

A DECISION SUPPORT SYSTEM FOR INTEGRATED OPTIMIZATION OF
PRODUCTION, TRANSPORTATION AND PRICING DECISIONS
IN A SUPPLY CHAIN

MURAT UMUT İZER

BOĞAZİÇİ UNIVERSITY

2016

A DECISION SUPPORT SYSTEM FOR INTEGRATED OPTIMIZATION OF
PRODUCTION, TRANSPORTATION AND PRICING DECISIONS
IN A SUPPLY CHAIN

Thesis submitted to the
Institute for Graduate Studies in Social Sciences
in partial fulfillment of the requirements for the degree of

Master of Arts
in
Management Information Systems

by
Murat Umut İzer

Boğaziçi University

2016

DECLARATION OF ORIGINALITY

I, Murat Umut İzer, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature.....

Date29.02.2016.....

ABSTRACT

A Decision Support System for Integrated Optimization of Production, Transportation and Pricing Decisions in a Supply Chain

In supply chains, the decisions regarding production, transportation and pricing are often handled separately. In this thesis study, a nonlinear programming model (NLP) is generated to optimize the production, transportation, and pricing decisions in a supply chain where substitute products are produced in multiple factories and sold at several markets. Under the cooperative competition of substitute products, these decisions are given centrally to maximize the total profit. Demands for the substitute products are realized as functions of their prices where the market shares are expressed as market share attraction models from the marketing literature. The NLP model is solved with different parameter settings and sensitivity analysis on the input parameters is made to provide managerial insights. Finally, a decision support system (DSS) is developed to provide an efficient, effective and flexible decision making environment. The DSS includes a relational database for input and output data, a model base that includes the generated NLP model and a graphical user interface that provides interaction between the user, the database, and the solver for the NLP model.

ÖZET

Üretim, Taşıma ve Fiyatlandırma Kararlarının Bütünleşik Olarak Eniyilendiği Bir Tedarik Zinciri İçin Bir Karar Destek Sistemi

Tedarik zincirlerinde üretim, taşıma ve fiyatlandırma ile ilgili kararlar genellikle ayrı ayrı ele alınmaktadır. Bu tez çalışmasında ikame ürünlerin birden çok fabrikada üretildiği ve çeşitli pazarlarda satıldığı bir tedarik zincirinde üretim, taşıma ve fiyatlandırma kararlarını eniyilemeyi amaçlayan doğrusal olmayan bir eniyileme modeli geliştirilmiştir. İkame ürünlerin rekabetçi işbirliği altında bu kararlar merkezi olarak toplam karı ençoklayacak şekilde verilmektedir. İkame ürünlerin taleplerinin kendi fiyatlarının birer fonksiyonu olarak oluştuğu bu ortamda pazar payları ise pazarlama literatüründe yer alan pazar payı çekim modelleri ile açıklanmaktadır. Doğrusal olmayan programlama modeli farklı parametre grupları kullanılarak çözülmüştür ve girdi parametreleri üzerinde yönetsel içgörüler sağlayan duyarlılık analizi yapılmıştır. Son olarak etkili, etkin ve esnek bir karar verme ortamı sağlamak için bir karar destek sistemi (KDS) geliştirilmiştir. Geliştirilen KDS girdi ve çıktı verileri için tasarlanan ilişkisel bir veritabanı, doğrusal olmayan programlama modelini içeren bir model tabanı ve kullanıcı, veritabanı, doğrusal olmayan programlama modelinin çözücüsü arasında iletişim sağlayan bir kullanıcı arayüzü içermektedir.

ACKNOWLEDGEMENTS

First of all I would like to honor my dear family for giving me the urge and potential to complete my thesis. Special thanks to my thesis advisor Prof. Aslı Sencer for opening new horizons and guiding me all the way from the start to the end. I would like to thank Assoc. Prof. Bertan Badur and Prof. Peter Schuur for their insights and valuable contribution on the conceptualization. It was a pleasure to work with such qualified three academicians. Also I would like to thank my colleague Canan Ağlan (PhD) for helping me writing down the GAMS model. Lastly, I would like to thank Merih Küpeli for his support.

This M.A. thesis has been supported by Bogazici University Research Fund Grant Number 6949.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION	1
CHAPTER 2: LITERATURE REVIEW	4
2.1 Integrated models for production and transportation decisions	4
2.2 Dynamic pricing models	7
2.3 Price-demand relations in dynamic pricing theory	10
2.4 Market share models in marketing theory	13
2.5 Decision support systems for dynamic pricing	15
2.6 Contribution to the literature	16
CHAPTER 3: DEVELOPMENT OF AN INTEGRATED MODEL FOR PRODUCTION, TRANSPORTATION AND PRICING DECISIONS.....	18
3.1 Model formulation	18
3.2 Additional constraints for the solution of the model in GAMS	21
3.3 Solution of the model in single period	22
3.4 Solution of the model in multiple periods.....	38
CHAPTER 4: DEVELOPMENT OF A DECISION SUPPORT SYSTEM FOR THE INTEGRATED MODEL.....	51
4.1 DSS framework and the information flow	51
4.2 DSS database.....	53
4.3 DSS development.....	59
CHAPTER 5: CONCLUSION.....	79
APPENDIX A: GAMS MODEL	82
APPENDIX B: PARAMETER TABLES	85
APPENDIX C: VARIABLE TABLES	87
APPENDIX D: SOURCE CODE OF VIEW OPTIMAL PLAN FORM	88
REFERENCES.....	97

LIST OF TABLES

Table 1. Studies on Production, Transportation, Pricing and DSS in the Literature ...	5
Table 2. Input Parameters of the Single Period Problem	22
Table 3. Sensitivity Analysis 1: Changing the Attraction Sensitivity of Product Type 1	23
Table 4. Changing the Attraction Sensitivity of Product Type 1 in Capacitated Setting	31
Table 5. Sensitivity Analysis 2: Changing the Unit Production Cost of Product Type 1	34
Table 6. Input Parameters of the Multiple Period Problems	39
Table 7. Optimal Solution with Respect to Relative Capacities in Multiple Periods	40
Table 8. Testing the Models with Ten Different Parameter Sets in Multiple Periods	49

LIST OF FIGURES

Figure 1. Graphs of demand functions	11
Figure 2. Market share-attraction sensitivity graph for Sensitivity Analysis 1.....	24
Figure 3. Optimal total profit as a function of attraction sensitivity – the uncapacitated setting and the capacitated setting	32
Figure 4. Market share-unit production cost graph for Sensitivity Analysis 2	34
Figure 5. Optimal total profit as a function of attraction sensitivity	36
Figure 6. Optimal total profit as a function of unit production cost	37
Figure 7. Network representation with low unit stockout costs ($\beta=75\%$)	43
Figure 8. Network representation with high unit stockout costs ($\beta=75\%$)	44
Figure 9. Network representation with low unit stockout costs ($\beta=125\%$)	46
Figure 10. Schematic view of the DSS for the integrated model.....	52
Figure 11. Information flow in the DSS	53
Figure 12. The relational database diagram for the output parameters of the DSS ...	57
Figure 13. The relational database diagram for the input parameters of the DSS	58
Figure 14. PTP-DSS welcome page.....	61
Figure 15. View parameters form	62
Figure 16. List of factories screen and list of markets screen.....	63
Figure 17. List of product types screen and list of time periods screen.....	63
Figure 18. Attraction sensitivity and unit stockout cost table screen.....	64
Figure 19. Solver message box	65
Figure 20. Problem information screen.....	66
Figure 21. Optimal transportation quantity table screen.....	68
Figure 22. View optimal plan form.....	69
Figure 23. Production quantity and inventory level graph screen	70
Figure 24. Optimal production quantity sample graph	70
Figure 25. Optimal inventory level sample graph.....	71
Figure 26. Unit price and market share graph screen.....	72
Figure 27. Optimal unit price sample graph.....	72
Figure 28. Optimal market share sample graph	73
Figure 29. Transportation quantity graphs screen.....	73
Figure 30. Optimal transportation quantity sample graph	74
Figure 31. View optimal objective values screen	74
Figure 32. Warning message box.....	76
Figure 33. Edit parameters for what-if analysis form	77
Figure 34. View what-if optimal plan form	78

LIST OF ABBREVIATIONS

BI	Business Intelligence
DSS	Decision Support System
EOQ	Economic Order Quantity
GAMS	General Algebraic Modeling System
GUI	Graphical User Interface
MCI	Multiplicative Competitive Interaction
MNL	Multinomial Logit
MSA	Market Share Attraction
MSAGL	Microsoft Automatic Graph Layout
NLP	Nonlinear Programming
NP-Hard	Non-deterministic Polynomial-time Hard
OLAP	Online Analytical Processing
PTP	Integrated Production, Transportation and Pricing Problem
SQL	Structured Query Language
TP	Transportation Problem
ZIO	Zero Inventory Ordering

CHAPTER 1

INTRODUCTION

Decisions of how much to produce a product, how much quantity of the product will be transported and what will be the selling price of a product is given separately in the supply chain literature. These strategic decisions that affect both the success and the profitability of the company and the product can be given in long terms, in addition centralization of these decisions require an effective information system infrastructure. In a supply chain where decisions are given centrally, problems like stockout and excess demand are very common. In order to prevent these problems, systems that enable optimization of production, transportation, and pricing decisions with an integrated approach is required in supply chains.

Nowadays usage of information systems and information technologies in supply chain management become important day by day. Information systems are tools that play an important role on enabling centralization of decisions at the strategic and tactical levels. Information systems allow continuous tracing, immediate response to problems and full control throughout the process in supply chains from raw material entry to the product offered to the customers including support services.

Rapid development of information technologies also revealed the concept of dynamic pricing. Prices may change in short periods, even instantaneously, instead of changing in long periods in the past. Dynamic pricing allows the estimation of demand for a product easier and more accurately, determination of costs more rapidly, and adjustment to changing market conditions immediately.

Dynamic pricing is also known as intertemporal pricing. Elmaghraby and Keskinocak (2003) determined the factors that arisen from the concept of dynamic pricing as more accessible demand data, easier changing of prices by new technologies and decision support systems suitable for analyzing demand data and dynamic pricing. In short life-cycle products inventory replenishment is not possible whereas for other products the seller can replenish inventory periodically. If durable goods are considered, the seller changes prices while customers cause variance on demand regarding their price information of the product. When nondurable goods are considered, customers don't have the chance to gain price information of the product as the sale period is too short. Myopic customers tend to buy the product as soon as the price of the product is lower than their estimation of the price of the product whereas strategic customers give their purchasing decisions regarding the changes in prices in the future. With the common usage of online price search engines, the customers can trace changes in prices periodically, which cause increased strategic behavior of the customers.

As an example, when a firm producing and selling printers is considered, products of same type can be produced in multiple factories. The cost of production in these factories will vary depending on some factors such as the labor force of the country in which the factories are built, raw material costs etc. But when products are put on different markets for sale, their prices can differentiate without examining in which factory they have been produced.

Decision making in supply chains is a complex process. Biswas and Narahari (2004) determined the causes of this phenomenon as: (a) large scale of supply chain networks, (b) hierarchic structure of decisions, (c) existence of random inputs and processes, (d) dynamic interaction between the supply chain elements.

Kärkkäinen, Elfvengren, Torkkeli, and Tuominen (2001) defined Decision Support Systems as information systems that support semi-structured and non-structured decisions of business executives. Mansouri, Gallear, and Askariazad (2012) concluded that a DSS should consist of four main components: (a) database, (b) model base, (c) information base, (d) graphical user interface (GUI). The database stores the data, model base and information base mediates the storage of model and information batches, and the GUI provides the users to interact with the database, model base and the information base.

The originality of this thesis study is the addition of market share concept and DSS into the integrated model. A mathematical optimization model that handles production, transportation and pricing decisions integrated together will be developed in this study. Mathematical models that are developed for the transportation model are commonly used for lots of logistics and supply chain problems in real life. In the standard transportation problem, demand occurred at the markets are considered to be constant. The objective function is to minimize the total transportation costs or to maximize the total profit under certain demand and production capacity constraints. In this study, market share models that have a wide coverage of marketing literature will be used by handling product demand at different markets as a function of determined price of the product. The built integrated model will construct the model base of the DSS that will be developed later.

CHAPTER 2

LITERATURE REVIEW

In this section, models that handle production, transportation and pricing decisions together are analyzed and grouped initially. Then, production-transportation integration is reviewed. Later, dynamic pricing concept is reviewed and dynamic pricing models are investigated. After that, price-demand relations in dynamic pricing theory are reviewed. Then, market share models are analyzed with special emphasis on market share attraction models. Lastly, DSSs that are built for integrated production, transportation and pricing decisions are researched.

As this thesis is a part of a scientific research project, studies on both cooperative and noncooperative cases are reviewed but only the cooperative studies are included in the literature review. Among 140 studies, 40 related studies are analyzed and grouped according to the existence of production and inventory decisions, transportation decisions, pricing decisions and DSS. It is seen that there are lots of studies on the integrated models of production and transportation in the literature. On the other hand, few models that handle production, transportation and pricing decisions integrated together exist in the literature. These studies are summarized in Table 1.

2.1 Integrated models for production and transportation decisions

The standard TP model is insufficient in conditions where the sellers' market power is high and the seller is able to affect demands by adjusting prices. As these conditions are considered, the standard TP becomes NP-Hard because of nonlinearity and the complexity of the problem due to additional variables and constraints.

Table 1. Studies on Production, Transportation, Pricing and DSS in the Literature

No.	Author(s)	Production/ Inventory	Transportation	Pricing	DSS
1	Shetty (1959)	+	+	-	-
2	Sharp et al. (1970)	+	+	-	-
3	Leblanc and Cooper (1974)	+	+	-	-
4	Ishii et al. (1988)	+	+	-	-
5	Youssef and Mahmoud (1996)	+	+	-	-
6	Gupta and Maranas (2003)	+	+	-	-
7	Chen (2004) (review paper)	+	+	-	-
8	Gen and Syarif (2005)	+	+	-	-
9	Park (2005)	+	+	-	-
10	Ekşioğlu et al. (2006)	+	+	-	-
11	Gnanendran and Chien (2006)	-	+	+	-
12	Whitin (1955)	+	-	+	-
13	Wagner and Whitin (1958)	+	-	+	-
14	Assunção and Meyer (1993)	+	-	+	-
15	Ardalan (1994)	+	-	+	-
16	Gilbert (2000)	+	-	+	-
17	Elmaghraby and Keskinocak (2003)	+	-	+	+
18	Bitran and Caldentey (2003)	+	-	+	-
19	Chen and Simchi-Levi (2004)	+	-	+	-
20	Biller et al. (2005)	+	-	+	-
21	Yano and Gilbert (2005)	+	-	+	-
22	Deng and Yano (2006)	+	-	+	-
23	Adida and Perakis (2006)	+	-	+	-
24	Tim-Huh and Janakiraman (2008)	+	-	+	-
25	Song et al. (2009)	+	-	+	-
26	Chen and Simchi-Levi (2010)	+	-	+	-
27	Sharkey (2011)	+	-	+	-
28	Gümüş et al. (2011)	+	-	+	-
29	Burwell et al. (1997)	+	+	+	-
30	Chan et al. (2004)	+	+	+	-
31	Ahn et al. (2007)	+	+	+	-
32	Pujari et al. (2008)	+	+	+	-
33	Gümüş and Kaminsky (2010)	+	+	+	-
34	Lambin (1970)	-	-	+	-
35	Weiss (1968)	-	-	+	-
36	Kotler (1971)	-	-	+	-
37	Naert and Bultez (1973)	-	-	+	-
38	Bell et al. (1975)	-	-	+	-
39	Cooper and Nakanishi (1988)	-	-	+	+
40	Keskinocak et al. (2001)	+	-	+	+

In one of the earliest studies, Shetty (1959) considers production and transportation decisions together in a model to give a solution to the generalized TP with nonlinear costs, where it is required to find the amount of resource to be supplied from each source to each market, to minimize the total transportation and production costs. Sharp, Snyder, and Greene (1970) develop a decomposition algorithm to solve the multifacility production-transportation problem with nonlinear production costs. Leblanc and Cooper (1974) consider a transportation-production problem with increasing marginal production costs and linear shipping costs, which turns into a convex programming problem. Ishii, Takahashi, and Muramatsu (1988) handle the problem of how to determine economic levels for the base stock and lead times for production and transportation in integrated production, inventory and distribution systems. Youssef and Mahmoud (1996) work on the uncapacitated production-distribution problem under concave cost function. They state that when production economies of scale are introduced, the production-distribution decisions in a firm should be operated simultaneously. Gupta and Maranas (2003) work on incorporating demand uncertainty in midterm planning of multisite supply chains with a stochastic programming based approach.

Chen (2004) is an inspiring study where the aim of coordination of production and transportation functions is determined as reaching optimal operational performance. In the study, the models are classified based on decision levels, the level of production-transportation coordination and problem parameters and defined five different model classes. When problems classified as general tactical production-distribution problems are analyzed, it is seen that a general model including dynamic demand, one period or multiple periods, one producer, multiple customers and products is developed. Thus; production, inventory and transportation

functions are gathered together in a single integrated model which is adoptable to special cases. Chen also denotes that a more general model should address multiple products and a nonlinear transportation cost structure which is common in practice.

Yet there are several recent studies where production and transportation decisions are handled together. Gen and Syarif (2005) develop a hybrid genetic algorithm for the multi-time period production-distribution planning problem, in order to ensure that products are produced and distributed at the right quantities, to the right customers, and at the right time, in order to minimize system wide costs while satisfying all demands required. Park (2005) builds an integrated approach for production and distribution planning in supply chain management, where the objective is to maximize the total net profit in a multi-plant, multi-retailer, multi-item, and multi-period logistic environment. Ekşioğlu, Romeijn, and Pardalos (2006) consider an integrated production and transportation planning problem in a two-stage supply chain which consists of a number of facilities, and a number of retailers in a deterministic demand setting, without any production or transportation capacity constraints.

2.2 Dynamic pricing models

In the previous section, production-transportation integration is analyzed whereas pricing decisions are not added into these models. Gnanendran and Chien (2006) is the only study in the literature that integrates transportation-pricing decisions without including production decisions. They consider the condition where demand is a function of prices set by the decision maker, and they develop a new model that aims to optimize prices and transportation quantities simultaneously. Price variation is provided by upper and lower bounds on the selling price.

The production-pricing integration has been researched for more than fifty years. Whitin (1955) can be considered as the first study to add pricing decisions into the inventory problem, setting selling price and order quantity at the same time. The model has deterministic demand which is a linear function of price, and the objective is set to maximize the expected profit. Wagner and Whitin (1958) is the first to incorporate the pricing decision into the economic lot sizing model without capacity constraint based on the zero-inventory-ordering (ZIO) property where no order is placed when the inventory level is positive.

Chan, Shen, Simchi-Levi, and Swann (2004) give a comprehensive review on supply chain integration and dynamic pricing problems. They classify the literature based on problem characteristics or researchers' assumptions by analyzing the current models, as below:

- Period length (single period, multiple period, infinite horizon)
- Price (static, dynamic)
- Demand type (deterministic, stochastic)
- Demand functional form (linear, additive, multiplicative, exponential, Poisson)
- Demand input parameters (price, time, inventory, advertisements, promotions, sales, product characteristics)
- Sales (backlogging, lost sales)
- Restocking (yes, no)
- Production set-up cost (yes, no)
- Capacity limits (yes, no)
- Products (single, multiple)

Yet there is lots of research done on production and pricing integration. Assunção and Meyer (1993) explore the rational effect of price variation on sales and consumption in markets where uncertainty exists about the future price of a product among consumers. Ardalan (1994) analyzes the change in price and the resulting change in demand and their effect on inventory policy and total profit simultaneously to determine retailer's optimal price and optimal ordering policies. An EOQ-based model is developed in this study, where the net present values of the pricing policies are evaluated. Burwell, Dave, Fitzpatrick, and Roy (1997) incorporate quantity and freight discounts in inventory decision making, also demand is considered to be price-dependent. They develop an algorithm to determine the optimal lot size and selling price for a class of demand functions, including constant price-elasticity and linear demand. Gilbert (2000) addresses the problem of jointly determining prices and production schedules for a set of items that are produced on the same line under several assumptions such as negligible production setup costs and seasonal but price dependent demand. Bitran and Caldentey (2003) analyze dynamic pricing policies and their relation to revenue management. Chen and Simchi-Levi (2004) analyze a finite horizon, single product, periodic review model in which pricing and production/inventory decisions are made simultaneously. Demands in different periods are random variables that are independent of each other and their distributions depend on the product price. Yano and Gilbert (2005) provide a comprehensive review of analytical models on simultaneous pricing decisions which influences demands and production decisions, focusing on models in which external demand is price-sensitive.

