SUPPLY CHAIN OPTIMIZATION IN THE PROCESS INDUSTRY. Methods and case-study of the glass industry

Nicolas Miegeville

ÉCOLE CENTRALE DES ARTS
ET MANUFACTURES
« ÉCOLE CENTRALE PARIS »

THÈSE

présentée par NICOLAS MIEGEVILLE
pour l'obtention du GRADE DE DOCTEUR

Spécialité : Génie industriel

Laboratoire d'accueil : Génie industriel

SUJET

SUPPLY CHAIN OPTIMIZATION IN THE PROCESS INDUSTRY
Methods and Case Study of the Glass Industry

soutenue le 21 septembre 2005
devant un jury composé de :

Luk VAN WASSENHOVE – Professeur – INSEAD	Président
Stéphane DAUZERE-PERES – Professeur – Ecole des Mines de Saint-Etienne	Rapporteur
Jérémie GALLIEN – Professeur – MIT Sloan School of Management	Rapporteur
Jacques-Olivier MOUSSAFIR – Chef de groupe – Saint-Gobain Recherche	Examinateur
Yves DALLERY – Professeur – Ecole Centrale Paris	Directeur de thèse
Michel MINOUX – Professeur – Université Paris VI	Co-directeur de thèse

2005 - 28

École Centrale Paris
Grande Voie des Vignes
92295 CHÂTENAY-MALABRY

SUPPLY CHAIN OPTIMIZATION IN THE PROCESS INDUSTRY

Methods and Case Study of the Glass Industry

Nicolas Miègeville

Remerciements

Pour commencer, je remercie sincèrement Luk Van Wassenhove, Stéphane Dauzére-Pérès et Jérémie Gallien d'avoir accepté de faire partie de mon jury de thèse malgré leurs emplois du temps chargés. J'en suis très honoré.

Cette thèse, fruit de trois années de travail intense, constitue avant tout un grand défi personnel. Même si elle ne représente que mon infime contribution à la production scientifique mondiale, elle a beaucoup de valeur sentimentale. Tout au long de cette aventure, j'ai vécu l'ivresse des sommets comme la fatigue éprouvante des grands cols. Une solide motivation personnelle m'a permis d'arriver jusqu'à ces quelques lignes, derniers mots avant de tourner une page importante de ma vie. C'est avec beaucoup de reconnaissance que je me retourne à présent pour saluer ceux qui ont peuplé la route sinueuse d'encouragements et de confiance, ceux sans qui rien de tout cela n'aurait été possible.

Je pense tout d'abord à Yves Dallery, mon directeur de thèse bien aimé, qui a fait germer l'idée d'une thèse CIFRE dans mon esprit de jeune centralien un peu perdu à la sortie de l'Ecole. Bientôt rejoint par Michel Minoux, ils ont toujours su trouver les mots pour m'encourager et me guider. Soudés par un grand plaisir à travailler ensemble, nous avons su persuader nos partenaires industriels de nous faire confiance en nous laissant travailler sur des problématiques parfois très sensibles.

Je salue chaleureusement les dirigeants de Saint-Gobain qui ont pris le temps de nous présenter leurs problématiques, de nous écouter et de nous guider dans nos travaux : je pense naturellement à Françoise Bitchakdjian et Didier Gautron, en charge respectivement de la planification centrale des lignes de production et de la direction logistique internationale de Saint-Gobain Glass.

Je remercie de même mes collègues directs dans le développement et la mise au point d'outils d'optimisations exploitables : sans François Geoffray, Philippe Pilon, Tomasz Wojtczak, Marc Germain et Remy La, aucun de mes travaux n'aurait pu être appliqués en si peu de temps. Un grand bravo à toute l'équipe pour cette collaboration sympathique et efficace.

Un grand merci également à ceux qui sont à l'origine de cette collaboration fructueuse : Pierre Jeanvoine (responsable R&D Saint-Gobain Glass), Catherine Langlais et Paul-Henri Guering (respectivement directrice et chef de service à Saint-Gobain Recherche).

Je tiens aujourd'hui à souligner le grand plaisir que j'ai éprouvé à travailler à Saint-Gobain Recherche, dans le groupe Mathématiques Appliquées du service

Capteurs et Mesures. En premier lieu, les conseils toujours pertinents de Jacques-Olivier Moussafir (chef de groupe) ainsi que la confiance et le soutien inconditionnels de Paul-Henri Guering (chef de service) ont été les conditions nécessaires à la réalisation de mes travaux de recherche. Mais je tiens aussi à souligner l'excellente ambiance du service et la grande complicité qui y règne entre tous ses membres. Les longues discussions passionnées et passionnantes qui ont ponctué ces trois années ont toujours été une grande bouffée d'oxygène dans un travail de fond long et difficile. Un grand merci à Ludivine, Hélène, Aurélia, Anca, Stéphanie, David, Michele, les Oliviers, Patrick, Hugues, Jean, Jingwei, Jean-Philippe, Alessandro, Clément, Jacques-Olivier et Paul. Je remercie particulièrement mes ex-collègues de bureau avec qui j'ai beaucoup échangé : Cyril, Guillaume et Michael.

Merci également à tous les membres du Laboratoire de Génie Industriel de l'Ecole Centrale de Paris qui ont agrémenté mes visites de leur bonne humeur et de leur ouverture d'esprit, avec une pensée particulière pour Anne, Sylvie, Asma, Evren, Jean-Philippe, Hakim, Zied et Jean-Claude.

Cette thèse aura aussi été l'occasion d'un périple extraordinaire aux Etats-Unis, au coeur d'une université qui m'a toujours fait rêver, le Massachusetts Institute of Technology. Je remercie chaleureusement Jeremie Gallien, Professeur à la Sloan School of Management, qui m'a invité à rejoindre son équipe pendant plusieurs mois. Un grand merci également à Catherine Langlais et Jean-Paul Straetmans (directrice et secrétaire général de Saint-Gobain Recherche) qui ont soutenu financièrement ce projet atypique. Pour finir, j'ai une agréable pensée envers ceux qui ont agrémenté mon séjour outre-atlantique de leur acceuil chaleureux puis de leur amitié : Théophane, Yann, Felipe, Pierre-Alain, Grégoire et Adrien à Boston, David, Paul et Maurice à New-York, et bien sûr mon colocataire hollandais Peter. Merci également à Saif Benjaafar de m'avoir invité à donner une conférence sur mes travaux à Minneapolis : ce fut une expérience passionnante.

Il est temps désormais de se consacrer à l'essentiel : la famille. Comment aurais-je pu surmonter toutes les difficultés rencontrées sans le soutien inconditionnel et illimité de mes parents Christiane et René et de mon frère Anthony, sans oublier mes oncles, tantes et cousins -avec une attention particulière pour Roland et Monique, mon oncle étant jusqu'à ce jour le seul docteur de la famille- ? Quelque soit la distance qui nous séparait, j'ai toujours apprécié trouver le réconfort dont j'avais besoin. Evidemment, je garde le meilleur pour la fin : mes plus vifs remerciements vont naturellement à celle qui éclaire ma vie depuis sept ans passés (et ce n'est qu'un début !), celle en qui je puise ma force, mon dynamisme et ma joie de vivre, en bref, ma moitié : Noémie. Elle seule connaît la version "off" de cette thèse, et elle peut être aussi fière que moi du résultat !

Pour finir sur une note humoristique, j'adresse enfin un grand merci à tous ceux qui m'ont expliqué que faire une thèse ne m'apporterait rien -et ils sont nombreux !-. Mon esprit de contradiction m'aura poussé jusqu'au bout à les contredire, et j'en suis aujourd'hui très heureux !

Nicolas Miègeville

Résumé

L'importance croissante que le client accorde à la manière dont une entreprise satisfait sa demande bouleverse les fondements des organisations anciennement pensées sous l'angle de la production. Phénomène tout à fait perceptible dans un grand groupe industriel comme Saint-Gobain, à forte culture ingénieur, cette prise de conscience donne un nouvel élan aux métiers transversaux focalisés à la fois sur l'optimisation du schéma industriel et de la chaîne logistique. Cette thèse est une illustration de cette évolution : l'intérêt porté aux problèmes d'optimisation des systèmes industriels et logistiques est relativement récent à Saint-Gobain Recherche. Nous nous sommes intéressés dans nos travaux à différents problèmes industriels complémentaires rencontrés chez Saint-Gobain Glass, leader de la production de verre plat en Europe. Nous avons apporté des solutions mettant en lumière l'interdépendance de différentes décisions à des problèmes industriels complexes, avec un souci constant de produire des outils d'aide à la décision utiles et appréciés.

Après un avant-propos rappelant le sens de notre démarche, nous découvrirons dans le chapitre 1 le contexte industriel qui a motivé notre recherche. Nous présentons les métiers du groupe - produire, transformer et distribuer du verre plat - et les différents niveaux de décision que nous avons décidé d'aborder. Les chapitres suivants présentent les problèmes d'optimisation que nous avons identifiés et qui nous sont apparus comme clés.

Nous abordons dans le chapitre 2 un modèle permettant de déterminer les dimensions des produits standards. L'intégration verticale du groupe permet l'étude du meilleur compromis entre les chutes de verre tout au long de la chaîne logistique et le nombre de références à gérer. La suite de la thèse tend à aboutir à une modélisation complète du schéma industriel et logistique et fait l'objet du chapitre 6. Pour cela, nous traitons les questions de localisation d'installations logistiques (chapitre 3) et de modélisation des processus de production : le chapitre 4 présente notre modèle et l'illustre avec la production de verre plat, tandis que le chapitre 5 présente un travail complémentaire permettant de l'appliquer aux lignes de transformation. Finalement, nous intégrons dans le chapitre 6 tous ces travaux dans un modèle linéaire en nombres entiers.

Fruit d'une véritable collaboration entre chercheurs et industriels, ce travail présente un modèle générique déterministe d'optimisation de la chaîne logistique appliqué avec succès à l'industrie du verre. De nombreuses perspectives dignes d'intérêt sont imaginables, autant théoriques que pratiques.

Mots clés: Schéma industriel et logistique, Programmation Mathématique, Planification de production.

Abstract

Nowadays, a highly competitive environment makes of the service level impact a fundamental element for formerly production oriented companies. Global supply chain thinking gives a new impetus to transversal missions such as logistics management. This thesis is nothing but an illustration of this new philosophy within the Saint-Gobain group, which has decided to create an operations research group (based in Saint-Gobain Recherche) to identify and solve high potential optimization problems. Our research has been full granted by the Saint-Gobain Glass company, the European leader of flat glass production. We worked on several complementary subjects on which we developed original solutions with successful industrial final applications.

After a brief overview of our approach, we discover in chapter 1 a synthesis of the Saint-Gobain Glass business -producing, adding value and delivering flat glass- and the various decisions that make the supply chain management a very complex task. Following chapters present optimization problems that we identified as key ones.

First of all, we define in chapter 2 a model to handle the yearly standard product determination, based on the economic trade-off between the glass loss along the vertically integrated supply chain and the management cost of various references. Other chapters aim at introducing an original global method for modelling complex industrial supply chains, fully presented in chapter 6. To do so, we study successively the definition and localization of facilities in chapter 3 and the modelling of production processes: chapter 4 introduces our production planning model and illustrates it on the flat glass production, whereas chapter 5 presents the complementary work required to apply it to transformation lines. Finally, chapter 6 deals with the synthesis of all these works into a unique mixed integer linear program.

Our research is the result of a successful collaboration between academics and industrials. We have developed an original deterministic model that captures various industrial supply chains, and we applied it to the float glass manufacturing industry, identifying huge savings. Numerous outlooks of both theoretical and practical interest are possible.

Keywords: Integrated production-inventory-distribution systems, Supply Chain Design, Mathematical Programming, Production planning.

Avant-Propos

Le plus important avant de commencer la lecture de ce travail est de le situer dans son contexte. Avant toute chose, quel était l'intérêt de l'entreprise de travailler en étroite collaboration avec une équipe de chercheurs de l'École Centrale ? Pour synthétiser en quatre mots le fameux roman initiatique d'un dirigeant d'entreprise [Gol92], l'unique but de l'entreprise est simple et sans équivoque : gagner de l'argent. Pour ce faire, elle met en œuvre un ensemble de ressources pour satisfaire les besoins de ses clients. Ainsi, cette thèse est avant tout une illustration de l'intérêt que les outils théoriques développés en recherche opérationnelle (ou en management des opérations) présente pour de nombreuses applications de la vie économique. Développer des modèles mathématiques originaux peut ainsi permettre d'améliorer la compréhension de phénomènes économiques tout en améliorant leur maîtrise.

Les entreprises manufacturières issues des différentes révolutions industrielles furent longtemps centrées sur leurs compétences techniques et la qualité de leurs produits. Les notions de service client, de management stratégique des relations clients ou encore de management de la chaîne logistique ont vu le jour depuis quelques décennies. De nombreuses réussites d'entreprises dans ce domaine (Toyota, Dell, Amazon, etc.) ont mis en lumière l'avantage concurrentiel décisif dont peut bénéficier une entreprise qui innove dans sa gestion de la chaîne logistique. Repenser sa chaîne logistique peut par exemple amener l'entreprise à modifier aussi bien son schéma industriel et logistique que sa politique commerciale.

L'étude de modèles quantitatifs caractérisant différents compromis économiques (d'où la notion d'optimisation) permet de mieux comprendre et de faire progresser la chaîne logistique. Ce domaine de recherche a été très actif depuis les années 1970 autant sur le plan pratique que théorique. Tandis qu'une partie des travaux motivés par des applications permet de faciliter la prise de décision ou encore la définition de la stratégie des entreprises étudiées, les travaux théoriques posent des problèmes à la frontière de la recherche opérationnelle, de la théorie des jeux, et de la microéconomie ([SLWS04]). Appartenant à la première catégorie, nos travaux de recherche ont visé à répondre de manière pertinente et originale à des attentes de nos partenaires industriels.

Même si nous nous limitons dans cette première étape de partenariat avec un industriel à des modèles déterministes, une multitude de travaux prenant en compte l'incertitude des phénomènes réels met aujourd'hui en lumière de nouvelles règles de gestion. Notre travail présente donc de nombreuses perspectives dans cette voie. A titre indicatif, l'effort bibliographique de la thèse [Gay04] donne un bon aperçu des travaux de recherche sur la catégorisation des clients et la création de couples prix/délai, etc. De même les travaux de gestion dynamique des prix couplée à la gestion des stocks sont très bien synthétisés par les auteurs de [CSSLS04].

Adoptons un instant le regard des dirigeants de l'entreprise. Considérons que la satisfaction du client est un compromis entre d'une part le prix du produit et d'autre part sa qualité, la durée et la fiabilité du délai de livraison annoncé (ou la disponibilité en magasin), la flexibilité correspondante (facilités de paiement, achat et livraison groupés avec d'autres produits, taille du lot, mise à jour de la date de livraison, choix entre différentes options, etc.), etc. Il s'avère que sur un marché concurrentiel le prix est souvent une variable exogène, c'est à dire une contrainte imposée par le marché. Le coût global des opérations effectuées pour servir les clients est donc directement corrélé à la marge désirée : achat des matières premières, production, transport, stockage, etc.

Avant de chercher à optimiser sa chaîne logistique, l'entreprise doit en premier lieu définir sa vision stratégique, le marché visé et le service ou le bien qu'elle souhaite vendre aux clients correspondants ([CM01c]). Dans un second temps, elle cherchera à respecter sa politique au moindre coût. Si l'entreprise respecte ses engagements, elle pourra les communiquer à ses clients sans hésitation et jouir d'une excellente réputation, amorçant ainsi une spirale vertueuse.

L'optimisation de la chaîne logistique, et nous verrons dans la thèse qu'il est intéressant de penser cette expression en termes de maillons inter-connectés, prend alors tout son sens : définir le meilleur schéma possible, puis déterminer les flux et l'organisation des processus en minimisant l'ensemble des coûts variables correspondants.

Il apparaît clairement qu'une problématique si complexe ne peut être résolue par un simple travail de thèse. Cependant, nous présentons dans les chapitres qui suivent comment nous avons identifiés, traités et résolus des problèmes originaux au coeur du processus de décision de la chaîne logistique.

Notre plus grande satisfaction n'est autre que le sentiment d'avoir été utiles aux opérationnels de Saint-Gobain Glass. Une grande richesse est née de la rencontre de points de vue fondamentalement différents. Nous attirons l'attention du lecteur sur une des principales difficultés que nous avons surmontées pendant nos recherches, et que nous avions a priori sous-estimée. Il est en pratique relativement complexe d'aller identifier des problèmes d'optimisation à résoudre, de convaincre les intervenants industriels de l'intérêt de la démarche, et enfin de dresser un cahier des charges pertinent.

Contents

Chapter 1

Introduction to the industrial context

In this first chapter, we introduce the main issues that supply chain managers of Saint-Gobain Glass are facing before motivating the forthcoming chapters.

The group is the European leader for both the production and the transformation of flat glass. The group core know-how lies both in the production processes and in the logistics of flat glass. This introduction aims at explaining and summarizing the industrial context of our study. We focus on the European zone, where many different factories produce a wide product line, offered and delivered to various customers by a reliable supply chain.

We present the different product families (§ 1.2) as well as the production tools (§ 1.3) of flat glass. Then, we illustrate in (§ 1.4) the difficulties of this supply chain by a presentation of distribution issues and the corresponding present organization. Finally, a short summary (§ 1.6) of the history and of the competition on the market highlights the strategy of the firm. This raises the question of the ways to achieve the strategic fit through the supply chain.

To analyse the supply chain of Saint-Gobain Glass, we have borrowed from two classical references, [CM01c] and [SLKSL03]. Let us recall the definition of the supply chain according to [CM01c] (Part 1, Chapter 1).

Definition 1 *A supply chain consists of all parties involved, directly or indirectly, in fulfilling a customer request. The supply chain not only includes the manufacturer and suppliers, but also transporters, warehouses, retailers, and customers themselves. Within each organization, such as a manufacturer, the supply chain includes all functions involved in receiving and filling a customer request. These functions include, but are not limited to, new product development, marketing, operations, distribution, finance, and customer service.*

In section (§ 1.7) we highlight the complexity of the Supply Chain Management in a real industrial context. Based on a classical decision classification into three main groups (tactical, strategical and operational), we list possible decisions that have to be tackled in the flat glass industry. Finally, we introduce in (§ 1.8) the structure of the thesis.

1.1 Interest of the Supply Chain Management

As explained in [KG03], Supply Chain Management (SCM) has been a very visible and influential research topic in the field of operations research (OR) over the course of the last decade of the twentieth century, by providing either ideas for new models or applications for existing ones. Authors try to summarize what are the main business trends that created SCM.

First of all, authors recall the success (in terms of its adoption by global companies) of the core-competency strategy, based on the thought developed in [PH90]: a number of companies have achieved significantly better results than their competitors by focusing on only a few competencies, so-called core competencies, and by out-sourcing other non-core activities to companies that have a core competence on those activities.

Some companies (such as Hewlett-Packard (HP) or Dell) recognized that SCM was one of their core competencies. HP[a] outsourced manufacturing and focused on research and development as well as marketing and sales; Dell decided to sell direct to the customer by using the Internet as its marketing and sales channel. Both examples are showcases of "world-class" SCM. HP introduced the concept of postponement, implying that product diversity is created as close as possible to the consumer, thereby allowing for efficiencies upstream in the supply chain, while Dell shows the potential for operating low-inventory, high-flexibility and customized-product supply chains ; they both underline the strategic trade-off between customer service, market diversity and supply chain flexibility.

Authors in [KG03] consider that the adoption of new practices such as outsourcing the final assembly[b] of a product in regions where the labour cost is low has created more and more complex distribution patterns, and hence more and more complex supply chain planning and control activities. The same way, outsourcing the physical distribution function has stimulated the emergence of third party logistics (3PL) service providers, specialized in optimizing transportation of customers by capturing economies of scale.

Finally, another important element lies in the experimental research that revealed that demand variations amplify from link to link going upstream in the supply chain (from customers to raw materials). The communication about this phenomenon (so-called the Bullwhip effect ; [For58],[LPW97]) has increased mutual understanding across different (inter-function and inter-company) actors of the supply chain, leading to a global improvement of the overall knowledge base on SCM.

Last but not least, we shall see in section (§ 1.5) that the emergence of Information Technology systems allows nowadays companies to operate efficiently.

[a] [LB93] and [LB95] discuss the main ideas behind the HP approach

[b] in businesses in which the labour cost is greater than inbound and outbound transportation ones

1.2 Products of Saint-Gobain Glass

The float process for manufacturing glass, a Pilkington development announced in 1959, is recognized as the world standard for flat glass production. Since 1962, all glass manufacturers have licensed the float process from Pilkington, including Saint-Gobain Glass.

Today a new float line costs around 80 millions of euros, and the operating life of a furnace is from nine to twelves years. At the end of this cycle, the float must be rebuilt and relined with new refractory materials. This rebuilt costs around 10 millions of euros and takes three to four months to complete.

Glass is globally made of the fusion of silica sand and other components (such as limestone, soda ash, dolomite, cullet[c]) melted in a furnace. A continuous ribbon of molten glass mixture floats from the tank over a bath of molten tin where its speed and temperature are computer monitored and controlled to give the finished glass its proper thickness and characteristics. There are three separated temperature zones in the bath. The first zone is the healing zone where irregularities in the glass surface are melted out and both surface become flat and parallel. The fire polishing zone is where the glass acquires its brilliant surfaces. The final zone is the cooling zone, where the glass cools sufficiently for it to touch the rollers without spoiling the fire-polished surfaces.

The ribbon of glass then moves from the bath onto the annealing zone where precise gradual cooling relieves stresses in the glass. Following cooling and a series of quality-control inspections, the continuous ribbon enters the cut area where the glass is cut into sizes for storage, distribution, or fabrication into value-added products. Each sheet of glass may thus be either sold or coated, tempered as well as laminated on dedicated production lines.

In a nutshell, the float glass industrial process is continuous: at first sight, a plant produces as long as its furnace is able to. We can imagine a plant as producing a

[c]which is crushed recycled glass

glass ribbon that has to be steadily cut into pieces which are then stacked up on a dedicated support. On the contrary, rolled glass is produced on dedicated lines.

Figure (1.1) summarizes the different possible steps of production and transformation of flat glass. It underlines that products may be bought by customers whatever their state in the process.

Finished products of Saint-Gobain Glass are parallelepiped glass stacks, characterized by many attributes of different kinds:

- One stack is made of several similar sheets, characterized by:
  - A colour, which depends on the composition
  - A thickness and a brute width, which depend on technical parameters used during the production before cooling the glass ribbon.
  - A quality, which refers to conformance to product specifications. For instance, required quality for automotive applications is higher than the one for building markets: the distribution of optical defaults[d] on the glass is more restrictive.
  - Net dimensions: length and width, which depend on the cut step. We can note here that the final dimensions are either the on-line ones or those obtained after another cut step on a specific cutting line.
  - An additional one side metal coating, which is laid:
    - Either during the float process: we call it a hard-coat. When this on-line pyrolitically-coated product is being produced, a chemical vapor is released in the float bath over the semi-molten surface of the ribbon. The reaction of the vapor with the glass surface forms the reflective coating.
    - Or on a specific coating production line, referred to as a soft-coat.

[d]optical distorsions are watched by an on-line laser system

- various states:
  - Laminated or not.
  - Tempered or not.
  - Printed or not.

- The size of the stack, depending on the number of pieces. We must notice the fact that *a stack cannot be easily divided into smaller ones*, and that the bigger the stack:
  - The lower the handling cost (the time to manipulate a stack doesn't depend on its weight).
  - The more powerful the required handling tool.

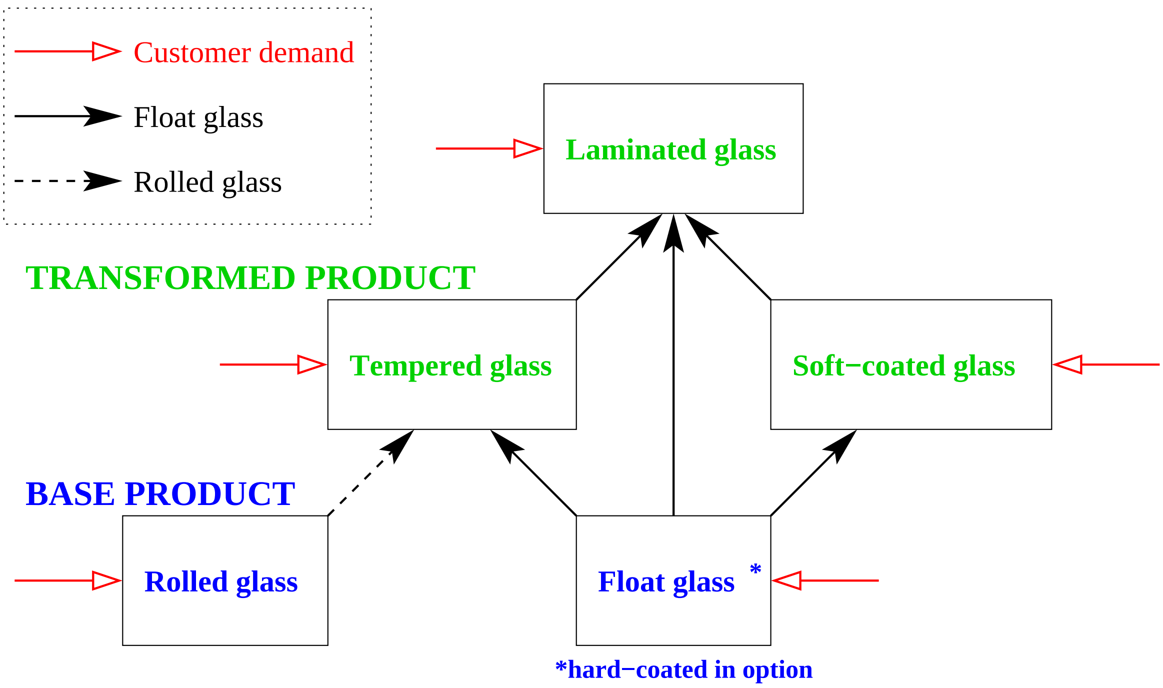


Figure 1.1: Steps of production and transformation of flat glass

1.3 Production tools of Saint-Gobain Glass

For a good understanding of the Saint-Gobain Glass supply chain, we need to describe different production processes. We introduce an assumption under which we always work in the following.

Assumption 2 *The replenishment of raw materials for production process of base products (float process) is perfect, in the sense that raw materials can be considered as always available.*

This assumption is realistic to the extent that except energy and cullet, raw materials are cheap and the storage capacity is huge on each plant site. Energy is provided through vendor managed inventories and we assume so far that cullet is self-produced by the line in sufficient quantities. We will discuss this assumption in section (§6.7.4).

First of all, the float process[e] produces basic flat glass, each sheet having a colour, a thickness, a quality and given dimensions. We can consider each float plant as:

- Using a **continuous process on a unique production line**. We consider in the following that each line satisfies assumption 2. We insist on the fact that the smoothness of the production is a constraint: **we assume that production can not be stopped**, due to a huge shut down cost ; in addition, the capacity of the line has very limited flexibility, with a maximum ±10% variability. As said above, the operating life of a plant is several years.
- Being **technically forced to organize production by dividing time into periods (referred to as campaigns)**, each period (campaign) being characterized by the choice of a colour, a thickness, etc. Stability constraints of the process must be satisfied. For instance, we argue in (§ 6.7) that the single

[e] a detailed description is given in section (4.2.1) as an illustration of our methodology of modeling production processes

most important consideration in planning flat glass operations is the transition schedule, that is, the scheduling of production time for colour campaigns. Typically, changeovers between two different colours can take up to several days, whereas those between distinct thickness values only take several hours. **Glass produced during a changeover is lost**[f]. Because building this process requires major capital investment, it is crucial that it constantly produces high volumes of output at the correct level of quality. To achieve this goal, it has a high degree of automation, is operated continuously, produce one product at a time, and maintenance is usually scheduled during changeovers. As product changeovers result in long downtimes and considerable set-up costs, products are often produced in long campaigns and are inventoried.

- Having **a huge capacity for on-line cutting operations, but limited capacity on optional specific cutting lines** (which require some additional workforce). Maintaining a high percentage of on-line cutting is another crucial factor in operating a flat glass plant. Standard sized glass sheets are cut on-line at the end of the line as the glass is being produced. However, glass that is cut to customer order may be cut either on-line or off-line after a period of storage. **On-line cutting yields are higher** for various reasons. However, some level of off-line cutting is mandatory. Otherwise, it would be impossible (considering a unique plant) to fulfill an order of non standard dimensions in a colour different from the one being in production. Since customer orders for cut-sizes are rarely known more than a month in advance, most orders for cut-size products, other than clear (which is the dominating colour), must be cut off-line.

- Packaging glass by stacks of homogeneous sheets, which are laid on trestles at the end of any production process.

[f] it is broken and melted again in the oven

- Having a wide range of products, due to the important number of colours, thickness values, and specially dimensions. Globally, most of the features can also be produced by another factory. Nevertheless, **each factory has some particular skills to produce special products**.

Secondly, we mention that soft-coating production lines (which are still less numerous than float plants) have been located on different existing float plant sites. Each soft-coating line can be considered as:

- Adding valuable metal coatings on flat glass sheets.

- Producing by campaigns of transformations (see chapter 5). Given the type of coated metals, we can coat any type of flat glass stacks (and we can exceptionally change the size of the stack on purpose).

- Being more flexible than a float line, to the extent that the process can be shut down at a reasonable cost and the speed of the line is therefore easier to control.

In the same way, laminating and annealing production lines are also located on several float plant sites. We notice that these production processes are much more flexible than the previous ones. Lines can be stopped, and changeover times between different products are insignificant.

- Laminated glass is a kind of "hamburger" structured safety glass. It is firstly laminated with a Polyvinyl Butyral (PVM) film between at least two glass sheets by a special equipment and is then preheated and prepressed before entering an autoclave for permanent heat pressing and forming.

- Concerning tempered glass, the heat treating process produces highly desirable conditions of induced stress which result in additional strength, resistance to thermal stress, and impact resistance.

Finally, rolled glass is produced in few plants by a rolling process[g]. The process needs smoothness and big lot sizes.

Reduction in downtimes (especially on float plants, see 6.7.4) and in costs associated with campaign switch-overs, holding, and transportation costs are critical at Saint-Gobain Glass. This is because their products are typically commodities with market-defined prices, and profits can be increased only by reducing costs and by increasing output by minimizing downtimes.

1.4 The distribution of flat glass

Due to the production process characteristics, **almost all of the products are made to stock**. In this paragraph we briefly introduce the transportation and the inventory issues.

Transport issues

First of all, how is flat glass transported? For intercontinental delivery, sea shipping is the cheapest way of conveyance, in spite of high lead times and additional handling and packaging costs. Otherwise, land transport is both possible by train and by truck. Train can only be used for huge quantities and lead times are important. For short distance deliveries and small quantities, trucks are the most flexible means of transportation.

An **inloader** is a special truck designed specifically to transport glass sheets of big dimensions (called PLF). The unusual point is that inloaders can carry nothing but glass (PLF or smaller dimensions DLF): it loads directly the glass trestle (see Figure (1.2)). Moreover, PLF can only be transported by **inloaders**. Figures (1.3) to (1.6) describe how the inloader does load the glass trestle, whereas Figure (**??**) shows a loaded inloader ready to go. Naturally, the inloader has been designed

[g]which was used widely before the float process invention

so that the loading and unloading times are minimized for dedicated glass trestles. However, **the inloader fleet is limited**, and peak periods may be critical to find an on-hand truck.

Nowadays, in the case of internal as well as external transportation, the payment of each delivery covers the round-trip kilometers. Therefore, **transportation cost is higher than traditionally**. This issue is tackled by maximizing the **reloading rate** on deliveries: if an inloader is unloaded at a place nearby its future loading place, we minimize the empty truck[h] kilometers. This is the reason why we consider the transportation cost dependent on the flow type: **transfers between plants or replenishment of any logistic platform may be cheaper than any customer delivery**. In the first case, the reloading rate may indeed ideally be around 100%, whereas it may be up to 75% in the second one.

Finally, given the common cost structure of basic products, **transportation costs appear to be a key factor of an efficient supply chain in the glass business**.

Figure 1.2: Trestle of glass ready to be loaded by an inloader

[h]useless but paid

Figure 1.3: Inloader loading a trestle, step 1

Figure 1.4: Inloader loading a trestle, step 2

Figure 1.5: Inloader loading a trestle, step 3

Figure 1.6: Inloader loading a trestle, step 4

Figure 1.7: Inloader ready to go

Glass warehouses and inventory management

During their transportation, glass stacks are carried on dedicated trestles which are expensive and available only in limited quantity. We could imagine a glass warehouse as a big inventory of loaded trestles, but this would require too much space. Each glass stack is stacked up on bigger stacks with higher compactness. A **glass warehouse** can be viewed as several big glass stacks, made of smaller stacks. Given that obsolescence of glass increases quickly, particularly in conditions such as humid atmosphere, glass storage requires closed buildings. The Figure (1.8) presents the global aspect of a glass warehouse.

Figure 1.8: Warehouse of glass in a Saint-Gobain Glass plant

Loading and unloading operations of glass stacks from trestle to inventory and vice versa requires dedicated carrying machines with trained hand workers (Figure (1.9)). Due to the risk of human manipulation, every move of glass is time-consuming. Therefore, **glass handling costs are relatively high**, and big inventory facilities require high investment in carrying materials.

Figure 1.9: Workers carrying a glass stack in a Warehouse

Taking into account the description of float plants we gave in section (§ 1.3), it appears that an important area of concern is inventory management. Because plants take so long to make colour transitions, their inventory can be very large (up to 60000 Tons). To manage inventory successfully, the plants must balance the risks of obsolescence against those of stock outs. So far, managers specify minimum safety-stock levels in terms of equivalent days of sales. Furthermore, to cover demand for products of a particular colour during the interval between successive occasions when the float is producing that colour, plants maintain minimum cycle stocks. For a given product, this cycle stock must be sufficient to cover all demand for that product plus all demand for products cut off-line from the given product.

Saint-Gobain Glass transportation and ordering policy

To deal with this transportation issue, the firm owns dedicated trucks and tries to minimize costs. One obvious way to reduce it lies in **optimization of the vehicle routes**, maximizing the average carried volume per kilometre while satisfying specific transportation constraints. At the same time, efforts are made to tend to achieve **transportation planning smoothness** to avoid demand peaks which can not be quickly fulfilled.

In addition, the company policy forces the use of **full truckload delivery** in order to benefit as much as possible from **economies of scale**. Nowadays, an order equals a full truck and all the Enterprise Resource Planning[i] system has been built on this simple principle.

To have a more precise idea of the impact of the full truckload delivery rule, we studied past year data. To understand present flows of products we define notions of product, order and mixed order as well as mixed origin delivery.

Definition 3 ***A product*** *is an homogeneous*[j] *stack of flat glass sheets.*

Two various sizes of stack made of similar glass sheets correspond to two references. Each product may be produced in at least one plant.

An order is triggered by a customer. It is made of a set of at least one product. Nowadays, it corresponds to a full truckload and is thus sent from one unique shipping plant.

Definition 4 *A* ***mixed order*** *contains at least two different products.*

A delivery is triggered by an order of a customer. Each delivery has nowadays a unique shipping plant. To a mixed order corresponds a delivery of at least two different products. It may be either a mixed delivery or a mixed origin delivery.

Definition 5 *A delivery is a* ***mixed delivery*** *when every requested product has been produced at the shipping plant.*

Definition 6 *A delivery is* ***a mixed origin delivery*** *whenever it contains at least one product whose production plant is not the shipping plant.*

Due to the large number of references as well as the financial cost of inventory (increased by obsolescence of some high-value products), most customers try to minimize their inventory. Given that orders are forced to be full truckload we understand the interest of ordering mixed orders.

[i] Saint-Gobain Glass uses SAP products

[j] i.e. same properties, same dimensions

Present organization

In this study, we assume that a policy overlooking the possibility of mixed orders can not be forced in a strong competitive environment.

Assumption 7 ***We assume that no limiting rules can be imposed on mixed orders****, i.e. we do not force a maximal number of various products in each order.*

Therefore, an order may count as many products as possible. For instance, under the full truckload assumption, given that a truck is full with one trestle and knowing that a trestle is loaded with eight stacks of glass, each order may count at most eight different products.

When a customer orders a truck, the seller (after a quick check of both product availability and location) proposes him (or her) a delivery date. After acceptance, this date becomes the order due date. **Service level is then measured in terms of proportion of orders fulfilled (completed delivery) before the due date**.

So far, there is no commitment on lead time by marketing teams, and prices include transportation. We point out that **if an order is too complicated or can not be fulfilled, the seller may deny it**. The impact of such phenomena can not be taken into account by the present service level measure. However, in addition to revenue reduction, the firm may incur a **loss of customer goodwill** that would result in reduced future sales. This **lost sales cost** is very difficult to quantify as it represents the future unknown impact from present poor service.

Nowadays, each plant is able to send mixed orders which are only made of its own produced goods. It is more complicated to fulfil mixed origin deliveries[k] because there is neither a clear policy of replenishment between plants nor dedicated logistic platforms.

[k] that require products made by different plants

However, **transfers** between plants are used to satisfy some mixed orders. To characterize flows, we define the notions of transfer, and both direct and indirect flows.

Definition 8 ***A transfer*** *is a flow of products between two different plants.*

Definition 9 ***A direct flow*** *is a flow of product sent directly from the producer to the customer. The sender is the producer.*

Definition 10 ***An indirect flow*** *is a flow of product sent from a sender which is not the producer to the customer. It is the result of a flow between the producer and the sender and a flow between the sender and the customer.*

As explained in section (§1.4), it makes sense to consider a discount of 20% on inter-plant transportation costs. Thus, transfers may be the best way to deal with the issue of difficult mixed orders. We present real flows as an example of the power of the Enterprise Resource Planning system implemented by Saint-Gobain Glass in section 1.5.

1.5 On-hand Data: the power of ERP systems

As emphasized by the authors in [KG03], the company-wide implementation of so-called Enterprise Resource Planning (ERP) system across the group Saint-Gobain provided us all required data for our research. Even if the mixture of transactional system and decision-support system makes it hard to define an ERP system in a rigorous manner, we consider here that it is a system that both

- enables the execution of all business processes, such as processing, invoicing, transportation, warehouse picking, work order release and purchase order release.
- supports various decision-making processes, such as inventory management, production planning, forecasting, etc.

We recommend for a brief history of ERP systems the article [GS04]. The authors also summarize the motivation of firms which have spent tremendous amount of money in the implementation, and recall famous horror stories that have been published during the nineties: FoxMeyer Drug Company, Dell, Hershey Foods, Intel, etc.

However, the ERP experience at Saint-Gobain Glass has confirmed the idea of Kok and Graves when they claim in [KG03] that ERP systems are a condition sine qua non, a prerequisite for implementation of intra- and inter-company Supply Chain Management. During all our research we have obtained inputs by downloading from Information Technology (IT) backbone system. User of our models then uploaded results again, either manually or using an IT interface.

To highlight the great interest of ERP, let us focus here on the glass distribution issue. By extracting yearly[a] past data we were able to track all glass moves and thus to compute indicators about mixed trucks (described in part 1.4). Based on past yearly[a] sales, we found globally that:

- **47% of delivered trucks contain more than two different products**, whatever the origin of the product. Figure (1.10) shows that the complexity of the order decreases exponentially in the number of requested products. Half of the mixed trucks count indeed only two products and 95% of the global amount of deliveries do not count more than 4 products.

- **22% of delivered trucks contain at least one mixed origin product**, but 75% of mixed origin trucks count only one mixed origin product (see Figure (1.11)).

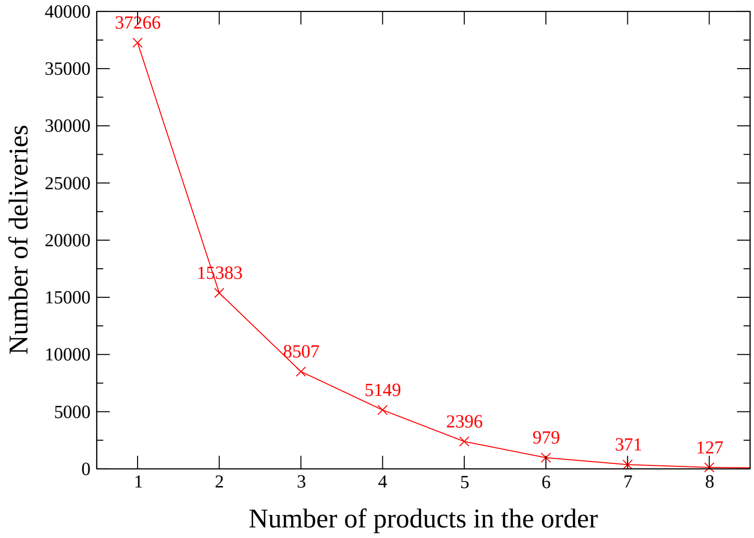


Figure 1.10: Mixed order complexity of yearly[a] past data

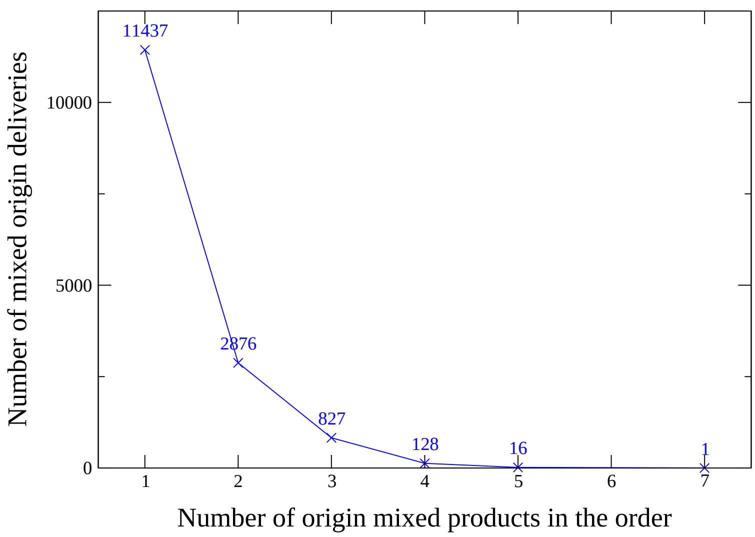


Figure 1.11: Details of mixed origin deliveries of yearly[a] past data

[a]from November 2002 to September 2003

1.6 Targeted markets, historic competition, and strategy

In the following, we deliberately do not make any difference between products: we use only the expression of flat glass, including basic, coated, tempered, laminated and rolled glass.

Flat glass is mainly intended for the building and the automotive markets. Demand nature characteristics and customer relationships depend on the market. However, **in both cases, Saint-Gobain Glass has followed a vertical integration strategy.**

On the one hand, the automotive market is evolving fast year after year. As a result, strong demand forecast methods are used by Saint-Gobain Sekurit. From the Saint-Gobain Glass point of view, the main problem of the automotive market is to update on a yearly basis the dimensions of standard products in order to find the best trade-off between global glass loss and inventory management costs (both depending on the number of references). We deal with this interesting problem in chapter 2.

On the other hand the building market is relatively stable and products are highly normalized. However, the number of references is potentially large. In the following of this section, we only focus in this study on the building market.

We can consider flat glass as a raw material for many standard applications, and thus as a low added-value product. It is sold to a network of **numerous customers (independent or subsidiary), which can be separated into two classes**:

- **Industrial transformers**, who own transformation lines and produce in batches several normalized finished goods (windows, windscreens, etc). They are used to buying huge quantity of few products, and they serve either retailers with transformed goods or directly other industrial companies (which assemble glass with various components to create more complicated products).

- **Specialized retailers**, who are used to buying a wide range of different products in small quantities, because they serve small customers requiring a high level of personalization. It is common to find a retailer with manually operated machines which allow him to answer to specific orders (e.g. to cut exactly at the ordered size).

Historically, many customers have been bought by the Saint-Gobain Group, creating **a huge subsidiary network**. A traditional way of understanding it is to consider Saint-Gobain Glass as an industrial supplier willing to ensure the outlets of its glass, due to a strongly inflexible production process.

This **vertical integration** has resulted in a **strong standardization of products**, especially concerning dimensions. We can for instance consider two main dimension standards, called PLF for the big one and DLF for the smaller one. Many transformation lines and handling tools have thus been designed to work on it. Vertical integration has also allowed a **high price policy**: Saint-Gobain Glass is the standard supplier of subsidiary companies which are leaders on their respective market.

In addition, the **glass market has experienced a steady growth** of several percents per year, due to the fashion of using more and more glass materials in building and automotive as well as interior designs. All these reasons encouraged other competitors to challenge the group, despite high barriers of initial investment.

Innovation has been a key factor in supporting the growth of glass market. Double glass windows, new forming techniques, new compositions, and especially new coating techniques have made the norms evolve, by allowing more security, more energy saving, as well as better mechanical properties.

At the same time, productivity progress has steadily shrunk the production costs, transforming **flat glass into a commodity product**.

The glass market features several big competitors in Europe with different backgrounds and strategies. Saint-Gobain is the oldest one and benefits from an excellent image and reputation. The group is known for its capacity to innovate and offers **a wide variety of products, while manufacturing highly innovative products**. Each new high-value line of products often requires high investment, arguing for the specialization of different sites.

On the contrary, some competitors focus on high volume products and try to provide a relatively narrow product line at the lowest costs.

The global strategy of the group is to meet a very high service level to allow relatively high prices, while trying to make standards evolve towards higher added-value products.

To be ready to fulfill new market trends, the group keeps providing a wide range of products by developing research and taking out patents for new products or processes.

To achieve the strategic fit, the objective is to find the lowest cost supply chain allowing Saint-Gobain Glass to:

- **Keep on providing as much as possible standard (high volume) products at the lowest possible cost**. This market segment is highly competitive and thus margins are limited. However, it is the core of the business. The key factor is mainly the price because every competitor provides approximately the same service level.

- **Catch a high market share on low volume and high added-value products**. A weaker competition can indeed leave a high margin because anything that is in short supply is expensive. These products can be either at the beginning of their life cycle or only luxurious ones (for instance a rare colour glass for particular architecture needs). In the first case the marketing team tries to create new standards (especially on patented processes or products) and it needs to be supported by a reliable supply chain. In the second case, it may be important for the goodwill of the company. Keys for success can thus lie in offering higher variety of on-hand products and better delivery lead times as compared with competitors, at a reasonable price.

As a conclusion, **the product portfolio offered to the customers is relatively wide**. With a multitude of specific products, customers may like to pick and choose as freely as possible, whereas common product prices may become a very large factor in the decision-making process. Due to the process constraints, most of the products are made to stock: the complexity is thus compounded since as in any business, early forecasts contain a tremendous amount of variability.

To simplify, we use the simple model of generic strategies developed by Michael Porter in [Por98] which outlines three main strategic options open to organization that wish to achieve a sustainable competitive advantage. Each of the three options are considered within the context of two aspects of the competitive environment:

- Sources of competitive advantage: are the products differentiated in any way, or are they the lowest cost producer in an industry?
- Competitive scope of the market: does the company target a wide market, or does it focus on a very narrow, niche market?

The three options are the cost leadership, the differentiation, and the focus strategy. A competitive advantage exists when the firm is able to deliver the same benefits as competitors but at a lower cost (cost advantage), or deliver benefits that exceed those of competing products (differentiation advantage). Thus, a competitive advantage enables the firm to create superior value for its customer and superior profits for itself.

It appears that the strategy of Saint-Gobain Glass lies in a wide-range differentiation. The question is then: ***How to build a supply chain that achieves this strategic fit?***

1.7 A decision classification of planning tasks in the supply chain

To build a supply chain that achieves a given strategic fit, managers have to answer many questions: we recall here a classical decision classification according to both the decision level (in term of impact) and the concerned operations. Since the fundamental work of Anthony ([Ant65]), three levels of managerial decision making are referred to (see [BT93] and [Mil01]). They mainly differ with respect to the time during which the decisions will have an impact on the future development of a supply chain or a company. According to this categorization and their planning horizon, planning tasks are commonly assigned to one of the three planning levels "long-term", "mid-term" and "short-term" planning (also called strategic, tactical and operational planning).

"Long-term" planning prepares decisions whose implications on the supply chain can be felt for several years. These decisions essentially determine the physical structure of a supply chain and should directly reflect a company's business strategies. "Mid-term" planning has to effectively use and act within the infrastructure set by the long-term "strategic" planning. According to [SPP98], the validity of a mid-term plan ranges from half a year to two years. The planning horizon of "short-term" planning is restricted to a few weeks or at most a few months. Short-term planning has to put into practice the guidelines given by the upper two levels and to prepare detailed instructions for immediate execution and control of the operations.

In their article [FMW02], the authors make use of the supply chain processes *procurement, production, distribution and sales* to further classify the planning tasks typically emerging for each member of the supply chain. The supply chain matrix (denoted SCP-matrix by the authors) is recalled in Figure (1.12).

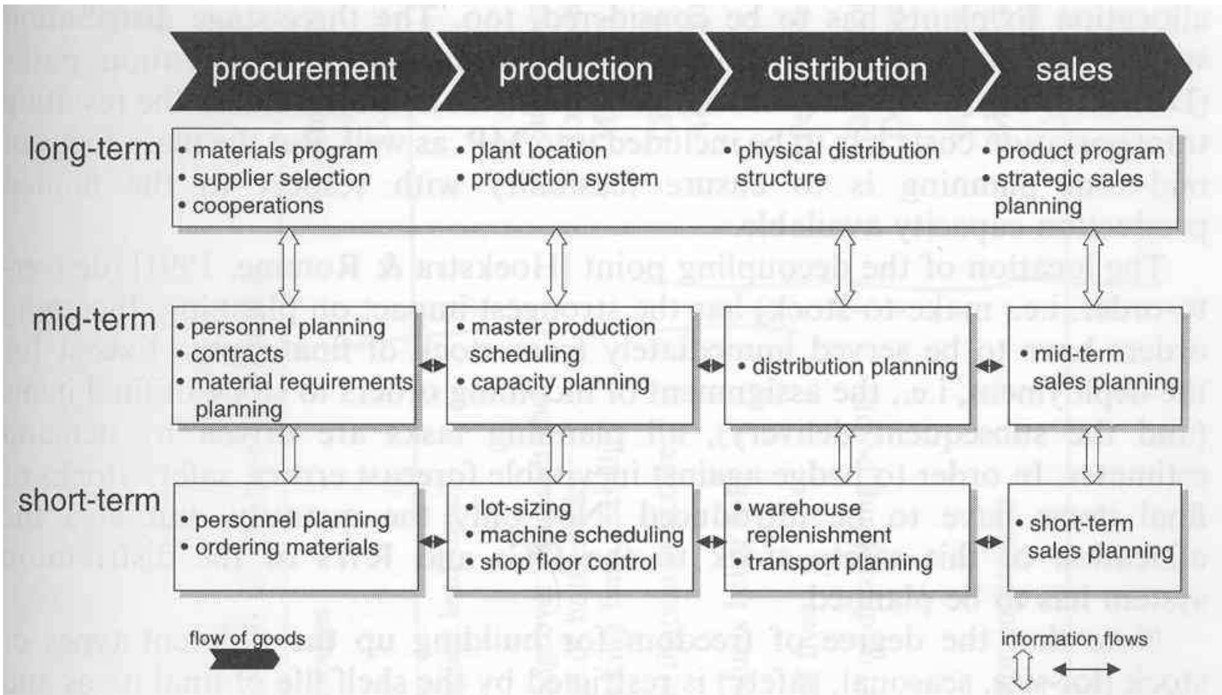


Figure 1.12: Planning tasks according to the SCP-matrix (cf. [FMW02], Fig 4.3)

Of course in some cases, this general overview based on the assignment of planning tasks to planning levels and supply chain processes may be somewhat fuzzy. However, we found that this topology suited pretty well issues of the glass industry, and we even simplified it. Figure (1.13) summarizes our six class classification: to focus our research on a limited perimeter of the supply chain, we assume that operations pertaining to both sales and procurement are ideal. Thus, we do not capture optimization issues on these parts of the supply chain. More generally we should take into account six more classes, dealing with both procurement and sales issues. To understand the complexity of real industrial problems, we have decided to create a list of potential decisions that managers of Saint-Gobain Glass have to tackle.

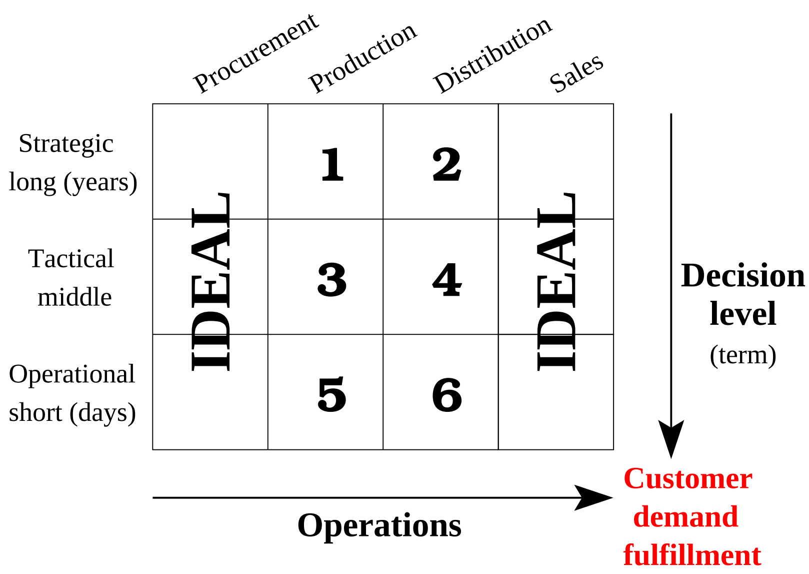


Figure 1.13: Our simplified classification of problems according to both operations and decision level

Here is an example of non-exhaustive list of decisions that may be faced. Technical vocabulary is introduced in part (§ 1.2) and (§ 1.3). To simplify it, we do not precise at each decision level that managers still have the choice to contract a job out: this is the famous "Make or Buy" formula.

1. **Strategic decisions (long term horizon, for instance years) on the industrial schema: industrial supply chain design**

    - About the flat glass production lines: float lines.
        - Localization of each line? Do we create or close a line?
        - Which skill portfolio do we allocate to each production line? Is it a better strategy to specialize lines or to create versatile ones? Which colour set, thickness and width ranges do we assign to each line?
        - What is the capacity (called the pull, in T/day) of each line?
        - How do we fit production pull of plants according to demand forecasts? We may place maintenance periods such as reconstructions of furnaces whose duration may be up to six months.

- About the transformation lines of flat glass: (coating lines, laminated glass lines, tempering lines, cutting lines, etc.) :
  - Localization of lines? If this may be potentially an on-line transformation (directly on the float line), do we choose on-line (low investment, huge inertia) or off-line (important investment but maximal flexibility) process?
  - If we choose an off-line process, where do we localize production lines?
    * on a float line site?
    * on an industrial platform?
    * on an independent site?
  - How do we decide to assign skills to lines? For each line, which product portfolio is it possible to produce and with which capacity?

2. **Strategic decisions on the logistic schema: logistic supply chain design:**

- About the supplies and materials inventories, work-in-process inventories, finished good inventories:
  - For each product, how many inventory levels do we want to use? Where do we place them?
  - Where do we localize platforms (we use this generic term to denote inventories independently from other elements of the supply chain)? : on production sites (float lines and/or off-line transformation ones) or on independent ones?
  - How do we determine the size of inventory platforms? Which global inventory capacity do we allocate, which maximal input and output flows do we set (for instance, how many platforms do we build?), which transportation means are usable (truck, train, etc.)?

  - Which high-value operations do the platforms provide? Transforming products (for instance, with an off-line cutting line), cross-docking different product flows, packaging orders (for instance, preparing containers for exportation), picking-up orders (with a suitable information system), delivering orders?

- Transportation of work-in process and finished products:
  - Do we deliver orders to final customers? If yes, do we own partly or fully our transportation means? How do we bill customers for it?
  - Which transportation means do we use? Which flow do we use for every product (inter-plant, plant-primary platform, primary-secondary platform, secondary platform-customer flows, etc.): How do we define each product route-to-market?
  - Which packaging do we use for each transportation means (for instance, which glass stack size do we define for each standard?) ?
  - Which trestle do we use for carrying glass stacks?

3. **Tactical decisions (middle term horizon, such as six months or one year) on industrial schema:**

- How do we fit production pull of plants according to demand forecasts? We may place light maintenance periods (such as facing of furnaces), adjust pulls in a given narrow range, or decide to shut down temporarily a plant (technical unemployment).
- How do we set the tactical production planning? For float lines, how do we create the yearly colour plan ; for each colour campaign, how do we plan different products' families, and how do we define these families? Is it important to plan simultaneously float and off-line transformation lines?

- For each product, is it better to use a make-to-order or a make-to-stock policy? For products made of several production steps, where do we place the decoupling point?

4. **Tactical decisions on logistics issues:**

   - Inventories:
     - Maximal inventory level determination at each point of the supply chain for make-to-stock policies.
     - Which inventory management method do we use (periodic or continuous reviews?) ? How do we compute safety stocks?
     - What is the inventory unit, depending on the packaging at the different levels of the supply chain?
     - What is the commercial policy for every product? Do we set-up a delivery lead time?
   - Transportation:
     - Do we use vehicle routes? If yes, do we authorize multiple pick-up routes and/or multiple delivery routes?
     - Are there ordering rules? Do we enforce some mixing rules? Do we force an order to correspond to an unique truck?
     - How do we deal with limited required transportation resources which are under limited during peak seasons?

5. **Operational decisions (short term horizon, from several minutes to days) about industrial issues:**

   - How do we schedule plants on the very short-term?
   - How do we face unforecasted events (shut downs, extraordinary big orders, etc.)?

- How do we minimize useless flows (for instance, we may try to send directly production without handling glass in and out of inventories)?

6. **Operational decisions about logistic issues:**

- How do we capture quickly and efficiently product flows?
- How do we set priorities in preparing and delivering orders?
- How do we work with our transportation means suppliers? How do we built vehicle routes?
- And finally, how do we set a reliable set of performance indicators?
- How do we define and check our customer service level?

This non exhaustive long list of decisions to face underlines the complexity of the supply chain management in a real industrial context. We may now introduce the overview of this thesis, which answers partially to some of the most important unsolved questions we have identified.

1.8 Thesis Overview

This section presents the overview of this thesis. Given information from the industrial context that we have introduced so far, we motivate the different parts of our research and link them together through a global purpose that makes sense: we aimed at improving the supply chain management of the business, based on the statement we faced three years ago.

As explained in section (§ 1.6), flat glass is mainly produced for the building and the automotive markets. Saint-Gobain Glass has followed a vertical integration strategy in both of them.

First of all, we deal with the determination of standard-product dimensions in chapter 2, which is a tactical decision updated yearly. On the one hand, the building market is highly standardized, the demand is pretty steady and thus standard products are imposed by the market.

On the other hand, the automotive market is evolving fast every year. Basically, Saint-Gobain Glass supplies trestles of big dimension glass sheets ; customers then cut it into pieces adapted to their own demand. Given that customers are subsidiaries of the group, it makes sense to try to minimize the global loss of glass during different cutting operations along the supply chain, by adapting standard products to demand forecasts. Thus the main problem of the automotive market is to update yearly the dimensions of standard products in order to find the best trade-off between global glass loss and inventory management costs of numerous references. We deal with this interesting problem by introducing an original multi-format structure that makes the cutting optimization problem interesting. The non-negligible impact (estimated around several millions of euros a year) of the decision-support tool we provided to industrial managers underlines the potential of designing a specific tool fitting a particular unsolved problem.

The important issue of glass loss minimization being tackled, we evolve gradually in the following of the thesis from chapter 3 to chapter 6 towards the definition of both a framework and an original integrated production-inventory-distribution model which captures the specific supply chain of glass under deterministic assumptions.

We have developed our research step by step, from basic models to an integrated one, which is now used by practitioners as a both tactical and strategic decision-support tool.

In chapter 3, we start by a study of supply chain design methods used for simple localization problems. Given the structure and the costs of a simple supply chain, how is it possible to build models that help managers to determine both the number and the location of facilities?

Despite being too simplified to capture real industrial issues, results of this chapter are used for customer aggregation in chapter 6, providing an interesting approximation to industrial size problems.

Chapter 4 introduces an original production modeling framework that has a great particularity in our research: we apply it at both the operational scheduling level and the tactical planning one. Based on the decomposition of products into characteristics, we have developed and factorized existing models capturing sequence dependent set-up times and costs to be able to tackle practical issues we have faced in the glass industry.

Given a production planning decision level, we define a method in which meaningful product characteristics are divided into attributes and sub-attributes, corresponding to big and small time buckets. Our model is in a way an original synthesis of different modeling methods we have found in the scientific literature. We applied it to different decision levels of production planning in the float glass industry, and our belief is that this is an illustration of its adaptability to other process industries.

Using the adaptability of our production planning model, we apply it in chapter 5 to other jobs of Saint-Gobain Glass (cf. Figure (1.1)), which are transformations of float glass. To do so, we underline an interesting design problem we identified as a prerequisite for applying it to the coating lines. Basically, coating lines are made of metallic cathodes that are used on-line to splutter nanometric metallic coats on flat glass sheets. Before optimizing the production planning of coating lines, managers needed to have a decision-support tool to configure the on-line cathode sequence. Finally, chapters 4 and 5 give us a method for modeling all production jobs of Saint-Gobain Glass, allowing us to integrate production tools in our final model.

As a result, chapter 6 provides a global framework for modeling multi-location supply chains, by capturing and integrating all deterministic production-inventory and direct distribution systems. We address the problem of developing a decision tool for both the production planning and the logistic decisions in the glass manufacturing industry.

In chapter 4, we develop a generic production planning model allowing us to capture some continuous process industries. We highlight that it may be used at every level of a hierarchical production planning process. However, it may appear that for a given business it does not make sense to optimize the production planning independently from the distribution context. Using our production model as a building block, we integrate this work in a multi job, multi machine and multi location model.

We apply our research to different decisions we have met and solved in the glass industry. Firstly, float glass is mainly transformed through different processes to provide commodity products, such as laminated glass or coated glass. We explain how our model capture these production and transformation processes. Secondly, we apply our tool to the tactical production planning, minimizing both production, storage and transportation costs. Finally, we present how we do create a generic decision support tool for strategic decisions such as the localization of new facilities. We provide several practical approximations allowing overcoming the tremendous size of industrial applications, using for instance results of chapter 3.

As a conclusion, our thesis is a step by step research that we applied to the glass industry through applications covering strategic and tactical as well as operational issues (points 1, 2, 3, 4 and 5 of our simplified classification, see Figure (1.1)). At each step, based on a solid literature review, we extend up-to-date models to more complex ones suiting industrial problems. In each chapter, we underline potential or real savings we have identified with managers of Saint-Gobain Glass on real data cases. This thesis is the result of a highly motivating collaboration between industrials and academics and we believe that this work present large possible outlooks that we hope will be developed in the future.

Chapter 2

Determination of standard products

2.1 Introduction

In this chapter, we address the problem of the yearly determination of standard product dimensions for the automotive market of Saint-Gobain Glass.

In the Saint-Gobain case, the provider and its customers (Saint-Gobain Sekurit plants) belong to the same company. This vertical integration allows us to minimize the global loss of glass during the various cutting operations. Based on the yearly demand forecasts we aim at determining the cheapest trade-off between the loss of glass in the supply chain and the cost of inventory management of numerous references.

This chapter presents first a brief introduction to the industrial problem. We aim at minimizing glass loss cost under constraints on the number of standard products. Then comes the model and the way we have implemented the algorithm. The complexity of the optimization comes from the introduction of a format structure within standard products that creates a double objective.

Last but not least, our conclusion points out what has been the industrial interest of such a tool: we estimated savings of several millions of euros per year for the Saint-Gobain group.

2.2 Industrial context

Depending on their dimensions, various glass sheets are classified within classes, which by convention are called formats. After a brief paragraph developing the trade-off motivating the determination of standard products, we present the data set provided to Saint-Gobain Glass by its customers and the underlying formats' structure.

2.2.1 Trade-off on the number of standard products

Why is this determination of standard products a tactical decision which is updated yearly by industrial managers? Understanding the trade-off between the costs of inventory management on the one hand and the costs of the raw material loss on the other hand gives a sense to our optimization problem.

First of all, why does the number of standard products increase inventory management costs? We may consider two cases following a make-to-stock policy: a case in which we have several customized dimension products and another one in which we have only one standard product. We distinguish at least three reasons arguing that inventory management costs are lower in the later case: ordering costs, uncertainty covering costs and storage costs.

Ordering costs are not only proportional to the ordered quantity but also often fixed by order. Naturally, the less numerous the references, the less costly the replenishment management.

In Chapter 11 of [CM01c], the authors recall clearly the impact of aggregation of non substitutable products on safety inventory: the basic idea lies in the fact that the variance of a sum of independent random variables is not the sum of their variances, but its square root (the so-called "square root law"). Therefore the uncertainty of a standard product demand is lower than the sum of uncertainties of corresponding customized products. Thus, for a given service level, the safety stock corresponding to a standard product is significantly less than the sum of safety stocks of former products.

Finally, and this statement is particularly important in the glass industry, given a constant global volume, the space required for storing products is increasing in the number of references. For instance, glass sheets are stored by stacks and handling operations require that a sufficient space is kept between different stacks. It is much less space consuming to use big stacks than small ones, because a stack must contain identical glass sheets to avoid time-consuming operations due to the fact that a stack is a *last in first out* system. Storage costs are thus increasing in the number of stored references.

On the other hand, standardization of glass sheets leads to the rise of glass loss during cutting operations at customers' plants: we may compute a cost not only corresponding to this loss, but also integrating the cost of both carrying useless glass through the distribution channels and then recycling it (pick up the glass loss and clean it as well as carrying it back to glass plants to melt it). It may be objected that it is possible to produce on-line a unique standard size that is stored, before being cut-to-order as soon as the customer request is known. However, as explained in section (§ 1.3), on-line cutting of the float glass ribbon is cheaper than off-line operations. This crucial factor makes that the proportion of on-line cutting operations is larger than off-line ones. At first sight, our discussion above makes sense, but it would be interesting to capture in a model the impact of both on-line and off-line cutting-operations.

The determination of standard products is thus an optimization problem aiming at finding simultaneously the optimal number of standard references and their dimensions, in order to minimize the sum of inventory management costs and glass loss costs - respectively increasing and decreasing in the number of standard references.

We had the choice to consider these costs either as costs or as constraints. **Due to practical considerations, we decided to take into account the number of standard products as a constraint while minimizing the glass loss cost in the overall operations**. We may motivate this choice by emphasizing the difficulty of cost computations: managers were unable to determine inventory management costs as a function of the number of standard products. Finally, managers use our tool to determine optimal standard products for a given restricted number of final references. Following an iterative method, their determine what seems to be the optimal number, according to their knowledge of the business.

2.2.2 Data Set

First of all, we present the problem and the data set structure. Each data set is composed by a list of demand forecasts for every product wanted by Saint-Gobain Sekurit plants. Each customer sends to Saint-Gobain Glass a list of its yearly forecasts, which are made of a given quantity for each product. It exists for Saint-Gobain Glass various dimension ranges that define for instance “big”, “medium” and “small” sizes of glass sheet (so-called PLF, DLF and so on) that may be provided to Saint-Gobain Sekurit, depending on the customer equipment. We denote these ranges by introducing the notion of format.

Definition 11 *We define a format as a rectangle set whose length and width belong to given ranges.*

Basically, colour and thickness being defined, a product is defined by:

- a width.

- a length.

- The format of the glass sheet in which the customer plans to cut it. This format is due to some industrial constraints: basically, the production line of each customer can often only be fed by one particular format.

- A boolean variable corresponding to the possibility to exchange its dimensions. This notion is due to the usual impossibility to cut a sheet of glass in a bigger one in any direction. Float glass is indeed marked by the direction of the ribbon: some physical constraints remain in the direction of pull ; they can alter the optical properties of the glass, which can be bothering for some applications of the automotive market (windscreens, etc.).

  By **convention**, we use in this chapter the term **length for the dimension along which the ribbon has been pulled during the float process and width for the orthogonal dimension**. Following the final application specifications, this variable determines whether or not we have the choice to produce the sheet in one or another direction.

Figure (2.1) illustrates for one single format the on-hand industrial data (in blue) and the variables we have to optimize (in red).

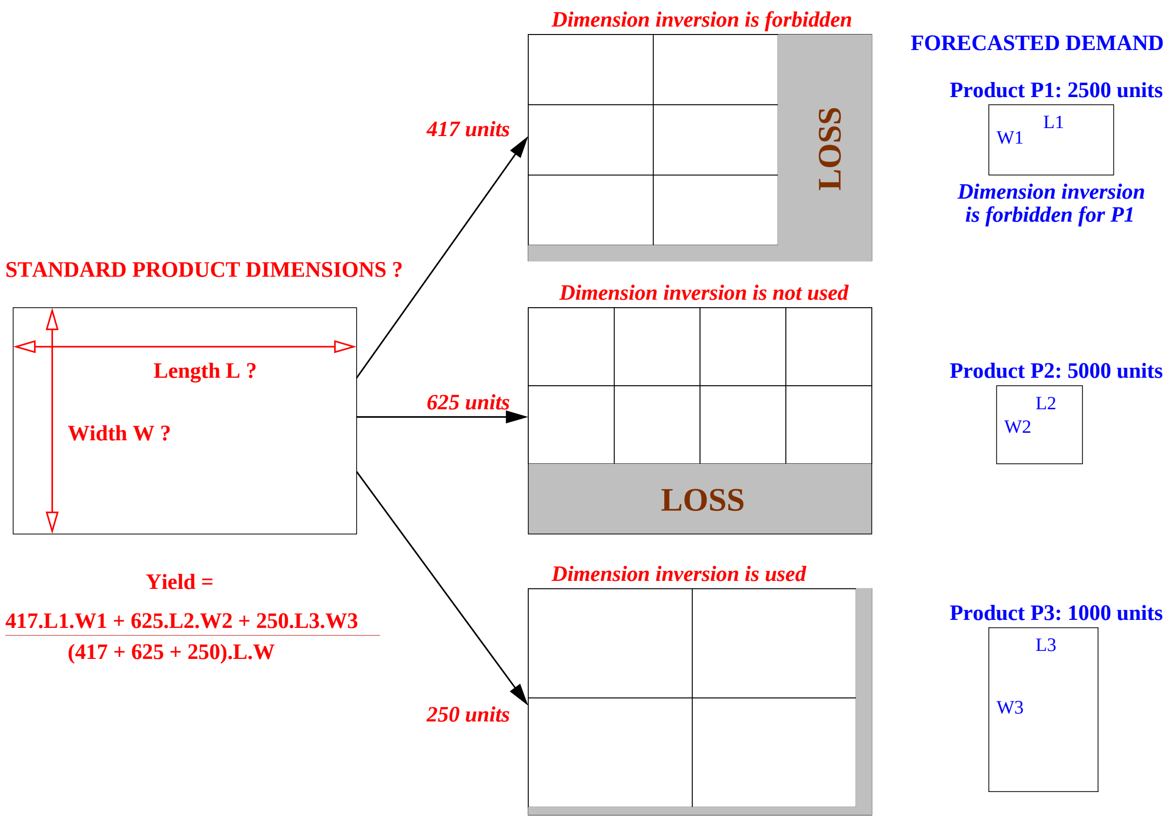


Figure 2.1: Illustration of the industrial data for one given format

2.2.3 The Formats' structure

In practice, every piece of glass belongs to a class. Each class is characterized by a given range for each dimension. We call each class a format. Let us consider a known set of different formats. We denote it $\{f = 1 \ldots F\}$. Each format captures glass sheets belonging to a range of lengths $[L^f_{min}; L^f_{max}]$ and a range of widths $[W^f_{min}; W^f_{max}]$.

A hierarchy exists between formats. Some formats can be cut from bigger ones. We model the hierarchy between formats by a formats' oriented graph in which each arrow indicates a relationship between two formats (cf. Figure (2.2)): an arrow from the format f_1 to the one f_2 indicates that f_2 can be obtained from f_1 by cutting.

It often happens that a given format (for instance f_5) can be obtained from several formats (f_1 and f_3), and can be cut into several other formats (f_6, f_7 and f_8). By convention, we call a format a root of the directed acyclic graph when it cannot

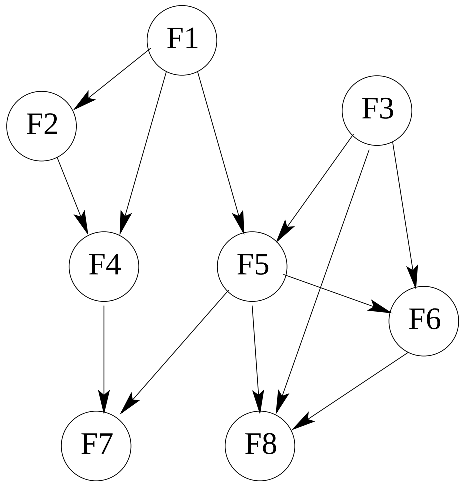


Figure 2.2: The formats' graph representing the hierarchy within formats.

be derived from another format (f_1 and f_3 are the only roots of the example herein). To characterize this directed acyclic graph, we define two functions:

- $R : \mathbb{N} \longrightarrow \{0, 1\}$ is 1 if the format f is a root of the formats tree. For instance, $R(f_i) = 0$ for $i \notin \{1, 3\}$.

- $F : \mathbb{N} \longrightarrow \mathbb{N}^n$ links for a given format f the list of the direct derived formats. For instance, $F(f_5) = \{f_6, f_7, f_8\}$ and $F(f_8) = \emptyset$.

Following the basic idea of the generation of all possible standards in each dimension, we have to create a set of all possible standards in each format using a specified discrete step for each dimension (length and width).

For each format f, the method to compute all the possibilities is to generate all possible widths within the range $[W^f_{min}; W^f_{max}]$ using the specified step S_W. Concerning the length, we generate all possible lengths in the range $[L^f_{min}; L^f_{max}]$ using the specified step S_L. From these two lists of both possible widths and lengths, we may generate all possible standards in each format.

2.2.4 Constraint set

In a nutshell, we have a data set made of a list of forecasts. We aim at optimizing the yield while satisfying some general constraints, such as a maximal number of standard products, a maximal number of used ribbon widths, some minimal yields by customer, by products, etc. Naturally, taking into account the previous paragraph, we find again the classification by formats within the standards that we aim at determining. Therefore, the constraint set can take into account various constraints such as a maximal number of standards per format, a minimal yield per format, a list of mandatory (or potential) lengths (or widths) of standards per format, some mandatory (or potential) standard dimensions, etc.

To simplify the reading, we chose not to be exhaustive in the list of constraints here: the reader will discover in the following all the optional constraints that can be activated by the user.

Each data is associated with at least one constraint set: we aim at solving each hypothesis to allow the user to choose the best solution. For both a given data set and a given constraint set, we use the same optimization method, presented in section (§ 2.3)

2.3 Optimization Process

In this part, we work with a given constraint set associated with a given data set. We aim at minimizing the global loss of glass during the cutting operations.

2.3.1 Optimization Goal

Due to the nature of the problem, the objective of minimizing the loss of glass is not trivial: which loss of glass?

Firstly, the user may want to minimize exclusively the loss of the cutting operations of each final product in its associated standard. In this simple case, we do not need to take into account the formats' directed acyclic graph because every format is independent from one another. Therefore, we can optimize separately in the second step each format and thus decrease the global computation time.

Secondly, the user may ask to take into account the formats' graph and its relationships between formats and to minimize the global loss of glass, which is both the sum of the loss of the cutting operations of the final products in their standard and the sum of the loss corresponding to the formats' graph, each standard being cut in another standard of one of its father formats. For instance, this may aim to have multiplicity relationships between two standards whose formats are linked. In this complex case with two different objectives, we need to find both:

- For each format, the relationship between the standards and the final products.
- Between standards of two linked[a] formats, the relationship between them.

In the following, we present the most general model. Depending on the user's specifications, we simplify it as soon as possible to minimize the problem size and thus the computation time.

2.3.2 Optimization Method

Following our idea to generate all possible standards in each format, we could solve the global problem in one step by generating simultaneously all possible lengths and widths (as described in section (§ 2.2.3)). This would lead directly to the general model described in section (§ 2.3.2). Unfortunately, such a method is not usable in practice due to the huge number of integer variables. Using the best available commercial solver (e.g. Ilog Cplex, [ILOa]), we do not succeed in solving real case data sets.

[a]linked by an arrow in the formats' graph

Simplification

Fortunately, due to the sense of the glass ribbon (cf. qualitative explanations in section (§ 2.2.2)) we can divide the problem into two successive subproblems. The first sub problem allows us to simplify the second general one.

Firstly, we solve the width problem: we optimize the standard widths of the float glass ribbon. We found a list of optimal values $\{W_1^*, W_2^*, \ldots, W_S^*\}$. This may allow us to reduce the number of possible widths for the formats which are directly using the width of the ribbon.

Secondly, we solve the general problem by using the following simplification as soon as it is relevant: for each format f, we generate all possible standard dimensions by the following procedure:

1. generating all the discrete possible lengths (using the format's range $[L^f_{min}; L^f_{max}]$ and the specified precision step S_L).

2. generating the widths:

   - if there exists at least one glass ribbon width W_i^* that belongs to the format's range of widths $[W^f_{min}; W^f_{max}]$, we use the set of compatible widths solution of the first sub problem $\{W_i^* \ / \ i = 1 \ldots S$ and $W_i^* \in [W^f_{min}; W^f_{max}]\}$.

   - otherwise, we generate also all the possible widths (using the format's range and the specified precision step S_W).

3. creating all possible pairs {Length ; Width}: each pair corresponds to a possible standard product.

This simplification allows us to considerably reduce the size of the problem. In addition, we believe that this approximation is relevant because the direction of the glass is globally constrained: the two directions are thus quasi independent.

First step: Optimization of the width of the float glass ribbon

Basically, this step aims at finding a discrete number of widths of the glass ribbon: we know which products are going to be produced, and we want to use a limited number of ribbon widths on the float line.

- In the data set we have:
  - A set of potential widths of the ribbon $\mathcal{W} = \{W_i; i = 1 \dots n\}$. We obtain it by using the specified minimal and maximal ribbon widths $[W_{min}; W_{max}]$ combined with a specified discrete step $S_{\mathcal{W}}$.
  - A set of products $\mathcal{P} = \{p = 1 \dots P\}$. To capture the fact that product dimensions may be inversed or not, we build a set of so-called virtual products: to one product correspond two orthogonal virtual products whether the inversion is possible. Each product is characterized as described in section (2.2.2). We create from $\mathcal{P}$ a set of virtual products $\mathcal{J} = \{j = 1 \dots m\}$ with a corresponding set of incompatible virtual products sets $\mathcal{K} = \{K_p; p = 1 \dots P\}$. We obtain these two sets from the original set $\mathcal{P}$ by a single procedure:
    1. $\mathcal{J} = \emptyset$ and $\mathcal{K} = \emptyset$
    2. $\forall p$:
       * If the dimensions of product p can not be inversed[a], we take it without any change: p is added in $\mathcal{J}$ and $K_p = \{p\}$.
       * Otherwise, when dimensions can be exchanged[a], we create one virtual product p' with exchanged dimensions. p and p' are added to $\mathcal{J}$ and these two products constitute a new set of incompatible products $K_p = \{p, p'\}$. This captures the fact that one real product can only be produced in one unique way.
  - Constraints:

  * The number of used widths must belong to a given range $[N^W_{min}; N^W_{max}]$.
  * Some widths are mandatory. Let us denote $\mathcal{M}_W = \{W_i\}$ the corresponding list.

- Definitions:
  - $L : \mathcal{J} \longrightarrow \mathbb{R}$ links to a product j its length.
  - $Q : \mathcal{J} \longrightarrow \mathbb{R}$ links to a product j its quantity.
  - Δ_l[a] $: (\mathcal{W} \times \mathcal{J}) \longrightarrow \mathbb{R}$ links a couple $\{i, j\}$ to the maximal integer number of products j that could be cut into the i^{th} width.
  - Π_l[a] $: (\mathcal{W} \times \mathcal{J}) \longrightarrow \mathbb{R}$ links a couple $\{i, j\}$ to the linear proportional loss associated to the cut of $\Delta_l(i, j)$ products j into the i^{th} width.
- Variables:
  - Y_i is a Boolean variable representing whether or not the i^{th} width W_i is used.
  - X_{ij} is a Boolean variable representing whether or not the i^{th} width is used to produce the j^{th} product.
- Model:

$$\mathrm{Min}\left(\sum_{i=1}^{n}\sum_{j=1}^{m} X_{ij} \times W_i \times L(j) \times \Pi_l(i,j) \times \frac{Q(j)}{\Delta_l(i,j)}\right) \tag{2.1}$$

[a] cf. qualitative discussion in section (2.2.2) about the direction of the glass ribbon

$$N^W_{min} \leq \sum_{i=1}^{n} Y_i \tag{2.2}$$

$$\sum_{i=1}^{n} Y_i \leq N^W_{max} \tag{2.3}$$

$$\forall p \qquad \sum_{i=1}^{n} \sum_{j \in K_p} X_{ij} = 1 \tag{2.4}$$

$$\forall i \qquad Y_i \leq \sum_{j=1}^{m} X_{ij} \tag{2.5}$$

$$\forall i \qquad \frac{\sum_{j=1}^{m} X_{ij}}{m} \leq Y_i \tag{2.6}$$

$$\forall W_i \in \mathcal{M}_W \qquad Y_i = 1 \tag{2.7}$$

$$\forall \{i,j\} \text{ s.t } \Delta_l(i,j) = 0 \qquad X_{ij} = 0 \tag{2.8}$$

The objective function (2.1) represents the loss of glass due to the cut of every product in the different used widths. It is directly proportional to the requested quantity of each product.

The inequalities (2.2)and (2.3) force that the total number of used widths satisfies the specified range. The equalities (2.4) represent that there is exactly one width used for each set of incompatible products. Knowing the definition of these sets, it corresponds to a unique width used for each real product. The inequalities (5.6) and (2.6) imply the structural relationship between the Boolean variables Y_i and X_{ij}: a width is used if and only if there is at least one corresponding product. Finally, (2.7) forces the use of the mandatory widths and (5.7) forbids the use of impossible links between a width and a product (when no product can be cut into the width).

This first step gives us a list of S possible ribbon widths that we consider given in the next step. If we classify them from the smallest to the biggest, we denote them: $W^*_1 \leq W^*_2 \leq \cdots \leq W^*_S$. We use it for generating the data of the second step, following the procedure described in section (§ 2.3.2).

Second Step: Optimization of the standard dimensions

In this paragraph we present the more general model. Depending on the optimization goal (cf. discussion in section (§ 2.3.1)), we optimize independently each format or not. If the formats are independent, we use this model without the index f.

Following the method of generation of all possible standards (also called fathers), we have this set of data and functions:

- Data:
  - We consider a set of formats $\mathcal{F} = \{f = 1 \dots F\}$ whose hierarchy can be represented by a formats' tree similar to the one described in section (§ 2.2.3). We know the functions[a] R and F characterizing it.
  - We know a list of customers $\mathcal{C} = \{c = 1 \dots C\}$. The function $\phi_c^{min} : \mathbb{N} \longrightarrow \mathbb{R}$ links to each customer its minimal asked yield for its set of products.
  - **In each format** f, we denote:
    * $\mathcal{S}_f = \{i = 1 \dots n_f\}$ **is the set of possible standard dimensions** determined by the procedure described in section (§ 2.3.2). By convention, we denote $\{f, i\}$ the standard $i \in \mathcal{S}_f$. We note:
      · $\mathcal{W}_f$ the set of possible widths of the standards $\mathcal{S}_f$
      · $\mathcal{L}_f$ the set of possible lengths of the standards $\mathcal{S}_f$
    * $\mathcal{P}_f = \{p = 1 \dots P_f\}$ is the set of real products, from which we derive (following the same reasoning as in the optimization of the widths) two sets:
      1. **a set of virtual products:** $\mathcal{J}_f = \{j = 1 \dots m_f\}$. Each product $j \in \mathcal{J}_f$ has a given:
         · length $L_f(j)$
         · width $W_f(j)$

[a]defined in section 2.2.3 on page 42

· quantity $Q_f(j)$

· customer $C_f(j)$

· minimal yield $\phi_f^{min}(j)$

2. a set of incompatible virtual products sets: $\mathcal{K}_f = \{K_p^f; p = 1 \ldots P_f\}$

– It appears that there are two main objectives. First, we try to maximize the yield of the association between fathers and products. Second, we try to maximize the yield of the derivation of non-root fathers from one another. We introduce two coefficients to weight the importance of each objective in the function to minimize:

  * $0 \leq \lambda_1 \leq 1$ **is the weight of the first objective** (yield of the cutting operations of the final products in their standards).

