MANAGING PRODUCT VARIETY
THROUGH DELAYED PRODUCT DIFFERENTIATION
USING VANILLA BOXES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ÖZLEM BURHAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

DECEMBER 2004

Approval of the Graduate School of Natural and Applied Sciences

____

Prof. Dr. Canan Özgen
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

____

Prof. Dr. Çağlar Güven
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

____

Asst. Prof. Dr. Sedef Meral
Supervisor

Examining Committee Members

Prof. Dr. Ömer KIRCA	(METU, IE)	______________________
Asst. Prof. Dr. Sedef MERAL	(METU, IE)	______________________
Assoc. Prof. Dr. Levent KANDİLLER	(METU, IE)	______________________
Asst. Prof. Dr. Haldun SÜRAL	(METU, IE)	______________________
Mehmet KILIÇ	(TÜBİTAK)	______________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Özlem Burhan

Signature :

ABSTRACT

MANAGING PRODUCT VARIETY THROUGH DELAYED PRODUCT DIFFERENTIATION USING VANILLA BOXES

Burhan, Özlem

M.S., Department of Industrial Engineering
Supervisor: Asst. Prof. Dr. Sedef Meral

December 2004, 107 pages

In an attempt to reduce costs and improve customer satisfaction, manufacturers have been adopting strategies such as Delayed Product Differentiation (DPD) while managing broader product lines. In this study, first a general framework on DPD is formed in the light of basic articles in the literature. The vanilla box assembly process which is a special form of modular design type of DPD is modeled and analyzed. In the vanilla box assembly process, inventory is stored in a special form of semi-finished products, called vanilla boxes, that can serve more than one final product. We model the vanilla box assembly process considering the costs of inventory and unsatisfied demand under the capacity limitations, stochastic demand and bill of material requirements. We formulate the model as an extensive form of stochastic integer program in which stochastic demand is modeled using a set of demand scenarios each of which is assigned a probability of occurrence. The model is solved as a standard integer programming model that minimizes the expected value of the objective function. The impact of product demand scenarios, common component levels, shortage penalty cost to holding cost ratio levels and capacity restrictions on the total cost and fill rates is studied. We compare the performance of vanilla box assembly process to assemble-to-order process and provide insights on their performances. Computational results indicate that the vanilla box assembly process is a promising alternative to the assemble-to-order process in most of the problem instances.

Keywords: Delayed Product Differentiation, Vanilla Box Assembly Process, Assemble to Order, Stochastic Programming, Product Variety

ÖZ

VANİLYA KUTUSU KULLANILARAK GECİKTİRİLMİŞ ÜRÜN FARKLILAŞMASI İLE ÜRÜN ÇEŞİTLİLİĞİNİN YÖNETİMİ

Burhan, Özlem

Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Yrd. Doç. Dr. Sedef Meral

Aralık 2004, 107 sayfa

Üreticiler, geniş ürün çeşitliliğini yönetirken, maliyetleri azaltmak ve müşteri memnuniyeti artırmak amacı ile Geciktirilmiş Ürün Farklılaşması (GÜF) gibi stratejileri uygulamaktadırlar. Bu çalışmada, ilk olarak, konu ile ilgili temel makalelerin ışığında GÜF konusunda genel bir çerçeve çizilmiştir. GÜF'ün modüler tasarım tipinin özel bir şekli olan ve envanterin birden fazla son ürün için kullanılabilecek yarı-tamamlanmış ürünler (vanilya kutuları) halinde depolanmasını temel alan vanilya kutusu montaj süreci modellenmiş ve analiz edilmiştir. Vanilya kutusu montaj süreci, kapasite kısıtları, rassal talep ve ürün ağacı gereksinimleri çerçevesinde envanter maliyeti ve karşılanamayan talep miktarı dikkate alınarak modellenmiştir. Model, genişletilmiş rassal tam sayı programı olarak tasarlanmış ve rassal talepler her biri belirli bir gerçekleşme olasılığına sahip talep senaryoları kullanılarak modellenmiştir. Model, amaç işlevinin beklenen değerini enküçükleyen standart bir tam sayı programı olarak çözülmüştür. Ürün talep senaryolarının, ortak bileşen seviyelerinin, karşılanamayan ürünlerin birim maliyeti değerlerinin ve kapasite kısıtlarının, toplam maliyet ve ürün taleplerinin karşılanma yüzdeleri üzerindeki etkisi araştırılmıştır. Vanilya kutusu montaj süreci, siparişe göre montaj süreci ile karşılaştırılmış ve süreçlerin performansları ile ilgili bilgiler elde edilmiştir. Sonuçlar göstermektedir ki, bir çok durumda vanilya kutusu montaj süreci, siparişe göre montaj sürecine iyi bir alternatif olmaktadır.

Anahtar Kelimeler: Geciktirilmiş Ürün Farklılaşması, Vanilya Kutusu Montaj Süreci, Siparişe Göre Montaj, Rassal Programlama, Ürün Çeşitliliği

TABLE OF CONTENTS

PLAGIARISM iii

ABSTRACT..iv

ÖZ vi

TABLE OF CONTENTS..viii

LIST OF TABLES... x

LIST OF FIGURES.. xii

CHAPTER

1. INTRODUCTION.. 1

2. LITERATURE REVIEW.. 4

    2.1. Articles on The Definition and Types of DPD.................................. 6

    2.2. Geographic (or Logistics) Postponement.. 12

        2.2.1 Time Postponement.. 14

        2.2.2. Place Postponement... 15

    2.3. Postponed Manufacturing (Form Postponement)............................ 16

        2.3.1. Standardization.. 17

        2.3.2. Modular Design.. 21

        2.3.3. Process Restructuring... 24

3. PROBLEM ENVIRONMENT AND PROBLEM DEFINITION.......... 27

    3.1. Delayed Product Differentiation Using Vanilla Boxes.................... 27

    3.2. The Problem Environment 33

    3.3. The Problem Definition... 39

4. THE SOLUTION APPROACH.. 42

    4.1. The Stochastic Integer Program 43

    4.2. The Solution Method... 47

5. EXPERIMENTAL ANALYSIS 51
5.1. Experimental Factors 51
5.2. Data Generation 53
5.3. GAMS Model 60
5.4. Analysis 60
5.4.1. Initial Observations 61
5.4.2. Statistical Analysis 75
6. CONCLUSIONS AND FURTHER RESEARCH ISSUES 81
REFERENCES 86

APPENDICES

A. Bill of Material for Final Products and Vanilla Boxes 90
B. Gams Model of The Integer Programming Model of VBA Process 93
C. Results of The GAMS Model for VBA Process and ATO Process 97
D. Formulation and Results of Duncan’s Tests 104

LIST OF TABLES

TABLES

Table 1. Sample product structure 54

Table 2. Generated demand scenarios 56

Table 3. Probability of scenarios for different demand distributions over periods 59

Table 4. Percentage decrease of total cost in VBA process according to ATO process in mixed distribution and high level of component commonality 63

Table 5. Percentage increase of fill rate in VBA process according to ATO process in mixed distribution and high level of component commonality 64

Table 6. Percentage decrease of total cost in VBA process according to ATO process in increasing distribution and high level of component commonality 66

Table 7. Percentage increase of fill rate in VBA process according to ATO process in increasing distribution and high level of component commonality 67

Table 8. Percentage decrease of total cost in VBA process according to ATO process in first-increase-then-decrease distribution and high level of component commonality. 70

Table 9. Percentage increase of fill rate in VBA process according to ATO process in first-increase-then-decrease distribution and high level of component commonality 72

Table 10. Percentage decrease of total cost in VBA process according to ATO process in first-decrease-then-increase distribution and high level of component commonality 73

Table 11. Percentage increase of fill rate in VBA process according to ATO process in first-decrease-then-increase distribution and high level of component commonality 74

Table 12. Probability of scenarios for increasing demand distribution 76

Table 13. Probability of scenarios for first-increase-then-decrease demand distribution 77

Table 14. Probability of scenarios for first-decrease-then-increase demand distribution 77

Table 15. Duncan’s test results for high level of component commonality.......... 79

Table 16. Bill of material for final products.. 90

Table 17. Bill of material for vanilla boxes... 91

Table 18. Results of GAMS model 97

Table 19. Formulation of Duncan’s Tests 104

Table 20. Results of Duncan’s Tests 104

LIST OF FIGURES

FIGURES

Figure 1. Implementation of VBA process in current period T 41

Figure 2. Total cost versus penalty cost for ATO and VBA processes in mixed distribution and 1600 time units of capacity 63

Figure 3. Fill rate versus penalty cost for ATO and VBA processes in mixed distribution and 1200 time units of capacity 65

Figure 4. Total cost versus penalty cost graph for ATO and VBA processes in increasing demand distribution and 2000 time units of capacity.. 66

Figure 5. Fill rate versus penalty cost for ATO and VBA processes in increasing distribution and 1400 time units of capacity................ 68

Figure 6. Total cost versus penalty cost for ATO and VBA processes in decreasing distribution and 2400 time units of capacity............... 69

Figure 7. Fill rate versus penalty cost for ATO and VBA processes in decreasing distribution and 2000 time units of capacity............... 69

Figure 8. Total cost versus penalty cost for ATO and VBA processes in first-increase-then-decrease distribution and 1400 time units of capacity 70

Figure 9. Fill rate versus penalty cost for ATO and VBA processes in first-increase-then-decrease distribution and 1600 time units of capacity 71

Figure 10. Total cost versus penalty cost for ATO and VBA processes in first-decrease-then-increase distribution and 1600 time units of capacity.. 72

Figure 11. Fill rate versus penalty cost for ATO and VBA processes in first-decrease-then-increase distribution and 1600 time units of capacity 74

CHAPTER I

INTRODUCTION

Supply chain is the network of facilities consisting of suppliers, manufacturing centers, warehouses, distribution centers and retailers that perform the functions in the order fulfillment cycle. Materials in the form of raw materials, work-in-process inventory and finished products flow among these facilities in the supply chain. In order to reduce systemwide costs and improve service levels, supply chain management (SCM) approaches are utilized to efficiently integrate various levels of the supply chain so that merchandise is produced and distributed at the right quantities to the right locations at the right time.

Effective design plays several critical roles in the supply chain since certain product designs may increase inventory holding and/or transportation costs. Design for Supply Chain Management (DFSCM) concept is one of the SCM approaches stating that the product line structure, BOM and customization processes of a product should be designed in such a way that the logistics costs and customer service performance are optimized.

Delayed product differentiation (DPD), also referred to as postponement, is an approach that supports DFSCM. DPD means delaying activities in the supply chain until customer orders are received with the intention of customizing products as opposed to performing those activities in anticipation of future orders (Van Hoek, 2001). Thus the point in time when a product assumes its identity, i.e., a particular model in a particular region for a particular market segment, is delayed until customer orders are received.

In the global market, due to the different local requirements in taste, language, environment and government regulations; multiple versions of a single product are required, each meeting the specific requirements of a local geographical region. Even within the same geographical region, there may also be multiple models of the product that relate to different functionalities or capabilities of the product in order to satisfy the needs of different market segments (such as education, personal, business and government). Thus, product variety is required to compete in the world market. However, product variety makes it difficult to forecast demands accurately, and consequently leads to high inventory investment and poor customer service. In an environment where there is a tradeoff between keeping inventory and satisfying customer demands on time due to product variety, delayed product differentiation can be a good strategy to achieve flexibility of the supply chain in meeting uncertain and changing customer demands, and to reduce inventory and improve customer service performance simultaneously. DPD also decreases ‘cost of complexity’ by reducing the variety of components and processes in prior stages of manufacturing.

The concept of delayed product differentiation was first introduced in the marketing literature by Alderson in 1950 as stated by Lee and Tang (1997). Several recent articles have been written to explain and quantify the operational benefits of DPD. However, there are serious differences in the definitions of DPD and modeling approaches in these articles. In general, the studies carried on DPD are independent from each other. Research is not well integrated across disciplines. Yet a thorough review that organizes and summarizes the research literature is lacking. One of the aims of this thesis has been to contribute to the synthesis of these studies and form a general framework on DPD in the light of the basic articles on this subject, and define the main gaps in DPD literature, presenting ideas in order to cover these gaps.

Main types of DPD as defined in the literature are geographic/logistics postponement, time postponement, place postponement, postponed manufacturing (form postponement / manufacturing postponement), standardization, modular design and process restructuring. This classification is made to examine and model the practical cases more easily and specifically. The articles are studied in the framework of this classification.

After the general framework is formed, we have studied the vanilla box assembly (VBA) problem (Swaminathan and Tayur, 1995, 1998a), which can be related to modular design type of DPD. In the vanilla box assembly process, instead of starting the final assembly of the products after a customer order is received, the special form of semi-finished inventory, called vanilla box is used to make different products by the addition of appropriate components. We model the vanilla box assembly process considering the costs of inventory and unsatisfied demand under the capacity limitations, stochastic demand and bill of material requirements. We formulate the model as an extensive form of stochastic integer program in which stochastic demand is represented by a number of demand scenarios with a rolling horizon of three periods. The model is solved as a standard integer programming model that minimizes the expected value of the objective function. The optimal vanilla configuration and their inventory levels are found simultaneously through minimizing the sum of inventory holding cost of vanilla boxes and penalty cost of unsatisfied demand. Furthermore, we study the impact of different demand scenarios, shortage penalty cost and capacity restrictions on the optimal vanilla configuration and their inventory levels. Vanilla box assembly process is also compared to the assemble-to-order system in which the final product is assembled from the subassemblies only upon receipt of a customer order. The two systems are compared in terms of their costs and service levels.

The rest of the study is organized as follows. In the following chapter we review the relevant literature. In Chapter 3, we discuss our problem environment and define the problem. In Chapter 4, we present our model and explain the solution approach. In Chapter 5, we present our computational results and provide insights into the vanilla box assembly process. Finally, in Chapter 6, we summarize the key results of the study and comment on further research issues.

CHAPTER II

LITERATURE REVIEW

Delayed Product Differentiation (DPD) is a recent concept in the Operations Management literature. Hence, most of the articles on DPD deal with the definition of this new concept and try to explain what DPD is, for which purposes it can be used, what costs and benefits come out. Since the concept has arisen out of real life experiences, the aim of the articles mostly has been to share these experiences. In order to present the use of DPD, the models formed are related to specific real cases, but most of these models are not explained in detail. It is hard to generalize these models to any supply chain and it is not clearly defined how the parameters, especially the cost parameters, are found.

The main gaps in the systematic approach of the studies on DPD are insufficient integration of studies and findings, lack of studies in specific areas and inadequate support to (modern) operations management decision making. In order to fill these gaps, DPD should be conceived as a supply chain concept instead of being related to the marketing and distribution channel only. Relations between postponement and other supply chain concepts, such as Mass Customization, Just in Time (JIT) manufacturing, Vendor Managed Inventory (VMI), Efficient Consumer Response (ECR) and the associated Quick Response (QR) distribution techniques should be identified in order to integrate DPD with them.

In the literature, there are only a few studies on the relation of mass customization and DPD. Mass customization is a process by which firms apply information technology and management methods to provide product variety and customization through flexibility and quick response. Thus, it breaks with the dilemma that one has to choose between the two options: low volume-high variety and high volume-low variety. van Hoek et al. (1999) mention that postponement may provide a practical solution for realizing the benefits of mass customization by combining push and pull forces in one operating system. Feitzinger and Lee (1997) define the key to mass customization effectively as postponing the task of differentiating a product for a specific customer until the latest possible point in the supply network. van Hoek (2001) also mentions DPD as one of the central features of mass customization. The ways to achieve mass customization of goods are described as follows:

1. Create products and services that are customizable by customers (involving the design function).

2. Modularize components to customize finished products and services (involving manufacturing, distribution, marketing, and product design).

3. Provide quick response throughout the value chain (involving design, manufacturing, distribution, and marketing).

4. Customize services around standard products or services (involving distribution and marketing).

5. Provide point-of-delivery customization (involving the marketing function) such as adjusting clothes in the store.

6. Offer logistics support to sales and marketing incentive programs (involving the distribution function) such as assembly of promotion displays or shelf management to assure availability.

7. Offer customized logistics service levels (involving the distribution function) such as regionally targeted distribution.

As will be mentioned later, each of these ways refers to a type of DPD.

In short, the concept of DPD raises several issues: amount and level of post application; customization; supply chain structure; operating circumstances in technology (or product design; processes; product and markets) and change management.

As mentioned earlier, there are differences on the DPD definitions and DPD types in the articles published. First, the articles that are related to the definition and types of DPD will be explained and differences will be pointed out. In these articles, *delayed product differention* and *postponement* are used interchangeably. In the summary of each article, the term used by the authors is left as it is.

2.1. ARTICLES ON THE DEFINITION AND TYPES OF DPD

van Hoek (2001) reviews the literature on postponement from various perspectives in a systematic framework. 19 articles published between 1965 and 1998 are characterized and classified according to their research methods and the key elements of postponement concept defined by:

- The type of postponement and level of application in the supply chain,
- The amount of customization,
- The spatial configuration of the chain,
- The role of operating circumstances,
- The role of changing management.

The classification on the type of postponement shows how the perception of postponement has developed over time. Three types of postponement defined are:

- Time postponement (delaying the forward movement of goods until customer orders are received)
- Place postponement (storage of goods at central locations in the channel until customer orders are received)
- Form postponement (delaying product finalization until customer orders are received).

Postponed manufacturing is defined as a combination of these three basic types within one operating system. The articles examined are classified according to these types. When examined chronologically, it is seen that the concept of postponement, which is limited to time and place postponement up to 1980’s is now extended to postponed manufacturing.

Although the survey of van Hoek includes only the studies up to 1998 and the representativeness of the examined articles for the research on postponement is limited, van Hoek (2001) provides general concepts and perception of postponement in an historical perspective, relationships between postponement and supply chain, and other concepts of supply chain; and most importantly it provides crucial feedbacks on the studies in this area.

Bowersox and Closs (1996) divide postponement into two as manufacturing (or form) postponement and geographic (or logistics) postponement. Manufacturing (or form) postponement is defined as manufacturing a standard or base product in sufficient quantities to realize economy of scale, while deferring finalization of features, such as color or accessories until customer orders are received. Mixing paint colors at retail stores according to individual customer request rather than maintaining inventories of premixed color paint; postponing final packaging configuration until customer orders are received; installing accessories at automobile, appliance and motorcycle dealerships at the time of purchase are given as examples of form postponement. The common point of these examples is reducing the number of stock keeping units, while providing product variety and retaining mass manufacturing economies of scale.

Geographic (or logistics) postponement, on the other hand, is defined as building and stocking a full-line inventory at one or a few strategic locations and postponing the forward deployment of inventory until customer orders are received. For example in service of supply parts, critical parts are stored in a central warehouse and when demand occurs, electronically transmitted orders are directly shipped to the service center using fast, reliable transportation.

Lee and Tang (1997) is one of the basic articles on delayed product differentiation. A simple model is provided to capture the costs and benefits associated with delayed product differentiation strategy and to analyze special cases motivated by real examples. Lee and Tang formalize three different product/process redesign approaches: standardization, modular design and process restructuring.

Standardization (using common components and processes) reduces the complexity of the manufacturing system, increases the “flexibility” of use for the WIP inventories and improves the service level of the system (due to risk pooling).

Modular design is defined as decomposing the complete product into sub-modules that can be easily assembled together. Component commonalities among products are also considered as decomposing different products into sub-modules.

Process restructuring refers to re-sequencing process steps in making a product so that common process steps shared by multiple products are performed before the product specific process steps.

Lee and Tang (1997) construct a discrete time model for a manufacturing system that produces two end products, where each end product requires processes performed in N stages and the first k operations are common operations to both products (operation k is defined as the last common operation). Product differentiation is defined by deferring the last common operation (increasing the value of k).

The demands of products are assumed to be normally distributed, correlated among each other but independent across time periods. Buffer inventories are held after each operation according to an “order up to level” policy and service levels for different buffers are assumed to be the same.

Under these assumptions, the total relevant cost function is written as a function of k and includes the total investment cost for designing the product/process, total processing costs, total WIP or in transit inventory costs and total buffer inventory costs. The total relevant cost is then used to analyze special cases of three approaches (standardization, modular design and process restructuring). Each case is explained in the following related sections.

In general, the article provides a general view of DPD. The model defines the important parameters and costs to be considered while deciding on the optimal point of differentiation. Although it is difficult to decide on the values of the parameters and compare the alternatives for complex cases, the general approach provided by the article can be used to improve the explained model or to develop new models. More quantitative parameters and cost measurements can be determined considering the main cost and benefits of DPD defined by the article.

Lee (1993) describes some basic concepts related to product and process design for SCM. These are:

- Design for modularity
- Design for localization (design for customization)
- Part commonality and interchangeable subassemblies (design for flexible manufacturing)
- Design for logistics
- Designing products so that anticipated engineering changes can be made easily
- Product line structuring

All these concepts are the ways of delaying product differentiation.

The article explains the general costs and benefits for each way of DPD and presents four real cases to illustrate some of the concepts, and provides significant issues involved in the implementation of these concepts.

The first case deals with a US disc drive manufacturer in a make to order environment, where demands are highly variable. The second part of the manufacturing process, involving the tests of drives, is relatively long. Product differentiation is proposed to be delayed from the beginning of the tests to the end of the common tests by insertion of a “generic” board, on which all common tests are made. After tests are done, the generic board can be removed and actual board can be inserted.

The second case is related to the DeskJet printers of HP Company. In order to delay the product differentiation, the company began to implement DC localization instead of factory localization, by redesigning its product so that the location specific power supply module is the last component to be added on. Thus the module, which is traditionally packaged in the factory, began to be added easily at the distribution centers.

The third case, which is related to design for commonality, is the same as the specific case for standardization of components mentioned in Lee and Tang (1997). The printer manufacturer is proposed to delay product differentiation by the use of a common head driver board and a common print mechanism interface.

The last case is related to the production of network printers for the European market. Traditionally, the main board for the printer is produced in Europe and shipped to Far East. Then printer is manufactured and integrated there and final product is shipped to Europe for distribution. After product line structuring is implemented to delay product differentiation, the production of the board in Europe and printer in Far East are made simultaneously. Then two products are integrated in Europe and distributed there.

For each case, the article explains the costs and benefits that should be considered before implementation. But these costs and benefits are not quantified and it is not clear how they should be compared. Similar to Lee and Tang (1997), this article provides almost all perspectives of delayed product differentiation.

According to the definitions of these four articles and in the light of other articles studied, the types of DPD (or postponement) can be classified and defined as follows:

- Geographic (or logistics) postponement: Building and stocking a full-line inventory at one or a few strategic locations and postponing the forward deployment of inventory until customer orders are received. This type of postponement is a combination of time and place postponement.
    - Time postponement: delaying the forward movement of goods until customer orders are received.
    - Place postponement: storage of goods at central locations in the channel until customer orders are received.
- Postponed manufacturing (form postponement/manufacturing postponement): Delaying product finalization (manufacturing, assembly or even design activities) until customer orders are received. The postponed manufacturing can also be divided into three as:
    - Standardization: using common components and processes.
    - Modular design: decomposing the complete product into sub-modules that can be easily assembled together.
    - Process restructuring: re-sequencing process steps in making a product so that common process steps shared by multiple products are performed before the product specific process steps.

This classification makes it possible to examine and model the cases more easily and specifically. The articles are summarized based on this main classification in the rest of the section.

2.2. GEOGRAPHIC (OR LOGISTICS) POSTPONEMENT

The supply network (the positioning of inventory and location, number and structure of manufacturing and distribution facilities) should be designed to supply the basic product to the facilities performing the customization in a cost effective manner, and it should have the flexibility and the responsiveness to take individual customer orders and deliver the finished, customized goods quickly. The geographic postponement strategy provides that flexibility in the supply chain.

Ernst and Kamrad (2000) introduce a framework for the characterization of different supply chain structures (rigid, postponed, modularized and flexible) which are defined according to the combined levels of modularization and postponement. Different from the most articles that perceive modularization as a way of postponement, the article defines them as two different approaches used in product and process design. This difference comes from the different and deficient perception of postponement. Postponement is only related to the distribution function in the article. Although it is not mentioned by this name in the article, this perception of postponement is equivalent to time and place postponement defined before.

In the article, modularization is linked to the inbound logistics (providing all materials and goods required for making the products) and postponement is linked to the outbound logistics (the flow of the manufactured products from the factories to the customer). Postponement is defined as “a value added process for a set of end products whereby the common processing requirements among them is maximized”. This definition of postponement is restrictive since it is only related to end products. In fact, postponement is an approach that can be used in every stage of manufacturing and distribution process.

The fundamental principle of both modularization and postponement is stated as combining the advantages of scale and scope, but modularization obtains this from a product design point of view where postponement attains it from a process design perspective.