In the more recent studies, Ahn, Gümüş, and Kaminsky (2007) analyze pricing and manufacturing decisions when demand is a function of prices in multiple

periods with the assumption that the consumers' purchase decisions are made with respect to price, not only in the current period but also in the past and future periods. Pujari, Day, Huq, and Hale (2008) construct a framework for an integrated distribution system optimization model in a multi-plant, multi-product, multi-customer supply chain with deterministic demands. A mixed-binary integer programming model with maximization objective, including capacity and demand constraints is proposed to integrate the capacitated location, production, and distribution functions within a supply chain. Tim-Huh and Janakiraman (2008) analyze a stationary, single-stage inventory system, under periodic review, with fixed ordering costs and multiple sales levers (such as pricing, advertising, etc.). Song et al. (2009) study the optimal dynamic decision-making problem for a retailer in a price-sensitive, multiplicative demand framework. The model incorporates lost sales, holding cost, fixed and variable procurement costs, as well as salvage value. Sharkey (2011) models the supply chain as a network flow problem including pricing decisions where a single price is aimed to be set throughout the network. In the study, it is stated that pricing decisions and the network flow decisions should be handled simultaneously in order to maximize profits. Gümüş, Kaminsky, and Mathur (2011) analyze the impact of store capacity and extent of inter-product substitution in a retailer's assortment on the optimal timing and depth of price promotions using a two-step approach.

2.3 Price-demand relations in dynamic pricing theory

Chen and Simchi-Levi (2010) gave me inspiration for the research that explain price-demand relations. They conduct a literature research on price optimization models where the emphasis is on integrated production/inventory and pricing models that

may be used for decision support at both the operational and tactical levels. They present commonly used demand models, followed by a survey of deterministic periodic review inventory and pricing models.

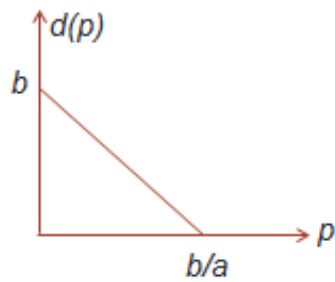
In Chen and Simchi-Levi (2010), deterministic demand models where demand of a product is a decreasing function of the price of that product, and revenue is a concave function of price is analyzed. Commonly used price-demand models for a single product are given in Figure 1 as follows:

(a) linear demand, $d(p) = b - ap$, $p \in [0, b/a]$, $a > 0$, $b \geq 0$

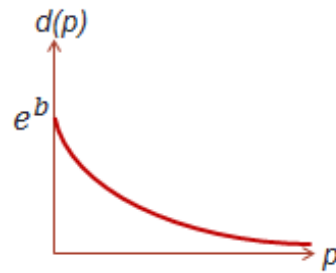
(b) exponential demand, $d(p) = \exp\{b - ap\}$, $a > 0$, $b > 0$

(c) iso-price elastic demand, $d(p) = ap^{-b}$, $a > 0$, $b > 1$

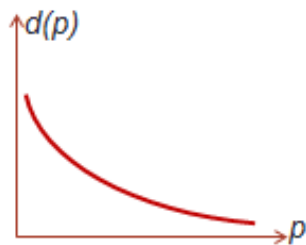
(d) logit demand. $d(p) = N \exp\{-ap\}/(1 + \exp\{-ap\})$



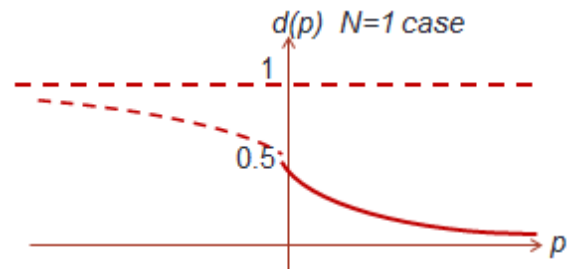
(a) linear demand



(b) exponential demand



(c) iso-price elastic demand



(d) logit demand

Figure 1. Graphs of demand functions

Chen and Simchi-Levi (2010) also state that when the multiple product case is considered, the model formulations change. Here, $p = (p_1, p_2, \dots, p_n)$ is the price vector of n products and $d(p) = (d_1(p), d_2(p), \dots, d_n(p))$ is the associated demand vector. The multi-product price-demand models reviewed in their study is given as follows:

- (a) linear demand, $d(p) = b + Ap$, $b = (b_1, b_2, \dots, b_n)$, $A = [a_{ij}]$
- (b) exponential demand, $d_i(p) = \exp\{b_i + A_i^T p\}$, $i = 1, 2, \dots, n$
- (c) iso-price elastic demand, $d_i(p) = b_i p_1^{-a_{i1}} p_2^{-a_{i2}} \dots p_n^{-a_{in}}$, $i = 1, 2, \dots, n$
- (d) logit demand. $d_i(p) = N \exp\{-a_i p_i\} / (1 + \sum_{j=1}^n \exp\{-a_j p_j\})$, $i = 1, 2, \dots, n$

In addition to the models proposed by Chen and Simchi-Levi (2010), Gümüş and Kaminsky (2010) introduce the classifications of inventory and pricing decisions from another perspective for modeling price-demand relationship. They aim to coordinate pricing and inventory decisions across time in a multi-product setting. Accordingly, models can be generated including the substitution effect showing that the prices of substitute products affect the demands of each other, as well as an intertemporal effect showing that the price of a product affects the demand in other periods. The existing literature is briefly reviewed in this study and classified into three main areas: (1) pricing with intertemporal demand models, (2) pricing models with single inventory replenishment, (3) pricing models with multiple inventory replenishment.

It should be noted that in the first three of the price-demand models reviewed in Chen and Simchi-Levi (2010), the market size is not fixed whereas in the Logit demand model, the total market size is equal to N (fixed). The probability that a customer chooses product j is derived from the Multinomial Logit (MNL) model that

is commonly used in the marketing literature. It would be an interesting idea to incorporate the models with fixed market size assumption into dynamic pricing models integrating production and transportation decisions. In the next section, market share models with special emphasis on market share attraction models are analyzed.

2.4 Market share models in marketing theory

Market share models are frequently seen in the marketing literature and they can be used to explain price-demand relations, and these models act as a basis to my thesis study. Researchers in the literature defined the ‘market share’ as a function of four famous marketing mix variables, namely, price, product properties, promotion efforts and place of sales. Falkenberg (1984) gives brief information on concurrent market share models in his study. He stated that initially in the literature, Lambin (1970) defines the market share as the log linear ratio of marketing mix variables. Weiss (1968) defines the market share as a function of the sum of the ratio of the price of the company over the competitors’ average price and the ratio of the advertising effort of the company over its competitors’ average advertising effort. Kotler (1971) handles market share as a function of the ratio of the sum of marketing mix variables of a firm over the sum of competitors’ marketing mix variables.

The models expressed above include market shares that are obtained by the ratios of the functions of a set of marketing mix variables. However in this thesis study, the market share of a product is defined as the relative ratio of the ‘attraction’ of that product, defined as a function of marketing mix variables, to the sum of the attractions of the competing products. These types of models are generally referred to as market share attraction (MSA) models in the literature.

Bell, Keeney, and Little (1975) is the initial study in the literature where the term ‘attraction’ is used to express the market share. They build a market share theorem by improving Kotler’s market share model. They assume that several marketing models depend on the relationship that market share equals marketing effort divided by total marketing effort. They replace marketing effort with its resulting attraction. Here, attraction of a product is a function of marketing mix variables of the existing product and the substitute products. Thus letting a_i be the attraction of seller i , $i=1,2,\dots,n$, the market share of supplier i , m_i is a ratio of its own attraction over the total attraction of substitute products formulated as $m_i = a_i / \sum_{i=1}^n a_i$. The following assumptions are considered: (1) attraction variable takes nonnegative values, (2) equal attraction values resemble equal market shares, (3) the market share of a seller is affected in the same manner by the change in the attraction value of another seller.

Schuur, Badur, and Sencer (2015) denote that although Bell et al. (1975) introduce the attraction as a function of the marketing mix variables; they don't propose a structure for this relation. In further exploration, Kotler (1984) proposes that a firm's market share, s_i is equal to its share of marketing effort, M_i , $i = 1, 2, \dots, m$, i.e., $s_i = M_i / \sum_{i=1}^m M_i$. Here, the marketing effort is defined by price (P_i), advertisement expenditure (A_i), and distribution efforts (D_i) that can take several forms. The most frequently used forms are the multiplicative competitive interaction (MCI) model, $M_i = P_i^p A_i^a D_i^d$ where p , a , d are estimated parameters, and the multinomial logit (MNL) model, $M_i = \exp\{pP_i + aA_i + dD_i\}$.

Cooper and Nakanishi (1988) are interested in market-share analysis for evaluating competitive marketing effectiveness. They review the common forms of market share, namely, the linear model, the multiplicative model, and the exponential

model. However, these models suffer from lack of logical consistency, i.e., they do not include the assumption that market shares are between 0 and 1, and market shares should add up to 1 (Naert & Bultez, 1973). This brings several computational complexities in competitive environments.

The competitive environment, online auction, and electronic reverse auction concepts that exist in the marketing literature may end up with a connection between marketing theory and decision support systems (Cooper & Nakanishi, 1988).

2.5 Decision support systems for dynamic pricing

Elmaghraby and Keskinocak (2003) state that the availability of smart decision support tools makes offerings to the customers about periods to buy the product by collecting price data easier. It is also stated in the same study that companies can automatically track their competitors' prices considering their own pricing decisions with the help of information technologies.

Cooper and Nakanishi (1988) propose a framework to develop a decision support system using simulation. They develop an example system called CASPER (Competitive Analysis System for Promotional Effectiveness Research) and state that real-world systems can be designed by extending their example. They determine the functions of a DSS as learning from history, simulating the plans of sales and retailers of both manufacturers and retailers, and testing the marketing strategies in a dynamic, competitive environment.

The studies on decision support systems developed for production, transportation, and pricing integration is very limited in the literature. The most comprehensive study on that topic is Keskinocak, Goodwin, Wu, Akkiraju, Murthy, (2001) as far as I can see. Keskinocak et al. (2001) develop a framework for DSS in

supply chains. A DSS may help manufacturers give important decisions in a supply chain when looked at the point of view of the manufacturer. They define the aim of the DSS as supplying needs of different agents such as manufacturers, distributors, brokers, and consumers. The DSS that is about to be developed in the study should answer these three main questions: (a) what will be sold? (b) what will be bought? (c) what will be promoted? The developed DSS makes suggestions about inventory, capacity, price, and constraints, it gives support for e-inventory and e-capacity buying decisions in backlogging conditions. The DSS also works on sales in order to increase the attraction of a product towards customers. The developed DSS consists of these outer units: (a) primary sites or trading sites, (b) internet, (c) users, (d) backend system. The inner units of the DSS are: (a) query generating agents, (b) search agents, (c) matching agents.

2.6 Contribution to the literature

There are various studies on production-transportation integration in the literature. MSA models have a fifty years old history in marketing theory. Also, a number of different models have been developed to explain price-demand relations in the pricing literature. However, MSA models have not been widely used to determine price-demand relations until now.

The introduction of the MSA model to explain price-demand relations is the first contribution. In our modeling approach, the attraction of product type k , A_k is assumed to be a linear decreasing function of price, i.e., $A_k = 1 - b_k P_k$, where b_k is the price sensitivity of product type k and P_k is the price of product type k . A small b_k shows that the firm is robust enough to set high prices for product type k without causing significant losses in its attraction. Conversely, a large b_k shows that the firm

operates in a highly competitive environment where the attraction of product type k is highly sensitive to the choice of price.

The integration of production, transportation and pricing decisions using MSA models is the second contribution. Besides the fact that MSA models have been widely studied in the marketing literature, integration of pricing, production and transportation decisions using MSA models have not been thoroughly established. In our model, either (i) the uncapacitated case: each of both product types reserves enough capacity to fulfill the total market demand or (ii) the capacitated case: none of both product types reserves enough capacity to fulfill the total market demand but together they reserve enough capacity to do so is considered.

A comprehensive study should also include the process of building a DSS that will guide decision makers. The literature review proved that the lack of DSSs for integrated supply chains is a powerful challenge. Inputs, outputs, graphs and analyzes should be one-button-away from the decision makers to give the right decision at the right time.

The lack of a DSS would result in the lack of economic interpretations and the management would not be able to foresee possible future improvements. These facts affect the long-term plans and the strategy of the company in a negative way. Thus, the third contribution is the development of an efficient, user-friendly DSS that will ensure the centralization of decisions in a supply chain that will also give managerial insights on future improvements and the upper bounds for expenses.

CHAPTER 3

DEVELOPMENT OF AN INTEGRATED MODEL FOR PRODUCTION, TRANSPORTATION AND PRICING DECISIONS

3.1 Model formulation

A supply chain where multiple types of substitute products are produced and sold is handled in this study. By investigating the classification present in the literature, the model developed in the study contains dynamic pricing, stockouts and capacity limits, there is no production setup cost and demand is linked to price with a nonlinear function. The developed market share model is a linear approximation of the MNL model, which leads to easier computations. The model includes all assumptions of Bell et al. (1975) in order to establish logical consistency. The change in the attraction sensitivity of a product affects the market power of the product in the opposite way. In the market share models, there is no such assumption that the total market share should add up to 1. In the developed market share model, this assumption is included. In this chapter, a multiple period model with no competition, multiple factories, multiple markets, multiple product types and fixed demand is developed. The objective is to maximize the total net profit. The model is given below with the indices, formulation, parameters and variables:

Indices:

- i : factory index ($i=1,2,\dots,I$)
- j : market index ($j=1,2,\dots,J$)
- k : product type index ($k=1,2,\dots,K$)
- t : time period index ($t=1,2,\dots,T$)

Parameters:

- u_{ikt} : Unit production cost for product type k produced at factory i in period t
- h_{ikt} : Unit inventory holding cost for product type k stored at factory i in period t
- c_{ijkt} : Unit transportation cost for product type k from factory i to market j in period t
- s_{jkt} : Unit stockout cost for product type k to be sold in market j in period t
- q_{jt} : Total demand in market j in period t
- f_{ikt} : Production capacity of factory i for product type k in period t
- b_{jkt} : Attraction sensitivity for product type k in market j in period t

Decision Variables:

- X_{ijkt} : Transportation quantity of product type k produced at factory i and sent to market j in period t
- Y_{ikt} : Production quantity of product type k produced at factory i in period t
- P_{jkt} : Price per unit of product type k in market j in period t

Auxillary Variables:

- I_{ikt} : Inventory level of product type k hold in factory i in period t
- A_{jkt} : Attraction of product type k in market j in period t
- M_{jkt} : Market share of product type k in market j in period t

$$\begin{aligned} \max \sum_{j,k,t} P_{jkt} \sum_i X_{ijkt} - & \left(\sum_{i,k,t} u_{ikt} Y_{ikt} + \sum_{i,k,t} h_{ikt} I_{ikt} + \sum_{i,j,k,t} c_{ijkt} X_{ijkt} \right) \\ & - \sum_{j,k,t} s_{jkt} \left(M_{jkt} q_{jt} - \sum_i X_{ijkt} \right) \end{aligned} \quad (1)$$

$$\text{s.t.} \quad Y_{ikt} \leq f_{ikt} \quad \forall i, k, t \quad (2)$$

$$\sum_i X_{ijkt} \leq M_{jkt} q_{jt} \quad \forall j, k, t \quad (3)$$

$$M_{jkt} = \frac{A_{jkt}}{\sum_k A_{jkt}} \quad \forall j, k, t \quad (4)$$

$$A_{jkt} = 1 - b_{jkt} P_{jkt} \quad \forall j, k, t \quad (5)$$

$$I_{ik(t-1)} + Y_{ikt} - \sum_j X_{ijkt} - I_{ikt} = 0 \quad \forall i, k, t \quad (6)$$

$$X_{ijkt}, Y_{ikt}, I_{ikt}, P_{jkt}, A_{jkt}, M_{jkt} \geq 0 \quad \forall i, j, k, t \quad (7)$$

Equation (1) is the objective function which aims to maximize the total profit that is obtained by the sum of total production cost, total inventory cost, total transportation cost and total stockout cost subtracted from total revenue. Constraint (2) is the production capacity constraint for all factories in all periods where total production in factory i for all product types cannot exceed the production capacity in that factory in period t . Constraint (3) is the market demand constraint where the total transportation amount of product type k from all factories to market j in period t cannot exceed the demand of product type k in market j in period t . The demand of product type k in market j in period t is obtained by multiplying the market share of product type k (M_{jkt}) with the total demand in that market in that period. Constraint (4) gives the market share of product type k as a ratio of individual attraction of product type k in market j in period t divided by total attraction of all product types in market j in period t . Constraint (5) explains the attraction of product type k realized

in market j in period t as a decreasing linear function of the price of product type k in market j in period t . Here, b_{jkt} implies the attraction sensitivity of product type k realized in market j in period t , which is a measure of the effects of other product mix variables than price such as quality, social awareness, sustainability, etc. Constraint (6) is the inventory balance constraint between the production amount, total transportation amount to all markets, the inventory level in the previous period and the inventory level in the current period realized in every factory i for product type k in period t . Here, the model has the assumption that inventory is hold only at the factories, and not in the markets. Constraint (7) denotes that all decision variables for production, transportation and inventory amounts, prices, attractions, and market shares must be nonnegative.

3.2 Additional constraints for the solution of the model in GAMS

The problem is modelled with GAMS software version 23.5.1 which provides modelling and solving linear, nonlinear and mixed integer optimization problems and the problem is solved using CONOPT solver version 3.14U. As will be clear in the next sections, in the cooperative competition with MSA model, the optimal prices are very close to their upper bounds, leading the sum of attractions of all product types in any market in any period being equal to zero, i.e. $\sum_k A_{jkt} = 0$ causing division by zero error in GAMS for Constraint (4). To avoid this error, Constraint (4) is changed to Constraint (4') and Constraint (8) is introduced to refine the model from division by zero error.

$$M_{jkt} = \frac{A_{jkt}}{\sum_k A_{jkt} + (5 \times 10^{-13})} \quad \forall j, k, t \quad (4')$$

$$\sum_k M_{jkt} \geq 0.9999999 \quad \forall j, t \quad (8)$$

In the original model Constraint (4) assures that market shares of all product types add up to 1, i.e. $\sum_k M_{jkt} = 1$ in all markets and periods if the sum of attractions is positive. When both prices hit their upper bounds, then the sum of attractions of all product types are equal to zero, i.e. $\sum_k A_{jkt} = 0$ leading division by zero error for Constraint (4). To avoid this error, a negligibly small real parameter is added into the denominator in Constraint (4'). Furthermore, Constraint (8) is added into the model to assure that market shares of all product types approximately add up to 1, meaning that attractions of all product types cannot simultaneously be zero in any market in any period.

3.3 Solution of the model in single period

The single period model is solved and analyzed under different scenarios. Table 2 contains a sample test data in a single factory, single market, single period and two products environment.