  * $0 \leq \lambda_2 \leq 1$ **is the weight of the second objective** (yield of the cutting operations of the non-root standards in their father standards).

– **Constraints:**

  * The global number of used standards must belong to a given range $[N_{min}; N_{max}]$.

  * In each format f:

    · The number of used standards must belong to $[N^f_{min}; N^f_{max}]$.

    · Some lengths are mandatory. Let $\mathcal{M}^f_L$ be the corresponding given list. Of course, $\mathcal{M}^f_L \subset \mathcal{L}_f$.

    · The number of used lengths must belong to $[NL^f_{min}; NL^f_{max}]$.

    · Some widths are mandatory. Let $\mathcal{M}^f_W$ be a given list. Of course, $\mathcal{M}^f_W \subset \mathcal{W}_f$.

    · The number of used widths must belong to $[NW^f_{min}; NW^f_{max}]$.

    · Some standards are mandatory. Let $\mathcal{M}^f$ be a given list. Of course, $\mathcal{M}^f \subset \mathcal{S}_f$.

- Definitions:
  - We denote H_R the set of root formats. $H_R = \{f \in \mathcal{F} \ / \ R(f) = 1\}$.
  - We denote $f_f : \mathcal{L}_f \longrightarrow \mathbb{N}^n$ the function that associates to $l \in \mathcal{L}_f$ the set of standards $i \in \mathcal{S}_f$ whose length equals l. Of course, $\forall l \in \mathcal{L}_f : f_f(l) \subset \mathcal{S}_f$.
  - We denote $g_f : \mathcal{W}_f \longrightarrow \mathbb{N}^n$ the function that associates to $w \in \mathcal{W}_f$ the set of standards $i \in \mathcal{S}_f$ whose width equals w. Of course, $\forall w \in \mathcal{W}_f$: $g_f(w) \subset \mathcal{S}_f$.
  - We denote $\mathcal{J}_f^c$ the set of final products of format f whose customer is c: $\mathcal{J}_f^c = \{j \in \mathcal{J}_f \ / \ C_f(j) = c\}$. Of course, $\forall c \in \mathcal{C}$: $\mathcal{J}_f^c \subset \mathcal{J}_f$.
  - ${\Gamma_f}^a : \mathcal{S}_f \longrightarrow \mathbb{R}$ links to the standard $\{f, i\}$ its surface.
  - ${\Delta_f}^a : (\mathcal{S}_f \times \mathcal{J}_f) \longrightarrow \mathbb{R}$ links a couple $\{i, j\}$ to the maximal integer number of products $j \in \mathcal{J}_f$ that could be cut into the standard $i \in \mathcal{S}_f$.
  - ${\Pi_f}^a : (\mathcal{S}_f \times \mathcal{J}_f) \longrightarrow \mathbb{R}$ links a couple $\{i, j\}$ to the proportional loss associated to the cut of $\Delta_f(i, j)$ products $j \in \mathcal{J}_f$ into the standard $i \in \mathcal{S}_f$.
  - $\Delta^a : \big((\mathcal{S}_f \times \mathcal{J}_f) \times (\mathcal{S}_{f' \in F(f)} \times \mathcal{J}_{f'})\big) \longrightarrow \mathbb{R}$ links to a couple $\{\{f_1, i_1\}, \{f_2, i_2\}\}$ the maximal integer number of standards $\{f_2, i_2\}$ that could be cut into the $\{f_1, i_1\}$ standard. Of course, $f_2 \in F(f_1)$.
  - $\Pi^a : \big((\mathcal{S}_f \times \mathcal{J}_f) \times (\mathcal{S}_{f' \in F(f)} \times \mathcal{J}_{f'})\big) \longrightarrow \mathbb{R}$ links to a couple $\{\{f_1, i_1\}, \{f_2, i_2\}\}$ the proportional loss associated to the cut of $\Delta(f_1, i_1, f_2, i_2)$ standards $\{f_2, i_2\}$ (of format $f_2 \in F(f_1)$) into the standard $\{f_1, i_1\}$.
- Variables:
  - Y_{fi} is a Boolean variable representing whether or not the standard $i \in \mathcal{S}_f$ is used.
  - X_{fij} is a Boolean variable representing whether or not the standard $i \in \mathcal{S}_f$ is used to produce the product $j \in \mathcal{J}_f$.

– $N_{f_1 i_1 f_2 i_2}$ is a real variable representing the number of standard $\{f_1, i_1\}$ used to cut the standard $\{f_2, i_2\}$. Of course, $f_2 \in F(f_1)$.

– D_{fl} is a Boolean variable representing whether or not the length $l \in \mathcal{L}_f$ is used.

– D_{fw} is a Boolean variable representing whether or not the width $w \in \mathcal{W}_f$ is used.

- Model:

$$\text{Min}\left(\sum_f \sum_{i \in \mathcal{S}_f} \Gamma_f(i) \times \Bigg(\quad \lambda_1 \quad \Big(\sum_{j \in \mathcal{J}_f} X_{fij} \Pi_f(i,j) \frac{Q_f(j)}{\Delta_f(i,j)}\Big) \right. \tag{2.9}$$

$$\left. + \lambda_2 \Big(\sum_{f' \in F(f)} \sum_{i' \in \mathcal{S}_{f'}} N_{fiff'i'} \Pi(f, i, f', i') \Big) \Bigg) \right)$$

$$N_{min} \leq \sum_f \sum_{i \in \mathcal{S}_f} Y_{fi} \tag{2.10}$$

$$\sum_f \sum_{i \in \mathcal{S}_f} Y_{fi} \leq N_{max} \tag{2.11}$$

$$\forall f \qquad N^f_{min} \leq \sum_{i \in \mathcal{S}_f} Y_{fi} \tag{2.12}$$

$$\forall f \qquad \sum_{i \in \mathcal{S}_f} Y_{fi} \leq N^f_{max} \tag{2.13}$$

$$\forall f, \forall p \in \{1 \ldots P_f\} \qquad \sum_{i \in \mathcal{S}_f} \sum_{j \in K^f_p} X_{fij} = 1 \tag{2.14}$$

$$\forall f, \forall i \in \mathcal{S}_f \qquad Y_{fi} \leq \sum_{j \in \mathcal{J}_f} X_{fij} \tag{2.15}$$

$$\forall f, \forall i \in \mathcal{S}_f \qquad \frac{\sum_{j \in \mathcal{J}_f} X_{fij}}{m_f} \leq Y_{fi} \tag{2.16}$$

$$\forall f, \forall i \in \mathcal{S}_f, \forall f' \in F(f), \forall i' \in \mathcal{S}_{f'} \qquad N_{fif'i'} \leq \infty \times Y_{fi} \tag{2.17}$$

$$\forall f, \forall i \in \mathcal{S}_f \forall f' \in F(f), \forall i' \in \mathcal{S}_{f'} \qquad N_{fif'i'} \leq \infty \times Y_{f'i'} \tag{2.18}$$

$$\forall f \notin H_R, \forall i \in \mathcal{S}_f \qquad \sum_{(f_1 \text{ s.t. } f \in F(f_1))} \sum_{i_1 \in \mathcal{S}_{f_1}} N_{f_1 i_1 f i} \times \Delta(f_1, i_1, f, i) \tag{2.19}$$

$$= \sum_{(f_2 \in F(f))} \sum_{i_2 \in \mathcal{S}_{f_2}} N_{f i f_2 i_2} + \sum_{j \in \mathcal{J}_f} \frac{X_{fij} \times Q_f(j)}{\Delta_f(i,j)}$$

$$\forall f, \forall l \in \mathcal{L}_f \qquad \sum_{i \in f_f(l)} Y_{fi} \leq D_{fl} \times n_f \tag{2.20}$$

$$\forall f, \forall l \in \mathcal{L}_f \qquad \sum_{i \in f_f(l)} Y_{fi} \geq D_{fl} \tag{2.21}$$

$$\forall f, \forall w \in \mathcal{W}_f \qquad \sum_{i \in g_f(w)} Y_{fi} \leq D_{fw} \times n_f \tag{2.22}$$

$$\forall f, \forall w \in \mathcal{W}_f \qquad \sum_{i \in g_f(w)} Y_{fi} \geq D_{fw} \tag{2.23}$$

$$\forall f \qquad NL^f_{min} \leq \sum_{l \in \mathcal{L}_f} D_{fl} \tag{2.24}$$

$$\forall f \qquad \sum_{l \in \mathcal{L}_f} D_{fl} \leq NL^f_{max} \tag{2.25}$$

$$\forall f \qquad NW^f_{min} \leq \sum_{w \in \mathcal{W}_f} D_{fw} \tag{2.26}$$

$$\forall f \qquad \sum_{w \in \mathcal{W}_f} D_{fw} \leq NW^f_{max} \tag{2.27}$$

$$\forall f, \forall j \in \mathcal{J}_f \qquad \sum_{i \in \mathcal{S}_f} X_{fij} \Gamma_f(i) \Pi_f(i,j) \frac{Q_f(j)}{\Delta_f(i,j)} \tag{2.28}$$

$$\leq \left(1 - \phi^{min}_f(j)\right) \sum_{i \in \mathcal{S}_f} X_{fij} \Gamma_f(i) \frac{Q_f(j)}{\Delta_f(i,j)}$$

$$\forall c \qquad \sum_f \sum_{i \in \mathcal{S}_f} \sum_{j \in \mathcal{J}^c_f} X_{fij} \Gamma_f(i) \Pi_f(i,j) \frac{Q_f(j)}{\Delta_f(i,j)} \tag{2.29}$$

$$\leq \left(1 - \phi^{min}_c(c)\right) \sum_f \sum_{i \in \mathcal{S}_f} \sum_{j \in \mathcal{J}^c_f} X_{fij} \Gamma_f(i) \frac{Q_f(j)}{\Delta_f(i,j)}$$

$$\forall f, \forall l \in \mathcal{M}_L^f \qquad D_{fl} = 1 \tag{2.30}$$

$$\forall f, \forall w \in \mathcal{M}_W^f \qquad D_{fw} = 1 \tag{2.31}$$

$$\forall f, \forall i \in \mathcal{M}^f \qquad Y_{fi} = 1 \tag{2.32}$$

$$\forall f, \forall \{i,j\} \in (\mathcal{S}_f \times \mathcal{J}_f) \text{ s.t. } \Delta_f(i,j) = 0 \qquad X_{fij} = 0 \tag{2.33}$$

$$\forall f, \forall i \in \mathcal{S}_f, \forall f' \in F(f), \forall i' \in \mathcal{S}_{f'} \text{ s.t. } \Delta(f,i,f',i') = 0 \quad N_{fif'i'} = 0 \tag{2.34}$$

The objective function (2.9) represents the minimization of the weighted[b] loss of glass (equivalent to the maximization of the weighted yield) of both:

- the cut of each final product into its corresponding father.
- The cut of the non root formats in their origin father.

The constraints (2.10) and (2.11) force the total number of used standards to belong to the specified range. This constraint exists also for each format through the inequalities (2.12) and (2.13). The equalities (2.14) represent that each real product is linked in a unique standard of its format.

The inequalities (2.15) and (2.16) imply the structural relationship between the Boolean variables Y_{fi} and X_{fij}: a standard is used if and only if there is at least one corresponding product.

The inequalities (2.17) and (2.18) imply the structural relationship between the variables Y_{fi} and $N_{fif'i'}$ with $f' \in F(f)$: two standards are linked is if and only if they are both selected.

The most interesting constraints are the equalities (2.19). The Figure (2.3) may help the reader to understand the notions of direct and indirect demands. We have drawn the parallel figure in terms of demand to the Figure (2.2) of the formats' tree.

[b]by the lost surface and by the specified coefficients λ_1 and λ_2

For every non-root format, its captures both the indirect demand of each format due to the relationships between formats in the formats' tree (black arrows) and the direct demand from the corresponding products' demand of the market (red arrows).

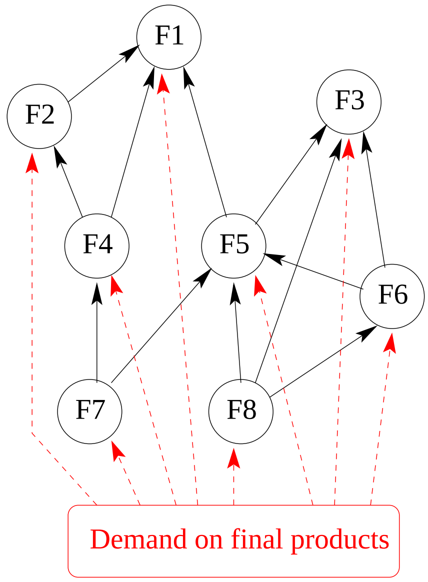


Figure 2.3: The direct (red arrows) and indirect (black arrows) demand in the formats' graph.

For every father i of a given non-root format f, it forces that the incoming flow (sum of both the direct affected demand $\sum_{j \in \mathcal{J}_f} \frac{X_{fij} \times Q_f(j)}{\Delta_f(i,j)}$ and the required number of glass sheets to be cut into derived formats $\sum_{(f_2 \in F(f))} \sum_{i_2 \in \mathcal{S}_{f_2}} N_{fif_2i_2}$) into the node is equal to the outgoing flow (required number of glass sheets of the different possible origins for the format $\sum_{(f_1 \text{ s.t. } f \in F(f_1))} \sum_{i_1 \in \mathcal{S}_{f_1}} N_{f_1 i_1 fi} \times \Delta(f_1, i_1, f, i)$).

The inequalities (2.20) and (2.21) imply the structural relationship between the variables Y_{fi} and D_{fl}: a length is used if and only if at least one corresponding standard is used. The inequalities (2.22) and (2.23) imply the structural relationship between the variables Y_{fi} and D_{fw}: a width is used if and only if at least one corresponding standard is used.

The inequalities (2.24), (2.25) and (2.26), (2.27) force the number of used lengths and widths to belong to the specified ranges. Finally, (2.30), (2.31) and (2.32) force the use of the mandatory lengths, widths and standards, whereas (2.35) and (2.36) forbid the use of impossible links both between a standard and a product and between two standards.

Last but not least, the (2.28) and (2.29) inequalities ensure that the solution satisfies the constraints of minimal yield per product and per customer.

Explanations about the computation of the yields

In the data set, we have used the notions of minimal yield per customer and per final product. In the model, we have written some constraints to take it into account (inequalities (2.28) and (2.29)). But what is the definition of a yield during some cutting operations?

- If we note $\Phi_f : \mathcal{J}_f \longrightarrow \mathbb{R}$ the function that associates to each final product j of format f its yield, the constraints (2.28) are equivalent to equations (2.35) and (2.36). Equation (2.36) defines the yield as one minus the proportional loss of glass. This proportional loss is the total lost surface of glass divided by the total surface required to produce the product j.

$$\forall f, \forall j \in \mathcal{J}_f \qquad \Phi_f(j) \geq \Phi_f^{min}(j) \tag{2.35}$$

  with:

$$\forall f, \forall j \in \mathcal{J}_f \qquad \Phi_f(j) = 1 - \frac{\sum_{i \in \mathcal{S}_f} X_{fij} \times \Gamma_f(i) \times \Pi_f(i,j) \times \frac{Q_f(j)}{\Delta_f(i,j)}}{\sum_{i \in \mathcal{S}_f} X_{fij} \times \Gamma_f(i) \times \frac{Q_f(j)}{\Delta_f(i,j)}} \tag{2.36}$$

- To simplify the notations, we also note $\Phi_f : \mathcal{S}_f \longrightarrow \mathbb{R}$ the function that associates to each standard product (determined by the optimization) $i \in \mathcal{S}_f$ its yield. Equation (2.37) defines this yield as one minus the proportional loss of glass *in the cutting operations of all the final products cut into* i.

$$\forall f, \forall i \in \mathcal{S}_f \qquad \Phi_f(i) = 1 - \frac{\sum_{j \in \mathcal{J}_f} X_{fij} \times \Gamma_f(i) \times \Pi_f(i,j) \times \frac{Q_f(j)}{\Delta_f(i,j)}}{\sum_{j \in \mathcal{J}_f} X_{fij} \times \Gamma_f(i) \times \frac{Q_f(j)}{\Delta_f(i,j)}} \tag{2.37}$$

- In the same way, if we note $\Phi_c : \mathcal{C} \longrightarrow \mathbb{R}$ the function that associates to customer $c \in \mathcal{C}$ its yield, the constraints (2.29) are equivalent to equations (2.38) and (2.39).

$$\forall c \in \mathcal{C} \qquad \Phi_c(c) \geq \Phi_c^{min}(c) \tag{2.38}$$

with:

$$\forall c \in \mathcal{C} \qquad \Phi_c(i) = 1 - \frac{\sum_f \sum_{i \in \mathcal{S}_f} \sum_{j \in \mathcal{J}_f^c} X_{fij} \times \Gamma_f(i) \times \Pi_f(i,j) \times \frac{Q_f(j)}{\Delta_f(i,j)}}{\sum_f \sum_{i \in \mathcal{S}_f} \sum_{j \in \mathcal{J}_f^c} X_{fij} \times \Gamma_f(i) \times \frac{Q_f(j)}{\Delta_f(i,j)}} \tag{2.39}$$

2.3.3 Remarks

If we use various values of coefficients to weight the two main parts of the objective function, we can optimize different scenarios:

- $\{\lambda_1 = 1; \lambda_2 = 0\}$ corresponds to the only minimization of the loss corresponding to the assignment of products of every format to selected fathers. In this case there is no link between selected fathers of various formats.

- $\{\lambda_1 = 0.5; \lambda_2 = 0.5\}$ corresponds to the global minimization of both:

  - The loss corresponding to the affectation of products of every format to selected fathers.

  - The loss corresponding to the derivation of non-root formats from root formats following the arrows of the formats' tree.

  In this case the link between formats is predominant (due to the backward information flow from derived formats) and the selected formats have corresponding dimensions (dimensions of a derived format tend to be multiples of its root).

- $\{\lambda_1 < 1; \lambda_2 > 0\}$ and $(\lambda_1 + \lambda_2 = 1)$ allows to find a trade off between the two objectives, depending on the industrial context. It is indeed often valuable to work with selected standards which are multiples of one another, in order to avoid inventory shortages and to simplify the inventory management. It can indeed maximize the risk pooling within references, because of a possible delayed differentiation of the smaller formats. However, it represents a potential loss in the direct demand satisfaction, when it is possible to satisfy it through a direct on-line cut of smaller formats.

2.4 Interpretation of the results

We have previously explained that we offer the possibility to optimize several constraint sets for a given data set. This part aims to highlight the interest to do so on two different applications.

2.4.1 Evolution of the yield depending on the maximal number of standards

Our goal is to provide an helpful tool and an easy interpretation of the interest to reduce or not the number of final standards.

For instance, we are going to solve one data set using several hypotheses. Our objective is to understand how much glass we could save by using more and more standard dimensions. Let us work with an unique format DLF and ten successive optimizations.

From the ten results' files, we can plot the following curve, representing the yield depending on the number of maximal standard products, from 1 to 10. Naturally, the yield is a monotone increasing function of the number of standard products.

Firstly, the possibility to compare the results of different scenarios of constraints sets on the same data set can be helpful for our user to support his decision. Knowing that each new standard product implies a more complicated inventory management and less risk pooling within final products, he can make sure to take the best decision.

N_{max}	Optimal yield
1	0.733973
2	0.85399
3	0.907709
4	0.941823
5	0.963184
6	0.978264
7	0.984417
8	0.9885
9	0.991866
10	0.993007

Table 2.1: Results of the ten cases of different constraints on N_{max}

Secondly, for each given value of number of standard products, our program makes sure to create the standard dimensions in order to obtain the proved optimal yield. Table (2.1) gives the optimal yield for each constrained scenario. We plot the result on Figure (2.4), which highlights the marginal cost of an increase of yield: the more standards we work with, the bigger the yield, but the more standards we use, the less the increase of yield when we add a new one.

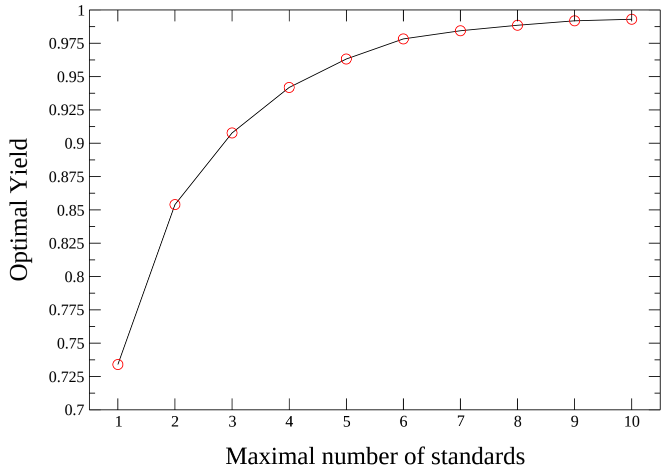


Figure 2.4: Optimal yield as a function of the maximal number of standards

2.4.2 Impact of the weights λ_1 and λ_2 in the objective function

In this example we would like to underline the importance of the weights that we use in the objective function to choose whether we want to consider or not the relationships between the standard products from different formats.

Using the same data set, we optimize successively four hypotheses. Globally, we increase gradually the weight of the cutting operations betweens standards and final products by creating four cases (cf. Table (2.2)).

Case	A	B	C	D
λ_1	0.5	0.6	0.8	1
λ_2	0.5	0.4	0.2	0

Table 2.2: Creation of four cases of different couples $\{\lambda_1; \lambda_2\}$

By convention, we note $\texttt{Yield}_2$ the yield of the cutting operations in the formats' tree (every non-root format is derived from another) and $\texttt{Yield}_1$ the one corresponding to the cutting operations between one standard and its derived products. The global yield is noted $\texttt{Yield}$. We compare the results of the four cases in Table (2.3), and we plot them in the Figure (2.5).

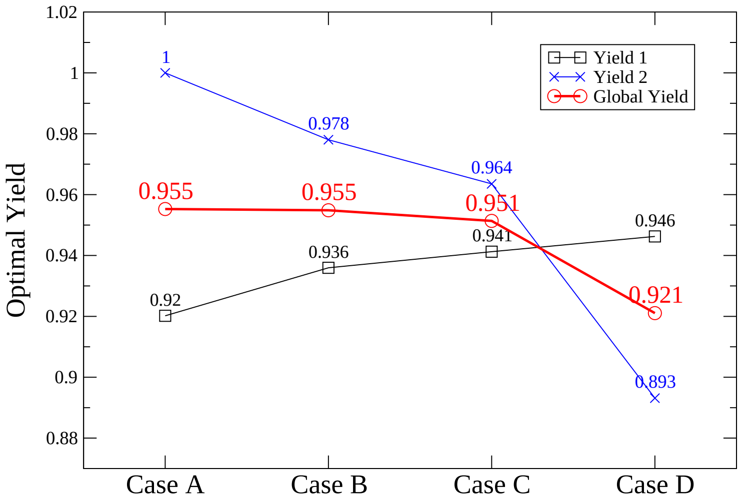


Figure 2.5: Plot corresponding to Table (2.3)

Case	A	B	C	D
$\mathtt{Yield}_1$	0.920181	0.935947	0.941255	0.946271
$\mathtt{Yield}_2$	1	0.97803	0.963531	0.893138
$\mathtt{Yield}$	0.955263	0.954851	0.951375	0.92106

Table 2.3: Optimal yields of the four cases of different couples $\{\lambda_1; \lambda_2\}$

In case A, we optimize equally the two set of operations. We increase gradually the weight λ_1 (with coefficients whose sum equals one) from case A to case D.

The best global yield corresponds to the global optimization (case A). We observe naturally that case A gives the best $\mathtt{Yield}_2$ and case D the best $\mathtt{Yield}_1$. On this example we discover the trade-off of designing the standards of non-root formats depending on either the standards of the father standard (in the formats' tree) or on the final products.

The table (2.4) shows the optimal corresponding dimensions. Case A gives a $\mathtt{Yield}_2$ equal to one because each DLF is perfectly derived from a PLF. In this case, we can cut exactly four DLF n°2 in the PLF n°1 and three DLF n°1 in PLF n°2.

It is important for the user to understand the impact of the coefficients $\{\lambda_1; \lambda_2\}$ on the results.

Format	Case A		Case B		Case C		Case D	
PLF n°1	5600	2800	5880	3160	5600	2800	5880	3160
PLF n°2	5520	3030	5600	3030	5520	3160	5600	2800
DLF n°1	1840	3030	1840	3030	1840	3030	1840	3030
DLF n°2	1400	2800	1400	2800	1400	2800	1400	2800

Table 2.4: Detailed results of the study of the impact of the weights $\{\lambda_1; \lambda_2\}$ on the optimal standards

2.4.3 Complexity of the optimization problem

It appears from our computational experience (we used real-life data sets and a commercial code [ILOa]) that the computational time required to reach the optimal solution rises exponentially in the number of boolean variables.

First of all, the larger the ranges defining each format (or the smaller the used precision), the longer the CPU time. We decomposed the resolution of our industrial problem into two successive steps because a unique one would have created such a big problem that no existing commercial code would have been efficient enough to solve it in a reasonable time. We explained in section (2.3.2) why this decomposition did make sense. It is thus important to define with the application user a precision as big as possible.

As a first step, we only focus on the first objective (i.e. $\lambda_2 = 0$). In this case, boolean variables $N_{f_1 i_1 f_2 i_2}$ are not used and the complexity of the formats'tree is not significant because each format may be considered and solved independently in the second step of the optimization. Therefore, the only significant parameter is the number of products associated to each format. We found that these cases were relatively easy to solve. Small data sets with dozens of products and several formats were optimally solved in few seconds, while bigger cases with hundreds of products and several formats were solved in few minutes.

Finally, the number of relationships (i.e. arrows) in the formats'tree becomes also a significant parameter when the second objective is minimized (i.e. $\lambda_2 > 0$).

This comes from the fact that the number of implicit boolean variables $N_{f_1 i_1 f_2 i_2}$ increases in this number. It took Cplex several hours to solve optimally cases with few formats and a dozen of products associated to each format.

The performance of our model was considered sufficient enough to fulfil our industrial partner need. However, it appears that interesting outlooks of our research lie in the determination of decomposition methods matching the structure of the formats'tree in order to decrease the computation time.

2.5 Gain and Conclusion

Our model and software have been designed according to the user's expectations. In this final version, we offer an open object-oriented model that captures every scenario we have imagined so far.

Based on our customer forecasts, we work on several formats and we aim at determining a limited number of standard products in each format. Our objective is naturally to minimize the glass loss. We capture the cutting operations of the links both between each standard and its associated final products and between the standards of different formats.

But what is the financial interest of this optimization tool? The estimated savings of the year 2003 have been such a surprise that our user asked for more. To compute them, we compare our model to the method used so far by our industrial partners, which was an heuristic method developped on MS-Excel. Last year, we have estimated the gain of this work on several cases by comparing the results of the human resolution and of the tool on different cases with identical data and constraints sets. The figure (2.6) summarizes this interesting result.

Globally, we have shown on average that around 2.48 % of the global cutting operations' loss can be saved using our optimization tool. In addition, the duration of the determination process by the user has been divided by two : everything can

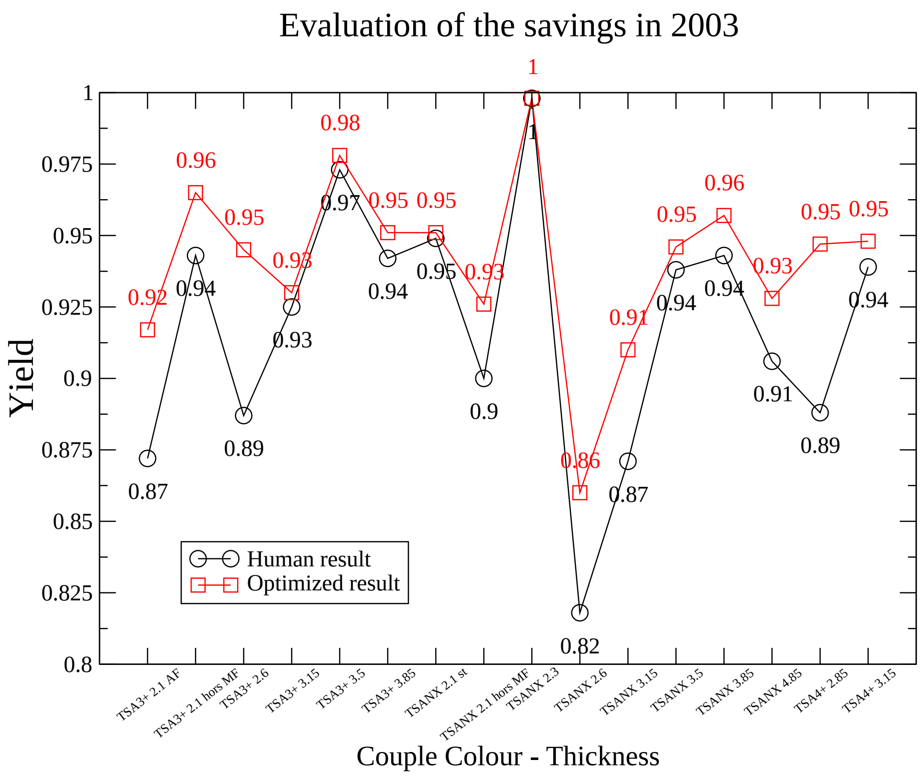


Figure 2.6: Results of the comparison between human and optimized results on the 2003 data.

now be accomplished within one week instead of two weeks formerly.

If we basically apply these savings to the global yearly sales of Saint-Gobain Glass to Saint-Gobain Sekurit (around 550.000 Tons for a total value of more than 200 M€), it represents tremendous savings of approximately 5 M€ a year.

Of course, it is always delicate to evaluate the exact real financial interest of any optimization tool. In this case, it just appears that the cost of the commercial solver used by the optimization method (around 7.500 €) is worth being invested. In addition, this standard product determination is also a crucial point for saving on the transportation cost. On average, a product designed for the automotive market travels around 450 kilometers between the plant of Saint-Gobain Glass and the customer. Knowing that the transportation cost of the glass is around 1€/Tkm, the decrease of the loss of glass in our customers' plants becomes quickly a strategic issue. . .

It could also be interesting to study the impact on the inventory management of the decrease in the number of references at a given fixed yield. This could provide a real savings estimation on the inventory costs. For instance, it would highlight the interest of our two optimization goals (cf. section (§ 2.3.1)): it could show the interest of risk pooling within different references which is possible when the formats' graph is taken into account (ex: a DLF is a perfect multiple of a PLF).

We hope that this note clarifies the basic notions of the work. Indeed, we think that a good understanding of the model is required to be able to capture the relevance of the tool as well as to criticize the results of any computation.

We give the user the possibility to take into account a great number of constraints. We just want to remind here that too many constraints could make a problem unsolvable. Our advice is to start to optimize a data set with a reasonable constraint set. Then, it is always interesting to add some new constraints to be able to point out how much yield do we lose.

This way of thinking allows a good understanding of the trade-off between the yield and the cost of managing a high number of standard products. From now on, our decision tool will give to the manager the power to compute as many times as necessary a quick calculation (it lasts few minutes for the bigger cases) of several hypotheses.

This concept of successive optimization of different constraints set scenarios on a given data set is a powerful option. We give in section (§ 2.4) two possible applications of this opportunity, but each user can define his or her own needs.

Chapter 3

A first approach of logistic platform design

3.1 Introduction

In this chapter, we deal with a simple strategical distribution issue: given a supply chain with defined plant-platform flows and platform-customer flows (so-called upstream and downstream flows), how do determine both the optimal number and positions of logistic platforms?

Before exploring the literature review (§ 3.2.2), we focus on a simple and unrealistic model (§ 3.2.1): given a set of customers (defined by a position and a deterministic demand on a single product) and some transportation and platform opening costs, what is the optimal way to serve each customer by one platform? Understanding this location-allocation problem will allow the reader to discover the highly combinatorial structure of this apparently simple question.

Based on existing results, we propose an original integration of upstream flows (we introduce two different scenarios) in section (§ 3.2.3). Firstly, we propose a mono-product model produced by several known capacitated plants. Secondly, we take into account customers who ask for mixed orders made of products whose origin plant is unique and known (uncapacitated and located). We provide a heuristic algorithm (§ 3.2.4) that tackles the problem. Results of this chapter are used for customer aggregation in chapter (6).

3.2 Theoretical Issues

3.2.1 Basic Problem

In this chapter we study a basic supply chain optimization problem. Given a set of customers whose demands and locations are known, we would like to determine what is the best way to fulfill them by a set of platforms. Given that platforms are uncapacitated, each customer is served by exactly one platform. We know both the platform opening costs and the transportation costs.

Thus, the question is: **what is the cheapest set (number and locations) of platforms that fulfills the customers' demand?**

For instance, Figure (3.1) presents a supply chain made of 80 customers (black points). To serve them, we propose a set of three platforms (red points) whose relationships with customers are described.

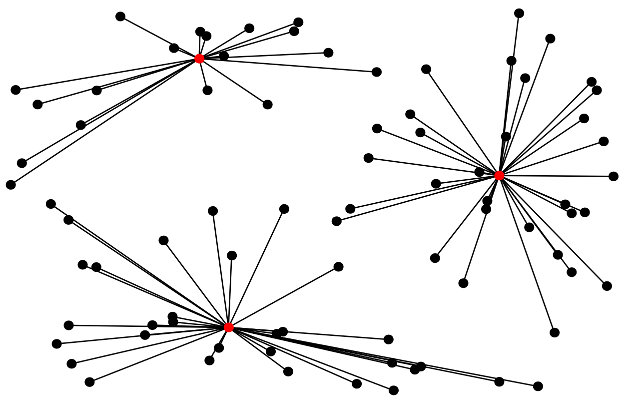


Figure 3.1: Example of supply chain with 80 customers and 3 platforms

Notation

We introduce some notation to model our problem:

- we work with Cartesian coordinates and the Euclidean distance[1] in the plane $\mathbb{R}^2$.

  1. we have a set of m different[2] customers $\mathcal{C} = \{C_j,\ j \in \{1, \dots, m\}\}$. Each customer C_j is characterized by its position $\{X_j, Y_j\}$ and its demand $\alpha_j > 0$ on a given time period.
  2. we define a set of platforms $\mathcal{P} = \{P_i,\ i \in \{1, \dots, n\}\}$. The positions of platforms are unknown.
  3. $\forall i$, $\forall j$ we denote f_{ij} the flow that is sent from the platform P_i to the customer C_j. Of course, we find that: $\forall i \quad \sum_i f_{ij} = \alpha_j$.

- we introduce the following costs:

  1. the individual platform opening cost is a concave function of the total number of platforms. We denote it $C_F(n)$ (see remark (1)).
  2. the transportation cost is a constant C_T (in €/unit/km).

Remark 1 *The fixed cost associated with any opening of platform clearly depends on the size of the platform. Given a set of customers, the more platforms we use, the smaller they are. Thus we consider that* C_F *is a strictly non negative convex function. In addition, we assume that* $n \times C_F(n)$ *is monotone increasing in n. If this property were not assumed, it would be cheaper to build smaller platforms than fewer bigger ones.*

[1]The Euclidean distance is defined by: $\forall A = \{x_a, y_a\}$ $\forall B = \{x_b, y_b\}$ $\quad d(A,B) = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2}$

[2]there is no couple of customers with identical positions

Definitions

Definition 12 *For any set of platforms $\mathcal{P}$, we denote $\bar{P}(j)$*[3] *the platform that we choose to serve customer C_j.*

Naturally, the cost $C^{[n]}$ associated with any solution set of n platforms is the sum of the transportation costs $C_T^{[n]}$ and of the platform opening ones $n \times C_F(n)$. Using the notation introduced before, we may easily write down the formula of transportation cost:

$$C_T^{[n]}(\{P_1, \dots, P_n\}) = C_T \times \sum_j \left(\alpha_j \times d(C_j, \bar{P}(j))\right)$$

Let us now decompose the original question into easier subproblems: given a number of platforms, are we able to determine the positions that minimize the transportation cost and thus the global one?

Definition 13 *We note $\mathcal{P}(n)$ the sub problem of the minimization of the global cost $C^{[n]}$ for a given n. The solution may not be unique and thus we note $\mathcal{S}^{[n]}$ its solution set*[4]*. The corresponding minimal value of the cost is $C^{*[n]}$.*

$$C^{*[n]} = \min_{\mathcal{P}} \left(C^{[n]}(\mathcal{P}) \right)$$

We note the optimum transportation cost: $C_T^{*[n]} = C^{*[n]} - n \times C_F(n)$.

From now on, we may easily write down the original question. Among all optimal solutions of subproblems $\mathcal{P}(n)$, we want to determine the cheapest. To keep the same notation, we want to determine the set of solutions $\mathcal{N}$ whose cost is the global optimum $C^* \in \mathbb{R}$:

$$\forall n^* \in \mathcal{N} \qquad C^{*[n^*]} = C^* = \min_{n \in \mathbb{N}^*} C^{*[n]} \tag{3.1}$$

[3] $\bar{P} : \{1, \dots, m\} \to \mathcal{P} \quad \forall j \quad \bar{P}(j) = \{P_i \in \mathcal{P} \text{ s.t. } f_{ij} > 0\}$; we denote its position $\{\bar{x}(j), \bar{y}(j)\}$

[4] that means $\forall \mathcal{P} \in \mathcal{S}^{[n]} \quad C^{[n]}(\mathcal{P}) = C^{*[n]}$

To keep on simplifying the problem, let us use some properties of the problem.

Properties of solutions

The formulation of our model may be simplified by showing intuitive basic results.

Proposition 14 *For a non negative transportation cost and under no capacity constraints, the optimal cost is obtained only if the platform serving any customer is the closest one in terms of Euclidean distance.*

Proposition 15 *If $C_T > 0$, then the optimal transportation cost of the subproblem $\mathcal{P}(n)$ is strictly less than the one of $\mathcal{P}(n-1)$.*

$$\forall\, 1 < n \leq m, \qquad C_T^{*[n]} < C_T^{*[n-1]}$$
$$\forall\, n \geq m, \qquad C_T^{*[n]} = 0$$

Proof: Let us prove the first assertion: For all $n \in \{1, \ldots, m\}$, for any set S solution of $\mathcal{P}(n-1)$, we define the set F of customers whose platform is not located at their own position. We have assumed that $(n-1) < m$ and that all customers are different, so there is at least one element in this set. Let us choose the bigger customer of this set C_{j_0} such as $\alpha_{j_0} = \max_{j \in F} \alpha_j$. It is served by the platform $\bar{P}(j_0) \in S$. Then, we create a set S' of n platforms by adding to S an n^{th} platform located on this particular customer. It comes $C_T^{[n]}(S') = C_T^{*[n-1]} - C_T \times \alpha_{j_0} \times d(\bar{P}(j_0), C_{j_0})$. Knowing that demand and transportation costs are strictly non negative and that $C_{j_0} \in F{:}d(\bar{P}, C_{j_0}) > 0$, we obtain $C_T^{[n]}(S') < C_T^{*[n-1]}$. In addition, by definition: $C_T^{*[n]} \leq C_T^{[n]}(S')$. The second assertion is more obvious: when we have as many platforms as customers, the optimal cost is null ($C_T^{*[m]} = 0$). For any $n > m$, there is $(n-m) > 0$ useless platforms.

From the proposition (15) we can reduce the domain of the optimal number of platforms. Obviously, it is at least one and at most the number of customers. The same way, we obtain inequalities on global costs.

Proposition 16 *If* $\forall n > m \; C_F(n) > 0$*, then :*

$$\min_{n \in \mathbb{N}^*} C^{*[n]} = \min_{n \in \{1,\ldots,m\}} C^{*[n]} \tag{3.2}$$

Proposition 17

$$\forall n > 1, \quad C^{*[n]} - \big(n \times C_F(n) - (n-1) \times C_F(n-1)\big) < C^{*[n-1]} \tag{3.3}$$

Remark 2 *If* $\forall n,\ C_F(n) = C_F > 0$ *(constant function), we have:*

$$C^{*[n]} - C_F < C^{*[n-1]}$$

To be more accurate on the structure of optimal solutions of subproblems $\mathcal{P}(n)$, we need to introduce some new mathematical notions.

Additional mathematical notions: Weber point and Voronoi diagrams

Definition 18 *The Weber point of a weighted point set is the point that minimizes the sum of all distances to the weighted points. Here we note* $\mathcal{W}_B : (\mathbb{R}^2 \times \mathbb{R})^K \to \mathbb{R}^2$ *the function that associates to a set of weighted points S its weber point* $\mathcal{W}_B(S)$*.*

$$\forall S = \{\{P_k; w_k\},\ k \in [1, K]\} \qquad \mathcal{W}_B(S) = W \quad \text{s.t.}$$

$$\sum_k w_k \times d(P_k, W) = \min_{U \in \mathbb{R}^2} \big(\sum_k w_k \times d(P_k, U)\big)$$

Remark 3 *The Weber point is different from the center of gravity that minimizes the sum of the distance squared.*

Weber points have been studied for a long time. The first formulation of the problem for m = 3 points is by Fermat (1600). Then it was studied under different assumptions: by Cavalieri (1647, three points vertexes of a triangle); Fagnano (1775, m=4); Tedenat (1810); Steiner (1837). However, Weber (1909) was probably the first who stated the problem with the purpose of minimizing the sum of the transportation costs from the plant to sources of raw material and to the market center: hence, this problem with m points has become known as the mutlisource Weber problem and is NP-hard. The history of the Weber problem is well documented in [Wes93].

Although it has no good exact solution (the solution point is a high degree polynomial in the size of the point set ([Baj88], [CM69]) one can easily solve it approximately by steepest descent (see [Wei37]).

Let us recall the *Steepest Descent Algorithm*, given a function $f(x)$ defined for $x \in \mathbb{R}^p$ that we aim at minimizing:

1. Given x^0, set $k = 0$.
2. $d^k = -\nabla f(x^k)$. If $d^k = 0$, then STOP.
3. Solve $\min_\alpha f(x^k + \alpha d^k)$ for the step size α^k, perhaps chosen by an exact or inexact line search (in this work we use a bisection line-search method).
4. Set $x^{k+1} \leftarrow x^k + \alpha^k d^k$, $k \leftarrow k + 1$. GO TO (2).

Claim 19 *We know how to solve optimally $\mathcal{P}(1)$, which is a classical non linear unconstrained problem.*

Figure (3.2) shows an example of Weber point for a set of 100 points. We plot on this figure the level curves of the minimized function of the sum of all distances (here all weights are equal to 1).

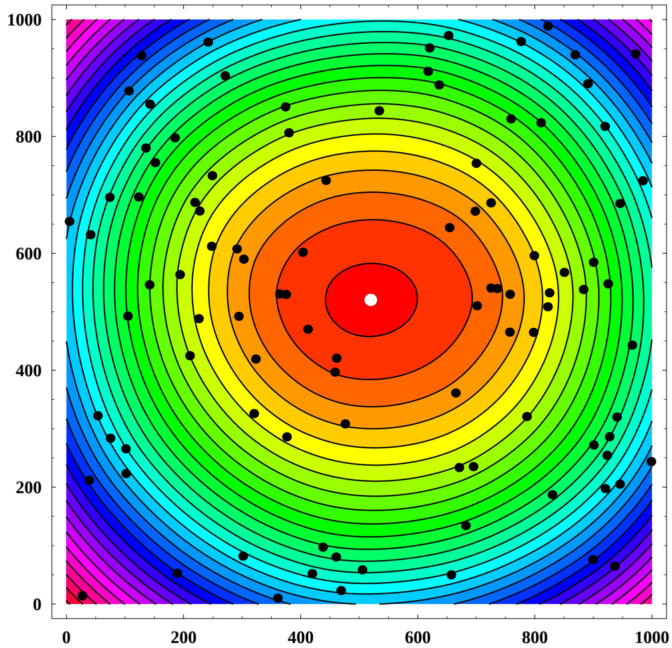


Figure 3.2: Example of Weber Point on a set of 100 points

Definition 20 *Given a set $\mathcal{S}$ of p distinct points, Voronoi diagram is the partition of the plane into p polyhedral regions $\mathsf{vo}_{\mathcal{S}}(\mathsf{P})$, $\forall \mathsf{P} \in \mathcal{P}$. Each region $\mathsf{vo}_{\mathcal{S}}(\mathsf{P})$, called the Voronoi cell of P, is defined as the set of points in $\mathbb{R}^2$ which are closer to P than to any other points in $\mathcal{S}$, or more precisely:*

$$\mathsf{vo}_{\mathcal{S}}(\mathsf{P}) = \{\, \mathsf{X} \in \mathbb{R}^2 \quad \text{s.t.} \quad \mathsf{d}(\mathsf{X},\mathsf{P}) \leq \mathsf{d}(\mathsf{X},\mathsf{Q}) \quad \forall\, \mathsf{Q} \in \mathcal{S} - \mathsf{P} \,\} \tag{3.4}$$

The set of all Voronoi cells and their faces forms a cell complex. The vertexes of this complex are called the Voronoi vertexes, and the extreme rays (i.e. unbounded edges) are the Voronoi rays. Figure (3.3) shows an example of Voronoi diagram for a set of 20 points. In our case, the interest of such a structure is obvious, given that the Voronoi diagram of any set of platforms gives us the zone of customers that each platform serves.

Thus for each platform we create the set of customers belonging to its Voronoi cell (i.e. the set of customers for which the closest platform is this one). We denote

$$\mathcal{C}'^{vo}_{\mathcal{P}}(P_i) = \{\, C_j \quad \text{s.t.} \quad C_j \in vo_{\mathcal{P}}(P_i) \quad \forall j \,\}$$

However, it is possible to find a customer C_{j_0} being exactly on the border within Voronoi cells of P_{i_1} and P_{i_2}. In this case, $C_{j_0} \in \left(\mathcal{C}'^{vo}_{\mathcal{P}}(P_{i_1}) \cap \mathcal{C}'^{vo}_{\mathcal{P}}(P_{i_2})\right)$.

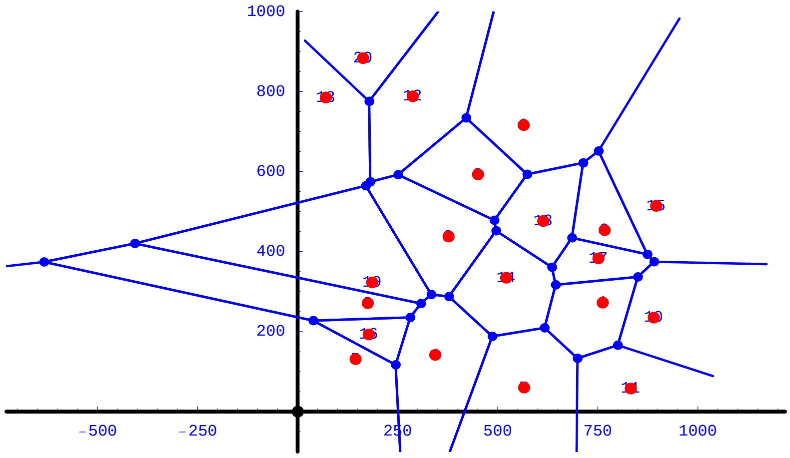


Figure 3.3: Example of Voroinoi diagram on a set of 20 points

To overcome this difficulty, we create arbitrarily new sets $\mathcal{C}^{vo}_{\mathcal{P}}(P_i)$ that have the following nice properties $\left(\forall i_1,\ \forall i_2 \neq ß_1 \qquad (\mathcal{C}^{vo}_{\mathcal{P}}(P_{i_1})) \cap (\mathcal{C}^{vo}_{\mathcal{P}}(P_{i_2})) = \emptyset\right)$ and $\left(\cup_i \left(\mathcal{C}^{vo}_{\mathcal{P}}(P_i)\right) = \mathcal{C}\right)$.

To do so, we define the set of platforms whose cell contains each customer ; $\forall \mathcal{P} = \{P_i,\ i \in \{1, \ldots, n\}\}$, we define:

$$\forall j,\ \Lambda(j) \subset \mathbb{N} \quad \text{s.t.} \quad \begin{cases} C_j = \cap_{i \in \Lambda(j)} \left(\mathcal{C}'^{vo}_{\mathcal{P}}(P_i)\right); \\ \forall i \notin \Lambda(j) \quad C_j \notin \mathcal{C}'^{vo}_{\mathcal{P}}(P_i). \end{cases}$$

These sets simplifies the forthcoming definition.

Definition 21 *We define sets* $\mathcal{C}^{vo}_{\mathcal{P}}(P_i)$ *such that* $\forall j, \exists! i \in \Lambda(j) \quad \text{s.t.} \quad C_j \in \mathcal{C}^{vo}_{\mathcal{P}}(P_i)$. *When* $|\Lambda(j)| > 1$, *the criterion to choose which platform to use to serve the customer* *is arbitrary and does not have any impact on the cost.*

From now on, for any set of platforms $\mathcal{P}$ we may compute the transportation cost using the previous definition:

$$\forall n \in \mathbb{N}^* \qquad C_T^{[n]}(\mathcal{P}) = C_T \times \sum_i \sum_{C_j \in \mathcal{C}_{\mathcal{P}}^{vo}(P_i)} \left(\alpha_j \times d(P_i, C_j)\right)$$

New properties of our problem

Based on the definitions of weber point and Voronoi diagram introduced above, we can deduce an important property of any optimal solution of the subproblem $\mathcal{P}(n)$: each platform is the weber point of the customers belonging to its Voronoi cell.

Proposition 22 *If* $C_T > 0$*, then* $\forall n \in \mathbb{N}^*$*,* $\forall \mathcal{P} = \{P_1, \ldots, P_n\} \in \mathcal{S}^{[n]}$*, we have:*

$$P_i = \mathcal{W}_B(\mathcal{C}_{\mathcal{P}}^{vo}(P_i))$$

Proof: Let us assume $\exists\, i_0$ s.t. $P_{i_0} \neq \mathcal{W}_B(\mathcal{C}_{\mathcal{P}}^{vo}(P_{i_0}))$. Thus we may define a new point $W = \mathcal{W}_B(\mathcal{C}_{\mathcal{P}}^{vo}(P_{i_0}))$. Then the set of n platforms $\mathcal{P}' = \{P_1, \ldots, P_{i_0-1}, W, P_{i_0+1}, \ldots, P_n\}$ has a smaller cost than $\mathcal{P}$: $C^{[n]}(\mathcal{P}') < C^{[n]}(\mathcal{P})$ and thus $\mathcal{P} \notin \mathcal{S}^{[n]}$, which is a contradiction.

Let us now discover how this basic problem is treated in the scientific literature, in which it is called the facility location problem, the multisource Weber problem as well as the location-allocation one.

3.2.2 Literature Review

State of the art

Location problems do not lack variety: depending on several hypotheses, it exists many kind of different nature problems. A problem is characterized by an objective function as well as its decision variables and constraints.

The location of facilities is a problem which exists in the private sector such as the location of plants, warehouses as well as in the public sector such as the location of hospitals, health centers, police stations etc. For the later one, different variant versions exist: for instance, we may try to create facilities within known potential points in order to cover a set of fixed points. Given a number of opened facilities, Hakimi proposed in 1964 the p-center problem which aims to minimize the maximal distance from a center to a customer ; Church and Revelle (1974) introduced the maximum covering problem in which covered demand is maximized. In 1989 Church and Meadous proposed a solution to the location set covering problem in which they aim at minimizing the number of opened facilities to cover the demand.

Many authors worked on the p-median problem, in which facilities can only be located on customer sites (see [BCTL83], [Das95] and [CDS02]).

As we saw earlier, the Weber problem is highly complicated due to the fact that facilities may be located inside a continuous set.

The multisource Weber problem introduced earlier may be generalized to the location-allocation model which is formulated as follows:

$$\text{Minimize} \quad \sum_i \sum_j w_{ij} \times d(P_i, C_j) \tag{3.5}$$

subject to:

$$\begin{aligned} \forall j, \quad \sum_i w_{ij} &= \alpha_j \\ \forall i, \forall j, \quad w_{ij} &\neq 0 \end{aligned}$$

where w_{ij} is the quantity assigned from facility i to fixed point j also denoting the allocation of customers to the open facilities. The problem is referred to as the multisource Weber problem when all quantities or weights are equal to unity, and as the generalized multisource Weber problem when they are unequal. Under the assumption that there are no capacity constraints at the new facilities, it can be shown that the demand at each point is satisfied at minimum cost by the nearest facility.

Cooper [Coo63] proves that the objective function (3.5) is neither concave nor convex, and may contain several local minima (confirmed by [EWGC71]). Hence, the multisource Weber problem falls in the realm of global optimization problems.

Many heuristic methods have been proposed in the literature beginning with the well-known iterative location-allocation algorithm of Cooper [Coo64] to solve the multisource Weber problem. Cooper's heuristic generates $\mathtt{p}$ subsets of fixed points and then solves each one using the exact method for solving a single-facility location problem.

Rosing ([Ros92]) divides the set of fixed points into non-overlapping convex hulls and generates the list of all feasible convex hulls where each fixed point must belong to exactly one of those convex hulls. The cost function associated with each convex hull is computed as a single Weber problem. This method produces the optimal solution to problems with up to 30 fixed points and 6 facilities. More recently, [Kra97] uses column generation to solve bigger problems up to 287 customers and 100 services. Given the restrictive use of these optimal methods, heuristics seem to be the only way forward to solve problems of larger size.

Brimberg and Mladenovic [BM96a] adopt a tabu search approach to the problem. Hansen et al. [HMT98] solve the continuous location-allocation problem via the p-median problem by considering all fixed points as potential facility sites. A genetic algorithm is designed by Houck et al. [HJK96] to solve this continuous location-allocation problem.

A variable neighborhood algorithm which uses Cooper's alternate algorithm to carry out the local descent is designed by Brimberg and Mladenovic [BM96b]. Finally, Brimberg and al. [BHMT00] compare and improve heuristics for solving the uncapacitated multisource Weber problem.

Recently, Gamal and Salhi [GS01] developed constructive heuristic which guides the search to generate better initial solutions within the classical multi-start heuristic. They consider the sparsity of the previously used locations while introducing strategies for forbidding and freeing some of these locations. In [GS03] they propose a learning scheme which uses previous solutions to discretize the continuous space into well-defined cells. This cells-based technique takes into account frequency of occurrence of already found configurations as well as the compatibility of these configurations. Computational results show that the cells-based approach can improve on the solutions found by the multi-start, especially for the problem with equal weight, without a considerable amount of computational effort.

Exhaustive reviews of the model under deterministic assumptions may be found in [OD98] and [KD05]. Extensions of the model may be found: first of all, some authors [BMS04] are working on the multisource Weber problem with constant opening costs. Otherwise, problems of facility location under uncertainty seem to be an interesting outlook [Sny04]. For instance, Shen and al. ([SCD04]) consider a joint location-inventory problem involving a single supplier and multiple retailers. Associated with each retailer is some variable demand. Due to this variability, some amount of safety stock must be maintained to achieve suitable service levels. However, risk-pooling benefits may be achieved by allowing some retailers to serve as distribution centers (and therefore inventory storage locations) for other retailers. The problem is to determine which retailers should serve as distribution centers and how to allocate the other retailers to the distribution centers. They formulate this problem as a nonlinear integer-programming model and then restructure it into a set-covering integer-programming model.

Last but not least, location models of facilities in the continuous plan integrating more than a simple stage supply chain are not to our knowledge existent in the literature. We found for instance two important references of location problem within potential discrete positions. Geoffrion and al. [GG74] are the first authors to introduce a location-allocation problem including capacitated supplying plants. Going further, some authors [JP01] integrate raw material replenishments: using the Lagrangian relaxation, they develop heuristics solving industrial size problems.

In the following, we contribute to a better understanding of the location problems in the continuous plan for two-stages supply chains.

Illustration of a result

Based on previous results, we have implemented existing methods to illustrate the trade-off between opening and transportation costs. In a supply chain in which each customer must be served by a platform, given a set of customers and knowing transportation and platform opening costs, what are the optimal number and locations of platforms ?

We provide the example of a realistic supply chain made of 80 customers. To simplify, all customers are considered identical. We create three scenarios of different individual platform opening costs $C_F(n)$, and we obtain three different results as indicated on Figure (3.4).

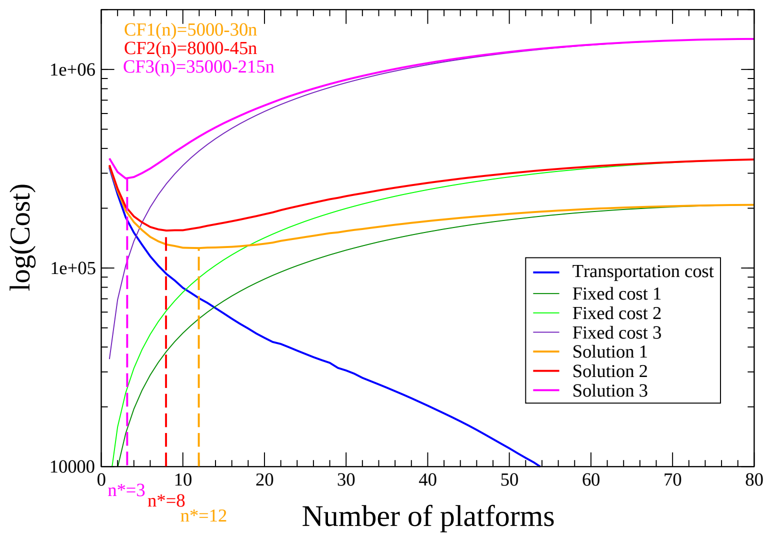


Figure 3.4: Results of three different scenarios on a supply chain made of 80 customers.

3.2.3 Integrating upstream transportation flows

Based on existing results, we propose an original integration of upstream flows, and we introduce two different scenarios. Firstly, we propose a mono-product model produced by several known capacitated plants. Secondly, we take into account customers who ask for mixed orders made of products whose origin plant is unique and known (uncapacitated and located).

A mono-product supply chain with capacitated plants

We consider a set of plants producing a unique product. Given a set of customers whose consumptions and locations are known, we would like to determine what is the best way to fulfill them through a set of platforms. We assume that each customer must be served by one platform which is replenished by plants. We know both the platform opening costs and the upstream and downstream transportation costs.

Thus, the question is: **what is the cheapest set (number and locations) of platforms that fulfills the customers' demand?**

For instance, we present a supply chain made of 80 customers (black points) and 5 identical uncapacitated plants (green points). To serve them, we propose a set of three platforms (red points) whose relationships with customers and plants are described on Figure (3.5). In this case, two plants are useless.

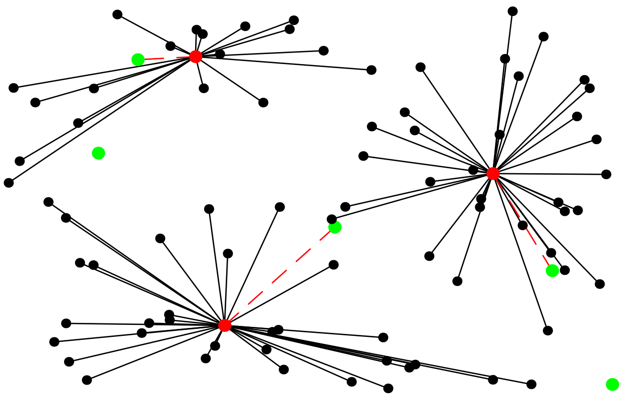


Figure 3.5: Example of supply chain with 80 customers, 5 plants and 3 platforms

We introduce some new notation in addition to the one introduced in 3.2.1. We denote $\mathcal{U} = \{(U_k, C_k),\ k \in \{1, \ldots, p\}\}$ the set of plants whose locations are given. Each plant U_k has a production capacity C_k on a given time period. We denote $\mathcal{U}^*$ the corresponding set of plants with infinite capacities.

Of course, to make the problem solvable, we assume that plants have enough capacity to fulfill the global demand.

$$\sum_k C_k \geq \sum_j \alpha_j \tag{3.6}$$

We assume that each customer must be served by one and only one platform. Thus, the number of platforms is strictly non negative. For each platform P_i of a set $\mathcal{P}$, we use the set of served customers $\mathcal{C}^{vo}_{\mathcal{P}}(P_i)$. These sets form a partition of $\mathcal{C}$.

Definition 23 *For any platform P_i of a set $\mathcal{P}$, we denote the set of plants replenishing it $\{(U_k, q^*_{ki} > 0)\}$, with q^*_{ki} optimal values of the following classic linear programming model: we denote $q_{ki} \in \mathbb{R}^+$ the real non negative variable capturing the quantity sent from plant U_k to the platform P_i. We compute these variables by* *solving (using GLPK, see [GNU]):*

$$\begin{aligned} & \mathrm{Min}\Big(\sum_k \sum_i (q_{ki} \quad \times \quad d(U_k, P_i))\Big) \\ \forall k \quad & \sum_i q_{ki} \quad \leq \quad C_k \\ \forall i \quad & \sum_k q_{ki} \quad = \quad \sum_{j \in \mathcal{C}^{vo}_{\mathcal{P}}(P_i)} \alpha_j \end{aligned}$$

We underline that we force every flow sent to the customer to go through a platform. Otherwise, it would be cheaper to send it directly from the plant to the customer, but it would not capture the reality of make to stock production.

Naturally, the cost $C^{[n]}$ associated with any solution set of n platforms is the sum of the transportation costs $C_T^{[n]}$ and of the platform opening ones $n \times C_F(n)$. Using the notation introduced before, we may easily write down the formula of transportation cost:

$$C_T^{[n]}(\mathcal{P}) = C_T \times \sum_k \Big(\sum_i q^*_{ik} \times d(U_k, P_i) + \sum_j \alpha_j \times d(\bar{P}(j), C_j)\Big)$$

Definition 24 *We define the constant transportation cost $\bar{C}_T$ corresponding to the case in which all flows from plants to customers are direct (there are no platforms):*

$$\bar{C}_T = C_T \times \sum_k \sum_j \big(\beta^*_{kj} \times d(U_k, C_j)\big)$$

with $\beta^*_{kj} > 0$ the optimal solutions of variables q_{kj} in the linear programming model introduced before, using either $\mathcal{P} = \mathcal{U}^*$ (one platform per plant and $\forall k,\ C_k = \infty$: upstream flow is null) or $\mathcal{P} = \mathcal{C}$ (one platform per customer: downstream flow is null). Thus,$\bar{C}_T = C_T^{[p]}(\mathcal{U}^*) = C_T^{[m]}(\mathcal{C})$

The formulation of our model may be simplified by showing intuitive basic results.

Proposition 25 *If $C_T > 0$, then the optimal transportation cost of the subproblem $\mathcal{P}(n)$ is not greater than the one of $\mathcal{P}(n-1)$.*

$$\forall\, n < m, \qquad C_T^{*[n+1]} \leq C_T^{*[n]}$$
$$\forall\, n \geq m, \qquad C_T^{*[n]} = \bar{C}_T$$

Basically, the first inequality comes from the idea that adding a new platform in an optimal solution of $\mathcal{P}(n-1)$ exactly on an existing one gives a possible solution with n platforms of cost $C_T^{[n]} = C_T^{*[n-1]}$. By definition of the optimal cost with n platforms we deduce the result. The equality in cases in which we have more platforms than customers is an obvious statement: in this case, $(n-m)$ platforms are useless after having located either m platforms on customers (in this case there are only only direct flows).

Proposition 26 *If $C_T > 0$ and $\forall k,\ C_k = \infty$, then:*

$$\forall\, n < \min{(m, p)}, \qquad C_T^{*[n+1]} \leq C_T^{*[n]}$$
$$\forall\, n \geq \min{(m, p)}, \qquad C_T^{*[n]} = \bar{C}_T$$

From the proposition (26) we can reduce the domain of the optimal number of platforms. Obviously, it is at least one and at most the number of customers or plants. The same way, we obtain inequalities on global costs.

Proposition 27 *If $\forall n > m\ C_F(n) > 0$ and $C_T > 0$, then :*

$$\min_{n \in \mathbb{N}^*} C^{*[n]} = \min_{n \in \{1,\ldots,m\}} C^{*[n]} \tag{3.7}$$

If $\forall n > m\ C_F(n) > 0$, $C_T > 0$ and $\forall k,\ C_k = \infty$ then :

$$\min_{n \in \mathbb{N}^*} C^{*[n]} = \min_{n \in \{1,\ldots,\min{(m,p)}\}} C^{*[n]} \tag{3.8}$$

Proposition 28

$$\forall n > 1, \quad C^{*[n]} - \big(n \times C_F(n) - (n-1) \times C_F(n-1)\big) \leq C^{*[n-1]} \tag{3.9}$$

A multi-product supply chain with specific uncapacitated plants

We consider a set of different products whose origins are all different: each product comes from a specific given plant. Given a set of customers whose requirements and locations are known, we would like to determine what is the best way to fulfill them through a set of platforms. We assume that each customer must be served by one platform which is replenished by plants. We know both the platform opening costs and the transportation costs. We try to answer the same question as before: **what is the cheapest set (number and locations) of platforms that fulfills the customers' demand?**

For instance, Figure (3.6) present a supply chain made of 80 customers (black points) and 5 products coming from specific plants (green points). To serve them, we propose a set of three platforms (red points) whose relationships with customers and plants are described.

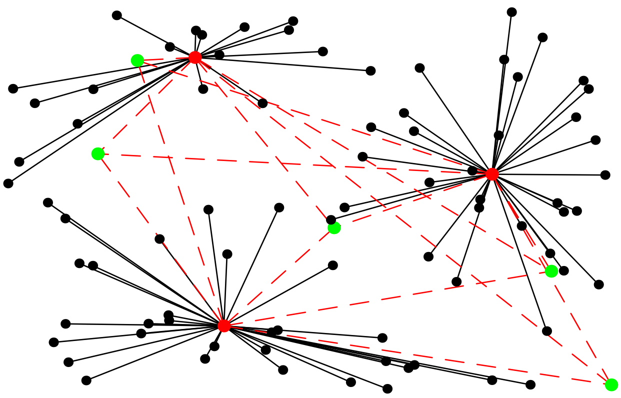


Figure 3.6: Relationships within customers, platforms and plants

We introduce some notation to model our problem: we consider a set of different products $P = \{k \in [1, p]\}$. Each product k is made by an unique plant U_k. We still denote $\mathcal{U} = \{U_k,\ k \in \{1, \ldots, p\}\}$ the set of plants whose locations are given ; in this scenario, plant do not have a limiting capacity. For each platform i and customer j, we denote f_{ijk} the flow of product k that is sent from the platform P_i to the customer C_j. Of course, we find: $\forall k, i \quad \sum_i f_{ijk} = \alpha_{jk}$. We assume that each customer must be served by one and only one platform. Using the notation introduced before, we can easily write down the formula of transportation cost:

$$C_T^{[n]}(\mathcal{P}) = C_T \times \sum_k \sum_j \Big(\alpha_{jk} \times \big(d(U_k, \bar{P}(j)) + d(\bar{P}(j), C_j)\big)\Big)$$

We define also the unrealistic[5] constant transportation cost $\bar{C}_T^0$ corresponding to the case without any product mixing constraints: each plant sends directly its product to each customer so that we do not need any platform, and we find:

$$\bar{C}_T^0 = C_T \times \sum_k \sum_j \big(\alpha_{jk} \times d(U_k, C_j)\big)$$

The same way, we define the more realistic case in which by definition we need to group all products before sending them to each customer but we do not have any platform: each plant orders products from other ones in order to serve its customer zone. This corresponds virtually to the case in which we have p platforms which are located in every plant. We denote $\mathcal{C}_{\mathcal{U}}^{vo}(U_k)$ the set of customers that are closer to the plant U_k than other plants: they are thus served by this plant.

$$\bar{C}_T^p = C_T \times \Big(\sum_{k_1} \sum_{C_j \in \mathcal{C}_{\mathcal{U}}^{vo}(U_{k_1})} \big(\sum_{k_2 \neq k_1} \alpha_{jk_2} \times d(U_{k_2}, U_{k_1}) + \sum_{jk_1} \alpha_{jk_1} \times d(U_{k_1}, C_j)\big)\Big)$$

Of course, $C_T^{*[p]} \leq \bar{C}_T^p$: mixing platforms in plants may not be the best solution in terms of transportation costs.

Proposition 29 *If $C_T > 0$, then the optimal transportation cost of the subproblem*

[5] due to transportation constraints, such as minimal lot size, etc.

$\mathcal{P}(n)$ *is not greater than the one of* $\mathcal{P}(n-1)$.

$$\forall\, n < m, \qquad C_T^{*[n+1]} \leq C_T^{*[n]}$$
$$\forall\, n \geq m, \qquad C_T^{*[n]} = \bar{C}_T^{p}$$

Basically, the first inequality comes from the idea that adding a new platform in an optimal solution of $\mathcal{P}(n-1)$ exactly on an existing one gives a possible solution with n platforms of cost $C_T^{[n]} = C_T^{*[n-1]}$. By definition of the optimal cost with n platforms we deduce the result. The equality in cases in which we have more platforms than customers is an obvious statement: for $n \geq m$, we have more platforms than customers. In this case, $(n-m)$ platforms are useless after having located m platforms on customers.

From the proposition (29) we can reduce the domain of the optimal number of platforms. Obviously, it is at least one and at most the number of customers or plants. The same way, we obtain inequalities on global costs.

Proposition 30 *If* $\forall n > m\ C_F(n) > 0$, *then :*

$$\min_{n \in \mathbb{N}^*} C^{*[n]} = \min_{n \in \{1,\dots,m\}} C^{*[n]} \tag{3.10}$$

Proposition 31

$$\forall n > 1, \quad C^{*[n]} - \big(n \times C_F(n) - (n-1) \times C_F(n-1)\big) \leq C^{*[n-1]} \tag{3.11}$$

3.2.4 An Original Heuristic

We have built a heuristic method based on different works we found in the literature. We compare it to a basic clustering method and to one based on the well-known location-allocation algorithm of Cooper [Coo64].

We define the fusion process of two platforms (also called centers). A platform produced by the fusion of two former ones is thus the Weber point of the union of their former point sets.

Definition 32 $\forall n \in [2, m]$, *for any set of platforms* $\mathcal{P} = \{P_i,\ i \in \{1, \ldots, n\}\}$, *we define the fusion transformation of two given platforms* P_{i_1} *and* P_{i_2} *by:*

$$F : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}^2$$

$$F(P_{i_1}, P_{i_2}) = \mathcal{W}_B\big(\mathcal{C}_{\mathcal{P}}^{vo}(P_{i_1}) \cup \mathcal{C}_{\mathcal{P}}^{vo}(P_{i_2})\big)$$

Thus the set of $(n-1)$ platforms post fusion is

$\mathcal{P}' = \{P_1, \ldots, P_{i_1-1}, P_{i_1+1}, \ldots, P_{i_2-1}, P_{i_2+1}, \ldots, P_n, F(P_{i_1}, P_{i_2})\}$ and we may define the cost of the fusion by: $C_F^{[n+1\to n]}(P_{i_1}, P_{i_2}) = C_T^{[n-1]}(\mathcal{P}') - C_T^{[n]}(\mathcal{P})$

Definition of a greedy clustering method

We define a basic clustering method by the following function `BasicClustering` (written in `C++` code). The class `_CLUSTER` is a set of weighted points (of class `POINT`) to which we associate a center which computed as the Weber Point. The class `PARTITION_CLUSTERS` represents a partition of the set of points into clusters. The algorithm is initialized by creating a `_CLUSTER` for each point. During each iteration, we reduce the size of the partition by merging clusters whose distance between centers is less than a given distance. Two parameters characterize this *greedy algorithm*: α is the initial distance used for building point associations during each iteration, while $\lambda > 1$ is the proportion to which we rise α from one iteration to another. We stop this clustering procedure by the maximal cardinal N (written `NbreMaxPF` in `C++`) that is specified by the user.

We apply the following procedure to the set of customers $\mathcal{C}$:

1. Set $k = 0$, $\alpha^{[k]} = \alpha$ and $\Phi^{[k]} = \mathcal{C}$.
2. Step k:
   (a) If $|\Phi^{[k]}| < N$ Then GO TO (3). Else we apply the function `BasicClustering`$(\alpha^{[k]}, N)$ which does basically::
   (b) While it exists two centers p_1 and p_2 such that their distance to each other is not greater than $\alpha^{[k]}$, then:

- We $\Phi^{[k]} = \Phi^{[k]} \setminus (\phi(p_1) \cap \phi(p_2))$, $\phi(p)$ being the cluster of points whose center is p.
- We compute ϕ as the cluster whose platform is centered on the weber point of the union of weighted points of p_1 and p_2 sets.
- $\Phi^{[k]} = \Phi^{[k]} \cap (\phi)$. GO TO (3b)

3. Set $\Phi^{[k+1]} \leftarrow \Phi^{[k]}$, $\alpha^{[k+1]} \leftarrow (\alpha^{[k]} \times \lambda)$ and $k \leftarrow k+1$. GO TO (2).

4. STOP. $\Phi^{[k]}$ is the partition of the set of customers $\mathcal{C}$.

The C++ implementation of the function BasicClustering($\alpha^{[k]}, N$) we used is proposed in Appendix (§A.5.1).

Heuristic based on the location-allocation algorithm

The idea of the location-allocation algorithm comes from mathematical properties of the solution we exposed in section (§ 3.2.1).

Basically, starting from an initial solution in which n centers are specified for covering m weighted points, it moves centers to weber points of their clusters before computing new corresponding Voronoï cells until centers being weber points of Voroinoi cells. The quality of the solution obviously depends highly from the initial situation, and it exists plenty of local optima. Thus, we build an algorithm following the simple idea that the best solution with n centers may be close the one with $n+1$ centers. Starting from the initial solution with $n = m$ centers (one center by point), we jump from the step with $n+1$ centers to the step with n ones by determining the cheapest (in terms of cost) fusion of centers.

We create a transformation to transform a solution of the problem $\mathcal{P}(n+1)$ into a solution of $\mathcal{P}(n)$. We denote it $\mathcal{T}^{[n+1\to n]}$:

Definition 33 $\forall n \in \mathbb{N}^*$, *we define the transformation* $\mathcal{T}^{[n+1\to n]}$ *as follows:*

$$\mathcal{T}^{[n+1\to n]} : (\mathbb{R}^2)^{n+1} \to (\mathbb{R}^2)^n$$

1. *based on a set of* $(n+1)$ *platforms* $\mathcal{P} = \{P_1, \dots, P_{n+1}\}$ *whose cost is* C, *we compute the* $\frac{n(n+1)}{2}$ *possible fusions* $F(P_{i_1}, P_{i_2})$ *within two platforms* $P_{i_1} \neq P_{i_2}$.

2. *we determine the cheapest fusion cost* $C_F^{*[n+1\to n]} = \min_{i_1 \neq i_2} C_F^{[n+1\to n]}(P_{i_1}, P_{i_2})$

3. *we choose arbitrary one fusion whose cost is* $C_F^{*[n+1\to n]}$ *and we apply it to* $\mathcal{P}$

4. *we obtain the set of* n *platforms* $\mathcal{P}' = \mathcal{T}^{[n+1\to n]}(\mathcal{P})$ *whose transportation cost is:* $C' = C + C_F^{*[n+1\to n]}$

Remark 4 *To reduce the computation time of* $C_F^{*[n+1\to n]}$*, we have noticed two excellent approximations:*

- *For cases in which the number of customers per platform of the closest couple is really low (strictly less than 3 or 4), an excellent approximation of* $C_F^{*[n+1\to n]}$ *may be obtained by fusion of the two platforms closest to each other.*
- *For any cases,* $C_F^{*[n+1\to n]} \approxeq \min_{i_1 \neq i_2 \in N(i_1)} C_F^{[n+1\to n]}(P_{i_1}, P_{i_2})$*, with* $N(i_1)$ *denoting the neighborhood of platform* i_1*. We define this neighborhood as the set of* i_2 *having a common Voronoi point with* i_1 *in the Voronoi diagram.*

Let us plot an example of a fusion on a simple problem: we use a set of ten identical[6] customers and we start from a solution with three platforms (i.e. a solution of $\mathcal{P}(3)$).

Figure (3.7) presents the solution before the fusion. Arbitrarily, we decide to merge the second and the third platforms. Figure (3.8) plots the new supply chain. Transportation cost of the solution before fusion was 24387 points, whereas after fusion the cost of the solution is 35896 points.

Finally, if we apply the transformation $\mathcal{T}^{[3\to 2]}$ to the initial system plotted on Figure (3.7), we find that the cheapest fusion is to merge platforms one and three: $C_F^{*[3\to 2]} = -518$. The result is plotted on Figure (3.9). Naturally, the transportation cost of the solution post transformation is: $24387 - 518 = 23869$.

[6]we assume each one consumes one unit of product

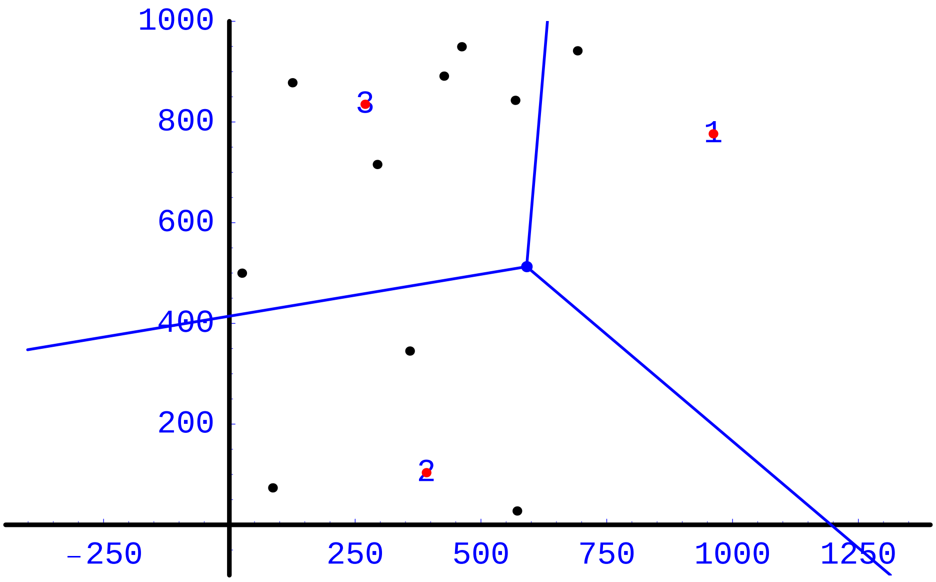


Figure 3.7: Supply chain with three platforms

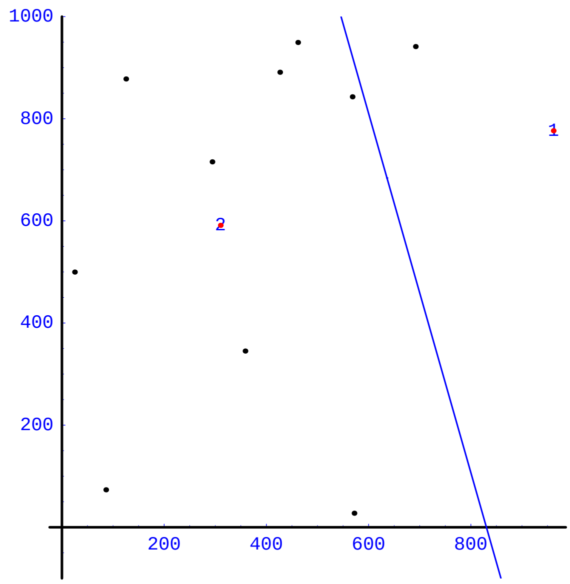


Figure 3.8: Example after the fusion of two platforms

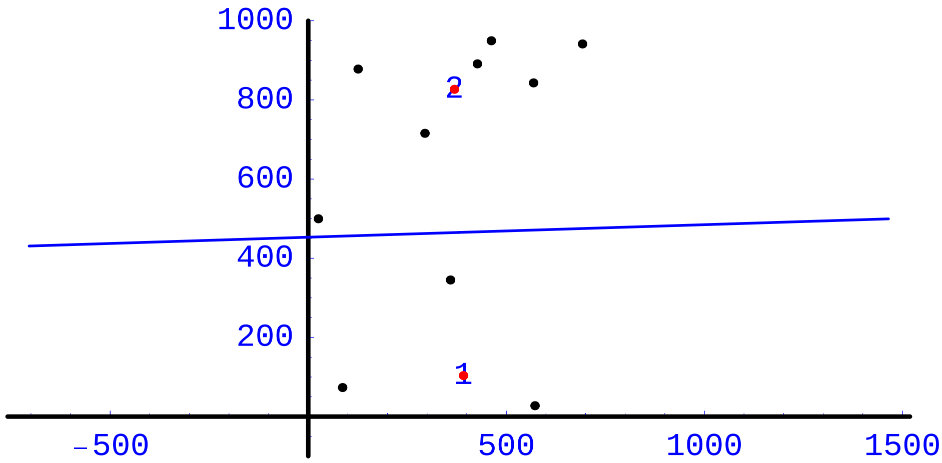


Figure 3.9: Example after the transformation $\mathcal{T}^{[3\to 2]}$

We denote as follows the *location-allocation algorithm* (see [Coo64]).

Definition 34 $\forall n \in \mathbb{N}^*$, *we denote the transformation* $\mathcal{G}^{[n]}$ *as follows:*

$$\mathcal{G}^{[n]} : (\mathbb{R}^2)^n \to (\mathbb{R}^2)^n$$

is the following algorithm:

1. *Given* $\mathcal{P}^0 = \{P_1^0, \ldots, P_n^0\}$, *set* $k = 0$.
2. $\forall i \in [1, n]$, *we compute the corresponding sets of customers* $\mathcal{C}^{vo}_{\mathcal{P}^k}(P_i^k)$.
3. *For each set of customers, we compute the Weber Point:* $W_i^k = \mathcal{W}_B(\mathcal{C}^{vo}_{\mathcal{P}^k}(P_i^k))$. *If* $\forall i$, $W_i^k = P_i^k$, *STOP*
4. *Set* $\forall i$, $P_i^{k+1} \leftarrow W_i^k$, $k \leftarrow k+1$. *GO TO (2).*

To illustrate the principle of this transformation which tries to find a stable solution in which each platform is the Weber point of its Voronoi diagram, we use an example based on an initial solution with $m = 50$ identical customers and $n = 10$ platforms (presented on Figure (3.10)). In this case, the algorithm converges after two iterations. Figures (3.11) and (3.12) present intermediate and final steps during the transformation $\mathcal{G}^{[10]}$.

At first sight, the reader could not notice the slight differences within figures. However, an accurate analysis shows clearly that platforms are moved.