On the basis of given definitions of inbound and outbound logistics, four supply chain structures (rigid, postponed, modularized and flexible) are defined. For simplicity, logistics flow pattern is divided into three steps as manufacturing, assembling and packaging.

The rigid structure is defined as the classical vertically integrated supply chain where the economies of scale is exploited by production of large runs and maintaining large inventories of finished goods. The flexible structure, on the other extreme, uses many subcontractors to make different components and assembles them in response to a specific demand. Modularized and postponed structures are intermediate structures between these extremes. Modularized structure uses multiple sources for the components but finished product is unique. The postponed structure exploits economies of scale in manufacturing but customizes the finished product to satisfy specific demand.

These four structures are compared on a differential cost basis. The evaluation is based on a specific environment: there is a product (ice-cream) that can either be made in house or be modularized, two different markets (England and France) with no demand correlation between them. The difference between two market requirements is in the packaging since the labels come in different languages. The demand is assumed to be normal. Order up to level inventory policy is used; only finished goods inventory is concerned; holding and backordering costs are assumed the same under the four structures and for the two markets.

Four structures are evaluated based on a total cost function (of production quantity) composed of fixed and variable costs of production and packaging, total holding and backordering costs. They are compared for different cases such as fixed cost of production/packaging is constant; variable cost of production/packaging is constant. In all cases, rigid structure is found worst, but the other three structures cannot be compared with the model since the decision depends on many actors that are not captured in the model. In fact, the article is only concerned with the development of a general framework upon which other future work can be based. Thus, a more detailed analysis of the cost elements and resulting service level implications is given as a direction for future research.

van Hoek et al. (1999) provide an analysis of the experiences of four companies (code named by Alpha Software, Beta Biotech, Gamma Equipment and Europe Delta Telecom) in managing the change process associated with the implementation of postponement strategies.

The article explains some important drivers of change in the (re)structuring of international supply chains and provides a framework for developing postponement strategies and implementing it as a part of supply chain restructuring programs to cope with these drivers of change.

The framework is defined in four steps: Visioning, Logistics Strategic Analyses, Logistics Planning and Managing Change. This framework is used to systematically describe the experiences of four case companies that used postponement strategies to restructure their European supply chains. The reasons of restructuring the supply chain, the way of implementing the postponement strategy, its benefits and difficulties are explained in each case. In all four cases, the strategy implemented is time and place postponement. After the detailed analysis of the experiences of four companies, several managerial implications are drawn and these are important points to consider before and during the implementation of postponement strategies. The main finding is that both strategic and operating characteristics influence the feasibility of postponement and it is not easy to gain the anticipated benefits of such strategies.

2.2.1 TIME POSTPONEMENT

Gavirneni and Tayur (1998) analyze four inventory control models to study the benefits of information flow and delayed product differentiation. The article provides procedures to compute the optimal parameters and compares information flow with time postponement under different values of parameters such as holding costs, capacity and variance.

2.2.2. PLACE POSTPONEMENT

Lee, Billington and Carter (1993) develop an inventory model that the Hewlett-Packard Company (HP)’s DeskJet-Plus Printer Division used to evaluate alternative product and process design for localization. To localize the DeskJet-Plus for different countries, HP packages the appropriate power supply module that has the correct voltage and plugs and the appropriate manual with the printer. Traditionally, the packaging process was performed in the factory (referred to as factory-localization). HP management later decided to implement what they call DC-localization, in order to decrease the effects of uncertainty (in demand, lead times, etc.), to reduce inventory of both end products and raw materials and to improve the service level. In DC-localization, the factory would manufacture and ship a generic DeskJet-Plus printer without the power supply module and manual, to the distribution centers (DCs) where the generic product would then be localized to the different specific options as needed. To implement DC-localization, printer should be redesigned so that the power supply module would be the last component that would be added on by DCs easily. An inventory model is developed for HP to compare DC-localization with factory-localization. Inventory system is modeled as a standard periodic review, order-up-to level system where demand for a product is stationary, normally distributed and independent between periods. For the same service level at DCs, inventory costs are found to be decreased in case of DC-localization. Design and investment costs, transportation costs, pros and cons of decentralizing or regionalizing supply of materials are all considered as well. However, they are not reflected in the model. Thus, it is not clear how all these criteria are evaluated and compared against the gains in inventory costs. The result is that HP has decided to apply DC-localization strategy after considering these criteria. According to Feitzinger and Lee (1997), the manufacturing costs are increased, but total cost of manufacturing, shipping and inventory is decreased by %25 after the implementation of the strategy.

Aviv and Federgruen (2001a) relax the assumptions in the model presented by Lee, Billington and Carter (1993) as:

- demands are dependent,
- capacity is limited,
- seasonality exists and can be forecasted,
- there exists more than one point of differentiation.

Then the model is formulated as a Markov Decision Process. It is found out that relative benefit due to postponement can be estimated better in case seasonality is forecasted; benefits of postponement are reduced if capacity is limited; and cost improvement decreases with (positive) demand correlation.

2.3. POSTPONED MANUFACTURING (FORM POSTPONEMENT)

Garg (1999) describes the Supply Chain Modeling and Analysis Tool (SCMAT) they developed and its application in designing products and processes for SCM at a Large Electronics Manufacturer. SCMAT models decentralized supply chains and it can be used to analyze inventory-service level tradeoffs, sourcing, location and transportation tradeoffs, effects of capacity limitations, impact of lot sizes and designing products/processes for SCM. SCMAT consists of two sub-models: the Inventory Network sub-model is used to model the supply chain assuming that the demand is stationary, uncorrelated and normally distributed and capacity is limited; the Queuing Network sub-model computes production lead time.

SCMAT transforms the nonlinear programming problem to one in which the decision variables are the fill rates for all stock keeping units. The model is used to analyze the inventory implications of three different product and process design scenarios. The model includes almost every criterion that should be considered in inventory modeling side of DPD concept. The advantages of postponement cannot be seen obviously since the design cost is not considered in the model. However, when the properties of the scenarios and the criteria included in the model are inspected in detail, it would not be hard to see the advantages of postponement.

Garg and Tang (1997) develop two models (based on centralized control policy and decentralized control policy) to study products with two points of differentiation, examine the benefits of DPD at each point and drives the necessary conditions (demand variability, correlations, relative magnitudes of lead times) when one point of differentiation is more beneficial than the other. Both models include only the inventory costs in an environment where demands are normally distributed, stationary, independent across periods and correlated within a period, capacity is unlimited and inventory system operates under a periodic review base stock policy. For both control policy (centralized and decentralized), the advantages of two points of differentiation (named as early postponement and late postponement) are compared under different conditions such as variability and correlation of demands and lead time of stages. The study is useful especially in examining the effect of postponement in inventory costs under different lead time criteria. But it is not enough to include only the inventory cost; production and investment costs should also be considered in comparing the two points of differentiation.

2.3.1. STANDARDIZATION

Increased part commonality and interchangeable subassemblies (named as “design for flexible manufacturing”) are closely related to DPD. Part commonality can result in cost savings in part number administration, inventory reduction and supplier management. Using common components in products at the beginning of the design stage decreases design cost (instead of two, only one component is designed), unit cost and manufacturing cost. On the other hand, it reduces the variety of products. Using DPD strategy by standardization of components is a useful approach to overcome this tradeoff.

In Lee and Tang (1997), the special case for standardization of components is motivated by printer manufacturing of a major computer manufacturer. Two types of printers, mono and color printers, whose demands are negatively correlated, have three major manufacturing processes: Printed circuit board assembly (PCA), final assembly and test (FA&T) and final customization. DPD is achieved by standardizing FA&T stage, which requires the standardization of a key component, the print mechanism interface, or by standardizing both PCA and FA&T stages, which requires the standardization of a key component known as the head driver board.

Total relevant cost functions are written in terms of parameters for each of the three cases ($k=0$ (no standardization), $k=1$ (standardization of both PCA and FA&T), $k=2$ (standardization of FA&T)). Then these costs are compared according to qualitative estimations such as “processing cost would not increase significantly when the print mechanism interface is standardized” or “the value of the common head driver board is high”. They do not give quantitative results and comparisons. But they provide the parameters and costs to be considered while determining the optimal point of differentiation. It is also mentioned that it may not pay to delay product differentiation when the standardization of parts is costly.

Lee and Sasser (1995) is related to the implementation of design for supply chain management (DFSCM) principles in the new product development at Hewlett-Packard Company (HP). HP’s new product should support power supply requirements for European and North American market (110V for North America and 220V for Europe). HP management explores the feasibility of incorporating a universal power supply capable of supporting both 110V and 220V standards, instead of regionally dedicated modules. A universal power supply would reduce inventory requirements, but it is not known whether the inventory savings and other benefits would offset the increase in material and manufacturing costs. Thus a model based on the principles of DFSCM is used to evaluate logistics costs and service performance arising from standardization to achieve product universality.

The model assumes that there are two markets each requiring different modules. The demands per period in these markets are stationary and independent across time periods and may be correlated between markets. The lead time of obtaining the module from supplier is assumed to be the same for both markets. Manufacturing center of each market uses a built to stock, periodic review, order-up-to level inventory system for modules. Target service level is defined as the percentage of demand met from stock where unfilled demands are backordered. It is assumed that coefficients of variation for the demands per periods for each module are equal. Under these assumptions, the basic model evaluates two design alternatives in terms of inventory (safety stocks).

The basic model is extended by relaxing stationary demand assumption; and each phase of product life cycle (introductory, mature and end-of-life phases) is modeled explicitly by its unique demand characteristics and cost structures. Another relaxation is for stock-outs; transshipments are allowed between two markets in case of stock-outs. A heuristic is used to deal with these relaxations. Unit inventory cost savings are compared with unit increase in material and manufacturing cost for the universal power supply. Universal power supply is said to be beneficial even under conservative assumptions, and demand variability for the two regions and the reconfiguration cost are found to be the key drivers of the cost-benefit analysis. According to Feitzinger and Lee (1997), HP reduced its total costs of manufacturing, stocking and delivering the finished product to the customer by 5% per year with the implementation of that strategy.

Srinivasan et al. (1998) and Tayur (1994) are not directly related to postponement. They analyze and model the use of common components in the assembly systems. But they are important for postponement literature since the models and solving methods of common component problems and postponement problems are similar to each other. Moreover, the advantages of using common components are applicable to the area of standardization.

Srinivasan et al. (1998) demonstrate the quantitative effects of the degree of commonality, target service level and the degree of variability of demand on inventory costs. Although they model the procurement planning problem, the method can also be used in evaluating the magnitude of the impact of standardization on inventory investments during the early design stages of the products. The problem is modeled as a two stage stochastic programming that is re-run in each time period on a rolling horizon basis.

Tayur (1994) provides a model and an algorithm to compute optimal values for component stocks in an assembly system with multiple products that share (some or all) common components in the presence of random demand. The model minimizes the total holding and backlogging costs in periodic review, finite horizon system where lead times for components are long, product demands are highly variable and correlated both among products and among periods, and assembly capacity is unlimited. The problem is modeled as a multi-period stochastic program. They show that the program decomposes into a set of nested stochastic programs each of which is equivalent to a one period recourse problem.

Desai et al. (2001) look at the postponement concept from a different angle. Contrary to all other articles in this area, they claim that standardization as a postponement strategy may not always be advantageous in marketing, because the customers may not perceive different products designed using standardization as a variety and this may decrease the demand of such products. It is emphasized that the cost advantages of postponement may not overcome the marketing disadvantages. Yet, the cost advantages (such as decrease in inventory costs) are only compared with investment and design costs assuming that marketing would always be in the advantageous side. It is shown that it is not that straightforward to decide whether to implement DPD strategy in a supply chain or not. There are many factors to be included in the analysis, one of which is the effect of the strategy on marketing.

Ma, Wang and Liu (2002) focus on the dynamics between the processing time and the component procurement lead time and its impact on commonality and postponement decisions. A multi-period model with multiple products and stochastic demands is formalized where the production capacity is assumed to be unlimited, demands are independent and periodic review base stock policy is used. The model finds out the stage where to use the common component in order to minimize the costs while satisfying the service level requirements. Since direct optimization is difficult, optimization problem is transformed into one of finding a set of base stock levels corresponding to a set of service level requirements.

2.3.2. MODULAR DESIGN

Product modularity, i.e. the division of a product into independent modules, allows a company to standardize components and to create product variety from a fixed set of modules. A product should be designed so that it consists of independent modules that can be assembled into different forms of the product easily in an inexpensive way. Modular design separates the composition of end products into parts or subassemblies, some of which are common to all product options and others are not. Then the assembly of the differentiating module(s) can be postponed until a later stage in the process.

Feitzinger and Lee (1997) define the benefits of modular design as follows:

- A company can maximize the number of common components, assemble them in the earlier stages and postpone the addition of non-common components.

- A company can make the modules of the product separately; different modules can be manufactured at the same time, shortening the total time required for manufacturing.
- A company can diagnose production related problems and isolate potential quality problems more easily.

Swaminathan and Tayur (1998a) present a delayed product differentiation strategy by exploiting component commonality in managing broader product line of IBM. Since the demands are stochastic, it is found to be beneficial to store inventory in the form of semi-finished products (named as vanilla boxes in the article) that can serve more than one final product. But it is hard to find the optimal configurations and inventory levels of the vanilla boxes. They model that problem as a two-stage integer program with recourse and utilize structural decomposition of the problem and (sub) gradient derivative methods to provide an effective solution procedure. The effect of demand variance, correlation and capacity limitations on the optimal configuration and inventory levels of vanilla boxes and the performance of a vanilla assembly process are provided as well. The performance of vanilla assembly process is compared to make-to-stock and assembly-to-order processes. The article discusses the characteristics of an IBM product line and the effectiveness of a heuristic tailored for that application.

In Lee and Tang (1997), special case for modular design is related to manufacturing dishwashers in different colors. Delayed product differentiation is proposed to be achieved by modular design of metal frames (division of metal frame into two modules: a generic metal frame and a plastic panel that specifies the color). The model is applied to a manufacturing system with three major steps and two types of dishwashers (black or white). Similar to the standardization example, the relevant costs are compared verbally, and hence no quantitative results are given. It is concluded from the comparisons that delayed product differentiation is beneficial when the lead time of the last common operation is long, or when the additional module is simple to handle, or when the modular design of parts is relatively inexpensive.

He, Kusiak and Tseng (1998) suggests three design rules for implementing the DPD strategy in manufacturing. The design rules presented are heuristic and developed around the concept of products with linear assembly structure and assembly line balancing, but they are not based on a well-built theory. They claim that in the design phase of a product, the manufacturing times of the subassemblies should be considered and the number of different part types should be reduced (referred to as minimum part count rule). Based on these rules, an integer program is formalized to select among different design alternatives in order to minimize the number of parts and manufacturing cycle time. The model does not include the effect of DPD on manufacturing, design and inventory costs.

Gupta and Krishnan (1998) develop concepts and algorithms for product family based subassembly design to take advantage of the commonalities among products. The algorithm finds the *generic subassembly* (subassembly shared by multiple products with common components and assembly connections) that maximizes the coverage among the entire product family. The model shows that safety stock is minimized by maximizing coverage and it is mentioned that having such a generic subassembly prior to the demand realization would enhance the ability to economically attain product variety.

Aviv and Federgruen (2001b) develop an analytical model to explain and quantify the inventory savings that result from DPD and quick response program. Contrary to most models developed assuming that demands in each period are random and independent across time and demand distributions are perfectly known, the model in the article characterizes these in more general settings, where parameters of the demand distributions fail to be known with accuracy and consecutive demands are correlated. The model also assumes that estimates of the parameters of the demand distributions are revised on the basis of observed sales data. The structure of close-to-optimal ordering rules is also characterized for a variety of types of order cost functions.

2.3.3. PROCESS RESTRUCTURING

Manufacturing processes should be designed so that they consist of independent modules that can be moved or rearranged easily to support different distribution network designs.

In multi-stage manufacturing environments with high product variety, the demands of the end products, and thus the production volumes of the intermediate stages are variable from period to period. The performance of such manufacturing processes can be improved by reducing variability through operations reversal (reversing two consecutive stages of the process).

Benetton case, analyzed in Lee and Tang (1998), is an example of delayed product differentiation through operations reversal. Benetton used to manufacture its products by first dyeing yarns into different colors and then knitting the colored yarns into different finished products (different styles and sizes). Since the color options are more than the style options and the variability of the color options among the periods are higher, they decided to reverse the “dyeing” and “knitting” stages. This change improved the operational performance of the company resulting in inventory reduction, better customer service, increasing sales and fewer end-of-season markdowns. Motivated by the Benetton application, authors formalized a model for a two stage manufacturing process that characterizes the impact of operations reversal (under what circumstances would it lead to variance reduction? what are the key drivers, etc.)

Brown, Lee and Petrakian (2000) share their experience on the implementation of process re-sequencing strategy applied in a semiconductor firm. The firm manufactures application-specific integrated circuits (ASICs) and decides to implement DPD strategy for its programmable devices in order to create a near-infinite number of design from a few thousand physical product permutations. Rather than going through all steps to create an integrated circuit in its finished form, they divide the process in two stages. In the first step, they produce unfinished generic products and hold inventory of them. Since the test of the product has the longest lead time in the manufacturing process, they have designed the generic product so that almost all tests can be applied on it before it is stocked. Then based on actual orders from customers, the generic products are pulled from inventory and customized into finished ASICs. The performance of the implementation is presented in terms of inventory investment vs. backorder cost data and the results are influential. The firm has achieved to reduce its inventory cost without changing the service level.

Lee and Tang (1997) provide two examples for process restructuring: one is achieved by “postponing” an operation downstream, second is achieved by “reversing” the order of two operations. First example (postponement of operation) is drawn from manufacturing of electronic devices. Second example is motivated by sweater manufacturing of Benetton. In both examples, the number of stages and types of products are taken as 3 and 2, respectively, for illustrative purposes. Total relevant costs of three alternatives (k=0, k=1, k=2) are again written in terms of parameters and compared verbally. The main results are as follows: DPD can result in bigger savings from inventory cost,

- if a high value added activity is delayed.
- if a short operation at an early stage is reversed with a long operation at a later stage.
- if a high value added operation at an early stage is reversed with a low value added operation at a later stage.

Swaminathan and Tayur (1999) model and analyze the assembly task design problem (finding the best sequence in which components should be assembled for a product family) within the context of delayed differentiation using vanilla boxes analyzed in Swaminathan and Tayur (1998a). Manufacturers like IBM, US Filter (a manufacturer of industrial pumps) and American Standard (a manufacturer of air conditioning systems, bath-tubs and anti-lock braking systems) integrate the design and manufacture of components within the context of delayed differentiation using vanilla boxes. But the success of vanilla box assembly process depends on the time to assemble products and the vanilla boxes that can be built, which depends on the assembly sequence.

The article builds a detailed model in order to provide insights to managers about the trade offs between different aspects of the problem through computational studies, instead of deriving analytical expressions and optimal solutions by capturing particular trade-offs only. The model determines both the optimal assembly sequence and operating parameters (such as inventory levels) simultaneously using the model of Swaminathan and Tayur (1998a) as a sub-model. The article analyzes the model used in Lee and Tang (1998) and shows that if the analysis of this model uses standard deviation rather than variance, some non-intuitive predictions of the analysis would be eliminated since the inventory costs can be described more appropriately using standard deviation instead of variance.

Lee (1996) presents two inventory models (one for build-to-order environment and one for build-to-stock environment) that can be used to support product/process redesigns for companies to gain control of inventory and service. The models are motivated by real application cases which are explained in the article. Both models assume stationary demands and costs, and the expected response time to customer orders is used as the specific service measure. The models aim to estimate only the inventory savings for finished goods, but Lee (1996) mentions that to fully evaluate the effectiveness of product/process redesign, one would have to assess the impact of inventory savings of the parts, material costs of parts and investment cost of the engineering change in addition to the impact of inventory savings for finished goods.

CHAPTER III

PROBLEM ENVIRONMENT AND PROBLEM DEFINITION

Product variety has increased due to competition in the industry. There is an increase in service expectations including time, quality and cost as well. In such an environment, a make-to-stock policy is not preferred due to high inventory carrying costs and risk of obsolescence. On the other hand, if make-to-order policy is used, it would be a problem to satisfy the customer demands in time especially when there is high demand. Storing inventory in the form of semi-finished products that can serve more than one final product can be a good strategy to overcome the tradeoff between carrying inventory and service level requirements. We focus on this strategy considering the parameters such as cost of inventory, capacity restrictions, service level requirements, demand information, bill of material (BOM) and assembly time/flow time/delivery time.

The problem dealt with in this study can be included in modular design type of DPD. Swaminathan and Tayur (1998a) and Swaminathan and Tayur (1995) are taken as a basis for the study. Our aim is to evaluate and model the vanilla box assembly (VBA) problem in a more realistic environment. First, it would be useful to explain Swaminathan and Tayur (1998a) and Swaminathan and Tayur (1995) in detail in Section 3.1.

3.1. DELAYED PRODUCT DIFFERENTIATION USING VANILLA BOXES

Swaminathan and Tayur (1998a) and Swaminathan and Tayur (1995) present the study carried out on the product line of IBM, where different models show a high degree of component commonality and have highly stochastic and correlated demands. Flexibility in the assembly process is tried to be achieved though DPD by keeping semi-finished inventory of common components. Instead of the traditional mode of operation (that is starting final assembly after a customer order was received), management piloted an assembly process based on semi-finished products called vanilla boxes, which are produced before the realization of demand and used in the assembly of end product after demand is realized. The special form of semi-finished inventory is called vanilla box (white in color) because they could be used to make different products (different colors) by the addition of appropriate components.

The example product structure used in the article includes three final products (P1, P2, and P3) and four components (a-memory card, b-processor, c-hard disc, d-floppy drive). Vanilla boxes (VB); VB1 (components a and b), VB2 (components b and d) and VB3 (components b and c) are given as examples of feasible vanilla boxes for the given product structure. It should be noted here that what is referred to as component is in fact a sub-assembly but not a raw component. Thus a vanilla box represents the assembled form of these sub-assemblies (named as components).

The vanilla box assembly process was piloted at a final assembly plant which had a nearby satellite plant that could produce, test and then ship vanilla boxes to the final assembly plant. Since vanilla boxes are built in a separate plant, the main issues at the final assembly plant were to determine how many and what type of vanilla boxes to keep and how to allocate vanilla boxes to final products in order to minimize the expected stock-out costs for lost product demand and holding cost for left-over vanilla boxes.

To model these issues, a discrete time model with finite assembly capacity is developed and a two-stage stochastic framework is utilized. The first stage corresponds to choosing the configuration of vanilla boxes and their inventory levels, whilst the second stage corresponds to how the vanilla boxes are allocated to the products within a limited assembly capacity, after the demand is realized.

The demand process is modeled using a set of demand scenarios each being assigned a probability of occurrence.

The first model addresses issues related to product line characteristics (such as commonality), demand characteristics (such as variance and correlation of product demands) and assembly time characteristics (such as reduction in assembly time when semi-finished products are used for customization) in an integrated manner.

The basic model considers a single period, where all vanilla boxes are produced in a separate plant before the beginning of the period. Demands are realized at the beginning of the period, so the decision of assembling a product either from its components directly or from any vanilla box is made after the demand realization. It is assumed that the bill of material in terms of the components is binary. Demands are random but follow one of the given scenarios, each with a specified likelihood.

The model simultaneously determines the optimal structure of the vanilla boxes and optimal inventory levels associated with them. The sequence of events is as follows:

1. The number of different types of vanilla boxes (K) is chosen.
2. The vanilla configuration is chosen.
3. The inventory levels for the vanilla boxes are determined.
4. On the realization of the demands, the allocation of assembly capacity and vanilla boxes to the different products is determined by solving a linear program.

The first step of the proposed approach is the enumeration of all possible vanilla box configurations using at most K different types. The number of vanilla box configurations to be considered is Comb [2^n-n-1, K], where n is the number of components (since vanilla boxes with zero or one component are excluded). Thus the algorithm is exponential and not suitable for large problems. To reduce the number of vanilla configurations that need to be considered, they define "maximal" vanilla box. A vanilla box is maximal if the addition of any component to it reduces the number of products that can be assembled using that vanilla box. It is mentioned in the paper that if the holding cost of all vanilla boxes are identical, then the optimal configuration includes maximal vanilla boxes only. For the IBM case, a greedy heuristic is developed to find the vanilla configuration (not necessarily optimal). Heuristic selects the vanilla box that provides the maximum cost reduction at the n^{th} stage when used with the n-1 previously selected vanilla boxes.