Table 2. Input Parameters of the Single Period Problem

Production Costs (u_{ikt})				Inv. Holding Costs (h_{ikt})				Transportation Costs (c_{ijkt})					Stockout Costs (s_{jkt})			
i	k	t	Value	i	k	t	Value	i	j	k	t	Value	j	k	t	Value
1	1	1	3	1	1	1	0.3	1	1	1	1	1	1	1	1	5
1	2	1	3	1	2	1	0.3	1	1	2	1	1	1	2	1	5
Demand (q_{jt})				Capacity (f_{ikt})				Attraction Sensitivity (b_{jkt})				Initial Inventory (I_{ik0})				
j	t	Value		i	k	t	Value	i	k	t	Value	j	k	t	Value	
1	1	500		1	1	1	10000	1	1	1	0.01	1	1	0	0	
				1	2	1	10000	1	2	1	0.01	1	2	0	0	

In the next two sections, sensitivity with respect to the attraction sensitivity (b_{111}) and unit production cost (u_{111}) are explored. The aim of the sensitivity analysis is to obtain the sensitivity of the optimal solution with respect to the changes

in the input parameters. Thus, the optimal market shares, prices, and the optimal profit are observed by changing the parameters stated above.

3.3.1 Sensitivity analysis with respect to attraction sensitivity: uncapacitated case

In the first sensitivity analysis, only the attraction sensitivity of the first product (b_{111}) is changed where other parameters are kept unchanged. There is a single factory ($I=1$), single market ($J=1$), single period ($T=1$) and there are two product types ($K=2$). Table 3 contains the outputs of the sensitivity analysis.

Table 3. Sensitivity Analysis 1: Changing the Attraction Sensitivity of Product Type 1

#	Input	Optimal Outputs										
	b_{111}	P_{111}	P_{121}	A_{111}	A_{121}	M_{111}	M_{121}	Y_{111}	Y_{121}	X_{1111}	X_{1121}	Σ Profit
1	0.0093	107.5268	100.0000	9.812E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	51763.335
2	0.0094	106.3829	100.0000	9.810E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	51191.385
3	0.0095	105.2631	100.0000	9.808E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	50631.476
4	0.0096	104.1666	100.0000	9.806E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	50083.291
5	0.0097	103.0927	100.0000	9.804E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	49546.208
6	0.0098	102.0407	100.0000	9.802E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	49020.308
7	0.0099	101.0100	100.0000	9.800E-07	0.000E+00	1.000	0.000	499.999	0.000	499.999	0.000	48504.952
8	0.0100	99.9999	99.9999	6.928E-07	6.928E-07	0.500	0.500	250.000	250.000	250.000	250.000	47999.931
9	0.0101	99.0099	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
10	0.0102	98.0392	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
11	0.0103	97.0874	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
12	0.0104	96.1538	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
13	0.0105	95.2381	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
14	0.0106	94.3396	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
15	0.0107	93.4579	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902

The aim of this sensitivity analysis is to see the effect of attraction sensitivity parameter (b_{jkt}) on the optimal solution obtained. In Table 3, the 8th scenario with $b_{111} = 0.01$ refers to the initial setting with two identical products and the other scenarios are obtained by changing b_{111} from its current level. As expected in a symmetric setting of input parameters, prices, attraction values, market shares, production amounts, and transportation amounts are identical for the two product

types in the optimal solution. Market shares are plotted with respect to attraction sensitivity of product type 1 in Figure 2.

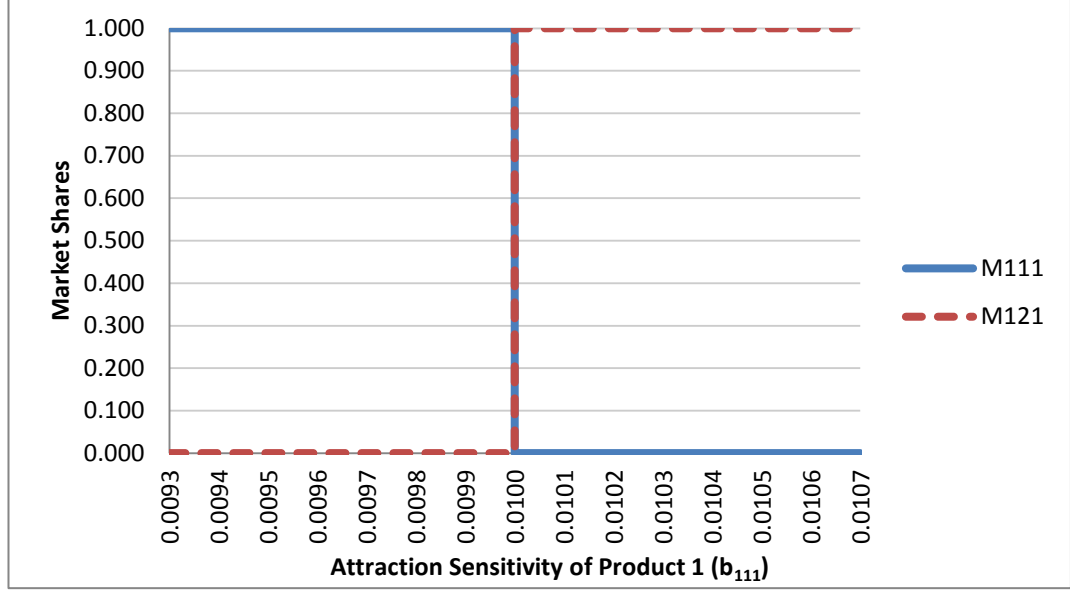


Figure 2. Market share-attraction sensitivity graph for Sensitivity Analysis 1

To interpret the findings in Table 3, let us refer to Theorem 1 by Schuur et al. (2015). They consider a problem with multiple product types, a single factory, a single market, and a single period in a cooperative competition environment with the same MSA model provided in Constraint (4) and Constraint (5). The aim of the theorem is to show how to optimize the pricing decisions under capacity restrictions in a cooperative environment where pricing decisions are given centrally so as to maximize the total profit. In the cooperative environment, explicit characterizations of the optimal prices are derived as a function of fixed production capacities.

In Schuur et al. (2015), the notations and the assumptions used are as follows: d is the total market demand, and c_i is the unit production cost of product type i , ($i=1,2,\dots,I$). cap_i is the production capacity reserved for product type i . β_i is the relative capacity of product type i , which is equal to cap_i/d . α_i is the coefficient

indicating decrease of attraction with price for product type i . It is assumed that $0 < \alpha_i < (\frac{1}{c_i})$. It is also assumed that there is enough production capacity to serve the market by all substitute product types, so $\sum_{i=1}^I \beta_i \geq 1$.

The decision variables are P_i , price per unit of product type i . The auxiliary variables are as follows: The attraction of product type i is given by $A_i = 1 - \alpha_i P_i$, the market share of product type i is given by $m_i = \frac{A_i}{\sum_{j=1}^I A_j}$, and the quantity produced of product type i is given by $Q_i = m_i d$.

The following assumptions and restrictions apply:

In view of the capacity restriction $Q_i \leq \beta_i d$, the market share of product type i is bounded by $m_i \leq \beta_i$. Here, m_i and Q_i are considered to be non-negative ($m_i \geq 0$ and $Q_i \geq 0$).

Attraction must be nonnegative for each product type i , i.e. $A_i = 1 - \alpha_i P_i \geq 0$. When this inequality is rearranged, it becomes $\alpha_i \leq \frac{1}{P_i}$. Moreover, to preserve the nonnegativity of the unit profit function, the price of a product type must be greater than or equal to the unit production cost of that product type, i.e. $P_i \geq c_i$. If the reverse of both sides of this inequality is taken, $\frac{1}{P_i} \leq \frac{1}{c_i}$ is obtained. Now, when these two inequalities are combined, it is seen that $\alpha_i \leq \frac{1}{P_i} \leq \frac{1}{c_i}$. Since both profit and attraction are assumed to be non-negative, i.e. $P_i - c_i \geq 0$ and $A_i \geq 0$, only values of the price P_i are considered such that $c_i \leq P_i \leq \frac{1}{\alpha_i}$.

The maximum obtainable attraction a_i is attained when $P_i = c_i$ at its lower limit, then $a_i = 1 - \alpha_i c_i \in (0,1)$. Hence, it is clear that $0 \leq 1 - \alpha_i P_i \leq 1 - \alpha_i c_i$, so A_i satisfies $0 \leq A_i \leq a_i$. Furthermore, there is a profit potential of product type i as π_i that implies the maximum profit possible, it can be obtained by subtracting cost

from maximum price, i.e. $\pi_i \equiv \frac{1}{\alpha_i} - c_i$. This equation can be written in terms of α_i and a_i by rearranging the terms such that $\pi_i \equiv \frac{1-\alpha_i c_i}{\alpha_i} = \frac{a_i}{\alpha_i}$. So, the profit potential of product type i is equal to $\pi_i \equiv \frac{1}{\alpha_i} - c_i = \frac{a_i}{\alpha_i}$ where $1/\alpha_i$ is the maximum price that can be set to obtain a nonnegative attraction. This profit potential can only be realized approximately as $P_i \uparrow \frac{1}{\alpha_i}$, since putting P_i equal to $\frac{1}{\alpha_i}$ would wipe out the production quantity Q_i . Let the profit potentials be ranked in non-increasing order as: $\pi_1 \geq \pi_2 \geq \dots \geq \pi_i$.

To maximize total profit, the function $d G(P_1, P_2, \dots, P_I)$ has to be maximized with $G(P_1, P_2, \dots, P_I) = \sum_{i=1}^I (P_i - c_i) m_i$ over all P_i satisfying $m_i \leq \beta_i$ and $c_i \leq P_i \leq \frac{1}{\alpha_i}$. Before tackling this issue, a helpful index can be introduced. Since $\sum_{i=1}^I \beta_i \geq 1$, there is an index i_0 such that $\sum_{i=1}^{i_0-1} \beta_i < 1$ and $\sum_{i=1}^{i_0} \beta_i \geq 1$.

The following theorem is proven in Schuur et al. (2015):

Theorem 1 (Schuur et al., 2015): The function $G(P_1, P_2, \dots, P_I)$ has as supremum value $\pi = \sum_{i=1}^{i_0} m_i^* \pi_i$, where the m_i^* are the optimal market shares given by $m_i^* = \beta_i$ for $i = 1, 2, \dots, i_0 - 1$ and by $m_{i_0}^* = 1 - \sum_{i=1}^{i_0-1} \beta_i$. This supremum can be approximated arbitrarily close by pricing the produced product types as follows: $P_i = \frac{1}{\alpha_i} (1 - \frac{m_i^*}{m_1^*} (1 - \alpha_1 P_1))$ for $i = 1, 2, \dots, i_0$ letting $P_1 \uparrow \frac{1}{\alpha_1}$ and $P_i = \frac{1}{\alpha_i}$ for $i = i_0 + 1, i_0 + 2, \dots, I$.

In the model, there are four indices. As there is a single factory, a single market, and a single period, the indices i, j , and t can be ignored. So, there is a single index of k , ($k=1, 2$) the product type index instead of i . The attraction sensitivity parameter b_{jkt} becomes b_k instead of α_i . Total demand can now be symbolized with q instead of d . The production capacity becomes f_k instead of cap_i , and the

production amount becomes Y_k instead of Q_i . The sum of unit production cost (u_{ikt}) and unit shipment cost (c_{ijkt}) of product type k corresponds to c_k .

As stated, when the problem is solved, the optimum attractions and prices can be obtained as appear in Table 3. When the attraction sensitivity of product type 1 is less than or equal to the attraction sensitivity of product type 2, i.e. $b_1 \leq b_2 = 0.01$, it is expected that the profit potential of product type 1 will be no less than the profit potential of product 2, i.e. $\pi_1 \geq \pi_2$ since there are identical unit costs, i.e. $c_1 = c_2$.

Let us consider the first scenario in Table 3 with attraction sensitivities of product type 1 and product type 2 being $b_1 = 0.0093$ and $b_2 = 0.01$. Referring to Table 2, c_1 and c_2 can be calculated as the sum of unit production cost and unit transportation cost of product type 1 and product type 2, i.e. $c_1 = c_2 = 3 + 1 = 4$. Then, the profit potentials of product type 1 and product type 2 will be $\pi_1 = \frac{1}{b_1} - c_1 = 103.527$ and $\pi_2 = \frac{1}{b_2} - c_2 = 96$. The profit potentials of product type 1 and product type 2 can be ranked as $\pi_1 \geq \pi_2$. The relative capacities β_k can be calculated as $\beta_1 = \frac{f_1}{q} = \frac{10000}{500} = 20$ and $\beta_2 = \frac{f_2}{q} = \frac{10000}{500} = 20$. It states that both products have enough capacity to satisfy the demand.

Using the definition of i_0 provided in the theorem, its corresponding value k_0 in this case is found as $k_0 = 1$ since $\sum_{k=1}^{k_0-1} \beta_k = \sum_{k=1}^0 \beta_k = 0 < 1$ and $\sum_{k=1}^{k_0} \beta_k = \sum_{k=1}^1 \beta_k = \beta_1 = 20 \geq 1$. Then, by calculating $m_{k_0}^* = 1 - \sum_{k=1}^{k_0-1} \beta_k$, it is seen that $m_1^* = 1 - \sum_{k=1}^0 \beta_k = 1$ and $m_2^* = 0$. Letting $P_1 \uparrow \frac{1}{b_1}$, P_1^* approximates arbitrarily close to its upper bound $\frac{1}{b_1} = 107.527$. Then, P_2^* can be calculated at its upper bound as $P_2^* = \frac{1}{b_2} = 100$. Now, the supremum value $\pi = \sum_{k=1}^{k_0} m_k^* \pi_k$ can be calculated as $\pi = \sum_{k=1}^1 m_k^* \pi_k = 103.527$. Product type 1 will have positive attraction ($A_1 > 0$)

whereas product type 2 will have zero attraction ($A_2 = 0$), resulting in market shares of product type 1 and product type 2 being $M_1^* = 1$ and $M_2^* = 0$.

The model produces the product with higher profit potential as much as the minimum of its available capacity and total demand. As the input production capacities are very high in this scenario, the production quantity of product type 1, Y_1^* is arbitrarily close to the total market demand, $qM_1^* = 500$ whereas $Y_2^* = qM_2^* = 0$. Since there is only one market, the total transportation amount $\sum_{k=1}^2 X_k$ is equal to the production amounts letting X_1^* arbitrarily close to $qM_1^* = 500$ and $X_2^* = 0$. The objective function value can now be calculated as $qG(P_1, P_2) = q \sum_{k=1}^K (P_k - c_k)m_k = (500)(103.527) = 51763.500$.

In the first 7 scenarios in Table 3 where $b_1 < b_2 = 0.01$, it is seen that product type 1 has always a market share equal to 1 ($M_1^* = 1$), and product type 2 has no market share ($M_2^* = 0$) since $\pi_1 > \pi_2$. In all these scenarios, the optimal prices for product type 1 are arbitrarily close to their upper bounds whereas the optimal prices are at their upper bounds for product type 2. As the attraction sensitivity of product type 1 increases ($b_1 \uparrow$), the optimal price of product type 1 decreases ($P_1^* \downarrow$) because the price upper limit decreases ($1/b_1 \downarrow$), which further leads to a decrease in the optimal objective function value.

Conversely, in scenarios 9-15 in Table 3, when the attraction sensitivity of product type 1 is greater than or equal to the attraction sensitivity of product type 2, i.e. $b_1 \geq b_2 = 0.01$, it is expected that the profit potential of product type 1 will be no more than the profit potential of product type 2, i.e. $\pi_1 \leq \pi_2$ since there are identical unit costs, i.e. $c_1 = c_2$. Considering the last scenario in Table 3 where the attraction sensitivity of product type 1 is equal to $b_1 = 0.0107$ and the attraction sensitivity of product 2 is $b_2 = 0.01$, and the unit costs being equal to $c_1 = c_2 = 4$,

the profit potentials of product type 1 and product type 2 will be $\pi_1 = 89.46$ and $\pi_2 = 96$. Then, the profit potentials of product type 1 and product type 2 can be ranked as $\pi_2 \geq \pi_1$. The relative capacities are calculated as $\beta_1 = \beta_2 = 20$.

Using the same theorem of Schuur et al. (2015) by interchanging the indices, it is seen that $m_2^* = 1$ and $m_1^* = 0$. Letting $P_2 \uparrow \frac{1}{b_2}$, P_2^* approximates arbitrarily close to its upper bound $\frac{1}{b_2} = 100$. Then, P_1^* can be calculated at its upper bound as $P_1^* = \frac{1}{b_1} = 93.458$. Now, the supremum value π can be calculated as $\pi = 96$. Product type 2 will have positive attraction ($A_2 > 0$) whereas product type 1 will have zero attraction ($A_1 = 0$), resulting in a market share of product type 2 equal to 1 ($M_2^* = 1$), and product type 1 will have zero market share ($M_1^* = 0$).

For scenarios 9 - 15 where $b_1 > b_2 = 0.01$, it is seen that product type 2 has always a market share equal to 1 ($M_2^* = 1$), and product type 1 has no market share ($M_1^* = 0$) since $\pi_1 < \pi_2$. The increase in the attraction sensitivity of product type 1 ($b_1 \uparrow$) still causes a decrease in the optimal price of product type 1 ($P_1^* \downarrow$) since $P_1^* = \frac{1}{b_1}$. Although P_1^* decreases, product type 1 is still not produced, i.e. $Y_1^* = X_1^* = 0$ since $\pi_2 > \pi_1$ in scenarios 9-15 in Table 3. Actually, in all these scenarios P_2^* is arbitrarily close to its constant upper bound $\frac{1}{b_2} = \frac{1}{0.01} = 100$ and handles the whole production, i.e. $Y_2^* = X_2^* = 500$. Hence, the optimal objective function value is fixed at 47999.902 since revenues and costs are all the same in these scenarios.

Symmetric attraction sensitivities for product type 1 and product type 2 can be considered as a special case where $b_1 = b_2 = 0.01$ and $\pi_1 = \pi_2 = 96$. Then, letting the prices of both product types $P_1 \uparrow \frac{1}{b_1}$ and $P_2 \uparrow \frac{1}{b_2}$, P_1^* and P_2^* approximate arbitrarily close to their upper bound $\frac{1}{b_1} = \frac{1}{b_2} = 100$. Looking at scenario 8 in Table

3, it is seen that both product types will have positive attractions ($A_1 = A_2 > 0$), so product type 1 and product type 2 share the market equally ($M_1^* = M_2^* = 0.5$). The production and transportation amounts of both products are equal to the market demand, i.e. $Y_1^* = Y_2^* = X_1^* = X_2^* = 250$.

In Figure 2, optimal market shares in Table 3 are plotted with respect to the attraction sensitivity of product type 1, b_{111} . It is clear that all market demand is satisfied by product type 1 when $b_1 > b_2$. Similarly, all market demand is satisfied by product type 2 when $b_2 > b_1$. When the attraction sensitivities of both product types are equal ($b_1 = b_2 = 0.01$), then there are multiple optimal solutions. Actually, any solution providing $\sum_{k=1}^2 M_k = 1$ where $0 \leq M_k \leq 1$ for $k = 1, 2$ is optimal for this symmetric case.

3.3.2 Sensitivity analysis with respect to attraction sensitivity: capacitated case

As a further step, the condition with capacity constraints can be considered now. Consider the case where the company has limited production capacity of both product types f_1 and f_2 such that none of them is able to fulfill the total demand q whereas they can satisfy the total demand when producing together, i.e. $f_1 + f_2 > q$.