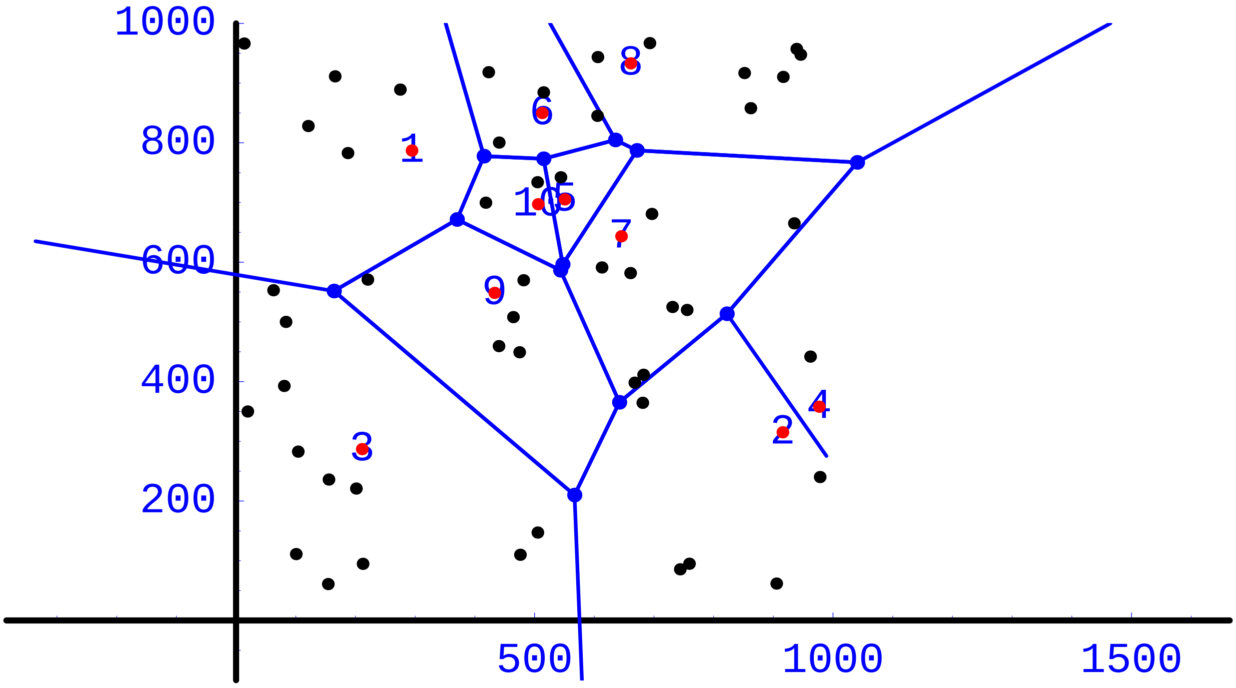


Figure 3.10: Initial supply chain: transportation cost 81903

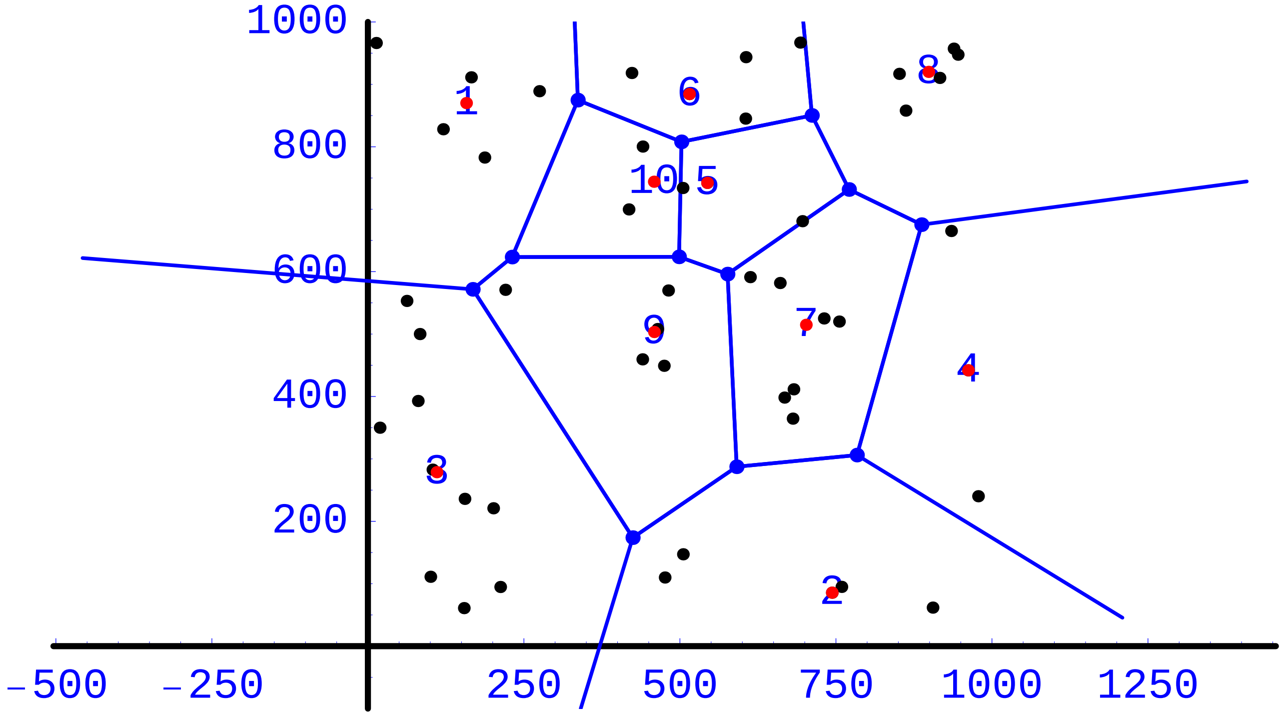


Figure 3.11: Supply chain after the first step: transportation cost 52800

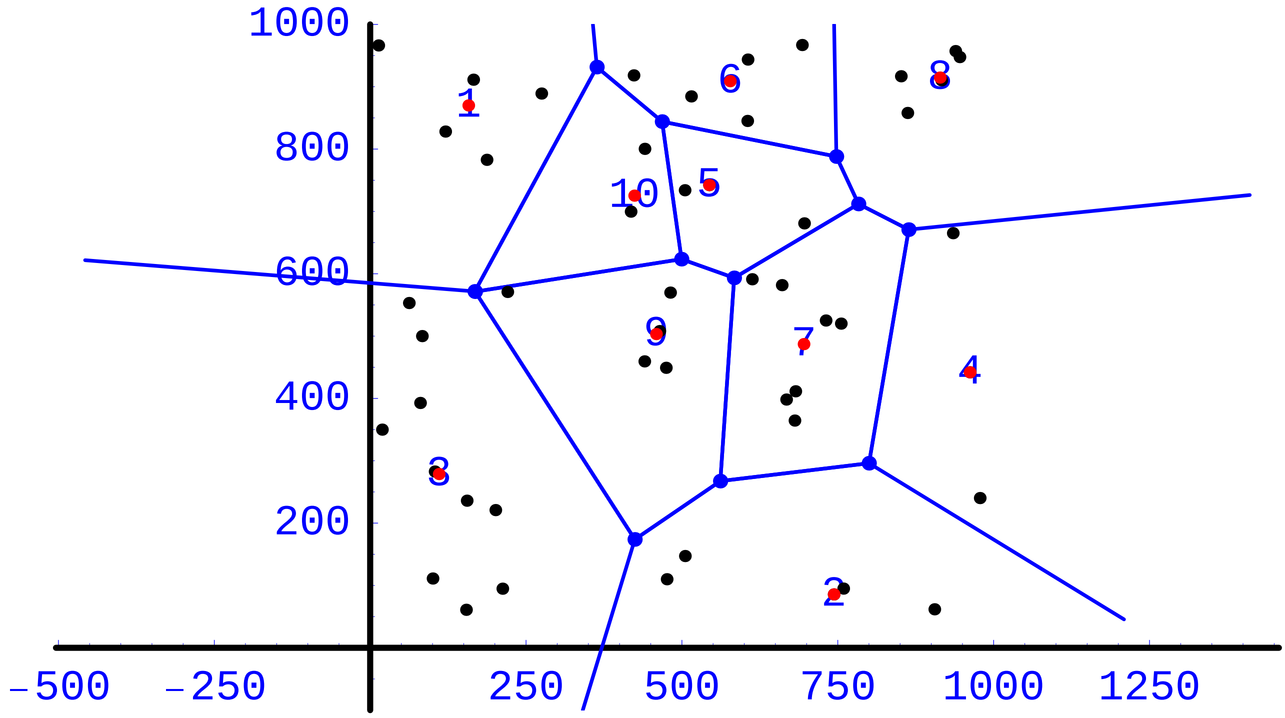


Figure 3.12: Supply chain after the transformation $\mathcal{G}^{[10]}$ in two steps: transportation cost 52074

From these two first heuristic method, we may build a mix by running the location-allocation algorithm between each iteration of the greedy algorithm. We will call this procedure the *greedy location-allocation algorithm*.

Our Heuristic

Based on previous results, we decided to use a *simulated annealing procedure* based on two types of elementary transformations to perform a local optimization given a solution of cost C. For cases in which clusters contain only few points, we define a perturbation (based on the function `ElementaryMovementSmallClusters` defined in Appendix (§A.5.2)) which modifies the partition by moving randomly some points to neighbor clusters. Otherwise, we define a perturbation that moves randomly some cluster centers (based on the function `ElementaryMovementBigClusters` defined defined in Appendix (§A.5.2)). Each perturbation may change the partition cardinal, unless we forbid such a movement.

Basically, we define a simulated annealing procedure with classical parameters: the initial temperature is $T_0 = \frac{C \times n}{1000}$, the freeze temperature $T_f = \frac{C \times n}{100000}$, the coefficient of temperature decrease $\rho = 0.95$ and the number of iterations by step $N = 100$.

Result of the comparison on our real-life data set

Given a set of real customers in the Saint-Gobain Glass case, we provide in Figures (3.13) and (3.14) the comparison between the four heuristics described so far. Computational effort is reasonable (at most a few seconds) for all the methods, according to our C++ program used on our lab-top.

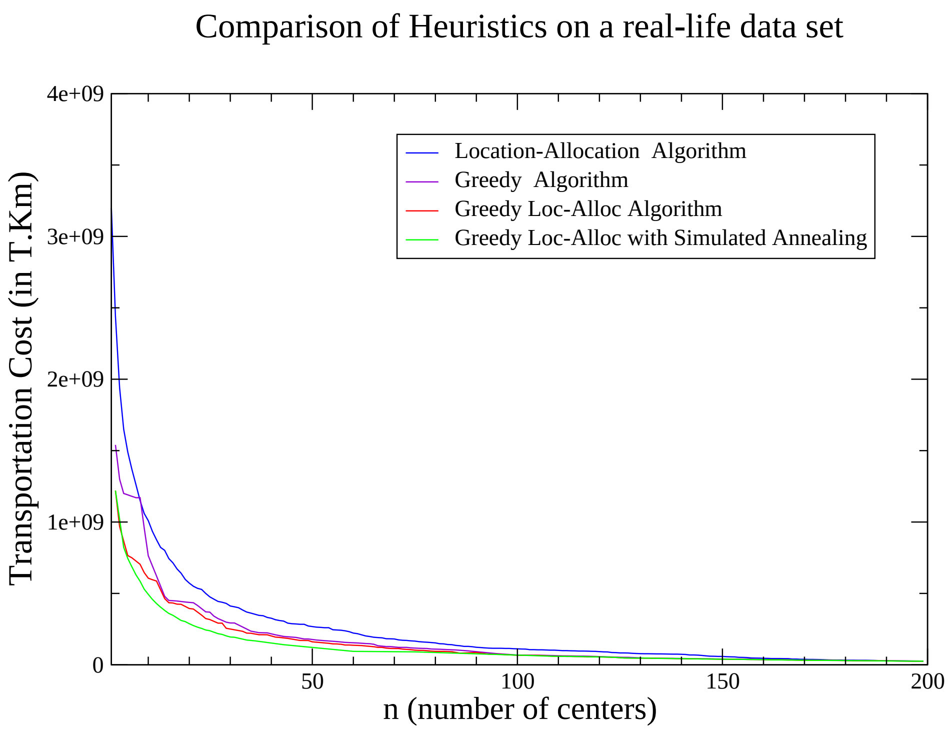


Figure 3.13: Comparison of four Heuristics of clustering on a real-life data set (200 customers)

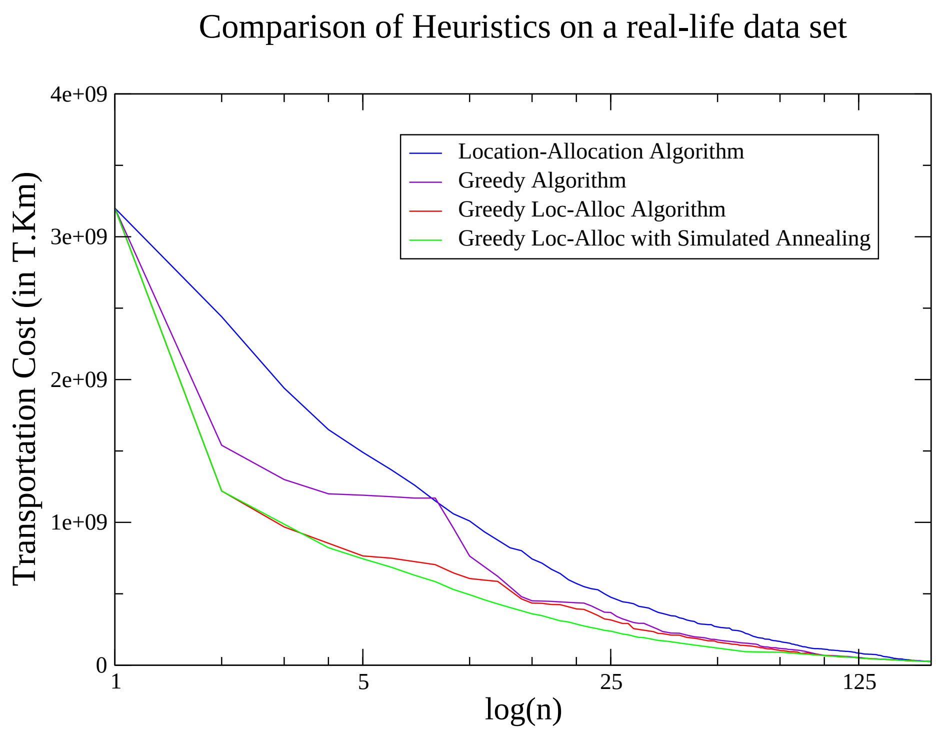


Figure 3.14: Comparison on a logarithmic scale

It appears that our heuristic method gives better results than others without intensive computational effort. We use it in chapter 6 in a preliminary step of customer aggregation into customer families.

So far, we have discovered that solving a simple model exactly may be theoretically very difficult, whereas underlying assumptions are highly simplified. However, these models highlight different trade-off managers face in the supply chain design. We are now able to lead a practical discussion about possible logistic organizations in the glass industry case.

3.2.5 Practical issues

In chapter 1, we have described the industrial context of Saint-Gobain Glass. Section (§ 1.4) presented more specifically distribution issues of flat glass, by emphasizing the difficulty to fulfil mixed orders (made of at least two different products) in the present organization. We propose in Appendix (A) a detailed analysis of the Saint-Gobain Glass case which develops ideas and prospects trying to fill in the gap between the theory exposed so far and real-life issues.

We analyze in section (§A.1) industrial past data to understand the underlying structure of demand in order to develop insights on non-optimal logistic phenomena. We study on past data the exact flow of products accross the supply chain. We follow a pragmatical method aiming at both discovering new concept of distribution and determining efficient distribution rules for a given design.

Our simulations (§ A.2) on past data allow us to capture the high potential of such an effort: few percents of present distribution costs may be saved through an organization improvement.

For each mixed order, simple computations point out the potential of an optimal choice of both producing and shipping plants. To achieve it in practice, we aim in section (§ A.3) at determining some practical rules that would be nearly optimal.

Finally, we expose in (§ A.4) possible evolutions of the supply chain of Saint-Gobain Glass that would potentially solve main problems managers face nowadays.

3.3 Conclusion

In this chapter and its corresponding appendix (A) we develop both theoretical and pragmatical methods to study how to determine an optimal supply chain design through facility location problems.

In this chapter, we deal with a simple strategical distribution issue: given a supply chain with defined upstream and downstream flows, how do determine both the optimal number and positions of logistic platforms?

Before exploring the literature review (§ 3.2.2), we focus on a simple and unrealistic model (§ 3.2.1): given a set of customers (defined by a position and a deterministic demand) and some transportation and platform opening costs, what is the optimal way to serve each customer by one platform? Understanding this location-allocation problem will allow the reader to discover the highly combinatorial structure of this apparently simple question.

Based on existing results, we propose an original integration of upstream flows (we introduce two different scenarios) in section (§ 3.2.3). Firstly, we propose a mono-product model produced by several known capacitated plants. Secondly, we take into account customers who ask for mixed orders made of products whose origin plant is unique and known (uncapacitated and located). We provide a heuristic algorithm (§ 3.2.4) that tackles the problem. Results of this chapter are used for customer aggregation in chapter (6).

We propose in Appendix A a detailed analysis of the Saint-Gobain Glass case which develops ideas and prospects trying to fill in the gap between the theory and real-life issues. We focus on past data in the logistic network because we aim at highlighting distribution issues in the glass industry. Finally, it appears that the simple models that this chapter present are quickly limited and inadequate to help managers to make a strategic decision.

We will discover in chapter 6 how we have developed a more general and complex model based on linear programming theory. In a nutshell, this chapter has been a necessary step in our research to motivate further work.

Chapter 4

The PLANEO project: a generic model for production planning

4.1 Introduction

We address the problem of developing a decision tool for both the production planning and the logistic decisions in the glass manufacturing industry.

First of all, we deal with the particular structure of Glass plants. Starting from the industrial float process, we propose a framework to structure the planning process in a hierarchical way by ordering decisions according to their relative importance. We base our model on the multi attribute product structure that can be highlighted in this particular business. At each level of hierarchy, we provide a mixed integer model to capture all the costs and constraints of both production and inventory systems. We use discrete time periods and both set-up costs and times. We propose an extension of a classical model found in the literature and we adapt it to the particular structure of our data. We provide several practical approximations overcoming the huge size of industrial applications. Then, we use a commercial solver to solve it efficiently.

Using the model of this chapter as a building block, we will integrate this work in chapter 6 in a multi job, multi machine and multi location model: Float glass is indeed mainly transformed through different processes to provide commodity products, such as laminated glass or coated glass. We will see in chapter 5 that all these possible steps can be captured by the production model of this chapter.

4.1.1 Production Planning in process industries

Our research has been originally motivated by the need for a practical decision support tool for one major Glass manufacturing industrial company. However, our contribution appears to be suited to tackle similar issues in any continuous process industry, such as steel and paper industries. This is why we present this work within the general framework of continuous process industries.

The main characteristics of continuous process industries lie in capacitated expensive industrial resources whose process is continuous and on which economies of scale are possible. This industrial reality often leads to long operating life and inflexible production lines as well as important set-up times and costs. Therefore, smoothness of production decisions may be mandatory to fulfill some process constraints and large lot sizes may be produced: these so called campaigns produce appropriate quantity, while avoiding part or all of the set-up costs incurred when switching between different products. In addition, it is common in process industries that "quality" refers to conformance to product specifications: quality is affected by the duration of the campaign: with increasing campaign length, variance in conformance and its associated rework or loss costs[a] are reduced. However, the inventory of other products must be sufficient to cover downstream demand, representing huge inventory costs.

To deal with this required make-to-stock policy, industrial companies have developed their own forecasting methods (see [CM01a]), used as deterministic data.

[a]In [Raj04], authors develop for multi-product batch operations an interesting quality model that calculates explicitly quality costs

Thus, production planning consists in solving the following constrained optimization problem: find the best way to fulfill demand forecasts while satisfying production constraints and minimizing relevant variable costs.

This is basically a Lot-Sizing and Scheduling problem. For cases in which downstream operations are typically run at stable utilization levels, a static setting of production lines may be sufficient. Models such as the campaign planning and scheduling problem (CLSP, see [Raj04]) are then reasonable. Otherwise, the general lot-sizing and scheduling problem (GLSP, see [FM97]) appears to be relevant for many industrial problems. We deal with this later model in our research.

Of course, we may address this general issue at different hierarchical levels. [HM75] introduced the notion of hierarchical production planning and provides a specific framework for this, whereby each level of hierarchy has its optimization model and the feasibility from a level to the next is ensured by an imposed additional constraint. [BT93] provides a survey of hierarchical planning methods. Thinking about the supply chain matrix (denoted SCP-matrix by the authors in [?]), the idea could arise to tackle all planning tasks with one comprehensive, overall planning model simultaneously. Clearly such an approach will never work for reasons of computational complexity. Furthermore, independently of the power of solution procedures and Operations Research methods, such an approach would not be useful, anyway, for the following reasons (see [Mea84] and [FM03]):

- the longer the planning horizon, the higher the uncertainty. What-if-analysis and risk scenarios (see [CM01b]) only play a dominant role in strategic planning.

- Different planning horizons imply different frequencies of planning: rolling horizon planning is very popular. Here the planning horizon is split into time buckets, but only the first one is put into practice. After this *frozen horizon* is elapsed, a re-planning is done considering new and probably more reliable information.

- Planning tasks on different planning levels need a different degree of aggregation (in terms of time, products, place, and resources).
- The higher the decision in the SCP-matrix, the longer its impact on the business. Of course, the higher the importance of the decision, the higher the rank on the decision maker in the company's hierarchy. In a nutshell, decisions are of various importance and are made by various actors.

Hierarchical planning seeks to coordinate planning modules such that the right degree if integration can be achieved. Information and guidelines between planning modules are exchanged in all possible directions. Dauzère-Pèrés and Lasserre discuss in [DPL02] the traditional hierarchical approach to production planning and scheduling, emphasizing the fact that scheduling constraints are often either ignored or considered in a very crude way. They review usual methods to handle capacity in theory or in practice, before presenting an approach that overcomes these drawbacks by capturing the shop-floor capacity through scheduling considerations.

In the following, we develop a lot-sizing and scheduling model that may be used at each level of a determined hierarchical production planning process. In this context, Kuik, Salomon and Van Wassenhove discuss in [KSW94] the impact of lot sizing and production planning at different decision levels in the organization and respond to some criticisms on lot sizing.

4.1.2 Literature review on lot-sizing models

The scientific literature provides plenty of models related to our industrial context. Excellent general references on production planning are [TM93], [Sha93] and [SPP98] as well as more recently [SK02] and the remarkable bibliography of the first chapter of Raf Jans's PhD ([Jan02]). In spite of the various number of models, they are easy to classify (see [Car03]).

An important class of production planning models analyze scheduling problems. They essentially determine start and finish times of jobs (scheduling) and the order in which jobs are processed (sequencing). Lawler et al. ([LLKS93]) give an extensive overview of models and algorithms for these problems. Lot sizing models determine the optimal timing and level of production. Various assumptions create many families of problems.

For instance, on the one hand, there are the continuous time scale, constant demand and infinite time horizon lot sizing models. The famous single-item Economic Order Quantity model (EOQ) and its extension to multiple items and constant production rate, known as the Economic Lot sizing and Scheduling Problem (ELSP, see [Elm78] and [Zip91] for excellent reviews) , have been successfully implemented in several businesses. The ELSP has been shown to be NP-Hard ([Hsu83]). Consequently, an effective method for computing the optimal solution to the general problem does not exist.

On the other hand, we have the discrete time scale, dynamic demand and finite time horizon lot sizing models. In our context we focus on these later optimization models. The ELSP with discrete time periods, variable demand, and sequence-dependent setups is known as the “product cycling” problem (see [KS85]). Furthermore, the ELSP with integer batch sizes addresses the scheduling and planning of batch operations. Reklaitis ([Rek92]) provides a comprehensive review about its application in the chemical processing industries. Recent research ([Raj04]) have been applied to the food-processing industry with huge savings.

Lot sizing models assume that demand is deterministic, whereas stochastic inventory theory analyzes models that assume a stochastic demand. An overview of stochastic inventory models can be found in Porteus ([Por90]). Lee and Nahmias ([LN93]) give a general introduction into models for production planning and inventory management. In our research we focus on deterministic models.

The trivial problem for the lot-sizing cases can be formulated for a single stage with infinite production capacity and a single product to be planned over a fixed number of time periods. The objective function corresponds to minimizing the total holding cost of the inventory resulting from the quantities produced. The main constraint to be handled is the inventory balance equality. Of course, the trivial solution is to produce exactly the amount of demand at each period and therefore there is no inventory. This model is a Linear Programming Model.

This model can be basically sophisticated if we add some capacity constraints, as well as some backorder costs, etc. However, as soon as we introduce set-up costs, the model is no longer a Linear Program, but becomes a Mixed Integer Program (MIP). As a first step, Wagner and Whitin consider in [WW58] the single-product single-machine problem, in which a set-up cost is charged in each period if the product is produced in that period. They consider the single item uncapacitated lot sizing model for the case where production costs are constant over time, and in 1960 for production costs which are not constant, and prove that there exists an optimal solution that satisfies the following property: one never produces in a period and at the same time has inventory coming in from the previous period. This is called the Wagner-Whitin property. This property also implies that one produces to satisfy the demand for an integral number of consecutive periods. Based on these special properties of the optimal solution, Wagner and Whitin formulate a dynamic programming (DP) recursion for solving this problem.

Other DP have been developed for special cases (capacitated lot sizing, backlogging, etc.). The drawback of these DP algorithms is that they are developed for single item problems and cannot be directly used for multi-item problems. However, these single item problems are very important as they appear as core structures in more complex problems such as the multi-item capacitated case. Decomposition methods are used to find tighter bounds for the more complex problems. For a good overview of the history of the single item lot sizing problem see Wolsey ([Wol95]).

Formulations of multi-item Lot-Sizing and Scheduling models may involve not only capacitated machines, backorder costs but also set-up costs and set-up times that can be fixed, vary according to the type of product or be sequence-dependent. Kuik, Salomon and Van Wassenhove ([KSW94]) provide a classification of the existing literature based on two parameters: stationary versus dynamic demand and infinite or finite capacity. Further reviews may be found in [DK97] and [BW00] as well as [SC01] and [Jan02]. It also appears that the time period modeling choice is a crucial choice: we particularly deal with it in our research.

First of all, the Capacitated Lot-sizing and Scheduling Problem (CLSP) [DK97], for a single-machine multi-product system aims at minimizing the sum of set-up and holding costs. In the Continuous Set-up Lot-sizing Problem (CSLP) we allow the system to produce under full capacity. The Proportional Lot-sizing and Scheduling Problem (PLSP, [DH95]) occurs when the CSLP model does not use the full capacity of a period, following the basic idea to use the remaining capacity for scheduling a second item.

The CLSP is called a large bucket problem [BW00] because several items may be produced at each given period. The case in which time periods become macro-periods which are subdivided in several micro-periods leads to the Discrete Lot-sizing and Scheduling Problem (DLSP) (see [Fle94]), called a small bucket problem because at most one item can be produced at each given period. The DLSP has the same objective function as the CLSP, but a new constraint corresponding to the all-or-nothing assumptions force the production at full capacity of at most one item at each period. A major advantage of the small-time bucket models against the CLSP is the exact control of the sequence of lots and, hence, the possibility to include sequence dependent setup costs. On the other hand, all solution methods developed for the CLSP are restricted to sequence independent setup costs, because a preferred procedure consists in decomposing the problem into single-products, either by Lagrangian relaxation ([Fle90]) or by column generation ([Sal91]). This product decomposition, however, is obstructed by the additional interdependence of the products caused by sequence dependent setup costs. In [Fle94], Fleischmann presents a new solution procedure based on the equivalence of the DLSP and a Traveling Salesman Problem with Time Windows (TSPTW), for which he describes a Lagrangian relaxation into a shortest path problem with time window (SPPTW) and determine lower bound for the DLSP.

Following this idea, set-up times are taken into account on a model based on the DLSP by Salomon and al. ([SSW+97]). They denote this NP-Hard problem the Discrete Lot-sizing and scheduling Problem with sequence dependent set-up costs and times (DLSPSD). Based on the dynamic programming approach developed by Dumas and al. ([DDGS95]), they solve moderate size problems to optimality with a reasonable computational effort. The set up time is an integral number of time periods. Jordan and Drexl ([JD98]) showed the equivalence between DLSP for a single machine and the batch sequencing problem ([Cla98]).

Willing to generalize the DLSP, Fleischmann and Meyr present in [FM97] the General Lot-Sizing and Scheduling Problem (GLSP) features multiple products, single-machine sequence-dependent set-up costs, but with neither set-up times nor backlogging. Deterministic, dynamic demand is to be met with the objective of minimizing holding and sequence-dependent setup costs. The GLSP is more general than the DLSP and PLSP because the number of products per (macro-)period is no longer restricted. Authors ([FM97]) prove that GLSP is NP-hard.

This general formulation is finally leading to the General Lot-Sizing and Scheduling Problem with Sequence-dependent Setup times (GLSPST, see [Mey99], [Mey00] and [Mey02]). Meyr introduces interesting heuristic methods based on dual reoptimization combined with local search heuristic for solving a MIP, following the method used by Kuik and al. in [KSWM93].

The work of Belvaux and Wolsey in their companion articles [BW00] and [BW01] oriented our strategy: in spite of the remarkable improvements in the quality of general purpose mixed-integer programming software (see Cplex, [ILOa]), they underline that the effective solution of a variety of lot-sizing problems depends crucially on the development of tight formulations for the special problem features occurring in practice. Our research is the exact illustration of this statement.

Finally, recent examples of successful industrial extensions of the discrete lot sizing and scheduling model convinced us of the interest of this method. For instance, Jans and Degraeve develop in [Jan02] and [JD04] a version of DLSP that capture general set-up times that may be fractional, multiple alternative machines as well as backlogging for a real life production problem they found at Solideal, an international tire manufacturer. They present a column generation based algorithm that gives excellent results on reasonable size data sets.

Of course, lot-sizing and scheduling models may be declined to multi-stage (also called multi-level, [SC01]) problems, as they arise as part of the Material Requirement Planning (MRP) logic ([Bak93]). Basically, the costs involved are fixed costs and holding costs. At each level the problem resembles the single level problem, but with the additional property that the lot sizes at each level, which form the solution, also cause part or all of the demand at the next level down the product structure. The problem is to simultaneously find a set of lot sizes at each level, that combined together, minimize the total fixed and holding costs in the system. We deal with these models in chapter 6, in which we provide a literature review (see §6.2).

4.1.3 Interests of our research

The contribution of our research is threefold. First, we solve a relevant business problem and the result of this research is being used in the Saint-Gobain Glass company. Thus, we demonstrate the usefulness, relevance and impact of OR methods in business practice. Secondly, we introduce a model in which we decompose products into attributes and sub-attributes: **we define an attribute as a product**

characteristic which takes an unique value per time period whereas a sub attribute can take several values. This structure appeared to be perfectly suited to the particular operation of glass production. However, we believe that it may be extended to other process industries. Thirdly, we formulate a lot-sizing problem not only mixing big time buckets for attributes and small ones for sub-attributes but also exploiting factorizations that we discovered from our practical experimentation, creating an original extension to the GLSPST. For practical applications we faced, the best on-hand commercial code (without any particular branch-and-cut strategies) we used (Cplex, [ILOa]) gave reasonable results. However, further research exploiting the particular structure of our model for developing particular decomposition methods seems to be an enjoying outlook.

At first sight, we tried to simplify some real life constraints to be able to use classical models and test them. Because of the large size of problems arising from practical industrial application, the formulation found in the literature of the GLSPST problem did not allow us to solve them. The number of integer variables in the corresponding mixed integer linear program increases dramatically when dealing with a high number of products.

In this chapter, we introduce several improvements in it. On the one hand, we reformulate in a first step the MIP by factorizing changeover characteristics (times and costs) within products. On the other hand, given common particular structure of continuous processes, **the key factor of our proposal is to divide multi characteristic products into attributes and sub-attributes** under simple assumptions. This structure matches a hierarchical framework to model production line skills, introducing relevant variable production costs at each level. Our reasoning is thus suited to any level of the hierarchical production planning system.

Finally, we introduce a general mixed integer linear model based on this product decomposition. We mix big and small time buckets and allow the user to define the precision of the results for each product characteristic. In a nutshell, this model captures a bigger scope than the former ones found in the literature and introduces less integer variables, being thus solvable by commercial codes[b].

To check the interest of our method, we apply it to the glass manufacturing industry. In a first part, we describe precisely the industrial context, to underline the complexity inherent to a real continuous process. We then describe precisely our general method for dealing with such an industrial problem. The third part presents the MIP model itself, and finally we conclude on interesting research outlooks.

4.2 Industrial Context

4.2.1 Application to the float glass manufacturing industry

Float glass manufacturing is not unlike the manufacturing of commodities like steel or plastic. Each of the processes requires raw materials to be weighed, mixed, melted at high temperatures, formed into continuous ribbons, cooled and cut into a size that fits its use. We point out that its particular features and our study could be translated on another process. We provide an accurate analysis of the process because we think that understanding the technical reality is important. It will indeed allow us to discuss about our assumptions further in the chapter. The Figure (4.1) presents an overview of the float glass process.

Float glass offers the quality of plate glass combined with the lower production cost traditionally associated with sheet glass manufacturing. Float glass is virtually distortion and defect free, making it ideal for various premium glazing applications in buildings and homes or for automotive glass along with hundreds of other glass fabrications. Float glass is made from a combination of several ingredients such as sand, soda ash, dolomite, limestone, salt cake, and cullet. Various combinations correspond to various colors. The raw materials are received and stocked in silos. The raw materials are then drawn down from the silos for batch weighing and mixing

[b]we used Cplex $v. 8.0$, product of ILOG™, see [ILOa]

(Figure (4.2)). Cullet, which is crushed glass, is blended with the mixed batch to make from up from 15% to 30% of each batch. The mixture is then delivered to the melting furnace by belt conveyor. The raw materials storage and handling is designed to suit the types of glass which will be produced along with the availability and cost of the raw materials. We can reasonably assume that raw materials are not critical in the process: we consider their availability as perfect, i.e. their on-hand quantity is infinite and their delivery lead time is zero. Therefore, we do not focus on it in the following. As the batch is fed into the furnace melter area it's heated by the natural gas burners to approximately 2900°Faraday (Figures (4.3), (4.4) and (4.5)). From the melter the molten glass flows (Figure (4.6)) successively through the refiner, the waist area and then into the working end where the glass is allowed to cool slowly to the proper temperature for delivery to the tin bath. The melting furnace consists of refractory bricks and special shapes, support and binding steel, insulation, a fossil fuel firing system, temperature sensors and a computerized process control system. The design of the furnace is carefully made to meet the plant's specific gross daily glass production tonnage goals, and its life expectation is around ten years. The molten glass, which by now has dropped to 2000°Faraday, forms a continuous ribbon that floats on the molten tin. The desired width and thickness is obtained through an operator controlled program which sets the speed of the annealing machines and the parameters of top rollers touching the ribbon in the tin bath (Figure (4.7)). The ribbon thickness can range from 1.5 to 20 millimeters. As the continuous ribbon moves through the tin bath its temperature is gradually reduced allowing the glass to become flat and parallel. Each tin bath is specifically designed to respond to heat flow balance, desired ribbon width, glass thickness, glass colour and the gross daily production tonnage. The glass leaves the float area (tin bath) at about 1100°Faraday and enters the annealing zone (Figure (4.8)), which controls the cooling of the glass. The temperature of the glass is reduced according to a precise time/temperature gradient profile to produce glass that meets industry standards. The design of the annealing zone (Figure (4.9)) is adapted to meet the critical cool down requirements of each float plant's gross daily production tonnage and glass colors.

The cooled glass ribbon exits the annealing zone (Figure (4.10)) and is conveyed to the cutting area by a system of rollers and drives. The glass is scored by carbide cutting wheels, parallel (Figure (4.11)) and perpendicular (Figure (4.12)) to the ribbon travel, into sizes that meet the plant's customer requirements. On each future cut sheet, the distribution of defaults must satisfy various tolerance ranges, depending on the sold quality.

The scored glass ribbon is then separated into sheets for packaging by unloading personnel or automatic equipment. Sheets are packaged by homogeneous stacks whose size may vary. Each stack is transferred either to the warehouse for storage or the expedition area for shipment to the customers.

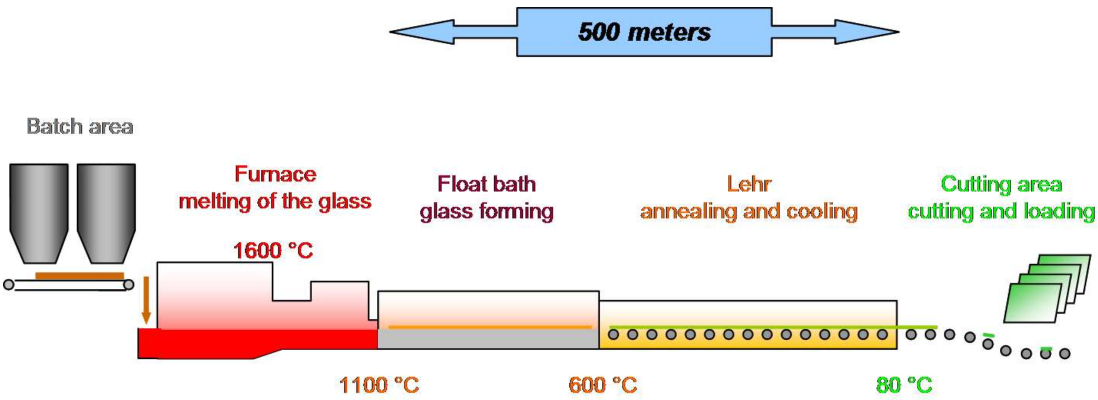


Figure 4.1: Global View of the float glass process

Figure 4.2: Weighing of raw materials

Figure 4.3: Raw materials entering the furnace

Figure 4.4: At the entrance of the furnace

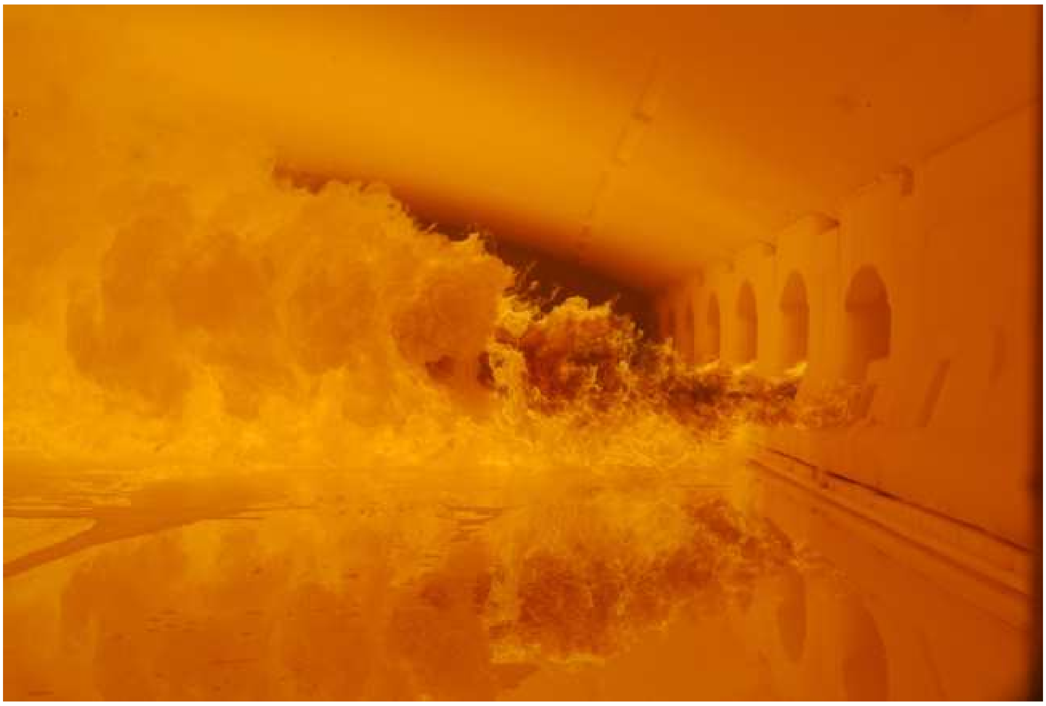


Figure 4.5: Inside the furnace

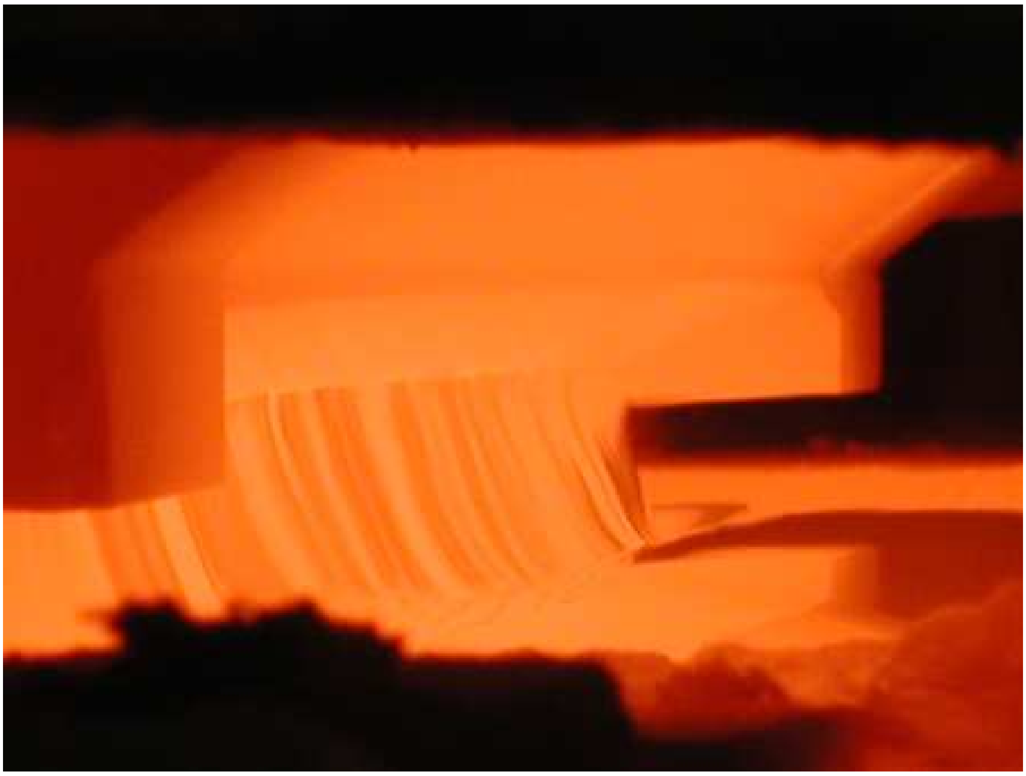


Figure 4.6: Under the spout lip: the glass flows from the furnace to the tin bath

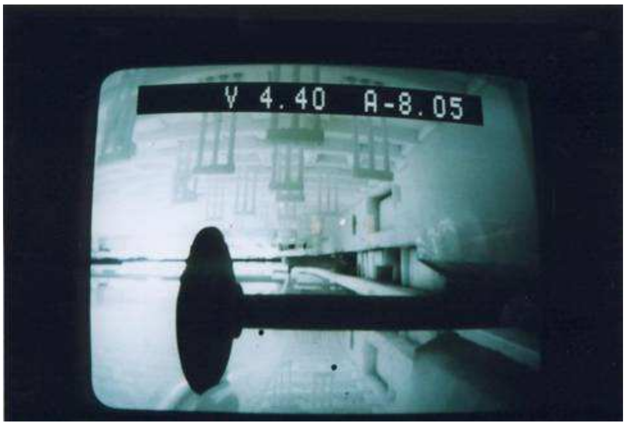


Figure 4.7: Top roller used in the tin bath to work on the glass ribbon

Figure 4.8: At the end of the tin bath the glass ribbon is entering the annealing zone

Figure 4.9: View of the annealing zone

Figure 4.10: The glass ribbon inside the annealing zone

Figure 4.11: The glass ribbon is then cut to its net width

Figure 4.12: The glass ribbon is then cut into glass sheets

4.2.2 Product decomposition into several independent characteristics

From the previous paragraph we can conclude in a nutshell that each finished product made on a float line is characterized by its colour, thickness, width, quality, cut to length and size of the stack. We intuitively range these characteristics according to their "importance" in a given configuration of the line, i.e. depending on their changeover flexibility. For instance, changeovers between various colors take one hundred times as much time as changeovers between widths.

Each characteristic can take various values, depending on the line. For instance, the colour skill set of a given line depends on the built furnace, the possible thicknesses depend on the float bath, etc. Thanks to a teamwork with manufacturers[c], we have concluded that production skills of a given line could be modeled as separable sets of skills for each characteristic. Each plant may thus be defined by its skill set for each product characteristic.

From a practical point of view, producers know pragmatically how much time it takes to switch to a product to another one. If the changeover time between two desired productions is strictly positive, the produced glass ribbon is continuously broken and sent to cullet silos in order to be melted later. At first sight, we may consider that changeover time has both a direct and an indirect cost. Direct costs come for instance from the reinforced manpower that is required, whereas indirect costs lie in both a time loss (we may define an opportunity cost) and in the taken risk, to the extent that producers may lose control of the glass ribbon and then shut down the production line for a week.

Of course, depending on the time horizon and the time scale we are trying to solve, we may distinguish characteristics whose value may be considered constant within a time period and others.

Implicitly, we assume that it makes sense to take into account mean values for

[c] we worked with both the production manager and the planner of the *Chantereine* line in France

changeover times and production capacities. We focus thus on classical industries whose processes are well mastered so that uncertainty be negligible.

The forthcoming Figure (4.13) illustrates the way we decompose product into characteristics, while integrating this decomposition into a hierarchical planning approach.

4.2.3 Relevant costs

So far, the basic trade-off of our problem is to minimize the sum of both variable production costs and inventory costs, to the extent that there is no resource acquisition matter. However, relevant costs included in these two categories depend on the level of the optimization. The more detailed level we work on, the more detailed cost are. Let us describe all possible costs before explaining which ones are relevant at each level.

Firstly variable production costs include the set-up related costs which only depend on the kind of chosen changeovers within products. These costs capture both the opportunity cost of a changeover due to its average duration and the cost of its associated risk. In the Float case, the risk can capture the probability to lose the control of the glass ribbon, the uncertainty of the duration, etc. From our team work with practitioners, we decided not to model the available workforce on the production line at any decision level. Instead of using human resource constraints, we model the manpower flexibility cost (the same way hiring or firing costs are classically taken into account) through two different time-dependent costs.

We remember in the case of a float line that production is continuous during many years. Of course, the workforce is organized by rolling teams, but there is still more workers during regular day hours. We thus found that a changeover cost may depend on the moment when it happens. To describe this time dependent cost, we use the term of additional set-up cost.

Besides, we have found that certain particular products need particular attention to be produced. For instance, we could quote an optional step during the float process. At the exit of the float bath, it is possible to coat some metal layers on the glass ribbon. This is called a hard coating operation. To do so, dedicated additional workers are needed. To catch this phenomenon we use a time dependent additional production cost that we denote a production over-cost.

Secondly inventory cost is the cost of carrying one unit in inventory for a specified period of time. It is a combination of the cost of capital, the cost of physically storing the inventory and the cost that results from the product becoming obsolete (see [CM01c]).

At an operational decision level where demand forecasts for the coming time periods are the most accurate, we may try to model the alternative between pushing production into the warehouse and shipping it directly to the customer. We can thus consider a handling cost, which corresponds to the long and expensive handling operations to put the production in and out of the warehouse.

In addition, we let the user authorize or not optional costs associated to imperfect service, such as backorder costs. This mainly allows us to check if it is possible to reduce global cost by postponing a particular production campaign.

We have seen that both production and inventory costs may be divided into several components, depending on the needed details. On the one hand, we will decompose precisely costs for operational planning: for an optimization on a short time horizon using a short time bucket, it makes sense to capture both the corresponding time dependent production costs and the handling costs penalizing indirect flows after production. On the other hand, tactical production planning is made on larger time horizon based on a longer time period. At this level, we only use basic traditional set-up costs and inventory costs.

4.3 A general method for planning decision support systems

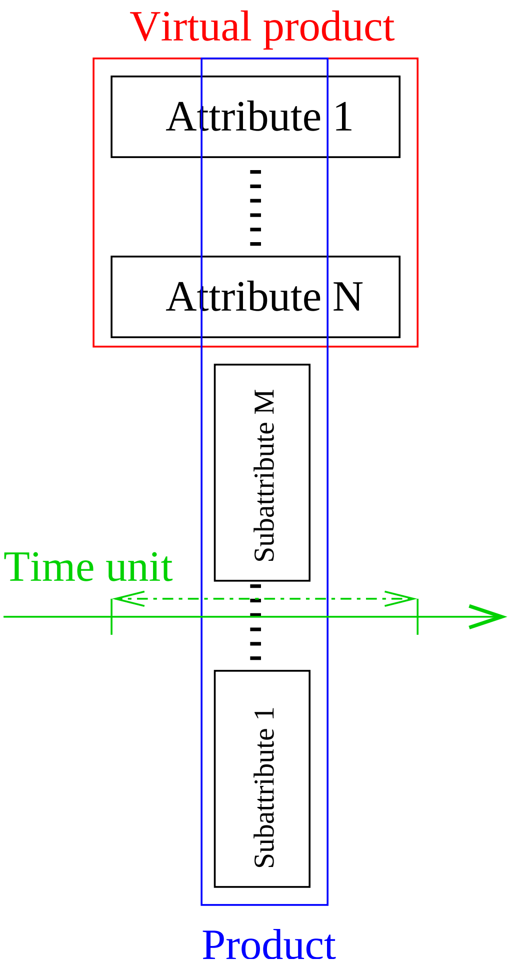
4.3.1 Concepts of job, attribute and sub-attribute

We have seen in section (§4.2.2) that continuous process industry creates products that may often be divided into several characteristics. Each job is defined by various particular characteristics. In this chapter we focus only on a unique job, which is the float glass production. We will see in chapter (5) that other jobs (such as glass transformations) of the Saint-Gobain Glass supply chain are also captured by our model.

Each characteristic of the finished good can take several values. In addition we are able to define all the changeover times between two given values of a given characteristic assuming that other characteristics remain unchanged.

According to the process, we can define the corresponding changeover cost as an opportunity cost due to the lost valuable production time. By definition we can also assume that an impossible changeover between two values correspond to an infinite cost.

This evaluation of all changeovers allows us to understand quantitatively the relative importance of each characteristic, and then to confirm the intuitive classification we may have made. For instance, in the float glass process, a changeover between two colors can take several days whereas it would take several hours between thicknesses and no time at all between two cut lengths. In the following, we plan to use a discrete time model. Let us assume the time horizon and the time period are fixed according to the objective and the level of the decision support model.



We define arbitrarily an attribute as a characteristic that can take only one value per time period and whose changeover times and costs between two values can be considered positive. On the contrary, we define a sub-attribute as a characteristic without any changeover times or costs and whose several values can be produced simultaneously. We assume that we can create a meaningful hierarchy between every attribute and sub-attribute according to their associated characteristic. This remark will lead to important assumptions.

By convention, we use the notion of virtual product to denote a state corresponding to a fixed value for each attribute. This way we also define a finished product as a state described by a fixed value for both each attribute and sub-attribute. Therefore at a given time period, the production line is producing a unique virtual product which is itself divided into several finished products according to the distribution of values of sub-attributes. For a given model, the set of skills of the line includes the gross tonnage of each virtual product. It can either be constant or in a given range. The figure above illustrates this product decomposition into attributes and sub-attributes.

As a conclusion, we define a given job by several attributes and sub-attributes. The distinction between attributes and sub-attributes comes from the various characteristics of products corresponding to a given job which are modeled based on a discrete time whose time period and time horizon are defined by the user, depending on the context of the model use.

4.3.2 Framework for hierarchical planning decisions

We have previously explained our industrial process-driven decomposition of finished products into several characteristics. Then we introduced the notions of attribute and sub-attribute, but we defined them relatively to an arbitrary parameter, namely the time period of the model.

It appears clearly that the definition of a given characteristic as either an attribute or a sub-attribute depends on the goal of the planning decision that we want to study. Would it make sense to plan all the real characteristics at the same time?

Obviously, strategic, tactical or operational planning decisions will use neither the same time period nor the same time horizon. Time periods and horizon are shrinking, and thus their own revision rate in a rolling horizon fashion is increasing (the plan must indeed be periodically revised due to uncertainties in the demand forecasts and production). Furthermore they do not use the same aggregation level of data, which is decreasing according to their decreasing importance. Of course care is required to ensure that at each stage the resulting aggregate plan can be reasonably disaggregated into feasible production plans at the downstream levels.

Various hierarchical levels must be defined by experts of the process, depending on the line and on the firm organization. From our previous decomposition, we immediately see that a characteristic may be considered an attribute in a level, whereas it is just a sub-attribute in the upper levels or it could just be considered constant in the lower levels. In our methodology, each level of optimization corresponds to given time period and horizon. In practice, we noticed that experts define the levels according to the real characteristics of the process. All characteristics can indeed be ranged according to their average changeover time, or average production campaign duration, etc. We can define a typical range of time for each characteristic of real products. For example, colour has a weekly characteristic period, whereas thickness corresponds to half a day.

The planning literature distinguishes between big bucket and small bucket time period. According to our definitions, we use both small bucket time periods for each attribute (one unique value per period) and big bucket time periods for each sub-attribute.

At each planning level, every attribute from upper levels is fixed. In case of the glass industry, tactical planning aims at determining the yearly color planning, whereas operational planning considers colors are fixed. Among the remaining characteristics, one is an attribute if the decision time period is much smaller than its characteristic period. In the opposite case, the characteristic is a sub-attribute.

We notice that sub-attributes can also capture the stochastic behavior of a characteristic, e.g. glass quality depending a distribution of defaults. Some characteristics may indeed not be well mastered to such an extent that it is impossible to produce a unique value of them in a given time period.

Finally, and that is one major remark in the hierarchical approach, the production capacity (or the optimal yield) taken into account at a given planning level must capture the overall time loss due to future changeover times at every lower planning level. The easiest way to deal with this point is to consider that the upper the planning level, the lower the production capacity. Otherwise, it is possible to add in our model at each level of planning optimization a virtual sub-attribute corresponding to the valuable production time, with two basic values: acceptable or not. We will see in the next section that we can easily create some constraints on the feasible domain of each sub-attribute value. An appropriate constraint on the maximum valuable production time would thus permit us to always use the real gross tonnage for virtual products while being sure that we do not take into account the global lost production time due to changeovers in lower levels. This sub-attribute could indeed correspond to the yield of the line at each level.

What is remarkable is that we may use the same optimization model at several levels. Thus in the following we do not precise which hierarchical level we are solving. Depending on the level, we just use various options of the model: the choice of included costs is of course critical. Figure (4.13) illustrates the concept of our decomposition in a hierarchical planning approach.

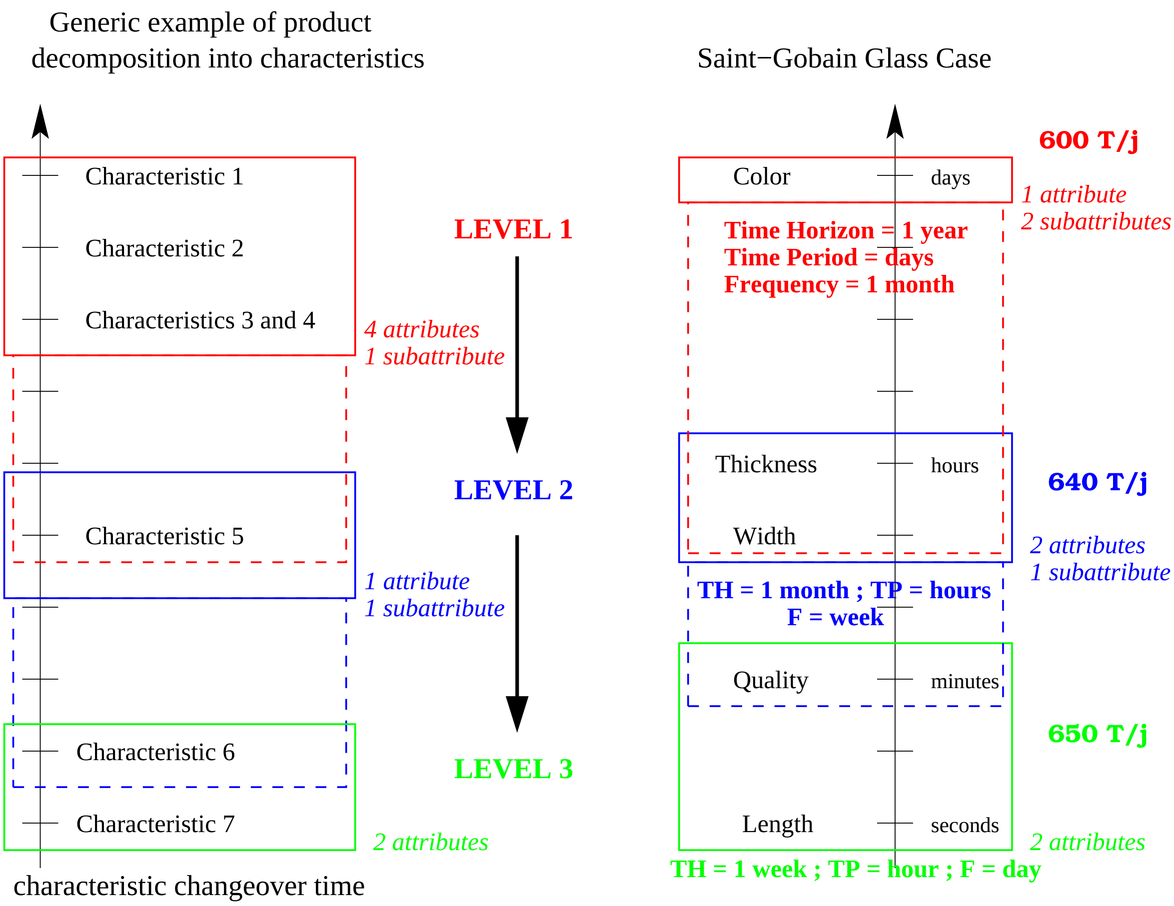


Figure 4.13: Product decomposition into characteristics that become either attributes or subattributes depending on the planning level in a hierarchical approach

4.3.3 Main assumptions

To be able to benefit eventually from the decomposition into attributes and subattributes we make at a given level, we need to assume that all attributes are independent from one another.

Definition 35 ***Two attributes are independent*** *if and only if at any time, for* *each attribute, each possible value or each possible changeover (with associated both time and cost) is independent from the value of any other attributes.*

Assumption 36 *We assume in our model that every attribute is independent from one another.*

At first sight, the assumption of independence between attributes could seem strong. In practice, we do not loose however any generality when two real characteristics are dependent. We can indeed model them simultaneously with a unique attribute which corresponds to a couple of characteristics. Finally, the attribute independence assumption appears to be general and non restrictive.

We have seen that each attribute corresponds to a real characteristic and that every line has its own set of skills for each characteristic. This set is made of the set of possible values and of the set of associated changeovers. Each changeover between two values has associated time and cost.

Assumption 37 *We assume in the remainder that every changeover time is a multiple of the model time period.*

For a given attribute, each changeover can be either a fixed or sequence-dependent set-up. In the later case, we can describe the skills of the line by parameter matrices (one for times and one for costs). We do not need to assume that these matrices satisfy any particular structure such as the triangle inequality. To compute the global transition cost between two different virtual products we need a strong assumption, namely the additive property of changeover costs among various attributes. In addition we use a more intuitive assumption, namely the changeover time between two virtual products is at least the biggest corresponding changeover time between values of a given attribute.

Assumption 38 *We assume the additive property of changeover costs among various attributes, whereas the changeover time between two virtual products is at least the biggest corresponding changeover time between values of a given attribute.*

If we consider a case with Ω independent attributes, each virtual product is a vector of dimension Ω. If we denote $\mathcal{C}$ the function giving the changeover cost and $\mathcal{T}$ the function of changeover time ($\mathcal{C}_\omega$ and $\mathcal{T}_\omega$ correspond to the attribute ω), we have the herein formulas:

$$
\begin{aligned}
\mathcal{C}(\vec{P_1}, \vec{P_2}) &= \sum_{\omega=1}^{\Omega} \mathcal{C}_\omega(P_{1\omega}, P_{2\omega}) \\
\mathcal{T}(\vec{P_1}, \vec{P_2}) &\geq \max_{\omega \in \{1,\ldots,\Omega\}} \mathcal{T}_\omega(P_{1\omega}, P_{2\omega})
\end{aligned}
$$

The assumption about changeover times seems us to be pretty realistic in the context. It is indeed easy to change for example the thickness of the glass ribbon during a longer colour changeover: the first one is done in hidden time. On the contrary the assumption on the changeover costs is for the same reason pretty strong. We hope to relax it in our further research.

Discussion on time scales of attributes

If we focus on the possible time models, we remember that the time period size is given by the decision level and the expectations about of the model outputs. We call it the reference time period, because we aim to personalize the time period for each attribute.

By nature, we have indeed explained that attributes have various characteristic time periods. We propose to decrease the number of Boolean decision variables by giving the possibility to experts to use one particular time scale for each attribute. From a practical point of view, it makes sense because attributes are often ranged according to the technical constraint of each plant. For operational planning of float glass plants, a thickness changeover can be more difficult and long than a width one, and thus experts would like to authorize in their planning at most one thickness change per day, whereas it could be possible to change width four times a day.

Therefore, **based on the given reference time period Δt** that is the basis of the real horizon time $\mathcal{T} = \{t \in [1, N]\}$, **we personalize the time period of each attribute**.

However, we assume that it is a multiple of the reference time period. For each attribute ω, we define:

- a fixed time factor $\eta^{[\omega]}$ that is the link within time scales:

$$t^{[\omega]} = \left\lfloor \frac{t}{\eta^{[\omega]}} \right\rfloor \tag{4.1}$$

- a time period $\Delta t^{[\omega]}$:

$$\Delta t^{[\omega]} = \eta^{[\omega]} \times \Delta t \tag{4.2}$$

Figure (4.14) gives an illustration of this time scale simplification on the first planning level of the generic example introduced on Figure (4.13). We emphasize that there are four attributes but only three different time scales.

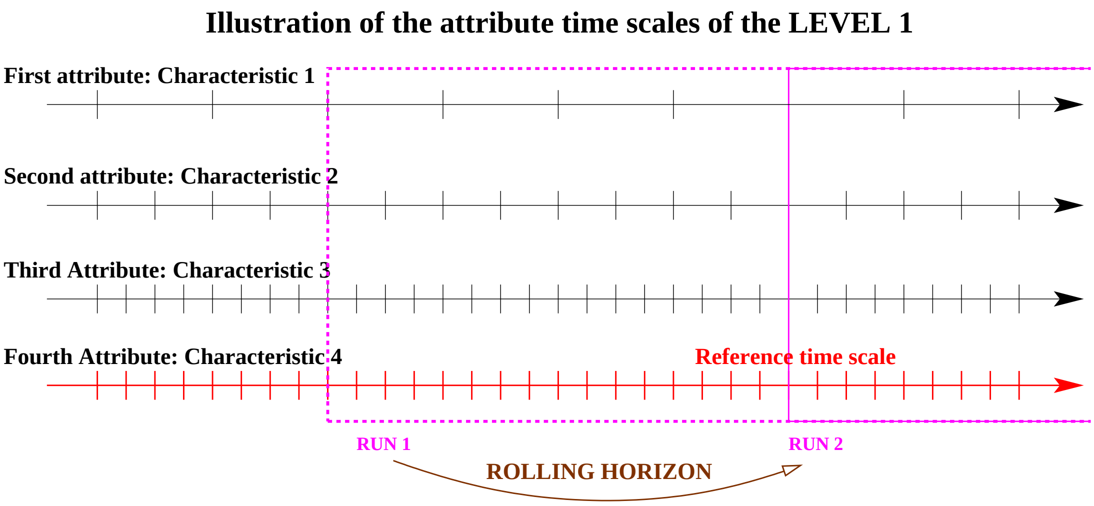


Figure 4.14: Focus on the first level of the generic example of Figure (4.13)

4.4 Development of a mixed integer program

4.4.1 Basic model for single attribute product planning

From the model of the GLSPST we can find in literature, we have derived our model that captures all costs and constraints we have met in an industrial environment. Let us first consider a single machine producing single-attribute products. This matches the classical multi product case with sequence-dependent set-up times and costs. To simplify in this paragraph we do not use the terms of attribute or sub-attribute. From a practical experience, we have noticed that changeover matrices often involve few different elements as compared with their size. For example, on a ten product case, we noticed that changeover times (and so costs too) given by the production experts only involved around ten various values, whereas it could have involved one hundred ones. We believe that this comes from the human evaluation of the matrices: experts first range changeovers by types and then give them values. Thus, our factorization by types of set-up times (and costs) stems from this remark and the purpose to decrease the number of integer variables in our model. To our knowledge, this is an original insight of this chapter.

We use discretized time and periods have a constant duration on the horizon $\mathcal{T}$. We want to plan the production of a set of products. To be exhaustive, we present in section (§4.4.1) the model with sequence-dependent set up times and costs. From this basic model, various extensions may be defined: we present a few in Appendix B.

Case with sequence-dependent set up times and costs

We propose to use the following notation:

- We aim at optimizing the production schedule of a set of products $\mathcal{P} = \{\mathsf{i} \in [1, \mathsf{P}]\}$ on a discrete horizon time $\mathcal{T} = \{\mathsf{t} \in [1, \mathsf{N}]\}$

- Parameters:
  - C_i is the net tonnage capacity corresponding for product i. To simplify this paragraph we consider it fixed (see section (§B.1.3) for an improvement on this point). It takes into account the overall time loss due to changeover times at all lower planning levels.
  - We have some constraints on campaign duration:
    * $\mathcal{D}_i^{m_0}$ and $\mathcal{D}_i^{M}$ are the minimal and the maximal campaign durations of product i. We define the function $\mathcal{D}_i^m : t \to \min\{\mathcal{D}_i^{m_0} \,;\, N - t + 1\}$
    * On particular changeovers we also find some minimal campaign duration before and after a transition. We denote it $\mathcal{D}_m^b(i_1, i_2)$ and $\mathcal{D}_m^a(i_1, i_2)$ for the changeover $\{i_1 \to i_2\}$.
  - D_i^t is the forecast of demand for product i in time-period t.
  - To capture the industrial context, we may impose two different types of constraints on the production, as explained in remark (5).

Remark 5 *Depending on the industrial context, we may impose two types of constraints on the final inventory level. In both cases, we know the initial inventory level (denoted I_i^0) and the demands (or forecasts) D_i^t for each product i over the time horizon $t \in \mathcal{T}$.*

* *If the production over the time horizon $\mathcal{T}$ covers a demand on a much bigger time interval, we must use a fixed constraint on the final inventory level. To do so, we may impose*
  · *either a minimal final inventory level I_i^m.*
  · *or a minimal total produced quantity Q_i^m over $\mathcal{T}$.*

  The arbitrary choice between these two equivalent solutions allows us to match common practice without loss of generality.

* *In particular cases, we need also some stability constraints. For instance, if the production over the time horizon $\mathcal{T}$ covers mainly the corresponding demand, we may satisfy a stability constraint on the global planning horizon, such as the final inventory level should be greater than the initial one:* $\forall\, i, \quad I_i^m \geq I_i^0$. *However, if we use the model following a rolling horizon fashion in which the frozen period is much shorter than the horizon, we may not need this stability constraint.*

In the sequel, we use the second option of the first scenario, based on the notion of minimal total produced quantity Q_i^m. *However, we may replace forthcoming constraints (4.5), (B.12), (4.30), and (4.40) by corresponding constraints of other options.*

- h_i is the inventory cost of product i per unit of product per unit of time.
- $\alpha \in \mathcal{A} = [1, A]$ denotes a type of strictly positive changeover cost
  * $C(i_1, i_2)$ is the function that gives the cost of the changeover $\{i_1 \rightarrow i_2\}$.
  * $\mathcal{T}^C(i_1, i_2)$ is the function that gives the type of cost of the changeover $\{i_1 \rightarrow i_2\}$.
  * C_α is the cost of type α. We notice that $C(i_1, i_2) = C_{\mathcal{T}^C(i_1,i_2)}$
- $\beta \in \mathcal{B} = [1, B]$ denotes a type of changeover duration.
  * $T(i_1, i_2)$ is the function that gives the duration of the changeover $\{i_1 \rightarrow i_2\}$.
  * $\mathcal{T}^T(i_1, i_2)$ is the function that gives the type of duration of the changeover $\{i_1 \rightarrow i_2\}$.
  * T_β is the duration of a changeover of time type β. We notice that $\mathcal{T}(i_1, i_2) = T_{\mathcal{T}^T(i_1,i_2)}$. By definition, we note:
    · $\mathcal{B}^*$ the set of duration types of strictly non negative changeover durations: $\mathcal{B}^* = \{\beta \quad \mathrm{s.t.} \quad T_\beta > 0\}$.

- $T_\beta(t) = \min\{T_\beta, t-1\}$
- $T_\beta^N(t) = \min\{T_\beta, N-t+1\}$

- Every changeover between two products $\{i_1 \to i_2\}$ is characterized by a couple $\{\alpha, \beta\}$.
- We introduce several subsets of products:
  - $S^1(i_2) = \{\, i_1 \quad \text{s.t.} \quad \{i_1 \to i_2\}\, \text{exists}\,\}$
  - $S_T(\beta) = \{\, (i_1, i_2) \quad \text{s.t.} \quad \mathcal{T}^T(i_1, i_2) = \beta\,\}$
  - $S_C(\alpha) = \{\, (i_1, i_2) \quad \text{s.t.} \quad \mathcal{T}^C(i_1, i_2) = \alpha\,\}$
  - $S_T^1(\beta) = \{\, i_1 \quad \text{s.t.} \quad \exists\, i_2 \quad \text{s.t.} \quad \mathcal{T}^T(i_1, i_2) = \beta\,\}$
  - $S_C^1(\alpha) = \{\, i_1 \quad \text{s.t.} \quad \exists\, i_2 \quad \text{s.t.} \quad \mathcal{T}^C(i_1, i_2) = \alpha\,\}$
  - $S_T^2(\beta) = \{\, i_2 \quad \text{s.t.} \quad \exists\, i_1 \quad \text{s.t.} \quad \mathcal{T}^T(i_1, i_2) = \beta\,\}$
  - $S_C^1(i_2, \alpha) = \{\, i_1 \quad \text{s.t.} \quad \mathcal{T}^C(i_1, i_2) = \alpha\,\}$
  - $S_C^2(i_1, \alpha) = \{\, i_2 \quad \text{s.t.} \quad \mathcal{T}^C(i_1, i_2) = \alpha\,\}$
  - $S_T^1(i_2, \beta) = \{\, i_1 \quad \text{s.t.} \quad \mathcal{T}^T(i_1, i_2) = \beta\,\}$
  - $S_T^2(i_1, \beta) = \{\, i_2 \quad \text{s.t.} \quad \mathcal{T}^T(i_1, i_2) = \beta\,\}$
  - *We may notice that:* $i_1 \notin (S_C^2(i_1, \alpha) \cup S_T^2(i_1, \beta))$, $i_2 \notin (S_C^1(i_2, \alpha) \cup S_T^1(i_2, \beta))$
- By convention the range [a, b] is empty if b<a.
- M is the number of integer variables.

- Decision variables: y_i^t is a Boolean variable indicating if product i is produced during period t.

- Stack variables:

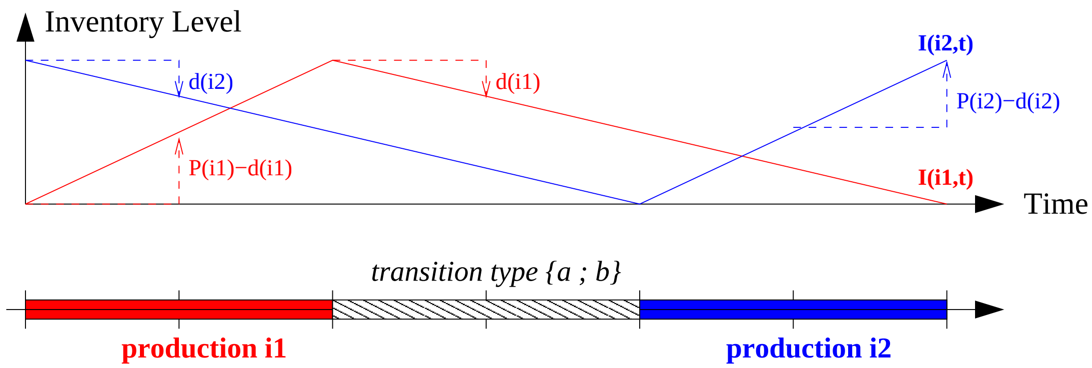
  - I_i^t is the on-hand inventory of product i at the end of time period t. This continuous variable must be non-negative because we forbid back-orders.

– w_α^t is a Boolean variable that equals 1 during the first period of set-up of type of cost α in time t.

– v_β^t is a Boolean variable that equals 1 during each period of the set-up time of type β in time t.

Remark 6 *To simplify the model desciption in the following, we do not write down the domain constraints on variables.*

Basically, we describe on Figure 4.15 the way binary variables y_i^t, w_α^t and v_β^t must be thought. This illustration is based on a simple mono-attribute model: two products exist, defined by two values i_1 and i_2 of a unique attribute. The production capacity and demands are constant. The transition from i_1 to i_2 is characterized by both the type of cost a and type of time b, denoted $\{a, b\}$.



y(i1,t)	**1**	**1**	0	0	0	0
y(i2,t)	0	0	0	0	**1**	**1**
v(b,t)	0	0	**1**	**1**	0	0
w(a,t)	0	0	**1**	0	0	0

Figure 4.15: Illustration of binary variables on a mono-attribute case

We can write the following MIP:

$$\min \Big(\sum_t \big(\sum_\alpha C_\alpha \times w_\alpha^t + \sum_i h_i \times \frac{I_i^t + I_i^{t-1}}{2}\big)\Big) \tag{4.3}$$

$$\forall i,\ \forall t \qquad I_i^{t-1} + C_i \times y_i^t \;=\; I_i^t + D_i^t \tag{4.4}$$

$$\forall i \qquad C_i \times \sum_t y_i^t \;\geq\; Q_i^m \tag{4.5}$$

$$\forall i_2,\ \forall t \in [2, N] \qquad \sum_{i_1 \notin S^1(i_2)} y_{i_1}^{t-1} \;\leq\; 1 - y_{i_2}^t \tag{4.6}$$

$$\forall \beta \in \mathcal{B}^*,\ \forall i_2 \in S_{\mathcal{T}}^2(\beta),\ \forall t \in [2, N]$$

$$\sum_{i_1 \in S_{\mathcal{T}}^1(i_2,\beta)} \sum_{k=1}^{T_\beta(t)} y_{i_1}^{t-k} \;\leq\; (1 - y_{i_2}^t) \times M \tag{4.7}$$

$$\forall \beta \in \mathcal{B}^*,\ \forall i_2 \in S_{\mathcal{T}}^2(\beta),\ \forall t \in [2 + T_\beta, N]$$

$$T_\beta \times \Big(y_{i_2}^t + \sum_{i_1 \in S_{\mathcal{T}}^1(i_2,\beta)} y_{i_1}^{t-T_\beta-1} - 1\Big) \leq \sum_{k=1}^{T_\beta} v_\beta^{t-k} \tag{4.8}$$

$$\forall i_1,\ \forall \beta,\ \forall t \in [T_\beta + 2, N] \qquad \big(y_{i_1}^{t-T_\beta-1} + v_\beta^{t-T_\beta}\big) - 1 \;\leq\; \sum_{i_2 \in S_{\mathcal{T}}^2(i_1)} y_{i_2}^t \tag{4.9}$$

$$\forall \beta,\ \forall t \in [2, N-1] \qquad T_\beta^N(t) \times (v_\beta^t - v_\beta^{t-1}) \;\leq\; \sum_{k=0}^{T_\beta^N(t)-1} v_\beta^{t+k} \tag{4.10}$$

$$\forall \alpha,\ \forall (i_1, i_2) \in S_{\mathcal{C}}(\alpha),\ \forall t \in [1, N - T_\beta]$$

$$y_{i_2}^{t+\mathcal{T}(i_1,i_2)} + y_{i_1}^{t-1} - 1 \;\leq\; w_\alpha^t \tag{4.11}$$

$$\forall \alpha,\ \forall t \in [2, N] \qquad w_\alpha^t \leq \sum_{i \in S_C^1(\alpha)} y_i^{t-1} \tag{4.12}$$

$$\forall \alpha,\ \forall t \qquad w_\alpha^t \leq \sum_{(i_1,i_2) \in S_C(\alpha)} y_{i_2}^{t+T_{\mathcal{J}\mathcal{T}(i_1,i_2)}} \tag{4.13}$$

$$\forall t \qquad \sum_i y_i^t + \sum_\beta v_\beta^t = 1 \tag{4.14}$$

$$\forall t \qquad \sum_\alpha w_\alpha^t \leq 1 \tag{4.15}$$

$$\sum_\alpha w_\alpha^1 + \sum_\beta v_\beta^1 = 0 \tag{4.16}$$

$$\forall i,\ \forall t < N \qquad y_i^t + y_i^{t+1} \leq 2 - \sum_\alpha w_\alpha^{t+1} \tag{4.17}$$

$$\forall i,\ \forall t \qquad \sum_{k=0}^{\mathcal{D}_i^m(t)} y_i^{t+k} \geq \mathcal{D}_i^m(t) \times (y_i^t - y_i^{t-1}) \tag{4.18}$$

$$\forall i,\ \forall t \in [1, N - \mathcal{D}_i^M] \qquad \sum_{k=0}^{\mathcal{D}_i^M} y_i^{t+k} \leq \mathcal{D}_i^M \tag{4.19}$$

$$\tag{4.20}$$

$$\forall \beta,\ \forall (i_1, i_2) \in S_T(\beta),\ \forall t \in [\mathcal{D}_m^b(i_1, i_2), N]$$

$$\mathcal{D}_m^b(i_1, i_2) \times \left((y_{i_1}^t + y_{i_2}^{t+T_\beta+1} + \sum_{k=1}^{T_\beta} v_\beta^{t+k}) - (T_\beta + 2) + 1 \right) \leq \sum_{k=0}^{\mathcal{D}_m^b(i_1,i_2)-1} y_{i_1}^{t-k} \tag{4.21}$$

$$\forall \beta,\ \forall (i_1, i_2) \in S_T(\beta),\ \forall t \in [1, N - \mathcal{D}_m^a(i_1, i_2) - T_\beta]$$

$$\mathcal{D}_m^a(i_1, i_2) \times \left((y_{i_1}^t + y_{i_2}^{t+T_\beta+1} + \sum_{k=1}^{T_\beta} v_\beta^{t+k}) - (T_\beta + 2) + 1 \right) \leq \sum_{k=1}^{\mathcal{D}_m^a(i_1,i_2)} y_{i_2}^{t+T_\beta+k} \tag{4.22}$$

The objective function (4.3) is the minimization of the sum of variable production costs and inventory costs. Global constraints are mainly the inventory balance equations (4.4), the respect of the minimal final inventory levels (4.5). Constraints (4.6) to (4.14) correspond to the structural relationships between the three families of Boolean decision variables. Constraint (4.6) forbids impossible changeover between products and (4.7) enforces i_1 to be at least not produced during the authorized transition period before the first production period of i_2. Constraints (4.8) (4.9) and (4.10) link y_i^t and v_β^t variables. We particularly notice that (4.9) forbids a changeover during a single product campaign and (4.10) enforces that a changeover lasts at least its particular duration. Constraints (4.11) to (4.13) link y_i^t and w_α^t variables. Equality (4.14) assures that the line is either producing or in transition. Constraint (4.18) enforces minimum lot sizes in order to avoid set-up changes without product changes, avoiding an incorrect calculation of set-up costs/times in an optimal solution if set-up costs/times do not satisfy the triangle inequality. Constraint (4.19) is a similar constraint on maximal lot sizes, whereas (4.21) and (4.22) deal with minimal campaign durations before and after a special changeover.

So far, we notice that taking into account various types β of changeover times has introduced various Boolean variables v_β^t. However, during each time period we force that at most one is equal to one. Thus **we propose to simplify it by using only the Boolean variables v^t** indicating whether the line is on transition between two products or not. If we consider the sequence-dependent MIP presented above, we only need to modify few constraints to introduce this simplifying change.

Firstly, we cancel the constraints (4.10). Secondly, we transform constraints (4.8) into (4.23), (4.9) into (4.24) and (4.14) into (4.25). We introduce new notation:

- $T_M = \max_{\beta \in \mathcal{B}} T_\beta$
- $S_T^2(i_1, \delta t) = \{\, i_2 \quad \text{s.t.} \quad T_{\mathcal{J}^T(i_1, i_2)} = \delta t \,\}$
- $$v^t(\delta t) = \begin{cases} v^t & \text{if } \delta t \in [1, T_M - 1]; \\ 0 & \text{if } \delta t = T_M. \end{cases}$$

The simplified constraints with the new variables v^t are as follows:

$$\forall \beta \in \mathcal{B}^*,\ \forall i_2 \in S_{\mathcal{T}}(\beta),\ \forall t \in [2 + T_\beta, N]$$

$$T_\beta \times \big(y_{i_2}^t + \sum_{i_1 \in S_{\mathcal{T}}^1(i_2, \beta)} y_{i_1}^{t - T_\beta - 1} - 1\big) \le \sum_{k=1}^{T_\beta} v^{t-k} \tag{4.23}$$

$$\forall i_1,\ \forall \delta t \in [1, T_M],\ \forall t \in [\delta t + 2, N]$$

$$\left(y_{i_1}^{t-\delta t-1} + \Big(\sum_{k=1}^{\delta t} \nu^{t-k} - (\delta t - 1)\Big)\right) - 1 \leq \sum_{i_2 \in S_T^2(i_1, \delta t)} y_{i_2}^t + \nu^t(\delta t) \tag{4.24}$$

$$\forall t \qquad \sum_i y_i^t + \nu^t = 1 \tag{4.25}$$

One interesting point lies in the constraint (4.24) which enforces that given an initial product, a changeover can either finish with possible final products (corresponding to its duration) or continues. It also forbids changeovers longer than the longest one.

From this basic model, various extensions may be defined: we introduce few ones in the appendix section (§B.1). Section (§B.1.1) presents how we do simplify the previous MIP when set-ups are not sequence-dependent. Finally, we provide in section (§B.1.3) an improvement in the model of the production line.

In this section (§4.4.1) we have proposed a general mixed integer linear programming formulation that allows us to model a multi product planning without either fixed or sequence dependent set ups. We show now how to use it as a building block in a more general structure based on our previous decomposition in several attributes and sub-attributes.

4.4.2 A planning model for multi attribute products

It appears intuitively that the assumptions of independence between attributes as well as additive changeover costs between attributes allow us to use the previous MILP individually for each attribute. We provide a model that can simplify a problem of industrial size corresponding to our product decomposition. To simplify the understanding, we assume that we do not work with sub-attributes in this paragraph. Besides, we work with a given hierarchical planning level. Thus, we know exactly which attributes we want to schedule simultaneously. We note $\mathcal{A} = \{\omega \in [1, \Omega]\}$ the set of attributes.

Multi attribute MILP

We aim at scheduling the production of a set of products $\mathcal{P} = \{i \in [1, P]\}$ which are decomposed into Ω attributes on a discrete horizon time $\mathcal{T} = \{t \in [1, N]\}$.

Correspondence between products and attributes is given by a matrix $\mathcal{M}$ of dimension $\Omega \times P$. $\mathcal{M}_{\omega i}$ represents the value of the product i for the attribute ω.

Remark 7 *Several products may be defined by the same vector of attribute values, i.e. the matrix $\mathcal{M}$ may have several identical columns. In such a case, we create the sets of twin products $\{\mathcal{S}_l \quad l \in [1, L]\}$ and their associated matrix $\mathcal{M}^*$ (of size $\Omega \times L$) created by keeping the L independent columns of $\mathcal{M}$. We denote $\mathcal{M}^*_{\omega l}$ its elements. Of course, we have:*

$$\forall l,\ \forall i \in \mathcal{S}_l \qquad \mathcal{M}_{\omega i} = \mathcal{M}^*_{\omega l} \tag{4.26}$$

The skills of a given production line lie both in the possible values and changeover matrices within values of each attribute and in the capacity of the line for each product. We work with the improved version of the model of production line.

On the one hand, we keep the notation of paragraph (§4.4.1) and (§B.1.3) which do not depend on the concept of attribute, i.e. the availability of the line $A(t)$, its capacities C_i^m and C_i^M, its campaign duration constraints $\mathcal{D}_i^m$ and $\mathcal{D}_i^M$, the demand D_i^t and Q_i^m, the inventory cost h_i. On the other hand, we modify all notation which depends on the concept of attribute:

- Each attribute w is characterized by a time scale (defined by (4.1) and (4.2)) on which we analyze at each time period $t^{[\omega]}$:
  - a set of possible values $j^{[\omega]} \in \mathcal{V}^{[\omega]} = [1, V^{[\omega]}]$. Of course, we have the relations:

$$\forall i,\ \forall \omega \quad \exists\,!\, j^{[\omega]} \in \mathcal{V}^{[\omega]} \quad \text{s.t.} \quad \mathcal{M}_{\omega i} = j^{[\omega]} \tag{4.27}$$

And thus from equalities (4.26): $\forall l, \forall \omega \quad \exists! j^{[\omega]} \in \mathcal{V}^{[\omega]} \quad \text{s.t.} \quad \mathcal{M}^*_{\omega l} = j^{[\omega]}$

  - a set of types of changeover costs $\alpha^{[\omega]} \in \mathcal{A}^{[\omega]} = [1, A^{[\omega]}]$
  - a set of types of changeover durations $\beta^{[\omega]} \in \mathcal{B}^{[\omega]} = [1, B^{[\omega]}]$

- Decision variables:
  - I_i^t is the on-hand inventory of product i at the end of time period t. This continuous variable must be non-negative.
  - Z_i^t is a Boolean variable indicating whether product i is produced during time period t.
  - P_i^t (real variable) represents the production of product i during time period t. It is a non negative variable.
  - For each attribute ω, we adapt the integer variables introduced in previous section (§4.4.1):
    * $y_{j^{[\omega]}}^{t^{[\omega]}}$ determines whether or not the value $j^{[\omega]}$ is produced during time period $t^{[\omega]}$.
    * $w_{\alpha^{[\omega]}}^{t^{[\omega]}}$ corresponds to a changeover cost of type $\alpha^{[\omega]}$ in $t^{[\omega]}$.
    * $v_{\beta^{[\omega]}}^{t^{[\omega]}}$ corresponds to a changeover of time type $\beta^{[\omega]}$ during $t^{[\omega]}$.