Once the vanilla box configuration is chosen, the optimal inventory levels of vanilla boxes and the optimal assembly plan for each scenario is determined by the two-stage stochastic program. The authors utilize the subgradient based approach to solve the problem. The subgradient is obtained from the average value of the dual variables corresponding to vanilla box inventory in the recourse step.

The basic model is extended to a multi-period problem and to the settings where the assembly capacity is used to produce vanilla boxes as well as final products. There are two additional assumptions for the extended model:

- Demands in consecutive periods are independent.
- A base-stock policy is used for managing the inventory of vanilla boxes (each period starts with the same number of vanilla boxes for each type). Overtime costs are assigned for bringing the inventory level to the target level if it is lower at the end of a period.

Under these assumptions, each period's planning problem is independent of other periods.

Some other extensions of the basic model are also modeled or described in the article. The first model is for a special case where each product can be made either from a unique vanilla box or from raw components. The problem is a continuous knapsack problem and can be solved using a greedy rule. The second extension is a variant of the basic model where capacity and speed-up restrictions are removed and costs are assigned for choosing a particular vanilla box for a product. The third extension is providing the customer with redundant components instead of loosing the demands. It is modeled by introducing a cost parameter reflecting the cost of additional components that are present in the vanilla box but not required in product. The fourth extension is related to the substitution among products. Substitution is modeled by introducing a cost for interchanging products. The fifth extension is the incorporation of the raw component inventory in the model. The last extension is the substitution among components. It is modeled by assigning a cost for using a "higher grade" vanilla box for a product.

The impact of different factors; such as capacity, correlation in demand, variance of product demand and number of vanilla boxes, on the total cost and the type of vanilla boxes that are optimal are studied and the results are discussed for the single-period model. (It is assumed that it takes 1 unit of time to assemble a component.) The findings can be summarized as follows:

- An increase in demand variance increases the cost incurred.
- The vanilla process incurs lower cost under negative as compared to positive correlation in product demands.
- Under very tight capacity restrictions, vanilla boxes cannot be utilized because there is not enough capacity to assemble products even from vanilla boxes, thus the optimum inventory levels of vanilla boxes are low. Under a larger capacity level, vanilla boxes can be utilized and the optimum inventory levels increase. Under high capacity, products can be assembled from components and vanilla boxes are not required, thus the optimum inventory levels decrease.
- As the number of vanilla box types increases, the total cost decreases.

The vanilla process is also compared to make-to-stock (MTS) and assembly-to-order (ATO) processes. MTS and ATO are considered as the extreme cases where the configuration of vanilla boxes corresponds to the set of final products and with K=0, respectively. K=0 means that all products are produced from their components (subassemblies) after the demand is realized. The results indicate that vanilla box assembly (VBA) process could perform better than MTS under the condition of negative correlation and medium-to-high capacity. MTS performs better than VBA process in all cases with positive demand correlation. The performance of both VBA process and MTS is better than or equal to ATO in all considered cases. ATO is always the worst because in this case each product should use the capacity to be produced since no vanilla boxes, assembled before, is used.

In general, the approach and the model reflect a good strategy for DPD. However, there are some shortcomings in the defined environment and some assumptions do not reflect the real production environment. For example, it is assumed that vanilla boxes are produced in a separate plant and they do not use the capacity of the final assembly plant, thus the capacity used to produce vanilla boxes is ignored. Because of that assumption, the VBA process mostly gives the best results when compared to MTS and ATO processes where all production activities are done in the final assembly plant. The holding costs of all vanilla boxes are assumed to be the same independent of the number of components they include. Thus, it is concluded that holding cost does not seem to play a very significant role in the configuration of the vanilla boxes, but changes the optimal vanilla box inventory levels (Swaminathan and Tayur, 1995). However, relaxing that assumption may lead to the selection of different vanilla configurations.

In the multi-period extension of the model, the assumption that each period starts with the same number of vanilla boxes for each type reduces the model to a single period model. However, the multi-period problem would be more realistic, if it considers the amount of vanilla boxes that remain from the previous periods and determines the inventory levels of vanilla boxes accordingly.

3.2. THE PROBLEM ENVIRONMENT

Integrating the design and manufacture of components within the context of delayed product differentiation using vanilla boxes is an appropriate strategy when there is a large number of common components among the products. Vanilla boxes can be useful in providing a quick response to customer demands if the product variety is high in a product line.

In this study, we aim to model the delayed product differentiation strategy using vanilla boxes in a more realistic environment in a multi period system. We intend to use a hybrid system of VBA and ATO processes. Because according to the realized demand in a period, it can be more beneficial to assemble some of the products from its components where vanilla boxes are used for the assembly of other products.

We focus our attention on the computer industry, where the market prices have come down drastically and the product variety has been increasing by a very large factor. Product proliferation is enlarged due to the combination of features each with several options. Such a variety in product portfolio results in uncertainty in customer demands and makes demand for the individual products highly stochastic.

In VBA process, vanilla boxes are assembled, tested and stocked before the receipt of a customer order. After the customer order is received, additional components are added to the appropriate vanilla box to make the final product and the product is then tested before being sent to the customer. Therefore the lead time experienced by the customer is reduced, while maintaining the product variety at the expense of holding inventory for the vanilla boxes.

Since the total lead time of assembly is not long, in case when the final assembly capacity is very large, the VBA process reduces to the ATO process, because demand can easily be satisfied by assembling each product from its components rather than using vanilla boxes.

A discrete time, multi-period model with finite capacity for final assembly is studied where the manufacturer experiences stochastic demand for multiple products. VBA process is modeled and analyzed considering costs of inventory, costs of unsatisfied demand, capacity restrictions, demand information, bill of material and resulting service levels under three periods rolling horizon method. The optimal vanilla configuration and their inventory levels are obtained simultaneously through the minimization of holding cost for vanilla boxes and penalty cost for unsatisfied demand. Furthermore, the impact of different product demand scenarios, capacity restrictions and penalty cost on the optimal vanilla configuration and corresponding inventory levels are studied.

Our problem environment is explained below in detail on specific sub-titles together with the assumptions related to them.

Product Variety

In the study, rather than considering all products manufactured in a manufacturing plant, a single product line consisting of different models of the same product is focused on. Different models across the product line show a high degree of component commonality. In addition, there is a one-to-one relationship between components and features. For example, *central processing unit (CPU)* as a feature in a personal computer product line can have two options as *Celeron* and *Pentium 4*, each of which is referred to as a component. As a result, a product is defined in terms of its features and their options that are required by the customer. In a product line, there are several features, some of which are essential, while others are optional. For instance, motherboard, CPU and memory card are essential features where sound card, TV card and fax/modem are optional. Each feature has several options, determining the components that make up the product.

The manufacturer has an unrestricted product portfolio. However, the features and their options are predetermined by the manufacturer. The customer chooses the option she prefers among the presented options of features and configures her own product. Thus, the products in the product line are determined according to these customers' specific orders. Product variety in the product line increases exponentially by the number of options presented for each feature.

Vanilla Boxes

Every product may be assembled either directly from its components or from any vanilla box whose component set is a subset of those required by the product, thus avoiding redundant components. Vanilla boxes can include any combination of components as long as they serve to the assembly of at least one product.

It is implicitly assumed that a vanilla box that contains a component that is not required by the product cannot be used to assemble the product by either stripping this component or giving it free.

All vanilla boxes to be used in the period are produced before the beginning of the period. Remaining vanilla boxes at the end of the period are held in inventory for the next periods' demands.

Demands

Demands are realized at the beginning of the period, before decisions need to be made regarding the assembly of the final products. The variety in product portfolio results in uncertainty in customer demands, the (often negative) correlation of demands between products makes it difficult to have an accurate estimate of the demand for the individual products and makes demand highly stochastic. Thus, demands for the final products are assumed to be random and the demand process is modeled using a set of demand scenarios each of which is assigned a probability of occurrence. They follow one of the given scenarios, each with a specified likelihood. It is assumed that demands are correlated among products and periods.

Type of Components

As stated in Garg (1999), Stadzisz and Henrioud classify components into three different categories: (i) invariant components that do not change their identity across the product line; (ii) pseudo variant components that are different in products but do not change anything in the assembly sequence since they are all placed at the same position in the assembly sequence; and (iii) variant components that are different in products and also could occur at different positions in the assembly sequence. A component or feature that has options associated with it is either pseudo variant or variant. The advantage of having pseudo variant components is that the addition of options does not change the assembly sequence for the product line.

We assumed that all of our components and subassemblies (vanilla boxes) are pseudo variant components.

Assembly Type and Assembly Times

On the basis of the above assumption, assembly time of each component is assumed to be one unit time and there is no precedence relationship in the assembly process among the components. Thus, assembly time of any component and vanilla box is sequence independent.

Production lead time for vanilla boxes is one period. Vanilla boxes that are used for the product demands in period T are to be produced in period T-1. A vanilla box produced in period T cannot be used in the assembly of period T demand.

Contrary to Swaminathan and Tayur (1998 a) where product may either be assembled directly from its components or from a feasible vanilla box (but not both), we assume that in the same period, a product can be produced either from any feasible vanilla boxes or from its raw components, or both. More than one type of vanilla boxes can also be used in satisfying different demands of a product in the same period, but in the assembly of each product, only one vanilla box is allowed to be used.

Bill of Material

Without loss of generality, it is assumed that the bill of material in terms of the components is binary for both products and thus for vanilla boxes.

Periods

Considering high turnover rates of inventory, highly stochastic demand and short lead times in the industry, periods are assumed to be weeks.

Raw Components

Raw components are assumed to be provided on a monthly basis. Thus, all raw components for vanilla boxes and products are available and there is no shortage of raw components during our three periods planning horizon.

Capacity

It is assumed that vanilla boxes are assembled using the same capacity with the end products in a single plant but in separate assembly lines. Thus, the finite assembly capacity is allocated between the two assembly lines: (i) vanilla box assembly capacity, C_{VB}^{t}, (which is only used for the assembly of vanilla boxes produced prior to the customer orders), (ii) final assembly capacity, C_{FG}^{t}, (which is used to assemble additional components to the vanilla boxes or assemble products from raw components only, after the realization of demand and within the customer response time window which is a period). C_{VB}^{t} and C_{FG}^{t} are decision variables in the model, but the sum of them, i.e. the total assembly capacity, is assumed to be fixed.

Costs

We minimize the expected stock-out costs for lost product demand and holding cost for left-over vanilla boxes in our model but the total of these costs over three periods is minimized. The unsatisfied demand in the same period is lost, cannot be backlogged and penalty cost for the lost product demand is assumed to be the same for all products in the product line. Holding costs are incurred for the unused vanilla boxes at a box type specific rate. It is assumed that holding cost of each vanilla box is proportional to its assembly time. This is because the assembly time is equal to the number of components included in the vanilla box, since assembly of each component is assumed to be one unit time. Furthermore, inventory holding cost is only charged for the value added during the assembly process, not for the raw components.

In Swaminathan and Tayur (1998a), the holding costs of the vanilla boxes are assumed to be independent of the components making up them. The reason of this assumption is explained as: “it may not be a good idea to include a very expensive component in the vanilla box if the speed-up provided is not adequate. In such cases we keep track of these expensive components and enumerate all the different possibilities in our logical choice”. However, when the example of vanilla boxes given by the article is inspected, it is seen that the components used are in fact subassemblies most of which are quite expensive, such as processor, memory card, etc. Since we deal with the computer industry and the final assembly process therein, it would be essential to consider the holding costs of components in order to find the holding costs of vanilla boxes. Assuming that all vanilla boxes have the same holding cost would directly affect the vanilla configuration chosen and may lead to ineffective solutions.

It is assumed that all raw components are available at the beginning of three periods planning horizon, thus component inventory costs are not included in the model since they do not have a direct effect on vanilla box assembly process.

3.3. THE PROBLEM DEFINITION

Based on the problem environment and assumptions defined above, our objective is to determine the final assembly plan for the end products with the optimal vanilla box configuration and optimal inventory levels associated with them in order to minimize the expected total cost of inventory holding and stock-outs incurred across all possible demand scenarios. The cost measure consists of holding cost for the leftover vanilla boxes and penalty cost for unsatisfied (lost) product demand.

The problem is considered to be a multi-period problem with a planning horizon of three periods. The solution of the problem is planned to be implemented on a rolling horizon basis. The implementation of this approach in the current period T can be described as follows (shown in Figure 1):

- Vanilla boxes produced in previous period(s) are available at the beginning of period T.
- As the demand is realized at the beginning of period T, the vanilla boxes on hand are allocated to the products that would be produced for demands of periods T, T+1 and T+2. In the allocation, the realized demands are considered for period T, while the demand scenarios are considered for periods T+1 and T+2. In final product assembly line, final assembly capacities are utilized in each period. The result gives the final product assembly schedule for the three periods, i.e. how many of each product would be produced, how they would be produced (how many of them from raw components, how many of them using vanilla box (es)). After this allocation, holding costs for vanilla boxes and penalty costs for unsatisfied demand would have been known.
- Given the types and amounts of remaining vanilla boxes at the end of period T from the final product assembly, in the vanilla box assembly the best vanilla configuration for period T+1 is found using demand

scenarios of periods T+1, T+2 and T+3; and vanilla box assembly is scheduled for period T simultaneously. In vanilla box assembly line, vanilla box assembly capacities of each period are utilized. The output is the next period's (T+1) initial vanilla box inventory, which turns out to be an input to the final product assembly of period T+1.

The results of vanilla box assembly and final product assembly are implemented only in period T. The above sequence of events recurs at the beginning of period T+1, as a result of one-period rolling ahead.

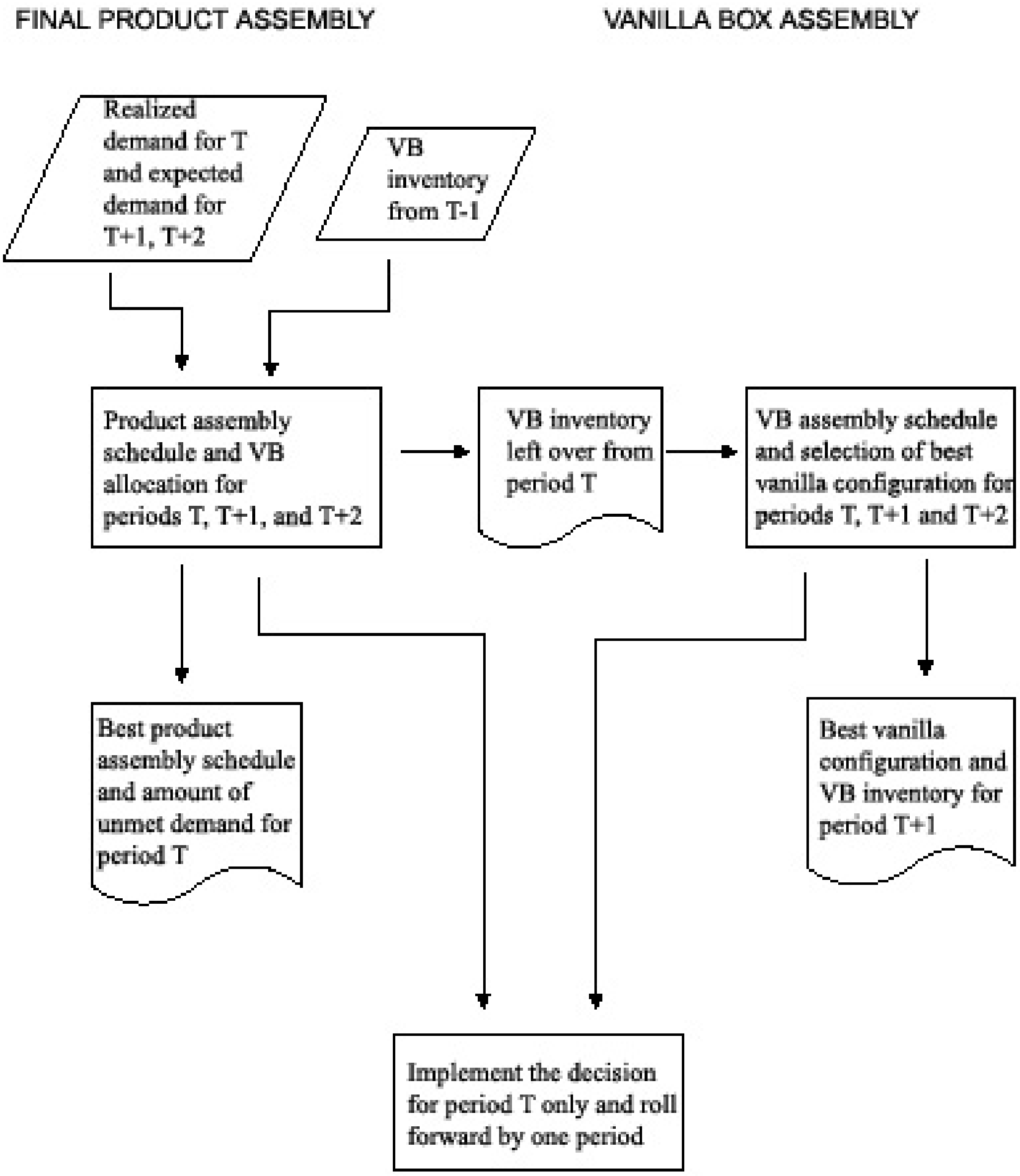


Figure 1. Implementation of VBA process in current period T

CHAPTER IV

THE SOLUTION APPROACH

The uncertainty in customer needs for combination of features (which determine the components in the product) makes demand for the products highly stochastic. Furthermore, the (often negative) correlation of demands between products makes it extremely difficult to have an accurate estimate of demand for the individual products. Hence, the stochasticity of demand is taken into account and the problem is modeled as an extensive form of stochastic integer program. A scenario based approach is utilized for the involvement of the stochastic demand into the problem. A scenario is defined as a combination of the realizations of the random variables, i.e. demand for all periods in the planning horizon, excluding the current period for which demand is known.

Scenario planning is very important in designing the model and analyzing the results of the stochastic program. Shapiro (2001) presents a ten-step methodology of scenario planning that centers on learning and exploring interrelationships among strategic trends and key uncertainties. This methodology requires the strategic analysis of the products, suppliers, markets; identification of major stakeholders, current trends and key uncertainties. Since our study is not based on a real life problem, it is hard to use a methodology like that. Thus, a simpler scenario planning approach is used in the study.

It is thought that it may not always be possible to estimate demand of each product in an environment where the product variety is very high. But the total demand of the product line and the demand proportions of characteristic features would be easier to estimate. Thus, the aggregate demand of products in the product line is fixed and demand proportions of each feature is estimated. Then the aggregated demand is disaggregated among products using these feature proportions in order to obtain individual demand scenarios.

4.1. THE STOCHASTIC INTEGER PROGRAM

We formulate the model as an extensive form of stochastic integer program in which stochastic demand is represented by a number of demand scenarios each of which is assigned a probability of occurrence. The model then turns out to be a large scale integer programming model with the objective function that minimizes the expected value of inventory costs for vanilla boxes and penalty cost for the lost demand. We simultaneously determine the optimal vanilla configuration and their inventory levels, the allocation of finite assembly capacity both to vanilla boxes and products, and the allocation of vanilla box inventory to different products in each period over a planning horizon of three periods for each demand scenario.

The notation used in the formulation is as follows. The products are indexed by *i*, the components by *j* and the vanilla boxes by *k*.

Parameters:

N	number of different products in the product line ($i=1,\ldots,N$);
n	number of components ($j=1,\ldots,n$);
K	number of different types of vanilla boxes being used ($k=0,\ldots,K$) ($k=0$ is used to represent the assembly from raw components.);
L	number of demand scenarios considered ($l=1,\ldots,L$);
u_{kj}	contents of the k^{th} vanilla box in terms of the component *j* (0-1 coefficient);
a_{ij}	bill of material (BOM) for the product *i* in terms of the component *j* (0-1 coefficient);

r_{ik}	0-1 coefficient that indicates if product *i* can be assembled from vanilla box *k* (0 if vanilla box *k* cannot be used for the assembly of product *i*, 1 otherwise) (r_{i0} =1 for all *i*);
h_k	holding cost per unit per period for vanilla box *k*;
p_i	lost sale penalty cost per unit per period for unsatisfied demand of product *i*;
D_i^t	realized (observed) demand for product *i* in period *t*;
ε_i^t	random demand for product *i* in period *t*;
ε	random demands for all products in all periods (matrix);
t_k	assembly time (capacity usage) of vanilla box *k* (= number of components in vanilla box $k = \sum_{j=1}^{n} u_{kj}$);
t_{ik}	assembly time (capacity usage) of product *i* when it is assembled from vanilla box *k* (= (number of components in the product)-(number of components in vanilla box *k*) $= \sum_{j=1}^{n} a_{ij} - \sum_{j=1}^{n} u_{kj}$);
t_{i0}	assembly time (capacity usage) of product *i* when it is assembled from its raw components (= number of components in the product $= \sum_{j=1}^{n} a_{ij}$);
C^t	total assembly capacity in period *t*.

Variables:

q_{i0}^{t}	production quantity of product *i* that is assembled from raw components in period *t*;
q_{ik}^{t}	production quantity of product *i* that is assembled from vanilla box *k* in period *t*;
q	quantity of products assembled during the recourse step (matrix), q_l refers to quantity of products assembled in scenario *l* (vector);
I_{k}^{t}	inventory level of vanilla box *k* at the beginning of period *t*;
I	vanilla inventory (vector);
v_{k}^{t}	production quantity of vanilla box *k* in period *t* (can be used for the assembly of products in period *t*+1);
C_{FG}^{t}	final assembly capacity in period *t*;
C_{VB}^{t}	vanilla box assembly capacity in period *t*;
S_{i}^{t}	unsatisfied quantity of product i in period *t*;
U	vanilla box configuration (matrix), U_k refers to *k*th vanilla box (vector);

$Q(I, \varepsilon)$ cost function when demand is ε.

The objective is to minimize the expected cost of holding vanilla boxes and penalty cost for unsatisfied demand. The model determines:

- Which vanilla box configuration should be used for the current period T?
- How many of each type of vanilla boxes should be assembled in period T to be used in the next period, T+1?
- How to produce the products whose demands are realized in the current period T (from raw components or vanilla boxes or both)?
- How should vanilla boxes be allocated to finished products?

The stochastic integer program is formulated as follows:

$$\min_{I,U} E\,[Q\,(I,\,\varepsilon)]$$

where $Q\,(I,\,\varepsilon) = \min_{q} \sum_{k=1}^{K} h_k (I_k^{T+1} + I_k^{T+2} + I_k^{T+3}) + \sum_{i=1}^{N} p_i \left(S_i^T + S_i^{T+1} + S_i^{T+2}\right)$

subject to

$$\sum_{k=1}^{K} t_k v_k^t = C_{VB}^t \qquad \forall t \qquad (1)$$

$$I_k^{t+1} = v_k^t + I_k^t - \sum_{i=1}^{N} r_{ik} q_{ik}^t \qquad \forall t,k \qquad (2)$$

$$\sum_{i=1}^{N} \sum_{k=0}^{K} t_{ik} r_{ik} q_{ik}^t = C_{FG}^t \qquad \forall t \qquad (3)$$

$$\sum_{k=0}^{K} r_{ik} q_{ik}^T + S_i^T = D_i^T \qquad \forall i \qquad (4)$$

$$\sum_{k=0}^{K} r_{ik} q_{ik}^{T+1} + S_i^{T+1} = \varepsilon_i^{T+1} \qquad \forall i \qquad (5)$$

$$\sum_{k=0}^{K} r_{ik} q_{ik}^{T+2} + S_i^{T+2} = \varepsilon_i^{T+2} \qquad \forall i \qquad (6)$$

$$\sum_{i=1}^{N} r_{ik} q_{ik}^t \le I_k^t \qquad \forall k,t \qquad (7)$$

$$C_{VB}^{t} + C_{FG}^{t} \leq C^{t} \qquad \forall t \qquad (8)$$

$$q_{ik}^{t} \in Z^{+} \qquad \forall i,k,t \qquad (9)$$

$$C_{VB}^{t}, C_{FG}^{t}, v_{k}^{t}, S_{i}^{t}, I_{k}^{t} \in R^{+} \qquad \forall i,k,t \qquad (10)$$

In the above formulation, constraint (1) assigns the total vanilla box assembly time used in a period to VBA capacity of the period. Constraint (2) is the inventory balance equation for vanilla boxes. Constraint (3) finds the capacity used for the product assembly in a period t as the final assembly capacity of the period t. Constraints (4), (5) and (6) restrict the amount of product produced in a period to be less than or equal to the realized/expected demand in that period in order to avoid excess production in that period. Constraint (7) restricts the amount of vanilla box of a particular type that can be used during a period to be less than the available inventory at the beginning of the period. Constraint (8) restricts the sum of the capacities required for both vanilla box assembly and final product assembly to the overall assembly capacity in a period. The constraints (9) and (10) are integrality and non-negativity constraints on the variables, respectively. Note that only the variables q_{ik}^{t}'s are restricted to be integer. All other variables are left as continuous variables, since they take on integer values in the solutions because of their dependence on q_{ik}^{t}'s. Solving the LP-relaxation of this integer program instead would lead to erroneous solutions, since the product demands are very low (even 1 or 2 in some cases) due to high level of product variety.