Now let us try to observe how the optimal solution is affected in this capacitated case. Letting the production capacities of product type 1 and product type 2 being $f_1 = f_2 = 300$, the model is solved once again by considering the capacity constraints. The total demand cannot be satisfied individually whereas both product types can satisfy the demand when they are produced together, so $f_1 + f_2 = 600 > q = 500$. Results are tabulated in Table 4 as follows:

Table 4. Changing the Attraction Sensitivity of Product Type 1 in Capacitated Setting

#	Input	Optimal Outputs										
	b_{111}	P_{111}	P_{121}	A_{111}	A_{121}	M_{111}	M_{121}	Y_{111}	Y_{121}	X_{1111}	X_{1121}	Σ Profit
1	0.0093	107.5268	99.9999	7.948E-7	5.299E-7	0.600	0.400	300.000	200.000	300.000	200.000	50257.992
2	0.0094	106.3829	99.9999	7.978E-7	5.319E-7	0.600	0.400	300.000	200.000	300.000	200.000	49914.821
3	0.0095	105.2631	99.9999	8.008E-7	5.338E-7	0.600	0.400	300.000	200.000	300.000	200.000	49578.875
4	0.0096	104.1666	99.9999	8.037E-7	5.358E-7	0.600	0.400	300.000	200.000	300.000	200.000	49249.928
5	0.0097	103.0927	99.9999	8.066E-7	5.378E-7	0.600	0.400	300.000	200.000	300.000	200.000	48927.764
6	0.0098	102.0407	99.9999	8.095E-7	5.397E-7	0.600	0.400	300.000	200.000	300.000	200.000	48612.174
7	0.0099	101.0100	99.9999	8.124E-7	5.416E-7	0.600	0.400	300.000	200.000	300.000	200.000	48302.959
8	0.0100	99.9999	99.9999	6.928E-07	6.928E-07	0.500	0.500	250.000	250.000	250.000	250.000	47999.931
9	0.0101	99.0098	99.9999	5.415E-7	8.123E-7	0.400	0.600	200.000	300.000	200.000	300.000	47801.910
10	0.0102	98.0392	99.9999	5.395E-7	8.093E-7	0.400	0.600	200.000	300.000	200.000	300.000	47607.773
11	0.0103	97.0873	99.9999	5.376E-7	8.064E-7	0.400	0.600	200.000	300.000	200.000	300.000	47417.406
12	0.0104	96.1538	99.9999	5.357E-7	8.035E-7	0.400	0.600	200.000	300.000	200.000	300.000	47230.700
13	0.0105	95.2380	99.9999	5.338E-7	8.006E-7	0.400	0.600	200.000	300.000	200.000	300.000	47047.551
14	0.0106	94.3396	99.9999	5.319E-7	7.978E-7	0.400	0.600	200.000	300.000	200.000	300.000	46867.857
15	0.0107	93.4579	99.9999	5.300E-7	7.950E-7	0.400	0.600	200.000	300.000	200.000	300.000	46691.521

Here, it would be interesting to apply Theorem 1 in Schuur et al. (2015) since the intention is to see the effect of the capacitated setting supporting Section 3.3. It is clear that none of the product types are capable of satisfying the demand individually. Since $\pi_1 > \pi_2$ for scenarios 1-7, the market share of product type 1 will be $M_1^* = \beta_1 = \frac{f_1}{q} = \frac{300}{500} = 0.6$. The rest of the market share will be captured by product type 2, so $M_2^* = 1 - \beta_1 = 1 - 0.6 = 0.4$. The prices of both product types are arbitrarily close to their upper limits, i.e. $P_1 \uparrow \frac{1}{b_1}$ and $P_2 \uparrow \frac{1}{b_2}$. Both product types have positive attractions, i.e. $A_1 > 0$ and $A_2 > 0$.

Similarly, scenarios 9-15 reflect the symmetric version and discussion. Since $\pi_2 > \pi_1$ in these scenarios, the market share of product type 2 will be $M_2^* = \beta_2 = \frac{f_2}{q} = \frac{300}{500} = 0.6$. The rest of the market share will be captured by product type 1, so $M_1^* = 1 - \beta_1 = 1 - 0.6 = 0.4$. The prices of both product types are arbitrarily close

to their upper limits, i.e. $P_1 \uparrow \frac{1}{b_1}$ and $P_2 \uparrow \frac{1}{b_2}$. Both product types have positive attractions, i.e. $A_1 > 0$ and $A_2 > 0$.

Looking at scenario 8 where both product types have equal attraction sensitivities, i.e. $b_1 = b_2 = 0.01$, the market is equally shared between two product types as both product types have enough capacity to satisfy the market demand individually as in the other solutions in Section 3.3.1. ($f_1 = 300 > M_1^*q = 250$ and $f_2 = 300 > M_2^*q = 250$).

The result of decreasing b_{jkt} - referred to as b_k - on the optimal total profit in the uncapacitated setting and the capacitated setting is plotted in Figure 3.

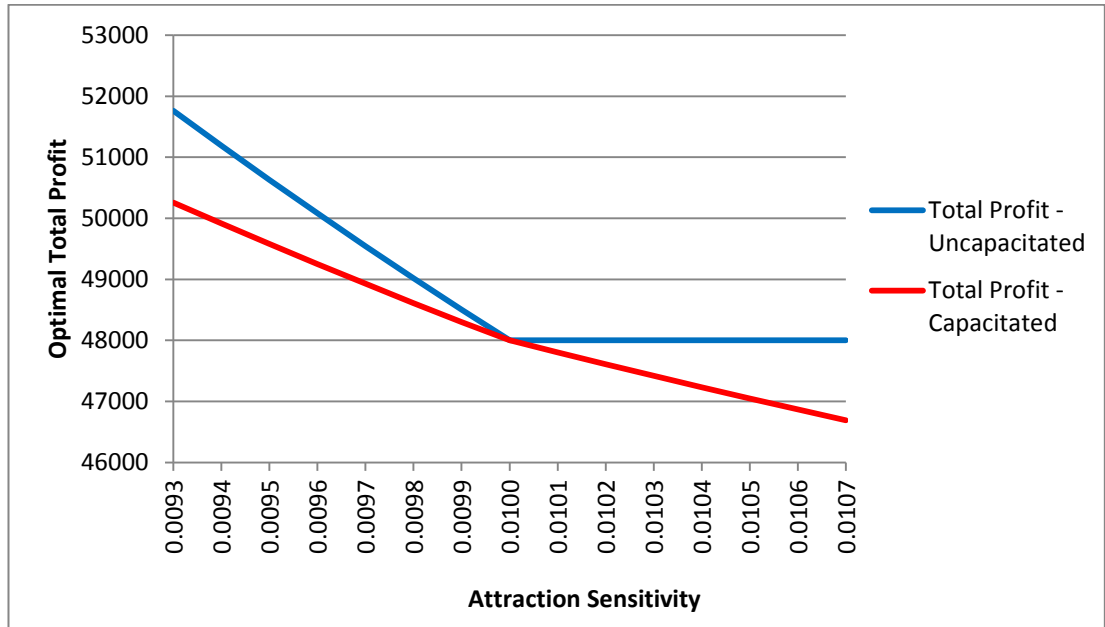


Figure 3. Optimal total profit as a function of attraction sensitivity – the uncapacitated setting and the capacitated setting

Looking at Figure 3, the objective function value of the uncapacitated setting is shown with the blue line whereas the objective function value of the capacitated setting is shown with the red line. As the attraction sensitivity of product type 1

increases ($b_1 \uparrow$), the profit potential of product type 1 decreases ($\pi_1 \downarrow$), which results in decreasing optimal total profit. It should be noted that both products are produced to fulfill the whole market demand in the capacitated setting scenarios. Hence, unlike in the uncapacitated setting, the optimal profits continue to decrease when $b_1 > 0.01$. Since increasing b_1 lowers the optimal selling price P_1^* , it results in decreased optimal total profits.

Comparing the optimal total profits in the uncapacitated setting and the capacitated setting as seen in Figure 3, it is clear that there is an increasing gap between the optimal total profit of the uncapacitated setting and the optimal total profit of the capacitated setting as moving away from the center where $b_1 = 0.01$.

3.3.3 Sensitivity analysis with respect to unit production cost: uncapacitated case

In this section, only the unit production cost of the first product type (u_{111}) is changed where other parameters are kept unchanged. There is a single factory ($I=1$), single market ($J=1$), single period ($T=1$) and there are two products ($K=2$). The aim of this sensitivity analysis is to see the effect of unit production cost (u_{ikt}) on the optimal solution obtained. In Table 5, scenario 4 with $u_{111} = u_{121} = 3$ refers to the initial setting with two identical product types and the other rows are obtained by changing u_{111} from its current level.

In this section, the similar discussion as in the sensitivity analysis is followed with respect to attraction sensitivity, b_{111} . In the first 3 scenarios, the unit production cost of product type 1 is smaller than the unit production cost of product type 2, i.e. $u_1 < u_2 = 3$, leading to the profit potential of product type 1 being greater than the profit potential of product type 2, i.e. $\pi_1 = \frac{1}{b_1} - c_1 > \pi_2 = \frac{1}{b_2} - c_2 = 96$.

Table 5. Sensitivity Analysis 2: Changing the Unit Production Cost of Product Type 1

#	Input	Optimal Outputs										
	u_{111}	P_{111}	P_{121}	A_{111}	A_{121}	M_{111}	M_{121}	Y_{111}	Y_{121}	X_{1111}	X_{1121}	Σ Profit
1	0	99.9999	100.0000	9.945E-07	0.000E+0	1.000	0.000	499.999	0.000	499.999	0.000	49499.901
2	1	99.9999	100.0000	9.900E-07	0.000E+0	1.000	0.000	499.999	0.000	499.999	0.000	48999.901
3	2	99.9999	100.0000	9.849E-07	0.000E+0	1.000	0.000	499.999	0.000	499.999	0.000	48499.902
4	3	99.9999	99.9999	6.928E-07	6.928E-07	0.500	0.500	250.000	250.000	250.000	250.000	47999.931
5	4	100.0000	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
6	5	100.0000	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
7	6	100.0000	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
8	7	100.0000	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
9	8	100.0000	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902
10	9	100.0000	99.9999	0.000E+00	9.798E-07	0.000	1.000	0.000	499.999	0.000	499.999	47999.902

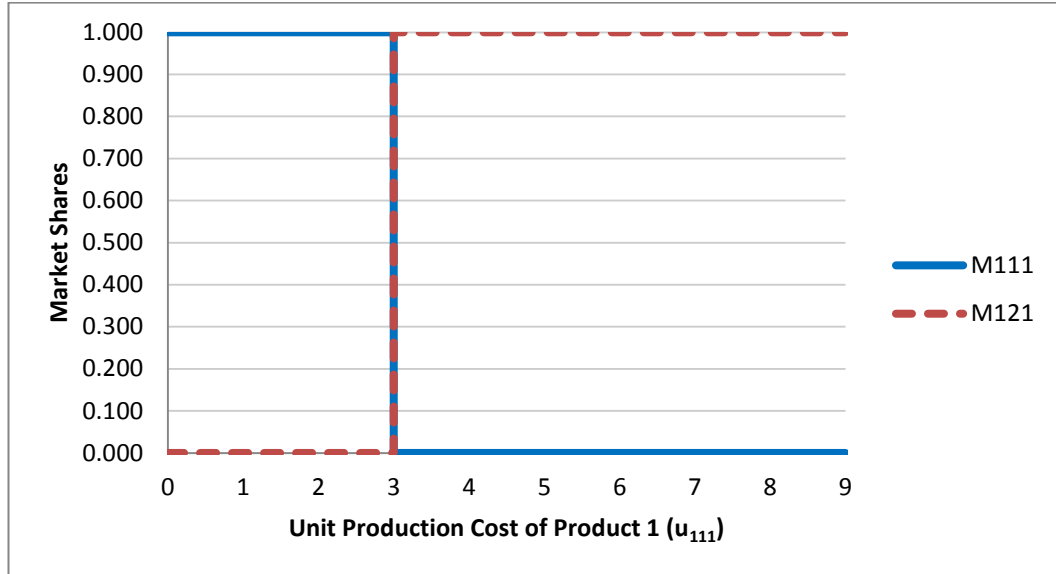


Figure 4. Market share-unit production cost graph for Sensitivity Analysis 2

Applying Theorem 1 in Schuur et al. (2015) with $k_0 = 1$, the optimal market shares are obtained as $m_1^* = 1$ and $m_2^* = 0$. Letting $P_1 \uparrow \frac{1}{b_1}$, P_1^* approximates arbitrarily close to its upper bound $\frac{1}{b_1} = 100$ whereas the optimal price of product type 2 is at its upper bound $P_2^* = \frac{1}{b_2} = 100$. This can also be followed in Table 5 from the decreasing optimal attraction values for product type 1 due to increasing

unit production costs, i.e. $A_1^* \downarrow$ and constant optimal attraction values for product type 2, i.e. $A_2^* = 0$ in the first 3 scenarios.

The production quantity of product type 1 is equal to the transportation quantity of product type 1, i.e. $Y_1^* = X_1^*$, and they are arbitrarily close to the market demand, i.e. $qM_1^* = 500$ whereas $Y_2^* = X_2^* = qM_2^* = 0$.

Looking at Table 5, it can be said that for scenarios 1-3 where product type 1 has a smaller unit production cost than product type 2 ($u_1 < u_2$), product type 1 captures the entire market share ($M_1^* = 1$ and $M_2^* = 0$), because product type 1 has a greater profit potential than product type 2 ($\pi_1 > \pi_2$). As the unit production cost of product type 1 increases ($u_1 \uparrow$), the optimal objective function value decreases because the profit potential of product type 1 decreases ($\pi_1 \downarrow$).

On the other hand, for scenarios 5-10 where the unit production cost of product type 1 is greater than the unit production cost of product type 2 ($u_1 > u_2$), product type 2 captures the entire market share resulting in $M_2^* = 1$ and $M_1^* = 0$, because product type 2 has a greater profit potential than product type 1 ($\pi_2 > \pi_1$). The objective function value does not change in these scenarios because the model tends to produce product type 2 with higher profit potential and fixed P_2^* and c_2 .

In Figure 4, optimal market shares in Table 5 are plotted with respect to the unit production cost of product type 1, u_{111} . It is clear that all market demand is satisfied by product type 1 when $u_1 < u_2$. Similarly, all market demand is satisfied by product type 2 when $u_2 < u_1$. When the unit production costs of both product types are equal ($u_1 = u_2 = 3$), then again there are multiple optimal solutions. Actually, any solution providing $\sum_{k=1}^2 M_k = 1$ where $0 \leq M_k \leq 1$ for $k = 1, 2$ is optimal for this symmetric case as stated in Section 3.3.1.

3.3.4 Managerial insights for the single period problem

Several managerial insights can be concluded from these results. The aim of a manager is to develop policies that will maximize the optimal total profit. A manager also aims to increase the optimal total profit by improving the system. The model satisfies the maximization of the optimal total profit in the first decision step. On the other hand, it also gives clues for the second decision step with the help of the sensitivity analyses about how the system can be improved.

The improvement of the system can be ensured by increasing the profit potential of the more attractive product. This can be achieved by either decreasing the attraction sensitivity which allows more space to increase the optimal prices or decreasing unit production, transportation or inventory holding costs.

If one or more of the 4P marketing mix variables except price are improved, the attraction sensitivity parameter (b_{jkt}) can be decreased which leads to higher profit potentials and greater revenues. The result of decreasing b_{jkt} - referred to as b_k - on the optimal total profit is plotted in Figure 5.

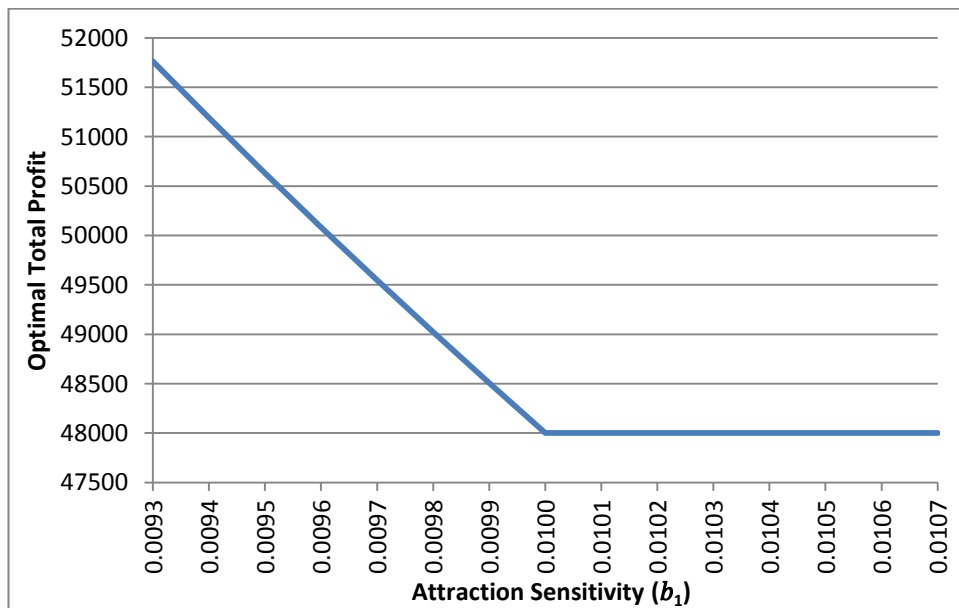


Figure 5. Optimal total profit as a function of attraction sensitivity

If the attraction sensitivity of product type 1 is greater than 0.01 ($b_1 > 0.01$), then an improvement should be made to decrease b_1 below the value of 0.01 so that product type 1 is more preferred than product type 2. Hence, product type 1 can increase its optimal total profit only when $b_1 < 0.01$. Furthermore, the plot in Figure 5 also helps to evaluate the maximum effort that has to be spent for such an improvement.

Another way of increasing the profit potential of a product is to decrease the unit production costs (u_{ikt}) or unit transportation costs (c_{ijkt}). Unit production and transportation costs can be decreased by process reengineering and/or better supply chain coordination. The result of decreasing u_{ikt} - referred to as u_k - on the optimal total profit is plotted in Figure 6.

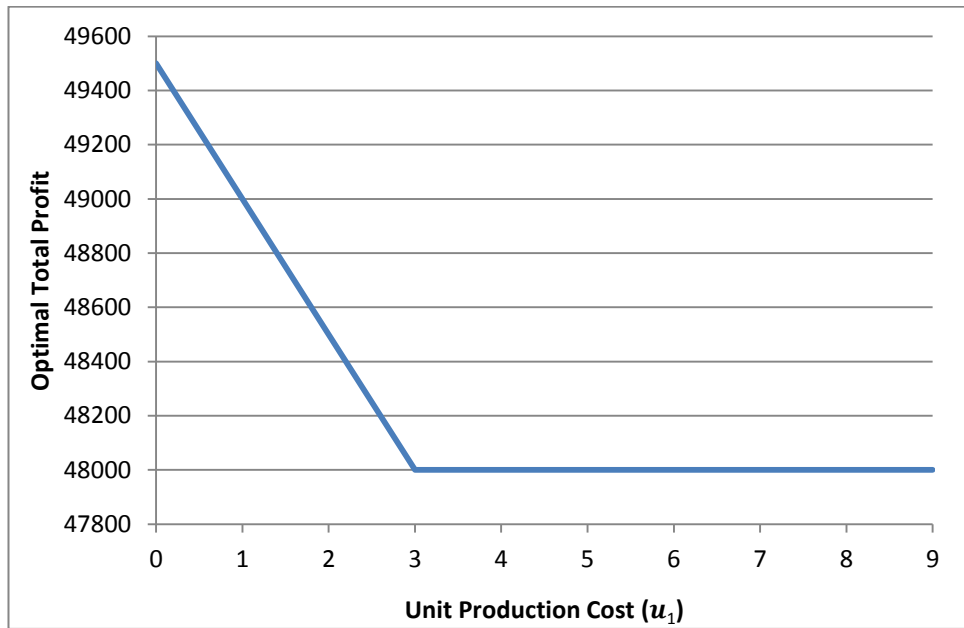


Figure 6. Optimal total profit as a function of unit production cost

Similar discussion about the optimal total profit and unit production cost can be made as stated in the previous paragraph. If the unit production cost of product

type 1 is greater than 3 ($u_1 > 3$), then an improvement should be made to decrease u_1 below the value of 3 so that product type 1 is more preferred than product type 2. Hence, product type 1 can increase the optimal total profit only when $u_1 < 3$. Furthermore, the plot in Figure 6 also helps to evaluate the maximum effort that has to be spent for such an improvement.