Basically, we describe on Figure 4.18 the way binary variables Z_i^t, $y_{j^{[\omega]}}^{t^{[\omega]}}$, and $w_{\alpha^{[\omega]}}^{t^{[\omega]}}$ as well as $v_{\beta^{[\omega]}}^{t^{[\omega]}}$ must be thought. This illustration is based on a simple 2-attribute model: six products exist, defined on various pairs of values taken by two attributes (cf. Figure 4.16).

Figure 4.17 defines the two attributes. The first attribute can take two values, is characterized by one type of transition cost $a^{[1]}$ and two types of times $b_0^{[1]}$ and $b_1^{[1]}$ and only needs a discrete time-scale based on a time period four times bigger than the reference one. The second one has three values, one type of transition cost $a^{[2]}$, three types of transition times $b_0^{[2]}$, $b_1^{[2]}$ and $b_2^{[2]}$, and is based on the reference time-scale.

Demand and production capacity for each product are assumed constant.

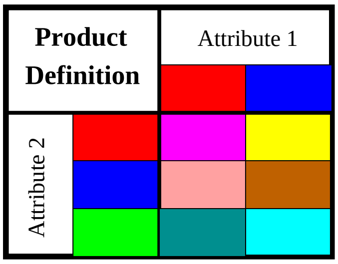


Figure 4.16: Definition of six products in a simple 2-attribute case

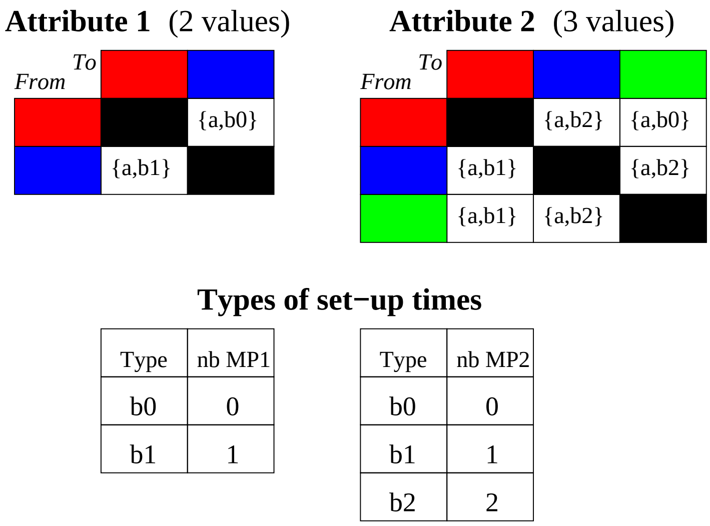


Figure 4.17: Definition of transition in a simple 2-attribute case

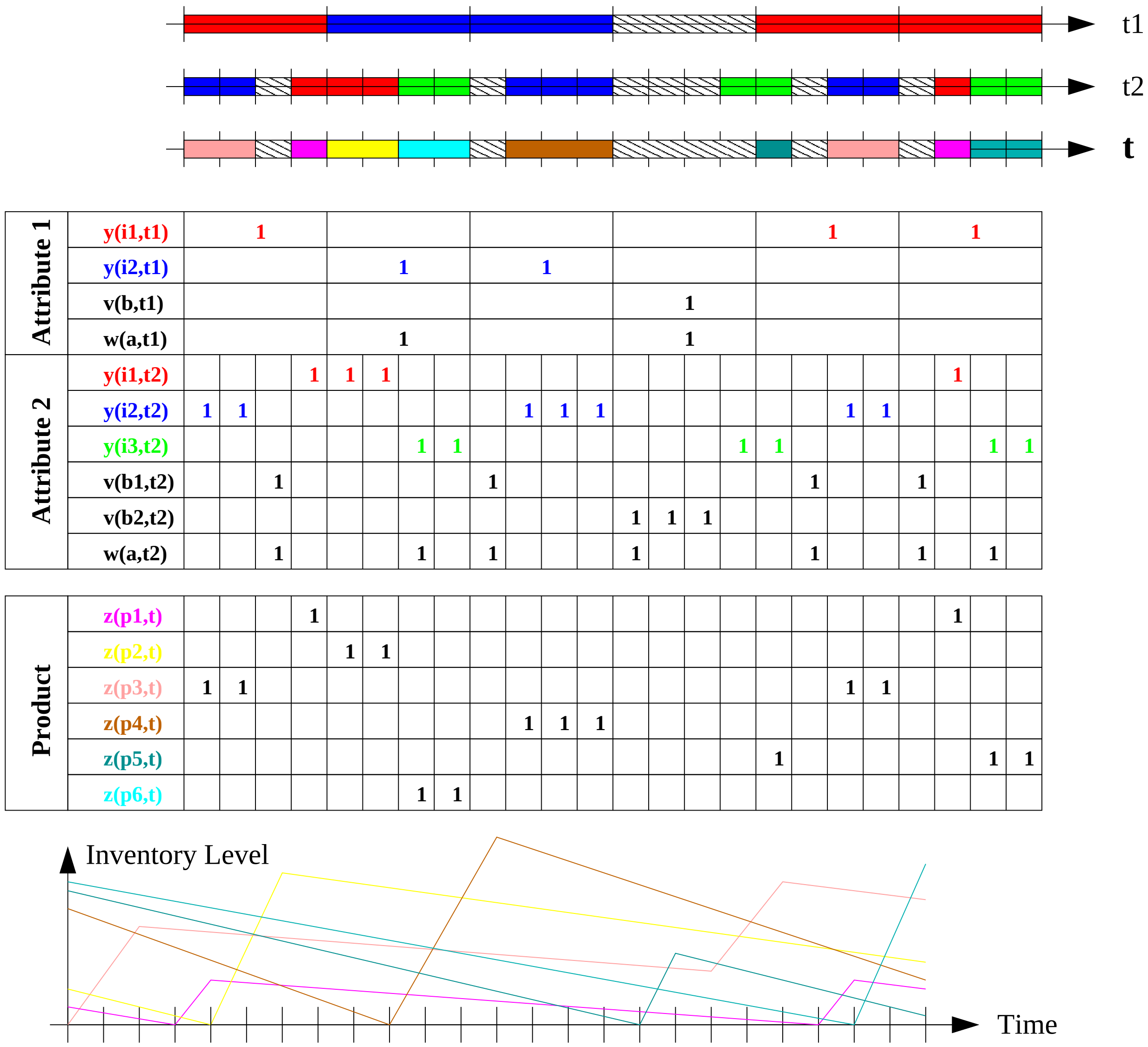


Figure 4.18: Illustration of binary variables on a multi-attribute case

We can write the following MIP:

$$\min\Big(\sum_{\omega}\sum_{\alpha^{[\omega]}}\sum_{t^{[\omega]}} C_{\alpha^{[\omega]}} \times w^{t^{[\omega]}}_{\alpha^{[\omega]}} + \sum_{t}\sum_{i} h_i \times \frac{I_i^t + I_i^{t-1}}{2}\Big) \tag{4.28}$$

$$\begin{aligned}
\forall i,\ \forall t \qquad I_i^{t-1} + P_i^t &= I_i^t + D_i^t && (4.29)\\
\forall i \qquad \sum_t P_i^t &\geq Q_i^m && (4.30)\\
\forall i,\ \forall t \qquad P_i^t &\leq C_i^M \times A(t) \times Z_i^t && (4.31)\\
\forall i,\ \forall t \qquad P_i^t &\geq C_i^m \times A(t) \times Z_i^t && (4.32)\\
\forall \omega,\ \forall l,\ \forall t \qquad \sum_{i\in\mathcal{S}_l} Z_i^t &\geq 1 - \Omega + \sum_{\omega} y^{t^{[\omega]}}_{\mathcal{M}^*_{\omega l}} && (4.33)\\
\forall \omega,\ \forall l,\ \forall t \qquad \Omega \times \sum_{i\in\mathcal{S}_l} Z_i^t &\leq \sum_{\omega} y^{t^{[\omega]}}_{\mathcal{M}^*_{\omega l}} && (4.34)\\
\forall i,\ \forall t \qquad \sum_i Z_i^t &\leq 1 && (4.35)\\
\forall i,\ \forall t \qquad \mathcal{D}_i^m \times (Z_i^t - Z_i^{t-1}) &\leq \sum_{k=0}^{\mathcal{D}_i^m(t)} Z_i^{t+k} && (4.36)\\
\forall i,\ \forall t \in [1, N - \mathcal{D}_i^M] \qquad \mathcal{D}_i^M &\geq \sum_{k=0}^{\mathcal{D}_i^M} Z_i^{t+k} && (4.37)
\end{aligned}$$

In addition, for each attribute ω, we add the constraints within variables $y^{t^{[\omega]}}_{j^{[\omega]}}$, $w^{t^{[\omega]}}_{\alpha^{[\omega]}}$ and $v^{t^{[\omega]}}_{\beta^{[\omega]}}$ derived from constraints (4.6) to (4.22) presented in section (§4.4.1).

The objective function (4.28) is the minimization of the sum of variable production and inventory costs. We use obviously the assumption of additive changeover costs between attributes. We notice that this assumption is not a key one, and that we could have used another one. For example, we could use various weights w_ω per attribute ω. The first member of the objective function would be:

$$\min \sum_{\omega}\sum_{\alpha^{[\omega]}}\sum_{t^{[\omega]}} w_\omega \times C_{\alpha^{[\omega]}} \times w^{t^{[\omega]}}_{\alpha^{[\omega]}}$$

Global constraints are written at the product level: (4.29) is the inventory balance equation, (4.30) enforces that minimal final inventory levels are satisfied, whereas (4.31) and (4.32) are the capacity and availability constraints. The link between the products and the attributes variables lies in the (A.9) and (A.10) inequalities. Constraint (4.35) enforces that at most one product is produced during each time period. Constraint (4.36) enforces minimum lot sizes on each product, and (4.37) is the equivalent constraint on maximal lot sizes.

Finally, we observe that the sets of structural constraints of each attribute are independent from one another. Thus, each attribute may either be characterized by sequence dependent set ups or by fixed ones, as well as by easier versions described in section (§4.4.1).

In an industrial context, we have met additional goals, such as to impose the final product, or to authorize an interruption in a campaign. We deal with these extensions in appendix (§B.2)

In this paragraph, we have voluntarily forgotten the concept of sub-attributes. To achieve our global model, we still need to explain how to take them into account.

4.4.3 Adding the sub-attributes

What happens when we go back to our multi attribute and multi sub-attribute products? By definition of the virtual product concept previously introduced (see (§4.3.1)) the model that we just have introduced in the later part is suitable to catch them. We note $\mathcal{B} = \{\lambda \in [1, \Lambda]\}$ the set of sub-attributes.

Henceforth, we aim at scheduling the production of a set of virtual products $\mathcal{P}_V = \{i \in [1, P_V]\}$ which are decomposed into Ω attributes on a discrete horizon time $\mathcal{T} = \{t \in [1, N]\}$. They correspond to a set of real products $\mathcal{P}_R = \{p \in [1, P_R]\}$ through Λ sub-attributes.

From each virtual product i, we can produce $\prod_{\lambda=1}^{\Lambda} V_S^{[\lambda]}$ various final products p.

Each final product p is indeed a vector of dimension $\Omega + \Lambda$, corresponding to one virtual product i (which is given by one fixed value per attribute ω, i.e. a vector of dimension Ω) and a fixed value for each sub-attribute λ. By convention, we denote $\mathcal{P}_R^{[i]} \subset \mathcal{P}_R$ the set of real products that may be derived from the virtual product i.

On the one hand, we use some former notation and parameters: virtual products correspond to former products $\mathcal{P} = \{i \in [1, P]\}$ introduced in previous section (i.e. section (§4.4.2)). We keep the notion of capacity of the production line (parameters C_i^m, C_i^M and $\mathcal{A}(t)$) and the constraints on campaign durations ($\mathcal{D}_i^m$ and $\mathcal{D}_i^M$). Naturally, we keep the variables Z_i^t representing the production of virtual product i. In the same way, attributes are characterized exactly as in the previous part.

On the other hand, we characterize each sub-attribute λ by:

- a set of possible values $k^{[\lambda]} \in \mathcal{V}_S^{[\lambda]} = [1, V_S^{[\lambda]}]$
- λ describes the rank of the sub-attribute. For two sub-attributes λ_1 and λ_2, the rank has a direct impact on the way we deal with the variables. Whether $\lambda_1 < \lambda_2$, then for all t we compute the values of λ_2 that are derived from each value of λ_1.

The main difference with section (§4.4.2) is that we now deal with inventory levels and demands at the real product $\mathcal{P}_R$ level. We use the following notation:

- Parameters:
  - D_p^t is the demand for real product p during time period t.
  - As explained in Remark 5[d], constraints on initial and final inventories depend on the context. Here we still use the the notion of minimal total produced quantity Q_p^m.
  - h_p is the inventory cost of product p.
- Decision variables:

[d]on page 131

- I_p^t is the on-hand inventory of final product k at the end of time period t. This continuous variable must be non-negative.
- Z_i^t is a Boolean variable indicating if the virtual product i is produced during time period t.
- P_i^t is the production of virtual product i during time period t. It is a non negative variable.
- R_p^t is the production of real product p during time period t. It is a non negative variable.

Clearly, a slight modification of the MILP presented in section (§4.4.2) gives us the new one. We just need to express the new objective function at the real product level and to update constraints on inventory balances and minimal produced quantities ((4.38), (4.39) and (4.40) clearly come from (4.28), (4.29) and (4.30)), while adding the relationships between virtual and real products through equalities (4.41).

$$\min\Big(\sum_{\omega}\sum_{\alpha^{[\omega]}}\sum_{t^{[\omega]}} C_{\alpha^{[\omega]}} \times w_{\alpha^{[\omega]}}^{t^{[\omega]}} + \sum_{t}\sum_{p} h_p \times \frac{I_p^t + I_p^{t-1}}{2}\Big) \tag{4.38}$$

$$\begin{aligned} \forall p,\ \forall t \qquad I_p^{t-1} + R_p^t &= I_p^t + D_p^t && (4.39)\\ \forall p \qquad \sum_{t} R_p^t &\geq Q_i^m && (4.40)\\ \forall i,\ \forall t \qquad \sum_{p \in \mathcal{P}_R^{[i]}} R_p^t &= P_i^t && (4.41) \end{aligned}$$

We may introduce optional linear constraints on the production line skills, at the sub-attribute level. The rank of each attribute makes it possible to capture precise industrial constraints we have met in practice. For instance, whatever the sub-attribute $\lambda > 1$, we may express minimal and maximal proportions on each produced value $k^{[\lambda]} \in \mathcal{V}_S^{[\lambda]}$ among each value of the upper rank $\lambda - 1$.

If we consider the planning of the Float process at an operational level, we model the quality and the cut length as sub-attributes. Quality is the most important one ($\lambda = 1$). If we consider two different qualities ($k_1^{[1]}$ and $k_2^{[1]}$) and two lengths ($k_1^{[2]}$ and $k_2^{[2]}$), we have met constraints such as:

- percentage of $k_1^{[1]}$ must belong to $[\frac{1}{10}, \frac{9}{10}]$.
- percentage of $k_1^{[2]}$ which is cut in the:
  - $k_1^{[1]}$ part of any virtual product must belong to the range $[\frac{2}{10}, \frac{8}{10}]$
  - $k_2^{[1]}$ part must belong to $[\frac{4}{10}, \frac{6}{10}]$.

In the beginning of the chapter, we claimed that various options were possible concerning the inventory costs. Firstly, we let the user authorize or not optional costs associated to imperfect service, such as backorder costs. Secondly, we can include a handling cost, which corresponds to the long and expensive handling operations to put the glass in and out of the warehouse. These extensions of the model on the inventory modeling are handled in appendix (§B.3).

4.5 Implementation and Gain

We used the best available commercial software CPLEX (see [ILOa]) to solve test problems based on real-life data sets. We found that the linear relaxation of the model was very poor, due to the bad lower bound obtained for changeover costs within various attribute values. Thus, we developed as a preprocessing step a simple dynamic program to compute a lower bound of these costs in appendix (§B.4.1).

From a practical point of view, we developed the PLANEO software based on a `C++` code, following an object oriented fashion which is introduced in appendix (§B.4.2). This software has been implemented and used in 2004 by four plants of Saint-Gobain Glass. Based on encouraging results exposed in section (§4.5.4), implementation in other plants keeps on going.

4.5.1 Software architecture

Our modeling aimed at making regular production planning possible on a PC. In order for the system to be accepted, the user friendliness of the tool was a very important feature.

We choose as a first step prototype an MS-ACCESS database to store all the necessary data, and a friendly user-interface was also programmed in MS-ACCESS. The interface invokes the model generator and solver, and interprets the output after optimization. This approach allows the developer to distinguish completely between the data and the structure of the model. Communication between user-interface and solver is exclusively by ASCII files. The interface includes a number of switches to enable the use of the model for various purposes.

One very important decision has been to implement the software not only as an optimization tool of a data set but also as a simulation tool of a human solution. Given a data set of production line parameters and inventory parameters as well as forecast demands, we offer the possibility to the end user to compare its own production plan to the optimized one.

Figures (4.19) and (4.20) present some screen shots of the final software we have developed for Saint-Gobain Glass.

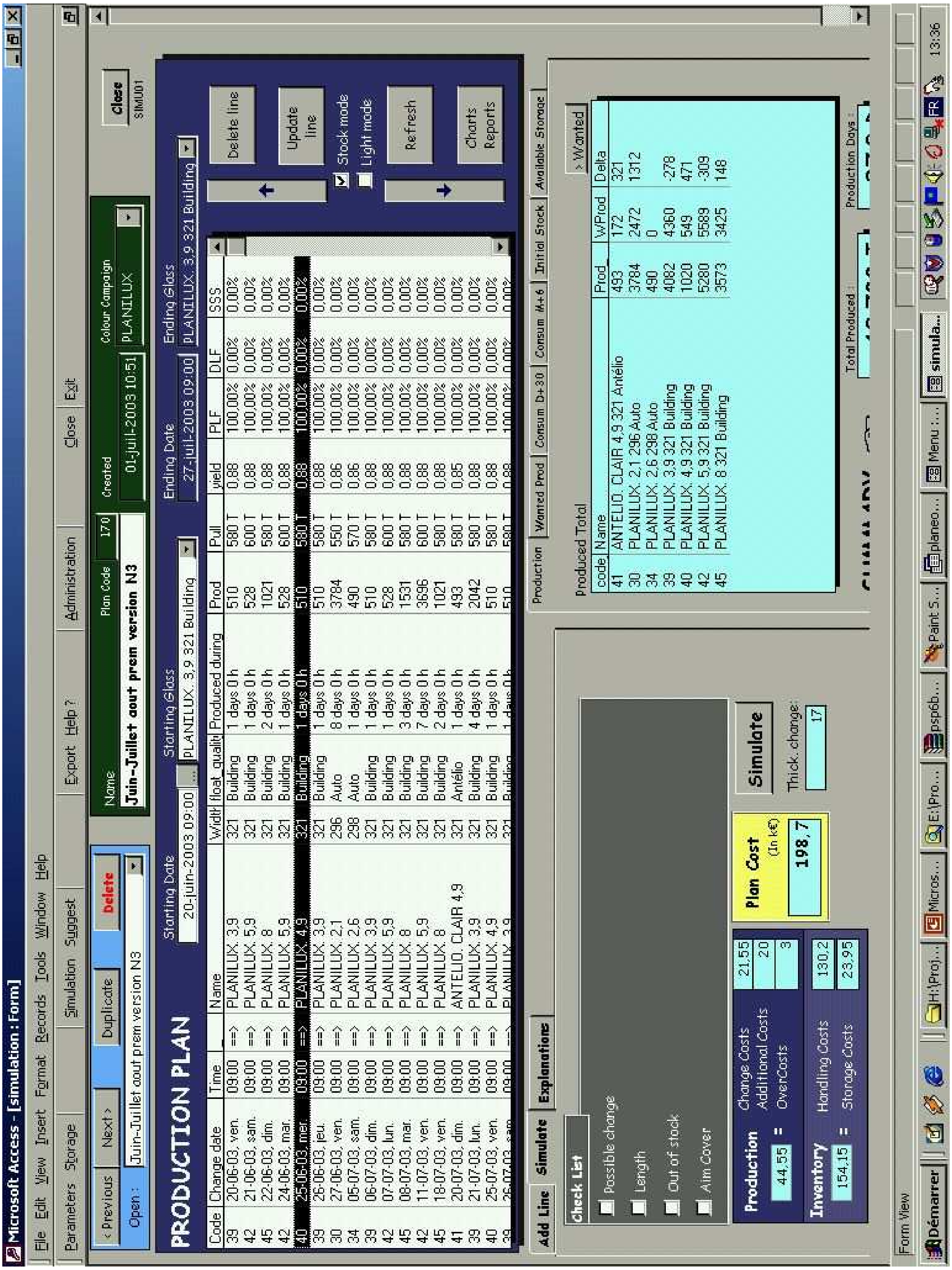


Figure 4.19: Simulation Module of the PLANEO Software

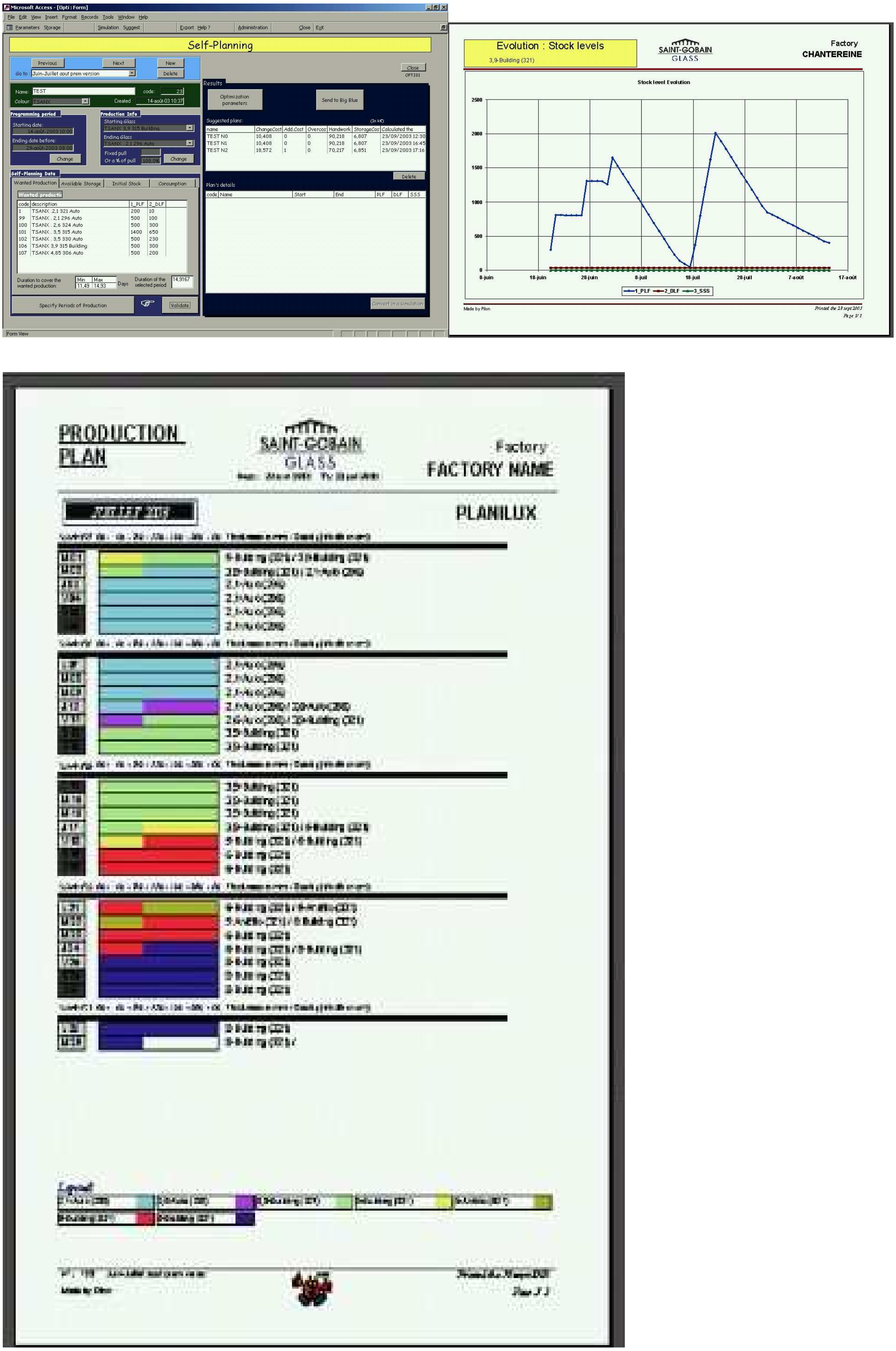


Figure 4.20: Screen shots of the PLANEO Software

4.5.2 PLANEO: an application to the operational scheduling in the glass manufacturing industry

In the frame of a project carried out in partnership between Saint-Gobain Glass and Saint-Gobain Recherche, we aimed at developing a tool to assist in the determination of the short-term float line production planning.

This operational scheduling problem considers a unique job (i.e. float glass production), made of two attributes which are thickness and width of the glass ribbon, and two sub-attributes, the quality and the size of the glass sheet. The most important glass characteristic, the colour, is neither an attribute nor a sub-attribute at this level of production planning. This tool could be used whatever the colour campaign, which is fixed. PLANEO aims indeed at scheduling production campaigns in a given colour over several weeks (time horizon), with discrete time periods of several hours. It is used separately in each plant by the production scheduling manager as a decision tool.

A production plan is a sequence of product campaigns. For some specified needs, the choice of a production plan is done to minimize mainly production costs and storage costs, while satisfying various constraints. Three types of production costs influence this choice: changeover costs, additional costs related to production changes done at some periods of the week and over-costs to produce some products during weekends. Fixed production costs which depend on the plant are not taken into account because we work on a single plant perimeter.

In his (or her) plant, the production manager specifies the set of costs and constraints. For instance, he defines production capacities, minimum and maximum campaign durations, the availability of the line, on-hand values for each attribute and sub-attribute of the line job, transition matrices between values of each attribute (i.e. thickness and width), etc. For instance, we illustrate on Figure (4.21) what may be a transition cost matrix within various possible thicknesses.

The user specifies manually initial and minimal final inventory levels and the deterministic dynamic demand over the time horizon. He sets the inventory cost formula as well as the handling cost (aiming at maximizing direct shipping without entrance into inventory) one, and define the way changeover costs for each attribute are summed to give a global changeover cost. The same way, he (or she) defines costs that depend on time: additional costs for every transition and over-costs for every production.

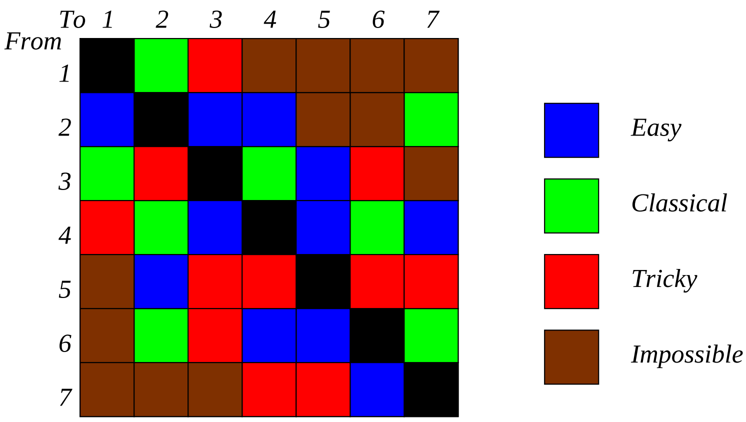


Figure 4.21: Example of transition cost matrix between various thickness values based on four cost types

Finally, the program tends to minimize what is defined as the objective function. Table (4.1) illustrates a real-life case in which the planner simulated his own plan before running three times PLANEO: firstly, it only minimizes production costs; secondly, it takes into account only inventory costs; finally, it minimizes their global sum. He assumed that back-orders were forbidden.

Objective function	Production costs	Inventory costs	**Global Sum**	*Hand-made plan*
Changeover costs	9.4	25.0	9.4	9.8
Additional costs	0.0	4.0	0.0	0.0
Over-costs	0.0	25.2	0.0	9.6
Inventory costs	43.0	38.4	41.1	49.4
Handling costs	15.0	14.3	14.8	17.0
Global cost	68.6	107.0	**65.2**	*85.8*

Table 4.1: Variation of the objective function on a real-life case

4.5.3 Complexity of the problem

Based on the scientific literature on the general lot sizing and scheduling problem with sequence-dependent set-up times and costs, we provide a mixed integer program that allows us to capture originally classical hypotheses while being for our industrial application solvable by on-hand commercial softwares (CPLEX, see [ILOa]).

At first sight, we tried to solve the global problem based on the MILP model proposed by the authors in [SSW$^+$97]. The authors introduce globally the following boolean variables:

- y_i^t for the production of product i over a time horizon of N time periods t,
- $v_{i_1 i_2}^t$ and $w_{i_1 i_2}^t$ for changeovers times and costs between products i_1 and i_2.

In spite of a long work on different parameters of the optimization process, the best on-hand commercial code [ILOa] was unable (with a time limit on the CPU time of several hours) to solve real-life data sets made of a hundred products with $N = 100$ times periods. At this time we defined a product as a thickness (n_1 values), a width (n_2 values) and a quality (n_3 values). There were thus $P = n_1 \times n_2 \times n_3$ products.

Reasonable computation times were obtained by decreasing the number of integer variables of the model. First of all, we proposed an original factorization of changeover times and costs which was inspired by practical observations of real-life data. Secondly, we simplified the modelling of changeover time in the model by using only variables v^t. These simplifications were sufficient enough to obtain integer solutions after several hours of computation, but the optimal one was never obtained after a reasonable time limit. We emphasize that the interest of the factorization depends on the studied case: the less different values involved in the changeover matrices, the higher the model simplification.

Last but not least, we introduced a relevant product-driven decomposition allowing us to simplify the production planning problem into a smaller problem. The introduction of attributes (for instance for thickness, width and quality) -and later of subattribute (for quality which was finally considered as a subattribute)- and the interesting idea of individual discrete time scale for each attribute allowed us to solve most of the real-life problems we were challenged on. Basically, even if we forget the changeover factorization and the ν^t simplification and we take similar time scales for every attribute, we obtain with the three previous attributes (P beeing the product of n_1, n_2 and n_3):

- $(2 \times (n_1^2 + n_2^2 + n_3^2) + (n_1 + n_2 + n_3) + P) \times N$ boolean variables with our model. The factor P comes from the additional Z_i^t boolean variables (see section (§4.4.2)).

- $(2 \times P^2 + P) \times N$ boolean variables in the initial model.

It appears that the more numerous attributes (and the more different values each attribute takes), the more efficient our decomposition. Despite our modeling tricks, we will see in the forthcoming section (§4.5.4) that sometimes we were unable to solve quickly the problems and we add to increase the time period sizes, leading to less precise but easier models. Table (4.2) has been realized on cases based on a 3-attribute structure (thickness, width and quality). Nowadays, most of the plant planners consider that quality is a sub-attribute: it simplifies thus the problem resolution by deleting binary variables.

Finally in the hierarchical approach, the more numerous levels we consider, the easier the model at each level. Naturally, it is important to keep in mind that this approach is justified only if such a simplification makes sense and does not give local optimal solutions far from the global one.

4.5.4 Gain associated with PLANEO

In order to evaluate the gain of the PLANEO tool, we gave it to the production planner of a plant in France. He kept on working manually, whereas he ran it for weeks in order to evaluate the potential gain associated with the optimization model. Table (4.2) summarizes the results he gave us few months later, by specifying whether each plan satisfies all the constraints and its associated cost.

Case	Hand-made plan		PLANEO Solution Cost	**GAIN**
	Cost	satisfies all Constraints		
1	81.1	no	UNFEASIBLE	X
2	43.6	yes	41.5	4.8 %
3	57.7	no	31.8	44.9 %
4	70.2	no	UNFEASIBLE	X
5	36.9	no	36.7	0.5 %
6	88.3	yes	71.2	19.4 %
7	81.0	yes	85.1	- 5.1 %
8	54.7	yes	41.3	24.5 %
9	64.4	yes	46.8	27.3 %
10	76.9	no	81.0	- 5.3 %
11	89.1	no	UNFEASIBLE	X

Table 4.2: Evaluation of the PLANEO gain on real-life cases

From these results and from the global collaboration around this project, we may find that two main cases happen. On the one hand, it happens that the optimized plan is pretty closed to the hand-made one, gain being quasi null or even negative (due to the discrete time in the optimization model which is continuous in the simulation part, see (§4.5.3)). On the other hand, PLANEO may propose a very different plan from usual ones (each production planner has some personal habits, according to his experience), for which the gain is pretty important. In most cases, PLANEO is able to compute a plan which satisfies all the constraints: it is thus an excellent tool to determine an initial plan that may be slightly modified manually. Sometimes, PLANEO identifies a problem without solution. That means that some constraints have to be relaxed. For instance, back-orders must be authorized and minimized.

First of all, our surprise lies in the fact that this optimization project has been first of all a knowledge management one. It was indeed the first time that people from various functions of a plant had to work together on the production scheduling problem. Thus, all our model has been designed according to the expert knowledge of the the industrial process: for the first time, people had to write down changeover times and costs, to explain what is a cost or a constraint on the process, to explicit minimal and maximal campaign duration, etc. This work has revealed that former constraints could be transformed into new costs (for instance, additional costs have replaced interdiction of a changeover during some time periods), opening new way of production planning. This benefit takes a greater impact if we consider that it is now possible to compare parameters from various plants: a new dynamic management has been possible based on these data.

Secondly, the simulation part of the PLANEO software plays a crucial role in the interpretation of results. It highlights the difficulty for the planner to determine manually a production plan that satisfies all the constraints of the model. For instance, it appeared commonly to find that a few products were on shortage situation for a few days. This part was at the beginning less important for us than the optimization one. Finally, it is absolutely necessary to mix both of them. Nowadays, it is common for production planners to use PLANEO as a first step to generate one or several initial production plans, before changing it manually using the simulation tool to take into account exceptional events that we did not capture, or essentially to relax the implicit constraint underlying our model: time is not considered as continuous.

Finally, for cases in which human results and PLANEO plans are comparable to each other, we obtain a mean gain around 13.8% on the global cost of a plan.

From this first experience on our prototype of PLANEO, we identified new outlooks, such as managing raw materials and equipments. The same way PLANEO is used to communicate on the forthcoming production to sales teams in order to help them giving right delivery dates to customers, it could be used to organize the management of trestles and inventory equipments. An engineer has been hired by Saint-Gobain Glass to keep on working on it, implement an industrial version of the software, connect it to the present ERP to import automatically demands and inventory levels and install it to volunteer plants. The bottom-up project is thus a great success.

4.6 Conclusion and research outlooks

We have developed a general method to model a single-stage continuous process planning. Based on a product-driven decomposition into several characteristics, we propose a generic model that may capture both tactical and operational decisions.

Depending on the time horizon and the time period we define, each characteristic of the production may be viewed as either an attribute taking one value by time period or a sub-attribute taking several values per time period.

Based on the literature on the general lot sizing and scheduling problem with sequence-dependent set-up times and costs, we provide a mixed integer program that allows us to capture originally classical hypotheses while being for our industrial application solvable by on-hand commercial softwares (CPLEX, see [ILOa]). Reasonable computation times were obtained by decreasing the number of integer variables of the model. First of all, an original factorization of changeover times and costs was inspired by practical observations of real-life data. Secondly, we simplified the modelling of changeover time in the model. Last but not least, we introduced a relevant product-driven decomposition allowing us to simplify the production planning problem into a much smaller problem by using various attributes with individual adapted time scales.

What is remarkable is that we may use the same optimization model at several levels of a hierarchical planning approach. Depending on the level, we just use various options of the model: the choice of included costs is of course critical. From the hierarchical planning point of view, the more levels we consider, the easier the model at each level. It is thus important to create as many levels as reasonable: this approach is justified only if such a simplification makes sense and does not gives local optimal solutions far from the global one.

We applied it successfully to the float glass manufacturing industry, for which we developed a software, PLANEO, aiming at scheduling on the short-term the campaigns of thickness and width values inside a given colour campaign.

This collaboration led to very encouraging results, not only from an economical point of view (we identified a potential important gain) but also for qualitative consequences, such as knowledge management, inter-function collaboration fostering, etc.

In the forthcoming chapters we apply this work to other jobs of Saint-Gobain Glass (chapter 5), before integrating this generic production model as a building block into a more general one, ROADEO, presented in chapter 6.

Chapter 5

Modeling transformation lines: the coater case

5.1 Capturing transformation lines in our generic production model

In our introduction to the industrial context (§1.2 and §1.3) we explained that Saint-Gobain Glass is the European leader of the glass industry, producing various products through different processes. Figure (1.1) represents product flows between different jobs.

Chapter 4 introduced an original production modeling framework that has a great particularity in our research: we apply it at both the operational scheduling level and the tactical planning one. Based on the decomposition of products into characteristics, we have developed and factorized existing models capturing sequence dependent set-up times and costs to be able to tackle practical issues we have faced in the float glass industry.

Given a planning decision level, we define a method in which meaningful product characteristics are divided into attributes and sub-attributes, corresponding to big and small time buckets. We applied it to different decision levels of production planning in the float glass industry.

Using the adaptability of our production planning model, we apply it in this chapter to other jobs of Saint-Gobain Glass (cf. Figure (1.1): laminated glass, tempered glass and soft-coated glass), which are transformations of float glass.

On the one hand, it appeared to us that laminated and tempered glass were produced on easy-to-model lines. These lines are indeed relatively flexible: we may stop whenever we want and there is no changeover costs within different production batches. Thus, we may capture them by introducing one unique attribute[a] and one unique sub-attribute[a]. The attribute would be the state: it would take two values, running or not, between which we may specify changeover costs due to the labor force required for starting and stopping the lines. The sub-attribute would be the family of transformed products, and it would take as many values as there are at a given level of product aggregation. The result of the production planning is thus the quantity of each product family which is transformed during each time period of the model. Of course, we take into account the production capacities of lines.

On the other hand, coating lines[b] were less easy to model. Basically, coating lines are made of metallic cathodes that are used on-line to sputter nanometric metallic coats on flat glass sheets. Before optimizing the production planning of coating lines, managers needed to have a decision-support tool to configure the on-line cathode sequence, so-called the set-up of the line: the notion of set-up is in this chapter called design, in order not to confuse with the traditional notion of set-up in batch production.

Once the line is configured with a given design, we may produce a set of transformations, so-called the portfolio of transformations of this design. Changeovers between designs are time-consuming and thus represent opportunity costs. How-

[a] notion defined in chapter 4
[b] in this chapter we use the expression "coating lines" as a simplification for "soft-coating lines"

ever, due to high raw material costs, the most important thing to use efficiently a coating line is to maximize the consumption of the metallic cathodes.

As a conclusion, we underline here an interesting design problem we identified as a prerequisite for applying our generic production model to the coating lines. Once different designs are settled, we will see in (§5.6.1) that it becomes indeed easy to capture the line using the notions of attribute and sub-attribute. Finally, chapters 4 and 5 give us a method for modeling all production jobs of Saint-Gobain Glass, allowing us to integrate production tools in our final model introduced in chapter 6.

5.2 The problem of coating line design

We have easily captured every glass transformation job with our production model developed in chapter 4, except sputtering lines. We have indeed discovered an important problem that production manager face: this chapter aims at presenting a solution approach (based on an original modeling) in order to tackle this problem. At the end of this chapter, we will see in (§5.6.1) how we may capture this process by our generic production planning model in order to plan or to schedule it.

Given a data set of demand forecasts, we aim at building an optimization model that helps the production manager to design his production line. We will see in (§5.2.1) what it does mean.

5.2.1 Data

Based on deterministic future demand forecasts for a given portfolio of coated products, we may classify them by coating transformations, independently of the exact product itself. Each transformation is defined by an ordered stack of thin metallic coats whose thickness is imposed. For instance, Table (5.1) presents the definition of a virtual transformation, made of successive ordered sputtering of three different metals $(\mathfrak{m}_1, \mathfrak{m}_2, \mathfrak{m}_3)$.

Sputtering order	Metal	Thickness
1	m_1	e_1
2	m_2	e_2
3	m_3	e_1
4	m_2	e_3
5	m_3	e_1
6	m_1	e_2

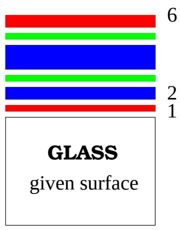


Table 5.1: Definition and Illustration of a transformation, with $e_1 < e_2 < e_3$

For each transformation, we know the forecast surface to coat with the set-up we would like to design. **The design of a coating line lies in defining position and orders of metallic cathodes on the line**. For instance, a line may potentially have 50 cathodes on line. For each one, we have the choice within a set of on-hand cathodes. Each cathode is characterized by a given metal and its volume. We may have different types of cathode for each metal. Figures (5.1) and (5.2) illustrates the general problem of designing the coating line.

We emphasize on Figure (5.2) that a coat (in this illustration the yellow one) may be sputtered by several cathodes made of the corresponding metal. When it is the case, the position of used cathodes may be strictly successive or not along the line. In this latter case (as on the figure), other cathodes between them can not be used simultaneously. We denote these two cases in the following by the expressions of “successive” and “non-successive” used cathodes.

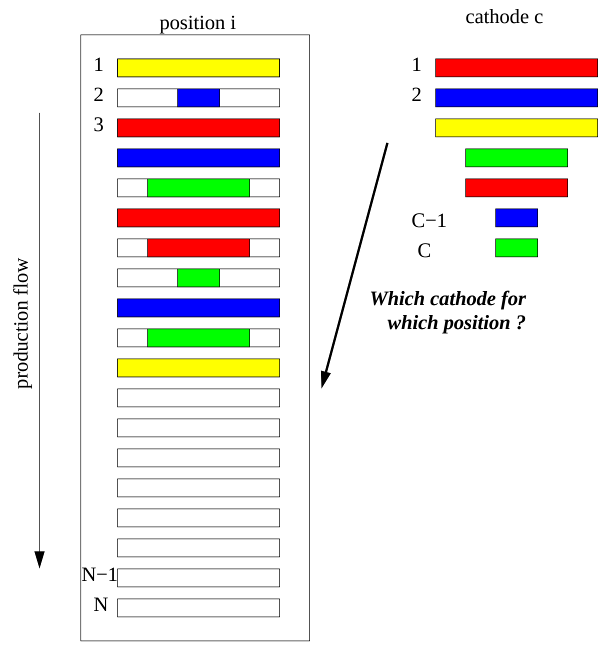


Figure 5.1: Design of the sputtering line

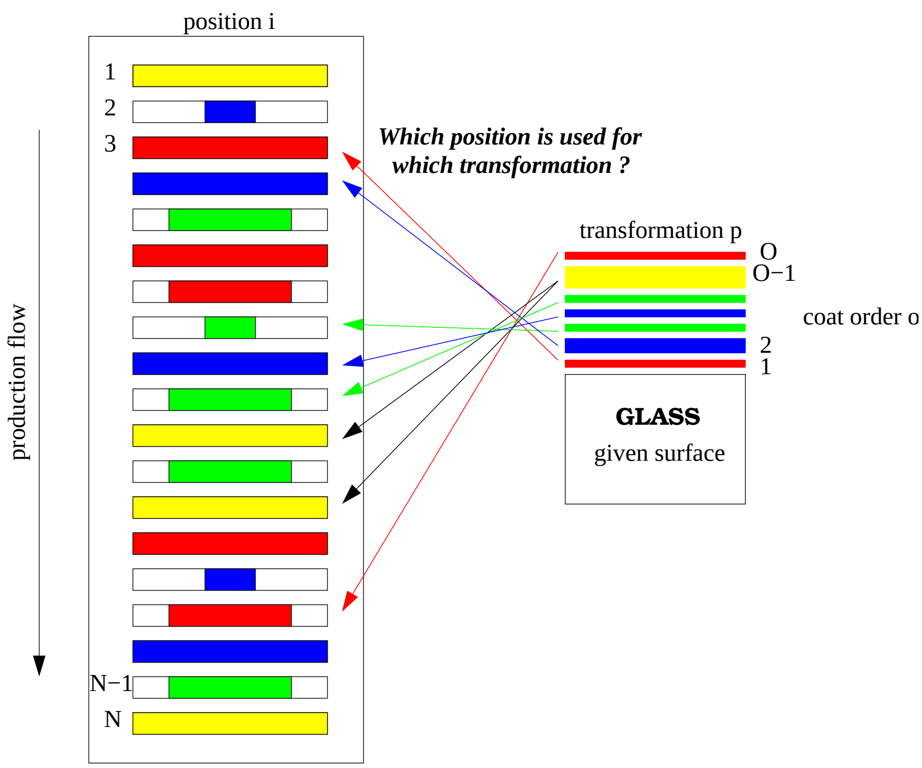


Figure 5.2: Impact of the design on the sputtering process

5.2.2 Assumptions on operating rules

We may define various machine running assumptions, depending on operating rules (and associated costs) of the sputtering line. K being the number of cathodes used in order to sputter a given coat on the glass, Table (5.2) defines four hypotheses.

Basically, hypothesis 3 (resp. 4) comes from hypothesis 1 (resp. 2) and corresponds to a cost going to infinity. Thus, it appears that these hypotheses become more and more general following the order $3, 4, 1, 2$. This means that corresponding optimal solutions on identical data and parameters set will be be ordered the same way: the more restricted the hypothesis, the more expensive the optimal solution.

Operating Rules		a coat is sputtered by K	
Hypothesis	Model description	successive cathodes	non-successive cathodes
1	(§5.3.2)	costs for $K > 1$	
2	(§C.1)	free $\forall K$	costs for $K > 1$
3	(§C.2)	$n = 1$ only	
4	(§C.2)	free $\forall K$	forbidden

Table 5.2: Definition of four assumptions on operating rules of the line

We have divided our chapter into three main parts. First of all, we present in part (§5.3) exact optimization models for every operations rule type. We then describe in (§5.4) heuristic methods that are useful whether exact ones are too hard-to-solve. Finally, we present in part (§5.5) results and examples of our work.

5.3 Exact optimization models

In part (§5.3.2) we present the more general model that allows us to solve exactly problems satisfying hypothesis 1. We see in part (§C.1) how we may simplify it if we assume hypothesis 2, in which we penalize only a repartition of a coat being sputtered by non-successive cathodes. To be exhaustive, we detail in part (§C.2) how we may forbid some operations to solve exactly hypotheses 3 and 4 based on models of hypotheses 1 and 2.

5.3.1 Notation

We consider the set of M metals $\mathcal{M}$, indexed by letter $m > 0$. Each one is characterized by a volumic cost, c_m (in €/ m^3).

We denote transformations by index p and its number of coat by O_p. The order of each coat is denoted by index $o \in [1, O_p]$. We denote $m_{po} \in \mathcal{M}$ and e_{po} the corresponding metal and thickness. We know that we must produce a total surface of S_p in each transformation p. According to our hypothesis, each coat may be sputtered by several cathodes (with identical metal). Nevertheless, we take into account a potentially useful maximal number of cathodes N_{po} used to do the o^{th} coat of transformation p. We notice that the quantity $e_{po} \times S_p$ is nothing but the total volume of needed metal m_{po} for the o^{th} coats of transformation p. We denote it v_{po}. By convention, we use the double sum symbol $\sum_p \sum_o$ to simplify the exact notation $\sum_p \sum_{o \in O_p}$.

We denote potential positions of cathodes by index i and N the maximal number of cathodes set on the line. We order the set of cathodes according to the orientation of production flow. The cathode $i+1$ is located just after the i^{th} one regarding to the production sense.

Finally, we denote the C cathodes by index c. Each cathode is characterized by a unique metal $m_c \in \mathcal{M}$ and a global volume V_c. By convention and in order to authorize free positions on the line, we introduce a virtual cathode and denote it $c = 0$. By convention, the corresponding metal is $m_0 = 0$ and its costs $c_0 = 0$. We denote $\mathcal{C}(m)$ the subset of cathodes c whose metal is m, and $\bar{\mathcal{C}}(m)$ its complementary subset. We assume that we may use at most N_c cathodes of type c. To match the production process, we consider the yield of sputtering operations constant. We denote it $\phi < 1$.

Depending on the context of the optimization, there are clearly several possible objectives that we may aim to minimize. First of all, we may also try to minimize partially consumed cathodes at the end of the production run. Assuming that some cathodes are not completely burnt would indeed impact the following run by imposing it a set of initial reduced capacity cathodes. In some cases production managers would even decide to drop the cathode definitively, leading to a direct loss cost. We may also try to minimize the number of required positions on the line to fulfill specified demand. In this way, successive optimizations and corresponding evolutions of the demand portfolio may be helpful for production managers: it may serve to use plainly the production line by minimizing global set-up times between configurations. Finally, given that we authorize a coat to be sputtered by several successive cathodes, we may penalize this splitting and try to minimize it.

Thus, we introduce three main objective coefficients, allowing the user to create the best objective function for each situation he would face. We denote them β_1, β_2 and β_3. Naturally, we may introduce as many weighted linear objective functions as necessary.

5.3.2 Exact solution of problems assuming hypothesis 1

In this paragraph we present the model capturing our industrial problem assuming that we satisfy operations rule of hypothesis 1. We solve it using a Mixed Linear Programming solver. We compare an excellent commercial one, CPLEX (see [ILOa]) and a free one, GLPK (see [GNU]).

Variables

We introduce the following variables :

- Some binary variables :
  - z_c^i equals one if and only if the cathode c is located in position i. We notice that $z_0^i = 1$ means that there is no cathode on position i.
  - y_{po}^i equals one whether the o^{th} coat of transformation p uses the cathode in position i.

- Some integer variables :
  - $r_{po}^{M} \in [1, N]$ equals the highest position i on which a cathode is used for sputtering the o^{th} coat of transformation p.
  - $r_{po}^{m} \in [1, N]$ equals the lowest position i on which a cathode is used for sputtering the o^{th} coat of transformation p.
- Some continuous variables :
  - $x_{po}^{i} \in [0, 1]$ equals the volume proportion of metal of the cathode in position i which is sputtered for the o^{th} coat of transformation p.
  - to capture the utilization level of cathodes, we introduce:
    * either ρ^{i} which represents the remaining volume of metal on the i^{th} position.
    * or γ^{i} which represents the cost (in €) associated to the unused metal in position i.

Model

$$Obj_1 = \begin{cases} \sum_i \rho^i \\ \sum_i \gamma^i \end{cases} \tag{5.1}$$

$$Obj_2 = \sum_i \sum_{c>0} z_c^i \tag{5.2}$$

$$Obj_3 = \sum_i \sum_p \sum_o y_{po}^i \tag{5.3}$$

$$Min \sum_k \beta_k \times Obj_k \tag{5.4}$$

$$\forall i, \quad \sum_{c=0}^{C} z_c^i = 1 \tag{5.5}$$

$$\forall i, \forall p, \forall o, \quad x_{po}^i \leq y_{po}^i \tag{5.6}$$

$$\forall i, \forall p, \forall o, \quad y_{po}^i \leq 1 - \sum_{c \in (\bar{\mathcal{C}}(m_{po}))} z_c^i \tag{5.7}$$

$$\forall i, \forall p, \forall o, \quad y_{po}^i \times i \leq r_{po}^M \tag{5.8}$$

$$\forall i, \forall p, \forall o, \quad r_{po}^m \leq y_{po}^i \times i + (1 - y_{po}^i) \times N \tag{5.9}$$

$$\forall p, \forall o, \quad r_{p,o}^m \leq r_{p,o}^M \tag{5.10}$$

$$\forall p, \forall o \in [1, 0_p - 1], \quad r_{p,o}^M < r_{p,o+1}^m \tag{5.11}$$

$$\forall p, \forall o, \quad \sum_i x_{po}^i = 1 \tag{5.12}$$

$$\forall p, \forall o, \quad \sum_i y_{po}^i \leq N_{po} \tag{5.13}$$

$$\forall c, \quad \sum_i z_c^i \leq N_c \tag{5.14}$$

$$\forall i \in [1, n-1], \quad z_0^i \leq z_0^{i+1} \tag{5.15}$$

$$\forall i, \quad \rho^i = \sum_c z_c^i \times V_c - \sum_p \sum_o x_{po}^i \times \frac{\nu_{po}}{\phi} \tag{5.16}$$

$$\forall i, \quad \gamma^i = \sum_c z_c^i \times c_{m_c} \times V_c - \sum_p \sum_o x_{po}^i \times c_{m_{po}} \times \frac{\nu_{po}}{\phi} \tag{5.17}$$

The objective function (5.1) represents the cost of lost residual metal on cathodes at the end of the production run, whereas the objective (5.2) penalizes the number of required positions on the line and objective (5.3) penalizes the splitting of a given coat on several cathodes. Finally, the global objective function (5.4) is a linear combinaison of these three sub-objectives.

Equalities (5.5) forces that a position may host at most one cathode. In cases in which $z_0^i = 1$, the position is not used.

The inequalities (5.6) links the continuous variables x_{po}^i to the integer ones y_{po}^i : a volume may be used if and only if the connexion is open. The same way (5.7) enforces that a connexion between a transformation and a position is open if and only if the metal is compatible.

Inequalities (5.8) and (5.9) ensure that r_{po}^{m} (resp. r_{po}^{M}) is lower (resp. higher) than the the lowest (resp. highest) position of used position to sputter the o^{th} coat of transformation p, while (5.10) and (5.11) make that the sputtering operations tend to satisfy the ordered definition of each transformation.

Finally, equalities (5.12) mean that cathodes are losing volume through production up to the sputtered volume, including the operation yield. (5.16) state that global initial volume of each cathode becomes either a sputtered coat or remains on it. Depending on the data, we penalize unused metal either by volume (with variables ρ_i) or by cost (variables γ_i). In the second case, (5.17) defines variables γ_i.

(5.13) and (5.14) are inequalities corresponding to upper bounds forced by the user: first ones constrain the number of cathodes working on the same operation while second ones constrain the on-hand cathodes quantity. The last inequalities (5.15) may be used in cases in which we want to keep empty positions at the end of the line.

A tricky way of solving it

It appears on highly combinatorial problems that it may be easier to prove its infeasibility (due to an insufficient number N of on-hand positions) than to solve it optimally with a larger number N. Thus, to determine the smallest number of useful positions in this step, it may be quicker to apply the following procedure than to introduce virtual cathodes ($c = 0$ by convention): We denote $\mathcal{P}(n)$ the problem with n potential positions and no virtual cathodes. This problem is made of the previous one except that we do not minimize Obj_2 (defined by (5.2)) because we know explicitly that:

$$\sum_{i} \sum_{m} z_{m}^{i} = n \tag{5.18}$$

$\mathcal{P}(n)$ may have either one optimal solution s_n or no solution at all.

We initialize $n = M$ (the number of different used metals). While $\mathcal{P}(n)$ is an infeasible problem, we add a position $n = n + 1$. As soon as the problem is feasible, we define n_c^* as the smallest feasible line size. To stop the procedure, we define two parameters: firstly, we may add a maximal solution set size L. Secondly, we define a maximal gap g to n_c^* and we stop the procedure as soon as $n > n_c^* + g$. The parameter g being a tolerance, we do not always obtain the exact optimal solution. We notice that in cases in which we only want to minimize Obj_2, we obtain the exact optimal solution with $g = 0$. Table (5.3) compares CPU times of a given case for two solvers: Cplex (see [ILOa]) (resp. GLPK (see [GNU])) is the best available commercial software (resp. freeware).

Model	CPLEX	GLPK
Original model	46 s	?>> 3600 s
Procedure with $g = 2$ and $L = \infty$	6 s	393 s

Table 5.3: Impact of the procedure on CPU times

In this example the optimum is obtained with $n_c^* + 1$ cathodes, illustrating that g is an important parameter to reach optimality. It appears clearly that this procedure is efficient. As a consequence, we use it in the following. However, this exact model seems to be hard-to-solve as soon as the data set becomes realistic. Thus, we have developed some heuristic models, that we introduce in section (§5.4).

Similar developments are exposed in Appendix C: section (§C.1) presents the exact model under assumption 2 and section (§C.2) introduces solutions under assumptions 3 and 4.

As a conclusion, we are able to model exactly every defined hypothesis. Let us now focus on a particular model which may be useful in part (§5.4): given a line design, what is the cost of its corresponding optimal use under each hypothesis?

5.3.3 Retrieving utilization of the line for a given design

Let us now answer a basic question: what is the optimal utilization of the line for a given design? Which transformation coat is sputtered by which cathode? We first solve this problem under hypothesis 1, before explaining how to transpose the model to every hypothesis.

Under hypothesis 1

We have an ordered set of defined positions $\mathcal{I}$. Each position corresponds to a cathode, and thus position i has a given metal m_i and volume V_i. We introduce the compatibility function between positions and transformation's coat:

$$\mathcal{C}(i,p,o) = \begin{cases} 1 & \text{if } m_i = m_{po}; \\ 0 & \text{if } m_i \neq m_{po}. \end{cases}$$

We use similar notation to the one in part (§5.3.2), except that variables z_c^i and ρ_i are useless.

$$\mathrm{Obj}_1 = \begin{cases} \sum_i \rho^i \\ \sum_i \gamma^i \end{cases} \tag{5.19}$$

$$\mathrm{Obj}_3 \quad = \quad \sum_i \sum_p \sum_o y_{po}^i \tag{5.20}$$

$$\mathrm{Min} \sum_k \beta_k \times \mathrm{Obj}_k \tag{5.21}$$

$$\forall i,\ \forall p,\ \forall o, \quad x^i_{po} \leq y^i_{po} \tag{5.22}$$

$$\forall i,\ \forall p,\ \forall o, \quad y^i_{po} \leq \mathcal{C}(i, p, o) \tag{5.23}$$

$$\forall i,\ \forall p,\ \forall o, \quad y^i_{po} \times i \leq r^M_{po} \tag{5.24}$$

$$\forall i,\ \forall p,\ \forall o, \quad r^m_{po} \leq y^i_{po} \times i + (1 - y^i_{po}) \times N \tag{5.25}$$

$$\forall p,\ \forall o, \quad r^m_{p,o} \leq r^M_{p,o} \tag{5.26}$$

$$\forall p,\ \forall o \in [1, O_p - 1], \quad r^M_{p,o} < r^m_{p,o+1} \tag{5.27}$$

$$\forall p,\ \forall o, \quad \sum_i x^i_{po} = 1 \tag{5.28}$$

$$\forall p,\ \forall o, \quad \sum_i y^i_{po} \leq N_{po} \tag{5.29}$$

$$\forall i, \quad \rho^i = V_i - \sum_p \sum_o x^i_{po} \times \frac{\nu_{po}}{\phi} \tag{5.30}$$

$$\forall i, \quad \gamma^i = c_{m_i} \times V_i - \sum_p \sum_o x^i_{po} \times c_{m_{po}} \times \frac{\nu_{po}}{\phi} \tag{5.31}$$

Basically, solving this subproblem gives us the way known cathodes are used by transformations.

Under hypotheses 2, 3 and 4

From the model explained in previous part, we may infer easily how to build a slightly different one for solving models under hypothesis 2.

Basically, we just have to use notation explained in part (§C.1), to create known sets $\mathcal{C}_i$ of cathodes of same metal (denoted m_i) whose rank in the metal sequence is i, and to exchange equations (5.30) and (5.31) by following equations (5.32) and (5.33):

$$\forall i, \quad \rho^i = \sum_{c \in \mathcal{C}_i} V_c - \sum_p \sum_o x^i_{po} \times \frac{\nu_{po}}{\phi} \tag{5.32}$$

$$\forall i, \quad \gamma^i = c_{m_i} \times \sum_{c \in \mathcal{C}_i} V_c - \sum_p \sum_o x^i_{po} \times c_{m_{po}} \times \frac{\nu_{po}}{\phi} \tag{5.33}$$

The same way, according to the methodology described in part (§C.2), we infer models for both hypotheses 3 and 4 from those for hypotheses 1 and 2 by exchanging continuous variables X_{ipo} by integer ones Y_{ipo}.

5.3.4 Conclusion

In this part, we have developed several exact optimization models based on linear programming theory. Despite some tricks in the way we use them, it may be too time-consuming to use them on real data sets. Thus, we introduce in the next part some heuristic methods that may give non-optimal solutions more quickly.

5.4 Heuristic methods

In part (§5.4.1) we provide a three-step decomposition that provides us a feasible solution. It determines first the metal sequence (metal nature and volume) and then the cathode affectation.

It appears that the best idea we have tried has been to use then a Simulated Annealing procedure to improve this initial solution. In part (§5.4.2) we explain how to use it based on meaningful elementary movements: cathodes' exchanges in the configuration.

5.4.1 Finding an initial solution using a heuristic three-step decomposition

It appears that our main model may be decomposed into three successive subproblems: this trick allows us to solve our problem without optimality but quicker. We write the methodology under hypothesis 1, but easy modifications may fit other ones.

First step: uncapacitated cathodes model

First, let us try to define a model assuming that on-hand cathodes have uncapacitated volumes. We aim at determining a lower bound on the minimal number of cathodes required to produce a transformation portfolio.

We keep the same notation as in paragraph (§5.3.2), except that we add new continuous variables V_i denoting the volume of the cathode used on position i. Of course, it is null whether there is no cathode on it. We also replace z_c^i by z_m^i, denoting the use of a cathode of metal m in position i. We keep the same convention: $z_0^i = 1$ for unused positions. Finally, we do not need to introduce former variables x_{po}^i because each coat is spluttered by a unique cathode whose volume is uncapacitated. The same way, ρ_i are useless because all the metal is used. We have the following model:

$$\mathrm{Min} \sum_i \sum_{m>0} z_m^i \tag{5.34}$$

$$\begin{aligned}
\forall i, \quad & \sum_m z_m^i &=& \; 1 & (5.35)\\
\forall p, \forall o, \quad & \sum_i y_{po}^i &=& \; 1 & (5.36)\\
\forall i, \forall p, \forall o, \quad & y_{po}^i &\leq& \; z_{m_{po}}^i & (5.37)\\
\forall i, \forall p, \forall o, \quad & y_{po}^i &\leq& \; 1 - \sum_{m \neq m_{po}} z_m^i & (5.38)\\
\forall i, \forall p, \forall o, \quad & y_{po}^i \times i &\leq& \; r_{po}^M & (5.39)\\
\forall i, \forall p, \forall o, \quad & r_{po}^m &\leq& \; y_{po}^i \times i + (1 - y_{po}^i) \times N & (5.40)\\
\forall p, \forall o, \quad & r_{p,o}^m &\leq& \; r_{p,o}^M & (5.41)\\
\forall p, \forall o \in [1, O_p - 1], \quad & r_{p,o}^M &<& \; r_{p,o+1}^m & (5.42)\\
\forall i, \quad & V_i &=& \; \sum_p \sum_o y_{po}^i \times \frac{\nu_{po}}{\phi} & (5.43)\\
\forall i \in [1, n-1], \quad & z_0^i &\leq& \; z_0^{i+1} & (5.44)
\end{aligned}$$

Solving this problem gives us the line design whether we would have neither discrete values for cathodes' volume nor limited on-hand cathodes. This relaxation of the main problem makes the computation time of its solving considerably decrease. We will present some results in part (§5.4.1). Let us denote I the number of used positions. For each position i, we compute its maximal division into successive cathodes during the next step:

$$\forall i, \quad N_i = \min_{(p,o) \text{ s.t. } y_{po}^i = 1} (N_{po}) \tag{5.45}$$

The constraint (5.54) will ensure in the second step that the final solution satisfies the maximal number of cathodes N_{po} used to sputter each coat o of each transformation p. If there exist at least one coat (p, o) which is sputtered by the i^{th} position (i.e. $y_{po}^i = 1$) and for which $N_i < N_{po}$, we may be constraining too much the solution set.

Thus, we do not use it in the first run of our three step solution. If the last step is unfeasible, we restart the second one by adding constraints (5.54) in the linear model.

It appears on highly combinatorial problems that it may be easier to prove its infeasibility (due to an insufficient number N of on-hand positions) than to solve it optimally with a larger number N. Thus, to determine the smallest number of useful positions n^* in this step, it may be quicker to apply the following procedure than to introduce virtual cathodes ($m = 0$ by convention): We denote $\mathcal{P}(n)$ the problem with n potential positions and no virtual cathodes ($m > 0$ by convention). This problem is made of the previous one except that we delete constraints (5.44) and in which we know explicitly that:

$$\sum_i \sum_m z_m^i = n \tag{5.46}$$

$\mathcal{P}(n)$ may have several feasible (and thus "optimal" because the objective function is fixed) solutions: we aim to determine all of them. Let us denote $\mathcal{L}$ the set of feasible metal sequences.

We initialize $n = M, \mathcal{I}_n = \emptyset$ (the number of different used metals). While $\mathcal{P}(n)$ is an infeasible problem, we add a position $n = n + 1$. As soon as it exists a feasible solution, we set $n^* = n$. For each feasible metal sequence size, we search every solutions. At each sub-iteration, we add the result to the infeasible sequence set $\mathcal{I}_n$ and to the result set $\mathcal{L}$. This set forces the next solution to be different from the found ones through constraints:

$$\forall S \in \mathcal{I}_n \quad z^1_{m_{S,1}} + z^2_{m_{S,2}} + \cdots + z^n_{m_{S,n}} < n \tag{5.47}$$

At each iteration, $\mathcal{I}_n$ contains one more element until the problem be not infeasible. We thus add a position $n = n + 1$ and reset the set $\mathcal{I}_n = \emptyset$.

This procedure gives us a list of feasible metal sequences. To stop it, we define two parameters: firstly, we may add a maximal solution set size L. Secondly, we define a maximal gap g to n^* and we stop the procedure as soon as $n > n^* + g$.

To understand the impact of our procedure on CPU times, we set $L = 1$ (we search one unique feasible sequence of size n^*). Table (5.4) presents the impact on CPU time of this trick on a simple data set corresponding to the one explained in part (§5.5.1).

Used Model	CPLEX[a]	GLPK[b]
Model with virtual cathodes	0.1 s	495 s
Incrementation of n until the problem is feasible	0.1 s	15 s

Table 5.4: Impact of the trick on the first step model resolution

As a conclusion, we use our incrementation procedure to solve the first step of our model. Thus we obtain a list of feasible metal (position and volume) sequences. We then use other steps to evaluate each of them.

Second step: introducing discrete capacities

Let us now take into account the fact that on-hand cathodes have discrete volumes and are limited. We aim to choose within a given set of cathodes those which are fitting as closely as possible the optimal design (with I used positions) given by the previous step (see part (§5.4.1)). We introduce the given set $\mathcal{C}$ of cathodes.

We introduce the following notation of (using the same logic as in (§5.3.2)): n_c^i is an integer variable indicating how many cathodes c is used for covering the given required volume V_i of metal m_i. ρ_i is the remaining volume on the set of cathodes used for position i. We introduce the compatibility function between positions and transformation's coat:

$$\mathcal{C}(i,c) = \begin{cases} 1 & \text{if } m_i = m_c; \\ 0 & \text{if } m_i \neq m_c. \end{cases}$$

The optimization model is the following:

[a] Cplex is the best available commercial software, see [ILOa]
[b] GLPK is the best available freeware, see [GNU]

$$\mathsf{Obj}_1 = \begin{cases} \sum_i \rho^i \\ \sum_i \gamma^i \end{cases} \tag{5.48}$$

$$\mathsf{Obj}_2 = \sum_i \sum_{c>0} n_c^i \tag{5.49}$$

$$Min \sum_k \beta_k \times \mathsf{Obj}_k \tag{5.50}$$

$$\forall i, \forall c, \quad n_c^i \leq \mathcal{C}(i,c) \times N_c \tag{5.51}$$

$$\forall i, \quad V_i + \rho_i = \sum_c n_c^i \times V_c \tag{5.52}$$

$$\forall c, \quad \sum_i n_c^i \leq N_c \tag{5.53}$$

$$\forall i, \quad \sum_c n_c^i \leq N_i \tag{5.54}$$

$$\sum_i \sum_c n_c^i \leq N \tag{5.55}$$

$$\forall i, \quad \rho^i = \sum_c n_c^i \times V_c - V_i \tag{5.56}$$

$$\forall i, \quad \gamma^i = \sum_c n_c^i \times c_{m_c} \times V_c - c_{m_i} \times V_i \tag{5.57}$$

Solving this subproblem gives us the final design of the line: we know exactly which cathode is used at each position.

Third step: computing optimal utilization

Finally, we use as a third step the model introduced in part (5.3.3) to retrieve the utilization of each cathode and compute the corresponding cost.

To simplify its resolution, we may use the following trick: From the optimal solution of the first step model (model with uncapacitated cathodes presented in (§5.4.1)), we know the global sequence of metals on line, as well as which one is used for each transformation coat. Let us denote by the index j the position of metals in this sequence (one given metal may be in several positions). Thus, we know the value of variables y^j_{po}. Let us introduce the sets J^0_{po} (and resp. J^1_{po}) corresponding to the set of positions j which are not used (resp. are used) for the production of the o^{th} coat of transformation p.

From the solution of the second step model (from (§5.4.1)), we know the set of used positions i (with capacitated cathodes) corresponding to each position j: we denote it S_j. Finally, we can introduce the following constraints in the third step model to speed up its resolution:

$$\forall p,\ \forall o,\ \forall j \in J^0_{po} \qquad \sum_{i \in S_j} y^i_{po} = 0 \tag{5.58}$$

$$\forall p,\ \forall o,\ \forall j \in J^1_{po} \qquad \sum_{i \in S_j} y^i_{po} \geq 1 \tag{5.59}$$

About the three-step decomposition

An accurate analysis of our decomposition highlights that our main error in which we create a huge gap from optimality (due to Obj_1) is to determine volumes based on each position in the metal sequence in the first step whereas remaining volumes are computed in step 2.

However, this heuristic methodology becomes not that bad when we try to solve our general model by minimizing only the number of used cathodes $\{\beta_1 = 0, \beta_2 > 0, \beta_3 = 0\}$. Unfortunately, managers aim mainly to maximize the metal utilization during production runs.

Is it really easy to solve it? Table (5.5) compares on a simple data set different CPU times corresponding to the use of either a commercial solver CPLEX (see

[ILOa]) or a free one GLPK (see [GNU]).

Model	CPLEX	GLPK
First step	0.2 s	70 s
Second step	0.1 s	0.1 s
Third step	0.1 s	1 s

Table 5.5: Impact of the solver on the three-step decomposition CPU times

Thus, it appears that this heuristic is not that good and pretty hard-to-solve! Let us focus on others methods to solve it quickly.

Using the same idea more efficiently

Based on the same three-step decomposition, let us now introduce pure heuristic methods to determine a feasible solution.

First of all, we use a basic procedure to determine a feasible metal sequence whose size is as small as possible. It is easy to understand that adding in a row metal sequences corresponding to every transformation, we create a feasible metal sequence. Based on it, we try to decrease its size while keeping it feasible. To do so, we delete randomly some elements of the metal sequence, before checking whether the result is feasible. In case it is, we simplify it.

To compute required metal volumes, we use the model introduced in (§5.4.1) in which we specify which metal is used in which position.

To transform this feasible metal sequence into a feasible cathode sequence, we compute for each rank of the metal sequence (whose we know the required volume) the minimal number of the biggest on-hand cathodes we need to cover it.

Based on the corresponding feasible cathode sequence, we use a local improvement method based on the following idea: picking up randomly a cathode, we replace it successively either by each other identical metal cathode or by nothing. To compute the cost of a given design, we use one model (depending on the hypothesis we satisfy) introduced in part (§5.3.3). Thus, we are able to compare each solution and to keep the cheapest one.

Applying this methodology makes it possible to determine quickly an initial solution (whose cost is denoted C_0) to start the simulated annealing procedure.

5.4.2 Simulated annealing procedure

A simulated annealing procedure is based on random elementary movements that disturb a given feasible solution in order to improve it. A new feasible solution may be temporarily accepted even if its costs C_c is superior to the so far best one C^* with the probability $e^{\frac{C_c - C^*}{T}}$, T being the so called temperature parameter. This way we try to avoid local optima. We use iteratively several decreasing temperatures, from the departure one T_0 to the freezing one $T_f << T_0$: we use the parameter $\rho < 1$ to make the temperature decrease ($T_{n+1} = T_n \times \rho$). For a given temperature, we try N elementary movements. We keep the current solution whether its cost passes the acceptance test. Before starting the next iteration, we apply to our solution a local improvement meta-heuristic procedure, based on randoms switch of cathodes with either nothing or other compatible ones.

In our case, we use a meaningful elementary movement: we select randomly two different cathodes and exchange them to create a new design. We may authorize or not to change the size of the design. To do so, we include a virtual cathode during the random selection. This way, we may either add or delete one cathode in the sequence. To compute the new solution, we use the model of (§5.3.3) that fits the right hypothesis. Whether the model is unfeasible, the design is unfeasible. Otherwise, we get the cost of the solution corresponding to the optimal utilization of the line to fulfill the demand.

We set the parameters of the procedure as follows: $T_0 = 0.15 \times C_0$, $\rho = 0.9$, $T_f = \frac{T_0}{3}$, $N = 250$. These parameters are pretty aggressive (we do not try plenty of solutions) because each new solution has to be evaluated by a linear model of part (§5.3.3): this evaluation may last several seconds, whereas we would like to limit the duration of this heuristic method.

In part (§5.5), we focus on the results we have obtained on two different data sets: it appears that our heuristic is efficient by providing good solutions in a reasonable computation time.

5.5 Results

In this section we present our first results obtained on imagined data sets. To solve our linear programs, we compare the best existing commercial solver Cplex (sold by ILOG, see ([ILOa])) to the the free GNU project called GLPK (see ([GNU])).

In all our examples we aim to minimize the three objectives described in part (§5.3.1), with the corresponding weights:

- $\beta_1 = 100$, corresponding to the first objective of minimization of remaining volumes on cathodes after production (we could have chosen to minimize the corresponding cost).

- $\beta_2 = 100$, corresponding to second objective which is to use as less positions as possible.

- $\beta_3 = 10$, corresponding to the third objective penalizing the use of several different cathodes for sputtering the same metal coat.

5.5.1 First example on a simple data set

Data

We denote D the reference length dimension (ex: nm or mm) in this problem. We define in Table (5.7) a basic demand portfolio made of three transformations combining four metals: Gold (associated cost 2 €/ D^3), Silver (1 €/ D^3), Titanium (6 €/ D^3) and Platinum (6 €/ D^3). On the other hand, we consider a line with 12 potential positions, a yield $\phi = 0.98$ and a set of on-hand cathodes defined by Table (5.6). We assume here that we have an infinite number of every cathode.

Metal	Volume (in D^3)
Silver	1000
Silver	3000
Titanium	3000
Titanium	1000
Gold	2500
Gold	4500
Silver	5000
Platinum	1000
Platinum	2000

Table 5.6: On-hand cathodes

Transformation			Metallic coats[a]			
Id p	Name	Surface[c]	Order o	Metal	Thickness[b]	Max Division
1	Planitherm	100	1	Silver	2.1	2
			2	Gold	4	3
			3	Titanium	1	5
			4	Silver	1	4
2	Planistar	200	1	Silver	2.1	3
			2	Gold	4	6
			3	Titanium	1	3
			4	Silver	1	2
			5	Gold	3	1
			6	Titanium	2	4
			7	Silver	4	6
3	PlaniNew	1000	1	Gold	4	2
			2	Platinum	1	4
			3	Titanium	1	6

Table 5.7: Simple data set

[a] D being the reference dimension
[b] in D
[c] in D^2

Results

$\mathtt{Obj}_1$	Hypothesis	Method			
		Exact Optimization		Heuristic Optimization	
		CPLEX	GLPK	CPLEX	GLPK
Volume	hyp1	407 182 (15s)	CPU > 3600s	457 182 (157s)	457 182 (1142s)
	hyp2	407 172 (14s)	CPU > 3600s	457 172 (145s)	557 072 (751s)
	hyp3	807 162 (7s)	CPU > 3600s	807 162 (129s)	907 262 (1092s)
	hyp4	607 162 (86s)	CPU > 3600s	607 362 (103s)	707 262 (477s)
Cost	hyp1	991 876 (15s)	CPU > 3600s	1 491 880 (144s)	?
	hyp2	991 866 (10s)	CPU > 3600s	1 491 870 (156s)	?
	hyp3	2 591 860 (5s)	CPU > 3600s	2 591 860 (162s)	?
	hyp4	1 591 960 (95s)	CPU > 3600s	1 591 960 (123s)	?

Table 5.8: Comparison of results: Exact *versus* Heuristic Optimizations, CPLEX *versus* GLPK.

Table (5.8) compares results of exact optimization versus heuristic optimization, and for each one gives the CPU time of either the best available commercial solver CPLEX (see [?]) or a free solver GLPK (see [GNU]). We notice that our remark about hypotheses (see part (§5.2.2)) is obvious looking at exact solutions: the more restrictive the hypothesis (rank is $2, 1, 4, 3$), the more expensive the optimum solution.

It appears that exact optimization is still possible on simple data set using CPLEX, whereas GLPK is not able to solve the problem. On the other hand, we notice that our heuristic is on average not too far from the optimal solution, specially if we do not take into account some exceptions (for which we recommend to run again the computation…). Even if the computation time appears to be longer (using CPLEX) on a heuristic method than on the exact one, this effect is due to the small size of the data set: we will see in part (§5.5.2) that exact methods become really hard to solve on larger problems.

To illustrate the industrial problem, we provide the optimal line in Table (5.9) corresponding to the problem with $\mathtt{Obj}_1$ expressed in remaining volume and assuming hypothesis 1. Table (5.10) gives the correspondence between the line and the production. We can check that the sequence of sputtering is satisfied. We may notice that the first coat of third transformation $(p = 3, o = 1)$ is sputtered by two cathodes.

Position	Metal	Initial Volume	Remaining volume
1	Silver	1000	357
2	Gold	2500	0
3	Gold	2500	102
4	Platinum	2000	980
5	Titanium	1000	265
6	Silver	1000	796
7	Gold	2500	1480
8	Titanium	1000	0
9	Silver	1000	82

Table 5.9: Exact optimal line for hypothesis 1, Obj_1 in volume.

Line			Volume sputtered to create the $\mathtt{o}^{\mathtt{th}}$ coat of the transformation $\mathtt{p}$: $\{\mathtt{p},\mathtt{o}\}$													
Position	Metal	Initial volume	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	2,5	2,6	2,7	3,1	3,2	3,3
1	Silver	1000	214	0	0	0	429	0	0	0	0	0	0	0	0	0
2	Gold	2500	0	0	0	0	0	0	0	0	0	0	0	2500	0	0
3	Gold	2500	0	0	0	0	0	816	0	0	0	0	0	1582	0	0
4	Platinum	2000	0	0	0	0	0	0	0	0	0	0	0	0	1020	0
5	Titanium	1000	0	0	0	0	0	0	204	0	0	0	0	0	0	531
6	Silver	1000	0	0	0	0	0	0	0	204	0	0	0	0	0	0
7	Gold	2500	0	408	0	0	0	0	0	0	612	0	0	0	0	0
8	Titanium	1000	0	0	102	0	0	0	0	0	0	408	0	0	0	490
9	Silver	1000	0	0	0	102	0	0	0	0	0	0	816	0	0	0

Table 5.10: Correspondence between the line and the production: optimal utilization.

5.5.2 Second example on a realistic data set

Let us now focus on a realistic data set that the curious reader may found in Appendix (§C.3). This time, we did not succeed in solving it in a reasonable CPU time. However, our heuristic using the CPLEX solver gives good results pretty quickly. We define five transformations (made of on average 10 coats) made of five different metals: Silver (1 €/ D^3), Gold (2 €/ D^3), Steel (0.5 €/ D^3), Platinum (4 €/ D^3) and Titanium (6 €/ D^3). We have a line which may support with at most 60 cathodes, has a yield $\phi = 0.95$, and a set of on-hand cathodes described in Table (C.2) of Appendix (C.3).

To be realistic, we minimize $\mathtt{Obj}_1$ in cost of remaining metal. We use the heuristic with the CPLEX commercial solver and we compare the results of different hypotheses. Hypothesis 3 is impossible because we forbid to sputter one coat by several cathodes and there is at least one coat that requires a nonexistent cathode. To relax this constraint, we add a virtual cathode of volume 10 000 for each metal.

Hypothesis	Best Found Cost (in €)	CPU time (in s)
1	932 775	543
2	932 845	660
3	IMPOSSIBLE *(relaxed : 5 843 830)*	IMPOSSIBLE *(relaxed : 163s)*
4	1 120 430	321

Table 5.11: Performance of our heuristic combining local search and simulated annealing with Cplex on a realistic case: comparison of different hypothesis results.

Finally, it appears to us that simulated annealing is time-consuming and not that useful. We have decided to improve our local search meta-heuristic methods (defined in Appendix (C.4)) and not to use simulated annealing. Table (5.12) summarizes our new results.

Hypothesis	Best Found Cost (in €)	CPU time (in s)
1	545 485	105
2	747 745	75
3	IMPOSSIBLE *(relaxed : 9 118 930)*	IMPOSSIBLE *(relaxed : 250s)*
4	805 325	43

Table 5.12: Performance of our heuristic combining local search and simulated annealing with Cplex on a realistic case: comparison of different hypothesis results.

5.6 Conclusion and perspectives

5.6.1 Extension to our production model

As emphasized in part (§5.1), the motivation of this chapter is that we consider it as a prerequisite for applying our generic production model to one of the most important transformation process for float glass, namely the soft-coating transformation line.

Coating lines were not that easy to capture. Basically, coating lines are made of metallic cathodes that are used on-line to sputter nanometric metallic coats on flat glass sheets. Before optimizing the production planning of coating lines, managers needed to have a decision-support tool to configure the on-line cathode sequence, the so-called set-up of the line: the notion of set-up was called design in this chapter, in order not to avoid confusion with the traditional notion of set-up in batch production.

Once the line is configured with a given design, we may perform a portfolio of transformations. Changeovers between designs are time-consuming and thus represent opportunity costs. Moreover, for a given design, it exists also changeovers between distinct transformations. However, the most important thing to reduce the overall production planning cost (and thus to use efficiently a coating line) is to minimize utilization costs by optimizing the metallic cathode use. In this chapter we dealt with this issue: given a product portfolio and a set of on-hand cathodes, we determine the optimal design under given deterministic anticipated requirements.

Once various designs are settled, it becomes indeed easy to capture the line using the notions of attribute and sub-attribute. Depending on the production planning level (see the chapter 4 for explanations about hierarchical planning), we may need one or several attributes.

On the one hand, at the tactical level, the only attribute we need is precisely the design of the line, which can take several values between which we have change-over times and costs, and the two sub-attributes are both the nature of the performed transformation and the nature of the transformed product[a].

On the other hand, for operational production scheduling, we consider the design as a data and we only use one attribute: the performed transformation.

At this point, we have confirmed the generic aspect of the production planning model introduced in chapter 4. We will see in chapter 6 how we use it as a building block for modeling the overall production-inventory and distribution processes of the supply chain.

Of course, the underlying idea of this work is to be able to better understand the interest of a simultaneous production planning of several different production lines: for instance, at the operational level, is there a "dominating" process that must be planned before planning other processes, or is it justified to plan on-line processes? Using the example of the glass industry, does the float line dominate coating lines?

5.6.2 Outlooks of our research

During this research on the coating line design problem, we have identified several research outlooks that may be highly interesting and motivated by industrial issues.

First of all, we have worked under the assumption (see §5.2.1) that we know the transformation portfolio to produce. It clearly appears that from a practical point of view the assignment of transformation quantities to several portfolios is a complementary optimization problem. We did not focus on it to simplify the

[a]which is in fact a family of aggregated products according to the business and of the decision level

problem we had to tackle. However, this is a critical step of our reasoning: optimize the design corresponding to a wrongly chosen portfolio may be far from the global optimum!

Based on our model, we have in practice followed a heuristic iterative methodology to determine the portfolio: this is the motivation we gave for our second objective $\mathtt{Obj}_2$ in (§5.3.1). We started with a small portfolio, and then we add products while the line was able to produce them. The global optimization coupling both the portfolio determination and the coater line design appears to be an excellent perspective for future research.

So far, we have worked under deterministic assumptions. Of course in practice, forecast demands are by nature uncertain. Modelling the robustness of a solution under stochastic inputs is in our opinion another motivating research opportunity. For instance, what would become this coater design optimization problem under a stochastic forecast demand?

Last but not least, a more strategic potential reflexion lies in the redefinition of the process. Since the creation of the industrial process in the late nineties, coating lines have been created as on-line metallic cathodes. It would be useful to take time to imagine alternative processes. For instance, what would be the gain of using several parallel lines, each one being sputtering a given metal as plotted on Figure (5.3)? Of course, some possibilities may not be feasible for technical reasons, but we think that an in-depth study of the impact of the technical choice on the overall flexibility of the line may have an industrial interest.