4.2. THE SOLUTION METHOD

The most straightforward method of solving a stochastic program is to develop a large scale integer linear program where each of the demand scenarios is assigned a probability of occurrence and the optimal values are found that minimizes the expected value of the objective function. In this way, the stochastic integer program is reduced to a large scale integer program. The resulting IP is as follows.

Notation is the same as the stochastic integer model above. The subscript *l* is now added to identify the value of variables in the *l*th demand scenario. Additional parameter $prob_l$ represents the probability of occurrence for demand scenario *l*. ε_{il}^{T+1} and ε_{il}^{T+2} are the realizations of product *i* demand for the two periods ahead in scenario *l*.

$$\min_q \sum_{l=1}^{L} prob_l \left(\sum_{k=1}^{K} h_k \left(I_{kl}^{T+1} + I_{kl}^{T+2} + I_{kl}^{T+3}\right) + \sum_{i=1}^{N} p_i \left(S_{il}^{T} + S_{il}^{T+1} + S_{il}^{T+2}\right) \right)$$

subject to

$$\sum_{k=1}^{K} t_k v_{kl}^t = C_{VB,l}^t \qquad \forall t,l \qquad (11)$$

$$I_{kl}^{t+1} = v_{kl}^t + I_{kl}^t - \sum_{i=1}^{N} r_{ik} q_{ikl}^t \qquad \forall t,k,l \qquad (12)$$

$$\sum_{i=1}^{N} \sum_{k=0}^{K} t_{ik} r_{ik} q_{ikl}^t = C_{FG,l}^t \qquad \forall t,l \qquad (13)$$

$$\sum_{k=0}^{K} r_{ik} q_{ikl}^T + S_{il}^T = D_i^T \qquad \forall i,l \qquad (14)$$

$$\sum_{k=0}^{K} r_{ik} q_{ikl}^{T+1} + S_{il}^{T+1} = \varepsilon_{il}^{T+1} \qquad \forall i,l \qquad (15)$$

$$\sum_{k=0}^{K} r_{ik} q_{ikl}^{T+2} + S_{il}^{T+2} = \varepsilon_{il}^{T+2} \qquad \forall i,l \qquad (16)$$

$$\sum_{i=1}^{N} r_{ik} q_{ikl}^t \le I_{kl}^t \qquad \forall k,t,l \qquad (17)$$

$$C_{VB,l}^t + C_{FG,l}^t \le C^t \qquad \forall t,l \qquad (18)$$

$$q_{ikl}^t \in Z^+ \qquad \forall i,k,t,l \qquad (19)$$

$$C_{VB,l}^t, C_{FG,l}^t, v_{kl}^t, S_{il}^t, I_{kl}^t \in R^+ \qquad \forall i,k,t,l \qquad (20)$$

For the model size used in the experiments (number of components: 10, number of products: 72, number of vanilla boxes: 200, number of scenarios: 31), the solution time is approximately 2 minutes for the integer program. In order to get an idea about the solution time of a larger size model, an example input data including 21 components, 720 products, 1905 vanilla boxes and 15 scenarios is generated and solved. Although this size is much more larger than a possible real-sized problem (in a real environment, although the number of products can increase up to that number, it is impossible for the number of vanilla boxes to be so high due to physical infeasibilities), the solution time of the problem is approximately 2 hours. For a planning model that would be solved once in a week, that solution time may still seem reasonable.

In order to compare VBA process to ATO process, an integer program is developed for ATO process. Since in ATO case all products are produced from components, the variables and parameters related to vanilla boxes are eliminated from the VBA model above. The objective function of the model for ATO process includes only the penalty costs associated with the unsatisfied demand. The integer program for ATO process is as follows. The notation used is the same as in the model for VBA process above.

$$\min_{q} \sum_{l=1}^{L} prob_l \sum_{i=1}^{N} p_i \left(S_{il}^{T} + S_{il}^{T+1} + S_{il}^{T+2} \right)$$

subject to

$$\sum_{i=1}^{N} t_{i0} q_{i0l}^{t} \leq C^{t} \qquad \forall t,l \tag{21}$$

$$q_{i0l}^{T} + S_{il}^{T} = D_{i}^{T} \qquad \forall i,l \tag{22}$$

$$q_{i0l}^{T+1} + S_{il}^{T+1} = \varepsilon_{il}^{T+1} \qquad \forall i,l \tag{23}$$

$$q_{i0l}^{T+2} + S_{il}^{T+2} = \varepsilon_{il}^{T+2} \qquad \forall i,l \tag{24}$$

$$q_{i0l}^{t} \in Z^{+} \qquad \forall i,t,l \tag{25}$$

$$S_{il}^{t} \in R^{+} \qquad \forall i,t,l \tag{26}$$

In the formulation of ATO process, constraint (21) restricts the capacity required for the product assembly in period t by the available assembly capacity in period t. Constraints (22), (23) and (24) restrict the amount of product produced in a period to be less than or equal to the realized demand /realization of demand in scenario *l* in that period in order to avoid excess production in that period. The rest of the constraints are integrality and non-negativity constraints on the variables. It should be noted that the production quantities of products in a period are limited either by the available final-assembly capacity or the demand quantities. The model can be solved separately for each period. The one-period model then is a knapsack problem, allocating the available capacity in a period to several products based on their shortage penalty costs.

In the VBA and ATO models above, the same environmental restrictions are set. It is implicitly assumed that there is no scarcity of components in the assembly process, i.e., infinite supply of components is assumed in both processes. Final product inventory is not transferred from period to period. Total assembly capacities are equal and unsatisfied demand is not backordered, but lost, in both processes. If some of the environmental conditions change, like for example, unsatisfied demand being backordered instead of being lost, the results of the comparison might have changed more in favor of ATO. Furthermore, it should be noted that in case the VBA process selects to assemble all product requirements directly from raw components, the process reduces to an ATO process.

CHAPTER V

EXPERIMENTAL ANALYSIS

In this chapter, we summarize our computational study and the various insights that we have obtained. We intend to analyze the benefits and costs of vanilla box assembly process in different environments, as well as the impact of different factors on the total cost and demand fill rate as a service criterion. Specifically we search for the conditions under which it is worthwhile to implement VBA process rather than ATO process, how total costs and demand fill rates change with respect to changes in available capacity, penalty cost to holding cost ratio, commonality of components in products and distribution of demand among periods.

In the following section, we define our product structure and state why the factors like capacity, penalty cost, level of commonality of components in products and distribution of demand among periods are selected as the main factors. We also describe how the levels of them are determined.

5.1. EXPERIMENTAL FACTORS

In order to analyze the costs and benefits of VBA process and compare it to ATO process, a full factorial experiment is constructed. For the experiment, four factors (capacity, penalty cost to holding cost ratio, level of commonality in products and distribution of demand among periods) are specified. By the full factorial experiment, we try to examine all effects of the environment by changing all possible levels of the experimental factors determined.

Capacity is selected as a factor in order to examine the effect of different total assembly capacity levels on the usage of vanilla boxes in the assembly process. The aim is to examine at which capacity levels, the VBA process is more beneficial than ATO process. Five levels of capacity are determined, according to the average total capacity required in the three periods of planning horizon.

Penalty cost, as another experimental factor, is defined as per unit per period cost incurred due to demand unmet in the period it occurs. Since the unmet demand in a period is lost, penalty cost is in fact the lost profit per unit per period. The ratio of penalty cost to holding cost is a critical factor for determining which process, VBA or ATO, should be chosen in order to minimize the total cost. Penalty cost has five levels determined according to the possible penalty cost – holding cost ratios.

Level of commonality in products is chosen as a factor as well. The aim of choosing it as an experimental factor is to analyze the effect of common components in VBA process in terms of holding and penalty costs, and to examine whether VBA process is more beneficial or not when commonality in products is increased. Two levels, namely high and low level of common components, are tried in the experiments.

Another factor is the distribution of demand over periods. The aggregated product demand of three periods is assumed to be almost fixed. However, the effect of different distributions of this aggregated demand (disaggregation) over periods is inspected. Six different distributions are determined, namely increasing trend, decreasing trend, first-increase-then-decrease, first-decrease-then-increase, fixed and mixed. In increasing (decreasing) distribution, demand increases (decreases) starting from period one to period three, but with different standard deviations. Similarly, in first-increase-then-decrease (first-decrease-then-increase) distribution, demand increases (decreases) from period one to period two but decreases (increases) from period two to period three, again with different standard deviations. In fixed distribution, the demand of each period is the same. In mixed distribution, there is not a certain pattern, but all the scenarios defined above are likely.

The aim of taking the distribution of demand over periods as an experimental factor is to analyze the benefit (if any) of VBA process in different demand distributions over periods and to obtain an insight about the use of pre-knowledge on the distribution of demand in order to determine whether VBA process should be preferred or not.

5.2. DATA GENERATION

The effects of several environmental characteristics as capacity, penalty cost, demand distribution over periods and level of commonality of components across products are investigated by constructing the integer models for VBA and ATO with the specific levels of experimental factors.

Since the computer industry is focused on as the industry, the example product line is assumed to be "*personal computers for home*" product line. In the study, a product structure consisting of five features is considered. Three of these features are defined as *necessary features* meaning that each product must have all these three features. Two of the features are *optional* and need not to be possessed by all products. Each feature has two options, resulting in a total number of 10 different components (since each feature with a different option is defined as a component) and a product with a certain feature can include only one of these two options.

Sample product structure is presented in Table 1.

Necessary features: r_1, r_2, r_3

Optional features: o_1, o_2

Components: r_{11}, r_{12}, r_{21}, r_{22}, r_{31}, r_{32}, o_{11}, o_{12}, o_{21}, o_{22}

Table 1. Sample product structure

FEATURE	**OPTION**	**FEATURE TYPE**	**COMPONENT**
Motherboard		required	
	Celeron		r11
	Pentium 4		r12
Central Processing Unit (CPU)		required	
	Intel Celeron		r21
	Intel Pentium 4		r22
Memory		required	
	256 MB SDRAM		r31
	512 MB SDRAM		r32
Sound card		optional	
	5.1		o11
	7.1		o12
Graphic card		optional	
	64 MB		o21
	128 MB		o22

Since there is not a specified and limited product portfolio offered, the manufacturer can produce any combination of these five features, assuming that there is no physical and technical infeasibility in their assembly, or combination of features.

The manufacturer can offer 2^3=8 different products considering only the necessary features; and each optional feature (o_{11}, o_{21}, o_{12}, o_{22}) and combinations of them can be included to each of these 8 products, totally resulting in 72 end products, as shown in Table 18 in Appendix A.

After determining final products and their bill of materials (BOM) in terms of components, feasible vanilla boxes and their BOMs are found. It is obvious that vanilla boxes such as (r_{11}, r_{12}) ... (o_{11}, o_{12}) and all other vanilla boxes including these as subsets are infeasible. Total number of feasible vanilla boxes is 200. BOMs for vanilla boxes are shown in Table 19 in Appendix A.

In order to carry out the full factorial experiment, demand scenarios and values for the levels of the experimental factors are defined first. Then, to obtain the numerical results for the specified analysis, the data is generated to be utilized in the integer program.

Demand Scenarios

The market demand is stochastic, thus, to introduce random market demand into the problem, a scenario-based approach is utilized. For the scenario-based approach, several integer alternative values are generated for each period.

In fact, in a real life problem, the demand of the first period is known since assembly process begins after the demand is realized. However, in order to generate unbiased experimental data, the demand of the first period is also included in the scenarios.

Since the assembly line dealt with includes the products that belong to the same product line, the total demand of the product family is assumed to be known and fixed. It is assumed that the aggregated demand of all 72 products is 900 in total for three periods.

For the generation of demand scenarios, the total demand of 900 is distributed over three periods with respect to the standard deviation. Among these different scenarios, 31 of them are selected in order to use in the experiments according to their standard deviations. The selected scenarios are presented in Table 2.

Table 2. Generated demand scenarios

Scenario no.	Demand of period 1	Demand of period 2	Demand of period 3	Standard deviation
1	0	0	900	520
2	0	900	0	520
3	50	50	800	433
4	50	800	50	433
5	100	150	650	304
6	100	300	500	200
7	100	500	300	200
8	100	650	150	304
9	150	100	650	304
10	150	650	100	304
11	200	300	400	100
12	200	400	300	100
13	250	300	350	50
14	250	350	300	50
15	300	100	500	200
16	300	200	400	100
17	300	250	350	50
18	300	300	300	0
19	300	350	250	50
20	300	400	200	100
21	300	500	100	200
22	350	250	300	50
23	350	300	250	50
24	400	200	300	100
25	400	300	200	100
26	500	100	300	200
27	500	300	100	200
28	650	100	150	304
29	650	150	100	304
30	800	50	50	433
31	900	0	0	520

Then the aggregate demand of each period is distributed over individual products according to the components they include, using the feature specific option percentages. It is assumed that the percentage demand of each option in each feature is known and fixed. For example, it may be known that 25% of the customers select option 1 for feature r_1 if their product includes that feature. Then percentage of option 1 in feature r_1 is 25%. If the total demand is 100, then total demand of products that include r_{11} is 25, demand of products that include r_{12} is

75. Similarly, assuming that percentage of option 1 of feature r_2 is 25%, then the total demand of products that include both r_{11} and r_{21} is approximately 6% (0,25*0,25=0,0625). With these demand percentages defined for each option of each feature, total demand of each period is disaggregated over individual products. Without loss of generality, the percentages for options are taken as 25% for the first option and 75% for the second option for all features.

By the design of our experiments, it is assumed that demands of individual products are negatively correlated.

Level of Commonality

In order to change the level of commonality of components without changing any other environmental characteristic, the demand percentages of options defined above are changed for feature r_1. The percentage of both option 1 and option 2 of feature r_1 is changed to %50, decreasing the level of component commonality among end product demands. Thus, two levels of common components are determined as low commonality (when percentages of r_{11} and r_{12} are 50%) and high commonality (when percentage of r_{11} is 25% and percentage of r_{12} is 75%).

Distribution of Demand over Periods

Six different demand distributions (increasing, decreasing, first-increase-then-decrease, first-decrease-then-increase, fixed, mixed) over periods are generated by changing the probabilities of occurrences for the scenarios in each experiment. In mixed distribution, it is assumed that the probabilities of all 31 scenarios are equal. In the other distributions, for example in increasing trend distribution, the scenarios that have increasing demand pattern only through the time horizon are selected and total probability (1.00) is divided among them equally. In fixed distribution only the eighteenth scenario (where the demand of every three period is 300) is used, giving a probability of 1.00 to scenario 18 and probability of 0.00 to other scenarios). The probability of occurrence for the scenarios in each distribution factor is given in Table 3.

Penalty Cost

Levels of penalty cost are determined according to its ratio to inventory holding cost. According to the sample product structure, the average number of components in products is 4. Since the unit holding cost for each component is one unit, the unit holding cost for each product is 4 units on the average. In order to examine the cost and benefits of vanilla box assembly process in different environments related to penalty cost to holding cost ratio, five different levels for penalty cost is determined as 3, 5, 7, 10 and 15 where the ratio of them to holding cost is 0.75, 1.25, 1.75, 2.50 and 3.75, respectively. Only the penalty cost of 3 represents an environment where the inventory holding cost is greater than the penalty cost. The others represent environments with greater penalty cost than holding cost with various ratios.

Capacity

Different capacity levels are determined considering the average assembly time needed for three periods. As mentioned above, according to the product structure determined, the average number of components in products is 4. Since the total demand of three periods is assumed to be 900 and the assembly time of each component is one time unit, total capacity needed for three periods is 3600 time units on the average. This refers to a capacity of 1200 time units in each period.

Table 3. Probability of scenarios for different demand distributions over periods

scenario number	mixed	increasing	decreasing	first-increase-then-decrease	first-decrease-then-increase	fixed
1	0.033	0.200	0.000	0.000	0.000	0.000
2	0.033	0.000	0.000	0.100	0.000	0.000
3	0.033	0.200	0.000	0.000	0.000	0.000
4	0.033	0.000	0.000	0.100	0.000	0.000
5	0.033	0.200	0.000	0.000	0.000	0.000
6	0.033	0.200	0.000	0.000	0.000	0.000
7	0.033	0.000	0.000	0.100	0.000	0.000
8	0.033	0.000	0.000	0.100	0.000	0.000
9	0.033	0.000	0.000	0.000	0.110	0.000
10	0.033	0.000	0.000	0.100	0.000	0.000
11	0.033	0.200	0.000	0.000	0.000	0.000
12	0.033	0.000	0.000	0.100	0.000	0.000
13	0.033	0.000	0.000	0.000	0.110	0.000
14	0.033	0.000	0.000	0.100	0.000	0.000
15	0.033	0.000	0.000	0.000	0.110	0.000
16	0.033	0.000	0.000	0.000	0.110	0.000
17	0.033	0.000	0.000	0.000	0.110	0.000
18	0.033	0.000	0.000	0.000	0.000	1.000
19	0.033	0.000	0.000	0.100	0.000	0.000
20	0.033	0.000	0.000	0.100	0.000	0.000
21	0.033	0.000	0.000	0.100	0.000	0.000
22	0.033	0.000	0.000	0.000	0.110	0.000
23	0.033	0.000	0.170	0.000	0.000	0.000
24	0.033	0.000	0.000	0.000	0.110	0.000
25	0.033	0.000	0.170	0.000	0.000	0.000
26	0.033	0.000	0.000	0.000	0.110	0.000
27	0.033	0.000	0.170	0.000	0.000	0.000
28	0.033	0.000	0.000	0.000	0.110	0.000
29	0.033	0.000	0.170	0.000	0.000	0.000
30	0.033	0.000	0.170	0.000	0.000	0.000
31	0.033	0.000	0.170	0.000	0.000	0.000

At the beginning, it was thought that this capacity level (1200) should be the medium level capacity to be considered in the experiments. However, in the sample experiments, it was seen that this level of capacity refers to very tight capacity in both vanilla box assembly process and assemble to order process. This may be due to the fact that, the distribution of demand among periods is not stable in most of the scenarios, and standard deviation of demand is high. Even in some scenarios, a total demand of 900 units is expected to be satisfied only in one period. Therefore, five capacity levels are determined based on the results of the sample experiments as 1200, 1400, 1600, 2000 and 2400 time units of assembly capacity which refers to a capacity utilization of 100%, 86%, 75%, 60% and 50%, respectively, on the average.

5.3. GAMS MODEL

In order to solve the integer program, the optimization software General Algebraic Modeling (GAMS) is used. This optimization software provides a high-level language for the compact representation of large and complex models, allows changes to be made in model specification simply and safely, and permits model descriptions that are independent of solution algorithms.

Since the model gets larger as the number of scenarios and products gets larger, the GAMS/CPLEX solver is selected. GAMS/CPLEX allows for combining the high level modeling capabilities of GAMS with the power of CPLEX optimizers. While numerous solving options are available, GAMS/CPLEX automatically calculates and sets most options at the best values for specific problems. An example for the integer model applied in GAMS is presented in Appendix B.

5.4. ANALYSIS

In order to analyze the benefits obtained by applying the VBA process, analysis is conducted on the performance measures, namely total cost and average fill rate over three periods. Although the total cost and average fill rate are calculated over three periods, only the values of the first (current) period would be the real total cost and fill rate values, and the values obtained for the next two periods would change in rolling horizon according to the realized demands in these periods. The fill rate for a problem instance is computed as follows:

Expected penalty cost/unit penalty cost=expected number of unsatisfied demand

Fill rate=1 – (expected number of unsatisfied demand / total demand)

Total demand over three periods is 900 in both processes. Expected penalty cost is directly the objective function value for ATO process, while in VBA process it is found by subtracting the expected holding cost from the expected total cost (objective function value).

The results of the integer programs solved by GAMS/CPLEX, considering all combinations regarding all levels of each factor, are given in Appendix C. Totally, 600 runs have been made.

5.4.1. INITIAL OBSERVATIONS

According to the results of the integer programs for VBA and ATO processes, some immediate conclusions can be drawn:

In all distribution types except the fixed distribution, total cost decreases as capacity increases for the same level of penalty cost. Total cost decreases as penalty cost decreases at the same level of capacity in both VBA and ATO processes. In the fixed distribution, since all capacities are above or equal to the required capacity, the total cost is always zero for both probabilities.

When penalty cost is 3, total costs and fill rates of VBA process and ATO process do not differ. This is because, when inventory holding cost is higher than the cost of unsatisfied demand, vanilla boxes are not produced and stocked to satisfy the next periods' demands, and VBA process reduces to an ATO process.

When penalty cost is 3, in all capacity levels, increasing demand distribution has the highest total cost. Cost decreases according to the distribution in the order of decreasing, first-increase-then-decrease, mixed, first-decrease-then-increase and fixed distributions. This order of distributions is valid for all penalty cost and capacity levels in ATO process. As penalty cost increases, the order changes in VBA process to decreasing, increasing, mixed, first-increase-then-decrease, first-decrease-then-increase and fixed distributions according to the descending order of total cost.

Total cost and fill rate differences between VBA process and ATO process (on the favor of VBA process) are the highest (2454 and 0.34 respectively) when commonality is high, distribution of demand is increasing, penalty cost is highest (15) and capacity level is lowest (1200). In general, VBA process is more beneficial than ATO process when demand distribution is increasing or first-increase-then-decrease or first-decrease-then-increase.

VBA and ATO processes do not differ in fixed or decreasing distributions for all levels of penalty cost and capacity. Two processes give the same total cost and fill rate values, because VBA process reduces to ATO process since there is no usage of vanilla boxes in these distribution types. If capacity is higher than the required capacity in the first period, then there is excess capacity to produce vanilla boxes, but since the next two periods' demands can be satisfied by producing the products directly by the assembly of their components, there is no need to produce vanilla boxes. If capacity is lower than the required capacity in the first period, then there is not any excess capacity to produce vanilla boxes. In both scenarios, VBA process reduces to ATO process yielding the same total cost and fill rate values.

The difference between VBA process and ATO process is summarized on the total cost (fill rate) versus penalty cost graphs at different capacity levels and demand distribution types for high commonality. Since the behaviors of the two systems, VBA and ATO, are almost the same at different capacity levels, the graph of only one capacity level is presented.

In Figure 2, it is seen that total cost increases as penalty cost increases in both ATO and VBA processes in mixed demand distribution. However, as penalty cost increases, the difference between the cost values of ATO and VBA processes increases in favor of the VBA process.

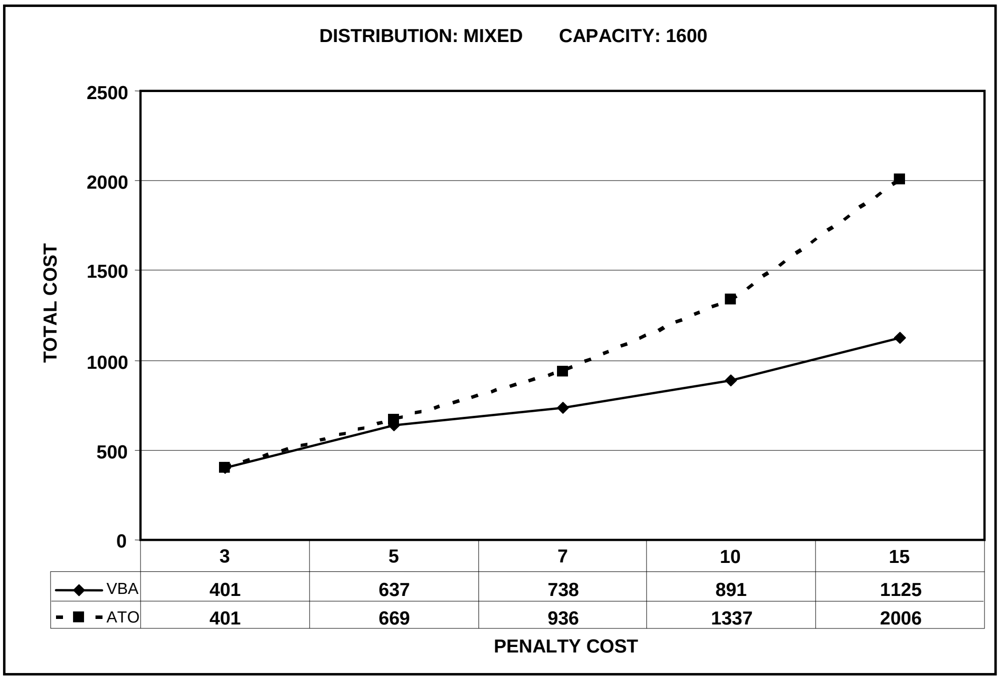


Figure 2. Total cost versus penalty cost for ATO and VBA processes in mixed distribution and 1600 time units of capacity

For a capacity level of 1600, the percentage decrease of total cost in VBA process according to ATO process is 0%, 5%, 21%, 33% and 44% for penalty cost levels of 3, 5, 7, 10 and 15 units respectively. Table 4 shows the percentage of decrease in total cost for all levels of capacity in mixed distribution.