3.4 Solution of the model in multiple periods

For the solution of the model in multiple periods, the sample data set will be extended to include four periods instead of one, i.e. $T=4$. The other indices are kept the same ($I=1, J=1, K=2$). The aim of this analysis is to observe the behavior of the model with respect to the changes in relative capacity and unit stockout costs.

There are two cases in which low unit stockout costs and high unit stockout costs, s_{jkt} are present where the relative capacities β , the ratio of the total capacity in all four periods over the total demand in all four periods is 50%, 75%, 100%, and 125%. The input parameter sets of these problems are given below in Table 6. As this is an extension of the single period model to multiple periods, all parameters are symmetrical for both products except b_{jkt} , attraction sensitivities which have been determined by uniform distribution.

After solution phase is completed, optimal stockout quantities, optimal prices, optimal market shares, optimal total profits, and optimal total stockout costs for each product type in each period are obtained for both cases with low unit stockout costs and with high unit stockout costs with respect to relative capacities, as seen in Table 7. In order to represent the results better, optimal stockout quantities are represented with D_{jkt} . The light blue rows indicate the case with low unit stockout costs and the light red rows indicate the case with high unit stockout costs.

Table 6. Input Parameters of the Multiple Period Problems

Production Costs (u_{ikt})				Inv. Holding Costs (h_{ikt})				Transportation Costs (c_{ijkt})					Stockout Costs (s_{jkt})				
i	k	t	Value	i	k	t	Value	i	j	k	t	Value	j	k	t	Lo	Hi
1	1	1	3	1	1	1	0.3	1	1	1	1	1	1	1	1	5	80
1	2	1	3	1	2	1	0.3	1	1	2	1	1	1	2	1	5	80
1	1	2	3	1	1	2	0.3	1	1	1	2	1	1	1	2	5	80
1	2	2	3	1	2	2	0.3	1	1	2	2	1	1	2	2	5	80
1	1	3	3	1	1	3	0.3	1	1	1	3	1	1	1	3	5	80
1	2	3	3	1	2	3	0.3	1	1	2	3	1	1	2	3	5	80
1	1	4	3	1	1	4	0.3	1	1	1	4	1	1	1	4	5	80
1	2	4	3	1	2	4	0.3	1	1	2	4	1	1	2	4	5	80
Demand (q_{jt})				Total Capacity ($\sum_{ikt} f_{ikt}$)				Attraction Sensitivity (b_{jkt})					Initial Inventory (I_{ik0})				
j	t	Value		f_{ikt}	β	Value		j	k	t	Value		i	k	t	Value	
1	1	1000		250	50%	2000		1	1	1	0.033		1	1	1	0	
1	2	1000		375	75%	3000		1	2	1	0.011		1	2	1	0	
1	3	1000		500	100%	4000		1	1	2	0.015		1	1	2	0	
1	4	1000		625	125%	5000		1	2	2	0.022		1	2	2	0	
Total		4000						1	1	3	0.025		1	1	3	0	
$\beta = \frac{\sum_{ikt} f_{ikt}}{\sum_{jt} q_{jt}}$								1	2	3	0.028		1	2	3	0	
								1	1	4	0.022		1	1	4	0	
								1	2	4	0.011		1	2	4	0	

Results indicate that, when unit stockout costs are relatively high, the model tries to avoid stockouts in order to reduce total stockout cost and maximize total profit. In the ample capacity cases with $\beta = 125\%$, this result is apparent where there is a positive optimal total stockout quantity when the unit stockout cost is low whereas no stockouts are made when the unit stockout cost is high. However when $\beta = 50\%$, it is seen that the optimal total stockout quantity for the low stockout costs and the high unit stockout costs cases are the same. Due to the capacity restrictions here, the model cannot further lower the optimal total stockout quantity when the unit stockout costs are higher.

Table 7. Optimal Solution with Respect to Relative Capacities in Multiple Periods

Variable	Stockout Cost	Product Type	Time Period	β Relative Capacities				
				50%	75%	100%	125%	
D_{jkt} Stockout Quantity (units)	Low	1	1	1000	1000	500	125	
			2	0	0	0	0	
			3	0	0	0	0	
			4	0	0	0	0	
		2	1	0	0	0	0	
			2	500	250	0	0	
			3	500	0	0	0	
			4	0	0	0	0	
		Total		2000	1250	500	125	
	High	1	1	1000	750	0	0	
			2	0	0	0	0	
			3	0	0	0	0	
			4	0	0	0	0	
		2	1	0	0	0	0	
			2	500	250	0	0	
			3	500	0	0	0	
			4	0	0	0	0	
		Total		2000	1000	0	0	
P_{jkt} Unit Price (\$/unit)	Low	1	1	29.999700	29.999700	29.999850	29.999888	
			2	89.999550	89.999325	89.999100	89.999100	
			3	64.999837	64.999756	64.999675	64.999594	
			4	44.999888	44.999944	45	45	
		2	1	40	40	39.999800	39.999750	
			2	34.999825	34.999913	35	35	
			3	44.999663	44.999719	44.999775	44.999831	
			4	89.999325	89.999212	89.999100	89.999100	
		High	1	1	29.999700	29.999775	29.999850	29.999888
	2			89.999550	89.999325	89.999550	89.999212	
	3			64.999838	64.999756	64.999675	64.999594	
	4			44.999887	44.999831	44.999775	45	
	2		1	40	39.999900	39.999800	39.999750	
			2	34.999825	34.999913	34.999825	34.999956	
			3	44.999663	44.999719	44.999775	44.999831	
			4	89.999325	89.999437	89.999550	89.999100	
	M_{jkt} Market Share		Low	1	1	1	1	0.5
		2			0.5	0.75	1	1
3		0.25			0.375	0.5	0.625	
4		0.25			0.125	0	0	
2		1		0	0	0.5	0.625	
		2		0.5	0.250	0	0	
		3		0.75	0.625	0.5	0.375	
		4		0.75	0.875	1	1	
High		1		1	1	0.75	0.5	0.375
			2	0.5	0.75	0.5	0.875	
			3	0.25	0.375	0.5	0.625	
			4	0.25	0.375	0.5	0	
		2	1	0	0.25	0.5	0.625	
			2	0.5	0.25	0.5	0.125	
			3	0.75	0.625	0.5	0.375	
			4	0.75	0.625	0.5	1	
		Total Profit (\$)	Low			132,799	186,524	238,048
High			-17,201	110,749	203,999	250,648		
Total Stockout Cost (\$)	Low			10,000	6,250	2,500	625	
	High			160,000	80,000	0	0	

Rowwise comparison in Table 7 shows that as the relative capacity increases, the model decreases the optimal total stockout quantity in order to satisfy more of the market demand which is quite intuitive for both low unit stockout case and high unit stockout case, respectively.

The initial analysis on the optimal unit prices in Table 7 shows that all optimal unit prices are either at their upper bounds or very close to that value for any product type at any period in any relative capacity setting. This result is quite intuitive since in a MSA model, the unit prices only affect the market shares however the total demand is not affected by the increased prices. Hence, in a cooperative competition environment it is always more profitable to increase the prices of both product types very close to their upper bounds so that the optimal total profit is maximized. However, when both prices are at their upper bounds, the market shares will be equal to zero, causing a violation in Constraint (8) where the market shares should add up to 1. So, the optimal solution provides at least one of the optimal unit prices strictly less than its upper bound.

The unit price of a product type and market share of a product type is inversely proportional. The market share analysis in Table 7 shows that, the market shares of all product types add up to 1 in any period in every relative capacity and unit stockout cost setting. It is also observed that when β increases from 100% to 125% for the low stockout costs case, the optimal market share of product type 1 decreases from 0.5 to 0.375 in period 1 whereas the optimal market share of product type 2 increases from 0.5 to 0.675. By doing this, the model allocates more demand to product type 2 which has a higher profit potential than product type 1 in period 1. Same result is observed in the high unit stockout cost case.

Looking at the optimal total profits and optimal total stockout costs in Table 7, it is observed that as the relative capacity increases, optimal total profits increase whereas optimal total stockout costs decrease, which is valid for both cases with low stockout costs and high stockout costs. This result is intuitive since better optimal total profits are expected when β gets larger by relaxing the capacity constraint (2).

Next, the behavior of the optimal production, inventory, transportation, and stockout quantities in multiple periods is aimed to be observed. The network representations show the optimal variables for the given input parameters. As an example, the network representation of the problem when the relative capacity is equal to 75% in the case of low stockout costs and high stockout costs are given respectively in Figure 7 and Figure 8. The green nodes represent the optimal production quantities, the red nodes represent the market demand of each product type, the arcs between the green nodes in sequent periods represent the optimal inventory levels, the arcs from red nodes to green nodes represent the optimal transportation quantities, and the arcs below the red nodes represent the optimal stockout quantities in each period. The red nodes with thick borders represent the periods where each product type has its highest profit potential. In the example problems, it is clear that the first product type has its highest profit potential in period 2, whereas the second product type has its highest profit potential in period 4.

In the network representations, Constraint (4), the demand constraint, and Constraint (6), the inventory balance constraint, is always satisfied so that the inventory level in any period subtracted from the sum of production amount in that period and the inventory level in the previous period is equal to the transportation quantity, and the sum of transportation quantity and stockout quantity is equal to the market demand of any product type in any period.

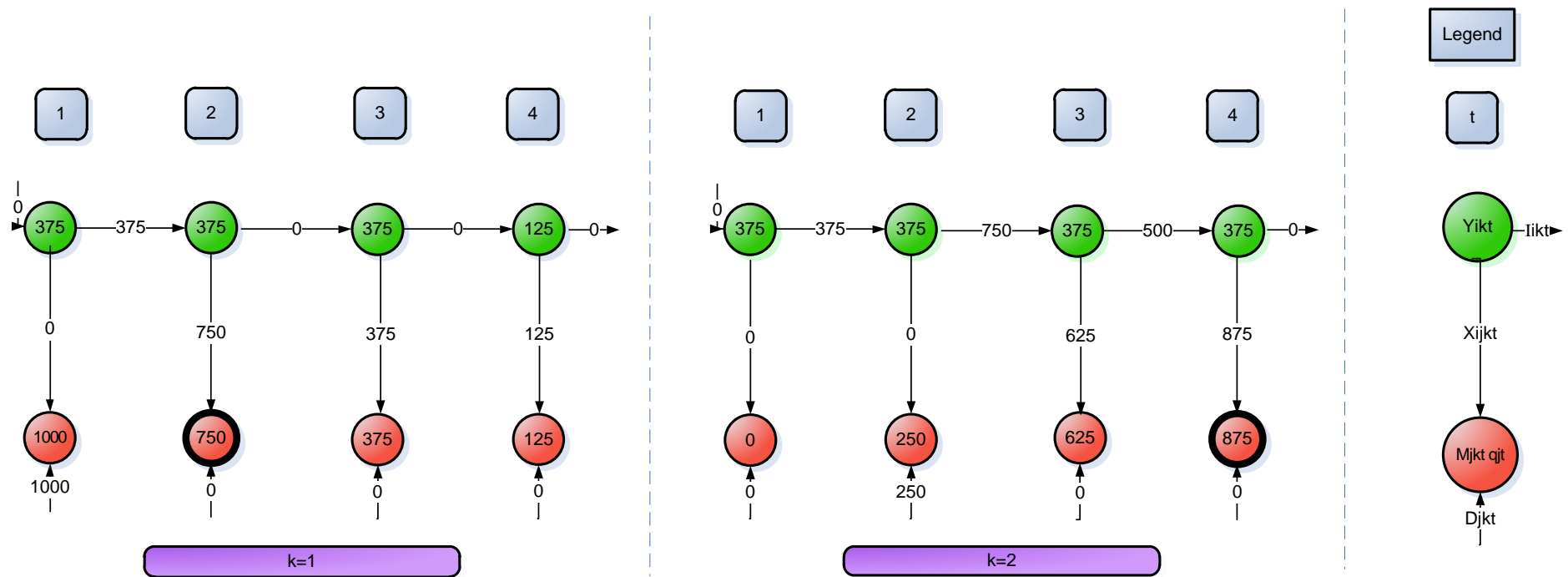


Figure 7. Network representation with low unit stockout costs ($\beta=75\%$)

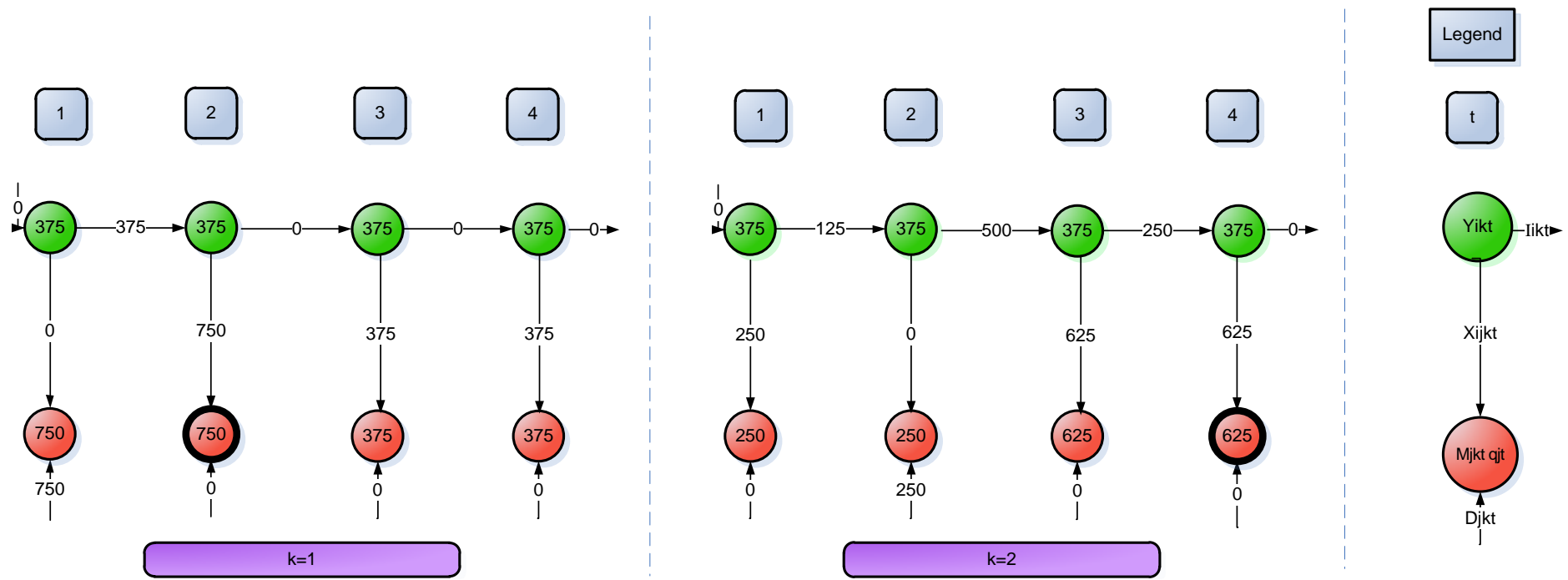


Figure 8. Network representation with high unit stockout costs ($\beta=75\%$)

Comparison between Figure 7 and Figure 8 show that the model tries to allocate the maximum amount of market demand to the product type with higher profit potential in a period. The demand of a period is satisfied by the production in that period and/or the inventory carried from the previous periods. In Figure 7, product type 2 has a demand of 875 units in period 4 which is satisfied by producing 375 units and carrying 500 units of inventory. The rest of the demand in period 4 is 125 units satisfied by product type 1. It can be observed that the optimal demand of product type 1 is lower than its production capacity in period 4 since it has significantly lower profit potential than product type 2 in that period. On the other hand, product type 2 with higher profit potential cannot further increase its sales in period 4 since the production capacity of 375 units puts a restriction on the quantity produced in period 4. Furthermore, if more inventories are carried from the previous periods, then the stockout costs in those periods gets so high that increasing the sales of product type 2 in period 4 is no more profitable. This conclusion is even more apparent in Figure 8, where the optimal solution with higher unit stockout costs is presented. Higher unit stockout costs force product type 2 to lower its sales in period 4 to 625 units, since fewer inventories of 250 units can now be carried from the previous periods to period 4 without further increasing the stockout costs in the previous periods.

The relative capacities in Figure 7 and Figure 8 are 75%. To observe the effect of increasing the relative capacity on the optimal behavior of the model, Figure 9 is introduced with $\beta = 125\%$ for low unit stockout costs case.

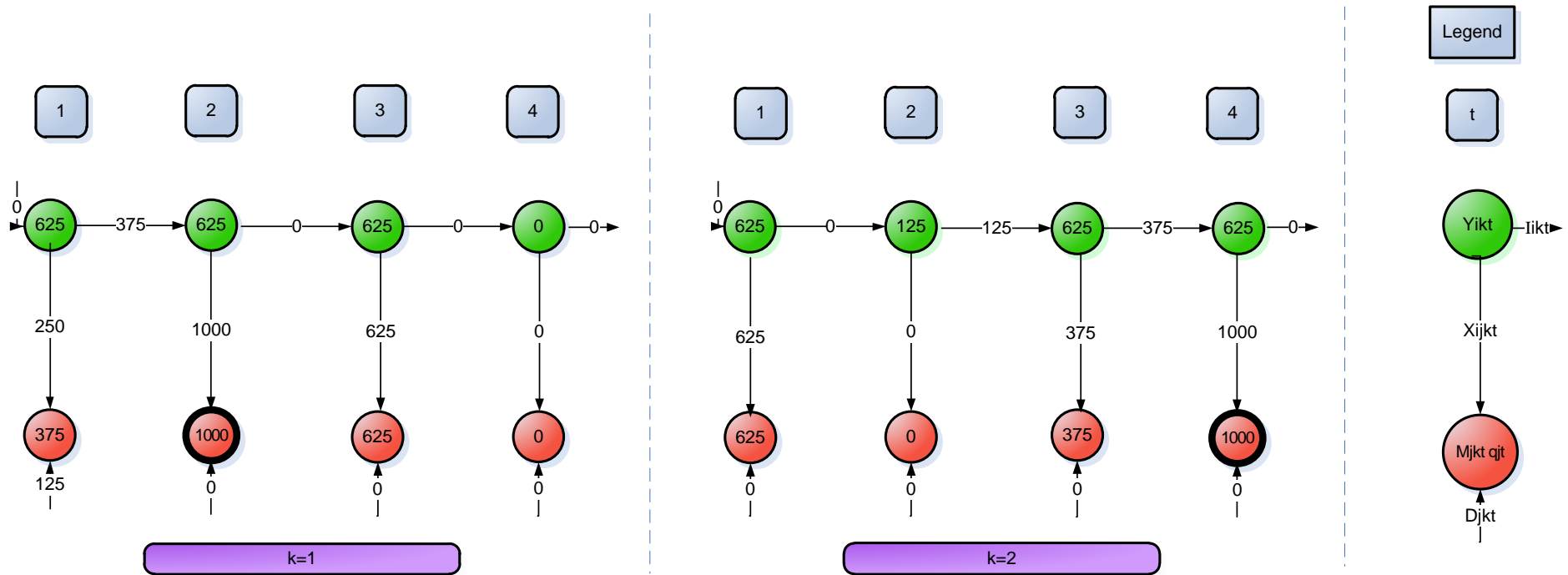


Figure 9. Network representation with low unit stockout costs ($\beta=125\%$)

In Figure 9, it can be seen that similar to the discussion above, the model tries to allocate the maximum market demand to the product type with higher profit potential in a period. Since the available capacities are larger when $\beta = 125\%$, the model can adjust the maximum demand of 1000 units to the product type with highest profit potential. As an example, product type 1 has a demand of 1000 units in period 2 which is satisfied by producing 625 units in that period and carrying 375 units of inventory from the previous period. In period 1, the model prefers not to fully satisfy the demand of product type 1 but chooses to hold some inventory to fully satisfy the demand in period 2, where product type 1 has its highest profit potential among all periods. In comparison between Figure 7 and Figure 9, it is seen that when $\beta = 75\%$ the limited capacity imposes a restriction in the amount that can be transported to the market in period 2. Obviously in Figure 7, the maximum amount of demand that can be satisfied is limited by the sum of production capacities in those two periods which is 750 units. It is interesting to see that in this case, the model decreases the demand of product type 1 in period 2 to 750 units by adjusting the optimal unit prices of product type 1 and product type 2. Surprisingly, the rest of the demand allocated to product type 2 is also left unsatisfied to stockout, so the total stockout costs in period 2 are not affected. Indeed, this allows the model to set a higher unit price for product type 1 in period 2 leading to higher revenue although the resulting total stockout costs in period 2 for both products are unchanged.