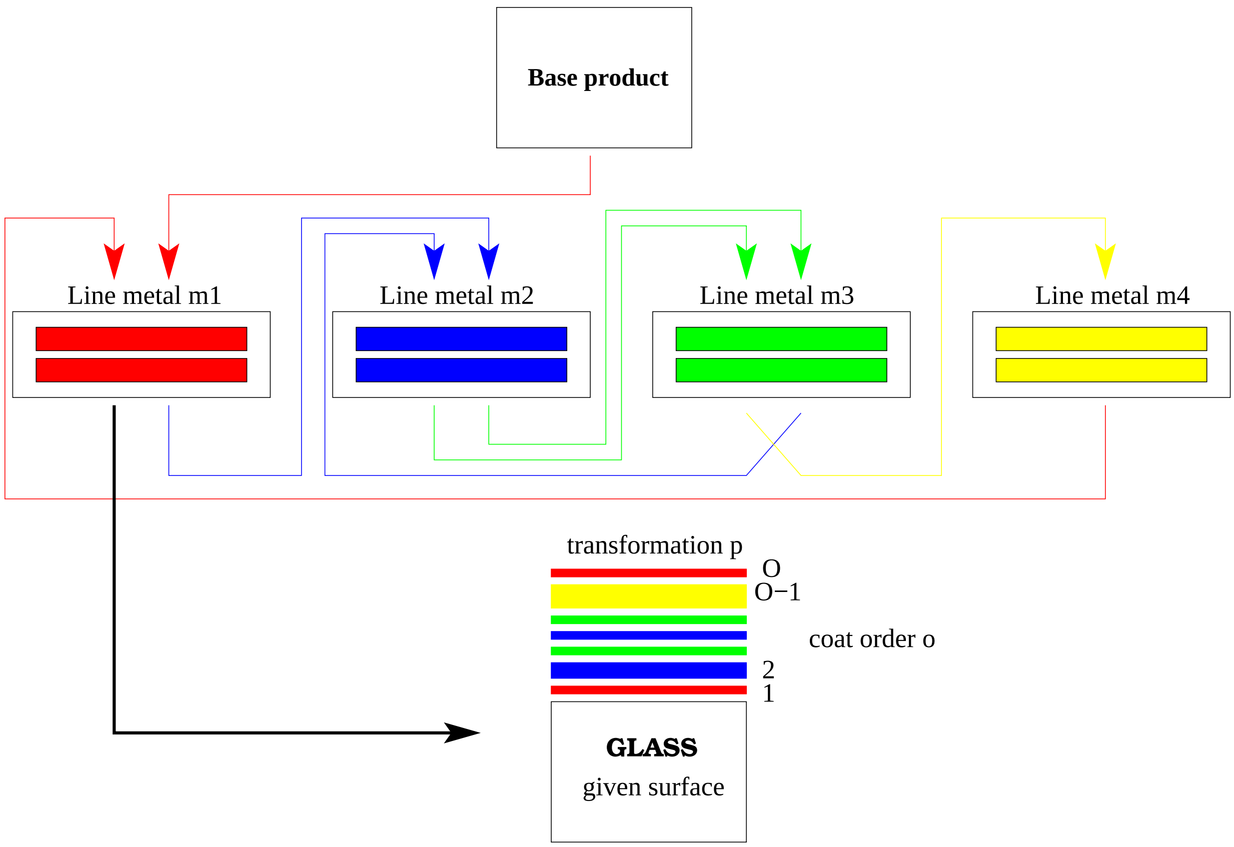


Figure 5.3: Example of redefinition of the sputtering process

Chapter 6

The ROADEO project: An integrated production-inventory and distribution model

We address the problem of developing a decision tool for both production planning and logistic decisions in the glass manufacturing industry.

In chapter 4, we have developed a generic production planning model allowing us to capture continuous processes. We have highlighted that it may be used at every level of a hierarchical production planning process. However, one may wonder whether it makes sense to separately optimize the production planning on the one hand and the logistic system on the other hand.

Using our production model as a building block, we integrate this work in a multi job, multi machine and multi location model. The focus of the present chapter is on providing a powerful modeling and optimization tool for combined production and logistic decision making.

We apply our research to different decisions we have met and solved in the glass industry. Firstly, float glass is mainly transformed through different processes to provide commodity products, such as laminated glass or coated glass. We explain how our model captures these production processes. Secondly, we apply our tool to the tactical production planning, minimizing production, storage and transportation costs. Finally, we present how do we create a generic decision support tool for strategic decisions such as the location of a new facility. We provide several practical approximations allowing overcoming the tremendous size of industrial applications.

6.1 Towards an integrated production-distribution model

In the literature review (§ 4.1.2) of chapter 4, we focused on lot-sizing models. These models may be taken into account as a part of an integrated production model. They may capture several levels. As recalled in [SC01], the terms "multi-stage" and "multi-level" have essentially the same meaning and therefore in this chapter we use the multi-stage term. At the single-stage version of the problem we are faced with a set of net requirements which are produced by the Material Requirement Planning (MRP, see [Bak93]) explosion and netting steps, and we must choose a set of lot sizes. At each level the problem resembles the single level problem, but with the additional property that the lot sizes at each level, which form the solution, also cause part or all of the demand at the next level down the product structure. The problem is thus to simultaneously find a set of lot-sizes at each level, that combined together, will minimize the total fixed and holding costs in the system. The multi-level capacitated lot-sizing problem (MLCLP), originally described in Billington and al. ([BBM+89]), deals with resource-constrained multi-stage systems so as to minimize the sum of production, set-ups and inventory costs. It is shown ([FABC97]) to be NP-Hard. Roux and al. ([RDPL99]) focus on an integrated multi-site environment in order to determine a feasible sequence in each site. Their method alternates between solving a planning problem in which lot-sizes are computed for a given sequence of jobs on each machine, and a scheduling problem in which sequences are computed independently in each site. We may remark that sometimes, single-stage multi-product problems can be multi-stage if the production of one item is dependent on another. Otherwise, the production can be called single-stage or single-level ([MW88]).

In the previous chapters, we dealt with production models. But the supply chain of a typical product starts with material input, followed by production, and finally distribution of the end product to customers. Therefore the cost of a product includes not only the cost of factory resources to convert materials into a finished item but also the cost of resources to make the sale, deliver the product to customers, and service the customers.

As a consequence, in order to reduce costs, firms have to plan all the activities in the supply chain in a coordinated manner. It is well recognized that there is a greater opportunity for cost savings in managing supply chain coordination than in improving individual function areas. Various types of coordination in a supply chain have been studied in the literature. We discuss here the coordination of production and distribution, which can be decoupled if there is a sufficient amount if inventory between them. So far, Saint-Gobain Glass managed these two functions independently with little coordination. Fierce competition in today's global market and increased expectations of customers have forced companies to invest aggressively to reduce inventory levels across the supply chain on one hand and to be more responsive to customers on the other. Reduced inventory results in closer linkages between production and distribution functions. Consequently, Saint-Gobain Glass decided to optimize production and distribution operations in an integrated manner to realize cost savings and improve customer service.

The interdependency between production and distribution operations, and the corresponding trade-off between the costs associated with them can be illustrated intuitively by the following simple example derived from the glass industry. Consider Saint-Gobain Glass producing various products for various customers, from various plants. We saw in chapter 1 that set-up costs of colored glass and transportation costs are key factors in this business because they represent huge potential savings.

To save distribution costs, orders of closely located customers may have to be produced at similar times so that they can be consolidated for delivery right after they are produced. However, orders of closely located customers may require very different production set-ups, and producing them at similar times may incur a large production cost. Of course, in addition to production and distribution, there are other factors such as inventory and capacity that play important roles.

6.2 Literature Review on integrated production-distribution models

Many different models in the literature involve joint considerations of production, inventory and distribution. Based on the framework for analysis developed by [Ant65], there are first strategic models that integrate design decisions in the supply chain such as location, plant capacity, and transportation channels. Excellent reviews on these models may be found in [VG97], but also in [OD98] already quoted in chapter 3. Recent results in this area are presented in papers such as [JP01] or [SCD04]. The ROADEO model developed in the present chapter may be used as a strategic model, as we see it in (§ 6.6), but was primarily intended to address tactical issues.

Sarmiento and Nagi analyzed in [SN99] work on integrated analysis of production-distribution systems and identified important areas where further research was needed. They reviewed work at either strategic or tactical levels that explicitly considers the transportation system in the analysis, since they were interested to understand how logistics aspects had been included in the integrated analysis. For instance, works such as [PC90] and [PC93] are not covered.

More recently, Chen ([Che04]) provided a comprehensive updated review on integrated production and distribution models. He focuses at both tactical and operational decision levels on models which explicitly involve both production and distribution models, so-called Explicit Production-Distribution models (EDP). Chen classifies various EDP models into five classes based on three dimensions: decision level (tactical, operational), integration structure (inbound and outbound transportations, production) and problem parameters (length of the planning horizon, nature of the demand).

The ROADEO model proposed here involves multiple products (based on our decomposition into attributes and sub-attributes, see chapter 4) and/or multiple time periods, as well as a finite horizon and a dynamic demand over time. It captures three (manufacturers, warehouses and customers) or more stages: we may capture various global supply chains by designing freely every component.

A typical model of this problem class involves two stages, i.e. one manufacturer and several customers. The manufacturer produces various products to satisfy dynamic customers' demand over several time periods. The demand is known in advance and must be satisfied without backlog. Production costs capture both fixed costs and set-up costs. Both the manufacturer and the customers can hold inventory. Unit inventory holding cost and initial inventory levels are given. Finally, each shipment from the manufacturer to customers is capacitated and costly. The problem is thus to determine in each time period:

- how much to produce at the manufacturer,
- how much to keep in inventory (at the manufacturer and at each customer),
- how much to ship from the manufacturer to customers,

so that the total cost including production, inventory and transportation is minimized.

Papers dealing with this general tactical model may be divided into two parts, depending on their static or dynamic division of time.

On the one hand, some authors deal with the static case, i.e. a single time period. Cohen and Lee [CL88] consider a four-stage model (multiple suppliers, plants, distribution centers and customer zones) in which demands are stochastic. They aim at determining ordering policies (lot-sizes, reorder points, etc.) so that total system-wide cost is minimum subject to a certain level of customer service level. Chen and Wang ([CW97]) focus on a three-stage problem (suppliers, plants and customers) inspired by an industrial steel production and inventory problem in which demands are deterministic. They maximize the revenue of operations through a Linear Program solved directly by a commercial code.

On the other hand, various dynamic models (i.e. with multiple time periods) have also been proposed in the literature. First of all, a single product model applied to a real-life case (through a MIP solved in an undescribed manner), manufacturing of Urea fertilizer in India, is developed in [Haq91]: Haq considers production stages, warehouses and retailers with deterministic demands. All stages can hold inventories and backlog is allowed. The objective function is to minimize total production, inventory and transportation cost plus backlog penalty.

In [CF94] Chandra and Fisher work globally on the general model described above with a single production facility , and they compare sequential (first production, then transportation) and integrated approaches. They highlight for instance that value of cooperation increases with relatively high transportation costs (fixed and variable) compared to production costs. In [FV99], Fumero and Vercellis add to the previous model a limited fleet of vehicles for product delivery. They solve the corresponding MIP by Lagrangian relaxation, following a method they develop in [FV97].

In [Seg96] Segerstedt presents a mathematical formulation of a capacity constrained multistage inventory and production control problem, which is formulated in a dynamic programming recursion.

More recently, Barbarosoglu and Ozgur [BO99] consider a 3-stage, multi-product problem involving one plant and multiple warehouses and customers. Demand is deterministic and dynamic and has to be satisfied in a Just-In-Time fashion, i.e. no inventory is allowed at the customers. Transportation cost has fixed and variable parts. They formulate a MIP and solve it by Lagrangian relaxation which decomposes the problem into two subproblems: production and distribution. Ozdamar and Yazgac [OY99] develop a hierarchical planning approach for such a problem: at the aggregated level (time periods are aggregated into bigger ones), production set-ups are ignored to drop some binary variables, while an iterative constraint relaxation scheme is used to solve the disaggregated MIP model. Finally, Ozdamar and Barbarasoglu combine Lagrangian relaxation with a simulated annealing procedure to solve the Multi-level Capacitated Lot-Sizing problem (MLCLSP, see [OB00]). Dualizing capacity constraints create the Multi Level Lot-Sizing Problem (MLLP), which is a hard-to-solve problem, for which the recent work of Moon et al. [MJH02] provides a good heuristic based on genetic algorithms in the case of minimization of the total tardiness in the supply chain, following previous works ([GOV96]). Thus, authors in [OB00] dualize both capacity and storage constraints to fall into a simple problem solved by the WWA (see § 4.1.2, [WW58]). Their heuristic appears to be very efficient.

Some papers capture original parameters, based on practical applications. For instance, Mohamed [Moh99] considers also a 2-stage, multi-product model with deterministic dynamic demands, but he defines production capacity as a variable: any capacity change involves a given cost. He also captures exchange rates of the host countries of the facilities in each time period, and illustrates the usefulness of such a model on an example.

Sambasivan and Schmidt present a heuristic approach to solve an integrated, multi-plant production planning problem (MLCLSP, [SS02]) that is observed in a large steel corporation in United States of America. Each plant is capable of producing all the products with various production costs, and demand occurring at one plant may be satisfied by producing and transferring from another plant. Authors briefly discuss the results obtained from the uncapacitated problem before proving that the capacitated one is NP-hard and introducing their heuristic, which is proved to be very efficient. At this time, they discuss the fact that a Lagrangian relaxation of capacity constraint would be useless because it would lead to the uncapacitated multi-plant problem which is known to be NP-Complete ([Sam94]). However, they themselves address this issue in their last article [MY05].

Otherwise, constrasting with existing Lagrangian Relaxation approaches that relax capacity constraints and/or inventory balance constraints, Chen and Chu's ([CC03]) approach only relaxes the technical constraints that each boolean setup variable must take value 1 if its corresponding continuous variable is positive. Numerical experiments show that their approach can find very good solutions for problems of realistic sizes.

Timpe and Kallrath ([TK00]) describe a general mixed-integer linear programming model based on a time-indexed formulation covering the relevant features required for the complete supply chain management of a multi-site production network. Their application is taken from the chemical industry (BASF), by they argue that the model provides a starting point for many applications in the chemical process industry, food or consumer goods industry. They introduce an interesting concept of different time scales attached to production and distribution, so that the resolution is chosen adequately for the purpose of both production planners and marketing people. They use a commercial software for resolution (XPRESS-MP), showing that it gives practical results even if it is sometimes hard to prove optimality.

Finally, Guinet ([Gui01]) proposes a two-level production management approach to control multi-site production systems. It integrates resource capacity constraints and optimizes variable costs (processing costs, transportation costs, holding costs, delay costs) and fixed costs (set-up costs). It results in a global multi-site production planning and in local multi-workshop scheduling. Material and capacity requirements are both included in production planning and multistage workshop scheduling. A primal-dual approach is proposed to solve this problem.

Examples of successful applications on real-life cases are pretty numerous.

Blumenfeld et al. [BBD+87] developed a decision support tool for the analysis of the logistics operations at General Motors, that identified a logistics cost savings opportunity of nearly three million dollars per year, while they only focus on the trade-off between inventory and transportation costs.

Zuo et al. [ZKM91] consider a real-life problem of seed corn production and distribution involving two stages with multiple production facilities and multiple sales regions. Demands are deterministic. Each facility produces either nothing or more than a given minimum quantity. They develop a MIP and apply it successfully, reporting savings of about ten millions of dollars.

Arntzen et al. [ABHT95] study a real life problem encountered at Digital Equipment Corporation. Their model capture multiple products in a two-stages supply chain (plants and customers): several transportation channels are available, demand is deterministic and dynamic and backlogging is forbidden. It is reported that the results of this study saved DEC over one hundred million dollars.

More recently, some authors worked under stochastic demand. Gnoni et al. ([GIM+03]) deal with lot sizing and scheduling problem (LSSP) of a multi-site manufacturing system with capacity constraints and uncertain multi-product and multi-period demand. Manufacturing capacity at each site is affected by machine failures and repairs as well as by sequence dependent setup times. LSSP is solved by an hybrid model resulting from the integration of a mixed-integer linear programming model and a simulation model. The model proposed is applied to a supply chain of a multi-site manufacturing system of braking equipments for the automotive industry. The hybrid modeling approach is adopted to test a local as well as a global production strategy in solving the LSSP concerned. The comparison is based on an overall economic performance measure defined as the sum of setup, holding, and delivery delayed costs. In the case study investigated, the local production strategy allowed a reduction of about 19% of average overall cost respect to the reference actual situation. The approach could thus help decision making in adopting a cooperative, rather than competitive, production strategy.

This is also highlighted by Lopez et al. ([PLYG03]) who describe a model based on predictive control strategy to find the optimal decision variables to maximize profit in supply chains with multi-product, multi-echelon distribution networks with multi-product batch plants. The key features of this paper are a discrete time MILP dynamic model and a general dynamic optimization framework that simultaneously considers all the elements of the supply chain and their interactions as well as a rolling horizon approach to update the decision variables whenever changes affecting the supply chain arise. The paper compares the behavior of a supply chain under centralized and decentralized management approaches, and shows that the former yields better results, with profit increases of up to 15% as shown in an example problem.

As said before, ROADEO is a project mainly designed for addressing tactical production, inventory and distribution issues. For instance, in (§ 6.7) we apply it as a model belonging to the fifth class of Chen's classification ([Che04]), so-called general tactical production-distribution problems. We extend a work pretty close to the FLAGPOL project ([MDE93]), an optimization model developed and applied in one of the strongest competitors of Saint-Gobain Glass, the Pilkington group, resulting in annual cost savings of two million dollars. In this latter model colour campaigns had to be specified, whereas ROADEO optimizes the production planning by minimizing the sum of production, inventory and distribution costs.

In a slightly different 4-stage, multi-product model which deals with strategic decisions such as facility, Dogan and Goetshalckx [DG99] solve a real-life problem in the packaging industry and achieve around eight million dollars of annual costs savings, representing 2 % of total cost.

Finally, Dhaenens-Flipo and Finke [DFF01] study a 3-stage, multi-product problem involving multiple plants (with multiple lines), warehouses and customers, then they apply it to a real-life problem, the manufacturing of metal items) by solving it directly by Cplex (see [ILOa]).

We introduce in the present chapter an original extension based on the production planning model we saw in chapter 4 which captures supply chains with as many stages as necessary, depending on the application to capture. We apply it to a complex industrial application in the glass industry by using a commercial software.

6.3 Introduction

In our chapter 4, we have introduced a framework to model some production planning issues, ideally all in process industry (we applied it to the glass manufacturing industry). We have developed the corresponding way to solve it through a mixed integer linear program: Given common particular structure of continuous processes, we have proposed a method dividing multi characteristic products into attributes and sub-attributes under simple assumptions. This structure matches with a hierarchical framework to model production line capabilities, introducing relevant variable production costs at each level. Our reasoning was thus suited to any level of the hierarchical production planning system. In a nutshell, this model captured a bigger scope than the former ones found in the literature and introduced less integer variables, being thus solvable by commercial codes.

Let us now consider this previous work as a building block. From now on, we will consider it as our production black box for modeling. Whatever the process we are dealing with, we assume that this model captures all constraints and costs. This way, we only need to understand its inputs and outputs to integrate it within our general model.

6.3.1 Application to the glass manufacturing industry

Float glass manufacturing industry has been precisely described in chapters 1 4. We saw two interesting characteristics of this business. Firstly, the replenishment of raw material is perfect (see Assumption (2) on page 7). Secondly, transportation cost has a key role in the supply chain. In chapter 4, we focused on the operational production planning problem at Saint-Gobain Glass. We will see further that PLANEO is a particular application of the ROADEO project, used as a single location three-stage (plant, inventory and customer) problem. In order to emphasize the wide range of applications of ROADEO, we focus in the forthcoming section on strategic and tactical issues.

6.3.2 Tactical and strategic decision levels for industrial and logistic issues

Our research deals with both tactical and strategic decision level, for both industrial and logistic issues. Let us precise what it does mean.

On the one hand, we define as an industrial issue any problem in which production facilities are not totally known or fixed. We distinguish two levels of decisions. Firstly, the strategic level corresponds to models in which plants' location, opening or skills are variables that must be determined. On the contrary, at the tactical level, the industrial scheme is fixed. Each production facility is perfectly defined by a location and a set of skills (capacities for all products). In this case we focus on tactical production planning, that captures product flows within facilities. This planning is characterized by longer time horizon and period than operational planning[1].

On the other hand, we define logistic issues the class of problems in which production is known and fixed. Thus production may be considered as product sources and by symmetry customers as product wells. Thus, variables may be the inventory

[1]that we tackled in chapter 4

facility location and skills at the strategic level whereas the tactical one deals with product flows through different transport resources.

6.3.3 Relevant costs

The basic trade-off to be handled by our tactical industrial and logistic problem is to minimize the sum of both variable production costs, inventory costs and transportation costs. At the strategic level, we may add up the facilities opening costs corresponding to the various facilities. Trade-off for logistic decisions is similar except for variable production costs that does not exist[2]. Let us describe each part of the potential objective function.

Variable production costs depend on facilities, given various national laws and various organizations. They may include the set-up related costs which only depend on the kind of chosen changeovers within products. These costs are precisely described in chapter 4.

Inventory cost is the cost of carrying one unit in inventory for a specified period of time. It is a combination of the cost of capital, the cost of physically storing the inventory and the cost that results from the product becoming obsolete (see [CM01c]). In addition, we let the user free to authorize or not optional costs associated with imperfect service, such as backorder costs. This mainly allows us to check if it is possible to reduce global cost by postponing a particular production campaign.

Transportation costs are associated with any product flows within different facilities. We define various transport resources: each one is characterized by a capacity and both a variable and a fixed cost. Transportation means have all their own cost. As a first step, we consider it as an affine function of the distance travelled. Depending on the goal of our optimization, we may use different distances. For tactical decisions in which the industrial and logistic sheme is fixed, we may use

[2] because production is fixed

real distances within facilities (assuming that corresponding distance matrices are available). In this case we may even use costs matrices. On the contrary, strategic decisions need more flexibility to be able to generate new facility locations and to compute distances and costs easily. In this case we use the Euclidean distance approximation: we approximate real distances proportionally[3] to the Euclidean one.

Finally, opening costs of facilities in strategic decision models capture the fixed costs of any facility construction. To be more specific, we consider an individual opening cost as a concave function of the associated product flow. Basically, the bigger the concerned product flow[4], the bigger the facility, the cheaper the average cost by flow value (due to important economies of scale).

6.4 A general framework for tactical production-distribution planning decisions

First of all, we need to clarify the way we model the industrial and logistic system.

Any spatial logistic organization is nothing but a network of facilities linked to one another by a set of oriented paths. Obviously, each facility is a node whose nature depends of its activity. For instance, a customer may be understood as a well whereas a plant is a source.

Flows within facilities are made possible by a set of transportation resources[5], which link defined types of facilities.

Finally, the whole scheme exists in order to create, store and serve a set of products to customers. In an industrial organization, these products are produced from scratch and then transformed by production lines. We assume that each production line is captured by our model framework presented in chapter 4.

[3]usually, the used coefficient is around 1.3
[4]production for plants and entering flows for warehouses
[5]means of conveyance: trucks, trains, planes, etc.

6.4.1 Geographical and functional zones

We first describe a spatial network through a set of nodes. Each node is characterized by a geographical position[6]: we denote it a *geographical zone*. On a given location, it may happen that several facilities have been gathered. For instance, in the float glass industry, each production line is coupled with a warehouse on a plant site.

By facility, we mean a defined resource whose activity is unique. This statement clearly implies the notion of *functional zone type*. For a given organization, we first study the list of functional zone type. Such types may be a production site of raw materials, a transformation site, a warehouse for row materials, another for transformed products, a first level storage platform, a second level one, and finally a customer. These distinctions within types will allow us (see paragraph (6.4.2)) to define precisely the authorized flows within functional zones. Of course, **one functional zone belongs to an unique geographical zone and is characterized by one unique functional zone type**. Figure (6.1) illustrates an example of supply chain made of five different functional zone types.

Thus, a geographical zone may gathered several functional zones. The most important interest of this distinction is the possibility to study precisely flows within functional zones.

In cases for which we use an approximation for computing distances and transportation costs within functional zones, we can define rebate factors, both for upstream and downstream flows, to capture discounts on particular links. For instance, trucks in the glass business are dedicated transportation resources: for each delivery, we must pay the round trip of the truck. Thus, delivering potential reloading zones is cheaper than delivering basic customers. This creates an upstream discount factor on reloading zones.

[6] defined through any coordinate system

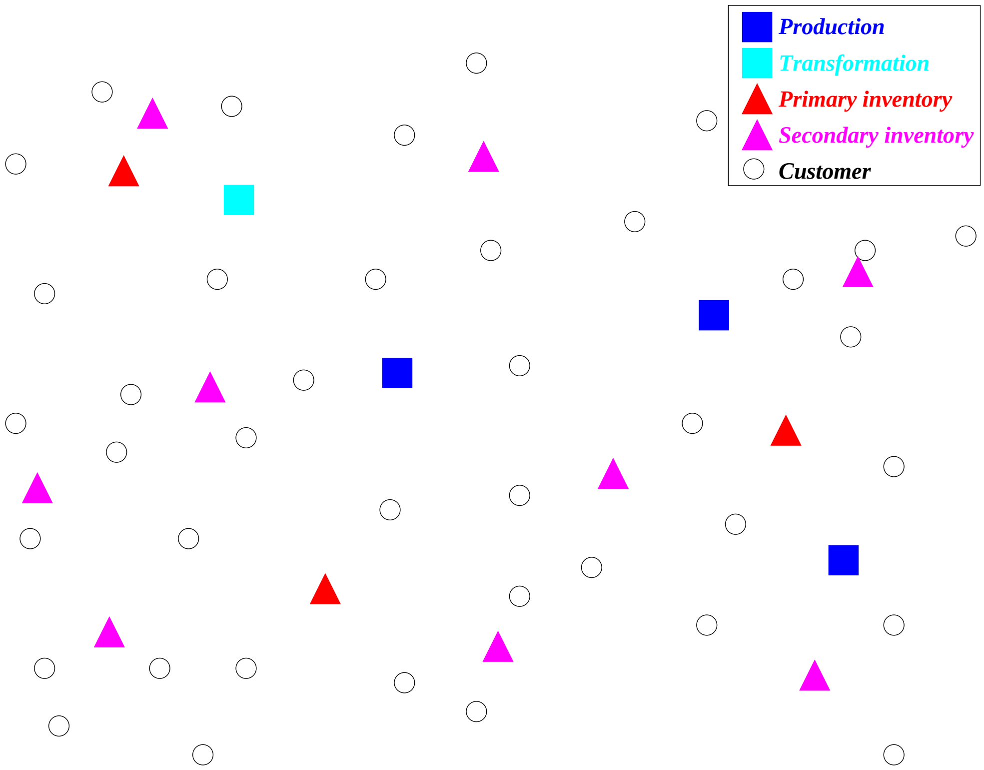


Figure 6.1: Example of industrial and logistic pattern based on five functional zone types.

6.4.2 Transportation within the industrial and logistic scheme

Links within functional zones are characterized by a distance, computed according to a variable methodology[7]. This link may be done by several transportation resources.

Each *transportation resource* is defined by a set of skills (defining the set of products that may be conveyed[8]), a capacity, and both a speed and cost. By definition, these resources are discrete: it is then possible to capture phenomenon such as mixing products in the same resource when it is cheaper than sending several nearly empty ones to fulfill a mixed customer order. However, we may chose to relax the global transportation problem through a linearization of the resources when we do not need to capture detailed flows. Naturally, each product may at least be conveyed using one resource. Figure (6.2) illustrates the addition of four transportation

[7]cf. discussion in 6.3.3

[8]we have taken into account the classical constraints of minimal and maximal proportion for each product

resources types in the previous example of supply chain. To keep it clear, we did not plot all links, but it shows that we are able to define links within functional zones depending on their types.

To underline the interest of the functional zone type notion, we have integrated the concept of *transportation skill*. Each skill defines for two given functional zone types of departure and arrival the set of competent[9] resources. Intuitively, each resource on each path is a continuous variable for each product. Using these skills, we generate only meaningful variables: each path within functional zone is thus oriented. For instance, in a three level supply chain (provider, national warehouse, local one and customer), we are able to define only possible paths: from the provider, paths to national warehouses are the only generated. Without this notion of transportation skill, the concept of functional zone type would be useless.

Finally, we have implemented the possibility to bound the number of used resources during each time period. In businesses in which availability of these resources is a key point this point may be crucial.

[9]We also define a minimal and maximal proportion for each resource

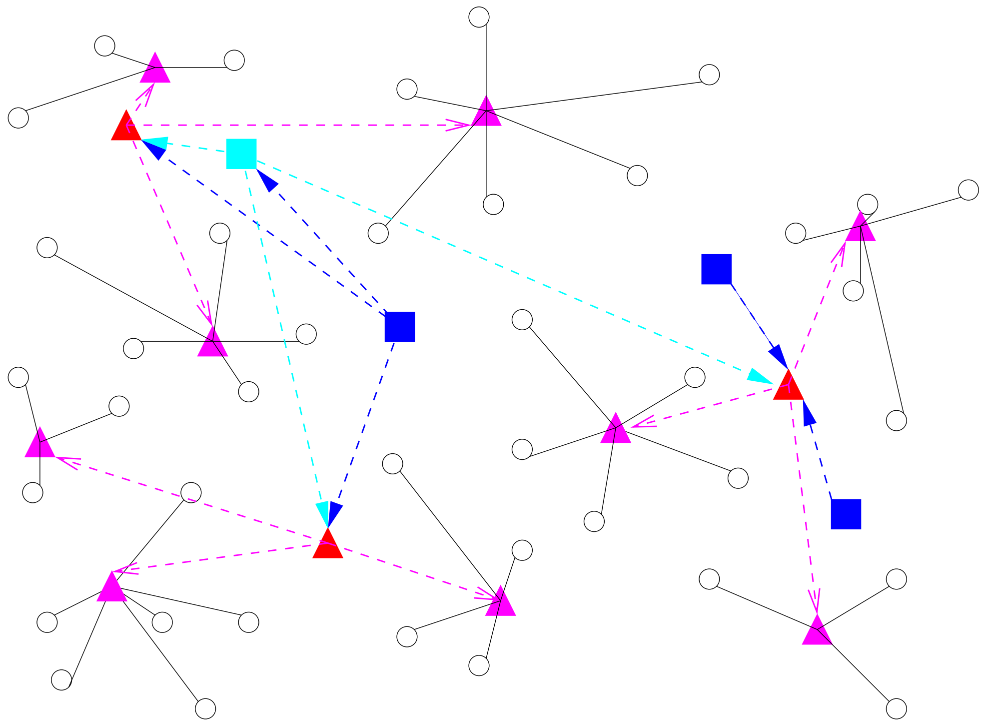


Figure 6.2: Defined flows within the example of industrial and logistic pattern

6.4.3 Absolute and Relative products

So far, we have dealt with facilities and transportation resources within them by using the notion of products. Thus implicitly we refer to the set of products we are dealing with. Clearly, we are working with aggregated real product sets whose aggregation level and accuracy depend on the hierarchical level at which our decision has to be taken.

In case of industrial issues, production planning is a variable. We have assumed that each production facility structure is captured by the framework developed in chapter 4. In this thesis, we have developed a decision tool in which we aimed at scheduling the production of a set of virtual products which were decomposed into attributes on a discrete horizon time. They corresponded to a set of products through sub-attributes. This powerful model allows us to define both production and transformation line (see paragraph (6.5.2)). However, it is fundamental not to confuse the notion of product in each chapter.

To use this production model as a building block in our generic framework, we introduce a difference within absolute products and relative ones.

On the one hand, *absolute products* are physically existing: they correspond to the set of aggregated sets of real products we are dealing with in our optimization model. Customers' demand, inventories and flows are expressed in absolute product units. Important specifications are the correspondence table within different *units* of a given absolute product.

On the other hand, *relative products* correspond to the ones used in our production building blocks. A relative product is defined by attributes and sub-attributes relative to the *job of the production line*. Each production line is defined by a set of skills and a capacity in a given unit. In case of a transformation process, each produced relative product is characterized by a weighted[10] list of absolute products which are necessary to its conception, and a yield[11] of the transformation.

We ensure the links between absolute and relative products through a given classification: **each relative product corresponds to a unique absolute product**. Thus, we may have one absolute product that is produced by different production lines of different processes.

6.4.4 Inventory zones and Others

Among functional zones, we create two main categories: inventory zones and others. Each class is characterized by specific properties and nature. To simplify this paragraph, we use the notion of products instead of absolute products.

Firstly, we consider some functional zones as inventory facilities. These zones are clearly defined by the fact that there is neither creation nor consumption of products on it.

[10] the sum of weights equals one

[11] belonging to $[0, 1]$.

To capture several optional real constraints such as limited workforce or facilities, we define for each storage zone limited maximal input, output and general product flows per time unit. We consider each storage zone divided into areas with dedicated products. Each area is defined by a list of potential stored products for which we can specify a maximal storage capacity.

Secondly, all other functional zones are either consumption or production zones. In both cases, we may consider them as either product wells or sources. As explained in introduction[12], production may be or not known.

On the one hand, customers and known production zones are similar to each other: they are characterized by a list of product flows. Each product flow is defined by a positive or negative (respectively for customers and known production zones) quantity per time unit which is consumed over a specified time interval.

On the other hand, any unknown production zone must be described clearly, based on our framework introduced in chapter 4. First of all, we may define all the jobs that we want to capture. It is required to describe each job as made of attributes and sub-attributes: each job is thus able to produce a list of corresponding relative products. Each unknown production zone may contain at least one production line. Each production line belongs to a job. We define each line by a set of production skills, changeover skills, etc. Several options such as forced initial or final products, production breaks or anticipated production end are possible, using all options described in chapter 4.

From a practical point of view, we present how we did develop the ROADEO software. We use the `C++` code and follow the object programming fashion. Definitions of classes and relationships between them come directly from the physical concept we have introduced so far. Appendix (D.1) illustrates the way we have worked.

[12]part (6.3)

6.5 Mathematical model for tactical production-distribution decisions

6.5.1 Notation

Data

First, we denote $\mathcal{P}_A$ the set of absolute products, that we index a. The same way we denote $\mathcal{P}_R$ the set of relative products, indexed r.

By convention, we define $\mathcal{P}(\mathsf{a})$ the set of relative products (at least only one) $\mathsf{r} \in \mathcal{P}_R$ that corresponds to a given absolute product $\mathsf{a} \in \mathcal{P}_A$.

We still use a discrete time model: time periods have a constant duration $\delta\mathsf{t}$ corresponding to the reference time scale in chapter 4. We index time period using $\mathsf{t} \in [1, \mathsf{N}]$. $\mathsf{N} \times \delta\mathsf{t}$ is thus the time horizon.

General options are specified as parameters of our model, such as whether we authorize back-orders or not, yearly financial interest rate used to compute storage cost (we denote it η), etc.

To denote the set of F functional zones we are dealing with, we use the notation of the set $\mathcal{F}$. Following explanations of paragraph (6.4.4), we create several subsets:

- $\mathcal{F}_\mathsf{K}$ is the set of known production zones or customers. We denote $Q^{\mathsf{t}}_{\mathsf{f},\mathsf{a}}$ the quantity of absolute product a that is consumed during the time period t at the functional zone $\mathsf{f} \in \mathcal{F}_\mathsf{K}$. By convention, a negative quantity is a known production.

- $\mathcal{F}_\mathsf{P}$ is the set of unknown production zones. Each unknown production zone $\mathsf{f} \in \mathcal{F}_\mathsf{K}$ contains a set of production lines $\mathsf{u} \in \mathcal{U}_\mathsf{f}$. Each line u is defined by a set of production skills of relative products $\mathcal{P}^{\mathsf{u}}_{\mathsf{R}}$, for which we may define a production cost[2] $c^{\mathsf{P}}_{\mathsf{f},\mathsf{u},\mathsf{r}}$. Each produced relative product may be the result of a transformation: it may require the consumption of absolute products. For each relative product r produced by the line u, we denote $\mathcal{T}_{u,r}$ the set of consumed absolute products (which are in this case raw materials). Each absolute product $a \in \mathcal{T}_{u,r}$ has a corresponding weight $w_{u,r,a}$ in the reaction[13], which is characterized by a yield $\rho_{u,r}$.

- $\mathcal{F}_I$ is the set of inventory zones. Each inventory zone $f \in \mathcal{F}_I$ may be characterized by maximal input, output and overall flows F_f^{in}, F_f^{out} and F_f^{all}. We also need the storage cost of the zone, denoted h_f (in money per time and per product unit). Each inventory is divided into areas $s \in \mathcal{A}_f$. This set is a subset of the global set of storage area $\mathcal{A}_f \subset \mathcal{A}$. For each area s, we denote C_s its storage capacity, and $\mathcal{P}_A^s$ its set of stored absolute products. In addition, each product is given with its a price cost[1] $c_{s,a}^S$, a handling cost[2] m_a^s and backorder cost[3] b_a^s. Let us assume that we know the initial inventory for every product. We denote it $I_{s,a}^0$.

Finally, we denote $\mathcal{L}$ the set of links among functional zones, according to the set of transportation skills within functional zone types. As explained in part (6.4.2), each path between two functional zones may be covered by some transportation resources. For each link $l \in \mathcal{L}$ whose distance is denoted d_l, we denote Ψ_l the set of competent transportation resources: it is a subset of the global set of transportation resources $\Psi_l \subset \Psi$. For a given path, each resource proportion must belong to a given range[5] $[w_{l,\psi}^m, w_{l,\psi}^M]$. In addition, in case of limited transportation resource, we define the function $N_\psi^M(t)$ that gives the maximal number of resources ψ used during the time period t.

Each resource $\psi \in \Psi$ is characterized by a set of absolute products $\mathcal{P}_A^\psi$ that may be carried, a capacity C_ψ, a transportation cost[4] c_ψ^T and a speed v_ψ^T. In a first step, **we assume that this speed is infinite and thus all transportation times**

[13]defined such as $\sum_{a \in \mathcal{T}_{u,r}} w_{u,r,a} = 1$

are zero. We will discuss how this assumption may be relaxed later.

Each product proportion must belong to a given range[5] $[w^m_{\psi,a}, w^M_{\psi,a}]$.

For each functional zone $f \in \mathcal{F}$, we denote $L^{in}(f)$ (respectively $L^{out}(f)$) the set of links which arrive in (respectively start from) this node.

Variables

Let us introduce main variables in order to be able to write down our mathematical model. Basically, we mix both real and integer variables:

- $X^t_{l,\psi,a}$ is the non negative quantity of absolute product a which is carried during time period t on the link l by the transportation resource ψ. It is a positive flow for the arrival zone and thus a negative one for the departure one.

- $N^t_{l,\psi}$ is the non negative integer number of transportation resource ψ on the link l during time period t. We will discuss later the important possibility to do a linear relaxation on those variables.

- $I^t_{s,a}$ is the on-hand inventory of absolute product a at the end of time period t in the storage area s. This continuous variable must be non-negative. In case of authorized back-orders, we introduce (see chapter 4) non negative real variables $I^t_{[+]a}$ and $I^t_{[-]a}$.

- $R^t_{f,u,r}$ is the production of relative product r during t on the production line u of the unknown production zone $f \in \mathcal{F}_P$. Naturally, we find the direct analogy between these variables and the ones defined by R^t_p in chapter 4. Each production line is defined by a set of Ω attributes and Λ sub-attributes. Its skills are modeled by integer variables $y^t_{f,u,w,i}$, $v^t_{f,u,w,\beta}$ and $w^t_{f,u,w,\alpha}$ whose definitions may be found in chapter 4. We denote the changeover cost[1] of

[1] in currency unit (such as €)
[2] in currency unit per product unit (such as €/ ton)
[3] in currency unit per product unit and per unit time (such as €/ ton / day)
[4] in currency unit per distance unit (such as €/ mile)
[5] with $w^m \geq 0$ and $w^M \leq 1$

type α by $c^{C}_{f,u,w,\alpha}$. In a nutshell, let us consider that output variables of this production *black box* are the variables $R^{t}_{f,u,r}$.

6.5.2 Generalization of the proposed production planning model without transportation time

Based on production building boxes defined in chapter 4, we generalize our production planning model to be able to capture different production units and transformation lines. What is required for a better understanding of forthcoming paragraphs is to keep in mind input and outputs of this building block.

General model for industrial and logistic pattern

We can divide the cost function into three main parts: production, transportation and inventory costs.

$$C^{obj}_{P} = \sum_{f\in\mathcal{F}_P}\sum_{u\in\mathcal{U}_f}\left(\left(\sum_{\omega}\sum_{\alpha}\sum_{t^{[\omega]}} c^{C}_{f,u,w,\alpha}\times w^{t^{[\omega]}}_{f,u,w,\alpha}\right)+\left(\sum_{r\in\mathcal{P}^{u}_{R}}\sum_{t} c^{P}_{f,u,r}\times R^{t}_{f,u,r}\right)\right) \quad (6.1)$$

$$C^{obj}_{T} = \sum_{l}\sum_{\psi\in\Psi_l}\sum_{t} c^{T}_{\psi}\times d_l\times N^{t}_{l,\psi} \quad (6.2)$$

$$C^{obj}_{I} = \sum_{f\in\mathcal{F}_I}\sum_{s\in\mathcal{A}_f}\sum_{a\in\mathcal{P}^{s}_{A}}\sum_{t}((c^{s}_{a}.((1+\eta)^{\frac{\delta t}{365}}-1))+h_f.\delta t)\times\frac{I^{t-1}_{s,a}+I^{t}_{s,a}}{2} \quad (6.3)$$

Finally we can write down the MILP corresponding to our model:

$$\min\left(C^{obj}_{P}+C^{obj}_{T}+C^{obj}_{I}\right) \quad (6.4)$$

$$\forall f\in\mathcal{F}_I, \forall s\in\mathcal{A}_f, \forall a\in\mathcal{P}^{s}_{A}, \forall t \qquad I^{t-1}_{s,a}+\sum_{l\in L^{in}(f)}\sum_{\psi\in\Psi_l} X^{t}_{l,\psi,a}-\sum_{l\in L^{out}(f)}\sum_{\psi\in\Psi_l} X^{t}_{l,\psi,a} = I^{t}_{s,a} \quad (6.5)$$

[1]in currency unit (such as €)

$$\forall f \in \mathcal{F}_I,\ \forall s \in \mathcal{A}_f,\ \forall a \in \mathcal{P}_A^s \qquad I_{s,a}^N \geq I_{s,a}^0 \tag{6.6}$$

$$\forall f \in \mathcal{F}_I,\ \forall s \in \mathcal{A}_f,\ \forall t \qquad \sum_{a \in \mathcal{P}_A^s} I_{s,a}^t \leq C_s \tag{6.7}$$

$$\forall f \in \mathcal{F}_I,\ \forall t \qquad \sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \leq F_f^{in} \tag{6.8}$$

$$\forall f \in \mathcal{F}_I,\ \forall t \qquad \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \leq F_f^{out} \tag{6.9}$$

$$\forall f \in \mathcal{F}_I,\ \forall t \qquad \sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t + \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \leq F_f^{all} \tag{6.10}$$

$$\forall f \in \mathcal{F}_K,\ \forall a,\ \forall t \qquad \sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} X_{l,\psi,a}^t - \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} X_{l,\psi,a}^t = Q_{f,a}^t \tag{6.11}$$

$$\forall f \in \mathcal{F}_P,\ \forall a,\ \forall t \qquad \sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} X_{l,\psi,a}^t - \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} X_{l,\psi,a}^t = \tag{6.12}$$

$$\sum_{u \in \mathcal{U}_f} \Bigg(\sum_{r \in (\mathcal{P}_R^u \text{ s.t. } a \in \mathcal{T}_{u,r})} \frac{w_{u,r,a} \times R_{f,u,r}^t}{\rho_{u,r}} - \sum_{r \in (\mathcal{P}_R^u \cap \mathcal{P}(a))} R_{f,u,r}^t \Bigg)$$

$$\forall l,\ \forall \psi \in \Psi_l,\ \forall t \qquad \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \leq C_\psi \times N_{l,\psi}^t \tag{6.13}$$

$$\forall \psi \in \Psi_l,\ \forall t \qquad \sum_{l} N_{l,\psi}^t \leq \mathcal{N}_\psi^M(t) \tag{6.14}$$

$$\forall l,\ \forall \psi \in \Psi_l,\ \forall t \qquad \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \leq w_{l,\psi}^M \times \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \tag{6.15}$$

$$\forall l,\ \forall \psi \in \Psi_l,\ \forall t \qquad \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \geq w_{l,\psi}^m \times \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \tag{6.16}$$

$$\forall l,\ \forall \psi \in \Psi_l, \forall a \in \mathcal{P}_A^\psi,\ \forall t \qquad X_{l,\psi,a}^t \leq w_{\psi,a}^M \times \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \tag{6.17}$$

$$\forall l,\ \forall \psi \in \Psi_l, \forall a \in \mathcal{P}_A^\psi,\ \forall t \qquad X_{l,\psi,a}^t \geq w_{\psi,a}^m \times \sum_{a \in \mathcal{P}_A^\psi} X_{l,\psi,a}^t \tag{6.18}$$

$$\forall f \in \mathcal{F}_I, \forall a \in \mathcal{P}_A \setminus (\cup_s \mathcal{P}_A^s),$$

$$\sum_t \Big(\sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} X^t_{l,\psi,a} + \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} X^t_{l,\psi,a} \Big) = 0 \tag{6.19}$$

$$\forall f \in \mathcal{F}_P, \forall a \in \mathcal{P}_A \text{ s.t.} \nexists\, r \in (\cup_{u \in \mathcal{U}_f} \mathcal{P}^u_R) \cap (\mathcal{P}(a)),$$

$$\sum_t \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} X^t_{l,\psi,a} = 0 \tag{6.20}$$

$$\forall f \in \mathcal{F}_P, \forall a \in \mathcal{P}_A \text{ s.t.} \nexists\, r \in (\cup_{u \in \mathcal{U}_f} \mathcal{P}^u_R) \text{ s.t. } a \in \mathcal{T}_{u,r},$$

$$\sum_t \sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} X^t_{l,\psi,a} = 0 \tag{6.21}$$

In addition, for each production line $u \in \mathcal{U}_f$ on each production zone $f \in \mathcal{F}_P$, we add constraints according to the model developed in chapter 4 within variables $y^t_{f,u,w,i}$, $v^t_{f,u,w,\beta}$ and $w^t_{f,u,w,\alpha}$. We thus capture potential production by batch with sequence dependent or not set-up costs and times.

The objective function (6.4) is the minimization of the sum of variable production, inventory and transportation costs. Equation (6.1) specifies that production costs may be the sum of both changeover and fixed production costs. (6.2) underlines that transportation costs are computed by transportation resources and (6.3) details the inventory cost computation: we sum the physical storing costs and the financial cost of immobilizations.

Constraints (6.5) to (6.10) deal with inventory functional zones. First of all, (6.5) is nothing but inventory balance equations. Let us recall here that flows going out from a node are negative. When the final inventory of one product is not imposed, we ensure that our result is robust by enforcing that final inventory is greater than initial one. This constraint is naturally directly depending on applications and may evolve according to the industrial context of the optimization. Storage capacities are satisfied thanks to (6.7) and maximal product flows are forced through constraints (6.8) to (6.10).

Fundamental equations at product creation or consumption zones are (6.11) for known zones (customers for which $Q^t_{f,a} > 0$ and fixed production zones for which $Q^t_{f,a} < 0$) and (6.13) for unknown production zones. The later one forces that transformations consume corresponding raw materials with the given yield and produce finished products.

Finally, constraints (6.13) to (6.18) describe the structure of transportation resources. (6.13) links product flows and used units of resources while (6.14) limits the on-hand resources. (6.15) and (6.16) ensure that proportion of each product in each resource satisfies given specifications whereas (6.17) and (6.18) restrict the proportion of each resource on each link between two functional zones.

Constraints (6.19) to (6.21) only forbid product flows when they do not exist: firstly, an inventory zone may not receive non-stored products. Secondly, an unknown production zone do not receive unconsumed raw materials and do not create non-produced products.

We have noticed that introducing global constraints help our solver to solve the problem quicker. Thus, we often add the basic echelon constraints:

$$\forall a \in \mathcal{P}_A,\ \forall t < N, \qquad \sum_{f \in \mathcal{F}_I} \sum_{s \in \mathcal{A}_f} I^N_{s,a} - I^t_{s,a} =$$
$$\sum_{\tau=t}^{N} \left(- \sum_{f \in \mathcal{F}_K} Q^\tau_{f,a} + \sum_{f \in \mathcal{F}_P} \sum_{u \in \mathcal{U}_f} \left(\sum_{r \in (\mathcal{P}^u_R \cap \mathcal{P}(a))} R^\tau_{f,u,r} - \sum_{r \in (\mathcal{P}^u_R \text{ s.t. } a \in \mathcal{T}_{u,r})} \frac{w_{u,r,a} \times R^\tau_{f,u,r}}{\rho_{u,r}} \right) \right) \tag{6.22}$$

Customer aggregation

Following our research presented in chapter 3 (see section (§ 3.2)), we use a reasonable simplification for solving real-life tactical production planning issues. We decompose the global optimization model into a two-step optimization.

Firstly, we solve a MILP with production variables on a simplified logistic basis: to reduce the size of the problem, we use clustering methods within known non inventory zones (customers or fixed product sources) to create aggregated non inventory zones. Thus, we simplify in this first step the supply chain graph. Based on our results presented in section (§ 3.2.4), we use our heuristic method which is an hybrid one, mixing a greedy clustering process, a location-allocation algorithm (see [Coo64]) as well as a local optimization simulated annealing process. Of course, we use it on homogeneous set of functional zones.

By minimizing the sum of transportation costs in this simplified graph, production costs as well as inventory ones we obtain a production planning for unknown production zones. At the end of this step, we are able to transform these zones into specified non inventory zones with known productions.

In the second step, we solve another MILP based on the real-life logistic network by transforming former unknown production zones into deterministic dynamic consumption zones. We solve it by minimizing the sum of inventory and transportation zones in this realistic supply chain.

What is the impact of this simplification on optimization results? Before starting the algorithm, we first class known functional zones $\mathcal{F}_K$ by functional zone types. We obtain a partition[14] of the set into subsets $\mathcal{F}_K^\theta$.

First, we define arbitrarily a stop criterion. For instance, we would like to divide approximatively by κ the number of known functional zones $\mathcal{F}_K$. Then, we define the two parameters α and λ (defined in section (§ 3.2.4)), based on industrial data.

[14] $\mathcal{F}_K = \cup_\theta \mathcal{F}_K^\theta$ and $\forall \theta_1, \theta_2,\ (\mathcal{F}_K^{\theta_1}) \cap (\mathcal{F}_K^{\theta_2}) = \emptyset$

We apply the following procedure to the set of known functional zones:

1. Compute $\kappa_\theta = \left\lceil \frac{|\mathcal{F}_K^\theta|}{|\mathcal{F}_K|} \times \kappa \right\rceil$.
2. Set $k = 0$, $\alpha^{[k]} = \alpha$ and $\Phi^{[k]} = \mathcal{F}_K^\theta$.
3. Step k:
   (a) If $|\Phi^{[k]}| < \kappa_\theta$ Then GO TO (3). Else:
   (b) If it exists two zones f_1 and f_2 of $\Phi^{[k]}$ such as their distance to each other is not greater than $\alpha^{[k]}$, then: .
      - $\Phi^{[k]} = \Phi^{[k]} \setminus (f_1 \cap f_2)$.
      - We compute ϕ as the functional zone whose geographical zone is centered on the weber point of the union of weighted points of f_1 and f_2 and characterized by union of product flows of f_1 and f_2.
      - $\Phi^{[k]} = \Phi^{[k]} \cap (\phi)$.
      - We apply the simulated annealing procedure coupled with the location-allocation algorithm to the set $\Phi^{[k]}$ (see section (§ 3.2.4)).
      - GO TO (3b)
4. Set $\Phi^{[k+1]} \leftarrow \Phi^{[k]}$, $\alpha^{[k+1]} \leftarrow (\alpha^{[k]} \times \lambda)$ and $k \leftarrow k+1$. GO TO (2).
5. STOP. $\Phi^{[k]}$ is the aggregated set of known functional zones $\mathcal{F}_K^\theta$.

Based on a real-life data set containing 675 customers, we study the impact of such a simplification in Table (6.1). We define **the objective function as being the sum of transportation costs**, in order to see clearly the impact of aggregation on it. We aim at minimizing neither production nor inventory costs.

Each aggregation is characterized by the number κ of virtual customers used in the first step, giving a level of aggregation ρ. The higher this number, the higher the simplification of the global model. We compare it the case without any simplification in which we solve the global problem in a unique step ($\kappa = 675$ and $\rho = 1$).

The case we are dealing with is nothing but the real-life application of section (§ 6.7): the supply chain of Saint-Gobain Glass is composed by fifteen geographical zones containing both an inventory zone and a production zone (including at least one production line). Globally, five different jobs are defined by various attributes and sub-attributes, and we use them to model twenty nine different production lines. We aim at planning production on a yearly time horizon based on monthly time periods. We compare impact on transportation costs of different levels of aggregation in the first step of our simplification, depending on the number of final aggregated zones κ.

We underline the efficiency of the heuristic method we present in section (§ 3.2.4) by comparing for an identical level of customer aggregation ($\rho = 22.5$) the result whether we only use the basic greedy clustering method introduced in section (§ 3.2.4). It appears than the difference δ_c of the optimal transportation cost in the two successive steps is around five percent, and is more sensitive to the clustering method than to the level of aggregation. This highlights the interest of using a good heuristic method in order to aggregate customers into customer family classes.

κ	ρ	Transportation Cost		
		Step 1	Step 2	δ_c
30 (greedy clustering)	22.5	72 795 500	79 844 600	*8.8 %*
30	22.5	73 790 200	79 136 800	6.8 %
48	14	75 096 800	78 964 800	4.9 %

Table 6.1: Impact of customer aggregation on transportation costs

As explained before, we solve our MILP program using the best on-hand commercial solver, i.e. CPLEX 9.0 (see [ILOb]). To work following practical requirements, we limit the optimization of each step to 1000 seconds. In case optimality is not proved, we give the gap between the best lower bound and the best found integer solution.

Table (6.2) provides the size of the model (given by the number of variables and constraints ; the number of integer variables is specified between brackets), the computational effort and the gap if the solution is not proved to be optimal. On the one hand, integer variables (used in the production process modeling) only exist during the first step of the process, which appears to be the hardest step to solve optimally. However, even if we do not prove optimality, the final gap to the optimum at the end of the limited computational time is less than 1 %. On the other hand, the number of variables (almost two millions) in the second step is larger than in the first one, due to the real-life supply chain design. It appears that the second step of the optimization may require up to 1.5 Go of memory on a classical lab-top (*Dell Inspiron 4150*).

κ (Step)	Variables	Integer Var.	Constraints	CPU time	Gap
675	1 945 591	2 040	259 672	1 466	0.78 %
30 (1)	204 091	2 040	35 212	1 014	0.57 %
30 (2)	1 944 871	0	255 045	47	0 %
48 (1)	252 691	2 040	41 476	1 016	0.68 %
48 (2)	1 944 871	0	255 045	62	0 %

Table 6.2: Interest of the two-step decomposition in terms of computational effort

Finally, Table (6.3) compares both CPU times and costs according to the four methods. It clearly shows that the impact on the optimal transportation cost of the two-step decomposition is negligible (Δ_c less than 0.3%), even for an aggregation level greater than $\rho > 20$, whereas it simplifies significantly the computational effort (gain Δ_t around 30%).

κ	Transportation Cost	Δ_c
30	79 844 600	1.15 %
30	79 136 800	0.26 %
48	78 964 800	0.04 %
675	**78 935 500**	

Table 6.3: Impact of the two-step decomposition on the optimal solution

We may need to apply our model to a more operational decision level on transportation operations in which we need to capture more precisely product flows. At this point, demand and production are assumed to be deterministic and dynamic over a short time horizon divided into small time periods. For instance, we would like to take into account improvements such as transportation times. Appendix (D.2) presents slight improvements of our model that tackle such an operational issue.

6.5.3 Implementation based on Object programming

From a practical point of view, we present in Appendix (D.1) how we did develop the ROADEO software. We use the `C++` code and follow the object programming fashion. However, we aim at underlining what we consider as original of this implementation, i.e. the great modularity of the code which can capture a wide panel of optimization problems.

Figure (6.3) presents how we do include within different classes (see section D.1) the different variable and constraint objects (based on the formalism of CPLEX, see [ILOa]) that will constitute the optimization model. Thus, once we have defined the supply chain, the model is already defined and start being solved. This very evolution-friendly way of programming has ensured that various versions of the model and all applications, including PLANEO presented in chapter 4, have been solved using an unique understandable code.

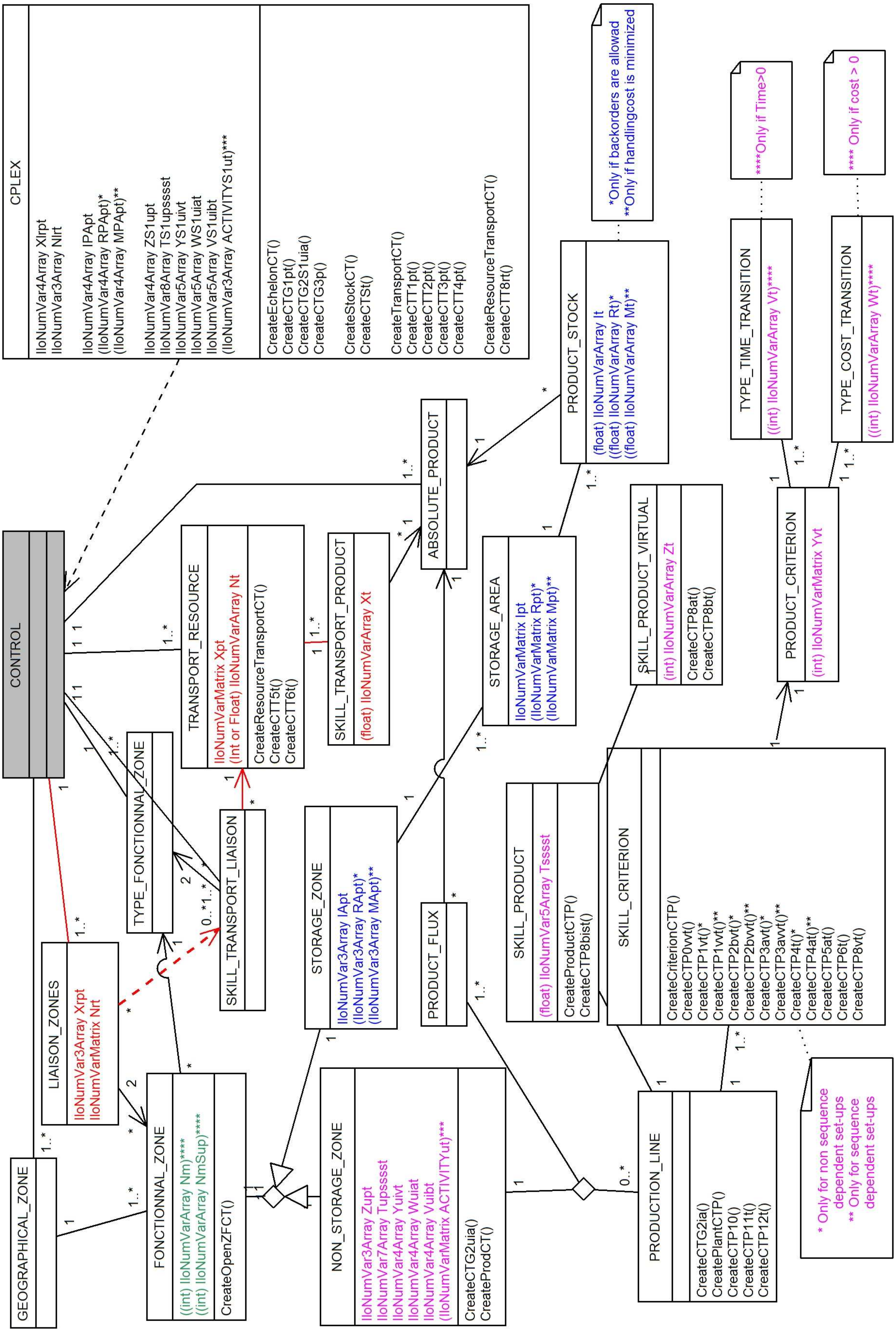


Figure 6.3: UML structure of ROADEO, linking each object to the model variables and constraints

The model presented in this part is perfectly adapted for tactical or operational applications, such as tactical production planning or operational logistic flow determination. Next part deals with the introduction of strategic topics, such as facility location.

6.6 A powerful location model and tool for industrial and logistic facilities

Let us now focus on strategic decisions such as studying facility openings and closings. In a first step, we will consider that potential facilities are given by the user, and that we just need to decide whether or not we open it. Furthermore, we will then propose a methodology to create from scratch potential facilities: in this case we determine its location.

6.6.1 Best facility locations among potential ones

Data

In this paragraph, each functional zone may have an additional parameter, which is a Boolean one indicating whether or not it physically exists. Thus, the user may differentiate between existing and potential facilities. We denote $\mathcal{F}^P \subset \mathcal{F}$ the set of potential functional zones. Following the reasoning of (6.5.1), we introduce the following notation:

- $\mathcal{F}^P_{\mathsf{K}}$ is the set of potential known production or consumption zones.
- $\mathcal{F}^P_{\mathsf{P}}$ is the set of potential unknown production zones.
- $\mathcal{F}^P_{\mathsf{I}}$ is the set of potential inventory zones.

We assume that these there subsets constitute a partition of $\mathcal{F}^P$.

For the potential facilities, we define opening costs as fixed costs which depend on the size of the facility. For a given functional zone $f \in \mathcal{F}^P$, we may define a list of fixed costs $\{c^F_{f,1}, \dots, c^F_{f,k}, \dots, c^F_{f,K_f}\}$ by associating to each one a range of product flow $[F^m_{f,k}, F^M_{f,k}]$. Of course, these ranges are all disjoints to each other ; here we assume without loss of generality that $\forall k > 1,\ F^M_{f,k-1} = F^m_{f,k}$. This notion of product flow is in our case (but this is an arbitrary decision) the mean daily outgoing product flow, except for customers for which we take the mean entering flow.

Finally, we introduce new constraints on the maximal number of opened potential functional zone. For each functional zone type τ (see discussion in (6.4.1)), we denote $\mathcal{F}(\tau)$ the corresponding set of zones and O^{τ}_{M} the maximal number of opened zones.

Variables and Model

Of course, we need to introduce new Boolean variables. For each potential zone $f \in \mathcal{F}^P$, we introduce:

- $O_{f,k}$ equals one whether we decide to open this zone and its optimal outgoing flow belongs to the k^{th} interval whose cost is $c^F_{f,k}$.
- $O^s_{f,k}$ equals one whether the outgoing flow is not greater than $F^M_{f,k}$ and 0 otherwise. By convention, $O^s_{f,0} = 1$ whether the flow is strictly non negative (we note $F^M_{f,0} = 0$) and 0 otherwise..

Of course, we may now extend our former MILP to capture new variables. Let us write down a list of new constraints.

$$C^{obj}_{O} = \sum_{f \in \mathcal{F}^P} \sum_{k=1}^{K_f} c^F_{f,k} \times O_{f,k} \tag{6.23}$$

$$\min\,(C^{obj}_{P} + C^{obj}_{T} + C^{obj}_{I} + C^{obj}_{O}) \tag{6.24}$$

$$\forall f \in (\mathcal{F}_I^P \cup \mathcal{F}_P^P), \quad \sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} \sum_{s \in \mathcal{A}_f} \sum_{a \in \mathcal{P}_A^s} \sum_t X_{l,\psi,a}^t + \sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} \sum_{s \in \mathcal{A}_f} \sum_{a \in \mathcal{P}_A^s} \sum_t X_{l,\psi,a}^t \leq \infty \times \sum_{k=1}^{K_f} O_{f,k} \tag{6.25}$$

$$\forall f \in \mathcal{F}_K^P, \ \forall a, \ \forall t, \ \text{s.t. } Q_{f,a}^t > 0,$$

$$\sum_{l \in L^{in}(f)} \sum_{\psi \in \Psi_l} X_{l,\psi,a}^t \leq Q_{f,a}^t \times \sum_{k=1}^{K_f} O_{f,k} \tag{6.26}$$

$$\forall f \in \mathcal{F}_K^P, \ \forall a, \ \forall t, \ \text{s.t. } Q_{f,a}^t < 0,$$

$$\sum_{l \in L^{out}(f)} \sum_{\psi \in \Psi_l} X_{l,\psi,a}^t \leq -Q_{f,a}^t \times \sum_{k=1}^{K_f} O_{f,k} \tag{6.27}$$

$$\forall f \in \mathcal{F}^P, \ \forall k \in [0, K_f],$$

$$\frac{\sum_{l \in F^{out}(f)} \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} \sum_t X_{l,\psi,a}^t}{N \times \delta t} - F_{f,k}^M \leq \infty \times (1 - O_{f,k}^s) \tag{6.28}$$

$$\forall f \in \mathcal{F}^P, \ \forall k \in [0, K_f],$$

$$F_{f,k}^M - \frac{\sum_{l \in F^{out}(f)} \sum_{\psi \in \Psi_l} \sum_{a \in \mathcal{P}_A^\psi} \sum_t X_{l,\psi,a}^t}{N \times \delta t} \leq \infty \times O_{f,k}^s \tag{6.29}$$

$$\forall f \in \mathcal{F}^P, \ \forall k \in [1, K_f], \quad O_{f,k} = O_{f,k}^s - O_{f,k-1}^s \tag{6.30}$$

$$\forall f \in \mathcal{F}^P, \quad \sum_{k=1}^{K_f} O_{f,k} \leq 1 \tag{6.31}$$

$$\forall \tau, \quad \sum_{f \in (\mathcal{F}^P \cap \mathcal{F}(\tau))} \sum_{k=1}^{K_f} O_{f,k} \leq O_M^\tau \tag{6.32}$$

To solve the strategic location problem, we create a MILP based on a mix of former and new constraints. Of course, we add the opening costs (6.23) to the objective function.

To deal with inventory zones, we still use former constraints (6.5) to (6.10). They ensure inventory balance, storage capacity and maximal product flow constraints. The same way, we keep (6.13) to model unknown production zone. However, we add constraints (6.25) to make sure that there is no flow going through closed facilities.

On the contrary, we transform former constraints (6.11) into new ones (6.26) and (6.27). This slight change forces that there is no flow whether the known zone is closed.

Finally, we keep both constraints (6.13) to (6.18) describing the structure of transportation resources and constraints (6.19) to (6.21) forbidding nonexistent product flows.

To give a sense to variables $\mathsf{O}_{\mathsf{f,k}}$ and $\mathsf{O}^{\mathsf{s}}_{\mathsf{s,k}}$, we introduce new constraints (6.28) to (6.32). (6.28) and (6.29) force $\mathsf{O}^{\mathsf{s}}_{\mathsf{f,k}}$ to be 1 whether the outgoing flow is not greater than $\mathsf{F}^{\mathsf{M}}_{\mathsf{f,k}}$. (6.30) links variables $\mathsf{O}_{\mathsf{f,k}}$ to $\mathsf{O}^{\mathsf{s}}_{\mathsf{s,k}}$, and (6.31) forces that a facility opening cost may not be counted twice for two disjoint activity ranges. The last constraints (6.32) satisfy the maximal number of opened potential facilities by functional zone type.

In this part we have studied how to model strategic decisions such as choices of opening or closure of potential existing facilities. This clearly imply that users have a good idea of potential locations because they have been thinking about it for a while. In the next part, we propose a methodology to start from scratch.

6.6.2 Facility location from scratch

Based on our notions of *functional zone* introduced in (6.4.1), we have imagined the concept of *virtual zone*. Based on it, we have developed an algorithm using our

MILP that solve industrial size problems.

Concept of virtual zone

So far, we have divided all our functional zones into different types, which are based on two main families: inventory zones and others. For cases in which user would like to determine optimal locations of a given functional zone type, we have created the concept of virtual zone. It is characterized exactly as another functional zone, except that it does not belong to any geographical zone.

Thus, we may define through the virtual zone exactly what kind of facility we aim at locating. It may be either a given type of inventory zone or a production zone as well as a customer.

Method and algorithm

Of course, the MILP presented in (6.6.1) may be used as soon as we have geographical positions of all functional zones and thus all distances and costs.

Based on the virtual zone description, we are thus going to define arbitrarily geographical positions for new potential functional zones identical to the given virtual one.

To do so, we have based our approach on an intuitive heuristic: we create a grid of the studied geographical zone (the convex hull of all existing geographical zones) with an initial (given or not) precision γ_M. Then, we apply the following iterative procedure until working with the given accuracy γ_m. This later parameter is very useful because in most cases we are working on international industrial and logistic network and thus we do not need to determine optimal locations for new facilities with a very high accuracy.

We use the following notation:

- We define the grid function $G(\gamma, \{[x_m, x_M], [y_m, y_M]\})$ that gives a set of points which are griding the rectangle $[x_m, x_M] \times [y_m, y_M]$ by squares of size γ.

- For any set of points $P \in \mathbb{R}^2$, we denote $F(V, P)$ the set of potential functional zones whose structure is similar to the given virtual zone V and locations are points of P.

- For any subset of potential functional zones $\mathcal{J} \subset \mathcal{F}$, we denote $\mathsf{Solve}(\mathcal{F}, \mathcal{J}) \in \mathbb{R}^2$ the set of geographical points of zones $\mathcal{J}$ on which potential facilities are open in the optimal solution of the MILP presented in (6.6.1) applied to the global set of functional zone $\mathcal{F}$.

Based on a given set of known functional zones $\mathcal{F}$, we aim to determine locations of a given virtual zone V with an accuracy γ_m. First, we compute the studied geographical zone ranges $[x_{min}, x_{max}]$ and $[y_{min}, y_{max}]$.

1. . We set $\gamma^{[k]} = \gamma_M$ and create the sets of points $P^{[k]} = G(\gamma^{[k]}, \{[x_{min}, x_{max}], [y_{min}, y_{max}]\})$ and $O^{[k]} = \emptyset$. We build the set of functional zones $\mathcal{F}^{[k]} = \mathcal{F} \cup F(V, P^{[k]})$. Set $k = 1$.

2. Step k:

   (a) Compute $O^{[k]} = \mathsf{Solve}(\mathcal{F}^{[k]}, F(V, P^{[k]}))$ the set of opened localized virtual zones.

   (b) If $\gamma^{[k]} < \gamma_m$ then GO TO (3), Else:

   (c) $\gamma^{[k+1]} = \frac{\gamma^{[k]}}{2}$

   (d) Create the set of points $P^{[k+1]} = \cup_{\{x,y\} \in O^{[k]}} \left(G(\gamma^{[k+1]}, \{[x - \gamma^{[k+1]}, x + \gamma^{[k+1]}], [y - \gamma^{[k+1]}, y + \gamma^{[k+1]}]\}) \right)$.

   (e) Generate the set of functional zones $\mathcal{F}^{[k+1]} = \mathcal{F} \cup F(V, P^{[k+1]})$.

   (f) $k \leftarrow k + 1$. GO TO (2).

3. The set $O^{[k]}$ contains optimal locations for creation of virtual functional zone.

If we focus on the step (2d) of our procedure, it appears that each open facility during the step k is translated into 9 new potential ones in the following step $k+1$. On the one hand, it appears that this factor is too important in practice, because the more potential zones, the more integer variables in the MILP, and thus the more difficult to solve it. On the other hand, this method allows us to cover the whole space of the concerned area. As a consequence, we propose another griding method that decreases the factor to 5 by keeping the later property.

We introduce two new griding functions:

- Firstly, we create a square around a given point $G_1(\gamma, \{x, y\}) = \{\{x, y\}, \{x - \gamma, y - \gamma\}, \{x + \gamma, y - \gamma\}, \{x - \gamma, y + \gamma\}, \{x + \gamma, y + \gamma\}\}$.
- Secondly, we create a $45°$ rotated square around a given point $G_2(\gamma, \{x, y\}) = \{\{x, y\}, \{x - \gamma, y\}, \{x + \gamma, y\}, \{x, y - \gamma\}, \{x, y + \gamma\}\}$.

And we modify the procedure by introducing a new variable $\mathsf{Grid}^{[k]}$. We initialize $\mathsf{Grid}^{[0]} = 1$ and modify the step (2d): at step k:

- If $\mathsf{Grid}^{[k]} = 1$ then $P^{[k+1]} = \cup_{\{x,y\} \in O^{[k]}} \left(G_1(\gamma^{[k+1]}, \{x, y\}) \right)$ and $\mathsf{Grid}^{[k+1]} = 2$.
- If $\mathsf{Grid}^{[k]} = 2$ then $P^{[k+1]} = \cup_{\{x,y\} \in O^{[k]}} \left(G_2(\gamma^{[k+1]}, \{x, y\}) \right)$ and $\mathsf{Grid}^{[k+1]} = 1$.

Using this trick we do not degrade results but we accelerate the procedure.

Figures (6.4) to (6.7) illustrates this griding process. We imagine an industrial and logistic network in which we aim at locating a given virtual functional zone. We only plot potential virtual zones. Starting from an initial griding with step μ_M, we compute four iterations before stopping with $\mu^{[4]} < \mu_m$. At each step, red circles highlight best locations after optimization (sets $O^{[k]}$).

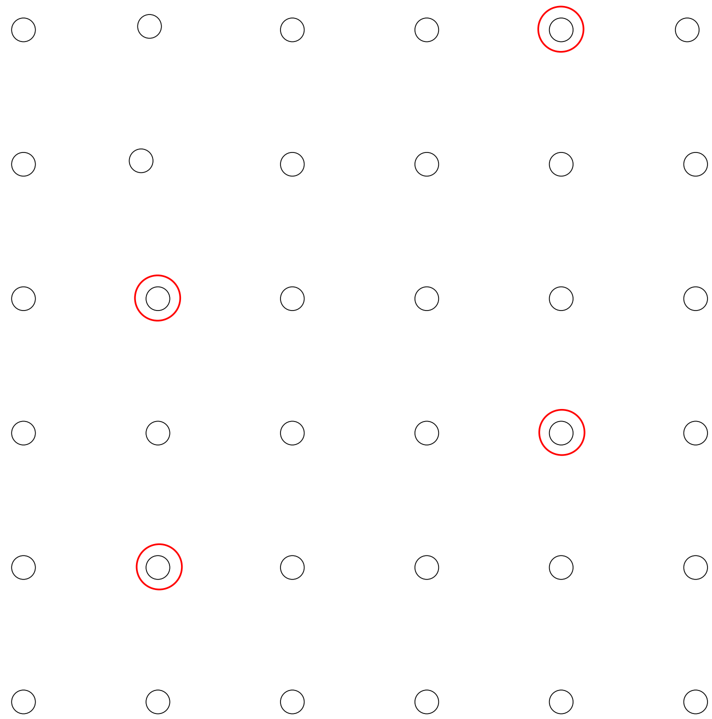


Figure 6.4: Result of the first step based on an initial discretization grid with accuracy μ_M.

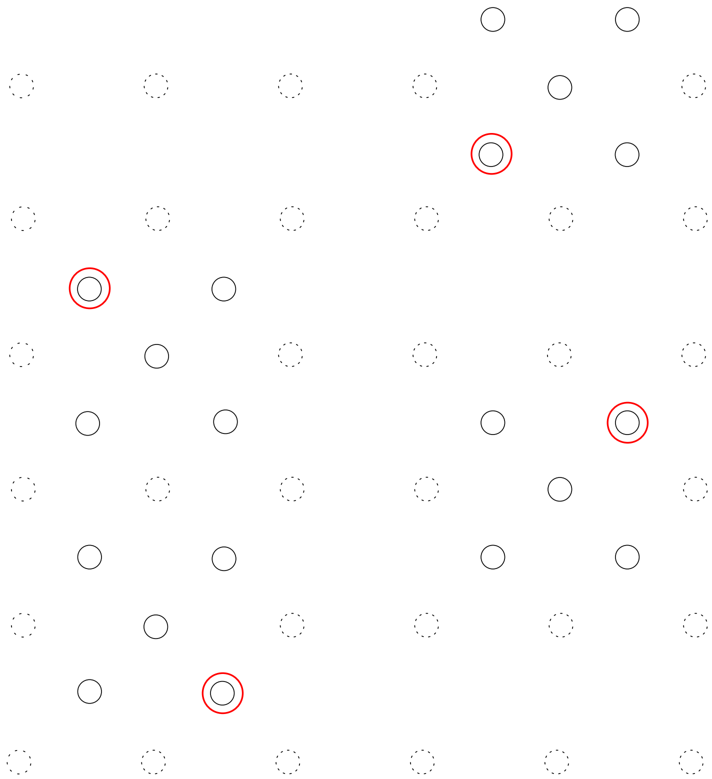


Figure 6.5: Result of the second step.

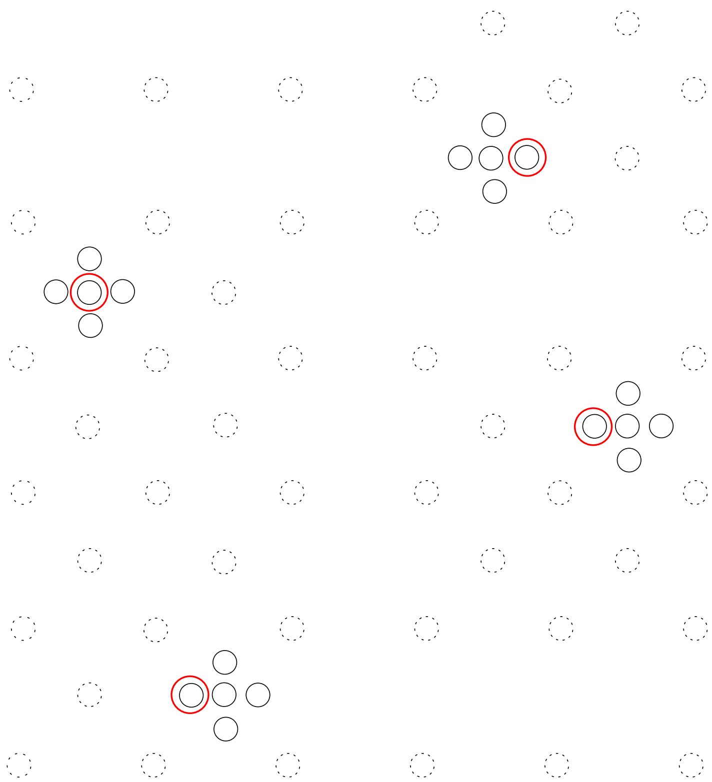


Figure 6.6: Result of the third step.

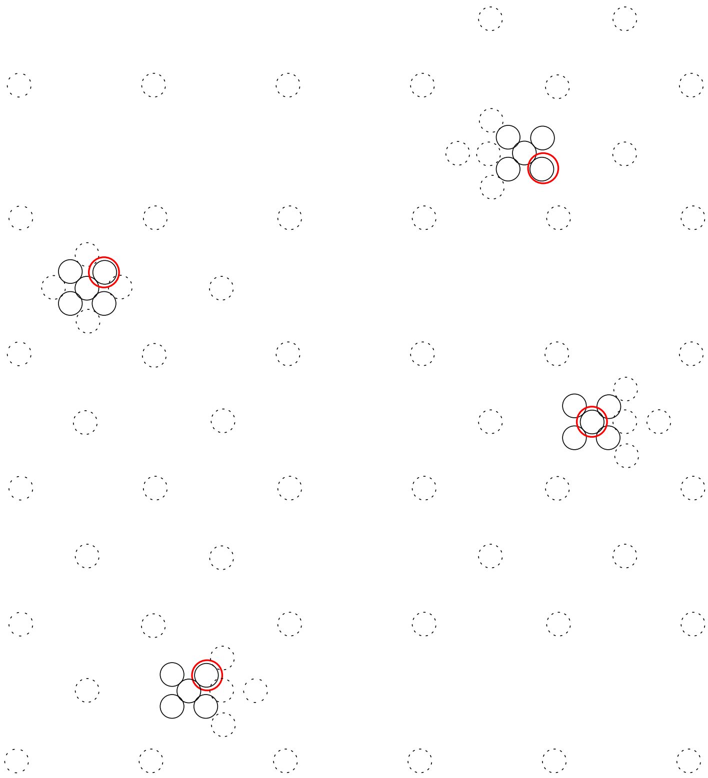


Figure 6.7: Final result, with accuracy of at least μ_m.

6.7 The tactical production planning in the glass manufacturing industry

Remembering that we have pursued this research for an industrial partner, we propose here to present the direct application of our work in the glass manufacturing industry. We use the production planning model developed in chapter 4 for modeling colour production processes of several plants and minimize the sum of transportation, inventory and production costs in our integrated model.

The company owns a dozen float glass plants in Europe. As explained in our process description in chapter 4, the most important characteristic of glass is its colour. To switch from one color to another, changeovers take on average several days of lost production. Thus, color production planning is computed twice a year according to a rolling horizon fashion, based on yearly demand forecasts of each customer. We explain how we did it and what we earned using our decision support tool.

6.7.1 A crucial factor in operating flat glass plants

In sections (§ 1.3) and (§ 1.2) we have described precisely the products and their corresponding processes of Saint-Gobain Glass.

Planning operations at a flat glass plant is complicated by a number of factors, including the necessity to plant for several hundreds of product types. The glass is produced in several colors, depending on additives that are included in the basic mixture (made of silica sand, etc.) molten in the large gas-fired furnace. Each colour is declined in nearly thirty thicknesses, three quality levels (depending on the defaults distribution identified by an automatic inspection equipment at the end of the line), several packaging modes, and various dimensions.

The single most important consideration in planning flat glass operations is the transition schedule, that is, the scheduling of production times for different colors. For instance, changing colour from clear to a tint (such as bronze, gray, etc.) results in up to eight days production of off-color glass that can not be sold as finished product. Changing from one tint to another results in two to four days lost production. Because of these huge transition losses, the plants naturally schedule long colour runs that take as much as a year to complete a full color cycle. The sequence of colors at a plant is generally unalterable because of the inventory levels held from previous color cycles. However, deciding the duration of colour runs is critical since poor choices can result in stock-outs in a colour that may not be produced again for many months.

Most of the plants are only able to produce clear glass. However, some particular plants are able to produce several tints.

6.7.2 Production planning prior to ROADEO

Prior to the development of ROADEO, a corporate production planner was responsible for generating the European colour production plans for each of the floats based on the marketing forecasts developed for each customer. This was a two-step plan: as a first step, the planner did choice manually the tint campaigns on competent plants (around seven plants) by taking into account their availability (plants may be rebuilt during several months as explained in section (§ 1.3)). As a second step, he or she used a commercial optimization software to dispatch the production of clean glass according to remaining capacities of plants (around twenty units), by minimizing the transportation costs.

The planner was working at a very aggregate level: a product was defined by a colour. Despite this usual way of planning colour campaigns, he (or she) envisioned various major benefits from building an integrated model based on ROADEO.

First of all, a more precise definition of products was mandatory: each plants has its own skill ranges of colors and thicknesses. A better level of aggregation appeared thus to define a product family by a colour and a thickness, to ensure an accurate assignment of products to plants.

Of course, the creation of an integrated model provided also the ability to efficiently incorporate additional plants into a coordinated production and distribution system. The same way, this is a means through which the occasionally conflicting views of various functional areas (marketing, manufacturing, transportation and so forth) could be examined and resolved.

Last but not least, we suspected savings (in time as well as in money) from planning on a one-step system-wide basis to be important.

In a nutshell, managers did not find a commercial software able to minimize simultaneously the sum of production costs due to both the production and the huge sequence-dependent set-ups, the transportation costs, and the inventory costs. We used ROADEO to present a prototype.

It is interesting to draw the parallel between our research and the one presented by Martin and al. in [MDE93]. Authors have developed what could be the forerunner of ROADEO, which they called FLAGPOL. In this model, they address a similar problem they face at Libbey-Owen-Ford, a company operating as an autonomous operating company of the Pilkington Group (one the main competitor for Saint-Gobain Glass).

The FLAGPOL model is a specialized version of a production, distribution, and inventory model. They also specify the structure and some technological factors unique to the particular business of glass. However, they chose to develop a model that may be defined as an iterative one, pretty close to the one used by the planner of Saint-Gobain Glass as the second step of its planning: they specify as a transition schedule parameter the number of days available each month by float by colour. In a word, they specify the colour planning and do not capture set-up costs as variables, whereas ROADEO generalize this approach by defining the production planning based on the model developed in chapter 4.