Table 4. Percentage decrease of total cost in VBA process according to ATO process in mixed distribution and high level of component commonality

Penalty Cost	3	5	7	10	15
Capacity: 1200	0%	5%	14%	21%	31%
Capacity: 1400	0%	5%	18%	27%	38%
Capacity: 1600	0%	5%	21%	33%	44%
Capacity: 2000	0%	3%	21%	35%	45%
Capacity: 2400	0%	1%	20%	34%	45%

As seen from the table, in mixed distribution where all scenarios have the same probability of occurrence, VBA process is better than ATO in all levels of factors except when the penalty cost is 3 units. The gain obtained from implementing VBA process increases, as penalty cost increases. For high levels of penalty cost, the gain of VBA process increases as capacity increases, however, after a certain capacity level (2000), the gain of VBA process remains the same or even decreases because in high capacity levels, the cost of ATO process decreases more than the VBA process and thus the difference between VBA process and ATO process decreases.

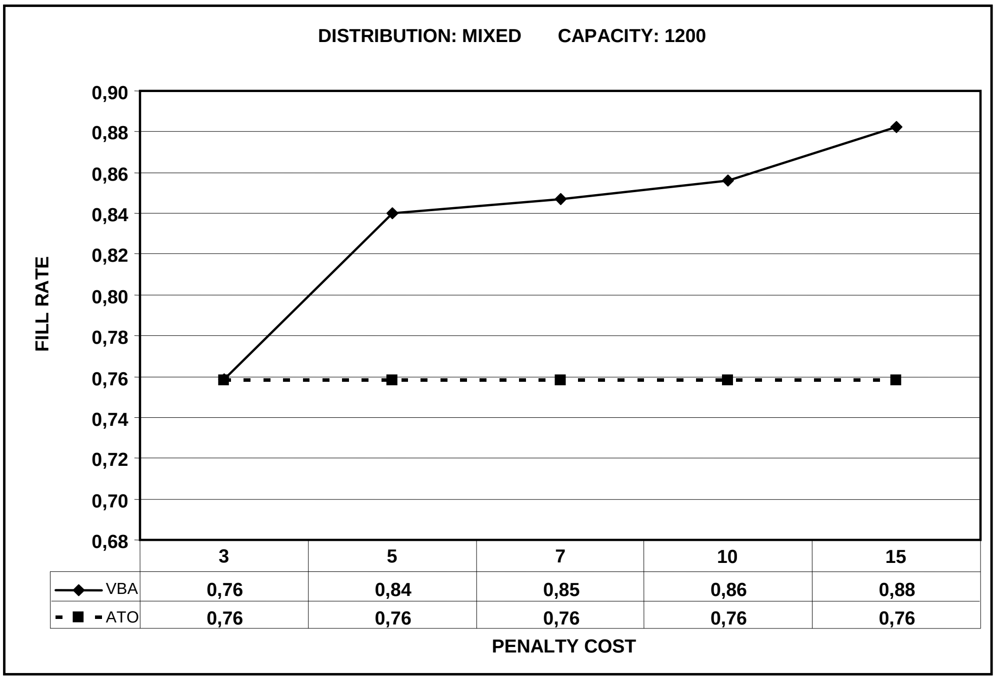
Table 5 shows the percentage of increase in fill rate gained by implementing VBA process rather than ATO process for all levels of capacity and penalty cost in mixed distribution.

Table 5. Percentage increase of fill rate in VBA process according to ATO process in mixed distribution and high level of component commonality

Penalty Cost	**3**	**5**	**7**	**10**	**15**
Capacity: 1200	0%	11%	12%	13%	16%
Capacity: 1400	0%	9%	11%	13%	13%
Capacity: 1600	0%	8%	11%	11%	11%
Capacity: 2000	0%	5%	7%	7%	7%
Capacity: 2400	0%	2%	4%	4%	4%

For higher levels of capacity, the increase in fill rate obtained by implementing VBA process is lower. As penalty cost increases for the same capacity level, the increase in fill rate increases. For high levels of both capacity and penalty cost, the decrease in total cost is high although the increase in fill rate is the same.

In Figure 3, the change in the value of fill rate with penalty cost is presented. In ATO process, the fill rate is stable for the same level of capacity, but increases with increasing capacity. In VBA process, fill rate increases as penalty cost increases at all capacity levels.



	3	5	7	10	15
VBA	0,76	0,84	0,85	0,86	0,88
ATO	0,76	0,76	0,76	0,76	0,76

Figure 3. Fill rate versus penalty cost for ATO and VBA processes in mixed distribution and 1200 time units of capacity

As presented in Figure 4, when distribution of demand is increasing throughout the periods, total cost of VBA process increases as penalty cost increases for a while, but then it remains stable since all demand is resulting in a fill rate of 1.00. The number of penalty cost levels where total cost is stable is 3 when capacity level is 2000. This number decreases, as capacity decreases since fill rate reaches to 1.00 in higher penalty cost levels when capacity is low.

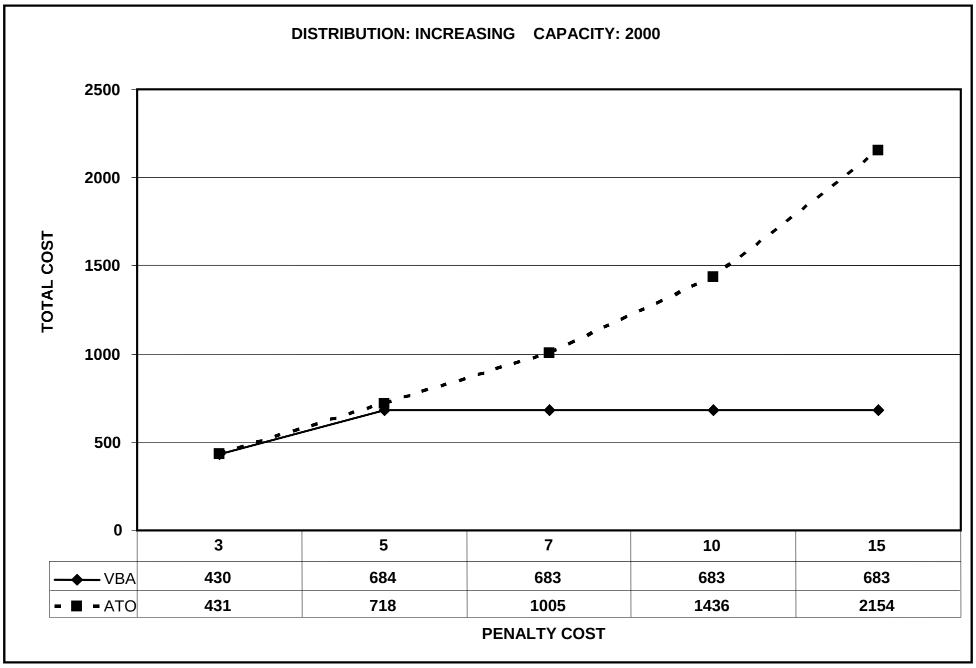


Figure 4. Total cost versus penalty cost graph for ATO and VBA processes in increasing demand distribution and 2000 time units of capacity

The percentage decrease of total cost in VBA process according to ATO process for all levels of capacity in increasing demand distribution is represented in Table 6.

Table 6. Percentage decrease of total cost in VBA process according to ATO process in increasing distribution and high level of component commonality

Penalty Cost	**3**	**5**	**7**	**10**	**15**
Capacity: 1200	0%	9%	20%	29%	52%
Capacity: 1400	0%	9%	25%	38%	58%
Capacity: 1600	0%	8%	31%	49%	66%
Capacity: 2000	0%	5%	32%	52%	68%
Capacity: 2400	0%	1%	30%	51%	67%

The percentages representing the gain by implementing VBA process reach their highest values in increasing distribution of demand, because the value of holding inventory increases since the second and third periods' demands cannot be satisfied with the capacity of those periods.

Table 7 shows the percentage of increase in fill rate gained by implementing VBA process rather than ATO process.

Table 7. Percentage increase of fill rate in VBA process according to ATO process in increasing distribution and high level of component commonality

Penalty Cost	**3**	**5**	**7**	**10**	**15**
Capacity: 1200	0%	25%	25%	31%	53%
Capacity: 1400	0%	27%	27%	33%	41%
Capacity: 1600	0%	26%	28%	30%	32%
Capacity: 2000	0%	15%	19%	19%	19%
Capacity: 2400	0%	11%	11%	11%	11%

Similar to equal distribution type, the gain of fill rate obtained by implementing VBA process decreases as capacity increases. For the same capacity level, high penalty cost increases the percentage increase in fill rate. For high levels of both capacity and penalty cost, the decrease in total cost is high although the increase in fill rate is the same.

In Figure 5, the change in the value of fill rate with penalty cost is presented. In ATO process, the fill rate is stable for the same level of capacity, but increases with increasing capacity. In VBA process, fill rate increases as penalty cost increases until it reaches to 1.00 at all capacity levels.

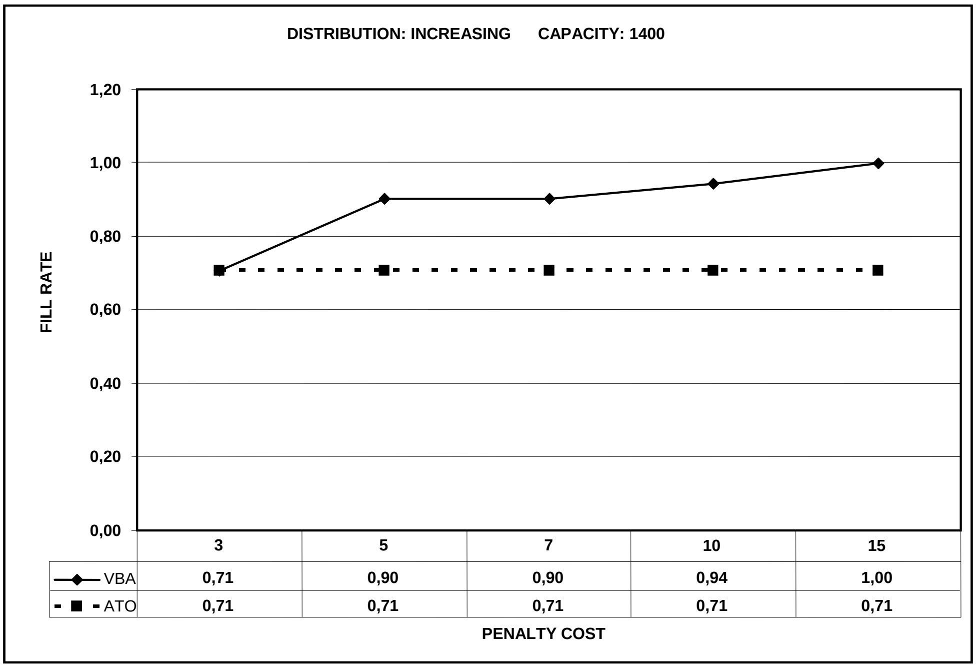
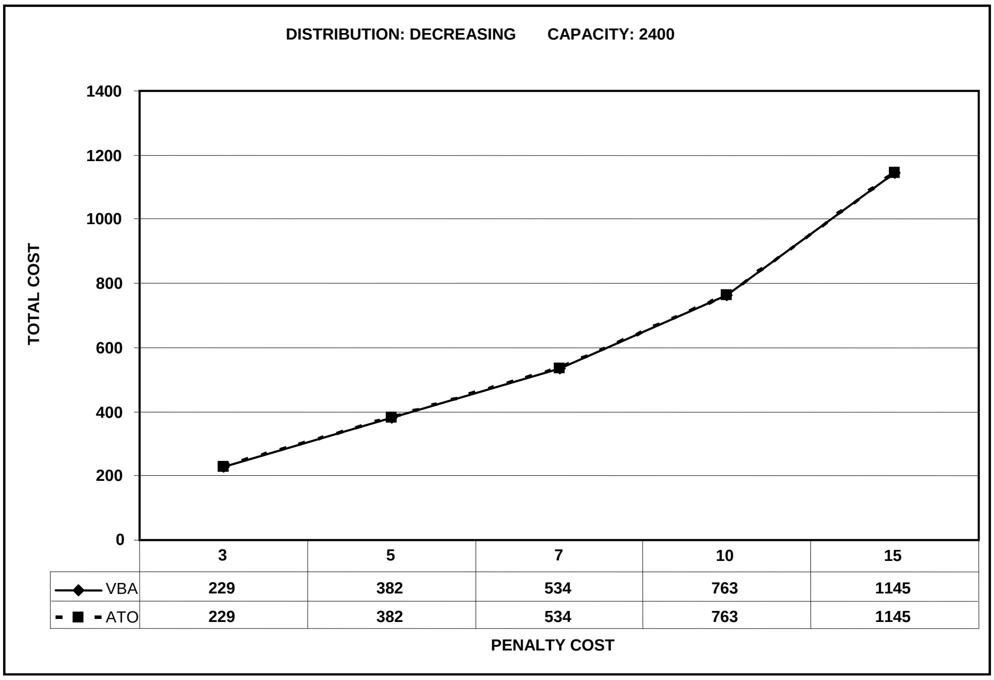


Figure 5. Fill rate versus penalty cost for ATO and VBA processes in increasing distribution and 1400 time units of capacity

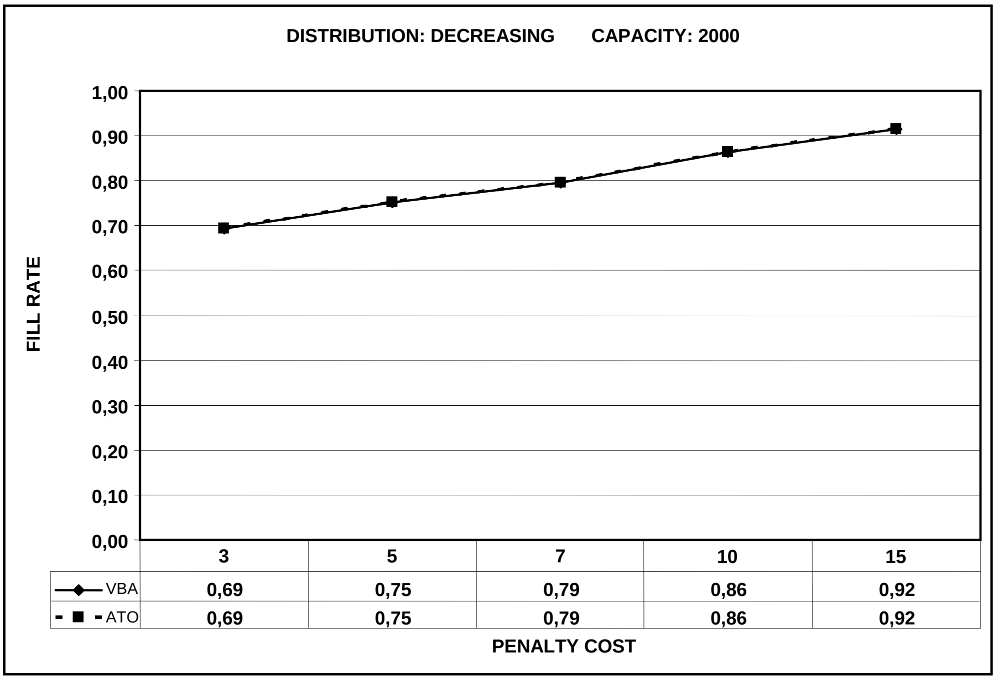
Figure 6 represents the change in the value of total cost according to penalty cost, when distribution of demand is decreasing throughout the periods. Total cost of VBA process and ATO process are the same for each level of capacity and penalty cost in decreasing distribution, meaning that implementing VBA process has no additional gain if the demand is decreasing throughout the periods. Because the highest demand in that distribution is in the first period and vanilla boxes cannot be used in the same period they are produced. Therefore, there is no need and in fact, no capacity to produce vanilla boxes and VBA process reduces to ATO process.

Figure 7 shows the change in fill rates with penalty cost. There is again no difference between VBA and ATO processes.



	3	5	7	10	15
VBA	229	382	534	763	1145
ATO	229	382	534	763	1145

Figure 6. Total cost versus penalty cost for ATO and VBA processes in decreasing distribution and 2400 time units of capacity



	3	5	7	10	15
VBA	0,69	0,75	0,79	0,86	0,92
ATO	0,69	0,75	0,79	0,86	0,92

Figure 7. Fill rate versus penalty cost for ATO and VBA processes in decreasing distribution and 2000 time units of capacity

As presented in Figure 8, when distribution of demand is first-increase-then-decrease, total cost of VBA process increases as penalty cost increases.

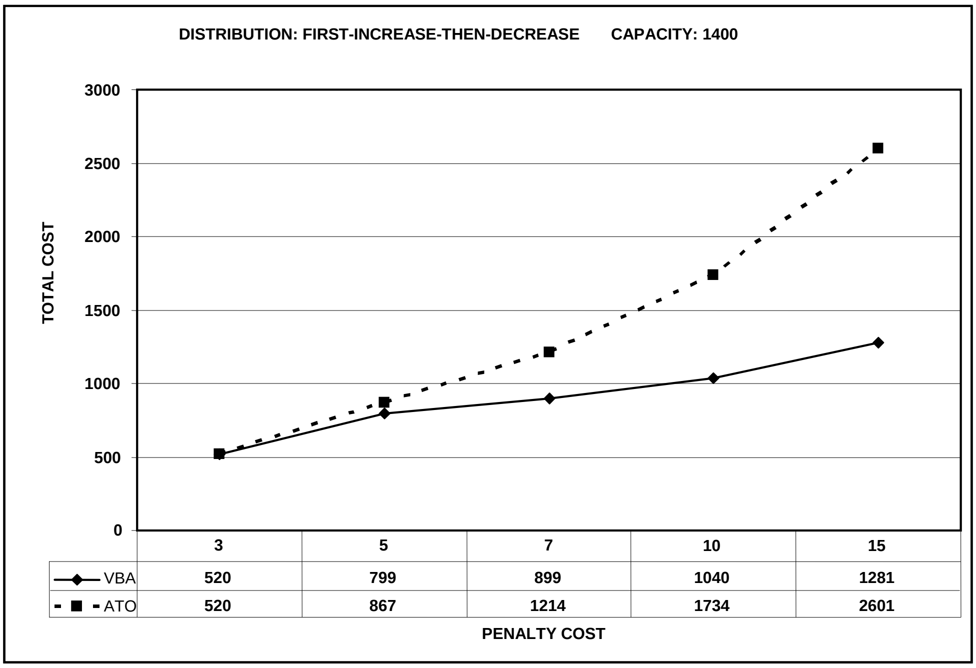


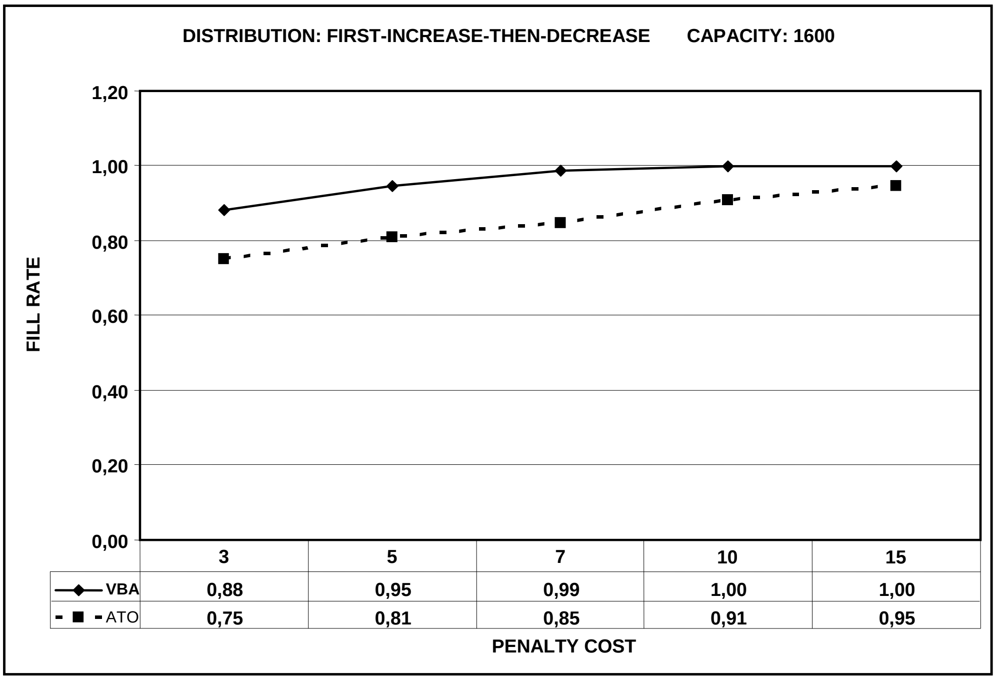
Figure 8. Total cost versus penalty cost for ATO and VBA processes in first-increase-then-decrease distribution and 1400 time units of capacity

The percentage decrease of total cost in VBA process according to ATO process for all levels of capacity in first-increase-then-decrease distribution is as shown in Table 8.

Table 8. Percentage decrease of total cost in VBA process according to ATO process in first-increase-then-decrease distribution and high level of component commonality.

Penalty Cost	**3**	**5**	**7**	**10**	**15**
Capacity: 1200	0%	8%	21%	30%	38%
Capacity: 1400	0%	8%	26%	40%	51%
Capacity: 1600	0%	7%	31%	49%	63%
Capacity: 2000	0%	4%	32%	52%	68%
Capacity: 2400	1%	2%	30%	51%	67%

Fill rate versus penalty cost graph for the same distribution is presented in Figure 9. It is similar to the graph in increasing distribution. However, as can be seen from Table 9, in first-increase-then-decrease distribution, the value of gain in fill rate is not as high as the gain in increasing distribution. Another difference is that, in the case of first-increase-then-decrease distribution, the gain in fill rate does not increase as penalty cost increases after a penalty cost level of 7. It means that for the same capacity level, the fill rate increase obtained by implementing VBA process is not affected by the penalty cost factor for high levels of it. However, the decrease in total cost obtained by implementing VBA process goes on increasing as penalty cost increases at the same capacity level.



	3	5	7	10	15
VBA	0,88	0,95	0,99	1,00	1,00
ATO	0,75	0,81	0,85	0,91	0,95

Figure 9. Fill rate versus penalty cost for ATO and VBA processes in first-increase-then-decrease distribution and 1600 time units of capacity

Table 9. Percentage increase of fill rate in VBA process according to ATO process in first-increase-then-decrease distribution and high level of component commonality

Penalty Cost	**3**	**5**	**7**	**10**	**15**
Capacity: 1200	0%	14%	17%	18%	18%
Capacity: 1400	0%	13%	17%	17%	17%
Capacity: 1600	0%	9%	16%	16%	16%
Capacity: 2000	0%	7%	10%	10%	10%
Capacity: 2400	0%	3%	6%	6%	6%

Figure 10 presents the change in value of total cost according to penalty cost, when distribution of demand is first-decrease-then-increase. Although the demand of the first period is high similar to decreasing distribution, in that case, VBA process is beneficial than ATO process in high levels of penalty cost. Contrary to decreasing distribution, vanilla boxes produced in the second period can be used for the demand of third period leading to higher demand satisfaction than ATO process in the third period.