In general it is observed that in all cases the model prefers to place the stockouts for any product type in the period with the lowest profit potential. In Figures 7, 8, and 9, it is seen that the model tries to allocate all stockouts for product type 1 to period 1 whereas it chooses to allocate all stockouts for product type 2 to period 2 where they both have their lowest profit potential among all periods.

3.5 An attempt to improve the execution time performance of the model

In this section, a variant of the original model is developed with the aim of decreasing the execution time of the original model. This can be achieved by decreasing the number of constraints and the number of variables.

The original model has two equality equations for the variables A_{jkt} and M_{jkt} in Constraints (4) and (5). Expressing them as functions of the variable P_{jkt} and replacing M_{jkt} with $M_{jkt} = \frac{1-b_{jkt}P_{jkt}}{\sum_k(1-b_{jkt}P_{jkt})}$ enables us to cancel these variables and the related constraints. Although the number of variables and the constraints decrease, the objective function gets a more complex form when M_{jkt} is expressed as a function of P_{jkt} .

To compare the performance of the model suggested above (model 2) with the original model (model 1), ten medium sized problems are generated with random parameters. These problems are again solved with both models in GAMS 23.5.1 with the solver CONOPT version 3.14U. Next, ten large sized problems are generated for each model with random parameters and solved. The resulting execution times and best feasible solutions are tabulated in Table 8.

In the medium sized problem, there are 4 factories, 4 markets, 2 product types and 12 periods, resulting in 864 variables and 480 constraints if model 1 is used and 672 variables and 288 constraints if model 2 is used. In the large sized problem, there are 4 factories, 8 markets, 3 product types and 12 periods, resulting in 2304 variables and 1152 constraints if model 1 is used and 1728 variables and 576 constraints if model 2 is used. Model 2 provides 22.2% decrease in the number of variables and 40% decrease in the number of constraints for the medium sized problem, and it provides 25% decrease in the number of variables and 50% decrease in the number of constraints for the big sized problem.

Table 8. Testing the Models with Ten Different Parameter Sets in Multiple Periods

Problem Size	Problem Number	Model 1		Model 2		Decrease in Execution Time (%)	Increase in Best Feasible Solution (%)
		Execution Time (seconds)	Best Feasible Solution	Execution Time (seconds)	Best Feasible Solution		
Medium Sized Problem (I=4, J=4, K=2, T=12)	1	4.33	1,780,486	3.13	1,544,596	28	-13
	2	3.49	1,391,302	2.73	1,293,708	22	-7
	3	4.49	995,693	3.00	964,313	33	-3
	4	3.92	1,798,978	2.86	1,644,531	27	-9
	5	3.39	1,711,169	3.58*	1,629,289*	-6	-5
	6	4.62	1,797,063	2.54	1,703,437	45	-5
	7	3.16	1,579,504	3.14	1,391,183	1	-12
	8	3.01	2,684,915	3.81	2,494,108	-27	-7
	9	3.55	1,072,590	3.13	981,262	12	-9
	10	4.32	1,957,887	3.73	1,813,668	14	-7
	Average	3.83	1,676,959	3.16	1,546,010	14.87	-7.69
Large Sized Problem (I=4, J=8, K=3, T=12)	1	12.24	3,722,226	16.93	2,632,652	-38	-29
	2	10.50	2,867,367	30.03	2,867,367	-186	0
	3	10.80	2,070,450	24.67*	2,070,451*	-128	0
	4	13.61	3,110,045	14.17*	2,049,683*	-4	-34
	5	13.14	3,425,702	20.16	3,425,213	-53	0
	6	15.69	3,243,401	21.16	3,152,342	-35	-3
	7	12.09	3,228,107	27.56	3,228,107	-128	0
	8	13.45	5,358,474	27.68	5,293,991	-106	-1
	9	9.54	2,234,316	29.98	2,234,316	-214	0
	10	8.11	4,029,625	17.69	3,009,273	-118	-25
	Average	11.92	3,328,971	23.00	2,996,340	-101.13	-9.27

Note: * indicates a feasible solution rather than optimal

The results in Table 8 show that for medium sized problems, the average execution time decreases from 3.83 seconds to 3.16 seconds when the suggested model 2 is used. This results in an average improvement of 14.87% in execution time. It should be noted that in two of the ten medium sized problems, the execution time gets 6% - 27% worse in the proposed model. However, it should be noted that the optimal solutions found by the suggested model are 7.69% worse than the original model for the medium sized problems.

Contrary to the findings for the medium sized problems, in the large sized problems the execution time of the suggested model 2 is 101.13% longer than the

original model in average. It should be noted that in six of the ten large sized problems, the execution time of model 2 reaches 2-3 times the execution time of model 1. On the other hand, the optimal solution performance of model 2 is 9.27% worse than the optimal solution performance of model 1 when the problem size is large.

Model 1 and model 2 can also be compared respectively by how their execution time increases when the size of the problem is large. It is interesting to see that the average execution time of large sized problems is 3.00 times the average execution time of medium sized problems when model 1 is used. However, when model 2 is used, the average execution time of large sized problems is 7.28 times the average execution time of medium sized problems.

Clearly, it can be concluded that the execution time performance of the suggested model slightly better than the original model when the size of the problem is not large. However, for the large sized problems, the suggested model significantly fails to provide a better execution time performance.

CHAPTER 4

DEVELOPMENT OF A DECISION SUPPORT SYSTEM FOR THE INTEGRATED MODEL

A decision support system (DSS) is a model-based or knowledge-based system that supports managerial decision making, as explained in Pol and Ahuja (2007). It uses data, provides a clear user interface, and can incorporate the decision maker's own insights. A DSS application contains five components: database, model base, knowledge base, GUI, and user. The database and the knowledge base can be found in a basic information system. A decision support system is an intelligent information system because of the addition of the model base. The model base has the models used to perform optimization, simulation, or other algorithms for advanced calculations and analysis. These models allow the DSS to not only supply information to the user but aid the user in making a decision.

In this section a DSS for optimizing the integrated decisions for production, transportation and pricing that will be referred to as PTP-DSS is developed.

4.1 DSS framework and the information flow

In the proposed DSS, a database where the data is stored in SQL Server is considered. In the database, there is information about the problem, i.e. product types, market demands, factory capacities, costs. Secondly, the model base contains a NLP model for optimization which is used to solve the integrated production-transportation and pricing problem running GAMS software.

The GUI, built on Visual Basic, allows the user to enter data or update data, run the chosen model, view the results of the model, and run the application several

times with different data. The GUI includes a welcome screen, view inputs screen, view outputs screen, and what-if analysis screen and it is aimed to be flexible and easy to use. The users of the system are both managers and pricing specialists. The schematic view of the DSS can be seen in Figure 10.

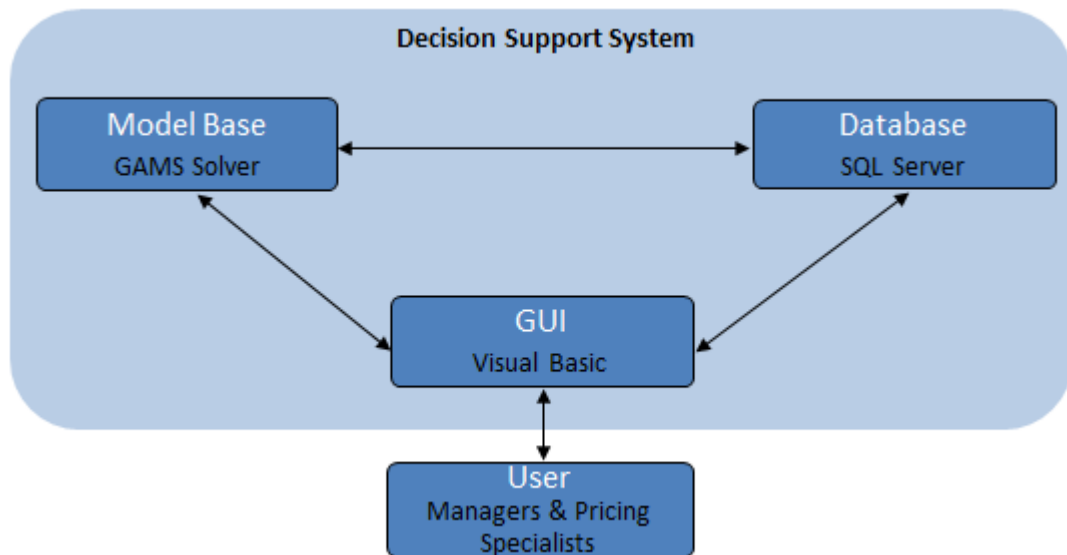


Figure 10. Schematic view of the DSS for the integrated model

A DSS framework is conceptualized that consists of interacting user, user interface, DSS database and NLP model as shown in Figure 11. The process starts with the interacting user viewing input values in the user interface, which are retrieved from the DSS database. When the interacting user gives the command to run the selected model, input values are initialized into parameters and they are sent to GAMS solver. Then, output variables that include the optimal solution are sent to the DSS database, which can be viewed from the user interface as output tables and graphs. What-if input values are entered in the user interface when the interacting user wants to temporarily change input parameters for what-if analysis, and these parameters are sent to temporary tables in the database. The user gives the command

to run the selected model, the optimal values of the output variables are sent to temporary data tables in the database which can be viewed in the user interface as output tables and graphs, and the process ends by temporary tables being cleared.

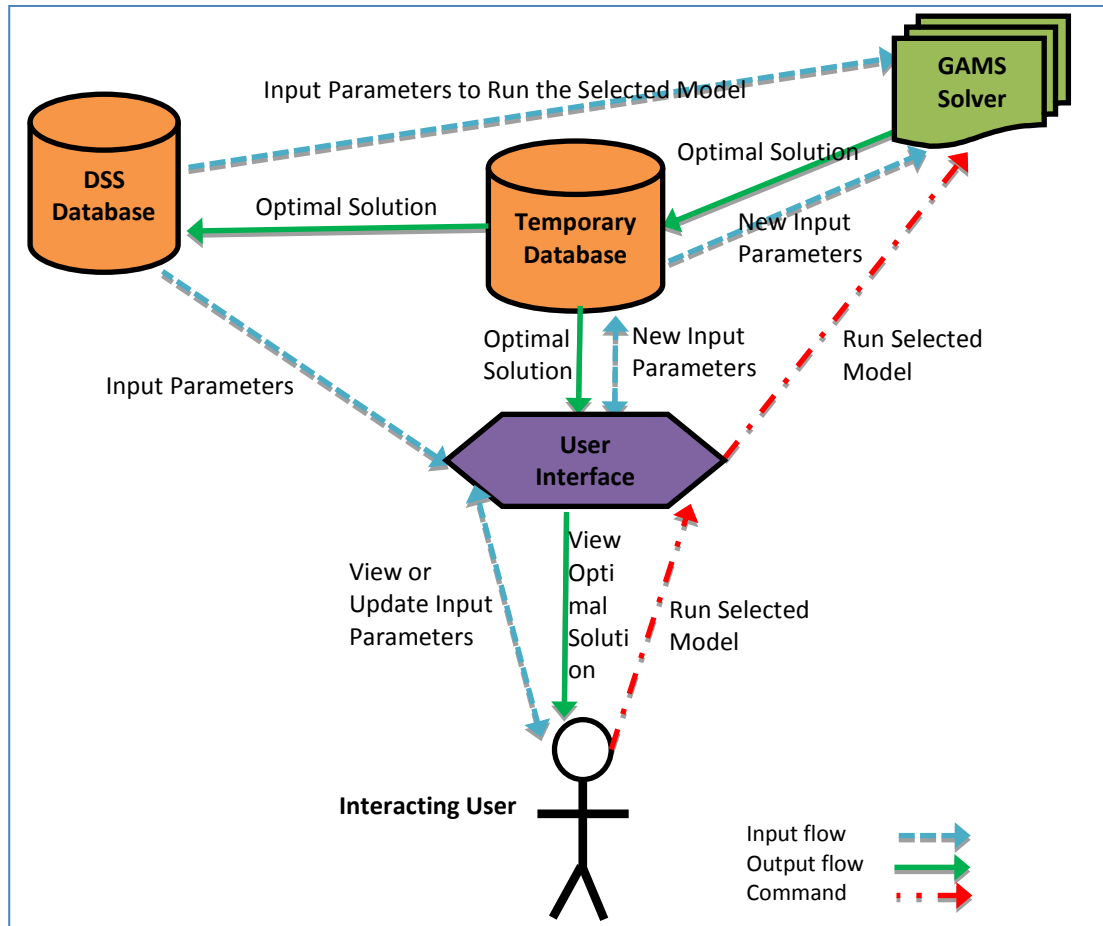


Figure 11. Information flow in the DSS

4.2 DSS database

Database is an important component of the DSS providing secure, robust, dynamic and systematic data to the DSS. A relational database diagram is a powerful tool to illustrate the relations between tables of the database. In the DSS framework, the inputs cannot be updated from the DSS as the system is designed in such way that

the inputs already exist in the database. Input data entries and updates are handled externally by database managers which is not in the scope of this DSS.

An SQL Server database is developed containing four index tables that are Factory, Market, Product, and Period. The products are produced in factories, the produced goods are shipped to markets, and they are sold in markets in given time periods. Input and output tables are designed separately to prevent data loss in cases there is an error while running the SQL queries. The relational database diagram for the DSS is shown in Figure 12 and Figure 13 for storing the optimal values of the output variables and input parameters, respectively.

All the columns in the index tables are stored in data type varchar with 50 characters length. The reason of choosing varchar(50) as the data type for indices is that it allows alphanumeric characters whereas float is used for input parameters and output variables as it is more appropriate as these columns contain numerical data. In the Factory table, there are two columns, factory index as the primary key column, and factory location column. In the Market table, there are two columns, market index as primary key column, and market location column. In the Product table, there are two columns, product type index as primary key column, and product type column. Period table contains only period index as primary key.

After the index tables are created, input parameter tables are created with the aid of SQL Cross Join statements. The reason for using Cross Join is that the GAMS solver needs the input parameters as Cartesian product of index rows, i.e. initial inventory parameter is controlled by factory and period indices, cross joining the Factory table and Period table results in the join table named FactoryProduct. With Alter Table Add statement, a new column is inserted into the join table FactoryProduct named initial inventory with data type float. The other join tables are

created in the same manner, so in the database there are FactoryProduct, MarketPeriod, FactoryProductPeriod, MarketProductPeriod and FactoryMarketProductPeriod join tables containing the necessary columns with data type float for the input parameters. The indices are set as primary keys of join tables to prevent duplicate entries, and they also act as foreign keys retrieved from the index tables.

The output tables are designed to contain the necessary outputs being retrieved from GAMS solver as seen in Figure 12. They include Objective, FactoryProductPeriodR, MarketProductPeriodR, and FactoryMarketProductPeriodR tables built in the same manner with input parameters. Here, the only difference is that in these output tables, the indices do not act as foreign keys of the index table columns because these tables will be updated to include the output variables obtained from the solution, so the output tables should be blank before the solution is completed.

When the output tables are created, a sample data set is generated in Excel using the randbetween function that generates random integers between the given intervals. Four factories (Gebze, Izmir, Bursa, Manisa), four markets (Aegean, Blacksea, Mediterranean, Central Anatolia), two product types (Basic and Premium) and twelve time periods (designed as months starting from January 2015 to December 2015) are considered in the sample data set. The sample data set is imported into the SQL Server database. After importing the data into the database, the sample data is ready as inputs for the solver. The input parameter tables of the sample data contains $4 \times 12 = 48$ rows in the FactoryProduct table and MarketPeriod table, $4 \times 2 \times 12 = 96$ rows in the FactoryProductPeriod table and MarketProductPeriod

table, $4 \times 4 \times 2 \times 12 = 384$ rows in the FactoryMarketProductPeriod table following the logic of Cross Join statements as Cartesian products.

All the relationships between the entities are many-to-many relationships, since many product types can be produced in many factories, many product types can be sold in many markets, and many factories can ship the finished goods to many markets.