Authors in [MDE93] describe precisely the implementation process and their interaction with practitioners, which is in a way comparable to the one we have known so far. They underline that it took more than two years to start having practical results and accurate insights on real data cases. Whereas we did face less difficulty in recovering data[15], we did not achieve yet such results. Our research is still on going and six months of work were not enough to complete our applications. However, the team of Saint-Gobain Recherche is still working on this project and we enjoy very encouraging results: despite the very large size of our model, the commercial solver we use (Cplex, see [ILOa]) provides so far optimal results.

6.7.3 Using ROADEO as a tactical production planning tool

Illustration on a virtual case

In this part, we provide an illustration of how ROADEO can be used for planning colour campaigns. However, for confidentiality reasons, we use virtual values for all parameters and a very simplified example without transformation lines issues.

As explained above, a product family is defined by a given colour. We could integrate thickness and quality features to capture constraints on transformation lines (for instance, mirror lines requires the highest quality) by adding another sub-attribute, but we do not aim at modeling a real problem here.

For each plant of the industrial pattern, we define its availability over the time horizon and its skill set in terms of colors. Using our production planning framework (see chapter 4), we define products by one attribute (colour). Users specify that the time period is one month. Thus, our time horizon is divided into $\mathsf{N} = 12$ time periods of duration $\delta\mathsf{t} = 30$ days.

For each plant, we have the set of colours that it is possible to produce, corresponding capacities -fixed value or a range-, and for each colour we define the list of

[15] see our discussion in section (§ 1.5) about the power of Enterprise Resource Planning systems

thickness ranges that we may produce with bounded proportion. Changeovers within colors are specified using factorized types of time and cost. Given the time period size which is much bigger than set-up times, we only consider sequence-dependent set-up costs here.

We define the fixed existing industrial and logistic pattern: we deal with a tactical industrial decision. Each plant is located on a specific geographical zone which also contains a glass warehouse, that is, two particular functional zones (production and inventory).

Global variable production costs being equal in every plant, we only take into account changeover costs. We also work with the set of located customers whose deterministic demand forecasts give us the consumptions of each product family during each time period over the time horizon.

In the forthcoming example, we work with five glass colors. Figure (6.8) plots the whole set of customers by specifying who consumes which color. Plants are also plotted. Figure (6.9) illustrates the global demand of each colour during each time period. Finally, Table (6.4) gives the colour skills of each plant. Of course there is an obvious correlation between the consumption of a colour and the number of plants being competent to produce it. To simplify this example, we consider that all changeovers within colors are possible and that each changeover cost is fixed and equals 150 000 units.

Plant | **Colour**	1	2	3	4	5
1			yes	yes	yes
2			yes	yes	yes
3	yes	yes	yes	yes	yes
4					yes
5	yes		yes	yes	yes

Table 6.4: Skills of plants

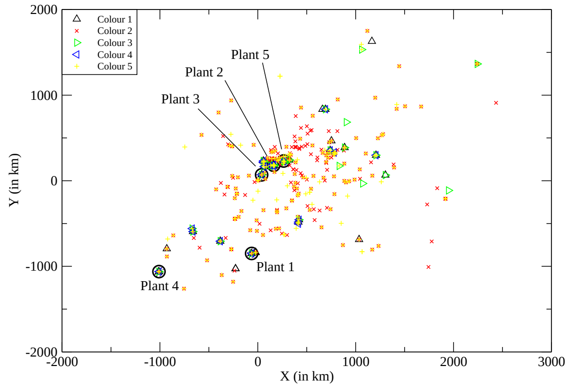


Figure 6.8: Map of customers and consumed colors. Position of plants

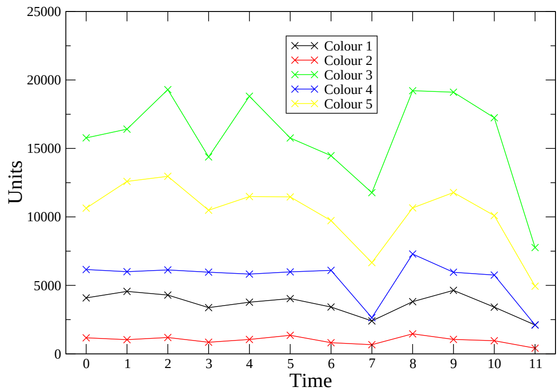


Figure 6.9: Global demand by colour over the time horizon

It clearly appears that each color has its specific characteristics: Colour 3 is the most consumed one by the number of consumers is highly limited. On the contrary, colour 2 represent the smallest sales but nearly all customers are concerned. We can summarize this remark by Figure (6.10) which classifies colors depending on their sales and number of customers.

Finally we define a unique transportation resource, glass dedicated trucks (also called inloaders), whose costs capture the fact that each delivery is billed for the whole round trip. Of course, this model mainly focuses on the production planning, and thus we do not use discrete transportation resource, but linear ones[16].

[16] variables $N^t_{\iota,\psi}$ are real

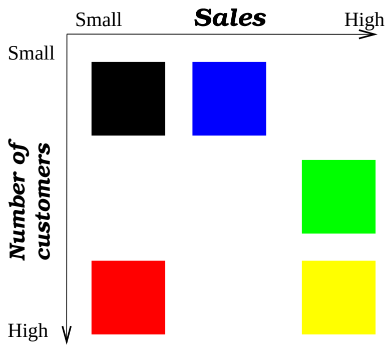


Figure 6.10: Colors classed by their sales and number of customers

We aim to minimize the sum of production $\mathsf{C}_{\mathsf{P}}^{\mathsf{obj}}$, inventory $\mathsf{C}_{\mathsf{I}}^{\mathsf{obj}}$ and transportation $\mathsf{C}_{\mathsf{T}}^{\mathsf{obj}}$ costs. Intuitively, the less changeovers we do in plants, the cheaper the production costs but the the higher the inventory and transportation ones. Knowing that changeover costs appear to be much less important than others, we guess that the fundamental trade-off lies in the minimization of both inventory and transportation costs.

In this part, due to confidentiality reasons, we do not use neither real initial inventory levels nor real costs and do not specify everything. On the five plants we authorize production breaks. In each product of each storage area, we just force that final inventory level be greater than initial one.

To understand the interest of our model, we optimize successively seven hypotheses: at each time, we change the objective function. Table (6.5) summarizes the costs of each optimal or pseudo optimal (we force a maximal computation time) solutions. The goal of this comparison is that global optimization gives a much cheaper result than others. We clearly see that key costs to master are inventory ones and then transportation ones.

For each hypothesis, we obtain the optimal colour planning with associated product flows and inventory levels over the time horizon. For instance, Figure (6.13) gives the color production planning for hypothesis 7 whereas Figure (6.14) gives corresponding global (we sum over all inventory zones) inventory levels by colour.

Hyp	Objective function	C_P^{obj}	C_I^{obj}	C_T^{obj}	Total
1	$\min(C_P^{obj})$	150 000	50 992 400	55 413 900	106 556 300
2	$\min(C_I^{obj})$	2 400 000	20 257 500	54 629 700	77 287 200
3	$\min(C_T^{obj})$	2 400 000	125 207 000	10 270 300	137 877 300
4	$\min(C_P^{obj} + C_I^{obj})$	1 200 000	20 437 300	50 471 700	72 109 000
5	$\min(C_P^{obj} + C_T^{obj})$	1 050 000	147 141 000	10 311 000	158 502 000
6	$\min(C_I^{obj} + C_T^{obj})$	3 150 000	21 403 200	18 768 900	43 322 100
7	$\min(C_P^{obj} + C_I^{obj} + C_T^{obj})$	2 400 000	21 385 900	18 863 700	**42 649 600**

Table 6.5: Costs of different optimizations

Many differences between solutions may be underlined. We notice intuitive results.

For cases in which we do not minimize transportation costs (hypotheses 1, 2 and 4, see blue costs), they are clearly raised by tremendous amount of indirect flows: each plant sends its production to distant warehouses (see Table (6.6)). Whether we only minimize production changeover costs we observe that minimal changeover costs are obtained when production of plants is steady. As much as possible plants are dedicated to as less as possible products. On the contrary, whether we minimize only inventory costs illustrates that there are numerous changeovers but no key plant.

Hypothesis	Objective function	Proportion of indirect flows[1].
1	$\min(C_P^{obj})$	60 %
2	$\min(C_I^{obj})$	67 %
3	$\min(C_T^{obj})$	0 %
4	$\min(C_P^{obj} + C_I^{obj})$	62 %
5	$\min(C_P^{obj} + C_T^{obj})$	0 %
6	$\min(C_I^{obj} + C_T^{obj})$	0 %
7	$\min(C_P^{obj} + C_I^{obj} + C_T^{obj})$	0 %

Table 6.6: Indirect flows

On the contrary, as soon as we minimize transportation costs it appears that plants 2 and 5 are mainly used to fulfill demand: this underlines that these two plants

[1] for which the distance between the producer is not the shipping plant

are located better than others to serve the market. Another interesting remark lies in the huge inventory costs that appear whether we minimize transportation ones without taking into account inventory ones (hypotheses 3 and 5, see green costs in Table (6.5)).

Among all solutions of transported flows, colour 2 does no difference: it is always produced by the unique competent plant[17] and sent to customers trough the corresponding warehouse. Let us focus on other colors and try to understand what change from one solution to another.

Let us try to understand the underlying framework of distribution in each hypothesis. To do so, we study for each colour during each time period which is the main provider of each customer. Of course, we only focus on cases in which we minimize transportation costs: otherwise, product flows are meaningless.

We discover **two types of distribution**. On the one hand, hypotheses 3 and 5 in which we minimize transportation costs (resp. alone and with production changeover costs) are characterized by a fixed distribution pattern: each customer is served by the same warehouse during every time period. In these cases, transportation represents the key cost: to minimize it, every plant makes to stock each one of its skills to fulfill the demand according to its capacity of a constant customer set.

On the other hand, as soon as we take into account both inventory and transportation costs (hypotheses 6 and 7), we obtain a new type of solution: the distribution rules clearly depend on the time period, that is the production planning. These cases prove that under our cost assumptions, inventory costs are the most important ones and drive the distribution structure: to avoid increasing the stock level, production campaigns directly serve the demand of all customers at this time.

To illustrate this important structure difference, we base our example on the deliveries of colour 5. We justify this choice due to its particular characteristics:

[17] plant 2

it is consumed by many customers in important quantities (see Figure (6.10)). We plot on Figure (6.11) (resp. Figure (6.12)) the fixed distribution pattern of this colour in the hypothesis 3 (resp. hypothesis 5) solution which does not depend on the time period. On the contrary, we plot some time dependent delivery solutions of hypotheses 6 and 7: our choice is arbitrary and aims at illustrating the evolution of the solution depending on time. Figures (6.15) to (6.26) represent the evolution of colour 5 product flows in the hypothesis in which we optimize all costs (hypothesis 7).

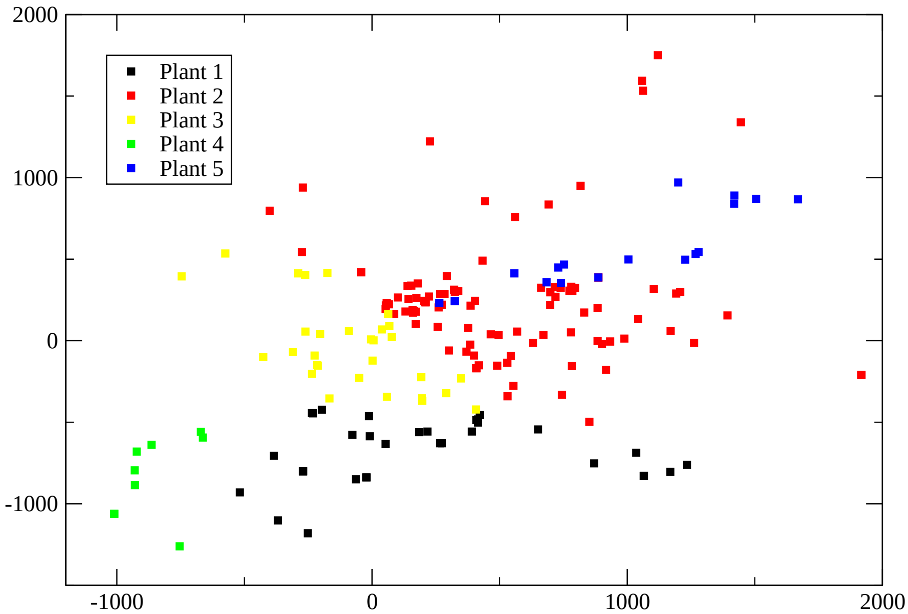


Figure 6.11: Static distribution pattern of colour 5 in the transportation cost minimization (hypothesis 3)

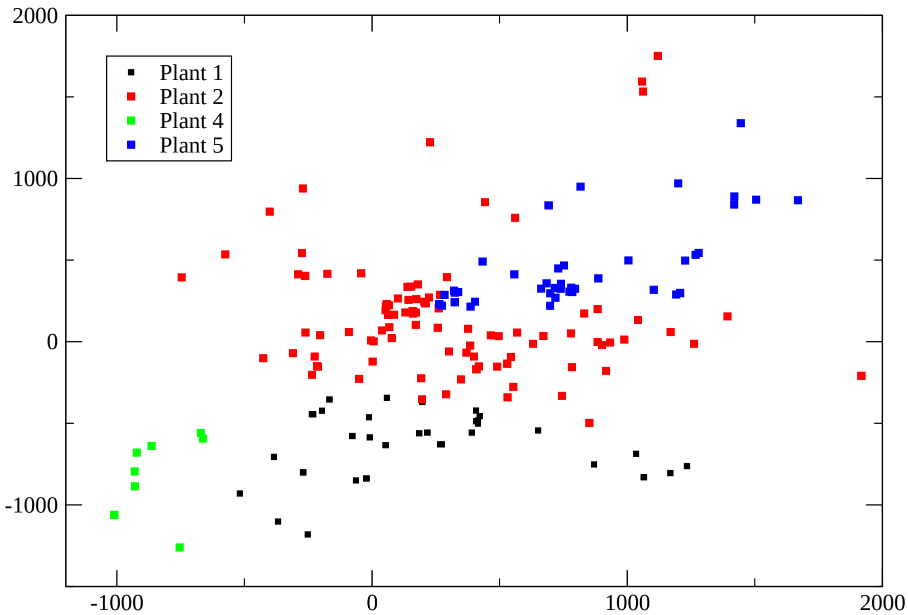


Figure 6.12: Static distribution pattern of colour 5 in the minimization of both production and transportation costs (hypothesis 5)

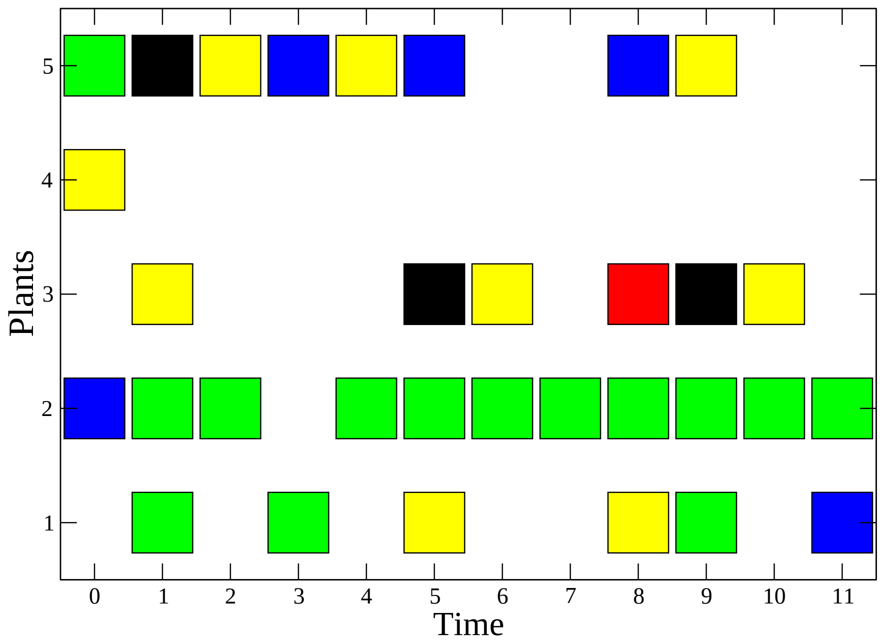


Figure 6.13: Production planning corresponding to the minimization of all costs.

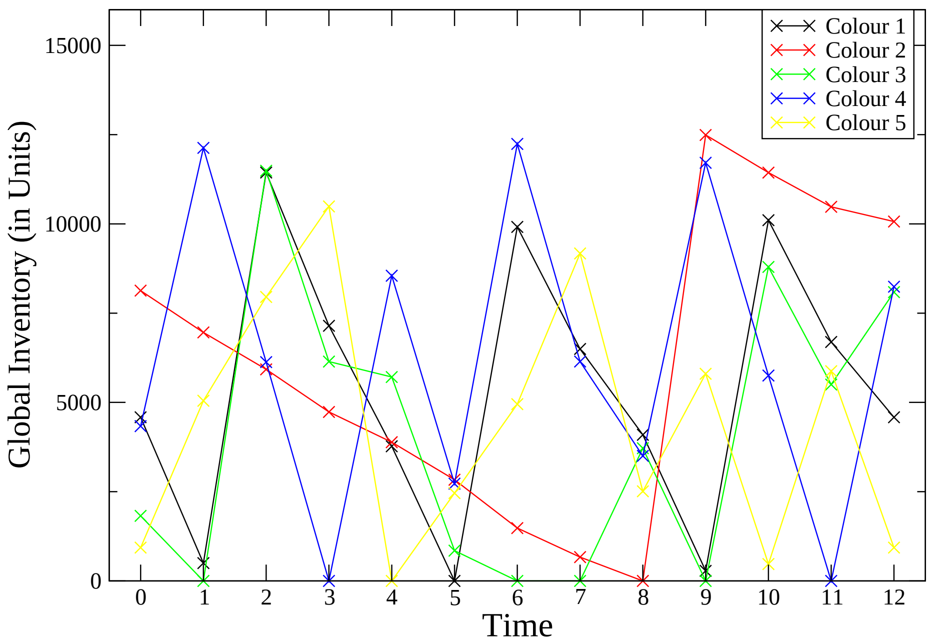


Figure 6.14: Inventory levels corresponding to the minimization of all costs.

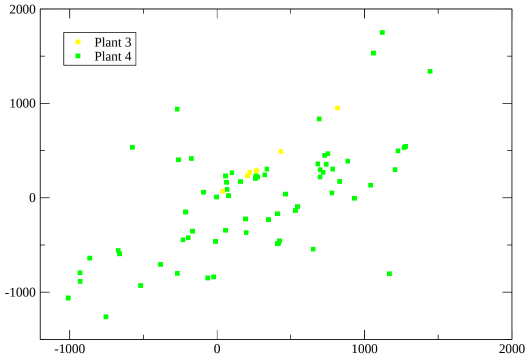


Figure 6.15: Hypothesis 7 ; colour 5, period 0

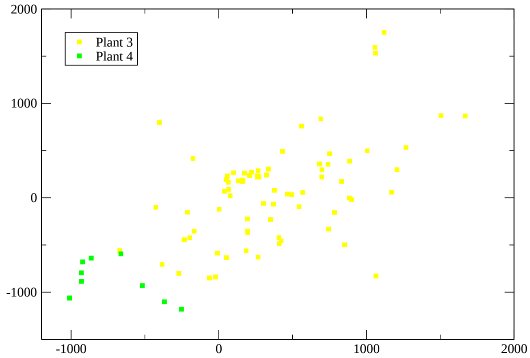


Figure 6.16: Hypothesis 7 ; colour 5, period 1

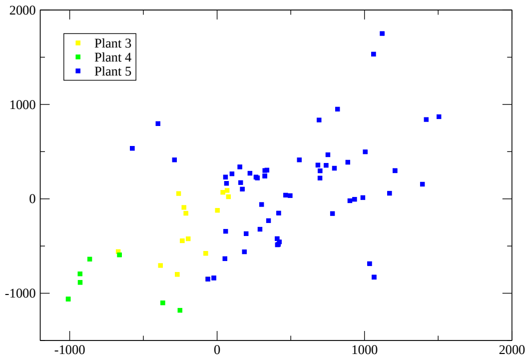


Figure 6.17: Hypothesis 7 ; colour 5, period 2

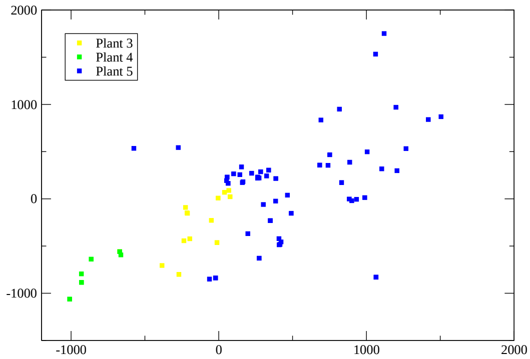


Figure 6.18: Hypothesis 7 ; colour 5, period 3

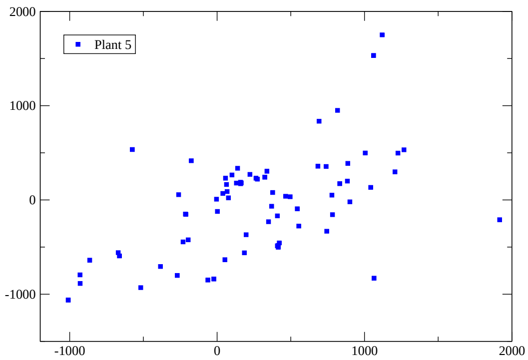


Figure 6.19: Hypothesis 7 ; colour 5, period 4

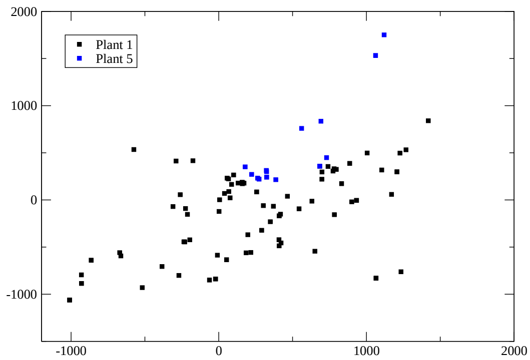


Figure 6.20: Hypothesis 7 ; colour 5, period 5

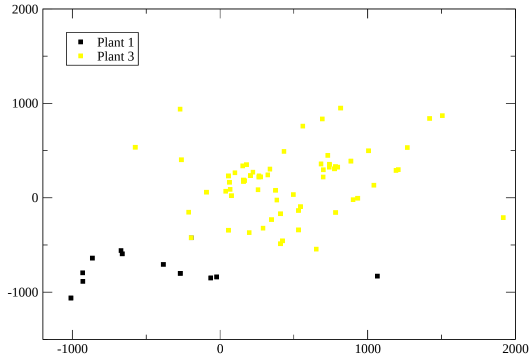


Figure 6.21: Hypothesis 7 ; colour 5, period 6

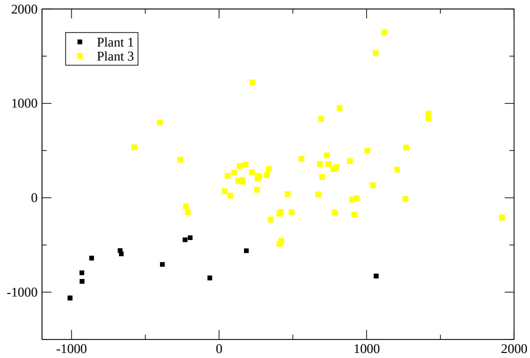


Figure 6.22: Hypothesis 7 ; colour 5, period 7

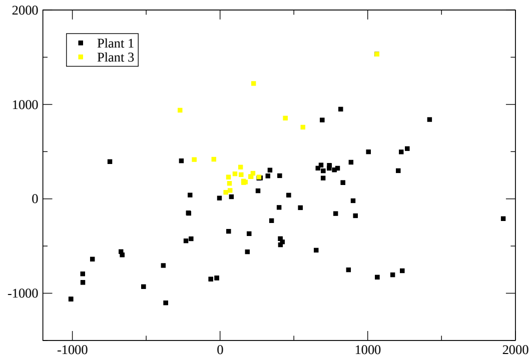


Figure 6.23: Hypothesis 7 ; colour 5, period 8

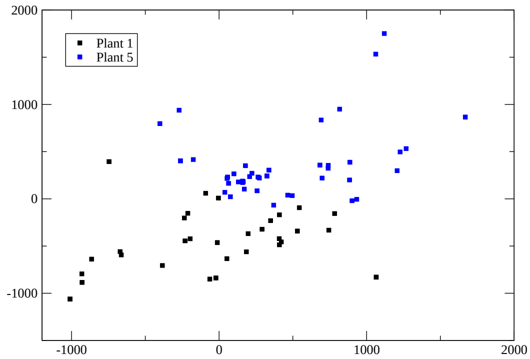


Figure 6.24: Hypothesis 7 ; colour 5, period 9

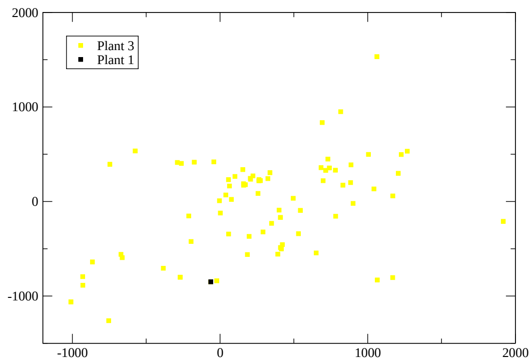


Figure 6.25: Hypothesis 7 ; colour 5, period 10

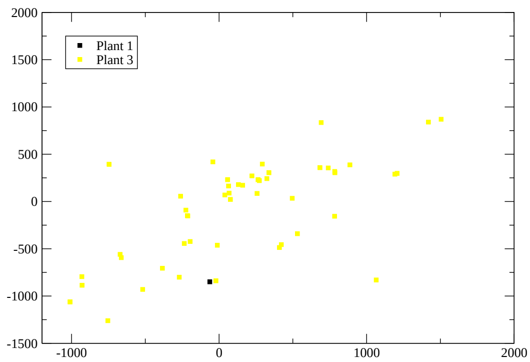


Figure 6.26: Hypothesis 7 ; colour 5, period 11

Thus in our virtual example under wrong cost assumptions, our model highlights that inventory costs are more important than transportation ones, while both of them are more important than changeover ones.

We may remark that this conclusion could have been deduced from Table (6.5), by looking at the individual optimal costs (colored in red). It appears clearly that

$$C_P^{obj^*} < C_T^{obj^*} < C_I^{obj^*}$$

. We thus obtain in the global optimum solution the corresponding individual gaps given by Table (6.7).

Cost	Local optimum		Global optimum		Gap
	Hypothesis	Value	Hypothesis	Value	
C_I^{obj}	2	20 257 500	7	21 385 900	6 %
C_T^{obj}	3	10 270 300	7	18 863 700	84 %
C_P^{obj}	1	150 000	7	2 400 000	1500 %
$C_I^{obj} + C_T^{obj}$	6	40 172 100	7	40 249 600	0.2 %
$C_P^{obj} + C_I^{obj}$	4	21 637 300	7	23 785 900	9 %
$C_P^{obj} + C_T^{obj}$	5	11 361 000	7	21 263 700	47 %

Table 6.7: Analysis of local and global optima

Outline of real-life cases

We may use ROADEO at different levels of aggregation, depending on the context. Here we introduce one real-life problem we faced and solved using ROADEO. Some fundamental costs such as fixed production costs are not taken into account here, for political reasons (the labor cost knows a high variability depending on countries).

First of all, we worked on a model derived from the one introduced previously. We consider five plants producing five colored tint for automotive market. They have various skills of production: each plant is able to produce a subset of tints with various specific transition costs. The same way, each plant has a specific range of thickness values. Extreme thickness values (either very thin or thick) are produced by few plants. To capture the global operation costs of producing these tints, we focus on the production planning of colors on a yearly time horizon divided into weekly time periods. We authorize idle periods on production lines.

We use a unique job, defined by one attribute, the colour, and a sub-attribute, the thickness family: we aggregate real thickness values into five ranges. In this model we do not use the customer aggregation simplification because their number is small enough to solve the global problem directly.

For each plant, we have the set of colour that it is possible to produce, corresponding capacities -fixed value or a range-, and for each colour we define the list of thickness ranges that we may produce with bounded proportion. Changeovers within colors are specified using factorized types of time and cost. Given the time period size which is much bigger than set-up times, we only consider sequence-dependent set-up costs here. According to the expert knowledge, we define six types of set-up costs.

Objective	Costs			
	Production	Inventory	Transportation	**GLOBAL**
C_P	**0**	6 103 790	72 365 000	78 468 790
C_I	17 100 000	**18 480**	69 426 000	86 544 480
C_T	58 590 000	28 074 200	**13 413 500**	100 077 700
$C_P + C_I$	0	75 225	72 092 000	72 167 225
$C_P + C_T$	4 320 000	57 680 400	13 488 100	75 488 500
$C_I + C_T$	56 700 000	2 931 640	14 529 600	74 161 240
$C_P + C_I + C_T$	8 460 000	5 886 260	17 318 800	**31 665 060**

Table 6.8: Costs of solution on a first real-life case

Figures (6.27) and (6.28) plots the results of global (we sum on all the fifteen products) production and inventory levels as function of time, whereas Table (6.8) gives the values of each cost in the best found solution after one hour of computation time. It is easy to determine which plant may need a capacity raise, which warehouse suffer from a tight capacity, etc.

These results with realistic costs prove that the integration of production, inventory and distribution is mandatory to obtain global interesting solutions. Whether we do not capture transportation flows, the corresponding cost literally explodes. On an homogeneous market, it is worth using transitions on each line to decrease both inventory and transportation costs.

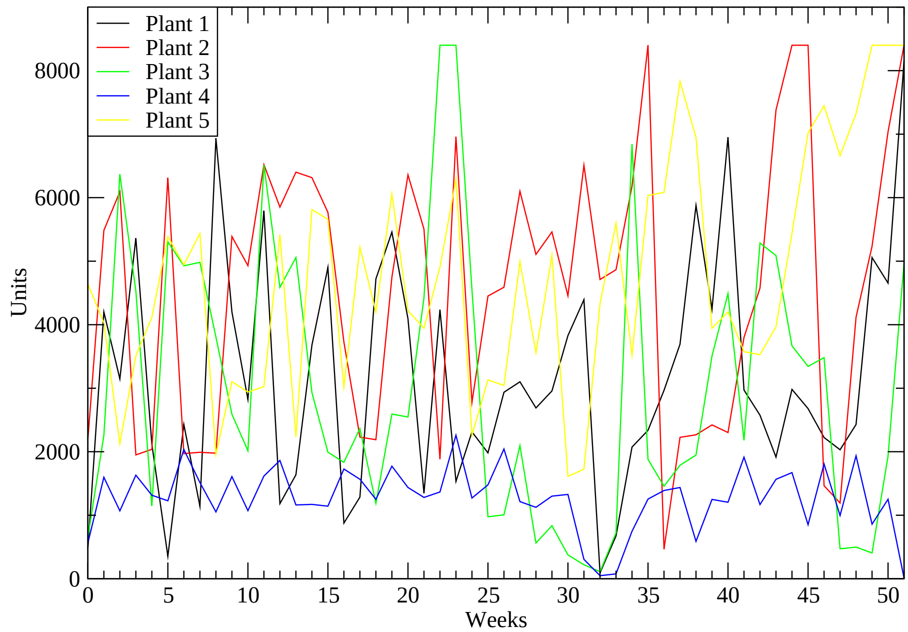


Figure 6.27: Production levels of the fifteen products as part of the optimization results

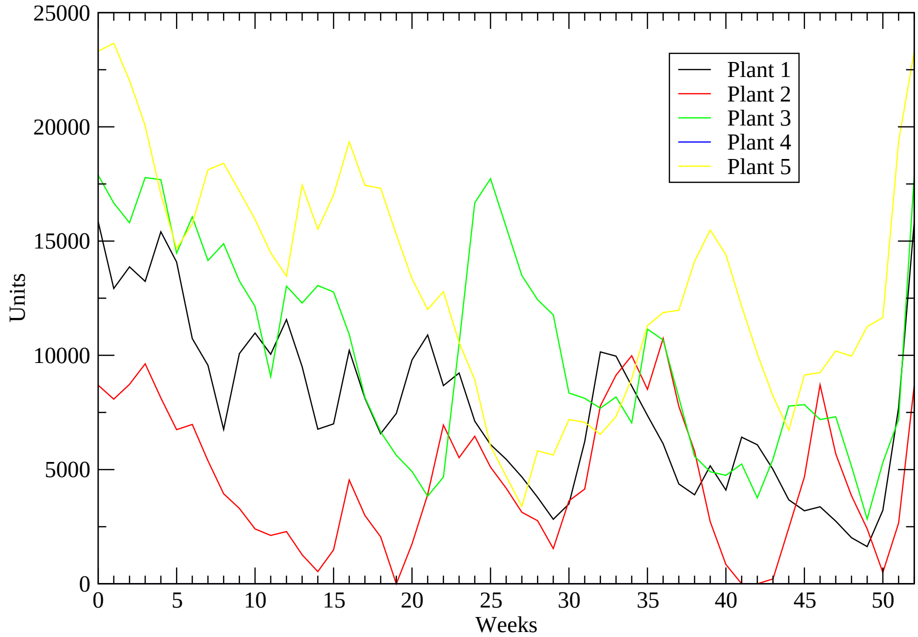


Figure 6.28: Inventory curves as part of the optimization results

6.7.4 The Flexicolor project: ROADEO applied to strategic process design

So far, from our introduction to the industrial context (section (§ 1.2) to ROADEO, one of the applications of our research at the Saint-Gobain Glass company, we have been working on the present float glass process.

The fundamental glass characteristic in this process is the colour because it depends directly on the composition of the melted mixture. As we saw in section (§ 6.7.1), transition times between two colour campaigns are huge and thus very costly not only in production costs but also in cycle inventory costs (due to long production campaign in a single colour).

A new technology may appear in the forthcoming years that would completely change the glass production planning issue. Based on new techniques of glass coloration, we may be able to change a clear (or any tint) glass produced continuously in a principal furnace into another tint just by adding some components melted in a secondary small furnace. Homogenization of the global mixture would be possible just before pouring glass onto the molten tin inside the float tank. The revolution comes from the fact that this new process (that we denote *Flexicolor*) would divide the transition times by a coefficient around ten. Obviously, the gain of flexibility on the process would have huge consequences on the supply chain operations cost.

Pyke and Cohen ([PC90]) have led an very interesting study on the impact of flexibility on global costs. More recently, a more general excellent literature review has been published by Bertrand ([Ber03]). Basically, a lack of flexibility to adapt the supply chain to emerging demand for various products frequently leads to lost sales for some products and product markdowns of excess inventory for other products. A drama may quickly happen for an industrial company which experiences an imbalance between demand uncertainty and supply chain flexibility.

Assuming that the basis of the flat glass process remains the float glass one, there is no flexibility in the amount of available capacity for production. It becomes thus crucial to create as much flexibility as possible in the timing and frequency of production. In this way, the Flexicolor process (illustrated by Figure (6.29)) would allow the glass manufacturer to decrease lead times for introducing new products variants, generating thus much less inventory.

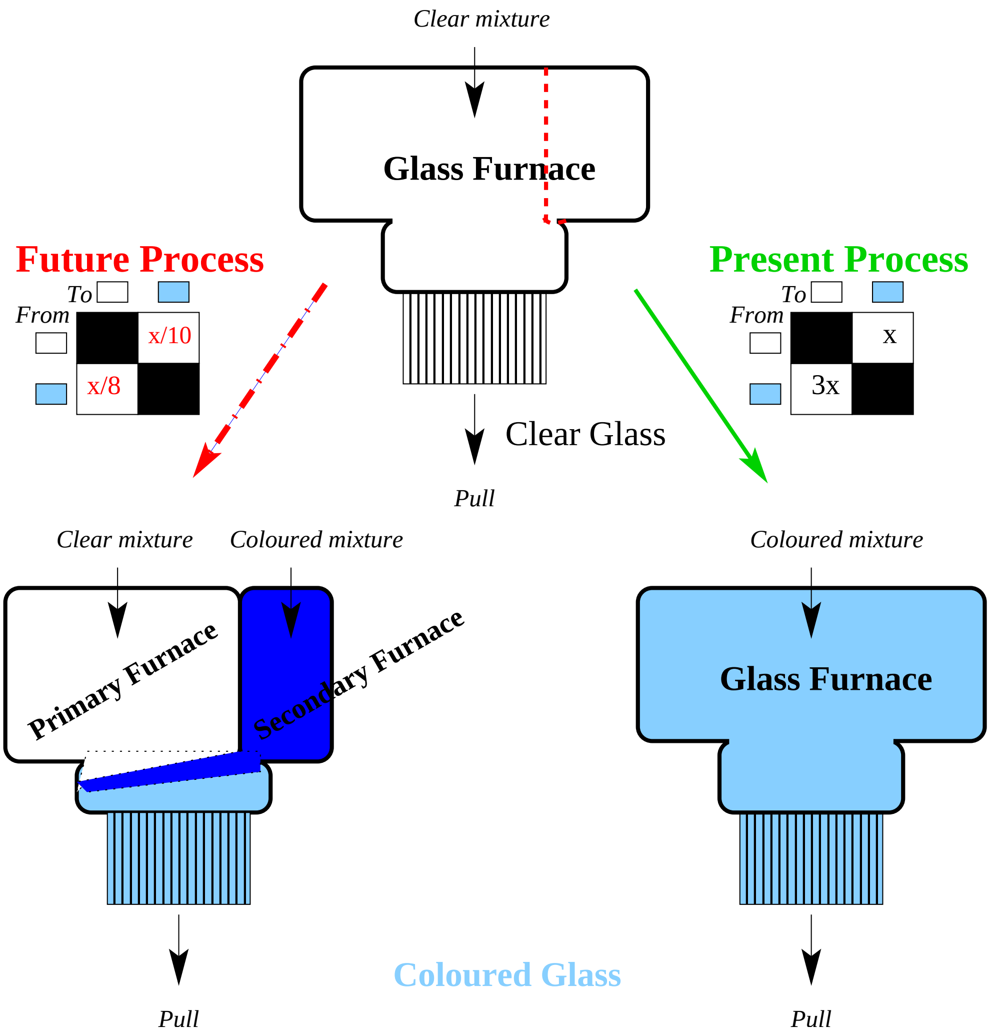


Figure 6.29: Illustration of a possible revolution in the coloration process

Comparing advantages and drawbacks of the two processes and defining the best one is nothing but a strategic industrial issue. We are going to illustrate how ROADEO may be a very powerful decision tool on such a question, at least as a first step in which we work under deterministic assumptions. We propose to compare different tactical optimizations to integrate them in a strategic thought.

To solve the real-life colour production planning issue requires to model the whole supply chain of Saint-Gobain Glass and minimizing the sum of production, inventory and transportation costs. We need to capture four jobs: float glass production and its three main transformations, i.e. laminating, soft-coating and mirror lines.

On the one hand, in the present situation we aim at determining all colour campaigns at once on float plants by fulfilling both basic and transformed product demands. Based on yearly demand forecasts for all customers (675, as in section (§ 6.5.2)) in Europe, we work on a yearly time horizon divided into monthly time periods.

At this tactical level of decisions, we define aggregated product families by a pair of values, one being the colour and the other one the state of transformation. The float process is defined by an attribute, the colour and a sub-attribute, the hard-coating state, while transformations are based on a unique attribute, the state (working or not).

On the other hand, assuming that the Flexicolor process is possible, we modify the float production job: as a first step, colour becomes simply a sub-attribute, because it is possible to produce several colors in a given time period, our discrete time being much bigger than new changeover times. In both of the models we capture that a certain amount of capacity will be lost in lower levels of decision by changeover between within colour, thickness and width values.

This new job of flat glass production based on two sub-attributes creates a new deep change into our former results. For instance, based on the same real-life data set than before, we optimize the production by minimizing the sum of global costs. Table (6.9) summarizes the results. It highlights that such a process evolution may be the source of savings of more than 10% of the global yearly variable cost. Of course, fundamental costs such as fixed production costs are not taken into account here, for political reasons (the labor cost knows a high variability depending on countries). The relative gain may be thus studied before making an hasty decision. As expected, most important savings are made on holding costs on which we save nearly 80%. The average level of inventory over the year falls from more than a year of production days to two months and a half. In addition, we note a gain of more than 2.5 millions of euros on transportation costs which is relatively less impressive but is still important. This gain mainly comes from the fact that colour campaigns are not a constraint any more for delivering each customer from the closest competent plant. This gain is however still constrained by capacity limits.

Process	Costs			
	Production	Inventory	Transportation	**GLOBAL**
Classical	4 770 000	3 236 130	81 739 200	**89 745 330**
Flexicolor	0	657 076	78 973 000	**80 260 897**

Table 6.9: Impact of the process evolution on the operations costs

This first step on the global supply chain gives us an upper bound on the potential gain on inventory and transportation costs. However, it is wrong to consider that changeover costs are null with the new process, because they are simply divided by a coefficient around ten.

In the same way, some indirect consequences of such a decision may become huge drawbacks. For instance, as we explained it in our introduction (§ 1.2), the melted mixture that is put continuously inside the furnace contains a high percentage of cullet, which is some recycled crashed glass. A fundamental data of the process is that we need use some cullet whose colour must be identical to the produced glass ribbon in order not to disturb the production. Presently, cullet inventory levels may be considered as infinite because there is no imbalance in the present process: the quantity of cullet produced unintentionally (side losses, changeovers, non-tolerated quality default ribbon parts, etc.) is greater than replenishment needs. This comfortable situation may be disturbed by the future Flexicolor process. The principal furnace with the unchanging reference tint would produce a wide range of other tints, but no self-adapted cullet. A deficit of reference tint cullet may become a limit to the process flexibility by either forcing some reference tint campaigns only destinate to fulfill the replenishment needs of the principal furnace or requiring longer campaign of reference tint and thus more cycle inventory of other tints. So far, we did not capture this issue in our model. In a second step, we integrate cullet products as absolute products into the ROADEO model in order to determine whether this phenomenon is critical or not.

To use ROADEO in this on-going study about the cullet, we only need to define absolute products corresponding to different tint cullets, and to transform each production of float glass into a transformation consuming the corresponding cullet: we denote $\mathsf{t}\%$ the percentage of cullet in the raw material mixture. In the Flexicolor process, we suppose that the proportion of required cullet in the primary tint is bigger than the one in the secondary cullet. We denote the ratio of primary tint in the consumed $\frac{1}{2} \leq \theta \leq 1$. To capture the production of cullet in the model, we need to add a sub-attribute to the float process, that is the state of the glass, broken or not. Breaking the glass ribbon creates some cullet of the produced colour. Even if we do not want to produce cullet, there is a minimal percentage of the production which is some (few percent of the pull), due to side loss, bad quality, changeovers within thickness and width values, etc. We denote this percentage which cannot be reduced $\mathsf{c}\%$. Thus, ROADEO is flexible enough to capture the cullet issue and to study whether it may become a costly problem.

On a single plant producing four tints, we obtain some very interesting results. In a realistic case in which $\mathsf{t} = 15\%$, $\theta = \frac{2}{3}$ and $\mathsf{c} = 8\%$, we compute three main data sets, based on constant and identical demand for all tints. The case A corresponds to the present process: the plant produces four tints and the cullet cycle is not a constraint because $t > c$. We introduce a primary colour in cases B and C from which the three other tints are secondary, obtained by the Flexicolor process. Each production of a secondary tint requires $\theta \times t\%$ of primary cullet and $(1 - \theta) \times t\%$ of secondary one.

The case B is a case in which we divide by ten the transition costs within all tints. Finally, the case C is the ideal Flexicolor process case in which there is no changeover cost any more and no discrete time (colour is a sub-attribute). We work on a yearly basis divided into three-day-long time periods, so that the discrete time is not a constraint neither in case A nor in case B. We assume there is no transportation costs and we only minimize changeover costs and holding costs.

In the ideal case C in which we do not need any cycle inventory (at least for this level of product aggregation), but we notice the unwilling inventory creation of cullet in secondary tints. Obviously, since $(1 - \theta) \times t < r$, each production of a secondary tint creates some cullet that is useless for Saint-Gobain Glass. This effect is also present in case B, but we ignore it (by not counting inventory costs) in table (6.10) which compares the results of cases A and B. The impact of dividing changeover costs by ten is a reduction up to 58% of the sum of holding costs and changeover costs.

Process	Costs		
	Production	Holding costs	**GLOBAL**
A	4 760 000	6 432 410	**11 192 400**
B	2 724 000	1 997 570	**4 721 570**
GAIN	43%	69%	**58%**

Table 6.10: Impact of a changeover cost reduction

As a conclusion, we may consider that a new process which divides changeover costs by ten may be create huge savings: almost 60% of the sum of changeover costs and holding costs and few percent of the dominating cost, i.e. transportation one. Globally, this may represent several millions of Euros a year.

As a first step, ROADEO appears to be a flexible key tool for the top management. Of course, further research is required to determine the impact of the process evolution under uncertainty: the potential gain of shorter production campaigns on safety stocks may be another crucial element that our deterministic model completely ignores.

Basic strategic issues may also be tackled based on the location model we introduced in section (§ 6.6). We are currently working on a real-life application of this part of the model.

6.8 Conclusion and research perspectives

Starting from the Glass production process, we have developed in chapter 4 a general methodology to model a continuous process production planning. Based on a product-driven decomposition into attributes and sub-attributes, we provided a useful mixed integer program that capture different levels of hierarchical production planning.

In this chapter, we pave the path of our ongoing work on solving real-life problems of industrial and logistic issues. We integrate our precedent work as a building block in a general methodology that captures many industrial industrial and logistic patterns. Our framework covers production and transformation facilities as well as inventories and customers, in a deterministic environment. Flows of products within the supply chain are possible, based on transportation resources whose skills are specified by the user.

By minimizing production, inventory and transportation costs, we provide in a first step a powerful decision tool for both tactical industrial and logistic decisions. At this tactical level, we consider the supply chain design as known and fixed. For tactical industrial decisions, production facilities have to be planned, based on principles developed in chapter 4. We introduce our customer aggregation method to make this step possible on industrial size data set.

Furthermore, we extend our program to strategic decisions, such as facility location, etc. We propose a first method assuming that users have a set of potential identified locations and want to optimize both opening, production, inventory and transportation costs. Based on it, we present a more general method based on specifications of the type of desired facility that tries to determine optimal locations from scratch.

All this work is applied successfully to the Saint-Gobain Glass company, at different levels, highlighting the powerful insights that operations research tools may provide to the industry. As ROADEO includes[18] the PLANEO project introduced in chapter 4, many practical results have been obtained.

The model is currently used to develop reaction procedures in various situations, such as:

- Given demand forecasts and all plant skills, what is the global colour planning that minimizes production, inventory and transportation costs?
- What is the impact of supply chain costs of a new process? Is it worth investing on it?
- Is is cheaper to develop on-line transformations or to build off-line specific production lines?
- Whether we introduce a new transportation resources in the supply chain such as train, is it interesting to open non-producing logistic platforms?
- What is the best response on the tactical planning to an unforecasted event such as a critical production problem or a lower than anticipated yield?

[18]PLANEO uses the same code, and is designed to deal with unique geographical zone at the operational production planning level

- What are the optimal location and the skill portfolio for building a new production facility?
- Is it worth specializing the float plants (in terms of colour skills)?

This on-going research -new applications often requires tight modifications in the model- aims at creating a very evolution-friendly object program whose the underlying linear program may be solved in a reasonable time by on-hand commercial solvers, such as CPLEX. The interest of Saint-Gobain Glass to develop its own optimization tools lies in the fact that commercial softwares do not capture industrial structure and constraints of the particular glass manufacturing business.

Appendix A

Practical approach of logistic platform design

A.1 Practical Ideas and Prospects for managers at Saint-Gobain Glass

In chapter 1, we describe the industrial context of Saint-Gobain Glass. Section (§ 1.4) presents more specifically distribution issues of flat glass. Let us develop in this section ideas and prospects trying to fill in the gap between the theory exposed in chapter 3 and real-life issues.

Structure of demand

First of all, we aim at understanding what the underlying structure of demand is. In order to classify all sold products, we use a classical Pareto decomposition, as plotted on Figure (A.1). This method is of course questionable, but its main advantage is to be simple and useful as a first step.

In this analysis, we did not take into account references with less than three sold trucks (66 T) during the year, considering it as punctual references.

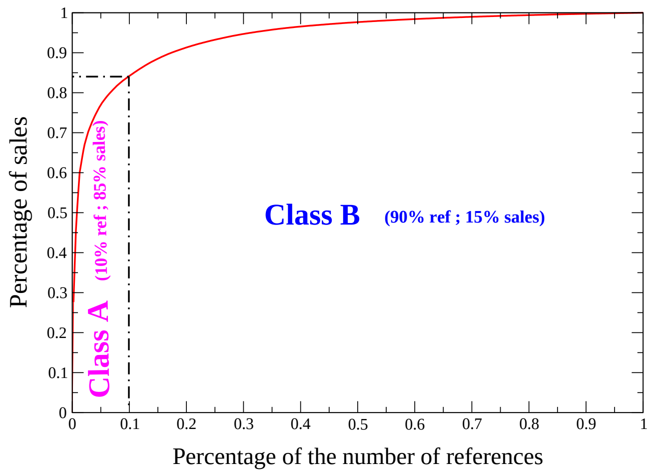


Figure A.1: Pareto decomposition of products according to their sales

Among the remaining 450 items, we show that 10% percent of the references correspond to 85% of the total amount of sales. We will denote them using the term of high volume items, called Class A products. Class A products are mainly untransformed float glass. Using the same idea, Class B products (also called low volume items) represent only 15% of the sales but 90% of the references.

It is not surprising to notice that the average number of factories able to produce a given product is much higher for the high volume products. We find that on average 4.3 different factories are able to produce each product A, whereas we only find 1.5 for products B. This gap is even more important (5.3 compared to 1.7) if we restrict this analysis to floated but untransformed products. This makes sense because most of plants are float plants.

Analysis of the mixed origin deliveries

If we analyze the global flows, we obtain Tables (A.1) and (A.2). Thus nearly 15% of the produced quantities are transferred between two plants before being sent to the customer. Surprisingly the level of mixed origin flows is higher for high volume products, whereas they are produced in more plants. Intuitively, it seems that the level of transfer on products B is weaker whereas level on products A is much higher than expected.

If we focus on more details, we can notice that *one shipping plant catches 60% of the mixed origin flows during the year*, and the explanation is easy: it was stopped during the year for being fixed. We discuss the impact of this phenomenon further (see remark (8) on page 282).

If we correct the data without taking into account this plant as a shipping plant, we find that products B are twice mixed origin as much as products A.

In addition their average real transfer distance (according to Table (A.2)) is bigger that the average of standard products, which seems coherent because the number of plants able to produce them is lower.

% of the indirect flow	with the plant being fixed	without it
Class A Products	0.15	0.07
Class B Products	0.12	0.10

Table A.1: Analysis of the weight of indirect flows

Average distance	Direct[2] Flow	Indirect[3] Flow	
Class of products	Producer → Customer	Producer → Shipper	Shipper → Customer
A	414	307	304
B	498	354	275

Table A.2: Analysis of the average distances of products flows

In a nutshell, we find that levels of mixed origin flows are low if we compare them to the proportion of mixed orders (nearly half of the total sales). However, we are pretty surprised by the important transfers of products A and and relatively limited transfers of B.

[2]see definition (9) on page 19
[3]see definition (10) on page 19

Let us try to focus on many possible non optimal phenomena which are at first sight perfectly invisible, such as:

- It is possible that transfers of products B appear low because they are often avoided by direct deliveries, although the order contains some standards products which could have been produced closer to the customer. This case is symbolized on Figure (A.2). We have two options to serve a mixed order of the customer:

  - Option 1: U_1 is the sender. We use a simple mixed order without any mixed origin product. A and B are produced in U_1 and directly sent to the customer C.
  - Option 2: U_2 is the sender. To do so, we replenish U_2 in product B which becomes a mixed origin product. The delivery is thus classified as a mixed origin one (following definition (6)) in our study.

  Depending on the value of the parameter p which represents the proportion of class B products in the mixed order, the cheapest solution may be either Option 1 or Option 2. These non trivial results are studied in part (A.3.2). It is possible that commercial people try not to use the Option 2 because there is presently no clear policy[1] of replenishment within plants.

- The high level of transfers of products A (on Table (A.1) we read that there is still 7% after correction of the Porz flow) may also be inducted by the utilization of Option 2 within the full truckload policy[2]. Each inloader must be full at any time. Thus, required transfers of some products B may be often filled with products A, as presented on Figure (A.3) with small values of $\mathsf{p} \lll 1$.

[1] see discussion in part (1.4)
[2] see part (1.4)

In the next paragraph, we try to determine what the structure of mixed orders is, to study whether one of these possible suboptimal practices is realistic.

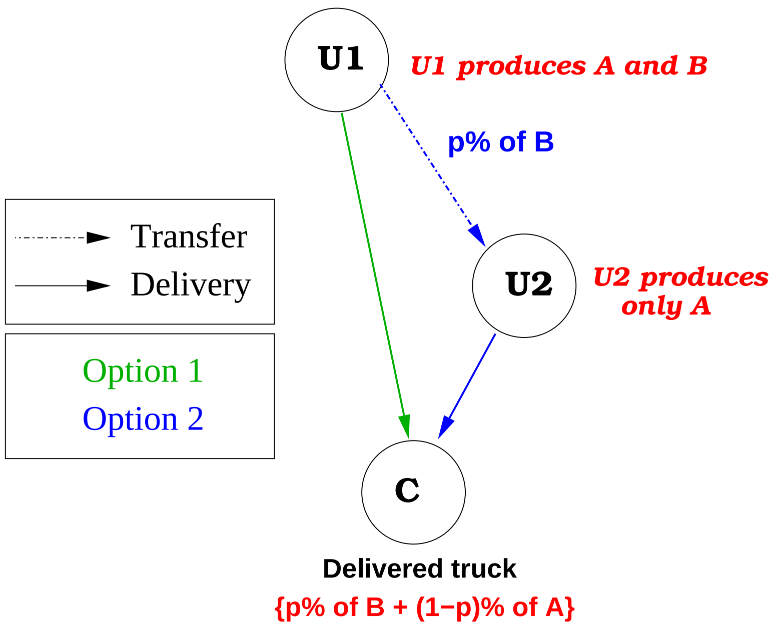


Figure A.2: Case of possible non optimal choice of the sender

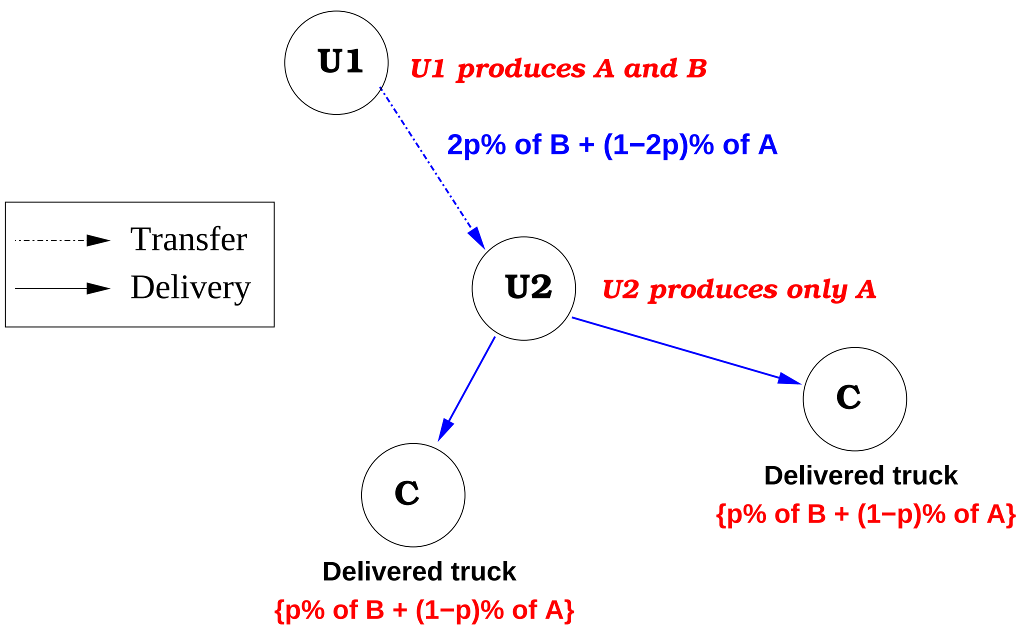


Figure A.3: Case of possible useless transfer of class A products

Analysis of the deliveries' structure

We can compute from the yearly data different information about the orders structure. Figures (A.4) and (A.5) summarize the results. The first one shows the distribution of trucks according to the percentage of high volume products contained in it.

The horizontal axis represents the percentage of cumulated (we sum on different products) products A in the truck. We can imagine its complementary axis, which is the decreasing percentage of products B, from 1 to 0. The two extremities of the axis ($x = 0$ and $x = 1$) represent the proportion of single product family trucks. It appears that 80% of the yearly delivered trucks contain only high volume products, whereas no full trucks of low volume products are found.

The transition range between these extreme values ($0 < x < 1$) is trivially included into the multi product trucks (i.e. mixed orders), and it deserves more attention.

We read on the first cumulated curve of cumulated percentage of trucks that 20% of trucks contain both A and B products. According to Figure (A.5), among these orders the mean is a truck with two third of A products and a third of B.

There is no contradiction with the previous[3] statement of 47% of multi product trucks. On the contrary, it highlights that one third of the mixed trucks corresponds to multi high volume product trucks.

What may be a qualitative analysis of such a logistic system?

[3]see Figure (1.10) on page 21

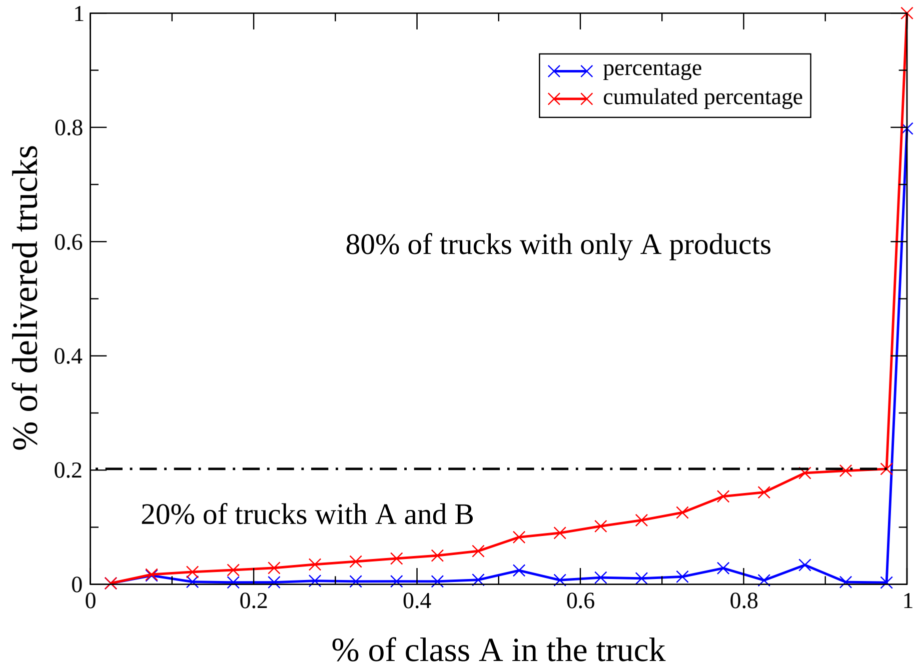


Figure A.4: Structure of delivered trucks

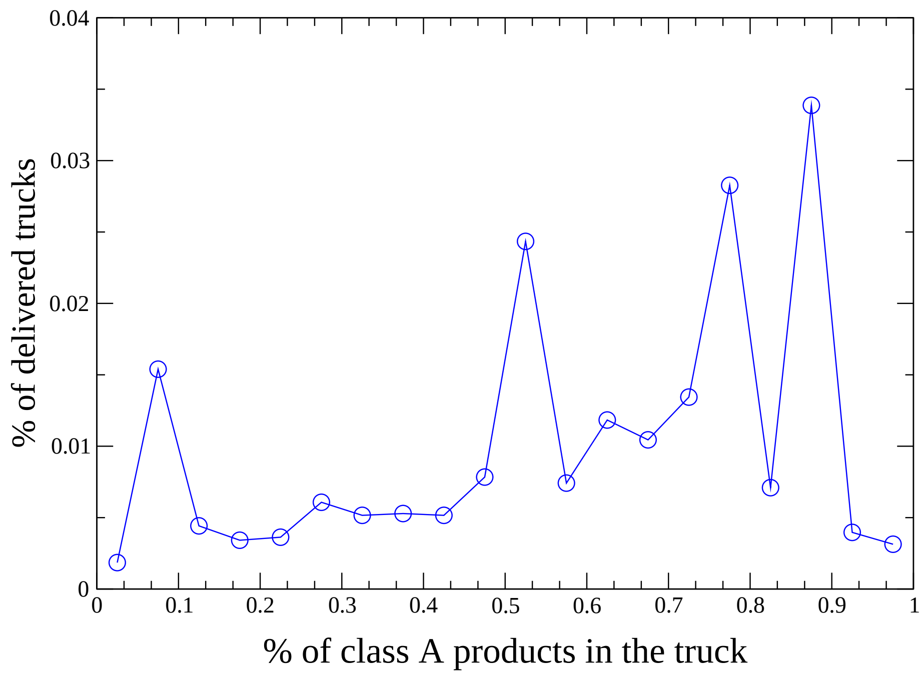


Figure A.5: Zoom on the multi product part ($x \in]0;1[$) of the curve of Figure (A.4)

A.1.1 Discussion on high volume product logistics

It is important to realize that the logistic organization for high volume products is quite basic, to the extent that possible cost reduction key factors are limited. There is indeed no possible economy of scale on means of conveyance.

Thus, we aim basically at minimizing production and inventory costs. This is a production and inventory optimization problem, which is quite difficult to solve due to the particular production constraint structure. Considering that demand is quite regular for those products, we may use a deterministic approximation, and model the problem as a linear programming one. This is the scope of forthcoming chapters.

In a nutshell, float glass plants produce in large lots (or campaigns) to exploit economies of scale in the production process. We explained indeed in paragraph (1.3) that changeovers may be long ; moreover, no valuable glass is produced during each changeover between two glass ribbon of same characteristics (colour, thickness, and width). The opportunity cost (defined as the cost of the loss of everything that could have been produced during the changeover) of each changeover is then important. We notice that this policy creates an important cycle inventory.

Our model (see chapter 6) allows us to propose an optimal production plan (at an aggregated level, corresponding to tactical decisions), which minimizes both production and inventory as well as transportation costs.

Therefore we can determine both cycle and seasonal inventories.

After that, it is important to define the safety stock for each period between two replenishment arrivals. However, due to a low demand uncertainty on those products, safety inventory remains low in comparison with cycle and seasonal ones.

All the high volume items correspond to normalized standards on the building market. As a conclusion, each customer consumes enough of those products to imagine that the cheapest way to serve them is to:

- Try to maximize the percentage of trestles filled at the end of the production process and directly sent to delivery point. This flow allows avoiding additional handling operations to get in or out of inventory. Furthermore, the probability to be producing the good reference (or to do it soon) at the time the order arrives is high, due to the high volumes of production necessary to fulfill demand.

- To minimize the global (direct and indirect) delivery distance. We have seen that it makes sense that high volume products be produced by many different factories (on average 4.3 competent plant per product). The choice of the glass origin is thus an important profitability key factor. It should be possible to avoid as much as possible the mixed origin flows on these products.

- To minimize the mixed trucks containing only high volume products. We can indeed guess that customers consume enough of each product to order it by full-load trucks. We have seen in section (§ A.1) that one third of the mixed trucks, i.e. 16% of the global sales, correspond to multi high volume product trucks. Potentially, such a policy may have an important impact.

Some of these costs savings will be passed along to the end user, so the improvement in supply chain management will result in a more competitive market position.

A.1.2 Low volume products issue

The main difficulty lies in the logistics of the low volume products. Globally, few production lines are able to product each item[4], and the production frequency is really low. Moreover, demand for these products is uncertain. It comes thus that cycle and safety inventories should be relatively higher than for products A.

[4] the average is 1.5 competent plant per product B

Risk pooling is thus an important topic in the design of distribution channels for low volume products. A priori, we want to centralize as much as possible the inventory of each product.

We notice that for the lowest volume products, it could be interesting to study the possibility to reduce the size of the minimal sold stack. This idea seems to be particularly pertinent for low volume products with high obsolescence. For instance, we quote some coated products. It could be possible to change the size of stacks through the coating process: stacks loaded at the end of the coated line could be smaller that the unloaded one at the beginning of the process.

We aim at providing the best service at a given logistic cost by offering great abilities to fulfill mixed order expectations. Given that the customer makes no difference[5] between an easy mixed order (without any indirect flow) and a difficult one, managers need to study different ways to treat these later ones.

Imagine an order that requires a mixed origin delivery (at least two different origin products according to definition (4)).

It exists many different ways to serve it, following the following decision tree (at each step it is possible to refuse the order). We relax different present constraints to be exhaustive, such as the full truckload order rule introduced in section (1.4).

- Transfers of products between two factories are not allowed:
  - A truck follows a tour and picks each product up at its production location. When the order is fulfilled, the truck goes to the customer's.
  - Each production plant sends a different non full truck. We can imagine that each site organizes its own delivery tours.
- Transfers are allowed:

[5]see discussion in paragraph (§1.4)

- Every site has on-hand inventory of all products: in this case the cheapest delivery is sent by the closest site to the customer.
- Partial transfers create favorite factories which have more products than others.
  - It exists one factory (not too far from the delivery point) which is able to fulfill the order
  - No factory have all asked products.
    - Pick up tour
    - Non full different trucks

We can also introduce the concept of logistic platform. To separate different core skills, we could forbid transfers betweens plants, and create some logistic platforms.

A.1.3 Concept of logistic platform

What are the main interests of creating a platform? For a given business, a platform mainly allows to:

- Get closer to its customers. This reduced distance increases the service level by decreasing the lead time of any delivery. Customers appreciate indeed to reduce uncertainty on their own replenishment. On certain markets (automotive market), such a platform might be mandatory: a platform can be vindicated by both strategic and context dependent argument.
- Create massive bulk flows. More important carried quantity often permits to decrease the transportation cost, by using cheaper means of conveyance. This is the main reason to build a platform. Usually, logistics managers need to bring together different products from various origins (for instance from several suppliers) to send it in an unique delivery. This need comes either from the particular demand of a customer who wants exactly all the products together

at the same time, or from the prohibitive price of shipping an insufficient total volume. In this later case, suppliers may force each order to correspond to a full truckload, etc.

- We will however broaden our reasoning to others pro-platform advantages, such as subcontracting, labor cost savings, etc.

In our case, we could imagine that plants send (directly when it is possible) their products on a platform, where every multi origin product orders may be fulfilled. The main question lies in the unknown profitability of this concept. In addition, we need to determine how many platforms is the optimal solution, and where them should be located.

Before trying to solve these questions, we need to understand the interest of a platform in the business of Saint-Gobain Glass. To study it, we first create some simulations (section (A.2)). Then, we lead a discussion on it based on simple cost models (section (A.3)).

A.2 Our simulation approach on real data: methods and insights

To simplify we keep the constraint of full truckload deliveries, but we do not consider the replenishment issue. We use the perfect replenishment assumption.

Assumption 39 ***Perfect replenishment assumption :*** *we authorize non full truckload replenishments during transfers between two plants or between a plant and a platform. Nevertheless, we use a constant transportation cost (in €/T/km) equal to the cost of a full truckload. This is a strong assumption.*

We propose a pragmatic method to simulate on real yearly past data the different distribution scenarios. We do not take into account production costs and constraints and we assume that we know different potential platform locations From a practical point of view, many potential platform are based on existing facilities. Thus our approach makes sense as a first step.

To the extent that single product trucks are always sent from the closest possible plant and thus can not be improved, we only take into account all the multi product orders, including those which can be served without any mixed origin. These orders represent around a half of the total sales on the studied geographical perimeter. We assume that each plant and each platform are potential senders for each delivery.

We compute the cost of a distribution solution by adding:

- Transportation costs (cost by unit and by distance). Traditionally, we work with C_T €/T/km. The cost is proportional to the Euclidean distance.
  - Transfers between plants can be discounted, because of the high refill rate[6].

[6] see discussion in paragraph (1.4)

  - Each flow from plant to platform can also be discounted.

- Handling costs (cost by unit) are added at each flow interruption in any inventory (plant or platform). By default, we take $\mathsf{C_M}$ €/T. We count two steps for each indirect flow. Platform cost can be discounted according to different supplier' offers.

On-hand past data give us for each accepted order the way Saint-Gobain Glass served it: we have for each product its producing plant and for each order the final sending plant.

We have built two different simulations to measure the potential savings in the distribution of glass. In our first simulation (section (A.2.1)), we check whether we have optimally chosen the sending plant, given the producing plant of each product of the order. In our second simulation (section (A.2.2)), we keep the real sending plant of every order but we take the best producer for each product by assuming that it is always on-hand in inventory of any competent plant.

Finally, these simulation will lead to section (A.3) in which we develop some practical rules to choose both a producer for any mixed part of the order and a sender for the final delivery.

A.2.1 First simulation: best choice of crossing point

For each multi product order, we find the best expedition site among all possibilities (plants or platforms) while keeping the real production plant.

That way, we have a first insight about the impact of different cost hypotheses and of the location of different platforms. In addition, we highlight that most of the savings in this simulation do not stem from the platform.

Definition of three hypotheses

We tested simultaneously three hypotheses of costs:

1. Hypothesis one: the handling cost in a platform is equal to the one in a factory; there is no discount on the transportation cost of links {plant $\rightarrow$ platform} whereas 20% discount is used on transfers (links {plant 1 $\rightarrow$ plant 2}) between plants. This hypothesis corresponds to the case where reloading rate[7] of the trucks is null on the platform.

2. Hypothesis two: we apply the 20% discount also on flows between plant and platform, but we keep the same handling cost in all warehouses.

3. Hypothesis three: We add a 20% discount on the handling cost on platforms.

These different simulations will give is more insights in the understanding of the underlying improvement key factors.

Results of the first hypothesis

The results are plotted on Figure (A.6). In the first scenario, we find that it would not have been cheaper to send any order from the platform. This is understandable because we are still working with the present demand structure, which is for the mixed orders mainly corresponding to a classical truck with a lot of products A and few products B, as explained in part (A.1). Thus sending a truck from a platform would be profitable only if the savings on transportation cost offset the handling costs of the replenishment of high volume products on the platform. Given the importance[8] of transportation cost compared to handling cost, this is not happening. In addition, the existing network of inventory facilities (located in plants) is quite wide on the covered perimeter.

[7]see discussion in part 1.4
[8]see remark (**??**)

However, we show that a 4% reduction of the total cost is possible in choosing a better shipping plant. In addition, savings come mainly from the high volume products, on which we can save on the unit cost 50% more than on the low volume ones. Thus, it seems that high volume products are presently traveling on a longer distance than possible, illustrating for instance cases described by Figures (A.2) and (A.3).

In particular, this computation highlights that the main part of the savings comes from a switch of the shipping role between two plants, that we denote here P_1 and P_2. We will discuss this question in part (A.3.1): when we have a two origin delivery, which sending plant is the cheapest one?

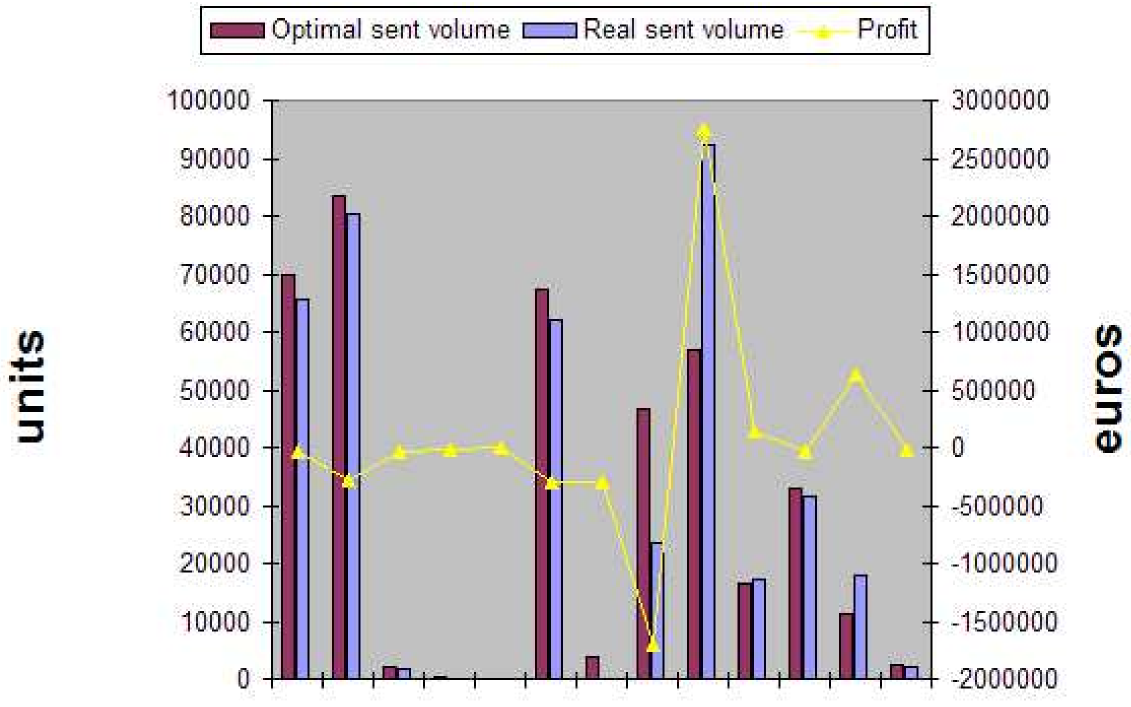


Figure A.6: Results (in Tons) of the simulation of the **first hypothesis** on past yearly data

If we study the phenomena in details, we can notice that P_1 is a glass plant which only sends its own production without any origin mixing. On the contrary, the other plant P_2 plays a historical role in the distribution: it is considered as a mixing platform-plant by operational teams. Nowadays, half of the quantity sent by P_2 in mixed trucks has not been produced there.

In this analysis, it appears that its role is not optimal, and that reinforcing the sending capacity of P_1 would be profitable: in the optimal case, half of the quantity sent by P_1 in mixed trucks is origin mixed. On the contrary, other plants still have the same level of activity, which makes us think that a correction in the distribution rules would be possible. Naturally, we can also imagine that for strategic reasons the management prefers to use an external platform. In this case, the platform could be located near the plant P_1.

Globally, we could say that in this hypothesis we defeat the platform solution by using a non discounted transportation cost between plant and platform. What happens if we use the same discount as for transfers between plants?

Results of the second and third hypotheses

The results of the second hypothesis is plotted on Figure (A.7).

The second scenario shows that a platform which is proposed to be near the MannheimP_1 plant naturally is a competitor for it. Thus, a part of volume (10.000 Tons during the year) is now sent by the external platform.

Obviously, the third scenario reinforces this result because we give an advantage to the platform with a cheaper handling cost: the volume sent by the platform is around 30.000 Tons during the year. The third hypothesis results are represented on Figure (A.8).

Globally, it appears that a platform may not be an additional cost under favorable conditions. However, potential savings are proportionally insignificant.

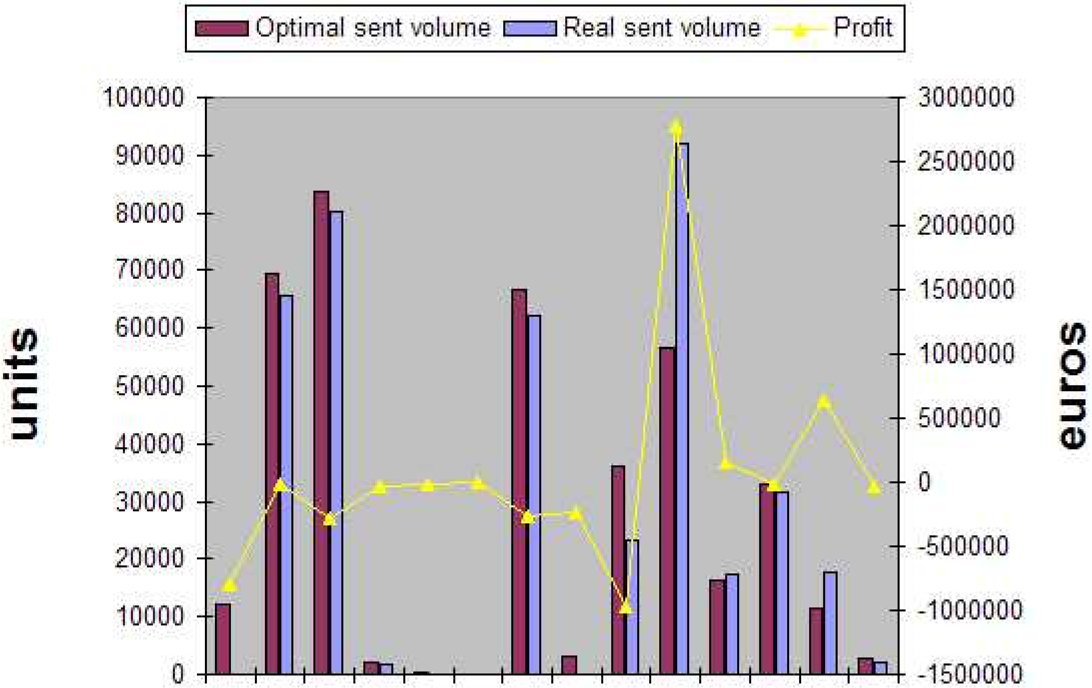


Figure A.7: Results (in Tons) of the simulation of the **second hypothesis** on past yearly data

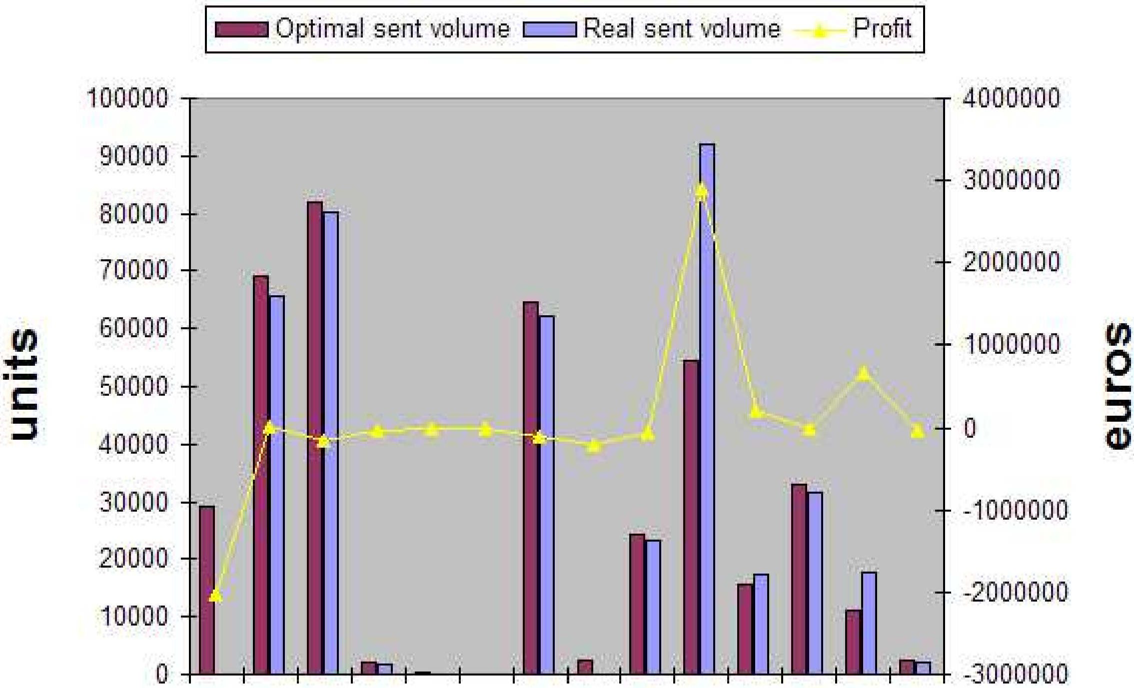


Figure A.8: Results (in Tons) of the simulation of the **third hypothesis** on past yearly data

A.2.2 Second simulation: the relaxation of production constraints

In the previous simulation, we used the real used production plant for each delivered products and tried to optimize the sending plant. In this part, we keep the real sending plant and optimize the producers. We use the skills' table, which gives for each plant the corresponding products it is able to produce. We implicitly use the assumption that *products are always on hand in each competent plant inventory.* Thus, we imagine *infinite capacity plants and no shortage.*

For each real order, we consider that the shipping plant is known. We try to optimize the origin plant of each product by checking all possibilities and keeping the cheapest one.

The main interest of this simulation is to have an idea of the financial gap between the constrained reality and an unconstrained virtual case. In addition, we hoped this analysis would give us new insights.

At least, it will help us to understand the trade-off between proportional volume and relative distance to the customer to develop an easy allocation rule. We will develop explanations in the paragraph (A.3.2).

In our relaxed simulation, it appears that we can cut 90% of the distribution cost of high volume products, whereas only a half of the cost of low volume ones. Those results appear really surprising, and we need to focus on it to explain it and highlight the limits of this result.

The Figure (A.9) shows both the difference[9] between real and optimal production of each plant and its associated profit. It appears that most of the changes lie in the transfer of production to the plant of Porz, and particularly for the production

[9]the difference (optimal volume - real volume) is denoted Delta Volume on Figure (A.9)

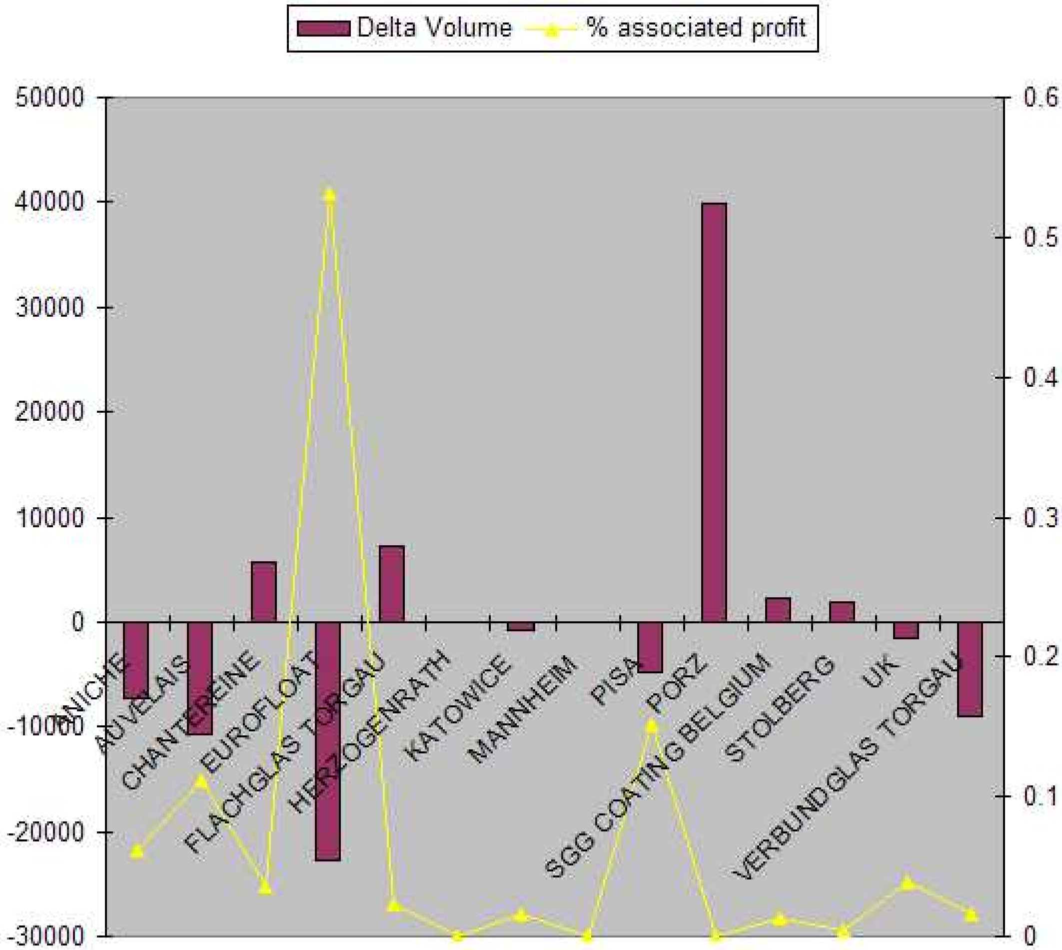


Figure A.9: Results of the simulation: Impact (on volume and gain) of the optimization of the producing plant

of a best-seller coated product from the French plant of Eurofloat. It seems that Eurofloat produced a lot of coated glass for German market that could have been produced in Porz.