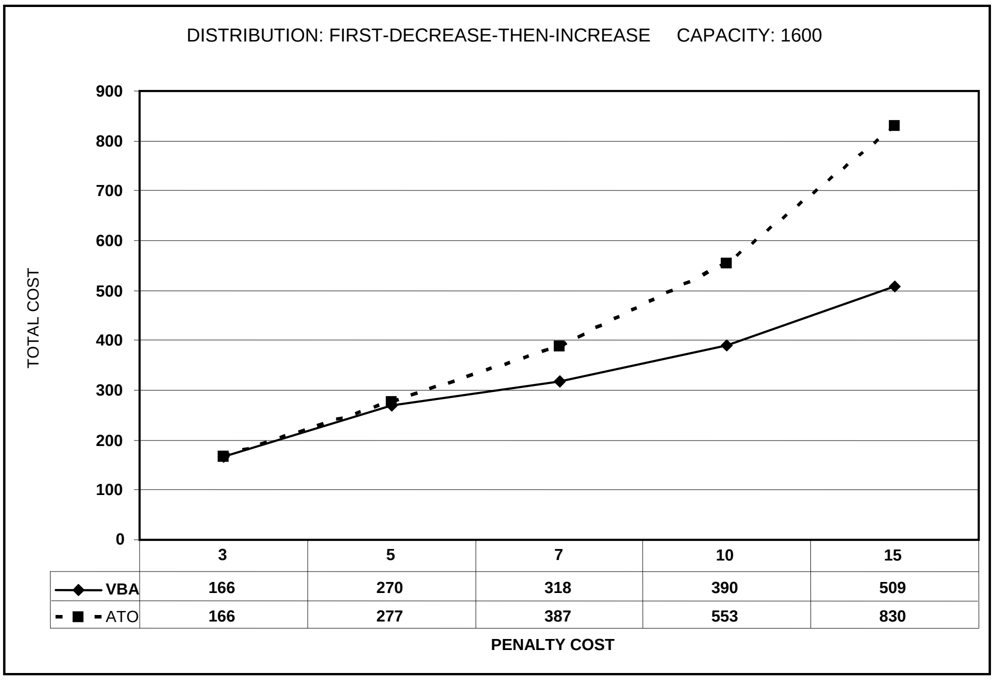


Figure 10. Total cost versus penalty cost for ATO and VBA processes in first-decrease-then-increase distribution and 1600 time units of capacity

Table 10 shows the percentage decrease of total cost in VBA process according to ATO process for all levels of capacity in first-decrease-then-increase distribution. The gain from VBA process in that distribution is lower than the gain in increasing and first-increase-then-decrease distributions. Similar to other distributions, after a certain capacity level (in that case it is 1600), the gain of VBA process remains the same or decreases because in high capacity levels, the cost of ATO process decreases more than the VBA process and thus the difference between VBA process and ATO process decreases.

Table 10. Percentage decrease of total cost in VBA process according to ATO process in first-decrease-then-increase distribution and high level of component commonality

Penalty Cost	**3**	**5**	**7**	**10**	**15**
Capacity: 1200	0%	4%	15%	24%	35%
Capacity: 1400	0%	3%	17%	27%	38%
Capacity: 1600	0%	3%	18%	29%	39%
Capacity: 2000	1%	1%	15%	25%	33%
Capacity: 2400	0%	3%	15%	25%	34%

Table 11 shows the percentage increase of fill rate in VBA process according to ATO process for all levels of capacity in first-decrease-then-increase distribution. The percentages of increases in fill rate are lower than both increasing and first-increase-then-decrease distributions. When capacity level is 2400, both VBA and ATO processes reaches to a fill rate of 1.00 resulting a 0% increase in fill rate. However, VBA process reaches to the same fill rate value with a lower total cost.

In Figure 9, fill rate versus penalty cost graph for the first-decrease-then-increase distribution is presented for a capacity level of 1600.

Table 11. Percentage increase of fill rate in VBA process according to ATO process in first-decrease-then-increase distribution and high level of component commonality

Penalty Cost	3	5	7	10	15
Capacity: 1200	0%	7%	7%	7%	9%
Capacity: 1400	0%	5%	5%	5%	6%
Capacity: 1600	0%	4%	4%	4%	4%
Capacity: 2000	0%	1%	1%	1%	1%
Capacity: 2400	0%	0%	0%	0%	0%

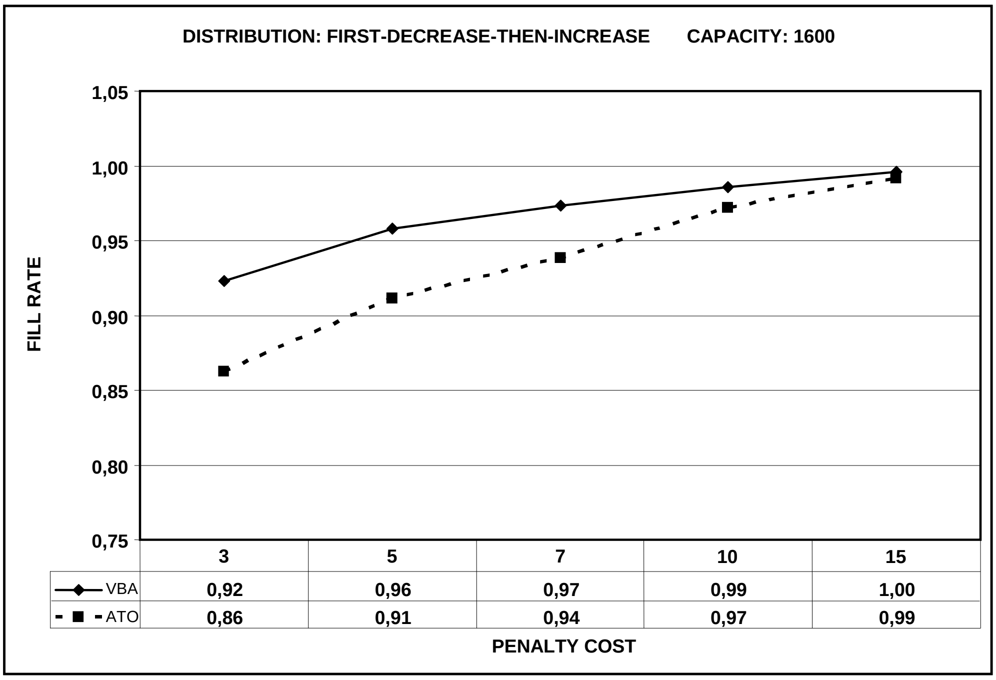


Figure 11. Fill rate versus penalty cost for ATO and VBA processes in first-decrease-then-increase distribution and 1600 time units of capacity

All of the above graphics and results according to the effects of penalty cost, demand distribution and capacity levels apply to the case where component commonality is low. However, in low component commonality, total costs of both VBA process and ATO process are higher than the values obtained in high component commonality case. Especially in high levels of demand and penalty cost, the gain obtained from VBA process increases at the case of high component commonality.

5.4.2. STATISTICAL ANALYSIS

In this section, the values of total cost and average fill rate over three periods obtained in VBA process and ATO process are analyzed statistically utilizing Duncan’s test. Duncan’s test is a result-guided test that compares the treatment means, while controlling the comparisonwise error rate and uses a least significance range value for sets of adjacent means (Hines and Montgomery, 1990). Duncan’s test is selected since it tells whether a given mean differs from a given number of adjacent means.

Statistical analyses are applied to cases where VBA process gives better results than ATO process in terms of both total cost and average fill rate. For these cases, additional results are obtained by changing the probability of occurrences of demand scenarios in order to apply the Duncan’s test, because the probability of occurrence is the only random factor dealt with.

We formulate the Duncan’s test in MS Excel. The formulation of the test and the mean square error values obtained in each case are presented in Appendix D.

The cases that are analyzed statistically are increasing, first-increase-then-decrease and first-decrease-then-increase demand distributions with factor levels of 5, 7, 10, 15 for penalty cost and 1200, 1400, 1600, 2000, 2400 for capacity. Mixed distribution is not included in the analysis, because if the probabilities of occurrences of all demand scenarios are changed randomly, the effect of demand distribution among periods would influence the results of the analyses.

For increasing, first-increase-then-decrease and first-decrease-then-increase distributions, five additional probability distributions are determined as shown in Tables 12, 13 and 14. Using these different demand scenario probabilities and factor levels of penalty cost and capacity as defined above, the integer program is run and combining the results of these runs with the previous ones, Duncan's test is applied for total cost and fill rate values. Means of total cost and fill rate values for VBA and ATO processes are compared and it is determined whether the performance of VBA process is significantly better than ATO process or not using the test value of Duncan's test with 95% confidence interval. The factor levels where the difference between VBA process and ATO process is found significant in favor of VBA process are represented with a mark (*) in the Tables 15, 16 and 17, that are formed for each demand distribution type. Insignificant differences are represented by a dot (.).

Table 12. Probability of scenarios for increasing demand distribution

SCENARIO NO.	**INCREASING 1 (ORIGINAL)**	**INCREASING 2**	**INCREASING 3**	**INCREASING 4**	**INCREASING 5**	**INCREASING 6**
1	0.2	0.3	0.1	0.2	0.05	0.3
3	0.2	0.25	0.15	0.3	0.05	0.3
5	0.2	0.2	0.2	0.05	0.3	0.3
6	0.2	0.15	0.25	0.2	0.3	0.05
11	0.2	0.1	0.3	0.25	0.3	0.05

Table 13. Probability of scenarios for first-increase-then-decrease demand distribution

SCENARIO NO.	FIRST-INCREASE-THEN-DECREASE 1 (ORIGINAL)	FIRST-INCREASE-THEN-DECREASE 2	FIRST-INCREASE-THEN-DECREASE 3	FIRST-INCREASE-THEN-DECREASE 4	FIRST-INCREASE-THEN-DECREASE 5	FIRST-INCREASE-THEN-DECREASE 6
2	0.10	0.25	0.05	0.05	0.10	0.05
4	0.10	0.20	0.05	0.05	0.15	0.05
7	0.10	0.10	0.05	0.15	0.05	0.20
8	0.10	0.10	0.05	0.05	0.15	0.10
10	0.10	0.10	0.05	0.10	0.20	0.05
12	0.10	0.05	0.10	0.15	0.05	0.25
14	0.10	0.05	0.10	0.15	0.05	0.15
19	0.10	0.05	0.10	0.15	0.05	0.05
20	0.10	0.05	0.20	0.05	0.10	0.05
21	0.10	0.05	0.25	0.10	0.10	0.05

Table 14. Probability of scenarios for first-decrease-then-increase demand distribution

SCENARIO NO.	FIRST-DECREASE-THEN-INCREASE 1 (ORIGINAL)	FIRST-DECREASE-THEN-INCREASE 2	FIRST-DECREASE-THEN-INCREASE 3	FIRST-DECREASE-THEN-INCREASE 4	FIRST-DECREASE-THEN-INCREASE 5	FIRST-DECREASE-THEN-INCREASE 6
9	0.11	0.15	0.05	0.10	0.25	0.05
13	0.11	0.15	0.05	0.20	0.05	0.10
15	0.11	0.15	0.05	0.20	0.05	0.10
16	0.11	0.15	0.05	0.10	0.05	0.25
17	0.11	0.15	0.10	0.05	0.20	0.05
22	0.11	0.10	0.15	0.05	0.15	0.20
24	0.11	0.05	0.15	0.05	0.15	0.05
26	0.11	0.05	0.20	0.10	0.05	0.10
28	0.11	0.05	0.20	0.15	0.05	0.10

Table 15 shows the results of the Duncan’s test for the three demand distribution types. As mentioned earlier, the percentages representing the gain by implementing VBA process reach their highest values in increasing distribution of demand. Thus, as expected, most of the differences are found to be significant in increasing distribution. VBA process results a significant fill rate increase according to ATO process in all levels of capacity and penalty cost (except when penalty cost is 3). On the other hand, the decrease in total cost due to implementing VBA process is not significant for capacity level 2400 unless the penalty cost is 15. In a low level of penalty cost, VBA process gives lower total cost value than ATO only when capacity is tight. In medium or large capacities, total cost does not decrease, but only the average fill rate increases significantly. Thus, it can be concluded that, if demand is known to have an increasing trend, then VBA process performs better than ATO process in terms of fill rates in all factor levels, when the penalty cost is higher than the inventory holding cost.

As seen from the results of the Duncan’s test for first-increase-then-decrease demand distribution in the table, VBA process does not significantly differ from ATO process when penalty cost is low. The value of gain in fill rate and the value of decrease in total cost are significantly high when penalty cost is high and capacity is tight or medium.

The results of first-decrease-then-increase demand distribution are similar to those obtained in case of first-increase-then-decrease distribution of demand. However, in first-decrease-then-increase distribution, VBA process performs significantly better than ATO process in terms of fill rate even in low penalty cost in tight capacity. When capacity is large, the gain obtained from implementing VBA process becomes insignificant for both total cost and fill rate values, because in high capacity levels, both VBA and ATO processes reaches a fill rate of 1.00.

Table 15. Duncan’s test results for high level of component commonality

Increasing Distribution					
	PENALTY COST	**5**	**7**	**10**	**15**
TOTAL COST	**Capacity: 1200**	*	*	*	*
	Capacity: 1400	*	*	*	*
	Capacity: 1600	.	*	*	*
	Capacity: 2000	.	*	*	*
	Capacity: 2400	.	.	.	*
FILL RATE	**Capacity: 1200**	*	*	*	*
	Capacity: 1400	*	*	*	*
	Capacity: 1600	*	*	*	*
	Capacity: 2000	*	*	*	*
	Capacity: 2400	*	*	*	*

First-Increase-Then-Decrease Distribution					
	PENALTY COST	**5**	**7**	**10**	**15**
TOTAL COST	**Capacity: 1200**	.	*	*	*
	Capacity: 1400	.	*	*	*
	Capacity: 1600	.	*	*	*
	Capacity: 2000	.	.	*	*
	Capacity: 2400	.	.	.	*
FILL RATE	**Capacity: 1200**	*	*	*	*
	Capacity: 1400	.	*	*	*
	Capacity: 1600	.	*	*	*
	Capacity: 2000	.	.	.	*
	Capacity: 2400	.	.	.	.

First-Decrease-Then-Increase Distribution					
	PENALTY COST	**5**	**7**	**10**	**15**
TOTAL COST	**Capacity: 1200**	*	*	*	*
	Capacity: 1400	.	*	*	*
	Capacity: 1600	.	.	*	*
	Capacity: 2000	.	.	.	.
	Capacity: 2400	.	.	.	.
FILL RATE	**Capacity: 1200**	*	*	*	*
	Capacity: 1400	*	*	*	*
	Capacity: 1600	*	*	*	*
	Capacity: 2000	.	.	.	.
	Capacity: 2400	.	.	.	.

Although our model for modular design of DPD is different from the model developed by Swaminathan and Tayur (1995, 1999) in a number of directions, some of the insights we have obtained for the VBA process are the same as theirs. Both studies indicate that under very tight capacity restrictions, vanilla boxes cannot be utilized, because there does not exist sufficient capacity to assemble products even from vanilla boxes; and in the other extreme of very high capacity case, products can be assembled directly from raw components within the period they are required and therefore VBA reduces to ATO, making ATO preferable to VBA.

CHAPTER VI

CONCLUSIONS AND FURTHER RESEARCH ISSUES

Delayed product differentiation (DPD), also referred to as postponement, is a strategy to achieve flexibility of the supply chain in meeting uncertain and changing customer demands and to reduce inventory and improve customer service performance simultaneously in an environment where there is a tradeoff between keeping inventory and satisfying customer demands due to high product variety. DPD delays the point in time when a product assumes its identity until customer orders are received.

In this study, we aim to form a general framework in the light of basic articles and classify the main types of DPD and associated studies in order to make it possible to examine and model the real cases more easily and specifically. We examine the articles in the literature according to this classification.

After the general framework is formed, we study on vanilla box assembly process, which can be related to modular design type of postponed manufacturing in DPD literature. We optimize the vanilla box assembly process, that is, we intend to find optimal configurations and inventory levels of the vanilla boxes, considering costs of inventory and unsatisfied demand in a capacitated assembly system in which several models of a product line with uncertain demand are assembled. In order to achieve this, we design the model as an extensive form of stochastic programming with a rolling horizon of three periods and solve it as a standard integer programming model that minimizes the expected value of the objective function. Then, we study the impact of product demand scenarios, distributions of demand throughout the periods, penalty cost, commonality of components and capacity restrictions on the performance measures like total cost and fill rate. Vanilla box assembly (VBA) process is also compared to assemble-to-order (ATO) process where all products are assembled after the demand is realized. We compare the costs and service levels of the two systems in order to find the environmental characteristics that support the use of VBA process and provide some insights on the performance of VBA process as well.

Our analyses show that VBA process outperforms the ATO process when distribution of demand throughout the periods is alternating between high and low values or has an increasing trend. Especially when the unit penalty cost of unsatisfied demand is higher than the unit inventory cost and under reasonable capacity constraints, VBA process outperforms ATO process. If demand is known to have an increasing trend, then VBA process performs better than ATO process in terms of fill rates in all factor levels, when the penalty cost is higher than the inventory holding cost. Furthermore, we observe that in case the demand distribution throughout periods is decreasing or fixed, the VBA process reduces to the ATO process. The same conclusion is valid for penalty cost level that is lower than inventory holding cost.

Our analyses also indicate that when the level of component commonality is high, the gain obtained by implementing VBA process is more than in case of low level of component commonality. Since the component commonality level is changed by changing the percentage demand of one option in one feature, the effect of common components cannot be observed as significant as thought it would be. However, if difference in component commonality had been made by another method without changing the effect of other factors, the effect of high level component commonality in increasing the gain obtained from VBA process might have been more obvious.

It is obvious that the values obtained for final product and vanilla box production quantities are not directly implementable. However, the expected values of the production quantities can be used in real life applications. All constraints, including the capacity constraints, are satisfied when the expected values are used for each variable. Only when the continuous expected values are rounded to integers, we cannot be certain that all constraints are satisfied.

Further Research Issues

The models for VBA and ATO processes are based on a three-period rolling horizon. However, we do not consider the implementation of rolling horizon method in our experiments. A more realistic and a more fair method of comparing the two processes, ATO and VBA, would be to simulate them on a rolling horizon basis by implementing the current period's decision only and then observe the actual total cost and the fill rate during the current period. A real-life system can be simulated with different realized demand values in each period of the 3-period horizon. In this method of comparison, cases may be observed in which ATO process outperforms VBA process in terms of total cost.

In this study, we assume that there is no need to redesign the products or processes for VBA process; hence the only cost of VBA process is the unit holding cost of vanilla boxes. But in some real cases, in order to implement VBA process, firms need to redesign their products and/or processes. For example, in order to increase component commonality among products, a component with higher cost can be used in a product although it is not necessary, or the sequence of the processes needs to be changed in order to utilize the vanilla boxes. In such cases, those additional costs (redesign costs) should be considered in determining the benefits of VBA process. As a future research, redesign cost can be included in the models in order to compare VBA process against ATO process more accurately.

Another future research direction can be the analysis of the environments where substitution among products/components and usage of redundant components is possible. Our model can be extended by introducing a cost for interchanging products/components in order to model substitution. In case of redundant components, the model can be extended by especially considering the optional features. It might be a good strategy to provide the customer with redundant optional features instead of losing the demand under stockout conditions. The model can be easily modified to address the component redundancy by introducing the cost of additional components included in the vanilla box but not required in the product.

We have been able to obtain results for a real-sized problem in reasonable computation times. However, the size of the model gets larger with the number of scenarios and components. For the large-scale industry size problems where the computation time of integer program is prohibitively long, the solution techniques, as the integer L-shaped method, dynamic programming and decomposition, can be utilized for solving the stochastic integer program. For using these techniques, the two-stage stochastic programming can be chosen for modeling, because the decision variables of the problem can be partitioned into two sets, the stages of the problem (first stage and second stage) should be made comletely independent from each other. This becomes possible if the vanilla box assembly capacity and the final product assembly capacity, which are now variables in our model, are predetermined and fixed. Then the stages could be defined as follows: the first stage variables correspond to the decisions that need to be made prior to the realization of demand (here and now decisions) which are the vanilla box configuration and amounts of vanilla boxes to be produced; the second stage decisions (wait and see decisions) are the allocation of vanilla boxes produced to the finished products and the amounts of finished products, given a realization of the random demand. In stochastic (mixed-) integer programming with recourse, integrality constraints are imposed on (some of) the first stage and/or second stage decision variables. When only the first stage decision variables are required to be integer, the problem can be approached using fairly conventional adaptations of stochastic linear programming (SLP) methods. When integrality is also required for the second stage decision variables, two difficulties arise. First, each evaluation of the second stage problem requires the solution of an integer program, which is in general NP-hard. Next, the recourse function does not conserve the desirable properties of continuous SLPs" (Kenyon and Morton, 2001). Since the properties for stochastic integer programs are mostly case specific, general efficient methods for solving them are lacking. Some techniques that attempt to exploit different properties of stochastic integer programs are discussed in the literature (Kenyon and Morton, 2001; Birge and Louveaux, 1997).

REFERENCES

[1] Y. Aviv and A. Federgruen, “The Benefits of Design for Postponement,” *Quantitative Modeling for Supply Chain Management*, S. Tayur, M. Magazine, R. Ganeshan, eds., Kluwer Academic Publishers, 1998, pp. 553-584.

[2] Y. Aviv and A. Federgruen, “Capacitated Multi-Item Inventory Systems with Random and Seasonally Fluctuating Demands: Implications for Postponement Strategies,” *Management Science*, vol. 47, no. 4, 2001a, pp. 512–531.

[3] Y. Aviv and A. Federgruen, “Design for Postponement: A Comprehensive Characterization of Its Benefits under Unknown Demand Distributions,” *Operations Research*, vol. 49, no. 4, 2001b, pp. 578–598.

[4] J.R. Birge and F. Louveaux, *Introduction to Stochastic Programming*, New York: Springer, 1997.

[5] D.J. Bowersox and D.J. Closs, *Logistical management: The Integrated Supply Chain Process*, McGraw-Hill, 1996. pp 16-19.

[6] A. Brown, H. Lee, and R. Petrakian, “Xilinx Improves Its Semiconductor Supply Chain Using Product and Process Postponement,” *Interfaces*, vol. 30, no. 4, 2000, pp. 65–80.

[7] P. Desai, S. Kekre, S. Radhakrishnan and K. Srivinasan, “Product Differentiation and Commonality in Design: Balancing Revenue and Cost Drivers,” *Management Science*, vol. 47, 2001, pp. 37–51.

[8] R. Ernst and B. Kamrad, “Evaluation of Supply Chain Structures through Modularization and Postponement,” *European Journal of Operational Research*, vol. 124, 2000, pp. 495-510.

[9] E. Feitzinger and H. L. Lee, “Mass Customization at Hewlett Packard: The Power of Postponement,” *Harvard Business Review*, January-February 1997, pp. 116-121.

[10] A. Garg, “An Application of Designing Products and Processes for Supply Chain Management,” *IIE Transactions*, vol. 31, no. 5, 1999, pp. 417-429.

[11] A. Garg and H. L.Lee, “Managing Product Variety: An Operations Perspective,” *Quantitative Modeling for Supply Chain Management,* S. Tayur, M. Magazine, R. Ganeshan, eds., Kluwer Academic Publishers, 1998.

[12] A. Garg and C.S. Tang, “On Postponement Strategies for Product Families with Multiple Points of Differentiation,” *IIE Transactions,* vol. 29, 1997, pp. 641–650.

[13] S. Gavirneni and S. Tayur, “Value of Information Sharing and Comparison with Delayed Differentiation,” *Quantitative Modeling for Supply Chain Management,* S. Tayur, M. Magazine, R. Ganeshan, eds., Kluwer Academic Publishers, 1998.

[14] D. Gupta and S. Benjaafar, “Make-to-Order, Make-to-Stock, or Delay Product Differentiation? A Common Framework for Modeling and Analysis,” *IIE Transactions,* vol. 36, 2004, pp. 529-546.

[15] S. Gupta and V. Krishnan, “Product Family-Based Assembly Sequence Design Methodology,” *IIE Transactions,* vol. 30, no. 10, 1998, pp. 933-945.

[16] D. He, A. Kusiak and T. Tseng, “Delayed Product Differentiation: A Design and Manufacturing Perspective,” *Computer-Aided Design*, vol. 30, no. 2, 1998, pp. 105-113.

[17] W.W. Hines and D.C. Montgomery, *Probability and Statistics in Engineering and Management Science,* Wiley, 1990. pp 353-409.

[18] A.S. Kenyon and D.P. Morton, “A Survey on Stochastic Location and Routing Problems” *Central European Journal of Operations Research,* vol.9, 2001, pp. 277-328.

[19] H.L. Lee, “Design for Supply Chain Management: Concepts and Examples,” *Perspectives in Operations Management,* R. Sarin ed., Kluwer Academic Publishers, 1993, pp. 43-65.

[20] H. L. Lee, “Effective Inventory and Service Management Through Product and Process Redesign,” *Operations Research,* vol. 44, no. 1, 1996, pp. 151-159.

[21] H. L. Lee, C. Billington, and B. Carter, “Hewlett-Packard Gains Control ff Inventory and Service through Design for Localization,” *Interfaces,* vol. 23, no. 4, 1993, pp. 1–11.

[22] H. L. Lee and C. S. Tang, “Modeling the Costs and Benefits of Delayed Product Differentiation,” *Management Science,* vol. 43, 1997, pp. 40–53.