Objective	
<input type="checkbox"/>	Total_Profit
<input type="checkbox"/>	Total_Revenue
<input type="checkbox"/>	Total_Prod_Cost
<input type="checkbox"/>	Total_Trans_Cost
<input type="checkbox"/>	Total_Inv_Cost
<input type="checkbox"/>	Total_Stockout_Cost
<input type="checkbox"/>	Total_Cost

FactoryProductPeriodR	
<input checked="" type="checkbox"/>	Factory_Index
<input checked="" type="checkbox"/>	Product_Index
<input checked="" type="checkbox"/>	Period_Index
<input type="checkbox"/>	Production_Qty
<input type="checkbox"/>	Inventory_Level

MarketProductPeriodR	
<input checked="" type="checkbox"/>	Market_Index
<input checked="" type="checkbox"/>	Product_Index
<input checked="" type="checkbox"/>	Period_Index
<input type="checkbox"/>	Price
<input type="checkbox"/>	Attraction
<input type="checkbox"/>	Market_Share
<input type="checkbox"/>	Stockout_Qty

FactoryMarketProductPeriodR	
<input checked="" type="checkbox"/>	Factory_Index
<input checked="" type="checkbox"/>	Market_Index
<input checked="" type="checkbox"/>	Product_Index
<input checked="" type="checkbox"/>	Period_Index
<input type="checkbox"/>	Trans_Qty

Figure 12. The relational database diagram for the output parameters of the DSS

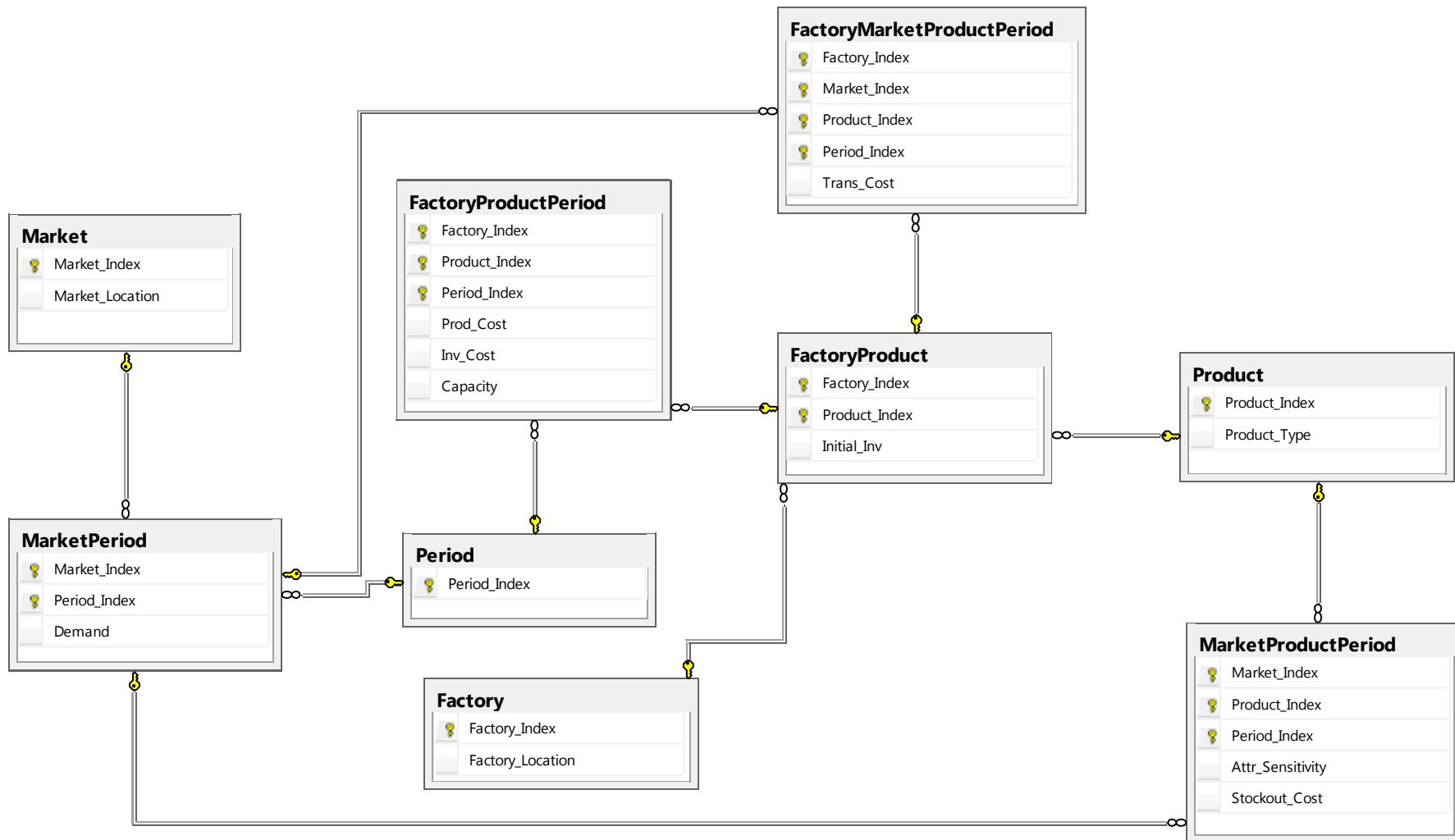


Figure 13. The relational database diagram for the input parameters of the DSS

4.3 DSS development

There are six basic steps for developing a DSS as explained in Şeref, Ahuja, and Winston (2007):

1. Application Overview
2. Worksheets
3. User Interface
4. Procedures
5. Resolve Options
6. Testing and Final Packaging

Application overview is considered as the most important step in DSS development according to Şeref et al. (2007). The aim is to prepare a brief summary of the DSS to check out the flow from the beginning to the end. The flow begins with the welcome sheet, and continues with input, model and calculations, output, and it ends with re-solve options. The welcome sheet describes the DSS and gives introductions on the usage of the DSS. Type of input can be sheets, forms, or input box depending on the properties of the DSS. The DSS may include an optimization model, a simulation model, or other algorithms. Formulations and functions can be calculated using Excel or they can be calculated dynamically in a coding environment. Type of output varies from charts, graphs to histograms, tables focusing on the needs of the users of the DSS. Here, it is decided whether to redisplay some inputs and the adequacy of the output. In resolve options, the input may be modified, and the constraints or objectives can be redefined. The worksheet user interface includes navigational buttons, functional buttons, controls on the worksheet, and user forms. The outline of the procedures in the application should include a main procedure, a clear previous procedure, a procedure to receive input,

some procedures and functions to perform calculations, and a procedure to generate output. In designing the re-solve options, the developers should check that the users are able to modify inputs, calculation options, constraints, and objectives; recalling that the aim of the DSS is to aid the decision-makers in making the best decision. The testing and final packaging step ensures that the DSS application works correctly and has a professional appearance for the end users.

The main expectation of the interacting user from a DSS is a functional, well organized structure that can be achieved by a holistic approach. Considering the needs of the interacting users, the main forms of the DSS are conceptualized as:

1. Welcome Screen
2. View Parameters Form
3. View Optimal Plan Form
4. What-if Analysis Edit Input Parameters Form
5. What-if Analysis View Optimal Plan Form

The welcome screen in Figure 14 is the initial screen of the DSS that welcomes the user. The processes within the DSS are briefly explained as follows: There are two functional buttons, Start button and Exit button. When the interacting user clicks on the Start button, the DSS application brings up the view problem parameters form. If the Exit button is clicked, the user exits from the DSS application. The Exit button is placed in every main form so that the interacting user can stop the process anytime. There are three images which represent production, transportation and pricing.

After clicking the Start button in the welcome screen, the user is directed to the next screen where the user can view parameters, view optimal plan and perform what-if analysis that will be explained respectively.

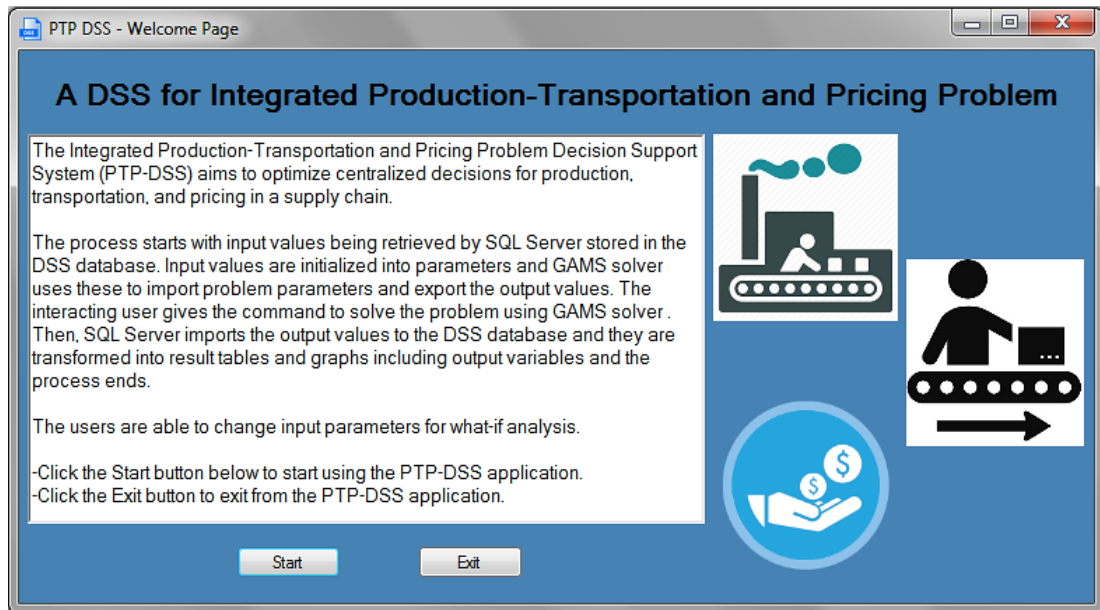


Figure 14. PTP-DSS welcome page

4.3.1 View parameters

The following form is view parameters form where the interacting user is able to view problem parameters and solve the problem. A network graph is drawn by using MSAGL (Microsoft Automatic Graph Layout) which is a .NET tool for graph layout and viewing.¹ The network graph contains the factories and markets as nodes, and contains arcs from each factory to each market representing a transportation network. This generic view of the transportation network is implemented in order to provide a brief explanation of the network to the interacting user. The view parameters form is shown in Figure 15.

On the view parameters form, the interacting user is able to view the list of factories, markets, product types, and time periods by clicking on the related button. The list of factories and the list of markets can be seen in Figure 16, the list of product types and the list of time periods can be seen in Figure 17.

¹ <http://research.microsoft.com/en-us/projects/msagl/>

PTP DSS - View Parameters

View Parameters

Click the related button to view the parameters.

Unit Production Cost (\$/unit)

Unit Inventory Holding Cost (\$/unit)

Factory Capacity (units/month)

Initial Inventory Level (units/month)

Attraction Sensitivity

Unit Stockout Cost (\$/unit)

Market Demand (units/month)

Unit Transportation Cost (\$/unit)

Product Types

Time Periods

Factories

M₁

M₂

M₃

M_J

Markets

Diagram showing connections between factories (F₁, F₂, F₃, F_I) and markets (M₁, M₂, M₃, M_J).

View Optimal Plan

Click the Solve button to solve the problem and view the optimal plan.

Production Quantity (units/month)

Inventory Level (units/month)

Unit Price (\$/unit)

Attraction

Market Share (%)

Stockout Quantity (units/month)

Transportation Quantity (units/month)

Total Profit (\$)

Total Revenue (\$)

Total Production Cost (\$)

Total Transportation Cost (\$)

Total Inventory Holding Cost (\$)

Total Stockout Cost (\$)

Total Cost (\$)

Factories

M₁

M₂

M₃

M_J

Markets

Diagram showing connections between factories (F₁, F₂, F₃, F_I) and markets (M₁, M₂, M₃, M_J).

Back
Solve
Exit

Figure 15. View parameters form

Factory Index	Factory Location
F1	Gebze
F2	Izmir
F3	Bursa
F4	Manisa

Market Index	Market Location
M1	Aegean
M2	Blacksea
M3	Mediterranean
M4	Central Anatolia

Figure 16. List of factories screen and list of markets screen

Product Type Index	Product Type
P1	Basic
P2	Premium

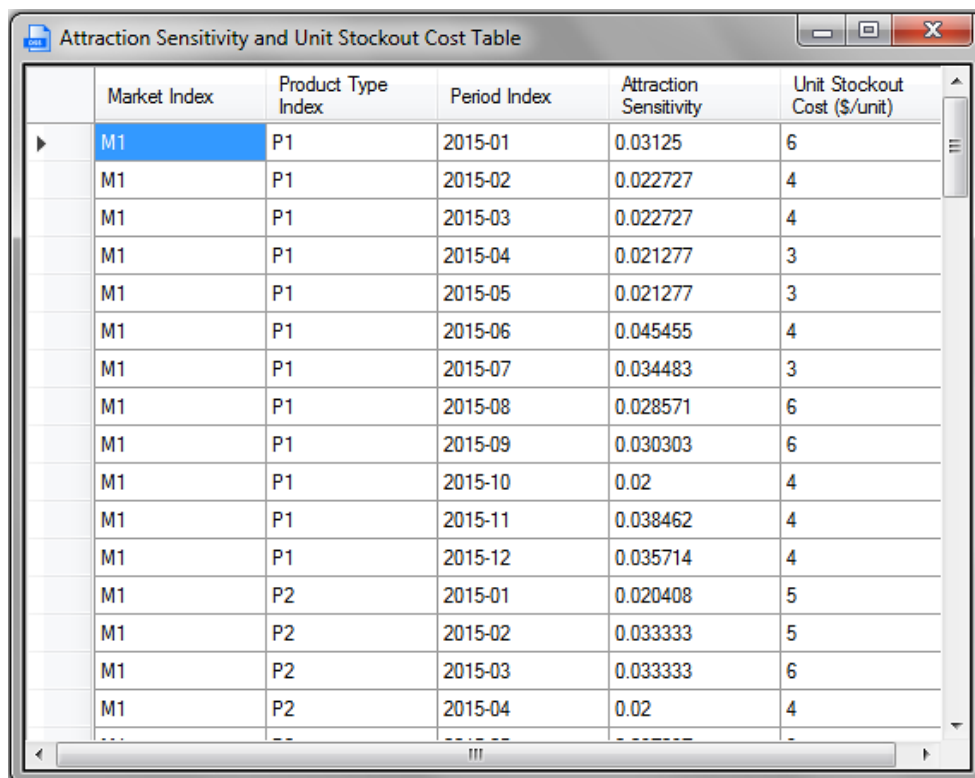
Period_Index
2015-01
2015-02
2015-03
2015-04
2015-05
2015-06
2015-07
2015-08
2015-09
2015-10
2015-11
2015-12

Figure 17. List of product types screen and list of time periods screen

To see the table of any parameter, the interacting user can click on the related buttons which have different colors representing the index or indices that the related parameters depend on i.e. market demand parameter is controlled by market index and time period index. The parameters that depend on the same indices are grouped into the same button so that the table of related parameters contains all data for the given parameters i.e. by clicking on the Attraction Sensitivity - Unit Stockout Cost (\$/unit) button, Attraction Sensitivity and Unit Stockout Cost table can be viewed as shown in Figure 18. The rest of the parameter tables including unit production costs,

unit inventory holding costs, factory capacities, initial inventory levels, and unit transportation costs can be found in Appendix B.

The optimal plan is not obtained yet as the model has not been solved, so this section is not highlighted in this screen and the buttons on this section are not functional. In Figure 15, Back button provides the ability to navigate to the previous screen when pressed.



Market Index	Product Type Index	Period Index	Attraction Sensitivity	Unit Stockout Cost (\$/unit)
M1	P1	2015-01	0.03125	6
M1	P1	2015-02	0.022727	4
M1	P1	2015-03	0.022727	4
M1	P1	2015-04	0.021277	3
M1	P1	2015-05	0.021277	3
M1	P1	2015-06	0.045455	4
M1	P1	2015-07	0.034483	3
M1	P1	2015-08	0.028571	6
M1	P1	2015-09	0.030303	6
M1	P1	2015-10	0.02	4
M1	P1	2015-11	0.038462	4
M1	P1	2015-12	0.035714	4
M1	P2	2015-01	0.020408	5
M1	P2	2015-02	0.033333	5
M1	P2	2015-03	0.033333	6
M1	P2	2015-04	0.02	4

Figure 18. Attraction sensitivity and unit stockout cost table screen

The Solve button is used to solve the problem with input parameters provided from the SQL database. When the Solve button is pressed, a message box is viewed on the screen indicating “Click the Ok button to solve the problem”. After pressing the Ok button, GAMS is called as an external process running in the background which is provided by the code `ProcessWindowStyle.Hidden` in Visual Basic. Also, in

order to wait for the solver to finish before viewing results, the code

`Process.Start.WaitForExit()` is implemented in the VB code.

A tool named SQL2GMS is used to convert data from an SQL database into a GAMS readable format, which can be called within GAMS by including the relevant line of code for the indices and parameters². The connection string is needed to establish a connection, and the initial catalog indicates the name of the database in SQL server. The SQL query `SELECT FROM` is used to read the data from the SQL database, i.e. the query to read the factory index is “`SELECT DISTINCT Factory_Index FROM Factory`”. When the solving process ends, a message box tells the user “Problem is solved”, and when the Ok button is pressed, View Optimal Plan form is viewed. The solver message boxes can be seen in Figure 19.

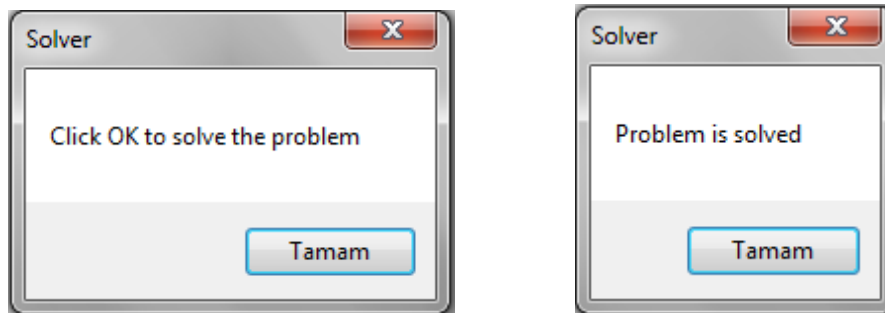


Figure 19. Solver message box

When the info button is clicked, problem information is shown including the NLP model and the indices of the model, model parameters, variables, equations and constraints. Each constraint is designed in such way that when the interacting user hovers on each equation and constraint, a tooltip is viewed explaining the equation or constraint. By clicking on the Close button, the screen can be closed anytime. The problem information screen is given in Figure 20.

² <https://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Ftools%2Fsql2gms%2Findex.html>

