Pyke, D. and Cohen, M. (1993). Performance characteristics of stochastic integrated production-distribution systems. *European Journal of Operational Research*, 68 (1) :23–48.
- Quesada, I. and Grossmann, I. (1995). Global optimization of bilinear process networks with multicomponent flows. *Computer & Chemical Engineering*, 19 :1219–1242.
- Reddy, P., Karimi, I., and Srinivasan, R. (2004a). A new continuous-time formulation for scheduling crude oil operations. *Chemical Engineering Science*, 59 :1325–1341.
- Reddy, P., Karimi, I., and Srinivasan, R. (2004b). A novel solution approach for optimizing crude oil operations. *A.I.Ch.E. Journal*, 50 :1177–1197.
- Riane, F. Artiba, A. and Iassinovski, S. (1999). Hybrid auto-adaptable simulated annealing based heuristic. *Computers & Industrial Engineering*, 37 :277–280.
- Saharidis, G. K., Dallery, Y., and Karaesmen, F. (2005). Centralized versus decentralized production planning in supply chains. *RAIRO*.
- Saharidis, G. K., Kouikoglou, V., and Dallery, Y. (2006a). Centralized versus decentralized optimisation of multi-stage manufactorig systems with subcontractoring. *Paper in progress*.
- Saharidis, G. K. and Minoux, M. (2006). Extention of benders decomposition method with multi-generation of valid inequalities. *Paper in progress*.

- Saharidis, G. K., Minoux, M., and Dallery, Y. (2006b). Centralized versus decentralized optimisation of multi-stage manufacturing systems with subcontracting. *Paper in progress*.
- Saharidis, G. K., Minoux, M., and Dallery, Y. (2006c). A general discrete-time formulation for the short-term scheduling of operations in a refinery. *Paper in progress*.
- Saharidis, G. K., Minoux, M., and Dallery, Y. (2006d). Optimization models for scheduling of crude oil in a refinery. *ROADEF-Lille*.
- Shah, N. (1996). Mathematical programming techniques for crudeoil scheduling. *Computers & Chemical Engineering*, 20 :S1227–1232.
- Shanthikumar, J. and Sargent, R. (1983). A unifying view of simulation/analytic models and modeling. *Operations Research*, 31 :1030–1052.
- Sherali, H. and Alameddine, A. (1992). A new reformulation-linearization technique for bilinear programming problems. *Journal Global Optimisation*, 2 :379–410.
- Susan, H. and Shanthikumar, J. G. (2001). Optimal expulsion control-a dual approach to admission control of an ordered-entry system. *Operations Recherche*, 41(6) :1137–1152.
- Symonds, G. (1955). Linear programming : The solution of refinery problems. *Esso Standard Oil Company*.
- Thomas, L. and McClain, J. (1993). *An overview of production planning, Handbooks in Operations Research and Management Science*, volume 4. North Holland.
- Towill, D., Naim, M., and Wikner, J. (1992). Industrial dynamic simulation models in the design of supply chains. *International Journal of Physical Distribution and Logistic Management*, 22 (5) :3–13.
- Towill, D. and Vecchio, A. (1994). The application of filter theory to the study of supply chain dynamics. *Production Planning and Control*, 5 (1) :82–96.
- Tzafestas, S. and Kapsiotis, G. (1994). Coordinating control of manufacturing/supply chains using multi-level techniques. *Computer Integrated Manufacturing Systems*, 7 (3) :206–212.
- Williams, J. (1981). Heuristic techniques for simultaneous scheduling of production and distribution in multi-echelon structures : Theory and empirical comparisons. *Management Science*, 27(3) :336–352.
- Williams, J. (1983). A hybrid algorithm for simultaneous scheduling of production and distribution in multi-echelon structures. *Management Science*, 29(1) :77–92.
- Yeralan, S. and Muth, E. (1987). A general model of a production line with intermediate buffer and station breakdown. *IIE Transactions*, 10 :130–139.
- Young, H. L. and Sook, H. K. (2002). Production-distribution planning in supply chain considering capacity constraints. *Computers & Industrial Engineering*, 43 :169–190.