[23] H. L. Lee and C. S. Tang, “Variability Reduction through Operations Reversal,” *Management Science,* vol. 44, no. 2, 1998, pp. 162–172.

[24] H. L. Lee and M. Sasser, “Product Universality and Design for Supply Chain Management,” *Production Planning and Control: Special Issue on Supply Chain Management*, vol. 6, no. 3, 1995, pp. 270–277.

[25] S. Ma, W. Wang, and L. Liu, “Commonality and Postponement in Multistage Assembly Systems,” *European Journal of Operations Research*, vol. 142, 2002, pp. 523-538.

[26] J. Shapiro, *Modeling the Supply Chain,* Duxbury, 2001, pp.382-387.

[27] D. Simchi-Levi, P. Kaminsky, and E. Simchi-Levi, *Designing and Managing the Supply Chain*, McGraw-Hill, 2000, pp.167-196.

[28] R. Srinivasan, R. Jayaraman, R. Roundy and S. Tayur, “Procurement of Common Components in Presence of Uncertainty,” *IBM Technical Report*, 1998.

[29] J. M. Swaminathan and S.R. Tayur, “Managing Broader Product Lines through Delayed Differentiation Using Vanilla Boxes,” *Working Paper, GSIA,* Carnegie Mellon University, PA, 1995.

[30] J. M. Swaminathan and S.R. Tayur, “Managing Broader Product Lines through Delayed Differentiation Using Vanilla Boxes,” *Management Science,* vol. 44, no. 12, 1998a.

[31] J. M. Swaminathan and S.R. Tayur, “Stochastic Programming Models for Managing Product Variety,” *Quantitative Modeling for Supply Chain Management,* S. Tayur, M. Magazine, R. Ganeshan, eds., Kluwer Academic Publishers, 1998b, pp. 585-622.

[32] J. M. Swaminathan and S.R. Tayur, “Managing Design of Assembly Sequences for Product Lines That Delay Product Differentiation,” *IIE Transactions,* vol. 31, 1999, pp. 1015-1026.

[33] S. Tayur, “Computing Optimal Stock Levels for Common Components in an Assembly System,” *Working Paper, GSIA,* Carnegie Mellon University, PA, 1994.

[34] R.I. Van Hoek, “The Rediscovery of Postponement A Literature Review And Directions For Research,” *Journal of Operations Management*, vol. 19, 2001, pp. 161-184.

[35] R.I. Van Hoek, B. Vos and H.R. Commandeur, “Restructuring European Supply Chains by Implementing Postponement Strategies,” *Long Range Planning*, vol. 32, no. 5, 1999, pp. 505-518.

APPENDIX A

BILL OF MATERIAL FOR FINAL PRODUCTS AND VANILLA BOXES

Table 16. Bill of material for final products

PRODUCT	COMPONENTS										PRODUCT	COMPONENTS									
	r11	r12	r21	r22	r31	r32	o11	o12	o21	o22		r11	r12	r21	r22	r31	r32	o11	o12	o21	o22
1	1		1		1						37		1	1		1					1
2	1		1			1					38		1	1			1				1
3	1			1	1						39		1		1	1					1
4	1			1		1					40		1		1		1				1
5		1	1		1						41	1		1		1		1		1	
6		1	1			1					42	1		1			1	1		1	
7		1		1	1						43	1			1	1		1		1	
8		1		1		1					44	1			1		1	1		1	
9	1		1		1		1				45		1	1		1		1		1	
10	1		1			1	1				46		1	1			1	1		1	
11	1			1	1		1				47		1		1	1		1		1	
12	1			1		1	1				48		1		1		1	1		1	
13		1	1		1		1				49	1		1		1		1			1
14		1	1			1	1				50	1		1			1	1			1
15		1		1	1		1				51	1			1	1		1			1
16		1		1		1	1				52	1			1		1	1			1
17	1		1		1			1			53		1	1		1		1			1
18	1		1			1		1			54		1	1			1	1			1
19	1			1	1			1			55		1		1	1		1			1
20	1			1		1		1			56		1		1		1	1			1
21		1	1		1			1			57	1		1		1			1	1	
22		1	1			1		1			58	1		1			1		1	1	
23		1		1	1			1			59	1			1	1			1	1	
24		1		1		1		1			60	1			1		1		1	1	
25	1		1		1				1		61		1	1		1			1	1	
26	1		1			1			1		62		1	1			1		1	1	
27	1			1	1				1		63		1		1	1			1	1	
28	1			1		1			1		64		1		1		1		1	1	
29		1	1		1				1		65	1		1		1			1		1
30		1	1			1			1		66	1		1			1		1		1
31		1		1	1				1		67	1			1	1			1		1
32		1		1		1			1		68	1			1		1		1		1
33	1		1		1					1	69		1	1		1			1		1
34	1		1			1				1	70		1	1			1		1		1
35	1			1	1					1	71		1		1	1			1		1
36	1			1		1				1	72		1		1		1		1		1

Table 17. Bill of material for vanilla boxes

VANILLA BOX	COMPONENTS										VANILLA BOX	COMPONENTS									
	r11	r12	r21	r22	r31	r32	o11	o12	o21	o22		r11	r12	r21	r22	r31	r32	o11	o12	o21	o22
1	1		1								101				1	1				1	
2	1			1							102				1	1					1
3		1	1								103				1		1			1	
4		1		1							104				1		1				1
5	1				1						105			1				1		1	
6	1					1					106			1				1			1
7		1			1						107			1					1	1	
8		1				1					108			1					1		1
9	1						1				109				1			1		1	
10	1							1			110				1			1			1
11		1					1				111				1				1	1	
12		1						1			112				1				1		1
13	1								1		113					1		1		1	
14	1									1	114					1		1			1
15		1							1		115					1			1	1	
16		1								1	116					1			1		1
17			1		1						117						1	1		1	
18			1			1					118						1	1			1
19				1	1						119						1		1	1	
20				1		1					120						1		1		1
21			1				1				121	1		1		1		1			
22			1					1			122	1		1		1			1		
23				1			1				123	1		1			1	1			
24				1				1			124	1		1			1		1		
25			1						1		125	1			1	1		1			
26			1							1	126	1			1	1			1		
27				1					1		127	1			1		1	1			
28				1						1	128	1			1		1		1		
29					1		1				129		1	1		1		1			
30					1			1			130		1	1		1			1		
31						1	1				131		1	1			1	1			
32						1		1			132		1	1			1		1		
33					1				1		133		1		1	1		1			
34					1					1	134		1		1	1			1		
35						1			1		135		1		1		1	1			
36						1				1	136		1		1		1		1		
37							1		1		137	1		1		1				1	
38							1			1	138	1		1		1					1
39								1	1		139	1		1			1			1	
40								1		1	140	1		1			1				1
41	1		1		1						141	1			1	1				1	
42	1		1			1					142	1			1	1					1
43	1			1	1						143	1			1		1			1	
44	1			1		1					144	1			1		1				1
45		1	1		1						145		1	1		1				1	
46		1	1			1					146		1	1		1					1
47		1		1	1						147		1	1			1			1	

Table 17. Bill of material for vanilla boxes (cont'd)

48		1		1		1					148		1	1			1				1
49	1		1				1				149		1		1	1				1	
50	1		1					1			150		1		1	1					1
51	1			1			1				151		1		1		1			1	
52	1			1				1			152		1		1		1				1
53		1	1				1				153	1		1				1		1	
54		1	1					1			154	1		1				1			1
55		1		1			1				155	1		1					1	1	
56		1		1				1			156	1		1					1		1
57	1		1						1		157	1			1			1		1	
58	1		1							1	158	1			1			1			1
59	1			1					1		159	1			1				1	1	
60	1			1						1	160	1			1				1		1
61		1	1						1		161		1	1				1		1	
62		1	1							1	162		1	1				1			1
63		1		1					1		163		1	1					1	1	
64		1		1						1	164		1	1					1		1
65	1				1		1				165		1		1			1		1	
66	1				1			1			166		1		1			1			1
67	1					1	1				167		1		1				1	1	
68	1					1		1			168		1		1				1		1
69		1			1		1				169	1				1		1		1	
70		1			1			1			170	1				1		1			1
71		1				1	1				171	1				1			1	1	
72		1				1		1			172	1				1			1		1
73	1				1				1		173	1					1	1		1	
74	1				1					1	174	1					1	1			1
75	1					1			1		175	1					1		1	1	
76	1					1				1	176	1					1		1		1
77		1			1				1		177		1			1		1		1	
78		1			1					1	178		1			1		1			1
79		1				1			1		179		1			1			1	1	
80		1				1				1	180		1			1			1		1
81	1						1		1		181		1				1	1		1	
82	1						1			1	182		1				1	1			1
83	1							1	1		183		1				1		1	1	
84	1							1		1	184		1				1		1		1
85		1					1		1		185			1		1		1		1	
86		1					1			1	186			1		1		1			1
87		1						1	1		187			1		1			1	1	
88		1						1		1	188			1		1			1		1
89			1		1		1				189			1			1	1		1	
90			1		1			1			190			1			1	1			1
91			1			1	1				191			1			1		1	1	
92			1			1		1			192			1			1		1		1
93				1	1		1				193				1	1		1		1	
94				1	1			1			194				1	1		1			1
95				1		1	1				195				1	1			1	1	
96				1		1		1			196				1	1			1		1
97			1		1				1		197				1		1	1		1	
98			1		1					1	198				1		1	1			1
99			1			1			1		199				1		1		1	1	
100			1			1				1	200				1		1		1		1

APPENDIX B

GAMS MODEL OF THE INTEGER PROGRAMMING MODEL OF VBA PROCESS

```
set i product / 1*72 /;
set j component / 1*10 /;
set k vanillabox / 1*200 /;
set t period / 1*4 /;
set l scenarios / 1*31 /;

parameter penalty(i) penalty cost for product i;
       penalty(i)=5;

parameter prob(l)  probability of scenario l

1     0.3              12    0               23    0
2     0                13    0               24    0
3     0.25             14    0               25    0
4     0                15    0               26    0
5     0.2              16    0               27    0
6     0.15             17    0               28    0
7     0                18    0               29    0
8     0                19    0               30    0
9     0                20    0               31    0/;
10    0                21    0
11    0.1              22    0


table u(k,j)  content of VB k in terms of components

      1     2     3     4     5     6     7     8     9    10
  1   1     0     1     0     0     0     0     0     0     0
  2   1     0     0     1     0     0     0     0     0     0
  3   0     1     1     0     0     0     0     0     0     0
  .
  .
199   0     0     0     1     0     1     0     1     1     0
200   0     0     0     1     0     1     0     1     0     1;

table a(i,j)  content of product i in terms of components

      1     2     3     4     5     6     7     8     9    10
 1    1     0     1     0     1     0     0     0     0     0
 2    1     0     1     0     0     1     0     0     0     0
 3    1     0     0     1     1     0     0     0     0     0
 .
 .
71    0     1     0     1     1     0     0     1      0    1
72    0     1     0     1     0     1     0     1      0    1;


parameter holding(k) holding cost per unit per period ;
       holding(k)= sum(j, u(k,j));
```

```
parameter initialinv(k) initial inventory for VB k;
       initialinv(k)=0;


table demand(i,l) demand for product i in scenario l at period 1

       1     2     3     4     5 ... 28    29    30    31
1      0     0     0     0     0 ...  3     3     3     4
2      0     0     1     1     1 ...  8     8     9    11
3      0     0     1     1     1 ...  8     8     9    11
4      0     0     2     2     4 ... 23    23    28    32
.
.
.
68     0     0     1     1     2 ... 13    13    16    18
69     0     0     0     0     1 ...  4     4     5     6
70     0     0     1     1     2 ... 13    13    16    18
71     0     0     1     1     2 ... 13    13    16    18
72     0     0     3     3     6 ... 39    39    47    53;

table expdemand2(i,l) expected demand for product i in scenario l at
period 2

       1     2     3     4     5 ... 28    29    30    31
1      0     4     0     3     1 ...  0     1     0     0
2      0    11     1     9     2 ...  1     2     1     0
3      0    11     1     9     2 ...  1     2     1     0
4      0    32     2    28     5 ...  4     5     2     0
.
.
.
68     0    18     1    16     3 ...  2     3     1     0
69     0     6     0     5     1 ...  1     1     0     0
70     0    18     1    16     3 ...  2     3     1     0
71     0    18     1    16     3 ...  2     3     1     0
72     0    53     3    47     9 ...  6     9     3     0;


table expdemand3(i,l) expected demand for product i in scenario l at
period 3

      1     2     3     4     5 ... 28    29    30    31
1     4     0     3     0     3 ...  1     0     0     0
2    11     0     9     1     8 ...  2     1     1     0
3    11     0     9     1     8 ...  2     1     1     0
4    32     0    28     2    23 ...  5     4     2     0
.
.
.
68   18     0    16     1    13 ...  3     2     1     0
69    6     0     5     0     4 ...  1     1     0     0
70   18     0    16     1    13 ...  3     2     1     0
71   18     0    16     1    13 ...  3     2     1     0
72   53     0    47     3    39 ...  9     6     3     0;


parameter r(k,i) coefficient of vanilla box usability;

r(k,i)$((a(i,'1') ge u(k,'1'))and(a(i,'2') ge u(k,'2'))and(a(i,'3') ge
u(k,'3'))and(a(i,'4') ge u(k,'4'))and(a(i,'5') ge u(k,'5'))and(a(i,'6')
ge u(k,'6'))and(a(i,'7') ge u(k,'7'))and(a(i,'8') ge
u(k,'8'))and(a(i,'9') ge u(k,'9'))and(a(i,'10') ge u(k,'10')))=1;
```

```
parameter VBasstime(k)  assembly time of vanilla box k ;
       VBasstime(k)= sum(j, u(k,j));

parameter FGVasstime(i,k)  assembly time of product i when it is
assembled from vanilla box k ;
       FGVasstime(i,k)= sum(j, a(i,j))- sum(j, u(k,j));


parameter FGRasstime(i)  assembly time of product i when it is assembled
from its raw components;
       FGRasstime(i)= sum(j, a(i,j));

variables
cost         the expected value of total inv holding cost and penalty
             cost over VB configuration and production quantities

qr(i,l,t)    production quantity of product i that is assembled from raw
             components at period t in the lth demand scenario

qv(i,k,l,t)  production quantity of product i that is assembled from
             vanilla box k at period t in the lth demand scenario

Inv(k,l,t)   inventory level of vanilla box k at the beginning of period
             t in the lth demand scenario

v(k,l,t)     production quantity of vanilla box k at period t

CVB(l,t)     vanila box assembly capacity at period t in the lth demand
             scenario

CFG(l,t)     final assembly capacity at period t in the lth demand
             scenario

S(i,l,t)     unsatisfied quantity of product i at period t in the lth
             demand scenario

totalholdingcost;

positive variables CVB, CFG, S, Inv, v, totalholdingcost;

integer variables qr, qv;

equations

costdef                   define objective function

InitialInventory (k,l)    initial inventory of VB k (at the beginning of
                          period 1)

VBassembly(l,t)           vanilla box assembly capacity for period t

Leftoverutil(k,t,l)       utilization of leftover VBs of previous
periods

FGassembly(l,t)           final assembly capacity for period t

DemandSatisfaction1 (i,l) amount of product produced for period 1 should
                          be less than or equal to the realized demand
                          at t for each product

DemandSatisfaction2 (i,l) amount of product produced for period 2 should
                          be less than or equal to the expected demand
                          at period 2 for each product
```

```
DemandSatisfaction3 (i,l) amount of product produced for period 3 should
                          be less than or equal to the expected demand
                          at period 3 for each product
DemandSatisfaction4 (i,l) amount of product produced for period 4 should
                          be less than or equal to the expected demand
                          at period 4 for each product

InventoryUsage (k,t,l)    VBs used in period t for demands of period t
                          should not exceed the vanilla box inventory at
                          the beginning of period t

TotalCapacity(l,t)        restricts the total capacity

HoldingCost               computes total holding cost


costdef..cost=e=sum(l,prob(l)*(sum(k,holding(k)*(Inv(k,l,'2')+Inv(k,l,'3
')+Inv(k,l,'4')))+sum(i,penalty(i)*(S(i,l,'1')+S(i,l,'2')+S(i,l,'3')))))
;

InitialInventory(k,l)..Inv(k,l,'1')=e=initialinv(k);

VBassembly(l,t)..sum(k,(VBasstime(k)*v(k,l,t)))=e=CVB(l,t);

InventoryUsage(k,t,l)..sum(i,r(k,i)*qv(i,k,l,t))=l=Inv(k,l,t);

Leftoverutil(k,t,l)..Inv(k,l,t+1)=e=v(k,l,t)+ Inv(k,l,t)-
sum(i,r(k,i)*qv(i,k,l,t));

FGassembly(l,t)..sum(i,sum(k,FGVasstime(i,k)*r(k,i)*qv(i,k,l,t))+FGRasst
ime(i)*qr(i,l,t))=e=CFG(l,t);

DemandSatisfaction1 (i,l)..sum(k, r(k,i)*qv(i,k,l,'1'))+qr(i,l,'1')+
S(i,l,'1')=e= demand(i,l);

DemandSatisfaction2 (i,l)..sum(k, r(k,i)*qv(i,k,l,'2'))+qr(i,l,'2')+
S(i,l,'2')=e= expdemand2(i,l);

DemandSatisfaction3 (i,l)..sum(k, r(k,i)*qv(i,k,l,'3'))+qr(i,l,'3')+
S(i,l,'3')=e= expdemand3(i,l);

DemandSatisfaction4 (i,l)..sum(k, r(k,i)*qv(i,k,l,'4'))+qr(i,l,'4')+
S(i,l,'4')=e= expdemand4(i,l);

HoldingCost..totalholdingcost=e=sum(l,prob(l)*(sum(k,holding(k)*(
Inv(k,l,'2')+Inv(k,l,'3')+Inv(k,l,'4')))));

TotalCapacity(l,t)..CVB(l,t)+CFG(l,t)=l=2400;

model vanilla /all/;
solve vanilla using mip minimizing cost;
display  qv.l,qr.l,v.l,S.l,Inv.l,CFG.l,CVB.l,totalholdingcost.l;
```

APPENDIX C

RESULTS OF THE GAMS MODEL FOR VBA PROCESS AND ATO PROCESS

Table 18. Results of GAMS model

FACTOR LEVELS				VBA PROCESS		ATO PROCESS	
Component Commonality	Distribution of Demand	Penalty Cost	Capacity	Total Cost	Fill Rate	Total Cost	Fill Rate
High	Mixed	3	1200	653	0,76	653	0,76
High	Mixed	3	1400	504	0,81	504	0,81
High	Mixed	3	1600	401	0,85	401	0,85
High	Mixed	3	2000	248	0,91	248	0,91
High	Mixed	3	2400	144	0,95	144	0,95
High	Mixed	5	1200	1029	0,84	1088	0,76
High	Mixed	5	1400	795	0,89	840	0,81
High	Mixed	5	1600	637	0,92	669	0,85
High	Mixed	5	2000	401	0,95	413	0,91
High	Mixed	5	2400	238	0,96	240	0,95
High	Mixed	7	1200	1304	0,85	1523	0,76
High	Mixed	7	1400	966	0,91	1176	0,81
High	Mixed	7	1600	738	0,94	936	0,85
High	Mixed	7	2000	456	0,97	578	0,91
High	Mixed	7	2400	270	0,98	336	0,95
High	Mixed	10	1200	1713	0,86	2176	0,76
High	Mixed	10	1400	1222	0,92	1681	0,81
High	Mixed	10	1600	891	0,94	1337	0,85
High	Mixed	10	2000	539	0,97	827	0,91
High	Mixed	10	2400	318	0,98	480	0,95
High	Mixed	15	1200	2244	0,88	3264	0,76
High	Mixed	15	1400	1570	0,92	2521	0,81
High	Mixed	15	1600	1125	0,95	2006	0,85
High	Mixed	15	2000	676	0,97	1238	0,91
High	Mixed	15	2400	398	0,98	720	0,95
High	Increasing	3	1200	945	0,65	945	0,65
High	Increasing	3	1400	790	0,71	791	0,71
High	Increasing	3	1600	652	0,76	653	0,76
High	Increasing	3	2000	430	0,84	431	0,84
High	Increasing	3	2400	269	0,90	269	0,90
High	Increasing	5	1200	1434	0,81	1575	0,65
High	Increasing	5	1400	1203	0,90	1318	0,71
High	Increasing	5	1600	1003	0,95	1088	0,76
High	Increasing	5	2000	684	0,96	718	0,84
High	Increasing	5	2400	443	1,00	449	0,90

Table 18. Results of GAMS model (cont'd)

High	Increasing	7	1200	1768	0,81	2205	0,65
High	Increasing	7	1400	1380	0,90	1845	0,71
High	Increasing	7	1600	1049	0,97	1523	0,76
High	Increasing	7	2000	683	1,00	1005	0,84
High	Increasing	7	2400	443	1,00	629	0,90
High	Increasing	10	1200	2246	0,85	3150	0,65
High	Increasing	10	1400	1645	0,94	2636	0,71
High	Increasing	10	1600	1120	0,99	2176	0,76
High	Increasing	10	2000	683	1,00	1436	0,84
High	Increasing	10	2400	443	1,00	898	0,90
High	Increasing	15	1200	2271	0,99	4725	0,65
High	Increasing	15	1400	1645	1,00	3954	0,71
High	Increasing	15	1600	1119	1,00	3264	0,76
High	Increasing	15	2000	683	1,00	2154	0,84
High	Increasing	15	2400	443	1,00	1347	0,90
High	Decreasing	3	1200	827	0,69	827	0,69
High	Decreasing	3	1400	672	0,75	672	0,75
High	Decreasing	3	1600	555	0,79	555	0,79
High	Decreasing	3	2000	366	0,86	366	0,86
High	Decreasing	3	2400	229	0,92	229	0,92
High	Decreasing	5	1200	1377	0,69	1378	0,69
High	Decreasing	5	1400	1120	0,75	1120	0,75
High	Decreasing	5	1600	925	0,79	925	0,79
High	Decreasing	5	2000	610	0,86	610	0,86
High	Decreasing	5	2400	382	0,92	382	0,92
High	Decreasing	7	1200	1928	0,69	1929	0,69
High	Decreasing	7	1400	1568	0,75	1568	0,75
High	Decreasing	7	1600	1295	0,79	1295	0,79
High	Decreasing	7	2000	854	0,86	854	0,86
High	Decreasing	7	2400	534	0,92	534	0,92
High	Decreasing	10	1200	2753	0,69	2756	0,69
High	Decreasing	10	1400	2241	0,75	2241	0,75
High	Decreasing	10	1600	1850	0,79	1850	0,79
High	Decreasing	10	2000	1221	0,86	1221	0,86
High	Decreasing	10	2400	763	0,92	763	0,92
High	Decreasing	15	1200	4130	0,69	4134	0,69
High	Decreasing	15	1400	3361	0,75	3361	0,75
High	Decreasing	15	1600	2774	0,79	2774	0,79
High	Decreasing	15	2000	1831	0,86	1831	0,86
High	Decreasing	15	2400	1145	0,92	1145	0,92
High	First-increase-then-decrease	3	1200	672	0,75	672	0,75
High	First-increase-then-decrease	3	1400	520	0,81	520	0,81
High	First-increase-then-decrease	3	1600	412	0,85	412	0,85
High	First-increase-then-decrease	3	2000	250	0,91	250	0,91
High	First-increase-then-decrease	3	2400	145	0,95	146	0,95
High	First-increase-then-decrease	5	1200	1030	0,86	1120	0,75
High	First-increase-then-decrease	5	1400	799	0,92	867	0,81

Table 18. Results of GAMS model (cont’d)