This remark is really interesting, because this anomaly stems from the arrest of the plant of Porz during the year, as shown by Figure (A.10). It plots the global sold quantity of float glass which had been produced by Porz on the time horizon that we are studying. A delay is obviously due to inventory.

Remark 8 *Given the operating life* [10] *of a float plant and the number of plants*

[10] see paragraph (1.3)

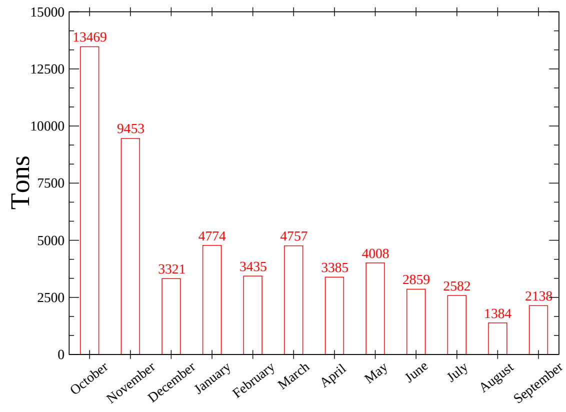


Figure A.10: Sales of float glass produced by the Porz plant

Saint-Gobain Glass owns in Europe, approximately ***every year one plant is being rebuilt****. Each repair lasts half a year, and in the particular case of Porz in 2003, it was exceptionally long. Given the number of plants,* ***every year, the logistic rules are disturbed by a new industrial scheme (one plant is stopped)****. Nevertheless, teams are still working in the plant, and the activity has to be kept as steady as possible. That is why we have seen in all the previous simulations that Porz has shipped a huge quantity of glass which had been produced somewhere else. We do not know if it would have been possible to avoid these flows.*

The interest of this simulation is limited, but it emphasizes the fact that strategic and tactical production scheduling is really a key factor in the glass business, given the on-hand capacities. This adds interest to the Linear Programming model developed in chapter 6 which permits the user to simulate and optimize all evolving scenarios of plant stops.

From a practical point of view, our simulations point out that potential savings may lie in the real time optimal affectation of orders to the cheapest producers and senders. Nowadays, we use some fixed rules: each plant covers its territory and serve orders within it.

To go furthermore, we need to analyze whether it is possible to choose easily not only the shipping plant but also the producers to fulfill mixed orders. Thus, we aim in section (§ A.3) at determining easy practical rules that minimize distribution costs.

A.3 Basic Models determining rules to serve mixed orders

For each mixed order, section (A.2) pointed out the potential of the optimal affectation of producers and sender. To achieve it in practice, we could try to find dynamically the optimal solution. For instance, it would be nice to implement in the information system a tool that helps sellers to affect the order optimally, according to on-hand inventories at different locations.

This is not presently the case, and it may be also interesting to determine some practical rules that would be nearly optimal.

Here we focus on practical rules which are by definition easier to implement. We still work under the perfect replenishment assumption (described on page 275).

To make it simple, we divide the problem into two decisions in a raw: the choice of the sender and the choice of providers (which are in our case the producers). But the decision sequence is not obvious. **What is the best option between either choosing the sender plant and then all the production plants or choosing the producers and then the sender?**

Firstly, if we consider only our existing plants as potential senders, we may wonder several questions:

May we choose the sender as the closest one to the customer because we use the strong assumption of perfect replenishment? Does the product corresponding to the biggest volume of the order determine the sender?

We call this dilemma the trade-off between the smallest distance and the biggest volume: is the cheapest sender the biggest producer of the order or the closest sender to the customer? We built in part (A.3.1) a simple model to tackle this question.

Secondly, let us consider new possibilities: is it valuable to send an order from a non producing location? We tackle this question of **profitability of non-producing crossing point** in section (A.3.2).

A.3.1 Trade-off between the smallest distance and the biggest volume: case without any platform

By dominated order we mean an order which contains a majority product. We propose to study the easiest model to deal with this question.

Definition of the model

We consider the following problem:

- we create a basic situation made of:
  - one customer C, two factories U_i and two products P_i ($i \in \{1, 2\}$).
  - $\forall i$, P_i is produced exclusively by the plant U_i. We consider P_1 and P_2 have the same production and inventory cost.
  - The customer triggers a **mixed order**, which is made of both product P_1 and product P_2.
- We introduce in several parameters and variables:
  - **We know the positions of U_1 and customer C. The position of U_2 is unknown. We use two variables $\{x, y\}$ to denote it.**
  - **The order is made of $p\%$ of product P_1 and $(1-p)\%$ of P_2.**
  - We consider the costs:
    - C_T is the transportation cost (in €/unit[11]/km). We allow a discount

[11]unit denotes the used unity of product: for instance it may be some Tons, square meters, etc.

parameter $0 \leq 1 - a \leq 1$ for transfers between plants (ex: $a = 0.8$: 20% discount). This parameter captures the reloading rate[12] of the incoming trucks that is possible in a plant.

* C_M is the handling cost (in €/unit) in any inventory location. We count it twice for an indirect flow (i.e. a two step flow, such as $\{U_1 \to U_2 \to C\}$).

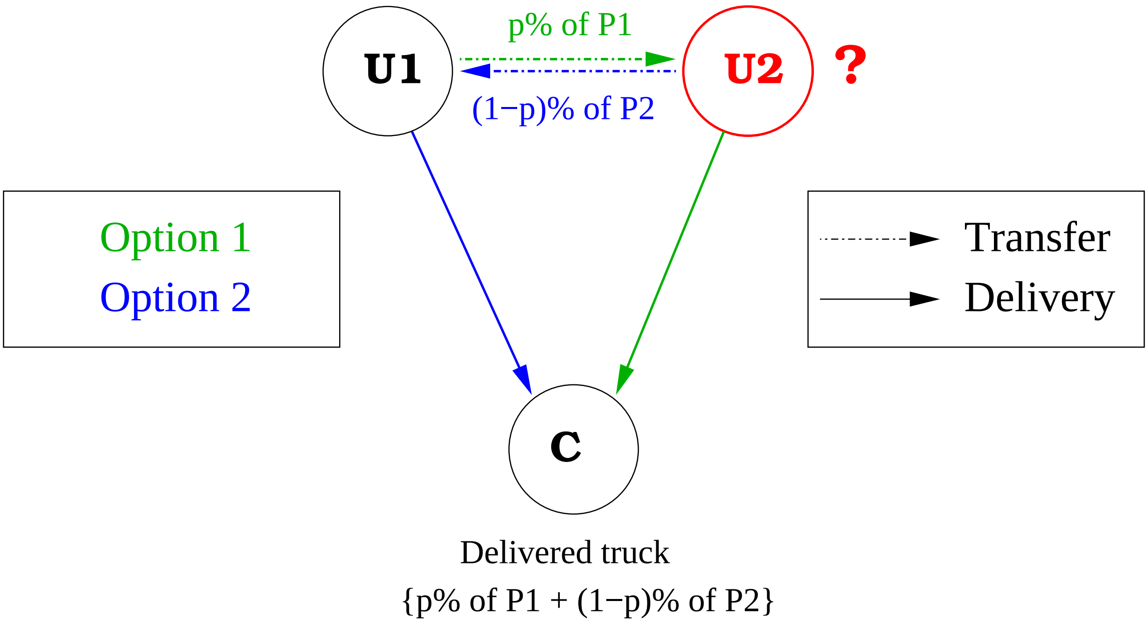


Figure A.11: Illustration of the definition of the model in part (A.3.1)

To serve the order, we have the choice between two options (as shown by Figure (A.11)). To compute transportation costs, we use the Euclidean distance[13]:

1. U_1 sends p% of P_1 to U_2 ; U_2 then sends the final truck to the customer C (*Option 1 on Figure (A.11)*). In this case, the cost per unit of the delivery is:

$$C_1(x, y, p) = C_T \times \big(a \times p \times d(U_1, U_2) + d(U_2, C)\big) + C_M \times (p + 1) \quad \text{(A.1)}$$

[12] cf. discussion in section (1.4)

[13] $d(A, B)$ denotes the Euclidean distance between two points A and B

2. U_2 sends $(1-p)$% of P_2 to U_1 ; U_1 is then the sending plant (*Option 2 on Figure (A.11))*. The cost (in €/unit) of this option is:

$$C_2(x,y,p) = C_T \times \left(a \times (1-p) \times d(U_1,U_2) + d(U_1,C)\right) + C_M \times (2-p) \quad \text{(A.2)}$$

Our goal is to answer the simple following question. **When is it cheaper to use U_1 as the final sending plant?**

To do so, we define by (A.3) the profit function $\mathcal{F}$ as the difference between the cost of the case with U_2 as sending plant and the case in which it is U_1. Using equations (A.1) and (A.2), we have:

$$\mathcal{F}(x,y,p) = C_1(x,y,p) - C_2(x,y,p) \quad \text{(A.3)}$$

Graphic interpretations of the results

For all figures of applications of the model in this part (i.e. part (A.3.1)), we use the following default numerical values for:

- the transportation cost $C_T = 0.08$ €/unit/km.
- the handling cost $C_M = 10$ €/unit.
- the fixed positions[14]:
  - the plant $U_1 = \{0,0\}$.
  - the customer $C = \{0,400\}$.

On 3D figures (A.12), (A.17), (A.18) and (A.19) we plot some 3D level curves $\{\mathcal{S}(c)\ /\ c \in]-\infty;+\infty[\ \}$ defined by relation (A.4). $\mathcal{S}(c)$ is the set of points of the

[14] thus, we have $d(U_1,C) = 400$ km

space $\{x, y, p\}$ for which the gain function $\mathcal{F}$ equals c.

$$\mathcal{S}(c) = \{ \{x, y, p\} \in (\mathbb{R} \times \mathbb{R} \times [0; 1]) \quad \text{s.t.} \quad \mathcal{F}(x, y, p) = c \} \tag{A.4}$$

The most interesting of these 3D level curves is the surface of null gain, i.e. $\mathcal{S}(0)$: if U_2 belongs to this surface, it costs the same price to use U_1 or U_2 as sending plant.

On Figure (A.12), we look at its behavior in a case where we do consider neither the handling costs nor a discount ($C_M = 0$ and $a = 1$).

On the two first axes, we plot the position of the second plant U_2. The third axis allows us to plot the result for all values of parameter p. We notice that $p = 0$ (respectively $p = 1$) corresponds to a direct shipping of a full truck of P_2 from U_2 (respectively P_1 from U_1).

To represent the solutions for each given values p_0 of our parameter p, we introduce the 2D level curves $\{\mathcal{S}_{p_0}(c) \ / \ c \in]-\infty; +\infty[\}$. Each curve $\mathcal{S}_{p_0}(c)$ corresponds to the set of point U_2 in the space $\{x, y\}$ where using U_1 instead of U_2 as sending plant provides a gain[15] of c €/unit. It is defined by the relation (A.5).

$$\mathcal{S}_{p_0}(c) = \{ \{x, y\} \in \mathbb{R}^2 \quad \text{s.t.} \quad \mathcal{F}(x, y, p_0) = c \} \tag{A.5}$$

Basically, the link between 3D and 2D curves is intuitive: for a given c, $\mathcal{S}_{p_0}(c)$ is the 2D curve resulting from the slice of the surface $\mathcal{S}_(c)$ through the plan $p = p_0$ (perpendicular[16] to the axis of p).

On 2D figures (A.13), (A.14), (A.15) and (A.16), we only vary the value of the parameter p_0. We have no handling cost ($C_M = 0$) and no discount on the transportation cost ($a = 1$).

[15]naturally, a negative gain is a loss
[16]horizontally in Figure (A.12)

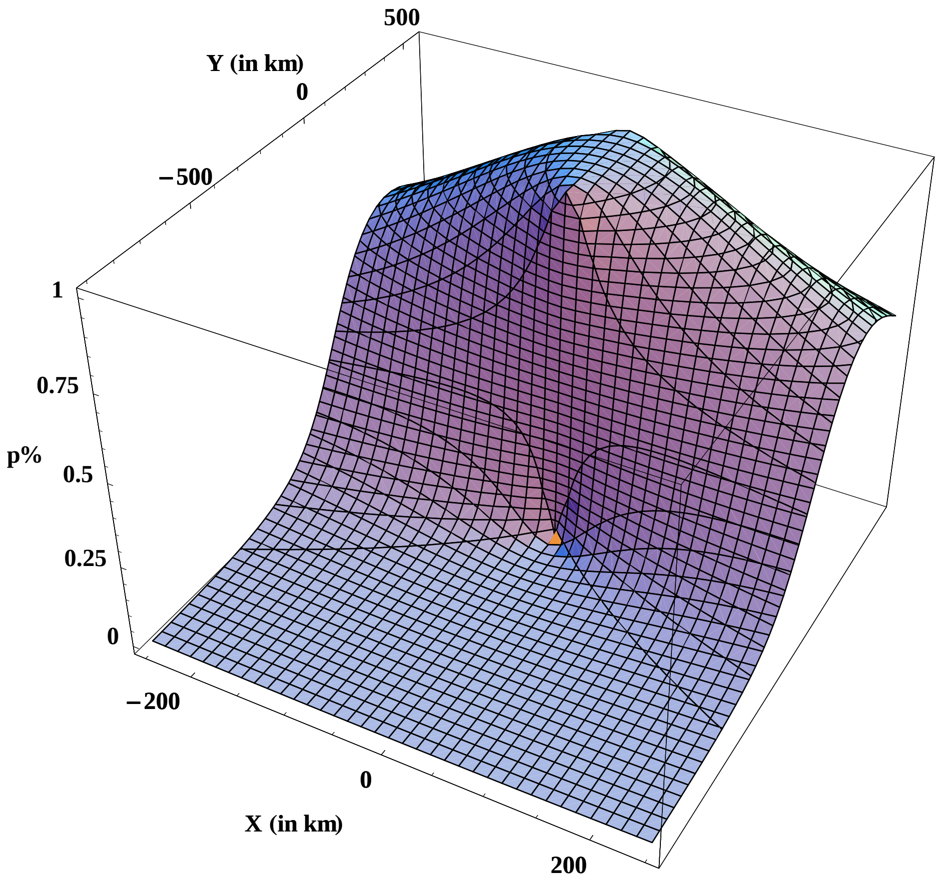


Figure A.12: 3D visualization of the null profit surface $\mathcal{S}(0)$; Case with $C_M = 0$ and $a = 1$

Blue level curves correspond to positions of U_2 for which U_1 is the cheapest sending plant ($c > 0$: we plot $\mathcal{S}_{p_0}(5)$ and $\mathcal{S}_{p_0}(10)$) **whereas red ones correspond to the contrary** ($c < 0$: we plot $\mathcal{S}_{p_0}(-5)$ and $\mathcal{S}_{p_0}(-10)$). The bolder **black curve is the null gain level one** $\mathcal{S}_{p_0}(0)$.

We obtain the following figures:

- for $p_0 = 0.8$, we obtain the Figure (A.13) made of convex sets. Obviously, for p tending to one we observe that the red zone tends to zero: U_1 produces the majority of the order and so is mostly the cheapest sending plant.

- for $p_0 = 0.5$, the cheapest sending plant is obviously the closest one, due to the symmetry of the pattern, and level curves are circles, as shown on Figure (A.14).

- in the $p_0 < 0.5$ case, we loose the convexity property. In Figure (A.15) we plot the $p_0 = 0.25$ solution.
- when p tends to zero, the blue zone tends to zero. For $p_0 = 0.05$, we even obtain Figure (A.16).

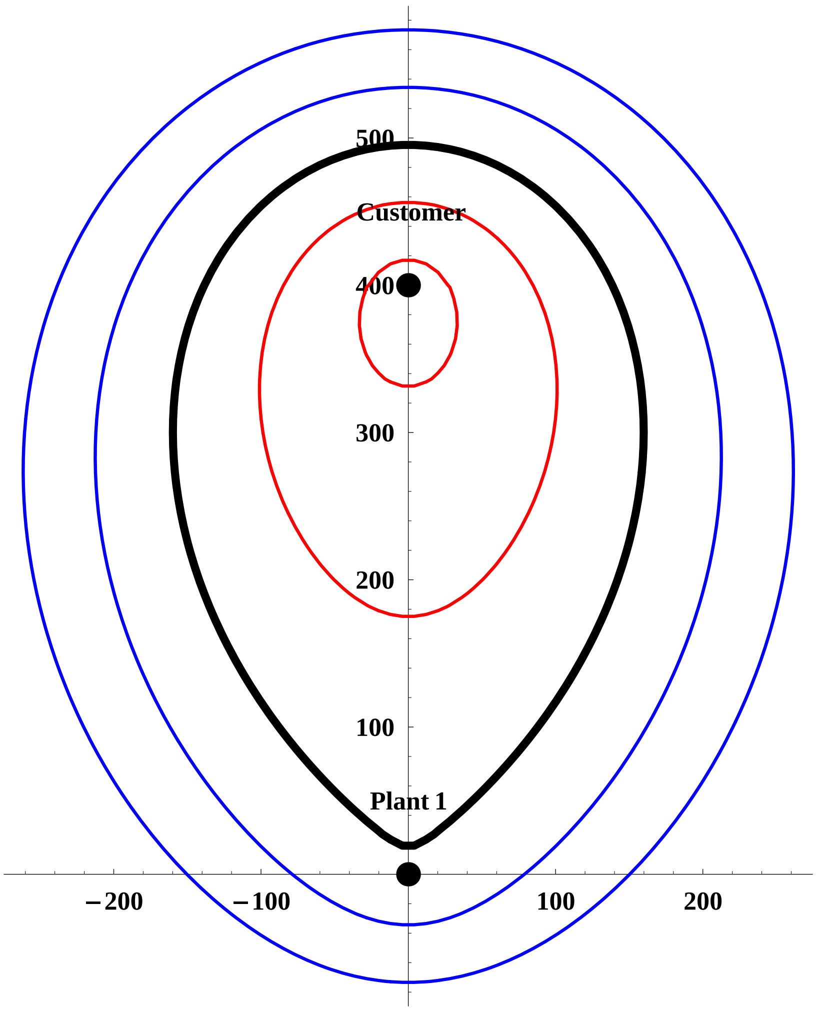


Figure A.13: 2D level curves $\mathcal{S}_{0.8}(c)$ of the profit function $\mathcal{F}$; $c \in \{-10, -5, 0, 5, 10\}$

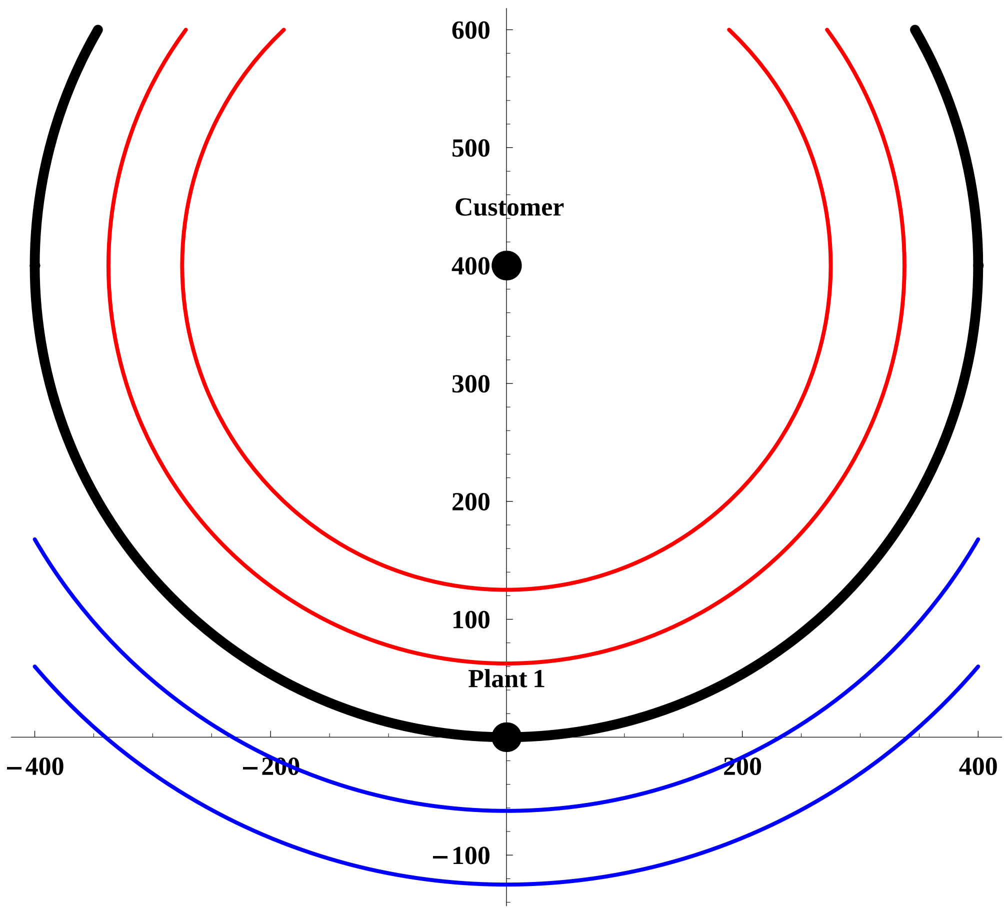


Figure A.14: 2D level curves $\mathcal{S}_{0.5}(\mathsf{c})$ of the profit function $\mathcal{F}$; $\mathsf{c} \in \{-10, -5, 0, 5, 10\}$

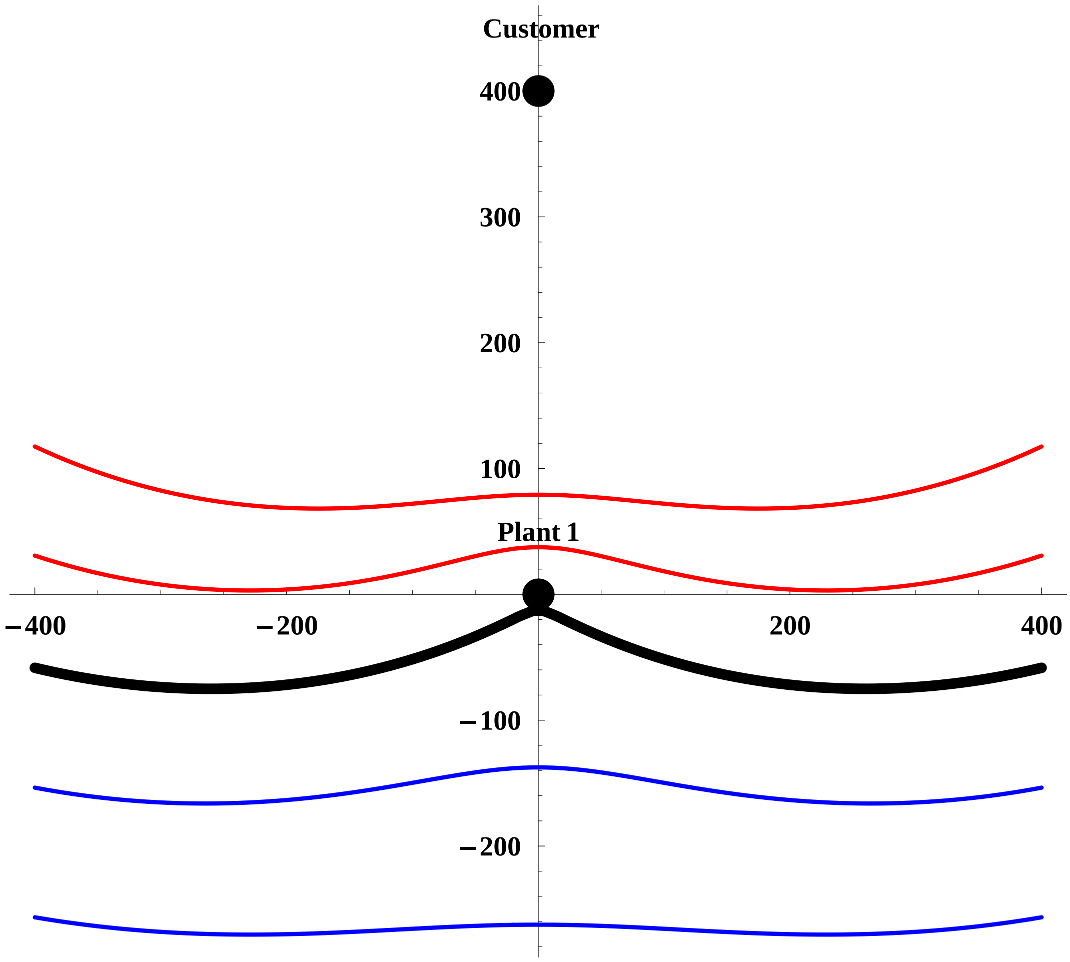


Figure A.15: 2D level curves $\mathcal{S}_{0.25}(\mathsf{c})$ of the profit function $\mathcal{F}$; $\mathsf{c} \in \{-10, -5, 0, 5, 10\}$

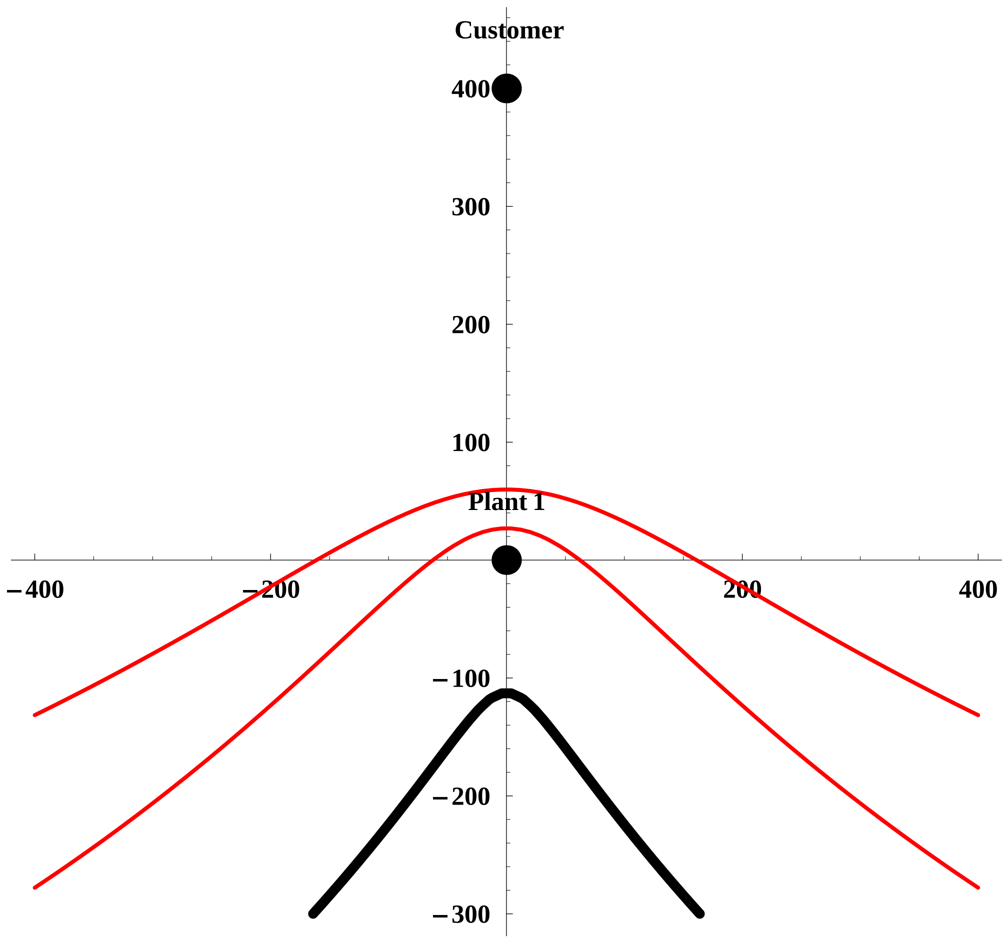


Figure A.16: 2D level curves $\mathcal{S}_{0.05}(c)$ of the profit function $\mathcal{F}$; $c \in \{-10, -5, 0, 5, 10\}$

The shape of the solution is of course also modified when we **modify others parameters**, such as the handling cost C_M or the discount on the transportation cost a.

Firstly, let us introduce the handling cost while keeping no discount ($a = 1$). This cost naturally tends to give a key role to the plant with the biggest production part. Figure (A.17) highlights that the global 3D curve is smoothed.

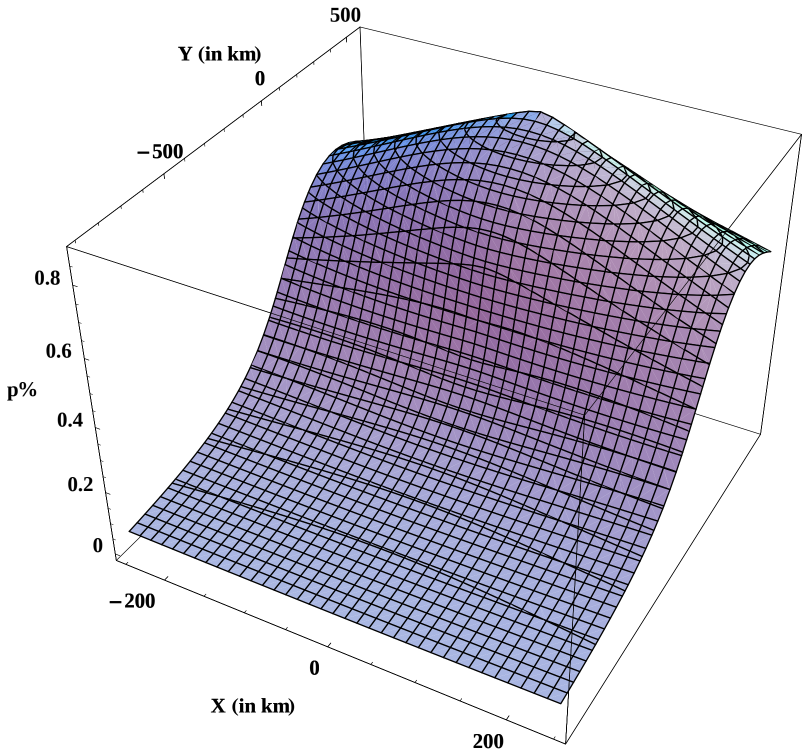


Figure A.17: 3D visualization of the null profit surface $\mathcal{S}(0)$; Case with $C_M = 10$ and $a = 1$

We notice that a unique solution (a dominating expedition plant) exists for extreme values of p.

In the same way, let us introduce the discount parameter ($a < 1$) while forgetting the handling cost ($CM = 0$). Figure (A.18) presents the corresponding 3D curve.

We can guess that **the asymmetry between upstream and downstream sending plant transportation costs is going to give a key role to the closest**

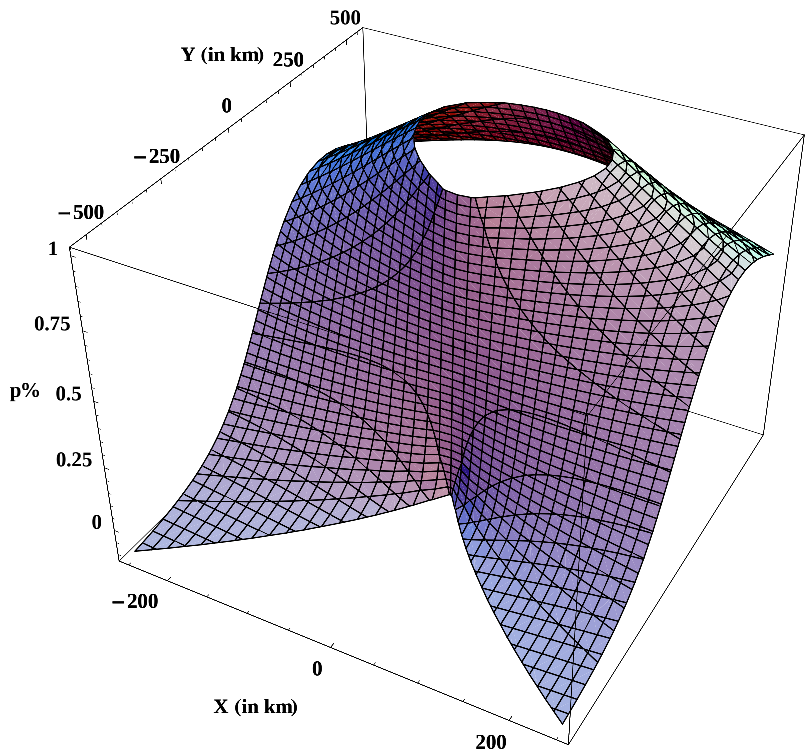


Figure A.18: 3D visualization of the null profit surface $\mathcal{S}(0)$; Case with $C_M = 0$ and $a = 0.8$

factory to the customer.

However for extreme values of p the discount parameter of our model is meaningless. Indeed, even for a single product order of P_1, our model makes it cheaper to send it via U_2. It does not make any sense because the reloading rate of the truck does not then have any meaning, and it was the main reason to take into account a discount parameter.

We just understand better the influence of different parameters on the choice of the cheapest sending plant.

If we introduce both the handling cost and the discount parameter, we find that the importance of the handling cost is the key factor: the curve on Figure (A.19) is really smoothed and looks like Figure (A.17) rather than Figure (A.18).

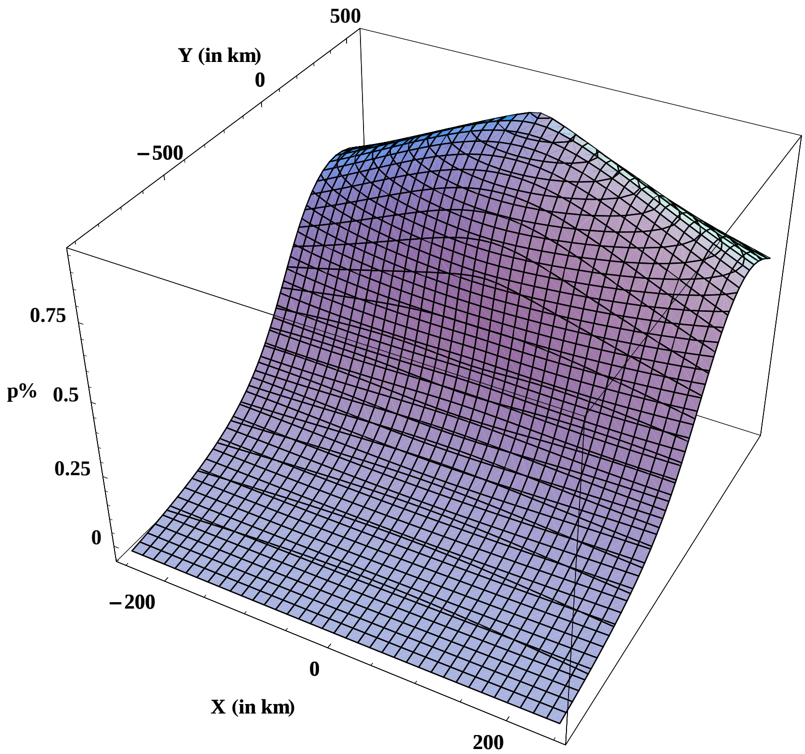


Figure A.19: 3D visualization of the null profit surface $\mathcal{S}(0)$; Case with $C_M = 10$ and $a = 0.8$

As a conclusion, we could assume that **for extreme values of p (approximately $p > 0.75$ or $p < 0.25$), the plant producing the majority product is globally the cheapest one to send the order** (even if it still depends on the real relative positions of customer and plants). **In the medium range, there exists a trade-off between the weighted distances** and there is thus no clear rule to serve such orders.

This simple model is far from the complex reality. However, it captures many phenomena, and especially the trade-off between the smallest distance rule and the biggest volume rule in the choice of the shipping plant. In addition, we have seen in the Figure (1.10) that two products orders are most of the mixed orders.

We have thus a good idea of the best treatment of dominated order. Firstly we find the closest (to customer) competent plant which produces the dominating product. This will be the sending plant. Then we find the closest (to the sending plant) competent plants for each other product.

If all products are equally distributed, we need to better understand the phenomena. Could it be valuable to use a non-producing crossing point, such as either a different plant (which is not involved in the production of the order) or a platform?

A.3.2 Profitability of a non-producing crossing point: case with a platform

Definition of the model

To deal with this question, we build a simple model:

- We create a basic situation:
  - two plants U_1 and U_2, one customer C, and one crossing point P.
  - U_i produces the specific product P_i.
  - The customer C orders a mixed truck with nearly half of P_1 and half of P_2.
- To compare the solution with a transfer between the two plants and the solution with the use of a third crossing point P, we introduce different parameters:
  - **We know the positions of U_1, U_2 and the customer C. The position of P is unknown**. We use two variables $\{x, y\}$ to denote it.
  - The order is made of $p\%$ of product P_1 and $(1-p)\%$ of P_2. Let us consider P_1 and P_2 as having the same value.
  - We consider the costs:
    - C_T is the transportation cost (in €/unit/km).We take into account some discount parameters:

· $0 \leq 1 - a_1 \leq 1$ for transfers between plants (ex: $a_1 = 0.8$: 20% discount on flows $\{U_1 \longleftrightarrow U_2\}$). This parameter captures the reloading rate of the incoming trucks that is possible in a plant.

· $0 \leq 1 - a_2 \leq 1$ for transfers between plants and crossing point (ex: $a_2 = 0.8$: 20% discount on flows $\{U_k \longrightarrow P\}$). This parameter captures the reloading rate of the incoming trucks on a platform.

∗ C_M is the handling cost (in €/unit) in any inventory location. We count it twice for a two step flow[17]. We introduce a discount parameter on the crossing point $0 \leq 1 - b \leq 1$ (ex: $b = 0.8$: 20% discount on C_M in the inventory of P).

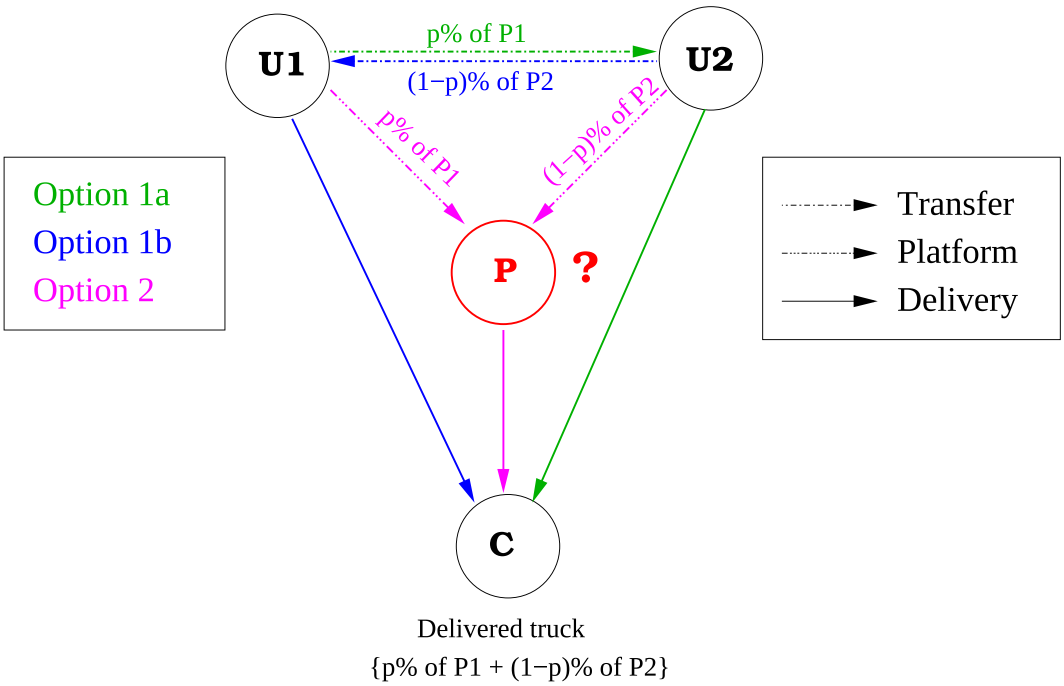


Figure A.20: Illustration of the definition of the model in part (A.3.2)

To serve the order, we have the choice between two options (as shown on Figure (A.20)):

[17]such as flows $\{U_1 \to U_2 \to C\}$ or $\{U_i \to P \to C\}$

1. Without a third point, we choose the sending plant which is the cheapest one, we denote it U_i. On Figure (A.20), we choose either *option 1a or option 1b.* We denote $\{X_2; Y_2\}$ the fixed point corresponding to U_2. Using the model of part (A.3.1) and the function $\mathcal{F}$ defined by equation (A.3)[1], we have:

$$U_i = \begin{cases} U_1 & \text{if}^2\ \mathcal{F}(X_2, Y_2, p) \geq 0; \\ U_2 & \text{if } \mathcal{F}(X_2, Y_2, p) < 0. \end{cases} \tag{A.6}$$

We denote $\bar{i}$ as the complementary of i in the set $\{1, 2\}$ and p_i the flow corresponding to U_i:

$$i = 1 \quad : \quad \begin{cases} \bar{i} = 2; \\ p_i = p. \end{cases}$$

$$i = 2 \quad : \quad \begin{cases} \bar{i} = 1; \\ p_i = 1 - p. \end{cases}$$

As a result, in this option U_i sends p_i% of P_i to $U_{\bar{i}}$ which then sends the final truck to the customer. In this case, the cost per unit of the delivery is:

$$C_1(p) = C_T \times \big(a_1 \times p_i \times d(U_i, U_{\bar{i}}) + d(U_{\bar{i}}, C)\big) + C_M \times (p_i + 1) \tag{A.7}$$

2. With a third point, each plant sends its own product to P which then sends directly the final truck to the customer. On Figure (A.20), it corresponds to *option 2.* We have globally only indirect flows with two handling costs.

$$C_2(x, y, p) = C_T \times \Big(a_2 \times \big(p \times d(U_1, P) + (1-p) \times d(U_2, P)\big) + d(P, C)\Big) + (1+b) \times C_M \tag{A.8}$$

[1]on page 287

[2]by convention: $\mathcal{F}(X_2, Y_2, p) = 0 \quad : \quad U_i = U_1$

Our goal is to answer the simple following question. **When is it cheaper to use a non-producing crossing point?** Thus , we study the profit function $\mathcal{G}$ defined by (A.9) as the gain that can be possible by the use of the point P instead of the cheapest plant as sender of the final delivery truck. Using equations (A.7) and (A.8), we define:

$$\mathcal{G}(x, y, p) = C_1(p) - C_2(x, y, p) \tag{A.9}$$

Graphic interpretation of the results

For a given value p_0 of p, the spacial set corresponding to a positive value of the function $\mathcal{G}(x, y, p_0)$ give us profitable positions of a third non-producing crossing point.

To plot it, we use level curves $\{\mathcal{L}_{p_0}(c) \ / \ c \in]-\infty; +\infty[\ \}$. Each curve $\mathcal{L}_{p_0}(c)$ corresponds to the set of points in the space $\{x, y\}$ where using a crossing point P permits a gain[18] of c €/unit. It is defined by the relation (A.10).

$$\mathcal{L}_{p_0}(c) = \{\ \{x, y\} \in \mathbb{R}^2 \quad \text{s.t.} \quad \mathcal{G}(x, y, p_0) = c\ \} \tag{A.10}$$

On following Figures (A.21), (A.22) and (A.23), **blue level curves correspond to profitable non producing crossing points** ($c > 0$: we plot $\mathcal{L}_{p_0}(5)$ and $\mathcal{L}_{p_0}(10)$) **whereas red ones correspond to non profitable positions** ($c < 0$: we plot $\mathcal{L}_{p_0}(-5)$ and $\mathcal{L}_{p_0}(-10)$). The bolder **black curve is the null gain level one** $\mathcal{L}_{p_0}(0)$.

To get used to our model, we take classical default values for:

- the transportation cost $C_T = 0.08$ €/unit/km.
- no discount on transportation costs $a_1 = 1$ and $a_2 = 1$.

[18]naturally, a negative gain is a loss

- the handling cost $C_M = 10$ €/unit.
- no discount on handling cost in the third crossing point P: $b = 1$.
- positions[19]:
  - first plant: $U_1 = \{-100, 0\}$.
  - second plant: $U_2 = \{100, 0\}$.
  - customer: $C = \{0, 400\}$.
- The order is equally distributed between P_1 and P_2: $p_0 = 0.5$.

On Figure (A.21), we use default values, except a discount on transfers between plants ($a_2 = b = 1$, but $a_1 = 0.8$). We show that a platform is never valuable: **with these financial parameters we can not justify a second stop on the global product flow**.

If we consider discounts on the platform both for handling costs and for upstream transportation costs ($a_1 = a_2 = b = 0.8$, as plotted on Figure (A.22)), naturally the blue zone is extended, but is still globally limited. **This confirms insights of our simulations in section (A.2): a non-producing crossing point is not in the glass business a generic source of profit (at least under present organization).**

To see the impact of the repartition of the products in the order, we only change the parameter $p_0 = 0.8$ on Figure (A.23). We confirm the results of the previous paragraph (A.3.1): **for extreme values of p plant producing the majority product is always the best solution**.

[19] we have $d(U_1, U_2) = 200$ km and $d(U_k, C) \approxeq 400$ km

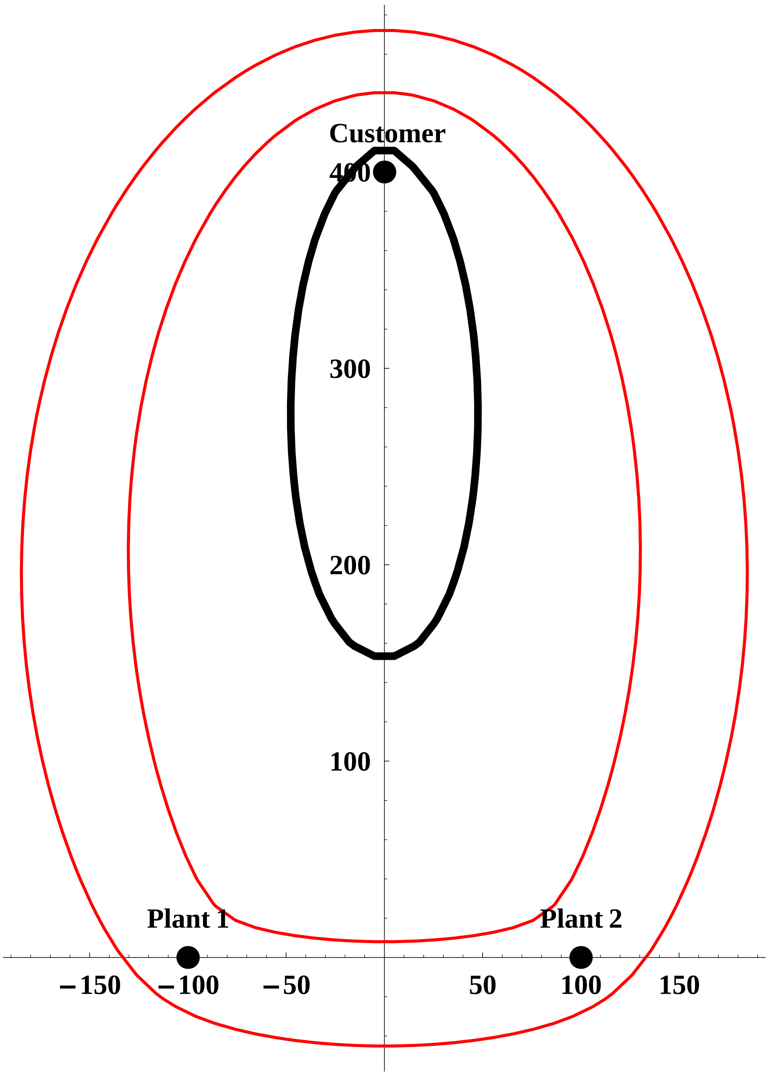


Figure A.21: 2D level curves $\mathcal{L}_{0.5}(c)$ of the profit function ; Case with $a_2 = b = 1$ and $a_1 = 0.8$

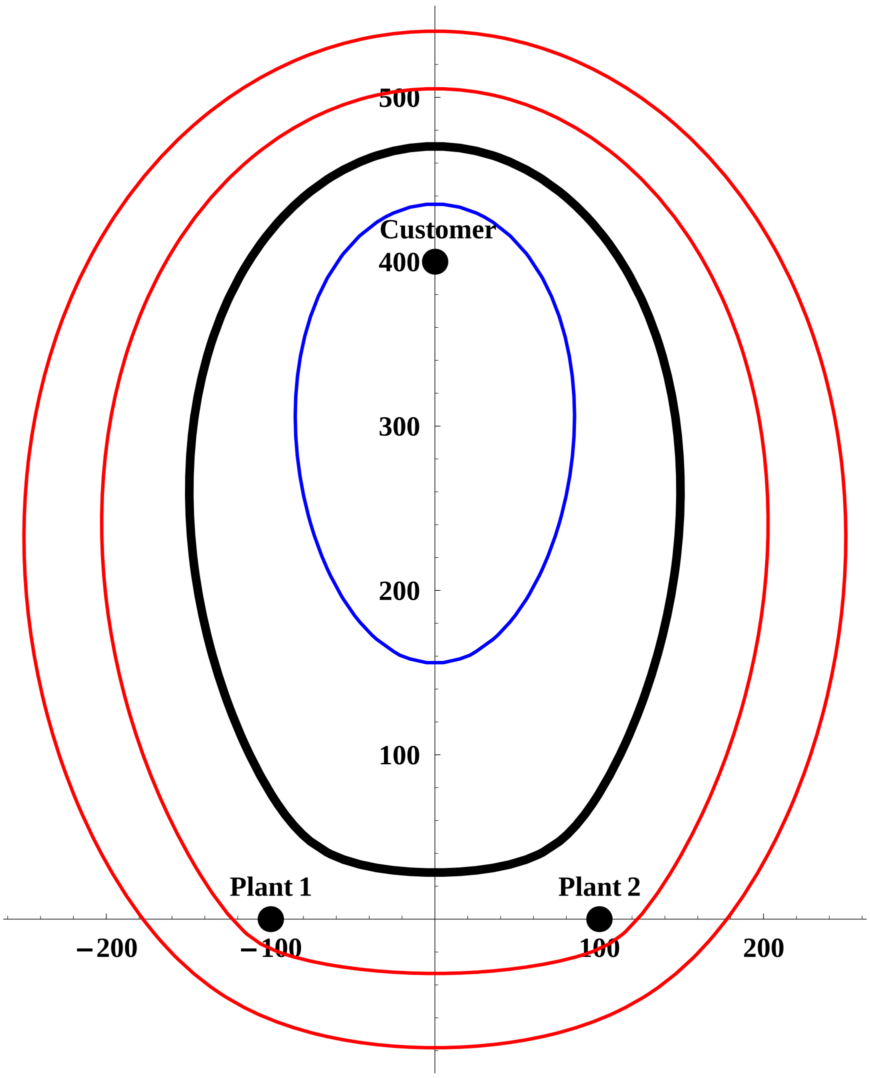


Figure A.22: 2D level curves $\mathcal{L}_{0.5}(c)$ of the profit function ; Case with $a_1 = a_2 = b = 0.8$

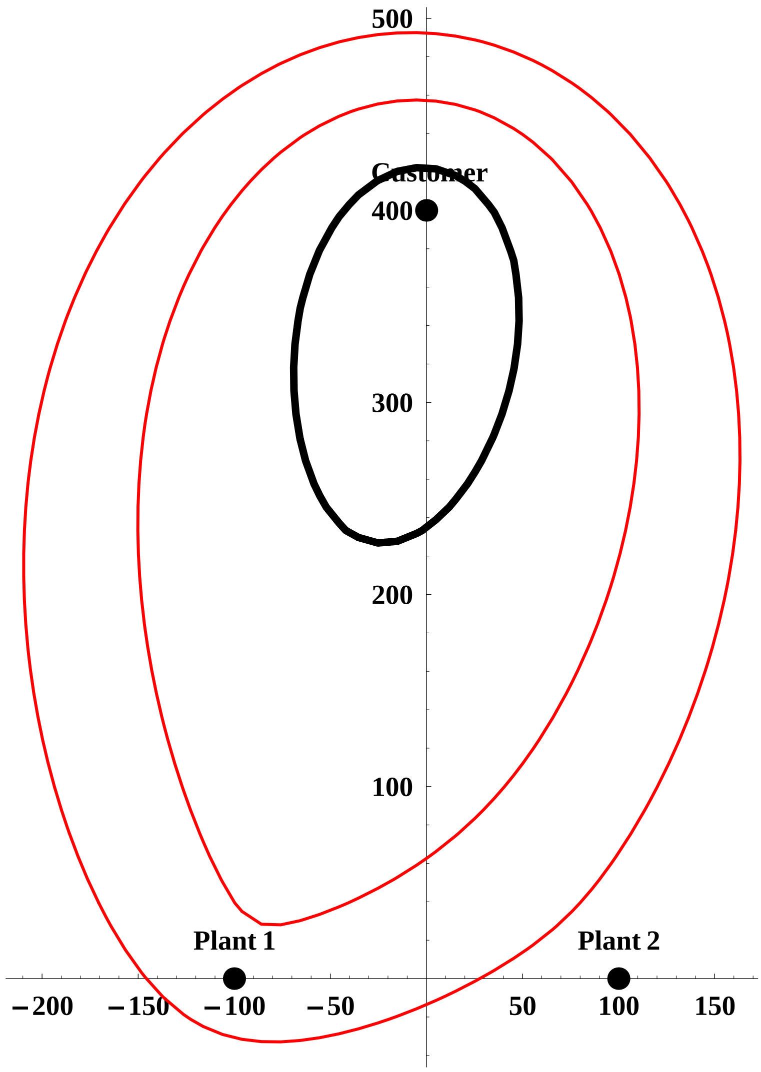


Figure A.23: 2D level curves $\mathcal{L}_{0.8}(c)$ of the profit function ; Case with $a_1 = a_2 = b = 0.8$

Extensions

The methodology followed in section (A.3.1) and (A.3.2) may be used to develop more complex models. For instance, we have generalized our model to M customers ordering in average some mixed orders made of products produced in N several plants.

Many variations of our basic model may be developed. For instance, we have generalize the model to the problem of localizing a platform for a portfolio of customers, based on a flow cartography.

For instance, we have used it to study the interest of a small platform to serve a given customer portfolio. Figure (A.24) illustrates a complex case: our model may be used for a numerous number of plants and customers. Colors indicate the gain associated to the creation of a platform in a zone.

Finally, we may determine whether or not it is profitable to add a new platform in a given supply chain made of plants, existing platforms and customers.

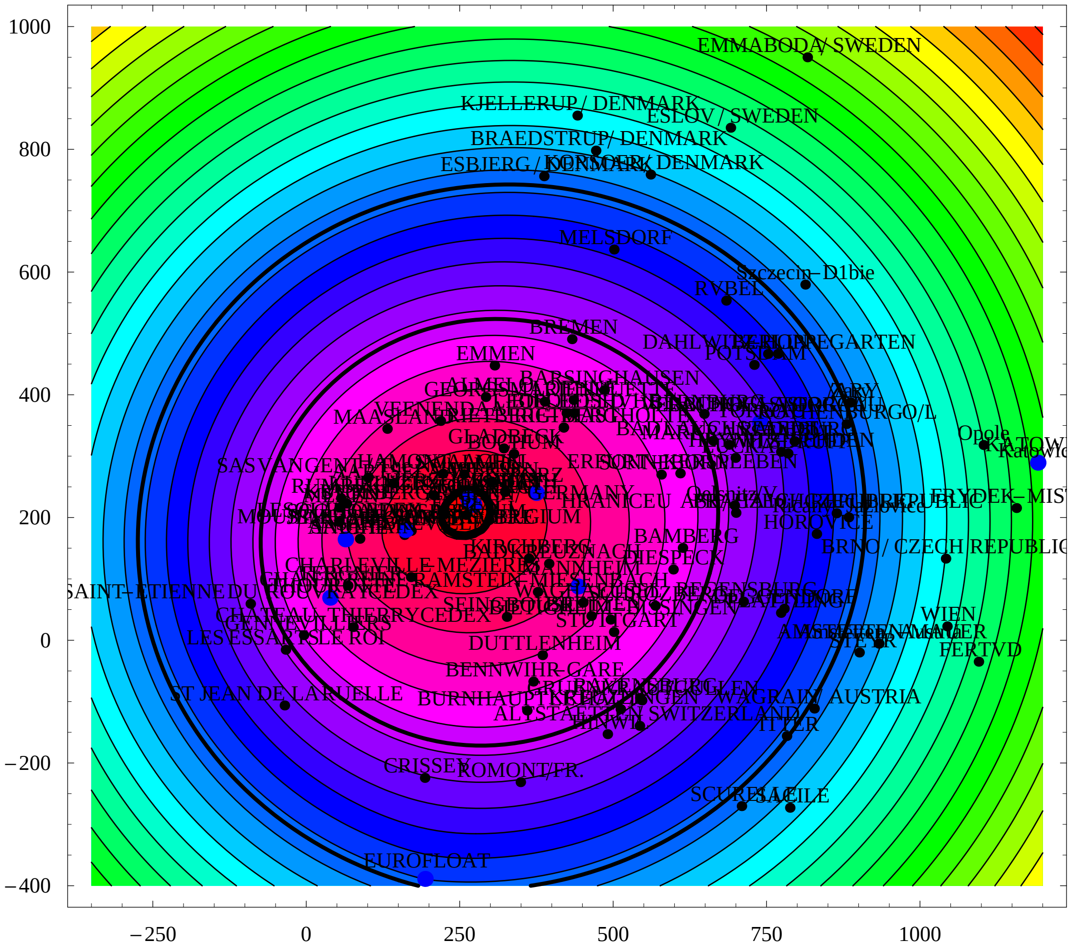


Figure A.24: Localization of a platform in a complex supply chain with given plants and customers

A.3.3 Conclusion on mixed origin deliveries

Mixed origin deliveries serve on past data around 40% on the total amount of mixed orders (and thus represent nearly 20% of the total deliveries). Let us focus now on their real structure.

If we compute for each order the proportion of the majority product, we can study the usual structure of the truck: is it usually equally distributed or not?

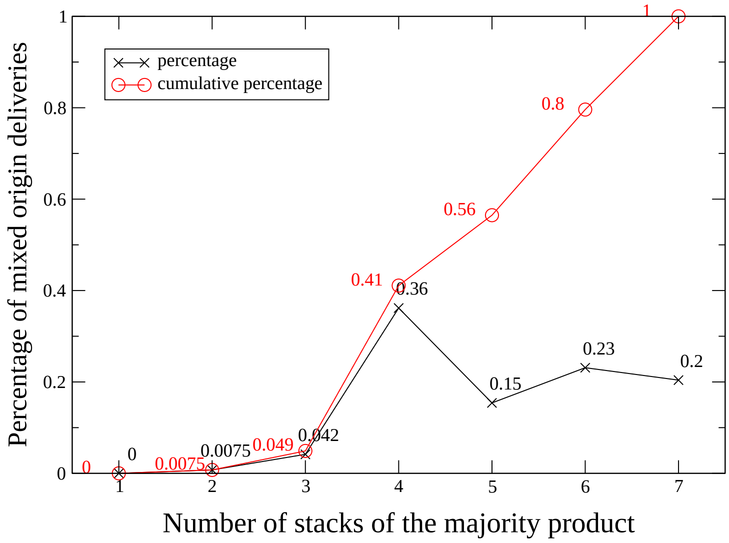


Figure A.25: Structure of mixed origin deliveries

We recall[20] that a truck contains eights stacks of glass, and so at most eight different products. Without surprise, Figure (A.25) shows that a third of the mixed origin deliveries are equally distributed (most of those orders have only two origins), whereas two third have a majority origin ($p \geqslant 0.75$).

Thus, it seems that an easy rule could perform well: for each order, the biggest producer sends the final truck to the customer. Remaining products are sent by transfers from the closest competent plant to the shipping one.

If we consider the global structure of demand, we have shown in section (A.1) that globally mixed trucks are made of a majority of standard products (high volume) filled with low volume one. In addition, most of plants are able to produce standards.

It comes from our simple model that under the full truckload order assumption, each plant should replenish specialties from other plants and send it to customers of its zone.

[20]see assumption (7) described in paragraph (1.4) on page 18

As a conclusion, we have pointed out that a platform is hardly valuable under the present full truckload assumption because of the need to replenish it with standard products which constitute on average the majority in the truckload. The key rule under this assumption appears to be the minimization of standard product transportation.

However, we have used the very strong assumption[1] of perfect replenishment within plants: we lead a discussion about this issue in the paragraph (A.4.1).

Finally, we propose in section (§ A.4.2) a discussion about a new organization that breaks partially the full truckload assumption: we introduce the concept of specialties' platform.

A.4 Prospects on interesting points

A.4.1 The replenishment of low volume products

In the sections (A.2) and (A.3) we have assumed that the replenishment was satisfying the assumption (39) described in paragraph (A.2) on page 275: it was possible to transport a non-full truck at the same cost as a full one.

In addition, we did not mention the differences between the two policies to trigger a transfer of products between plants:

1. we may use it in **a make to order fashion**. In this case, how should we replenish a plant that asks for a stack of a given specialty in order to mix it with its own standards?

    - On the one hand we would like to use a full truckload to keep a low transportation cost, but the way to fill in the truck is not trivial: we risk to fall in the case described on Figure (A.3) in paragraph (A.1), in which we transport a lot of standard products between plants. In this case, it

[1] i.e. assumption (39) described in paragraph (A.2) on page 275

would have been cheaper to send it following option 1 of Figure (A.2) in which the plant producing the specialty sends directly the final delivery to the customer without any mixed origin.

- On the other hand we benefit from several key factors:
  - We concentrate the inventory of specialties in their production plant and thus minimize the required safety stock.
  - We could imagine a kind of cross docking for the transferred specialty: we can indeed avoid to putting it into the inventory of the sending plant, saving thus some handling costs.

2. we can use **a make to stock policy**.

   - On the one hand it becomes easier to deal with the full truck upstream flow because several specialties can be replenished simultaneously.
   - On the other hand we push inventories in every plant, and obviously quickly increase the safety stock required to maintain the service level target (due to the loose of risk pooling in a disaggregate inventory).

To deal with this complexity, we propose a discussion for the specific Saint-Gobain Glass business.

Firstly, **there are many flows that are invisible in our study**. Many necessary flows indeed stem from the different transformation lines in several plants. For instance, float glass stacks are the raw materials for the laminating lines and the coating lines, while they are also finished goods. As explained in the paragraph (1.3), transformation lines are all on a float plant site.

It happens that the corresponding float plant do not produce the required components for the transformation line. In this case, a transfer of float glass between plants is mandatory. Thus, **we may imagine that supply chain managers try**

to match those flows and the specialty flows due to replenishment for mixed origin deliveries.

Secondly, each plant produces its own specialties. We guess that each transfer **truck could be filled with other low volume products, rather than with standard products**. This is obviously possible if the ordering plant keep inventories of specialties. In a replenishment-to-order policy, it is difficult to be done and increases lead times. Considering that each order delivery time is a compromise between the customer and its seller, we guess that it is still sometimes possible.

Thirdly, **it happens that some trucks travel empty between two plants**. Presently the reloading rate is indeed not at one hundred percent. In this case we can use these empty trucks to transfer few stacks of specialty without increasing the global cost. Considering the high rate of transfers, it is highly probable that a truck is traveling between two plants at a given time.

Finally, **it appears difficult to solve this question in a general way, and we even think that it is not possible**. We recommend that **every couple of plants studies together to determine what is their own best replenishment policy for involved low volume products**. It is indeed highly probable that global low volume products are locally high volume ones, or at least that demands between two products are correlated.

The objective of the decision must be:

- to keep the inventory level of transferred products as low as possible in the target plant when a make to stock policy is necessary

- to minimize useless transfers of standard products

- to try to fill the used trucks while minimizing the number of empty travels between plants

However in special cases (especially for products characterized by a high obsolescence) in which an aggregate inventory is mandatory and no full truckload or free replenishment is possible, a direct delivery to the customer is still the best solution, even if we send high volume products from further than possible (option 1 of Figure (A.2)).

As a conclusion, the replenishment of specialties must be studied carefully, locally, according to the expectations of the customer. What is its longest acceptable lead time? Is that possible to respect it with a make to order replenishment policy? How to fill in trucks without standard products?

In the next paragraph, **we focus on geographical areas where customers belong to the group (ex: France)**. We then try to imagine a new organization, in which we divide the mixed order question into two distinct logistic channels dedicated to their own products. To do so, let us imagine that we relax the full truckload assumption for the low volume products.

A.4.2 The concept of a specialties' platform

The analysis of the past demand data emphasized the structure of mixed orders. It appears that high volume products are often mixed because of the low volume ones. The customer does not need a lot of every class B product, and under the full truckload assumption (see section (§1.4)) he fills its order with standard products. Therefore, Saint-Gobain Glass need either to transfer the given product from its origin to the closest plant (where it is mixed with locally produced standards, as described by option 2 of Figure (A.2)) or to send the delivery from the class B origin (option 1 of the same figure).

In an area where we have many customers and few plants, we need to mix many trucks in far-off plants. This constraint stems directly from the customer optimization of low demand product replenishment cycles, given its limited inventory capacity.

In France, the firm owns most of its customers. Thus, **we propose a new organization in which we separate logistics of high volume products and logistics of specialties**. This new pattern implies a new commercial policy (pricing, incentives, etc). It globally consists on **stiffening the offer on high volume products while relaxing it on specialties**.

First, we agree with customers of the area to split products which are locally consumed into two categories: standards (denoted class A) and specialties (class B).

We propose them to deliver specialties by stack with a short delivery lead time, without any full truck ordering constraint. On the contrary, we force them to order full trucks of standards with present delivery lead time. Figure (A.26) illustrates the flows of the new organization.

We improve the global flows existing between present facilities with a new optimized location used as a platform for the specialties. **This new node concentrates all the replenishment of low volume products, from producing plants**. We try to minimize standards passing through the platform. It is clear that volume captured by the platform is by nature weak. One has to focus on that fact to check if the full truck replenishment of the platform is realistic from few plants.

We imagine that the **reloading rate**[21] could be also important on this facility, allowing a discount on upstream transportation costs.

Thus, we eliminate transfers between plants which were mostly involved by the mixing operations. Therefore we only use direct shipping for standard products, by full truckloads.

To deliver specialties from the platform to customers, **we imagine a system of**

[21] see discussion in paragraph (1.4)

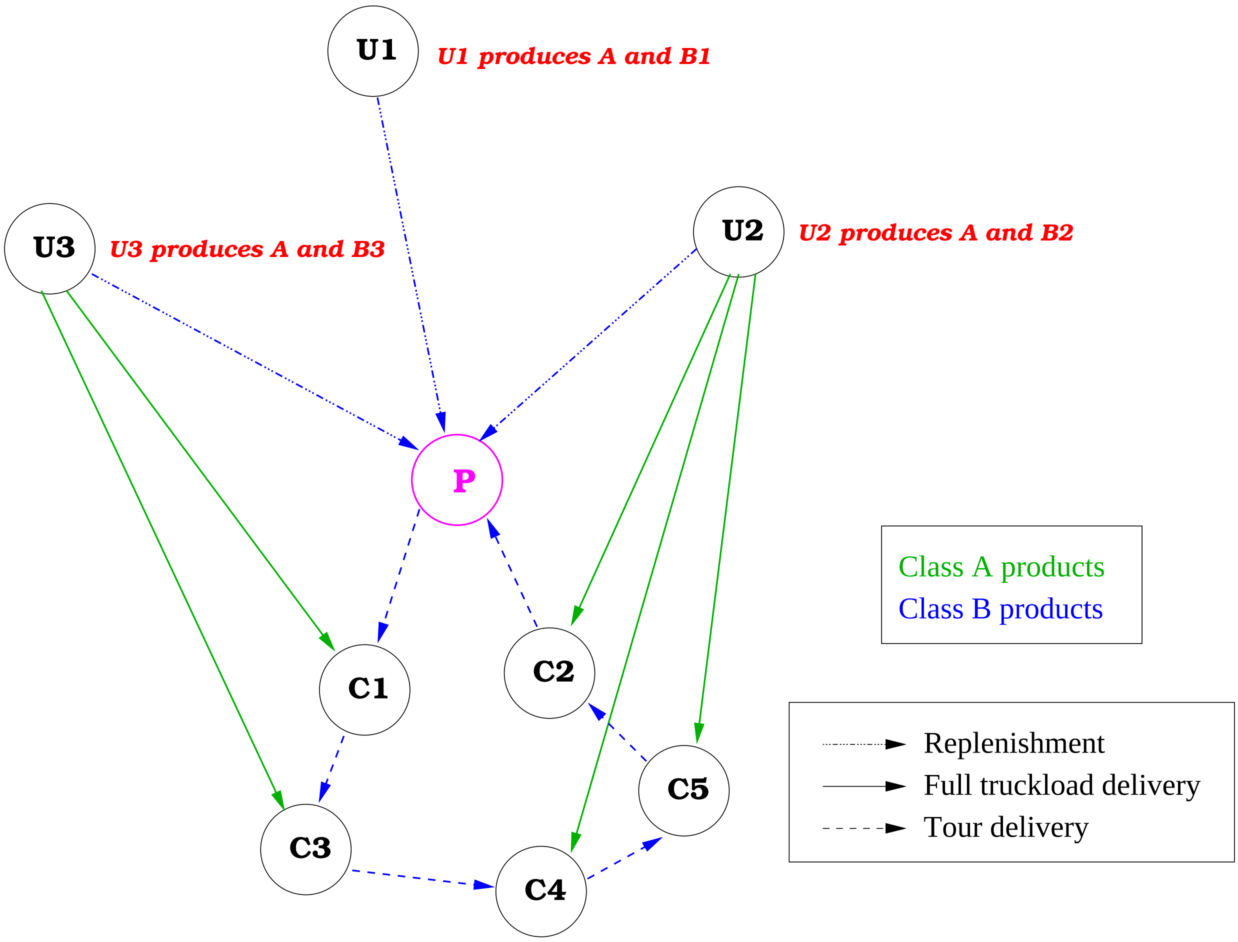


Figure A.26: Illustration of the platform of specialties

delivery tour (Saint-Gobain Glass Logistics has estimated the transportation cost of such a tour around one and a half the default cost). For instance the platform manager could find an agreement with its customers about a prefixed tour schema once upon several months. We think that it can make sense if we compare it to a round trip payment for each delivery.

To master the increasing transportation cost we may set up **a pricing policy encouraging full trucks** (special incentive) of specialties. Being able to command all its specialties at the same time, the customer could work on this synchronization. A new kind of orders would probably appear, mixing laminated glass, printed one and rare float references.

More possible flexibility and shorter lead time on the replenishment of critical products would surely **decrease the inventory level of each subsidiary** (customer). This may be globally interesting if we consider the global supply chain. If we use our idea of **decreasing the size of sold stack on low volume obsolescent products**, we can reduce global inventory.

Maintaining carefully an inventory as low as possible on the platform, we would in addition **aggregate at the same location all former disaggregate inventories** due to former make to stock transfers. This could probably help us to manage the inventory of specialties in a global zone. Nowadays, there is no centralized policy, and each plant uses its own way of serving the mixed orders.

We imagine that the inventory management in plants would be made easier because of its simplification: no more replenishment of specialties produced wherever. The inventory manager of each plant could focus on its produced goods and on the required delivery. On the contrary, the platform manager would be focused on its core business, which is dealing with safety stocks levels, optimizing the operational delivery planning, etc.

Of course, we could also imagine **a positive impact of the improvement of responsiveness and service on the market shares**. Unfortunately, this rise of service is difficult to quantify.

In a nutshell, this kind of new organization can not be seen as an example wherever the area we are focusing on. However, under particular assumptions managers must consider it as a real option. It can potentially simplify the global organization, by matching the particular demand nature of low volume products with a particular supply chain. **For a given customer portfolio, the more numerous the mixed origin deliveries, the more interesting the platform of specialties**.

We point out that this concept of platform may be used in an existing facility, such as a plant inventory.

A.4.3 Logistics of DLF: concept of industrial platform

In this part we deal with the logistics that is particular to DLF products. So far, we have been working exclusively on PLF products. The DLF format is smaller than the PLF one (see chapter 2). Two DLF may be cut into a PLF. As specified in paragraph (1.3), DLF may be cut either directly on the float line or on specific off line cutting machine.

Thus, there is a new logistic concept that may be introduced: the industrial platform. Such a platform is a classical one with additional skills that add some value to the product (such as cutting machines, packaging machines, etc).

For instance, in the glass business, all export sales are sent by sea: in this case glass must be cut in DLF and a special heavy packaging is required. Each glass stack is packaged into a wood box after being wrapped into a plastic or a metallic thin film.

Up to now, each plant owned its own little cutting and packaging machines that were operated by warehouse workers. Figure (A.27) describes the present organization. Of course, classical DLF references are cut directly on the float line according to a make to stock policy. On the contrary, special DLF references are produced in make to order policy, and requires off-line machines.

As a comparison, Figure (A.28) describes the organization corresponding to the creation of an industrial platform: the platform captures all the production of make to order references. It centralizes all flows that pass through either the off-line cutting or the packaging (or both) machines.

An industrial platform may be valuable for several reasons:

- It centralizes in one specialized industrial site skills that were formerly decentralized in several plants. Global yields may be increased by investments on more powerful machines: due to a bigger task, each machine requires a bigger capacity. At least, the pay back of an identical machine is quicker: any centralization allows to pool the risks due to uncertainties of different markets.

- It allows to subcontract minor skills and thus offers more flexibility at a reasonable cost: a specialized subcontractor may balance his workforce on several customers' planning. Thus, the global workforce cost may be cheaper. Depending on the contract negotiation, it may become really favorable to change former fixed costs into pure variable ones. In addition, subcontracting make it really easier to change the frequency of work teams depending on the forecasts.

- In an evolving environment and a strongly competitive market, it may be safer not to invest in buildings to be more reactive in case of crisis. An external subcontracted platform may be closed within few months without a big loss.

- It simplifies the supply chain of concerned products: Sellers know directly where to give the order. In addition, they may announce better lead times, provide a better quality (due to a bigger task: continue quality process is easier to implement) and thus capture a bigger market.

Remark 9 *It may occur that capacities of means of conveyance of upstream and downstream flows on the platform be different. The difficult case lies in a bigger potential upstream flow. For instance, let us take trucks: inloaders of the upstream PLF flow can load 22 Tons of glass whereas classical trucks of the downstream DLF flow load only 20 Tons.*

In such a case, we underline that it may be cheaper not to fill in completely the upstream resources to avoid having a products' cemetery on the platform: the inventory management is the success key factor of the platform manager. Easy rules such as an incoming truck equals an outgoing one may simplify the management of the platform.

Thus, many different logistic organizations may be implemented in Saint-Gobain

Glass. Every new concept is depending on the context, the market, and it may evolve with time.

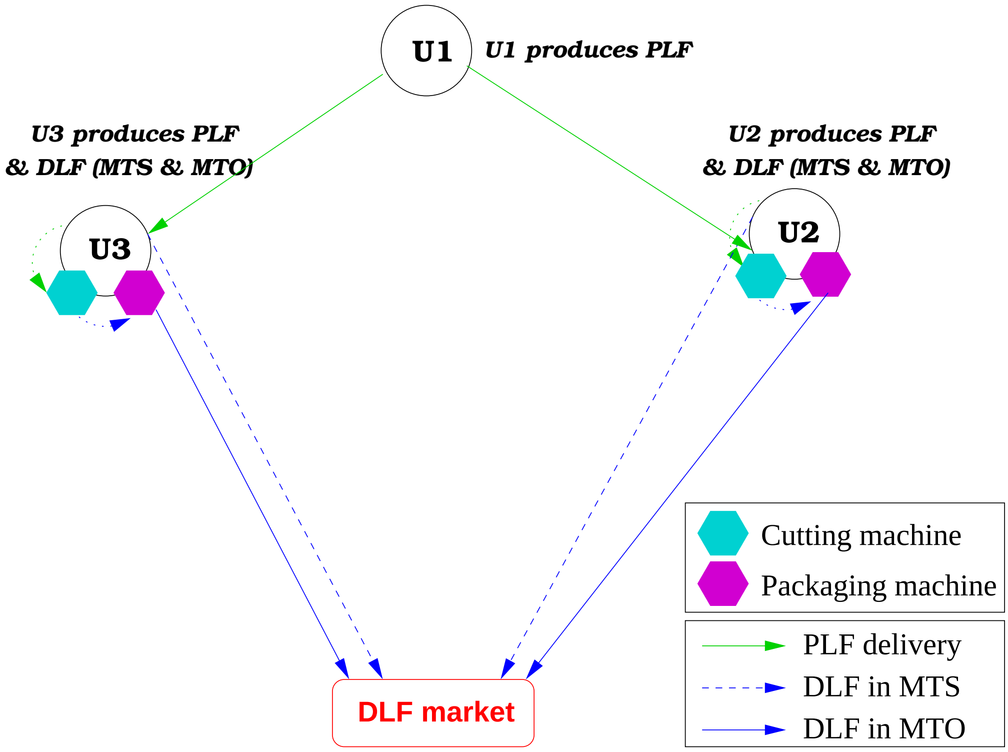


Figure A.27: Illustration of the present DLF logistics

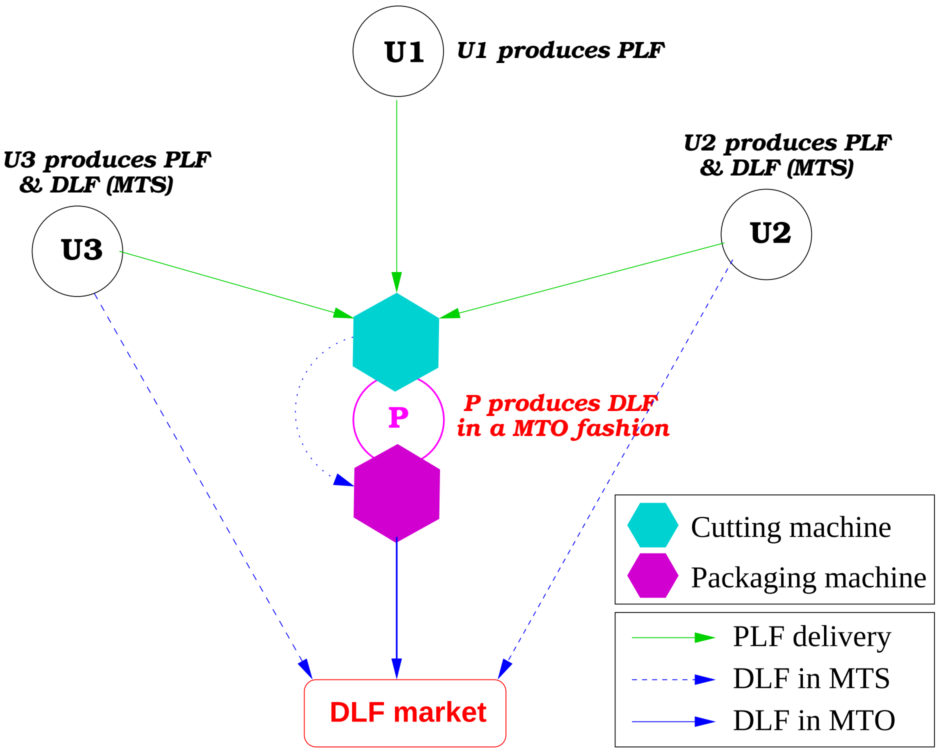


Figure A.28: Illustration of the industrial platform specialized in DLF logistics

A.5 Details on C++ code methods

A.5.1 Implementation of the greedy clustering method

Here is the C++ implementation of the function BasicClustering($\alpha^{[k]}$, N) we used in section (§3.2.4).

```
bool PARTITION_CLUSTERS::BasicClustering(const float alpha
                                          ,const short &NbreMaxPF)
{
        bool stop = false;
        vector<_CLUSTER> c(this->Partition);
        vector<_CLUSTER> nc(this->Partition);
        vector<bool> cused;
        bool flag = true;
        short boucle = 1;
        while((flag == true)&&(nc.size()<=c.size()))
        {       c.clear();
                c=nc;
                nc.clear();
                cused.clear();
                for(int k=0;k<c.size();k++){cused.push_back(0);}
                flag = false;
                int i=0;
                while(i<c.size())
                {       int j=i+1;
                        while(j<c.size())
                        {       if((cused[i] != 1)&&(cused[j] != 1)
```

```
                        &&(Distance(c[i].GetCentre(),c[j].GetCentre())
                           < alpha))
                        {       nc.push_back(Fusion(c[i],c[j]));
                                cused[i] = 1;cused[j] = 1;
                                flag = true;
                        }j++;
                }i++;
        }
        for(k=0; k<c.size(); k++)
        {       if(cused[k] != 1)
                {nc.push_back(c[k]);}
        }
        boucle ++;
        if(nc.size()<=NbreMaxPF)
        {       flag = false;
                stop = true;}
    }
    this->Partition.clear();
    this->Partition = nc;
    return stop;
}
```

A.5.2 Implementation of our heuristic

In section (§3.2.4) we define both a perturbation which modifies the partition by moving randomly some points to neighbor clusters and a perturbation that moves randomly some cluster centers.

The first one is based on the function `ElementaryMovementSmallClusters` whereas the second one uses the function `ElementaryMovementBigClusters` defined as follows.

```
vector<_CLUSTER> PARTITION_CLUSTERS::ElementaryMovementSmallClusters(
                                  const double &TailleMaxCluster
                                 ,const vector<_CLUSTER> &Ini
                                 ,const int &NbMovesMax
                                 ,const bool &ConstantCardinal) const
{
    vector<_CLUSTER> RES(Ini);
    float X=0;
    unsigned int NbMoves = 0;
    if(NbMovesMax <= 1)
        NbMoves =1;
    else
    {   while(NbMoves < 1)
        {       X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                NbMoves = floor(X*NbMovesMax);}}
    vector<int> TabChosenPoint,TabOriginCluster,TabDestCluster;

    while(TabChosenPoint.size() < NbMoves)
    {
        int ChosenPoint = 0,OriginCluster=0;
        bool Restart = true;
        int Compteur = 0;
        vector<int> PossibleDestClusters;
        while((Restart == true)&&(Compteur < 1000))
        {       Restart = false;
```

```
        X=(double)rand()/(double)RAND_MAX;
        while(X==1)
                X=(double)rand()/(double)RAND_MAX;
        OriginCluster = floor(X*RES.size());
        if(ConstantCardinal == true)
        {       while(RES[OriginCluster].GetSizeListePoints()==1)
                {       X=(double)rand()/(double)RAND_MAX;
                        while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                                OriginCluster = floor(X*RES.size());}}

        X=(double)rand()/(double)RAND_MAX;
        while(X==1)
                X=(double)rand()/(double)RAND_MAX;
        ChosenPoint = floor(X*RES[OriginCluster].GetSizeListePoints());

        if(ConstantCardinal == false)
                PossibleDestClusters.push_back(-1);

        vector<POINT> Centres = TabCentres(outstream,RES);
        int NbVoisins = 5;
        vector<int> ClustersVoisins = FindVoisins(Centres
                ,OriginCluster,NbVoisins);

        for(int c=0;c<ClustersVoisins.size();c++)
        {      if(TailleMaxCluster > 0)
                {
                      if(RES[ClustersVoisins[c]].GetPoids()
                        + RES[OriginCluster].ReadPoint(ChosenPoint)
                           ->GetVolumeGlobal()
                        <= TailleMaxCluster)
                        {PossibleDestClusters.push_back(ClustersVoisins[c]);}
                }
                else
                PossibleDestClusters.push_back(ClustersVoisins[c]);
        }
        if(PossibleDestClusters.size() == 0)
                Restart = true;
}
if(PossibleDestClusters.size() > 0)
{
        X=(double)rand()/(double)RAND_MAX;
        while(X==1)
                X=(double)rand()/(double)RAND_MAX;
        int DestCluster = floor(X*PossibleDestClusters.size());
        DestCluster = PossibleDestClusters[DestCluster];
        TabChosenPoint.push_back(ChosenPoint);
        TabOriginCluster.push_back(OriginCluster);
        TabDestCluster.push_back(DestCluster);

        if(DestCluster == -1)
        {
                _CLUSTER NewOne(*RES[OriginCluster].ReadPoint(ChosenPoint));
                if(RES[OriginCluster].GetSizeListePoints() == 1)
                {       vector<_CLUSTER> NEW;
                        for(int c=0;c<RES.size();c++)
                        {if(c!= OriginCluster){NEW.push_back(RES[c]);}}
                        //RES.erase(RES.find(RES[OriginCluster]));
                        RES = NEW;}
                else
                {       RES[OriginCluster].DeletePoint(ChosenPoint);
                        RES[OriginCluster].MoveCenterTOwp(outstream);}
                        RES.push_back(NewOne);}
        else
        {       RES[DestCluster].PushBackPoint(
                      *RES[OriginCluster].ReadPoint(ChosenPoint));
                RES[DestCluster].MoveCenterTOwp(outstream);
                if(RES[OriginCluster].GetSizeListePoints() == 1)
                {       vector<_CLUSTER> NEW;
                        for(int c=0;c<RES.size();c++)
```

```
                                        {if(c!= OriginCluster){NEW.push_back(RES[c]);}}
                                        RES = NEW;}
                                else
                                {       RES[OriginCluster].DeletePoint(ChosenPoint);
                                        RES[OriginCluster].MoveCenterTOwp(outstream);}
                        }
                }
        }
        return RES;
}



vector<_CLUSTER> PARTITION_CLUSTERS::ElementaryMovementBigClusters(
                                 const double &TailleMaxCluster
                                ,const vector<_CLUSTER> &Ini
                                ,const int &NbMovedPF
                                ,const bool &ConstantCardinal
                                ,const double &SautMax) const
{
        vector<_CLUSTER> RES(Ini);
        float X=0;
        int NbMoves = NbMovedPF;
        if(NbMoves == -1)
        {NbMoves = 0;}

        while(NbMoves < 1)
        {
                X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                NbMoves = floor(X*RES.size());}

        vector<int> TabChosenCluster;
        vector<int>::iterator it;
        while(TabChosenCluster.size() < NbMoves)
        {       X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                int ChosenCluster = floor(X*RES.size());
                it = find(TabChosenCluster.begin()
                         ,TabChosenCluster.end(),ChosenCluster);
                if(it == TabChosenCluster.end())
                        TabChosenCluster.push_back(ChosenCluster);}

        vector<double> TabChosenAngle;
        while(TabChosenAngle.size() < NbMoves)
        {       X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                double ChosenAngle = floor(X*360);
                TabChosenAngle.push_back(ChosenAngle);}

        vector<double> TabChosenJump;
        while(TabChosenJump.size() < NbMoves)
        {       X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                double ChosenJump = floor(X*SautMax);
                TabChosenJump.push_back(ChosenJump);}
        for(int m=0;m<TabChosenCluster.size();m++)
        {       double Xmove = TabChosenJump[m] *cos(TabChosenAngle[m]);
                double Ymove = TabChosenJump[m] *sin(TabChosenAngle[m]);
                double Xini = RES[TabChosenCluster[m]].GetCentre().GetX();
                double Yini = RES[TabChosenCluster[m]].GetCentre().GetY();
                RES[TabChosenCluster[m]].SetCentre(Xini+Xmove,Yini+Ymove);}
        return RES;
}
```

Appendix B

Details on PLANEO

B.1 Mono-attribute case

The object of this section is to present various extensions of the MILP introduced in section (§4.4.1).

B.1.1 Case with sequence independent set up times and costs

He we simplify the previous MIP when set-ups are not sequence-dependent. With the same notation, we just redefine a few parameter notation:

- $\alpha \in \mathcal{A} = [1, A]$ *still denotes a type of strictly positive changeover cost*
  - $C(i)$ is the function that gives the cost of any changeover towards i:$\{j \neq i \rightarrow i\}$.
  - $\mathcal{T}^C(i)$ is the function that gives the type of cost of any changeover $\{j \neq i \rightarrow i\}$.
  - *Remark:* C_α *is still the cost of type* α. We notice that $C(i) = C_{\mathcal{T}^C(i)}$.
- $\beta \in \mathcal{B} = [1, B]$ *still denotes a type of changeover duration.*

– $T(i)$ is the function that gives the duration of any changeover towards i: $\{j \neq i \to i\}$.

– $\mathcal{T}^T(i)$ is the function that gives the type of duration of any changeover $\{j \neq i \to i\}$.

– Remark: T_β *is still the duration of a changeover of duration type* β. We notice that $T(i) = T_{\mathcal{T}^T(i)}$. *We keep the definitions of the set* $\mathcal{B}^*$ *and functions* $T_\beta(t)$ *and* $T_\beta^N(t)$.

- In the same way, we redefine two sets of products i:

  – $S_C(\alpha) = \{i \in \mathcal{P} \quad \text{s.t.} \quad \mathcal{T}^C(i) = \alpha\}$

  – $S_T(\beta) = \{i \in \mathcal{P} \quad \text{s.t.} \quad \mathcal{T}^T(i) = \beta\}$

We can use globally the same MIP, except for a few changes. Firstly, we cancel the constraints (4.6) because every changeover is now possible. Secondly we need to replace some constraints: the former ones (4.7) to (4.13) are changed into (B.1) to (B.7).

$$\forall \beta \in \mathcal{B}^*,\ \forall i_2 \in S_T(\beta),\ \forall t \in [2, N]$$

$$\sum_{i_1 \neq i_2} \sum_{k=1}^{T_\beta(t)} y_{i_1}^{t-k} \quad \leq \quad (1 - y_{i_2}^t) \times M \tag{B.1}$$

$$\forall \beta \in \mathcal{B}^*,\ \forall i \in S_T(\beta),\ \forall t \in [2 + T_\beta, N]$$

$$T_\beta \times \left(y_{i_2}^t + \sum_{i_1 \neq i_2} y_{i_1}^{t-T_\beta-1} - 1\right) \quad \leq \quad \sum_{k=1}^{T_\beta} v_\beta^{t-k} \tag{B.2}$$

$$\forall i_1,\ \forall \beta,\ \forall t \in [T_\beta + 2, N]$$

$$\left(y_{i_1}^{t-T_\beta-1} + \nu_\beta^{t-T_\beta}\right) - 1 \;\leq\; \sum_{i_2 \in S_{\mathcal{T}}(\beta)\,;\, i_2 \neq i_1} y_{i_2}^t \tag{B.3}$$

$$\forall \beta,\ \forall t \in [2, N-1] \qquad T_\beta^N(t) \times (\nu_\beta^t - \nu_\beta^{t-1}) \;\leq\; \sum_{k=0}^{T_\beta^N(t)-1} \nu_\beta^{t+k} \tag{B.4}$$

$$\forall \alpha,\ \forall i_2 \in S_{\mathcal{C}}(\alpha), \forall t \in [2, N - T_\beta]$$

$$y_{i_2}^{t+\mathcal{T}(i_2)} + \sum_{i_1 \neq i_2} y_{i_1}^{t-1} - 1 \;\leq\; w_\alpha^t \tag{B.5}$$

$$\forall t \in [2, N] \qquad \sum_\alpha w_\alpha^t \;\leq\; \sum_i y_i^{t-1} \tag{B.6}$$

$$\forall \alpha,\ \forall t \qquad w_\alpha^t \;\leq\; \sum_{i \in S_{\mathcal{C}}(\alpha)} y_i^{t+\mathcal{T}(i)} \tag{B.7}$$

To be exhaustive, if we decide **to simplify it by using the Boolean variables ν^t** (indicating whether the line is on transition between two products or not) **instead of variables ν_β^t**, we need to modify few constraints. Firstly, we cancel the constraints (B.4) and exchange constraints (4.14) by (4.25). Secondly, we transform former constraints (B.2) and (B.3) into (B.8) and (B.9).

We modify some former notation and introduce a new one:

- *We keep the same definitions for* T_M *and for the function* $\nu^t(\delta t)$.
- We introduce: $S_T(i, \delta t) = \{\, i \quad \text{s.t.} \quad T_{\mathcal{J}^{\mathcal{T}(i)}} = \delta t \,\}$

The simplified constraints with the new variables ν^t are as follows:

$$\forall \beta \in \mathcal{B}^*,\ \forall i_2 \in S_{\mathcal{T}}(\beta),\ \forall t \in [2 + T_\beta, N]$$

$$T_\beta \times \left(y_{i_2}^t + \sum_{i_1 \neq i_2} y_{i_1}^{t-T_\beta-1} - 1\right) \leq \sum_{k=1}^{T_\beta} v^{t-k} \tag{B.8}$$

$$\forall i_1,\ \forall \delta t \in [1, T_M],\ \forall t \in [\delta t + 2, N]$$

$$\left(y_{i_1}^{t-\delta t-1} + \left(\sum_{k=1}^{\delta t} v^{t-k} - (\delta t - 1)\right)\right) - 1 \leq \sum_{i_2 \in S_T(i,\delta t)\,;\, i_2 \neq i_1} y_{i_2}^t + v^t(\delta t) \tag{B.9}$$

We point out that an accurate analysis of the industrial data is always a prerequisite before the use of our model. We hope indeed that it is possible to use as less as possible various types of changeover costs and times.

B.1.2 Case with linear relationship between changeover times and costs

Furthermore, if there is a perfect linear relation between the changeover duration and its associated cost, it is not worth using such a complicated model. Let us denote ($\mathcal{H} : \mathbb{N} \to \mathbb{R}$) the linear cost function which associates a cost to a changeover duration. In this case indeed, we only need the y_i^t and v^t variables (constraints (B.5) to (B.7) are forgotten), and thus the objective (4.3) becomes (B.10):

$$\min \left(\sum_t \left(\mathcal{H}(v^t) + \sum_i h_i \times \frac{I_i^t + I_i^{t-1}}{2}\right)\right) \tag{B.10}$$

Similarly, if all changeover times are equal to zero we do not need to introduce the set of Boolean decision variables v^t. We then only use the variables w_α^t to count the cost of changeovers. Finally, whether changeovers have neither cost nor time, we only use the production variables y_i^t. In these two later extreme cases, equation (4.14) enforces that only one item is produced per time period, corresponding to a small bucket time model.

B.1.3 Improvement of the the production line model

To be more realistic and to capture some specificities of the glass production, we may consider the capacity of the production line as a bounded decision variable on which we may add a given availability of the line. We propose to use the following notation:

- We introduce new parameters:
  - $A : \mathcal{T} \to [0, 1]$ is a function defining the availability $A(t) \in [0, 1]$ of the line, which is the proportion the capacity that we may use during each time period t.
  - C_i^m and C_i^M are the minimal and maximal possible net tonnage capacities on the line for the product i.
- We define new real variables $P_i^t \in [C_i^m, C_i^M]$ representing the production of product i during time period t.

To take this improvement into account in the model described in section (§4.4.1), we replace equations (4.4) and (4.5) by the constraints (B.11) to (B.14).

$$\begin{aligned}
\forall i,\ \forall t \qquad I_i^{t-1} + P_i^t &= I_i^t + D_i^t && \text{(B.11)}\\
\forall i \qquad \sum_t P_i^t &\geq Q_i^m && \text{(B.12)}\\
\forall i,\ \forall t \qquad P_i^t &\leq C_i^M \times A(t) \times y_i^t && \text{(B.13)}\\
\forall i,\ \forall t \qquad P_i^t &\geq C_i^m \times A(t) \times y_i^t && \text{(B.14)}
\end{aligned}$$

On the one hand, equations (B.11) and (B.12) correspond directly to former ones (4.4) and (4.5). On the other hand, inequalities (B.13) and (B.14) ensure the relationship between integer variables y_i^t and real ones P_i^t, taking into account the given availability of the line.

B.2 Improvements of the multi attribute model

In an industrial context, we have met additional goals compared to the model we propose in section (4.4.2), such as to impose the final product, or to authorize an interruption in a campaign. Thus, we propose to add to the set of real products one fake product corresponding to no valuable production, and by convention we note it the product $i = 0$. By transforming the inequalities (4.35) into equalities (B.15), we write down that the line is either producing a product or inactive.

$$\forall i,\ \forall t \qquad \sum_{i=0}^{P} Z_i^t = 1 \tag{B.15}$$

In the same way:

- if we wish to authorize some production campaign breaks, we switch (A.9) to (B.16).

$$\forall \omega,\ \forall l,\ \forall t \qquad Z_0^t + \sum_{i \in \mathcal{S}_l} Z_i^t \geq 1 - \Omega + \sum_{\omega} y_{\mathcal{M}^*_{\omega l}}^{t^{[\omega]}} \tag{B.16}$$

- we can force either the first or the final product by adding constraints (B.17), (B.18) and (B.19). We denote $i^{[ini]}$ and $i^{[fin]}$ the imposed initial and final products.

$$\begin{aligned}
& Z_{i^{[ini]}}^1 &=&\ 1 && \text{(B.17)}\\
\forall t \in [1, N-1] \qquad & Z_{i^{[fin]}}^t &\geq&\ \Big(\sum_{k=1}^{N-t} Z_0^{t+k} - (N - t - 1)\Big) - Z_0^t && \text{(B.18)}\\
& Z_{i^{[fin]}}^N + Z_0^N &=&\ 1 && \text{(B.19)}
\end{aligned}$$

B.3 Extensions of our model: Options on inventory costs and constraints

In the beginning of chapter 4, we claimed that various options were possible concerning the inventory costs. Firstly, we let the user authorize or not optional costs associated to imperfect service, such as **backorder costs**. Secondly, we can include a **handling cost** (in €/unit), which corresponds to the long and expensive handling operations to put the glass in and out of the warehouse.

We can easily make our model evolve to captures these new costs. Besides, we can add easy linear constraints, such as storage capacity constraints, at a more or less accurate level depending on the decision level.

We introduce the following notation:

- Parameters:
  - h_p is the inventory cost of the real product p, whereas b_p is its backorder cost and m_p its handling cost. By convention, this handling cost is the sum of the inventory entrance and exit costs.
  - The warehouse is decomposed in a set of various areas $\mathcal{S} = s \in [1, S]$ with limited inventory space capacities C_s. We denote $\mathcal{P}_R(s)$ is the set of real products stored in the area s. We assume that each product is stored in one unique area:

$$\forall p \in \mathcal{P}_R \qquad \exists\,!\, s \in \mathcal{S} \quad \text{s.t.} \quad p \in \mathcal{P}_R(s)$$

- Decision variables:
  - $I^t_{[+]p}$ is the on-hand inventory position of real product p at the end of time period t. **This continuous variable is non-negative**.

- $I^t_{[-]p}$ is the numbers of back-orders of real product k at the end of time period t. **This continuous variable is non-negative**.
- M^t_p is the quantity of real product p entering into inventory during t (continuous non-negative variable).

We just need to change the total inventory cost in the objective function (B.20), to make evolve the inventory balance equations (4.39) to (B.21) and to add some structure constraints between new variables ((B.22) to (B.25)) as well as to add the inventory capacity constraints (B.26).

$$\min\Big(\sum_{\omega}\sum_{\alpha^{[\omega]}}\sum_{t^{[\omega]}} C_{\alpha^{[\omega]}} \times w^{t^{[\omega]}}_{\alpha^{[\omega]}} + \sum_{t}\sum_{p}\big(h_p \times \frac{I^t_{[+]p} + I^{t-1}_{[+]p}}{2} + b_p \times \frac{I^t_{[-]p} + I^{t-1}_{[-]p}}{2} + m_p \times M^t_p\big)\Big) \tag{B.20}$$

$$\forall p,\ \forall t \qquad I^{t-1}_{[+]p} - I^{t-1}_{[-]p} + R^t_p = I^t_{[+]p} - I^t_{[-]p} + D^t_p \tag{B.21}$$

$$\forall p,\ \forall t \qquad M^t_p \geq I^t_{[+]p} - I^{t-1}_{[+]p} \tag{B.22}$$

$$\forall p,\ \forall t \qquad M^t_p \geq 0 \tag{B.23}$$

$$\forall p,\ \forall t \qquad I^t_{[+]p} \geq 0 \tag{B.24}$$

$$\forall p,\ \forall t \qquad I^t_{[-]p} \geq 0 \tag{B.25}$$

$$\forall s,\ \forall t \qquad \sum_{p\in\mathcal{P}_R(s)} I^t_{[+]p} \leq C_s \tag{B.26}$$

B.4 Details of the PLANEO implementation

B.4.1 Lower bound of changeover costs

We have noticed that the linear relaxation of our MIP is pretty bad: to help CPLEX, we compute a lower bound of the changeover costs before solving the model. In this section we describe the more general algorithm we propose, based on notation introduced in part (§4.4.1). The idea of our dynamic program is to determine the cheapest cost of production changeovers allowing to produce at least one campaign of each virtual product required to meet the market demand.

For each attribute ω, we define the following notation:

- the function $C_m^{[\omega]}$ associates to any attribute value $i \in \mathcal{V}^{[\omega]}$ the minimal cost of any changeover towards it: $C_m^{[\omega]}(i) = \min_j C^{[\omega]}(j, i)$.
- the same way, $C_M^{[\omega]}$ associates the maximal changeover cost: $C_M^{[\omega]}(i) = \max_j C^{[\omega]}(j, i)$ when the changeover is defined ($\forall i,\ C_M^{[\omega]}(i) << \infty$).

In addition, we denote Max the function that associates to any real matrix $M = (m_{ij})$ its biggest element ($\mathsf{Max}(M) = \max_{i,j} m_{ij}$) and Nc the function that gives the number of columns of any matrix.

Firstly, we compute (based on demand data) the set of virtual products $\mathcal{V}$ that must be produced: to do so, we determine the list of finished products for which initial inventory level minus the minimal final one is insufficient to fulfill the demand over the time horizon. We add the corresponding virtual product whenever it is not yet in the set $\mathcal{V}$. Each virtual product is nothing but a vector whose element are values taken for each attribute.

The result of this selection is a set $\mathcal{V}$ of N vectors of dimension Ω corresponding to each virtual product. We create thus the matrix A of dimension $(\Omega \times N)$.

Secondly, we create the matrix B of dimension $(\Omega \times \max_\omega V^{[\omega]})$ whose term $a_{\omega j}$ is the minimal changeover cost $C_m^{[\omega]}(j)$ towards the value j if $j \in \mathcal{V}^{[\omega]}$ and (-1) otherwise.

Finally, we compute the vector C of dimension Ω whose terms c_ω corresponds to $C_M^{[\omega]}(j^0)$ whether the initial product is constrained and $\max_j C_M^{[\omega]}(j)$ otherwise.

We then use A, B and C to solve the following dynamic program:

1. Set $\Omega^{(0)} = \Omega$, $A^{(0)} = A$ and $B^{(0)} = B$. Set $k = 0$ and $Lb = 0$.
2. Step k:
   - If $\Omega^{(k)} = 1$ then $A^{(k)}$ and $B^{(k)}$ are single row vectors of dimension N.
     (a) Set $Lb = Lb + A^{(k)}(B^{(k)})^T$.

(b) GO TO (3).

- Else $\Omega^{(k)} = \mathsf{Nc}(A^{(k)})$.
  (a) If $\Omega^{(k)} = 0$, GO TO (3).
  (b) Else:
     i. Set μ and ϕ such that $b_{\mu\phi} = \mathsf{Max}(B^{(k)})$.
     ii. Set $\Phi = \{j \in [1, N] \quad \text{s.t.} \quad a^{(k)}_{\mu j} = \phi\}$.
        – If $\Phi \neq \emptyset$ then:
           * $Lb = Lb + b_{\mu\phi}$
           * Set $\Omega^{(k+1)} = \Omega^{(k)} - 1$; We build the matrix $A^{(k+1)}$ by taking the column vectors $\{A^{(k)}_c \quad \text{s.t.} \quad c \in \Phi\}$ in which we delete the μ^{th} row ; The same way, $B^{(k+1)}$ is obtained from $B^{(k)}$ by deleting the μ^{th} row. We notice that $A^{(k+1)}$ and $B^{(k+1)}$ have $\Omega^{(k+1)}$ rows.
           * Set $k \leftarrow k + 1$. GO TO (2).
        – Else, set $b_{\mu\phi} = -1$. GO TO (2(b)i).

3. We subtract to the obtained result the maximal changeover costs that may be gained through the initial production campaign: $Lb = Lb - \sum_{\omega} c_{\omega}$.

Finally, the constraint (B.27) allows the solver to prove optimality quicker or at least to reduce the obtained gap[1] after a fixed computation duration.

$$\sum_{\omega} \sum_{\alpha^{[\omega]}} \sum_{t^{[\omega]}} C_{\alpha^{[\omega]}} \times w^{t^{[\omega]}}_{\alpha^{[\omega]}} \geq Lb \tag{B.27}$$

B.4.2 Object oriented implementation

In this section we present the practical method we use to implement this model in a usable decision tool, and we particularly focus on the interesting object programming methodology.

In the previous section we have developed a complex mixed integer program. However, it is based on few concepts that can be used in an object development, such as attributes, sub-attributes, products and virtual products. First of all, we have decomposed products into several attributes and sub-attributes. This leads to the distinction between virtual products (a given value for each attribute) and

[1]between the best found and the optimal solutions

products (a given value for each attribute and each sub-attribute). We will integrate this general production planning model into a global integrated model in chapter 6. At this time, we will recognize obvious objects derived from our approach to model the industrial and logistic schema: geographical points, geographical and functional zones, transport resources, etc. Details on this part of the implementation will be found in section (§D.1). We focus here on the production part of the model.

To solve real-life cases, we have developed a tool based on an easy principle: each object owns its own decision variables and constraints, in addition to its usual member methods and data. Thus, we generate our linear model in a very general manner, allowing a great modularity. This allowed us to unify all our models in an open unique software (named ROADEO), offering tremendous possibility to deal with various decisions, as we will discover in it chapter 6.

Figure (B.1) represents the way we have implemented the production planning model. The blue classes (`GLOBAL_JOB` and `ABSOLUTE_PRODUCT`) symbolize two very important concepts which will be linked to upper level of the general integrated model of section (§D.1).

We highlight the parallel structure of the classes: on the one hand, each job (class `GLOBAL_JOB`) is defined by a vector of attributes (`PRODUCT_CRITERION`) and a vector of sub-attributes (`PRODUCT_SUB_CRITERION`) . Each attribute and sub-attribute may take a set of various values. For each attribute, we define the set of possible types of transition time and cost (`TYPE_TIME_TRANSITION` and `TYPE_COST_TRANSITION`), allowing us to define for each production line (`PRODUCTION_LINE`) a set of skills. Products (`PRODUCT`) relative to this job are defined by a virtual product (`VIRTUAL_PRODUCT`), which corresponds to a given value for each attribute, and a given value for each sub-attribute.

On the other hand, we define the set of skills of each production line corresponding to this job. Each line is able to produce a subset of values of each attribute (`SKILL_CRITERION`), among which changeovers (`SKILL_CRITERION_CHANGE`) are either sequence-dependent or not and are characterized by a type of time and a type of cost. The same way, the line has a given range of production capacity for each virtual product that can be produced (`SKILL_PRODUCT` and `SKILL_VIRTUAL_PRODUCT`), associated to a subset of values for each sub-attribute (`SKILL_SUB_CRITERION`). This way, each virtual product corresponds to a subset of products relative to the job. In case of a transformation line, each transformation (`SKILL_PRODUCT_TRANSFO`) is characterized by a set of consumed product (`CONSUMED_PRODUCT`). Finally, each line has a given availability (`SKILL_PERIOD`) and a set of specified over-costs and additional costs (`SKILL_OVERCOST` and `SKILL_ADDITIONNAL_COST`).

In a nutshell, each production line of a given job is defined according to its own skills, which must be a subset of the set of products defined according to the definition of the job by attributes and sub-attributes.

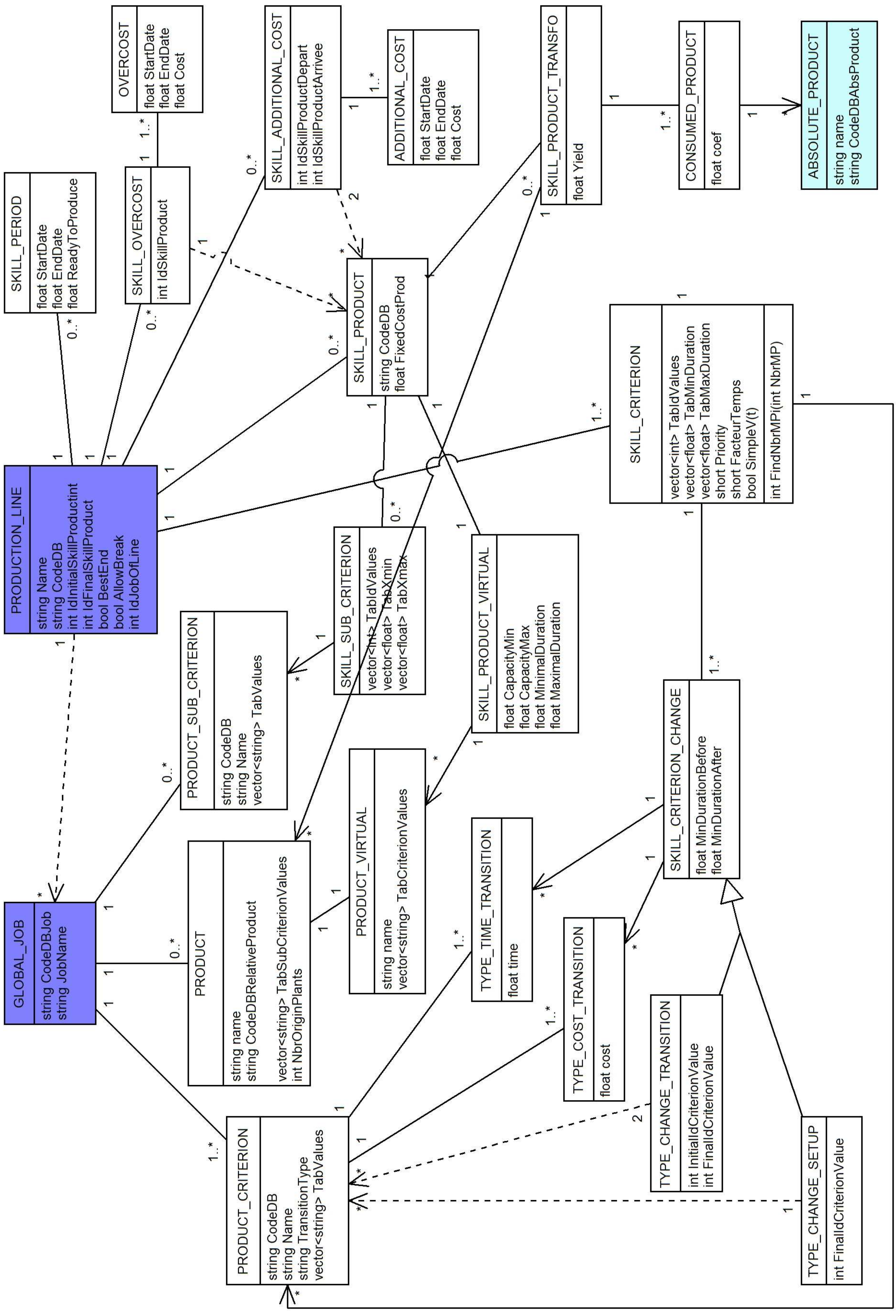


Figure B.1: UML classes of the production model

Appendix C

Details on coating lines

C.1 Exact model for hypothesis 2

It appears that our previous model does not fit problems under hypothesis 2 because we did not make differences between sputtering a unique coat from either successive or non-successive cathodes. Basically, we have adapted our model by focusing more on the metal sequence than on the cathode one. Thus we modify slightly the notation of paragraph (§5.3.2): We replace z_c^i by z_m^i, denoting the use of a cathode of metal m in position i. We keep the same convention: $z_0^i = 1$ for unused positions. The same way integer variables x_{po}^i determine the proportion of each coat sputters by the cathode set on position i.

We introduce new integer variables: n_c^i is an integer variable indicating how many cathodes c are used for covering the required volume of the metal at the i^{th} position. ρ^i is the remaining volume on the set of cathodes used for position i and γ^i is the cost of this remaining volume.

The optimization model is the following:

$$\mathrm{Obj}_1 = \begin{cases} \sum_i \rho^i \\ \sum_i \gamma^i \end{cases} \tag{C.1}$$

$$\mathrm{Obj}_2 = \sum_i \sum_{c>0} n_c^i \tag{C.2}$$

$$\mathrm{Obj}_3 = \sum_i \sum_p \sum_o y_{po}^i \tag{C.3}$$

$$\mathrm{Min} \sum_k \beta_k \times \mathrm{Obj}_k \tag{C.4}$$

$$\forall i, \quad \sum_m z_m^i = 1 \tag{C.5}$$

$$\forall i, \forall p, \forall o, \quad x_{po}^i \leq y_{po}^i \tag{C.6}$$

$$\forall i, \forall p, \forall o, \quad y_{po}^i \leq z_{m_{po}}^i \tag{C.7}$$

$$\forall i, \forall p, \forall o, \quad y_{po}^i \leq 1 - \sum_{m \neq m_{po}} z_m^i \tag{C.8}$$

$$\forall i, \forall p, \forall o, \quad y_{po}^i \times i \leq r_{po}^M \tag{C.9}$$

$$\forall i, \forall p, \forall o, \quad r_{po}^m \leq y_{po}^i \times i + (1 - y_{po}^i) \times N \tag{C.10}$$

$$\forall p, \forall o, \quad r_{p,o}^m \leq r_{p,o}^M \tag{C.11}$$

$$\forall p, \forall o \in [1, 0_p - 1], \quad r_{p,o}^M < r_{p,o+1}^m \tag{C.12}$$

$$\forall p, \forall o, \quad \sum_i x_{po}^i = 1 \tag{C.13}$$

$$\forall p, \forall o, \quad \sum_i y_{po}^i \leq N_{po} \tag{C.14}$$

$$\forall i \in [1, n-1], \quad z_0^i \leq z_0^{i+1} \tag{C.15}$$

$$\forall i, \forall c, \quad n_c^i \leq z_{m_c}^i \times N_c \tag{C.16}$$

$$\forall i, \forall m, \quad z_m^i \leq \sum_{c \text{ s.t.} m_c = m} n_c^i \tag{C.17}$$

$$\forall c, \quad \sum_i n_c^i \leq N_c \tag{C.18}$$

$$\sum_i \sum_c n_c^i \leq N \tag{C.19}$$

$$\forall i, \quad \rho^i = \sum_c n_c^i \times V_c - \sum_p \sum_o x_{po}^i \times \frac{\nu_{po}}{\phi} \tag{C.20}$$

$$\forall i, \quad \gamma^i = \sum_c n_c^i \times c_{m_c} \times V_c - \sum_p \sum_o x_{po}^i \times c_{m_{po}} \times \frac{\nu_{po}}{\phi} \tag{C.21}$$

Solving this subproblem gives us the final design of the line: we know exactly which cathode is used at each position, as well as which homogeneous (made of the same metal) cathode set is used for each transformation coat. To decrease the computation time we use a trick identical to the former one. We increase step by step the maximal size of the cathode sequence, until we determine n_c^* the smallest feasible cathode sequence. We use thus identical parameters g and L to limit our search. We obtain identical results: the procedure without virtual metal is quicker than the original model.

C.2 Forbidding some operations: From hypotheses 1 and 2 to 3 and 4

In order to transform previous models by forbidding some operations (sputtering a coat through several cathodes for hypothesis 3 and through several non-successive ones for 4), we modify slightly the notation of paragraph (§5.3.2) and (§C.1): we do not need variables x_{po}^i any more, because only one may be non zero (and thus equal to 1).

To illustrate this evolution, we apply it to the model of (§C.1) of hypothesis 2. It becomes the exact model for hypothesis 4. Basically, former equations (C.6) disappear and we replace continuous variables x_{po}^i by integer ones y_{po}^i in equations (C.13), (C.20) and (C.21), which become (C.32), (C.38) and (C.39)

Of course, the third objective in the objective function is now meaningless. The optimization model becomes:

$$\mathrm{Obj}_1 = \begin{cases} \sum_i \rho^i \\ \sum_i \gamma^i \end{cases} \tag{C.22}$$

$$\mathrm{Obj}_2 = \sum_i \sum_{c>0} n_c^i \tag{C.23}$$

$$\mathrm{Min} \sum_k \beta_k \times \mathrm{Obj}_k \tag{C.24}$$

$$\forall i, \quad \sum_m z^i_m = 1 \tag{C.25}$$

$$\forall i, \forall p, \forall o, \quad y^i_{po} \leq z^i_{m_{po}} \tag{C.26}$$

$$\forall i, \forall p, \forall o, \quad y^i_{po} \leq 1 - \sum_{m \neq m_{po}} z^i_m \tag{C.27}$$

$$\forall i, \forall p, \forall o, \quad y^i_{po} \times i \leq r^M_{po} \tag{C.28}$$

$$\forall i, \forall p, \forall o, \quad r^m_{po} \leq y^i_{po} \times i + (1 - y^i_{po}) \times N \tag{C.29}$$

$$\forall p, \forall o, \quad r^m_{p,o} \leq r^M_{p,o} \tag{C.30}$$

$$\forall p, \forall o \in [1, 0_p - 1], \quad r^M_{p,o} < r^m_{p,o+1} \tag{C.31}$$

$$\forall p, \forall o, \quad \sum_i x^i_{po} = 1 \tag{C.32}$$

$$\forall i \in [1, n-1], \quad z^i_0 \leq z^{i+1}_0 \tag{C.33}$$

$$\forall i, \forall c, \quad n^i_c \leq z^i_{m_c} \times N_c \tag{C.34}$$

$$\forall i, \forall m, \quad z^i_m \leq \sum_{c \text{ s.t.} m_c = m} n^i_c \tag{C.35}$$

$$\forall c, \quad \sum_i n^i_c \leq N_c \tag{C.36}$$

$$\sum_i \sum_c n^i_c \leq N \tag{C.37}$$

$$\forall i, \quad \rho^i = \sum_c n^i_c \times V_c - \sum_p \sum_o y^i_{po} \times \frac{\nu_{po}}{\phi} \tag{C.38}$$

$$\forall i, \quad \gamma^i = \sum_c n^i_c \times c_{m_c} \times V_c - \sum_p \sum_o y^i_{po} \times c_{m_{po}} \times \frac{\nu_{po}}{\phi} \tag{C.39}$$

C.3 Data of the realistic example of section (5.5.2)

Here is the data of the example we deal with in section (5.5.2).

Transformation			Metallic coats[1]			
Id p	Name	Surface[2]	o	Metal	Thickness[3]	Division
1	Planitherm	2000	1	Silver	2.1	4
			2	Gold	1.2	3
			3	Titanium	1	5
			4	Silver	1	4
2	Planistar	1000	1	Silver	2.1	3
			2	Gold	1.3	6
			3	Titanium	1	3
			4	Steel	1	3
			5	Gold	0.7	4
			6	Titanium	2	4
			7	Silver	4	6
3	PlaniNew	1000	1	Gold	2	4
			2	Platinum	1	4
			3	Titanium	1	6
			4	Steel	2.4	4
			5	Gold	1	4
			6	Platinum	2	4
4	PlaniMiege	1000	1	Steel	1.5	4
			2	Titanium	1	6
			3	Steel	2.4	4
			4	Gold	1	4
			5	Platinum	0.75	4
			6	Titanium	0.5	6
			7	Steel	1	4
5	PlaniMiege2000	1000	1	Gold	1	4
			2	Platinum	0.75	4
			3	Titanium	0.5	6
			4	Steel	1	4
			5	Silver	2	6
			6	Steel	1.5	4
			7	Titanium	1	6
			8	Steel	2.4	4

Table C.1: Definition of transformations of the section (5.5.2) example

[1] D being the reference dimension
[2] in D^2
[3] in D

Metal	Volume
Silver	300
Silver	500
Silver	1000
Silver	2000
Silver	3000
Silver	4000
Titanium	4000
Titanium	3000
Titanium	2500
Titanium	1000
Titanium	400
Gold	100
Gold	500
Gold	1000
Gold	2500
Gold	3500
Steel	500
Steel	750
Steel	1000
Steel	1500
Steel	2000
Platinum	500
Platinum	750
Platinum	1000
Platinum	1500
Platinum	2000

Table C.2: On-hand cathodes in a realistic case

C.4 Final heuristic C++ method

Here is the main part of the final heuristic C++ methods of section (5.5.2). We use succesively two local search methods:

```
SPUTTERING_LINE HEURISTIC::LocalImprovementsBig(ostream& outstream
                , double &IniCost, const int &hypothese, const int &Moteur
                , const int &ObjVolCost, const bool &UsingXipo
                , const OPTI_PARAMETERS &Parameters, SPUTTERING_LINE &Line)
{       bool change=false,MetalChange=false;
        SPUTTERING_LINE RES(Line),NewLine;
        double NewCost=0;
        vector<vector<int> > TabPossibleIN;
        for(int m =0; m < this->TabMetals.size() ;m++)
        {       vector<int> TabPossible(FindElectrodesOfMetal(m,this->TabCathodes));
                TabPossibleIN.push_back(TabPossible);}
        int count = 0;
```

```
        while(count < (int)RES.TabElectrodes.size()*(int)50)
        {       count++;
                double X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                int IdPosition = floor((double) X * (double)RES.TabElectrodes.size());

                int compte=0;
                vector<int> TabNbrRemplace;
                TabNbrRemplace.push_back(-1);
                while((compte<TabPossibleIN[RES
                        .TabElectrodes[IdPosition].GetIdMetal()].size())
                        &&(compte<2))
                {compte++;TabNbrRemplace.push_back(compte);}
                X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                int NbreRemplace = floor((double) X * (double)TabNbrRemplace.size());

                compte=0;
                vector<int> TabIdNewCathodes;
                if(TabNbrRemplace[NbreRemplace] <= 0)
                {TabIdNewCathodes.push_back(-1);}
                else
                {       while(compte<TabNbrRemplace[NbreRemplace])
                        {       X=(double)rand()/(double)RAND_MAX; // entre 0 et 1
                                        while(X==1)
                                        X=(double)rand()/(double)RAND_MAX;
                                int IdE = floor((double) X * (double)TabPossibleIN[RES
                                        .TabElectrodes[IdPosition].GetIdMetal()].size());
                                TabIdNewCathodes.push_back(TabPossibleIN[RES
                                        .TabElectrodes[IdPosition].GetIdMetal()][IdE]);
                                compte++;}
                }
                NewLine = this->SwitchCathodes(outstream
                                ,RES,IdPosition
                                ,TabIdNewCathodes);
                double NewCost = this->ComputeSolutionCost(outstream,IniCost
                                ,hypothese,Moteur,ObjVolCost
                                ,UsingXipo
                                ,Parameters,NewLine);
                if((NewCost != -1)&&(NewCost < IniCost))
                {       count = 0;
                        RES = NewLine;
                        IniCost = NewCost;      }
        }
        return RES;
}


SPUTTERING_LINE HEURISTIC::LocalImprovementsSmall(ostream& outstream
                , double &IniCost, const int &hypothese, const int &Moteur
                , const int &ObjVolCost, const bool &UsingXipo
                , const OPTI_PARAMETERS &Parameters, SPUTTERING_LINE &Line)
{       bool change=false,MetalChange=false;
        SPUTTERING_LINE RES(Line),NewLine;
        double NewCost=0;
        vector<vector<int> > TabPossibleIN;
        for(int m =0; m < this->TabMetals.size() ;m++)
        {       vector<int> TabPossible(FindElectrodesOfMetal(m,this->TabCathodes));
                TabPossibleIN.push_back(TabPossible);}
        int count = 0;
        while(count < (int)RES.TabElectrodes.size()*(int)0.7)
        {
                count++;
                double X=(double)rand()/(double)RAND_MAX;
                while(X==1)
                        X=(double)rand()/(double)RAND_MAX;
                int IdPosition = floor((double) X * (double)RES.TabElectrodes.size());
                for(int p=0;p<
                        TabPossibleIN[RES
```

```
                        .TabElectrodes[IdPosition].GetIdMetal()].size();p++)
                {       NewLine = this->SwitchCathodes(outstream
                                ,RES,IdPosition,TabPossibleIN[RES.TabElectrodes[IdPosition].GetIdMetal()][p]);
                        double NewCost = this->ComputeSolutionCost(outstream,IniCost
                                        ,hypothese,Moteur,ObjVolCost
                                        ,UsingXipo
                                        ,Parameters,NewLine);
                        if((NewCost != -1)&&(NewCost < IniCost))
                        {       RES = NewLine;
                                IniCost = NewCost;      }}}
        return RES;
}
```

Appendix D

Details on ROADEO

D.1 Implementation based on Object programming

From a practical point of view, we present how we did develop the ROADEO software. We use the C++ code and follow the object programming fashion. Definitions of classes and relationships between them come directly from the physical concept we have introduced so far.

Figure (D.1) summarizes for instance the structure of classes we use to model the supply chain. So far, we introduced various concepts which have corresponding classes (GEOGRAPHICAL_ZONE, FUNCTIONAL_ZONE, INVENTORY_ZONE, NON_INVENTORY_ZONE, PRODUCTION_LINE). The important difference between relative products (PRODUCT) and absolute ones (ABSOLUTE_PRODUCT) appears clearly: first one belong to a job (GLOBAL_JOB) for which we define various production lines (see explanation in section (B.4.2) whereas second ones are real-life products. Demand (PRODUCT_FLUX) and inventory levels (PRODUCT_STOCK) are referred to absolute products, so as are the flows between zones whose structure is details on Figure (D.2).

In the next paragraph, we introduce the notation and the MILP we build to solve questions based on our model.

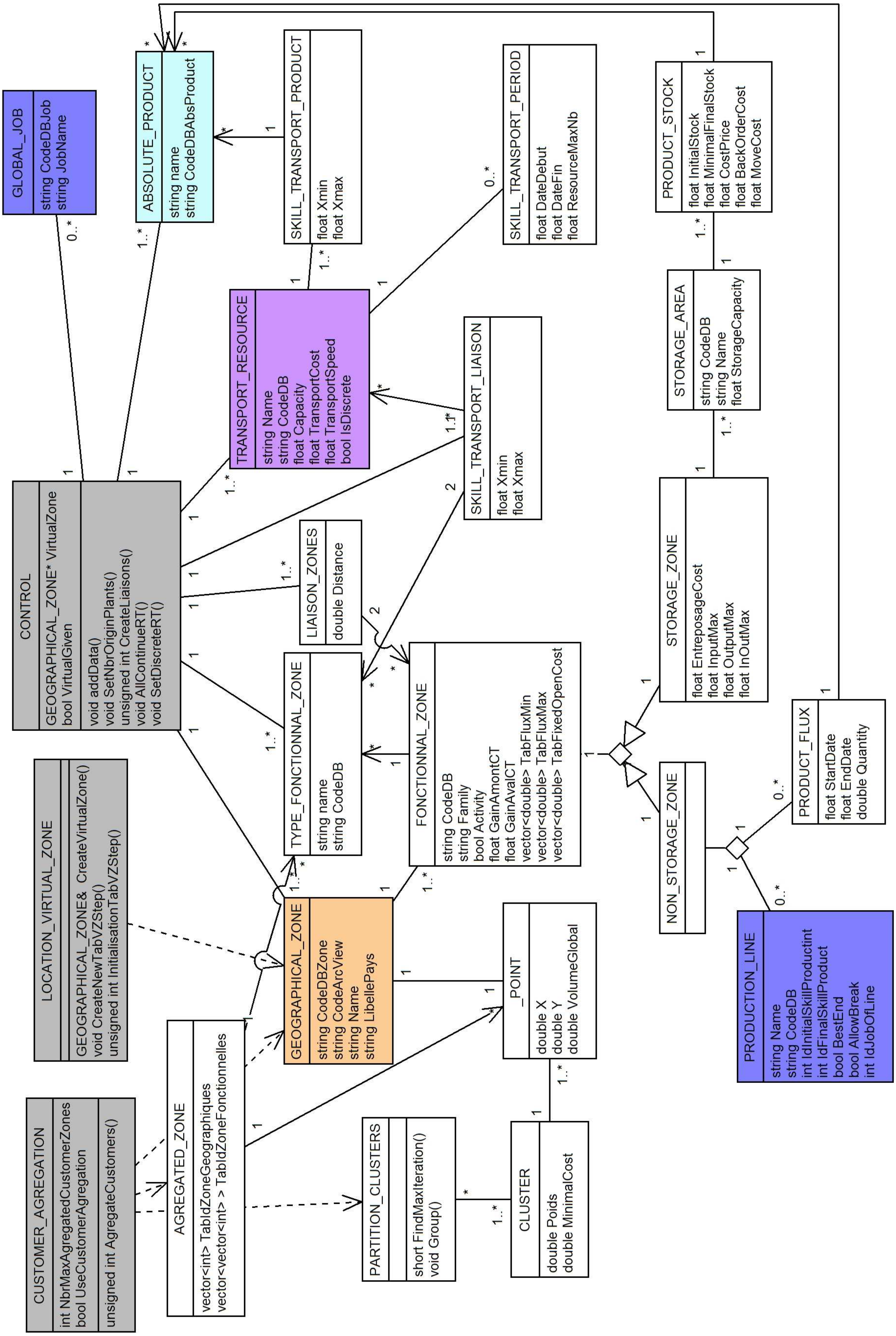


Figure D.1: UML structure of ROADEO on the supply chain design issues

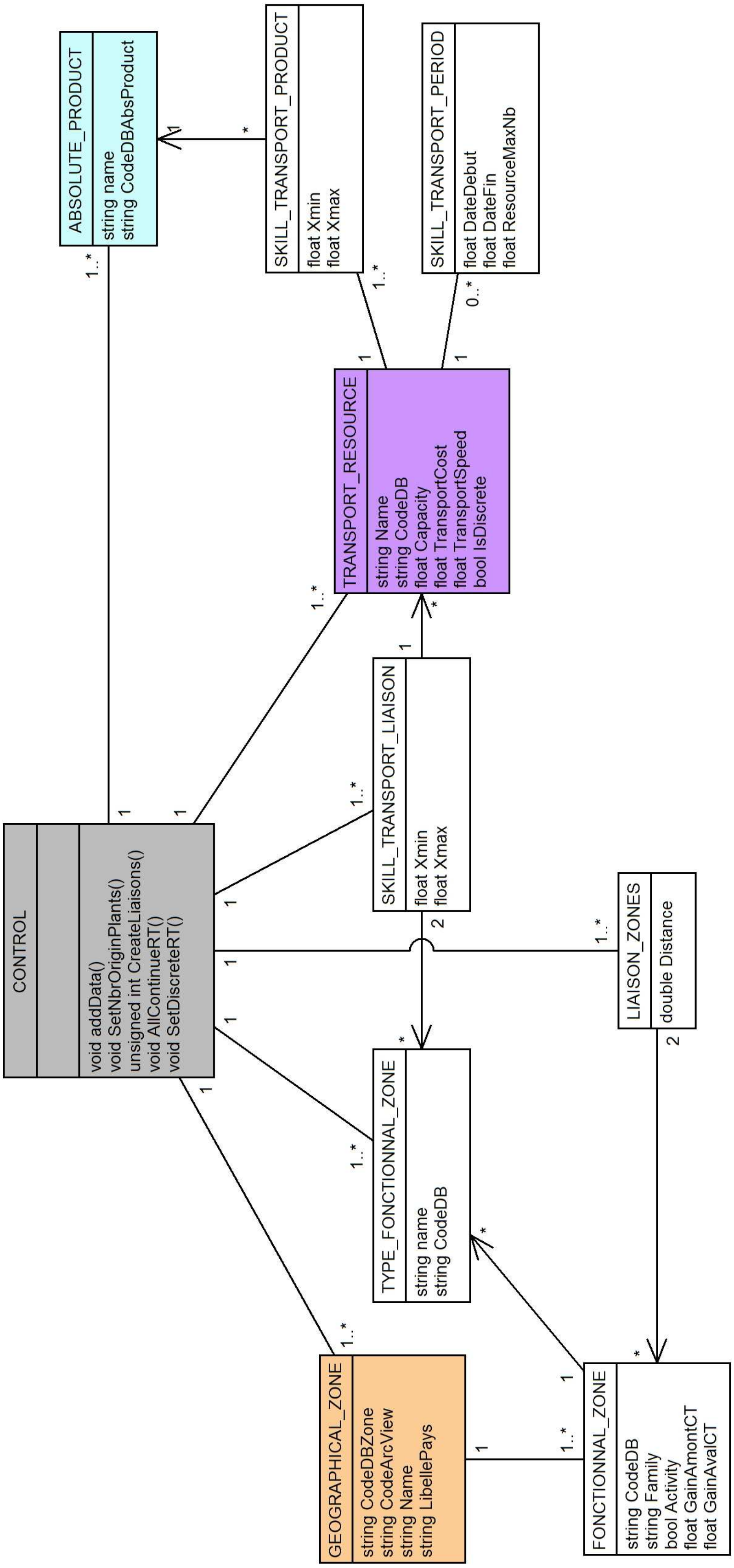


Figure D.2: UML structure of ROADEO on transportation issues

D.2 Extension to a model for operational transportation optimization

We may need to apply our model to a more operational decision level on transportation operations in which we need to capture more precisely product flows. At this point, demand and production are assumed to be deterministic and dynamic over a short time horizon divided into small time periods. For instance, we would like to take into account improvements such as transportation times. Slight improvements of our model allow us to deal with such an operational issue.

On the one hand, we need to relax our assumption of infinite transportation resource speed $\forall\psi \quad \nu_\psi^T << \infty$. To capture the transportation time, we introduce the function that gives the number of time periods required for the resource ψ to link the departure node of the link l (of distance d_l) to the arrival one.

$$T_\psi^l = \left\lfloor \frac{d_l}{\nu_\psi^T} \right\rfloor$$

For instance, if the resource leaves the departure node at time t, it is delivered to its destination at time $t + T_\psi^l$. We implicitly assume that transportation lead times are in multiple of time period.

On the other hand, we do not assume any more that demand must be fulfilled on time. To capture this phenomenon, we define customers as geographical zones on which there is not only non storage zone with a dynamic known demand but also a storage zone which represent an inventory (which may be virtual). We set the inventory costs at the customer as the penalty for delivering in advance, while the backorder costs capture the penalty for being late.

Given forecast productions and demands on a short-term horizon, ROADEO may thus become a decision-tool for operational transportation issues, under the assumption that there is no circle delivery routes. It captures not only economies of scale due to transportation resources but also delivery lead times.

Conclusion and Perspectives

Nowadays, a highly competitive environment makes of the service level impact a fundamental element for formerly production oriented companies. Global supply-chain thinking gives a new impetus to transversal missions such as logistic management. This thesis is nothing but an illustration of this new philosophy within the Saint-Gobain group, which has decided to create an operations research and management group (based in Saint-Gobain Recherche) to identify potential savings over the supply-chain, model real-life issues as optimization problems and solve them. Our research has been full granted by the Saint-Gobain Glass company, the European leader of flat glass production. We worked on several complementary subjects on which we developed original solutions with successful industrial final applications.

After a brief overview of different decisions that make the supply-chain management a very complex task, we have discovered in chapter 1 a synthesis of the Saint-Gobain Glass business: in few words, producing, adding value and delivering flat glass. Given information from the industrial context that we introduce, we motivate the different parts of our research and link them together through a global purpose that makes sense: we aimed at improving the supply chain management of the business, based on the statement we faced three years ago.

Flat glass is mainly produced for the building and the automotive markets. Saint-Gobain Glass has followed a vertical integration strategy in both of them.

First of all, we deal with the determination of standard-product dimensions in chapter 2, which is a tactical decision updated yearly. On the one hand, the building market is highly standardized, the demand is pretty steady and thus standard products are imposed by the market.

On the other hand, the automotive market is evolving fast every year. Basically, Saint-Gobain Glass supplies trestles of big dimension glass sheets ; customers then cut it into pieces adapted to their own demand. Given that customers are subsidiaries of the group, it makes sense to try to minimize the global loss of glass during different cutting operations along the supply-chain, by adapting standard products to demand forecasts. Thus, the main problem of the automotive market is to update yearly the dimensions of standard products in order to find the best trade-off between global glass loss and inventory management costs of numerous references.

We deal with this interesting problem by introducing an original multi-format structure that makes the cutting optimization problem interesting. We work on several formats and we aim at determining simultaneously a limited number of standard products in each format. Our objective is naturally to minimize the glass loss. We capture the cutting operations of the links both between each standard and its associated final products and between the standards of different formats.

In 2003, we have estimated the gain of this work on several cases by comparing the results of the human and of the tool on different cases with identical data and constraints sets. We have shown on average that around 2.48 % of the global cutting operations' loss can be saved using our optimization tool. In addition, the duration of the determination process by the user has been significantly reduced. It represents savings of approximately 5.000.000 €a year.

The important issue of glass loss minimization being tackled, we evolve gradually in the following of the thesis from chapter 3 to chapter 6 towards the definition of both a framework and an original integrated production-inventory-distribution model which captures the specific supply-chain of glass under deterministic assumptions.

We have developed our research step by step, from basic models to an integrated one, which is now used by practitioners as a both tactical and strategic decision-support tool.

In chapter 3, we start by a study of supply chain design methods used for simple location problems. Given the structure and the costs of a simple supply chain, how is it possible to build models that help managers to determine both the number and the location of facilities?

In this chapter we develop both theoretical and pragmatical methods to study how to determine an optimal supply chain design. We focus on the logistic network because we aim at highlighting distribution issues in the glass industry.

After a review on interesting mathematical models dealing with location-allocation problems under simple assumptions, we analyze industrial past data to understand the underlying structure of demand in order to develop insights on non-optimal logistic phenomena. We follow a pragmatical method aiming at both discovering new concept of distribution and determining efficient distribution rules for a given design.

Our simulations on past data allow us to capture the high potential of such a thought: a very important part of present distribution costs may be saved through an organization change.

Finally, the simple models that this chapter introduce are quickly limited and inadequate to help managers to make a strategic decision. We extend in chapter 6 this research by building a more general and complex model including distribution issues that we highlight in chapter 3.

Chapter 4 then introduces an original production modeling framework that has a great particularity in our research: we apply it at both the operational scheduling level and the tactical planning one. Based on the decomposition of products into characteristics, we have developed and factorized existing models capturing sequence dependent set-up times and costs to be able to tackle practical issues we have faced in the glass industry.

Given a planning decision level, we define a method in which meaningful product characteristics are divided into attributes and sub-attributes, corresponding to big and small time buckets ; depending on the time horizon and the time period we define, each characteristic of the production may be viewed as either an attribute taking one value by time period or a sub-attribute taking several values per time period.

Our model is in a way an original synthesis of different modeling methods we have found in the scientific literature. Based on the literature on the general lot sizing and scheduling problem with sequence-dependent set-up times and costs, we provide a mixed integer program that allows us to capture originally classical hypotheses while being for our industrial application solvable by on-hand commercial softwares (CPLEX, see [ILOa]). Reasonable computation times are obtained by decreasing the number of integer variables of the model. First of all, an original factorization of changeover times and costs is inspired by practical observations of real-life data. Secondly, we simplify the modelling of changeover time in the model. Last but not least, we introduce a relevant product-driven decomposition allowing us to simplify the production planning problem into a much smaller problem by using various attributes with individual adapted time scales.

What is remarkable is that we may use the same optimization model at several levels of a hierarchical planning approach. Depending on the level, we just use various options of the model: the choice of included costs is of course critical. From the hierarchical planning point of view, the more levels we consider, the easier the model at each level. It is thus important to create as many levels as reasonable: this approach is justified only if such a simplification makes sense and does not gives local optimal solutions far from the global one.

We applied it successfully to the operational planning of the float glass manufacturing industry, for which we developed a software, PLANEO, aiming at scheduling on the short-term the campaigns of thickness and width values inside a given colour campaign.

This collaboration led to very encouraging results, not only from an economical point of view (we identified a potential important gain of 16% of the concerned costs) but also for qualitative consequences, such as knowledge management, inter-function collaboration fostering, etc.

Using the adaptability of our production planning model, we apply it in chapter 5 to other jobs of Saint-Gobain Glass, i.e. transformations of float glass (laminating, coating, etc.). All jobs are easily captured by the model, except one.

Coating lines were not that easy to capture. Basically, coating lines are made of metallic cathodes that are used on-line to sputter nanometric metallic coats on flat glass sheets. Before optimizing the production planning of coating lines, managers needed to have a decision-support tool to configure the on-line cathode sequence, so-called the set-up of the line: the notion of set-up was called design in this chapter, in order not to confuse with the traditional notion of set-up in batch production.

Once the line is configured with a given design, we may perform a portfolio of transformations. Changeovers between designs are time-consuming and thus represent opportunity costs. Moreover, for a given design, it exists also changeovers between different transformations. However, the most important thing to reduce the overall production planning cost (and thus to use efficiently a coating line) is to minimize utilization costs by optimizing the metallic cathode use. In this chapter we dealt with this issue: given a product portfolio and a set of on-hand cathodes, we determine the optimal design under deterministic assumptions.

Once different designs are settled, it becomes indeed easy to capture the line using the notions of attribute and sub-attribute.

At this point, we have confirmed the generic aspect of the production planning model we introduced in chapter 4. We will see in chapter 6 how we use it as a building block for modeling the overall production-inventory and distribution processes of the supply-chain.

Of course, the underlying idea of this on-going research is to be able to better understand the interest of a simultaneous production planning of several different production lines: for instance, at the operational level, is there a "dominating" process that must be planned before planning other processes, or is it justified to plan on-line processes? Using an example of the glass industry, does the float line dominate coating lines?

We identified other outlooks that may be highly interesting. The global optimization coupling both the portfolio determination and the coater line design appears to be for instance an excellent outlook for future research. In addition, forecast demands are by nature uncertain. Try to model the robustness of a solution under stochastic inputs is in our opinion another motivating research opportunity. Last but not least, a more strategic potential reflexion lies in the redefinition of the coating process. Since the creation of the industrial process in the late nineties, coating lines have been created as on-line metallic cathodes. It would be useful to take time to imagine concurrent processes. We think that an in-depth study of the impact of the technical choice on the overall flexibility of the line may have an industrial interest.

Finally, chapters 4 and 5 give us a method for modeling all production jobs of Saint-Gobain Glass, allowing us to integrate production tools in our final model.

As a result, chapter 6 provides a global framework for modeling multi-location supply chains, by capturing and integrating all deterministic production-inventory and direct distribution systems. We address the problem of developing a decision tool for both the production planning and the logistic decisions. This tool covers both strategic, tactical and operational decision levels.

Starting from the Glass production process, we have developed in chapter (4) a general methodology to model a continuous process production planning. Based on a product-driven decomposition into attributes and sub-attributes, we provided a useful mixed integer program that capture different levels of hierarchical production planning.

In chapter 6, we pave the path of our ongoing work on solving real-life problems of industrial and logistic issues. We integrate our precedent work as a building block in a general methodology that captures many industrial industrial and logistic patterns. Our framework covers production and transformation facilities as well as inventories and customers, in a deterministic environment. Flows of products within the supply-chain are possible, based on transportation resources whose skills are specified by the user.

By minimizing production, inventory and transportation costs, we provide in a first step a powerful decision tool for both tactical industrial and logistic decisions. At this tactical level, we consider the supply chain design as known and fixed. For tactical industrial decisions, production facilities have to be planned, based on principles developed in chapter (4). We introduce the customer aggregation method developed in chapter 3 to make this step possible on real-life data set.

Furthermore, we extend our program to strategic decisions, such as facility location, etc. We propose a first method assuming that users have a set of potential identified locations and want to optimize both opening, production, inventory and transportation costs. Based on it, we present a more general method based on specifications of the type of desired facility that tries to determine optimal locations from scratch.

All this work is applied successfully to the Saint-Gobain Glass company, at different levels, highlighting the powerful insights that operations research tools may provide to the industry. As ROADEO includes the PLANEO project (introduced in chapter 4), many practical results have been obtained.

The model is currently used to develop reaction procedures in various situations, such as:

- Given demand forecasts and all plant skills, what is the global colour planning that minimizes production, inventory and transportation costs?

- Is is cheaper to develop on-line transformations or to build off-line specific production lines?
- Whether we introduce a new transportation resources in the supply-chain such as train, is it interesting to open non-producing logistic platforms?
- What is the best response on the tactical planning to an unforecasted event such as a critical production problem or a lower than anticipated yield?
- What are the optimal location and the skill portfolio for building a new production facility?
- Is it worth specializing the float plants (in terms of colour skills)?

This on-going research -new applications often requires tight modifications in the model- aims at creating a very evolution-friendly object program whose the underlying linear program may be solved in a reasonable time by on-hand commercial solvers, such as CPLEX. The interest of Saint-Gobain Glass to develop its own optimization tools lies in the fact that commercial softwares do not capture industrial structure and constraints of the particular glass manufacturing business.

As a conclusion, our thesis is a step by step research that we applied to the glass industry through applications covering strategic and tactical as well as operational issues (points 1, 2, 3, 4 and 5 of our simplified classification, see Figure (1.1)). At each step, based on a literature review, we extend up-to-date models to more complex ones suiting industrial problems. In each chapter, we underline potential or real savings we have identified with managers of Saint-Gobain Glass on real data cases. This thesis is the result of a highly motivating collaboration between industrials and academics and we believe that this work present large possible outlooks that we hope will be developed in the future.

Given the context of this applied research, we emphasize as a conclusion that prospects are twofold.

Firstly, there are various theoretical researches that may be applied to the deterministic models we have developed to capture real-life problems. On the one hand, each model we have exposed in this thesis may be studied from a theoretical point of view in order to develop specific methods to solve it more efficiently than we did. We believe that working on special decomposition in their resolution -inspired by special structures of our models which always include original constraints- is highly motivating for further research. On the other hand, the main limit of our research remains the deterministic context we always assume in the thesis. As a first step, it was justified to forget the uncertainty of every parameters we deal with: market demand, production capacities, lead times, etc. We underlined many times that deterministic models may be used on each sheet of a scenario tree capturing a form of uncertainty. However, to keep on working on similar models by introducing explicitly stochastic parameters constitutes a huge prospect. It may highlight new insights that are impossible to be captured by deterministic models.

Secondly, we strongly believe that our work may be easily extended and applied to various other real-life problems. The generic modeling structure proposed and all corresponding models may cover several types of supply-chain which do not contain vehicle routing problems. Finally, the adaptability of our work to all decision levels -which we underline by applying it to strategic, tactical and operational real-life problems- makes it in our opinion fit to many contexts. We hope that further research and on-going works at Saint-Gobain Recherche will confirm our conclusion.

Bibliography

[ABHT95] B. C. Arntzen, G. G. Brown, T. P. Harrison, and L. L. Trafton. Global supply chain management at digital equipment corporation. *Interfaces*, 25:69–93, 1995.

[Ant65] R. N. Anthony. Planning and control systems: A framework for analysis. *Division of Research, Harvard Business School*, 1965.

[Baj88] C. Bajaj. The algebraic degree of geometric optimization problems. *Disc. Comp. Geom*, 3:177, 1988.

[Bak93] K. R. Baker. Requirements planning. volume 4 of *Handbooks in OR and MS: Logistics of Production and Inventory*, chapter 11. North Holland, 1993.

[BBD+87] D. E. Blumenfeld, L. D. Burns, C. F. Daganzo, M. C. Frick, and R. W. Hall. Reducing logistics costs at general motors. *Interfaces*, 17(1):26–47, 1987.

[BBM+89] P. J. Billington, J. Blackburn, J. Maes, R. Millen, and L. N. Van Wassenhove. Multi-product scheduling in capacited mulmti-stage system. Technical Report 8908/A, Erasmus Universiteit Rotterdam, 1989.

[BCTL83] R. L. Francis B. C. Tansel and T. J. Lowe. Location on networks: A survey. *Management Science*, (29):482–497, 1983.

[Ber03] J. W. M. Bertrand. Supply-chain design: Flexibility considerations. volume 11 of *Handbooks in OR and MS*, chapter 4, pages 133–198. Elsevier B. V., 2003.

[BHMT00] J. Brimberg, P. Hansen, N. Mladenovic, and E. D. Taillard. Improvements and comparaison of heuristics for solving the uncapacitated multisource weber problem. *Operations Research*, (48):444–460, 2000.

[BM96a] J. Brimberg and N. Mladenovic. Solving the continuous location-allocation problem with tabu search. *Studies in Locational Analysis*, (8):23–32, 1996.

[BM96b] J. Brimberg and N. Mladenovic. A variable neighbourhood algorithm for solving the continuous location-allocation problem. *Studies in Locational Analysis*, (10):1–12, 1996.

[BMS04] J. Brimberg, N. Mladenovic, and S. Salhi. The multi-source weber problem with constant opening cost. Technical Report G-2004-14, Les cahiers du GERARD, February 2004.

[BO99] G. Barbarosoglu and D. Ozgur. Hierarchical design of an integrated production and 2-echelon distribution system. *European Journal of Operational Research*, 118:464–484, 1999.

[BT93] G. R. Bitran and D. Tirupati. Hierarchical production planning. volume 4 of *Handbooks in OR and MS: Logistics of Production and Inventory*, chapter 10, pages 523–568. North Holland, 1993.

[BW00] G. Belvaux and L. A. Wolsey. Lot-sizing problems: Modelling issues and a specialized branch-and-cut system bc-prod. *Management Science*, 46:724–738, 2000.

[BW01] G. Belvaux and L. A. Wolsey. Modelling practical lot-sizing problems as mixed-integer programs. *Management Science*, 47(7):993–1007, July 2001.

[Car03] T. Caroline. Modèle et mise en oeuvre pour l'aide à la décision à moyen terme. gestion des chaînes logistiques. Technical report, Université II Le Mirail, 2003.

[CC03] H. Chen and C. Chu. A lagrangean relaxation approach for supply chain planning with order/setup costs and capacity constraints. *Journal of Systems Science and Systems Engineering*, 12:98–110, 2003.

[CDS02] J. Current, M. Daskin, and D. Schilling. *Discrete Network Location Models*, chapter 3, pages 81–118. Facility Location Theory : Applications and Methods. Springer-Verlag, Berlin, 2002.

[CF94] P. Chandra and M. L. Fisher. Coordination of production and distribution planning. *European Journal of Operational Research*, 72:503–517, 1994.

[Che04] Z. L. Chen. *Integrated production and distribution operations: Taxonomy, Models, and Review*, chapter 17, pages 711–745. Handbook of Quantitative Supply Chain Analysis: Modeling in the E-business Era. Kluwer Academic Publishers, 2004.

[CL88] M. A. Cohen and H. L. Lee. Strategic analysis of integrated production-distribution systems: models and methods. *Operations Research*, 36(2):216–228, 1988.

[Cla98] A. R. Clark. Batch sequencing and sizing with regular varying demand. *Production Planning and Control*, 9(3):260–266, 1998.

[CM69] E. J. Cockayne and Z. A. Melzak. Euclidean constructability in graph minimization problems. *Math. Mag.*, 42:206, 1969.

[CM01a] S. Chopra and P. Meindl. *Demand Forecasting in a Supply Chain*, chapter 7, pages 171–204. Supply Chain Management: Strategy, Planning, and Operations. Pearson, Prentice hall, Upper Saddle River, New Jersey, second edition, 2001.

[CM01b] S. Chopra and P. Meindl. *Network design in an uncertain environment*, chapter 6, pages 136–168. Supply Chain Management: Strategy, Planning, and Operations. Pearson, Prentice hall, Upper Saddle River, New Jersey, second edition, 2001.

[CM01c] S. Chopra and P. Meindl. *Supply Chain Management: Strategy, Planning, and Operations*. Pearson ; Prentice hall, Upper Saddle River, New Jersey, second edition, 2001.

[Coo63] L. Cooper. Location-allocation problem. *Operations Research*, (11):331–343, 1963.

[Coo64] L. Cooper. Heuristic methods for location-allocation problems. *SIAM Review*, (6):37–53, 1964.

[CSSLS04] L. M. A. Chan, Z. J. M. Shen, D. Simchi-Levi, and J. L. Swann. *Coordinating Pricing and Inventory Decisions: A Survey and Classification*, chapter 9, pages 335–392. Handbook of Quantitative Supply Chain Analysis: Modeling in the E-business Era. Kluwer Academic Publishers, 2004.

[CW97] M. Chen and W. Wang. A linear programming model for integrated steel production and distribution planning. *International Journal of Operations and Production Management*, 17:592–610, 1997.

[Das95] M. Daskin. *Network and Discrete Location*. Wiley, New York, 1995.

[DDGS95] Y. Dumas, J. Desrosiers, E. Gelinas, and M. M. Solomon. An optimal algorithm for the travelling salesman problem with time windows. *Operations Research*, 43(2):367–371, March-April 1995.

[DFF01] C. Dhaenens-Flipo and G. Finke. An integrated model for an industrial production-distribution problem. *IIE Transactions*, 33:705–715, 2001.

[DG99] K. Dogan and M. Goetschalckx. A primal decomposition method for the integrated desing of multi-period production-distribution systems. *IIE Transactions*, 31:1027–1036, 1999.

[DH95] A. Drexl and K. Haase. Proportional lotsizing and scheduling. *International Journal of Production Economics*, 40:73–87, 1995.

[DK97] A. Drexl and A. Kimms. Lot sizing and scheduling - survey and extensions. *European Journal of Operational Research*, 99:221–235, 1997.

[DPL02] S. Dauzère-Pèrés and J. B. Lasserre. On the importance of sequencing decisions in production planning and scheduling. *International transactions in operational research*, 2002.

[Elm78] S. Elmaghraby. The economic lot scheduling problem (elsp): Review and extensions. *Management Science*, 24(6):587–598, 1978.

[EWGC71] S. Eilon, C. D. T. Watson-Gandy, and N. Christofides. *Distribution management: Mathematical modelling and practical analysis*. Hafner, New York, 1971.

[FABC97] P. M. França, V. A. Armentano, R. E. Berretta, and A. R. Clark. A heuristic for lot-sizing in multi-stage systems. *Computers and Operations Research*, 14(9):861–874, 1997.

[Fle90] B. Fleischmann. The discrete lot sizing and scheduling problem. *European Journal of Operational Research*, 44:337–348, 1990.

[Fle94] B. Fleischmann. The discrete lot sizing and scheduling problem with sequence-dependent setup-costs. *European Journal of Operational Research*, 75:395–404, 1994.

[FM97] B. Fleischmann and H. Meyr. The general lotsizing and scheduling problem. *OR Spektrum*, 18(1):11–21, 1997.

[FM03] B. Fleischmann and H. Meyr. Planning hierarchy, modeling and advanced planning systems. volume 11 of *Handbooks in OR and MS: Supply Chain Management: Design, Coordination and operation*, chapter 9, pages 457–523. Elsevier, 2003.

[FMW02] B. Fleischmann, H. Meyr, and M. Wagner. *Advanced planning*, chapter 4, pages 71–95. Supply Chain Management and Advanced Planning. Springer, Berlin, second edition, 2002.

[For58] J. W. Forrester. System dynamics: a major breackthrough for decision makers. *Harvard Business Review*, 36:37–66, 1958.

[FV97] F. Fumero and C. Vercellis. Integrating distribution, machine assignment and lot sizing via lagrangean relaxation. *International Journal of Production Economics*, 49:45–54, 1997.

[FV99] F. Fumero and C. Vercellis. Synchronized development of production, inventory, and distribution schedules. *Transportation Science*, 33:330–340, 1999.

[Gay04] JP. Gayon. *Commande optimale de systèmes de production par anticipation*. PhD thesis, Ecole Centrale des Arts et Manufactures (Ecole Centrale de Paris), 2004.

[GG74] A. M. Geoffrion and G. W. Graves. Multicommodity distribution system design by benders decomposition. *Management science*, (20), 1974.

[GIM+03] M. G. Gnoni, R. Iavagnilio, G. Mossa, G. Mummolo, and A. Di Leva. Production planning of a multi-site manufacturing system by hybrid modelling: A case study from the automotive industry. *International Journal of production economics*, 85:251–262, 2003.

[GNU] GNU. *GNU Linear Programming Kit, GLPK 4.7.* www.gnu.org/software/glpk/glpk.html.

[Gol92] E. M. Goldratt. *The Goal: A Process of Ongoing Improvement.* North River Press, second edition, 1992.

[GOV96] A. C. Garavelli, O. G. Okogbaa, and N. Violante. Global manufacturing systems : A model supported by genetic algorithms to optimise production planning. *Computers Industrial Engineering*, 31(1-2):193–196, 1996.

[GS01] M. D. H. Gamal and S. Salhi. Constructive heuristics for the uncapacitated continuous location-allocation problem. *Journal of the Operational Research Society*, (52):821–839, 2001.

[GS03] M. D. H. Gamal and S. Salhi. A cellular heuristic for the multisource weber problem. *Computers and Operations Research*, (30):1609–1624, 2003.

[GS04] P. M. Griffin and C. R. Scherrer. *Next generation ERP systems: scalable and decentralized paradigms*, chapter 18, pages 747–782. Handbook of Quantitative Supply Chain Analysis: Modeling in the E-business Era. Kluwer Academic Publishers, 2004.

[Gui01] A. Guinet. Multi site planning: A transshipment problem. *International Journal of production economics*, 74:21–32, 2001.

[Haq91] A. N. Haq. An integrated production-inventory-distribution model for manufacturing of urea: a case. *International Journal of production economics*, 39:39–49, 1991.

[HJK96] C. R. Houck, J. A. Joines, and M. G. Kay. Compararison of genetic algorithms, random restart and two-opt switching for solving large location-allocation problems. *Computer and Operations Research*, (23):587–596, 1996.

[HM75] A. C. Hax and H. C. Meal. Hierarchical integration of production planning and scheduling. volume 1 of *Studies of Management sciences*, pages 53–69. Elsevier, New York, 1975. Logistics.

[HMT98] P. Hansen, N. Mladenovic, and E. Taillard. Heuristic solution of the multisource weber problem as a p-median problem. *Operations Research Letters*, (22):55–62, 1998.

[Hsu83] W. Hsu. On the general feasibility test of scheduling scheduling lot sizes for several products on one machine. *Management Science*, 29(3):93–105, 1983.

[ILOa] ILOG, Gentilly, France. *ILOG CPLEX 8.0.* www.ilog.com/products/cplex.

[ILOb] ILOG, Gentilly, France. *ILOG CPLEX 9.0.* www.ilog.com/products/cplex.

[Jan02] R. Jans. *Capacitated Lot-Sizing Problems: new applications, formulations and algorithms.* PhD thesis, Katholieke Universiteit Leuven, September 2002.

[JD98] C. Jordan and A. Drexl. Discrete lotsizing and scheduling by batch sequencing. *Management Science*, 44(5):698–713, 1998.

[JD04] R. Jans and Z. Degraeve. An industrial extension of the discrete lot sizing and scheduling problem. *IIE Transactions*, 36:47–58, 2004.

[JP01] V. Jayaraman and H. Pirkul. Planning and coordination of production and distribution facilities for multiple commodities. *European Journal of Operational Research*, (133):394–408, 2001.

[KD05] A. Klose and A. Drexl. Facility location models for distribution system design. *European Journal of Operational Research*, (162):4–29, 2005.

[KG03] A. G. Kok and S. C. Graves. Introduction. volume 11 of *Handbooks in OR and MS*, chapter 1, pages 1–14. Elsevier B. V., 2003.

[Kra97] S. Krau. *Extensions du problème de Weber*. PhD thesis, Ecole Polytechnique de Montréal, 1997.

[KS85] U. S. Karmakar and L. Schrage. The deterministic dynamic product cycling problem. *Operations Research*, 32(2):326–345, 1985.

[KSW94] R. Kuik, M. Salomon, and L.N. Van Wassenhove. Batching decisions: Structure and models. *European Journal of Operational Research*, 75:243–263, 1994.

[KSWM93] R. Kuik, M. Salomon, L.N. Van Wassenhove, and J. Maes. Linear programming, simulated annealing and tabu search heuristics for lotsizing in bottleneck assembly system. *IIE Transactions*, 25(1):62–72, 1993.

[LB93] H. L. Lee and C. Billington. Hewlett-packard gains control of inventory and service through design for localization. *Interfaces*, 23:1–11, 1993.

[LB95] H. L. Lee and C. Billington. The evolution of supply-chain management models and practice at hewlett-packard. *Interfaces*, 25:42–63, 1995.

[LLKS93] E. G. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys. Sequencing and scheduling: Algorithms and complexity. volume 4 of *Handbooks in OR and MS: Logistics of Production and Inventory*, chapter 9, pages 445–522. North Holland, 1993.

[LN93] H. L. Lee and S. Nahmias. Single-product, single-location models. volume 4 of *Handbooks in OR and MS: Logistics of Production and Inventory*, chapter 1, pages 3–55. North Holland, 1993.

[LPW97] H. L. Lee, V. Padmanabhan, and S. Whang. Information distortion in a supply chain. *Management Science*, 43:546–558, 1997.

[MDE93] C. H. Martin, D. C. Dent, and J. C. Eckhart. Integrated production, distribution, and inventory planning at libbey-owens-ford. *Interfaces*, (23):68–78, May–June 1993.

[Mea84] H. C. Meal. Putting production decisions where they belong. *Harvard Business review*, 2:102–111, March-April 1984.

[Mey99] H. Meyr. *Simultane Losgrössen- und Reihenfolgeplanung für kontinuierliche Produktionslinien*. PhD thesis, Deutscher Universitäts-Verlag, Wiesbaden, 1999.

[Mey00] H. Meyr. Simultaneous lotsizing and scheduling by combining local search with dual reoptimization. *European Journal of Operational Research*, 120:311–326, 2000.

[Mey02] H. Meyr. Simultaneous lotsizing and scheduling on parallel machines. *European Journal of Operational Research*, 139(2):277–292, 2002.

[Mil01] T. C. Miller. *Hierarchical Operations and Supply Chain Planning*. Springer, Berlin, 2001.

[MJH02] C. Moon, K. Jongsoo, and S. Hur. Integrated process planning and scheduling with minimizing total tardiness in multiple plants supply chain. *Computers and Industrial Engineering*, 43:331–349, 2002.

[Moh99] Z. M. Mohamed. An integrated production-distribution model for a multi-national company operating under varying exchange rates. *International Journal of production economics*, 58:81–92, 1999.

[MW88] J. Maes and L. V. Wassenhove. Multi-item single-level capacited dynamic lot-sizing heuristics: A general review. *Operational Research Society*, 39(11):991–1004, 1988.

[MY05] M. Sambasivan Murali and S. Yahya. A lagrangean-based heuristic for multi-plant, multi-item, multi period capacitated lot-sizing problems with inter-plant transfers. *Computers and Operations Research*, 32:537–555, 2005.

[OB00] L. Ozdamar and G. Barbarosoglu. An integrated lagrangean relaxation simulated annealing approach to the multi level multi item capacitated lot sizing problem. *International Journal of production economics*, 68:319–331, 2000.

[OD98] S. H. Owen and M. S. Daskin. Strategic facility location: A review. *European Journal of Operational Research*, (111):423–447, 1998.

[OY99] L. Ozdamar and T. Yazgac. A hierarchical planning approach for a production-distribution system. *International journal of Production Research*, 37:3759–3772, 1999.

[PC90] D. F. Pyke and M. A. Cohen. Effects of flexibility through set-up time reduction and expediting on integrated production-distribution systems. *IIE transactions on Robotics and Automation*, 6(6):609–620, 1990.

[PC93] D. F. Pyke and M. A. Cohen. Performance characteristics of stochastic integrated production-distribution systems. *European Journal of Operational Research*, (68):23–48, 1993.

[PH90] C. K. Prahalad and G. Hamel. The core competence of the corporation. *Harvard Business Review*, 68:79–91, May-June 1990.

[PLYG03] E. Perea-Lopez, B. E. Ydstie, and I. E. Grossmann. A model predictive control strategy for supply chain optimisation. *Computers and Chemical Engineering*, 27:1201–1218, 2003.

[Por90] E. L. Porteus. *Stochastic Inventory Theory*, volume 2 of *Handbooks in OR and MS: Stochastic Models*, pages 605–652. Elsevier Science Publishers, North Holland, Amsterdam, 1990.

[Por98] M. E. Porter. *Competitive Advantage : Creating and Sustaining Superior Performance*. Free Press, June 1998.

[Raj04] K. Rajaram. Campaign planning and scheduling for multiproduct batch operations with applications to the food-processing industry. *Manufacturing and Service Operations Management*, 6(3):253–269, 2004.

[RDPL99] W. Roux, S. Dauzère-Pèrés, and J. B. Lasserre. Planning and scheduling in a multi-site environnment. *Production planning & control*, 10(1):19–28, 1999.

[Rek92] G. V. Reklaitis. Overview of scheduling and planning of batch process operations. In *NATO ASI Batch Process System Engrg.*, Antalya, Turkey, 1992.

[Ros92] K. E. Rosing. An optimal method for solving the (generalized) multi-weber problem. *European Journal of Operational Research*, (58):414–426, 1992.

[Sal91] M. Salomon. Deterministic lotsizing models for production planning. Technical report, Springer-Verlag, Heidelberg, 1991.

[Sam94] M. Sambasivan. *Uncapacitated and capacitated Lot Sizing for Multi-plant, multi-item, multi-period problems with inter-plant transfers.* PhD thesis, 1994.

[SC01] A. T. Staggemeier and A. R. Clark. A survey of lot-sizing and scheduling models. In *23rd Annual Symposium of the Brazilian Operational Research Society (SOBRAPO)*, Campos do Jordão, Brazil, November 2001.

[SCD04] Z. J. M. Shen, C. Coullard, and M. S. Daskin. A joint location-inventory model. *Transportation Science*, 37(1):40–55, february 2004.

[Seg96] A. Segerstedt. A capacity constrained multi level inventory and production control problem. *International Journal of Production Economics*, 45:449–461, 1996.

[Sha93] J. F. Shapiro. Mathematical programming models and methods for production planning and scheduling. volume 4 of *Handbooks in OR and MS: Logistics of Production and Inventory*, chapter 8, pages 371–443. North Holland, 1993.

[SK02] H. Stadtler and C. Kilger, editors. *Supply Chain Management and Advanced Planning.* Springer, Berlin, second edition, 2002.

[SLKSL03] D. Simchi-Levi, P. Kaminsky, and E. Simchi-Levi. *Designing and Managing the Supply Chain.* McGraw-Hill, second edition, 2003.

[SLWS04] D. Simchi-Levi, S. D. Wu, and Z. M. Shen. *Handbook of Quantitative Supply-Chain Analysis: Modeling in the E-business Era.* Kluwer Academic Publishers, 2004.

[SN99] A. M. Sarmiento and R. Nagi. A review of integrated analysis of production-distribution systems. *IIE Transactions*, (31):1061–1074, 1999.

[Sny04] L. V. Snyder. Technical Report 04T-015, Department of Industrial and Systems Engineering, Lehigh University, July 2004.

[SPP98] E. A. Silver, D. F. Pyke, and R. Peterson. *Inventory Management and Production Planning and Scheduling*. John Wiley and Sons, New York, third edition, 1998.

[SS02] M. Sambasivan and C. P. Schmidt. A heuristic procedure for solving multi-plant, multi-item, multi-period capacitated lot sizing problem. *Asia Pasific Journal of Operational Research*, 19:87–105, 2002.

[SSW+97] M. Salomon, M. M. Solomon, L. N. Van Wassenhove, Y. Dumas, and S. Dauzère-Pérès. Solving the discrete lotsizing and scheduling problem with sequence dependent set-up costs and set-up times using the travelling salesman problem with time windows. *European Journal of Operational Research*, 100(3):494–513, 1997.

[TK00] C. Timpe and J. Kallrath. Optimal planning in large multi site production networks. *European Journal of Operational Research*, 126:422–435, 2000.

[TM93] L. J. Thomas and J. O. McClain. An overview of production planning. volume 4 of *Handbooks in OR and MS: Logistics of Production and Inventory*, chapter 7, pages 333–370. North Holland, 1993.

[VG97] C. J. Vidal and M. Goetschalckx. Strategic production-distribution models: A critical review with emphasis on global supply chain models. *European Journal of Operational Research*, 98:1–18, 1997.

[Wei37] E. Weiszfeld. Sur le point pour lequel la somme des distances de n points donnes est minimum. *Tohoku Math.*, 43:355, 1937.

[Wes93] G. O. Wesolowsky. The weber problem: History and perspectives. *Location Science*, (1):5–23, 1993.

[Wol95] L. A. Wolsey. Progress with single-item lot-sizing. *European Journal of Operational Research*, 86:395–401, 1995.

[WW58] H. M. Wagner and T. M. Whitin. Dynamic version of the economic lot size model. *Management Science*, 9(1):89–96, 1958.

[Zip91] P. H. Zipkin. Computing optimal lot sizes in the economic lot scheduling problem. *Operations Research*, 39(1):56–63, 1991.

[ZKM91] M. Zuo, W. Kuo, and K. L. McRoberts. Application of mathematical programming to a large-scale agricultural production and distribution system. *Journal of Operational Research Society*, 42:639–648, 1991.