High	First-increase-then-decrease	5	1600	639	0,93	687	0,85
High	First-increase-then-decrease	5	2000	399	0,97	417	0,91
High	First-increase-then-decrease	5	2400	239	0,97	243	0,95
High	First-increase-then-decrease	7	1200	1244	0,88	1568	0,75
High	First-increase-then-decrease	7	1400	899	0,95	1214	0,81
High	First-increase-then-decrease	7	1600	662	0,99	962	0,85
High	First-increase-then-decrease	7	2000	399	1,00	584	0,91
High	First-increase-then-decrease	7	2400	239	1,00	340	0,95
High	First-increase-then-decrease	10	1200	1560	0,88	2240	0,75
High	First-increase-then-decrease	10	1400	1040	0,95	1734	0,81
High	First-increase-then-decrease	10	1600	698	0,99	1374	0,85
High	First-increase-then-decrease	10	2000	399	1,00	834	0,91
High	First-increase-then-decrease	10	2400	239	1,00	485	0,95
High	First-increase-then-decrease	15	1200	2085	0,88	3360	0,75
High	First-increase-then-decrease	15	1400	1281	0,95	2601	0,81
High	First-increase-then-decrease	15	1600	755	0,99	2061	0,85
High	First-increase-then-decrease	15	2000	399	1,00	1251	0,91
High	First-increase-then-decrease	15	2400	239	1,00	728	0,95
High	First-decrease-then-increase	3	1200	372	0,86	372	0,86
High	First-decrease-then-increase	3	1400	238	0,91	239	0,91
High	First-decrease-then-increase	3	1600	166	0,94	166	0,94
High	First-decrease-then-increase	3	2000	76	0,97	77	0,97
High	First-decrease-then-increase	3	2400	24	0,99	24	0,99
High	First-decrease-then-increase	5	1200	598	0,92	620	0,86
High	First-decrease-then-increase	5	1400	386	0,96	398	0,91
High	First-decrease-then-increase	5	1600	270	0,97	277	0,94
High	First-decrease-then-increase	5	2000	127	0,99	128	0,97
High	First-decrease-then-increase	5	2400	39	1,00	40	0,99
High	First-decrease-then-increase	7	1200	736	0,92	869	0,86
High	First-decrease-then-increase	7	1400	462	0,96	557	0,91
High	First-decrease-then-increase	7	1600	318	0,97	387	0,94
High	First-decrease-then-increase	7	2000	153	0,99	179	0,97
High	First-decrease-then-increase	7	2400	47	1,00	55	0,99
High	First-decrease-then-increase	10	1200	944	0,93	1241	0,86
High	First-decrease-then-increase	10	1400	577	0,96	795	0,91
High	First-decrease-then-increase	10	1600	390	0,97	553	0,94
High	First-decrease-then-increase	10	2000	191	0,99	255	0,97
High	First-decrease-then-increase	10	2400	59	1,00	79	0,99
High	First-decrease-then-increase	15	1200	1212	0,94	1861	0,86
High	First-decrease-then-increase	15	1400	745	0,96	1193	0,91
High	First-decrease-then-increase	15	1600	509	0,97	830	0,94
High	First-decrease-then-increase	15	2000	255	0,99	383	0,97
High	First-decrease-then-increase	15	2400	79	1,00	119	0,99
High	Fixed	3	1200	90	0,97	90	0,97
High	Fixed	3	1400	0	1,00	0	1,00
High	Fixed	3	1600	0	1,00	0	1,00
High	Fixed	3	2000	0	1,00	0	1,00

Table 18. Results of GAMS model (cont'd)

High	Fixed	3	2400	0	1,00	0	1,00
High	Fixed	5	1200	150	0,97	150	0,97
High	Fixed	5	1400	0	1,00	0	1,00
High	Fixed	5	1600	0	1,00	0	1,00
High	Fixed	5	2000	0	1,00	0	1,00
High	Fixed	5	2400	0	1,00	0	1,00
High	Fixed	7	1200	209	0,97	210	0,97
High	Fixed	7	1400	0	1,00	0	1,00
High	Fixed	7	1600	0	1,00	0	1,00
High	Fixed	7	2000	0	1,00	0	1,00
High	Fixed	7	2400	0	1,00	0	1,00
High	Fixed	10	1200	296	0,97	300	0,97
High	Fixed	10	1400	0	1,00	0	1,00
High	Fixed	10	1600	0	1,00	0	1,00
High	Fixed	10	2000	0	1,00	0	1,00
High	Fixed	10	2400	0	1,00	0	1,00
High	Fixed	15	1200	441	0,97	450	0,97
High	Fixed	15	1400	0	1,00	0	1,00
High	Fixed	15	1600	0	1,00	0	1,00
High	Fixed	15	2000	0	1,00	0	1,00
High	Fixed	15	2400	0	1,00	0	1,00
Low	Mixed	3	1200	646	0,76	646	0,76
Low	Mixed	3	1400	515	0,81	515	0,81
Low	Mixed	3	1600	405	0,85	406	0,85
Low	Mixed	3	2000	254	0,91	254	0,91
Low	Mixed	3	2400	147	0,95	147	0,95
Low	Mixed	5	1200	1017	0,84	1077	0,76
Low	Mixed	5	1400	814	0,89	858	0,81
Low	Mixed	5	1600	645	0,92	676	0,85
Low	Mixed	5	2000	412	0,95	424	0,91
Low	Mixed	5	2400	244	0,96	246	0,95
Low	Mixed	7	1200	1289	0,85	1508	0,76
Low	Mixed	7	1400	988	0,90	1202	0,81
Low	Mixed	7	1600	750	0,94	946	0,85
Low	Mixed	7	2000	468	0,97	593	0,91
Low	Mixed	7	2400	277	0,98	344	0,95
Low	Mixed	10	1200	1693	0,86	2154	0,76
Low	Mixed	10	1400	1249	0,91	1717	0,81
Low	Mixed	10	1600	906	0,94	1352	0,85
Low	Mixed	10	2000	552	0,97	847	0,91
Low	Mixed	10	2400	327	0,98	491	0,95
Low	Mixed	15	1200	2205	0,89	3231	0,76
Low	Mixed	15	1400	1603	0,92	2575	0,81
Low	Mixed	15	1600	1145	0,95	2028	0,85
Low	Mixed	15	2000	693	0,97	1271	0,91
Low	Mixed	15	2400	408	0,98	737	0,95
Low	Increasing	3	1200	940	0,65	940	0,65

Table 18. Results of GAMS model (cont’d)

Low	Increasing	3	1400	796	0,71	797	0,70
Low	Increasing	3	1600	658	0,76	659	0,76
Low	Increasing	3	2000	438	0,84	438	0,84
Low	Increasing	3	2400	272	0,90	272	0,90
Low	Increasing	5	1200	1426	0,81	1567	0,65
Low	Increasing	5	1400	1215	0,90	1328	0,70
Low	Increasing	5	1600	1015	0,95	1098	0,76
Low	Increasing	5	2000	694	0,97	730	0,84
Low	Increasing	5	2400	449	0,97	454	0,90
Low	Increasing	7	1200	1770	0,81	2194	0,65
Low	Increasing	7	1400	1398	0,90	1859	0,70
Low	Increasing	7	1600	1070	0,97	1537	0,76
Low	Increasing	7	2000	695	1,00	1022	0,84
Low	Increasing	7	2400	449	1,00	636	0,90
Low	Increasing	10	1200	2263	0,89	3134	0,65
Low	Increasing	10	1400	1671	0,94	2656	0,70
Low	Increasing	10	1600	1152	0,98	2196	0,76
Low	Increasing	10	2000	695	1,00	1460	0,84
Low	Increasing	10	2400	449	1,00	908	0,90
Low	Increasing	15	1200	2275	1,00	4701	0,65
Low	Increasing	15	1400	1670	1,00	3984	0,70
Low	Increasing	15	1600	1152	1,00	3294	0,76
Low	Increasing	15	2000	695	1,00	2190	0,84
Low	Increasing	15	2400	449	1,00	1362	0,90
Low	Decreasing	3	1200	825	0,69	825	0,69
Low	Decreasing	3	1400	683	0,75	683	0,75
Low	Decreasing	3	1600	560	0,79	560	0,79
Low	Decreasing	3	2000	372	0,86	372	0,86
Low	Decreasing	3	2400	232	0,91	232	0,91
Low	Decreasing	5	1200	1375	0,69	1375	0,69
Low	Decreasing	5	1400	1138	0,75	1138	0,75
Low	Decreasing	5	1600	933	0,79	933	0,79
Low	Decreasing	5	2000	621	0,86	621	0,86
Low	Decreasing	5	2400	386	0,91	386	0,91
Low	Decreasing	7	1200	1925	0,69	1925	0,69
Low	Decreasing	7	1400	1593	0,75	1593	0,75
Low	Decreasing	7	1600	1307	0,79	1307	0,79
Low	Decreasing	7	2000	869	0,86	869	0,86
Low	Decreasing	7	2400	540	0,91	540	0,91
Low	Decreasing	10	1200	2751	0,69	2751	0,69
Low	Decreasing	10	1400	2276	0,75	2276	0,75
Low	Decreasing	10	1600	1867	0,79	1867	0,79
Low	Decreasing	10	2000	1241	0,86	1241	0,86
Low	Decreasing	10	2400	772	0,91	772	0,91
Low	Decreasing	15	1200	4126	0,69	4126	0,69
Low	Decreasing	15	1400	3414	0,75	3414	0,75
Low	Decreasing	15	1600	2800	0,79	2800	0,79

Table 18. Results of GAMS model (cont'd)

Low	Decreasing	15	2000	1862	0,86	1862	0,86
Low	Decreasing	15	2400	1158	0,91	1158	0,91
Low	First-increase-then-decrease	3	1200	670	0,75	671	0,75
Low	First-increase-then-decrease	3	1400	532	0,80	533	0,80
Low	First-increase-then-decrease	3	1600	418	0,85	418	0,85
Low	First-increase-then-decrease	3	2000	257	0,91	257	0,90
Low	First-increase-then-decrease	3	2400	149	0,95	149	0,94
Low	First-increase-then-decrease	5	1200	1028	0,87	1117	0,75
Low	First-increase-then-decrease	5	1400	821	0,89	888	0,80
Low	First-increase-then-decrease	5	1600	650	0,91	697	0,85
Low	First-increase-then-decrease	5	2000	411	0,94	429	0,90
Low	First-increase-then-decrease	5	2400	246	0,97	248	0,94
Low	First-increase-then-decrease	7	1200	1233	0,89	1564	0,75
Low	First-increase-then-decrease	7	1400	919	0,95	1243	0,80
Low	First-increase-then-decrease	7	1600	678	0,98	976	0,85
Low	First-increase-then-decrease	7	2000	410	1,00	601	0,90
Low	First-increase-then-decrease	7	2400	246	1,00	347	0,94
Low	First-increase-then-decrease	10	1200	1543	0,89	2234	0,75
Low	First-increase-then-decrease	10	1400	1068	0,95	1776	0,80
Low	First-increase-then-decrease	10	1600	719	0,98	1394	0,85
Low	First-increase-then-decrease	10	2000	410	1,00	858	0,90
Low	First-increase-then-decrease	10	2400	246	1,00	496	0,94
Low	First-increase-then-decrease	15	1200	2057	0,89	3351	0,75
Low	First-increase-then-decrease	15	1400	1315	0,95	2664	0,80
Low	First-increase-then-decrease	15	1600	788	0,98	2091	0,85
Low	First-increase-then-decrease	15	2000	410	1,00	1287	0,90
Low	First-increase-then-decrease	15	2400	246	1,00	246	0,98
Low	First-decrease-then-increase	3	1200	365	0,87	365	0,86
Low	First-decrease-then-increase	3	1400	250	0,91	250	0,91
Low	First-decrease-then-increase	3	1600	167	0,94	167	0,94
Low	First-decrease-then-increase	3	2000	82	0,97	83	0,97
Low	First-decrease-then-increase	3	2400	28	0,99	28	0,99
Low	First-decrease-then-increase	5	1200	588	0,93	609	0,86
Low	First-decrease-then-increase	5	1400	406	0,96	417	0,91
Low	First-decrease-then-increase	5	1600	273	0,97	278	0,94
Low	First-decrease-then-increase	5	2000	137	0,98	138	0,97
Low	First-decrease-then-increase	5	2400	46	0,99	46	0,99
Low	First-decrease-then-increase	7	1200	724	0,93	853	0,86
Low	First-decrease-then-increase	7	1400	482	0,96	584	0,91
Low	First-decrease-then-increase	7	1600	319	0,97	390	0,94
Low	First-decrease-then-increase	7	2000	164	0,99	193	0,97
Low	First-decrease-then-increase	7	2400	55	0,99	65	0,99
Low	First-decrease-then-increase	10	1200	922	0,93	1219	0,86
Low	First-decrease-then-increase	10	1400	598	0,96	835	0,91
Low	First-decrease-then-increase	10	1600	389	0,97	557	0,94
Low	First-decrease-then-increase	10	2000	204	0,99	275	0,97
Low	First-decrease-then-increase	10	2400	69	0,99	92	0,99

Table 18. Results of GAMS model (cont'd)

Low	First-decrease-then-increase	15	1200	1168	0,95	1828	0,86
Low	First-decrease-then-increase	15	1400	768	0,96	1252	0,91
Low	First-decrease-then-increase	15	1600	504	0,97	835	0,94
Low	First-decrease-then-increase	15	2000	271	0,99	413	0,97
Low	First-decrease-then-increase	15	2400	92	0,99	139	0,99
Low	Fixed	3	1200	0	1,00	0	1,00
Low	Fixed	3	1400	0	1,00	0	1,00
Low	Fixed	3	1600	0	1,00	0	1,00
Low	Fixed	3	2000	0	1,00	0	1,00
Low	Fixed	3	2400	0	1,00	0	1,00
Low	Fixed	5	1200	0	1,00	0	1,00
Low	Fixed	5	1400	0	1,00	0	1,00
Low	Fixed	5	1600	0	1,00	0	1,00
Low	Fixed	5	2000	0	1,00	0	1,00
Low	Fixed	5	2400	0	1,00	0	1,00
Low	Fixed	7	1200	0	1,00	0	1,00
Low	Fixed	7	1400	0	1,00	0	1,00
Low	Fixed	7	1600	0	1,00	0	1,00
Low	Fixed	7	2000	0	1,00	0	1,00
Low	Fixed	7	2400	0	1,00	0	1,00
Low	Fixed	10	1200	0	1,00	0	1,00
Low	Fixed	10	1400	0	1,00	0	1,00
Low	Fixed	10	1600	0	1,00	0	1,00
Low	Fixed	10	2000	0	1,00	0	1,00
Low	Fixed	10	2400	0	1,00	0	1,00
Low	Fixed	15	1200	0	1,00	0	1,00
Low	Fixed	15	1400	0	1,00	0	1,00
Low	Fixed	15	1600	0	1,00	0	1,00
Low	Fixed	15	2000	0	1,00	0	1,00
Low	Fixed	15	2400	0	1,00	0	1,00

APPENDIX D

FORMULATION AND RESULTS OF DUNCAN'S TESTS

Table 19. Formulation of Duncan's Tests

	1	**2**	**3**	**4**	**5**	**6**	**total**	**average**	**Source of Variation**	**Sum of Squares (SS)**	**Degrees of freedom**	**Mean Square (MS)**
VBA	vba1	vba2	vba3	vba4	vba5	vba6	sum$_i$ (vba$_i$)	sum$_i$ (vba$_i$)/6	VBA-ATO	SS$_{Treat.}$	1	MS$_{Treat.}$
ATO	ato1	ato2	ato3	ato4	ato5	ato6	sum$_i$ (ato$_i$)	sum$_i$ (ato$_i$)/6	Error	SS$_{Error}$	10	MS$_{Error}$
									Total	SS$_{Total}$	11	

Standard Error of Each Mean (S)	**Least Significant Range for 95% CI (R2)**	**Difference Between Means of VBA and ATO (D)**	**DUNCAN result for 95% CI**
POWER(MS$_{Error}$/6;1/2)	3,15*S	[sum$_i$ (ato$_i$)-sum$_i$ (vba$_i$)]/6	IF(D>R2;"significant"; "not")

Table 20. Results of Duncan's Tests

			TOTAL COST				
Distribution	**Penalty Cost**	**Capacity**	**MS$_{Error}$**	**S**	**R2**	**D**	**DUNCAN Result**
Increasing	5	1200	76745	113	356	362	significant
Increasing	7	1200	98567	128	404	435	significant
Increasing	10	1200	124568	144	454	478	significant
Increasing	15	1200	252463	205	646	762	significant
Increasing	5	1400	78891	115	361	364	significant
Increasing	7	1400	96578	127	400	425	significant
Increasing	10	1400	111456	136	429	512	significant
Increasing	15	1400	247221	203	639	679	significant

Table 20. Results of Duncan’s Tests (cont’d)

Increasing	5	1600	76677	113	356	336	not
Increasing	7	1600	91456	123	389	392	significant
Increasing	10	1600	98745	128	404	412	significant
Increasing	15	1600	212223	188	592	601	significant
Increasing	5	2000	53649	95	298	278	not
Increasing	7	2000	72544	110	346	389	significant
Increasing	10	2000	85478	119	376	405	significant
Increasing	15	2000	198114	182	572	614	significant
Increasing	5	2400	27397	68	213	156	not
Increasing	7	2400	36475	78	246	231	not
Increasing	10	2400	48125	90	282	257	not
Increasing	15	2400	78457	114	360	380	significant
First-Increase-Then-Decrease	5	1200	86814	120	379	354	not
First-Increase-Then-Decrease	7	1200	98145	128	403	409	significant
First-Increase-Then-Decrease	10	1200	135465	150	473	487	significant
First-Increase-Then-Decrease	15	1200	257145	207	652	654	significant
First-Increase-Then-Decrease	5	1400	78115	114	359	342	not
First-Increase-Then-Decrease	7	1400	96477	127	399	403	significant
First-Increase-Then-Decrease	10	1400	111446	136	429	485	significant
First-Increase-Then-Decrease	15	1400	254123	206	648	653	significant
First-Increase-Then-Decrease	5	1600	76677	113	356	335	not
First-Increase-Then-Decrease	7	1600	91446	123	389	400	significant
First-Increase-Then-Decrease	10	1600	99257	129	405	467	significant
First-Increase-Then-Decrease	15	1600	202547	184	579	652	significant
First-Increase-Then-Decrease	5	2000	43649	85	269	258	not
First-Increase-Then-Decrease	7	2000	58984	99	312	304	not
First-Increase-Then-Decrease	10	2000	64751	104	327	339	significant
First-Increase-Then-Decrease	15	2000	197114	181	571	589	significant
First-Increase-Then-Decrease	5	2400	27397	68	213	199	not
First-Increase-Then-Decrease	7	2400	36474	78	246	218	not
First-Increase-Then-Decrease	10	2400	45758	87	275	255	not
First-Increase-Then-Decrease	15	2400	77447	114	358	416	significant
First-Decrease-Then-Increase	5	1200	65489	104	329	347	significant
First-Decrease-Then-Increase	7	1200	74261	111	350	379	significant
First-Decrease-Then-Increase	10	1200	124563	144	454	469	significant
First-Decrease-Then-Increase	15	1200	247569	203	640	688	significant
First-Decrease-Then-Increase	5	1400	63289	103	324	321	not
First-Decrease-Then-Increase	7	1400	72158	110	345	365	significant
First-Decrease-Then-Increase	10	1400	112639	137	432	458	significant
First-Decrease-Then-Increase	15	1400	218988	191	602	652	significant
First-Decrease-Then-Increase	5	1600	59784	100	314	305	not
First-Decrease-Then-Increase	7	1600	69124	107	338	324	not
First-Decrease-Then-Increase	10	1600	99548	129	406	428	significant
First-Decrease-Then-Increase	15	1600	201363	183	577	604	significant
First-Decrease-Then-Increase	5	2000	28546	69	217	201	not
First-Decrease-Then-Increase	7	2000	34787	76	240	215	not
First-Decrease-Then-Increase	10	2000	54967	96	301	278	not

Table 20. Results of Duncan’s Tests (cont’d)

First-Decrease-Then-Increase	15	2000	105462	133	418	359	not
First-Decrease-Then-Increase	5	2400	17566	54	170	165	not
First-Decrease-Then-Increase	7	2400	24541	64	201	186	not
First-Decrease-Then-Increase	10	2400	38978	81	254	216	not
First-Decrease-Then-Increase	15	2400	59898	100	315	304	not

			FILL RATE				
Distribution	**Penalty Cost**	**Capacity**	**MS$_{Error}$**	**S**	**R2**	**D**	**DUNCAN Result**
Increasing	5	1200	0,04	0,08	0,26	0,26	significant
Increasing	7	1200	0,02	0,06	0,18	0,24	significant
Increasing	10	1200	0,02	0,06	0,18	0,22	significant
Increasing	15	1200	0,01	0,04	0,13	0,17	significant
Increasing	5	1400	0,02	0,06	0,18	0,20	significant
Increasing	7	1400	0,01	0,04	0,13	0,19	significant
Increasing	10	1400	0,01	0,04	0,13	0,18	significant
Increasing	15	1400	0,01	0,04	0,12	0,17	significant
Increasing	5	1600	0,02	0,06	0,18	0,19	significant
Increasing	7	1600	0,02	0,06	0,18	0,19	significant
Increasing	10	1600	0,01	0,05	0,15	0,17	significant
Increasing	15	1600	0,01	0,04	0,14	0,16	significant
Increasing	5	2000	0,01	0,05	0,15	0,19	significant
Increasing	7	2000	0,01	0,05	0,15	0,17	significant
Increasing	10	2000	0,00	0,03	0,08	0,12	significant
Increasing	15	2000	0,00	0,02	0,06	0,09	significant
Increasing	5	2400	0,00	0,03	0,08	0,09	significant
Increasing	7	2400	0,00	0,03	0,08	0,09	significant
Increasing	10	2400	0,00	0,02	0,07	0,08	significant
Increasing	15	2400	0,00	0,01	0,04	0,07	significant
First-Increase-Then-Decrease	5	1200	0,02	0,06	0,18	0,23	significant
First-Increase-Then-Decrease	7	1200	0,02	0,06	0,18	0,21	significant
First-Increase-Then-Decrease	10	1200	0,01	0,05	0,15	0,20	significant
First-Increase-Then-Decrease	15	1200	0,01	0,04	0,13	0,18	significant
First-Increase-Then-Decrease	5	1400	0,02	0,06	0,18	0,17	not
First-Increase-Then-Decrease	7	1400	0,01	0,04	0,13	0,16	significant
First-Increase-Then-Decrease	10	1400	0,01	0,04	0,13	0,14	significant
First-Increase-Then-Decrease	15	1400	0,01	0,04	0,12	0,13	significant
First-Increase-Then-Decrease	5	1600	0,02	0,06	0,18	0,17	not
First-Increase-Then-Decrease	7	1600	0,01	0,05	0,15	0,16	significant
First-Increase-Then-Decrease	10	1600	0,01	0,04	0,14	0,17	significant
First-Increase-Then-Decrease	15	1600	0,01	0,04	0,13	0,16	significant
First-Increase-Then-Decrease	5	2000	0,01	0,05	0,15	0,14	not
First-Increase-Then-Decrease	7	2000	0,01	0,04	0,13	0,12	not
First-Increase-Then-Decrease	10	2000	0,00	0,03	0,08	0,07	not
First-Increase-Then-Decrease	15	2000	0,01	0,03	0,09	0,12	significant
First-Increase-Then-Decrease	5	2400	0,00	0,03	0,08	0,07	not
First-Increase-Then-Decrease	7	2400	0,00	0,03	0,08	0,05	not

Table 20. Results of Duncan’s Tests (cont’d)

First-Increase-Then-Decrease	10	2400	0,00	0,02	0,07	0,02	not
First-Increase-Then-Decrease	15	2400	0,00	0,01	0,04	0,01	not
First-Decrease-Then-Increase	5	1200	0,03	0,07	0,22	0,24	significant
First-Decrease-Then-Increase	7	1200	0,02	0,06	0,18	0,22	significant
First-Decrease-Then-Increase	10	1200	0,02	0,06	0,18	0,19	significant
First-Decrease-Then-Increase	15	1200	0,01	0,04	0,13	0,17	significant
First-Decrease-Then-Increase	5	1400	0,03	0,07	0,22	0,23	significant
First-Decrease-Then-Increase	7	1400	0,02	0,06	0,18	0,20	significant
First-Decrease-Then-Increase	10	1400	0,01	0,04	0,13	0,17	significant
First-Decrease-Then-Increase	15	1400	0,01	0,04	0,12	0,14	significant
First-Decrease-Then-Increase	5	1600	0,02	0,06	0,18	0,19	significant
First-Decrease-Then-Increase	7	1600	0,01	0,05	0,15	0,16	significant
First-Decrease-Then-Increase	10	1600	0,01	0,04	0,14	0,15	significant
First-Decrease-Then-Increase	15	1600	0,01	0,04	0,13	0,14	significant
First-Decrease-Then-Increase	5	2000	0,01	0,05	0,15	0,14	not
First-Decrease-Then-Increase	7	2000	0,01	0,04	0,13	0,10	not
First-Decrease-Then-Increase	10	2000	0,00	0,03	0,08	0,06	not
First-Decrease-Then-Increase	15	2000	0,00	0,02	0,06	0,03	not
First-Decrease-Then-Increase	5	2400	0,00	0,03	0,08	0,07	not
First-Decrease-Then-Increase	7	2400	0,00	0,03	0,08	0,06	not
First-Decrease-Then-Increase	10	2400	0,00	0,02	0,07	0,04	not
First-Decrease-Then-Increase	15	2400	0,00	0,01	0,04	0,02	not