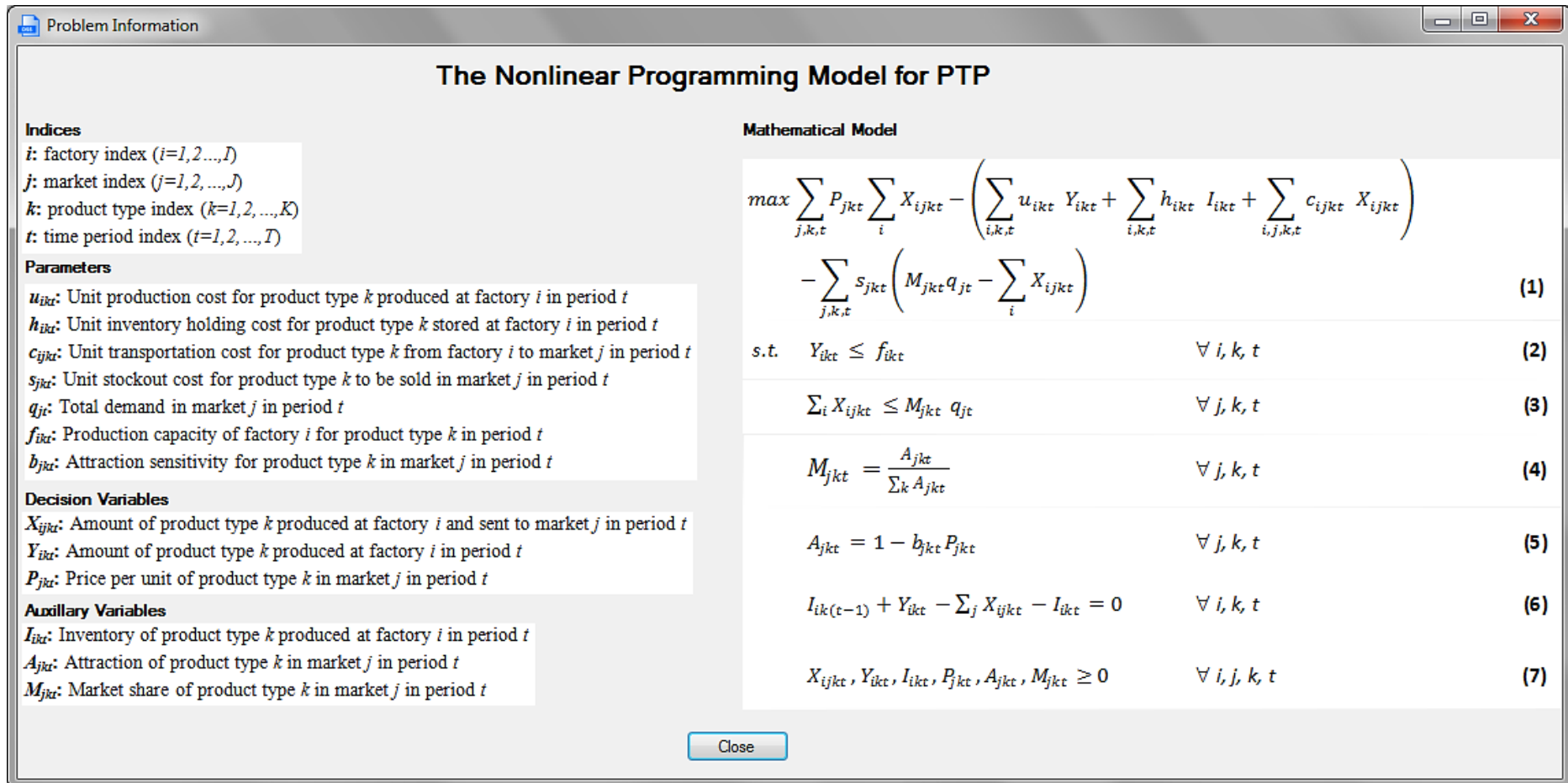


Figure 20. Problem information screen

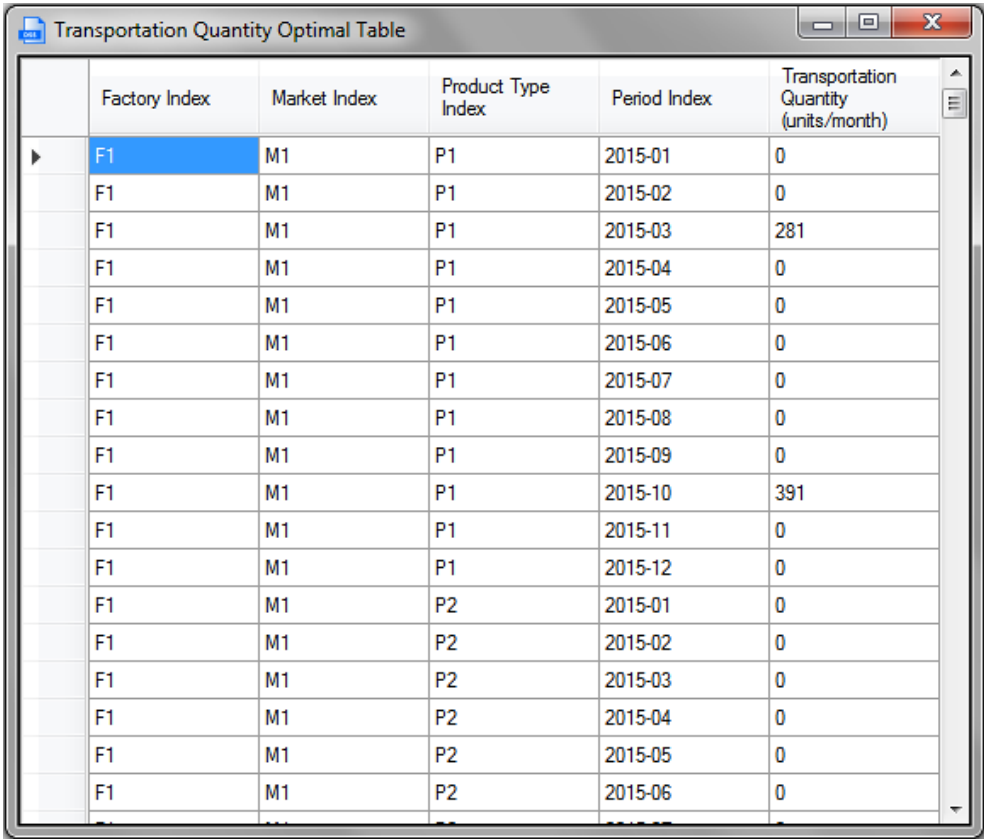
4.3.2 View optimal plan

The next process to perform in view input parameters form is to solve the NLP. View optimal plan form is shown after the solving phase is completed, as shown in Figure 22. As the solution process ends, output values should be transferred from GAMS to the SQL database. This is achieved by GDXViewer, a tool to view and convert data contained in GDX files which are generated within GAMS and they contain problem indices, parameters, and variables.³ Using GDXViewer, the output variables are saved in CSV files as comma-separated values. Temporary data tables are created with the related columns, and using SQL Bulk Insert statements, the CSV files are imported into these temporary data tables. Then, using Update statements, the actual result tables are updated, and the temporary tables are dropped from the database using Drop Table command. The outputs are desired to be viewed in three decimal places which is achieved by using the Round function, i.e. for the production quantity variable to be viewed in three decimal places, the syntax `Select Round (Production_Qty,3)` is used. For the market shares, results are expected to be viewed in percentages instead of decimals requiring the syntax `Market_Share=Convert (varchar(50), 100.00*Market_Share)`. Total Cost must be calculated as the sum of total production cost, total transportation cost, total inventory holding cost, and total stockout cost by using the Sum function within the Select statement. The full source code of the form with the relevant SQL queries is included in Appendix D.

When the problem is solved in GAMS, the form layout is kept unchanged, the problem parameters can still be viewed but the optimal plan is highlighted now because the model has been solved. The decision variables of the NLP are grouped under the related buttons in the same way as described in the view parameters form.

³ <https://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Ftools%2Fgdxviewer%2Findex.html>

The variables that depend on the same indices are grouped into the same button to provide efficiency in the view tables for the optimal solutions. Optimal solutions can be viewed either in table or graph format. The button with the data table icon brings up the table of variables, i.e. by clicking on the data table button next to Transportation Quantity (units/month) button; Optimal Transportation Quantity table can be viewed as shown in Figure 21. The rest of the tables for optimal production quantities, optimal inventory levels, optimal unit prices, optimal attractions, optimal market shares, and optimal stockout quantities can be found in Appendix C.



	Factory Index	Market Index	Product Type Index	Period Index	Transportation Quantity (units/month)
▶	F1	M1	P1	2015-01	0
	F1	M1	P1	2015-02	0
	F1	M1	P1	2015-03	281
	F1	M1	P1	2015-04	0
	F1	M1	P1	2015-05	0
	F1	M1	P1	2015-06	0
	F1	M1	P1	2015-07	0
	F1	M1	P1	2015-08	0
	F1	M1	P1	2015-09	0
	F1	M1	P1	2015-10	391
	F1	M1	P1	2015-11	0
	F1	M1	P1	2015-12	0
	F1	M1	P2	2015-01	0
	F1	M1	P2	2015-02	0
	F1	M1	P2	2015-03	0
	F1	M1	P2	2015-04	0
	F1	M1	P2	2015-05	0
	F1	M1	P2	2015-06	0

Figure 21. Optimal transportation quantity table screen

The button with the graph icon brings up the related graph screens to plot graphs by selecting the related indices from the dropdown menus and clicking on the View Graph buttons.

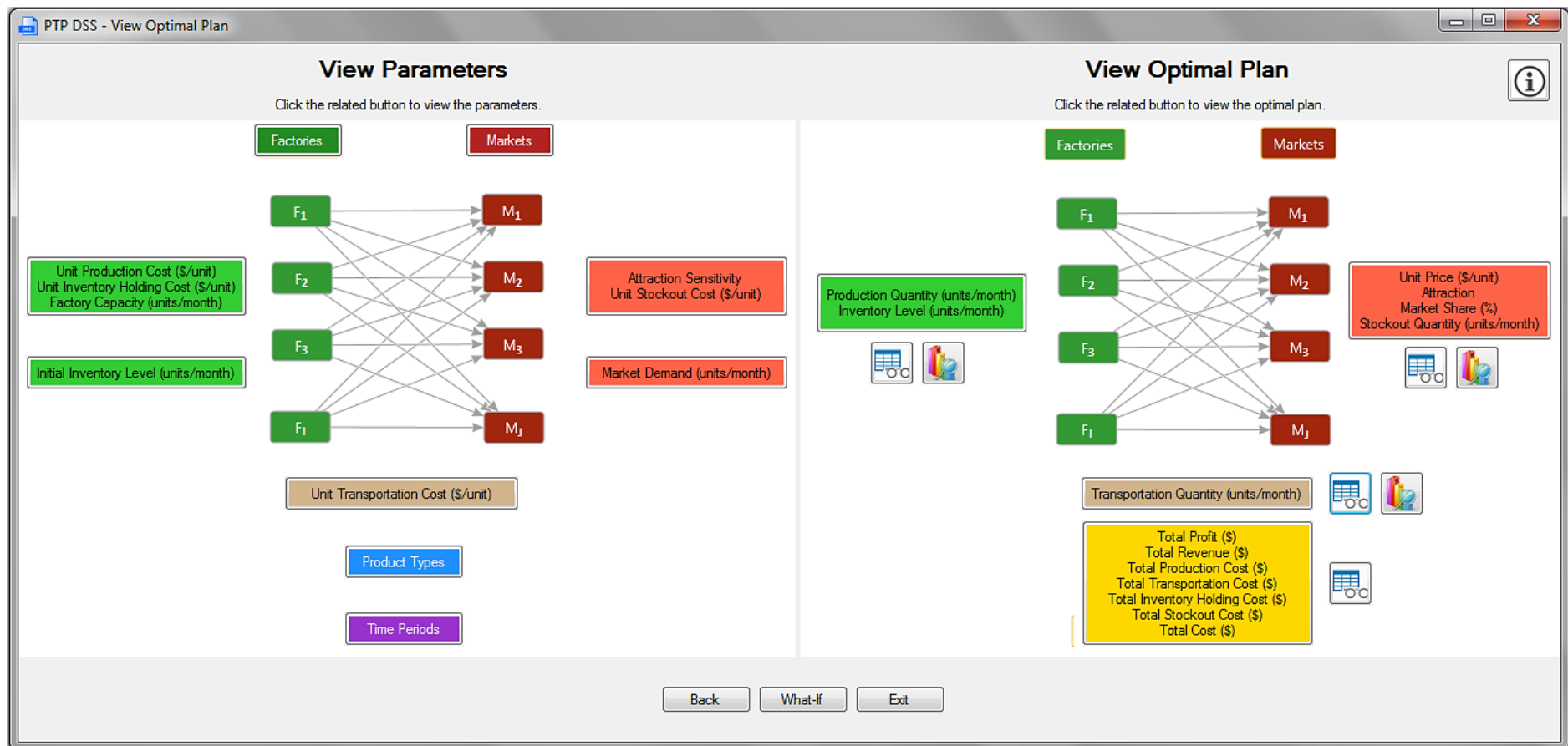
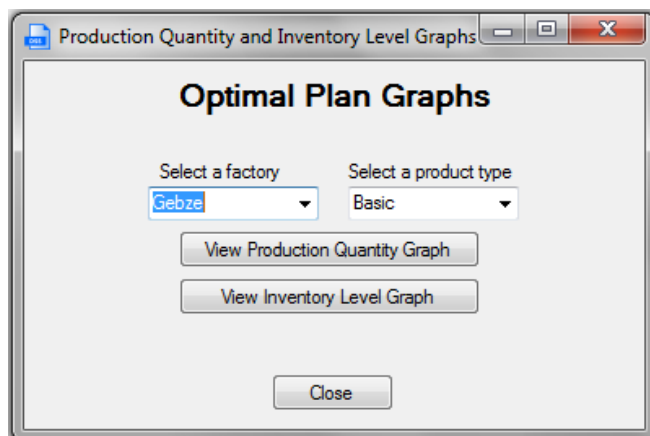


Figure 22. View optimal plan form

On the production quantity and inventory level graph screen which is given in Figure 23, the interacting user selects the factory and product type. Clicking on View Production Quantity Graph button gives optimal production quantities which are plotted for all time periods for the selected factory and product type, i.e. if the selected factory is Gebze and the product type is basic, then the resulting graph is plotted as seen in Figure 24.



Production Quantity and Inventory Level Graphs

Optimal Plan Graphs

Select a factory: Gebze

Select a product type: Basic

View Production Quantity Graph

View Inventory Level Graph

Close

Figure 23. Production quantity and inventory level graph screen

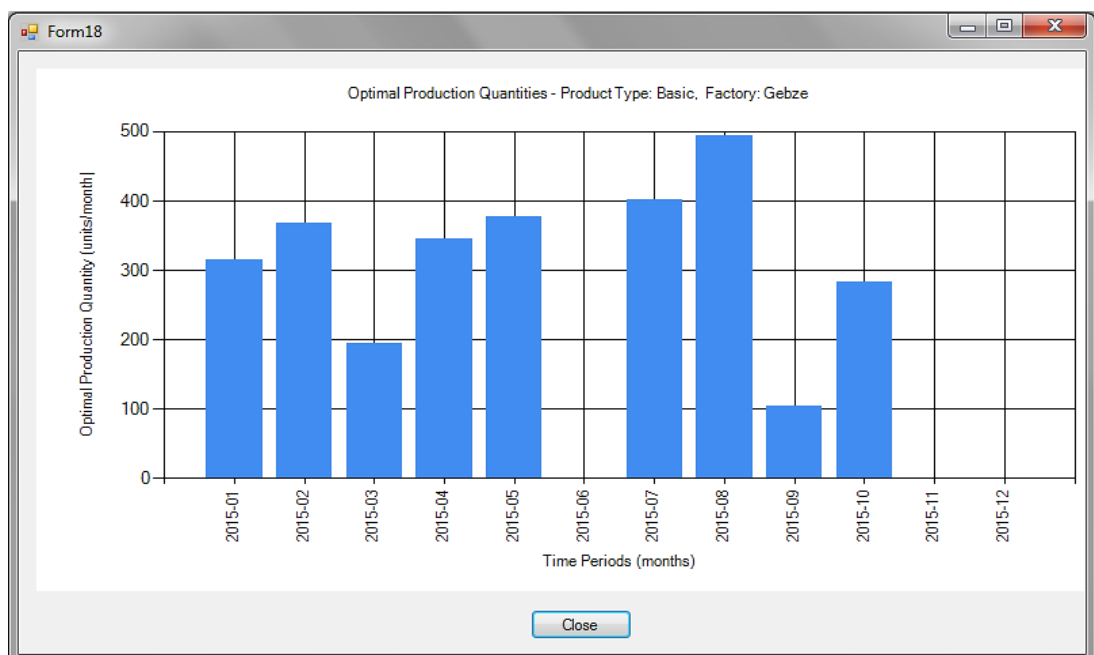


Figure 24. Optimal production quantity sample graph

Clicking on View Inventory Level Graph button gives optimal inventory levels which are plotted for all time periods for the selected factory and product type, i.e. if the selected factory is İzmir and the product type is basic, then the resulting graph is plotted as seen in Figure 25.

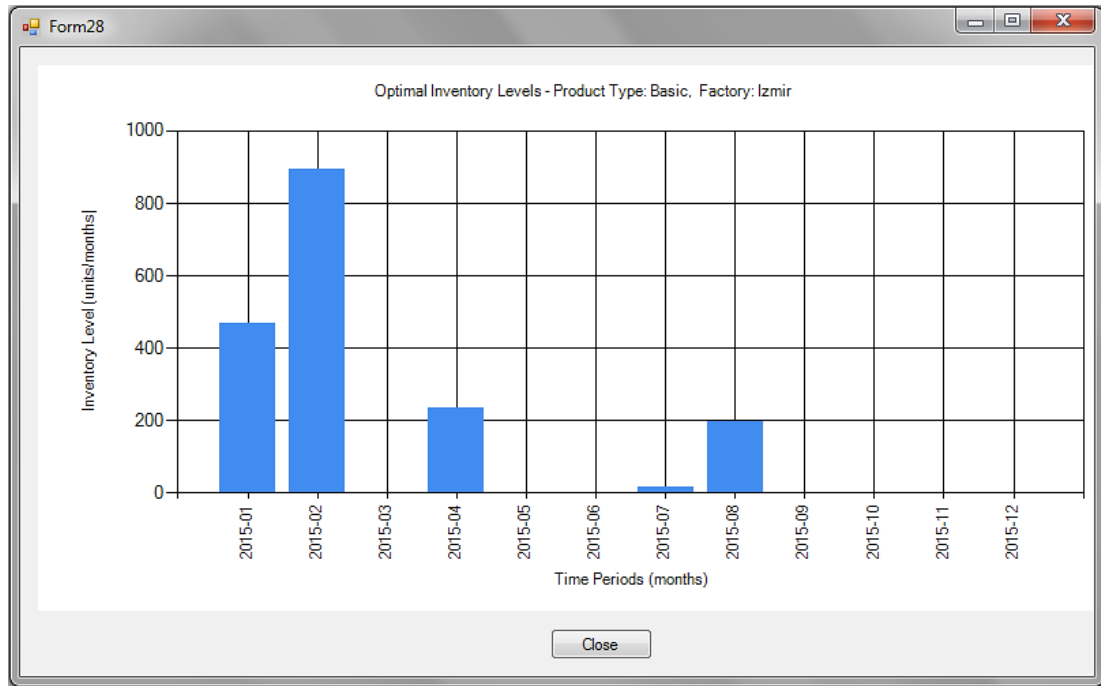


Figure 25. Optimal inventory level sample graph

On the unit price and market share graph screen which is given in Figure 26, the interacting user selects the market and product type. Clicking on View Unit Price Graph button gives optimal unit prices which are plotted for all time periods for the selected market and product type, i.e. if the selected market is Aegean and the product type is basic, then the resulting graph is plotted as seen in Figure 27.

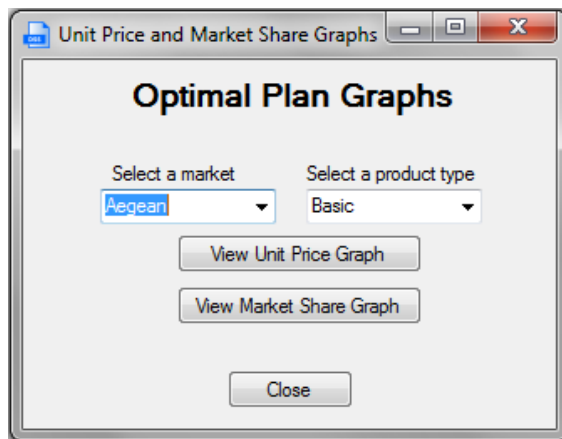


Figure 26. Unit price and market share graph screen

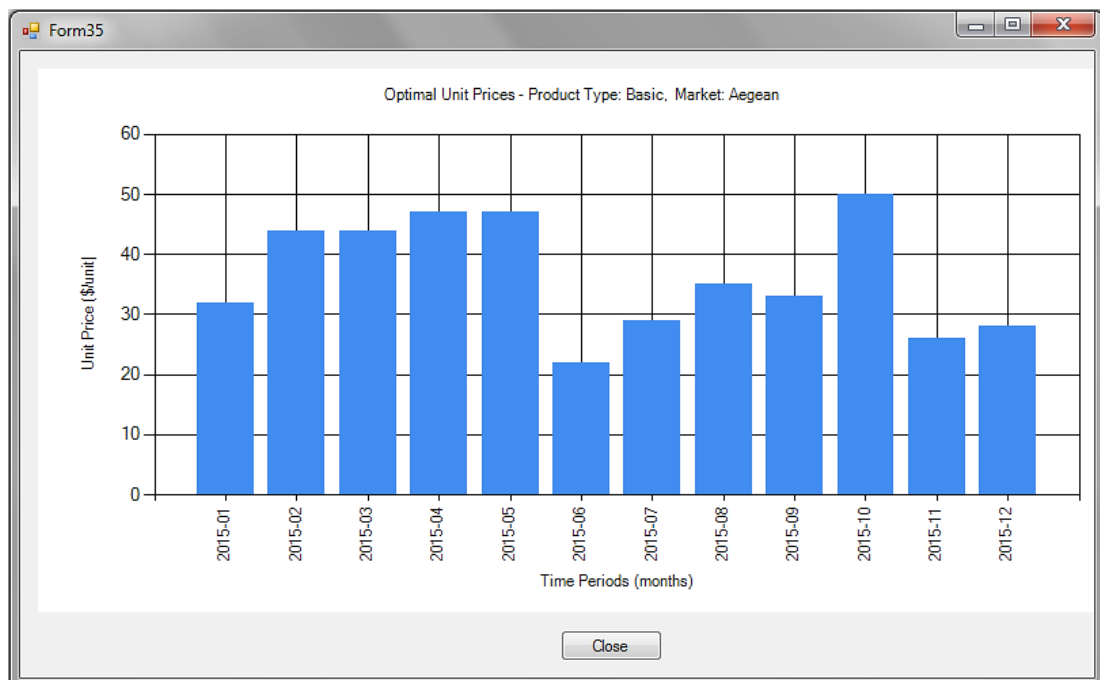


Figure 27. Optimal unit price sample graph

Clicking on View Market Share Graph button gives optimal market shares as percentages which are plotted for all time periods for the selected market and product type, i.e. if the selected market is Blacksea and the product type is premium, then the resulting graph is plotted as seen in Figure 28.

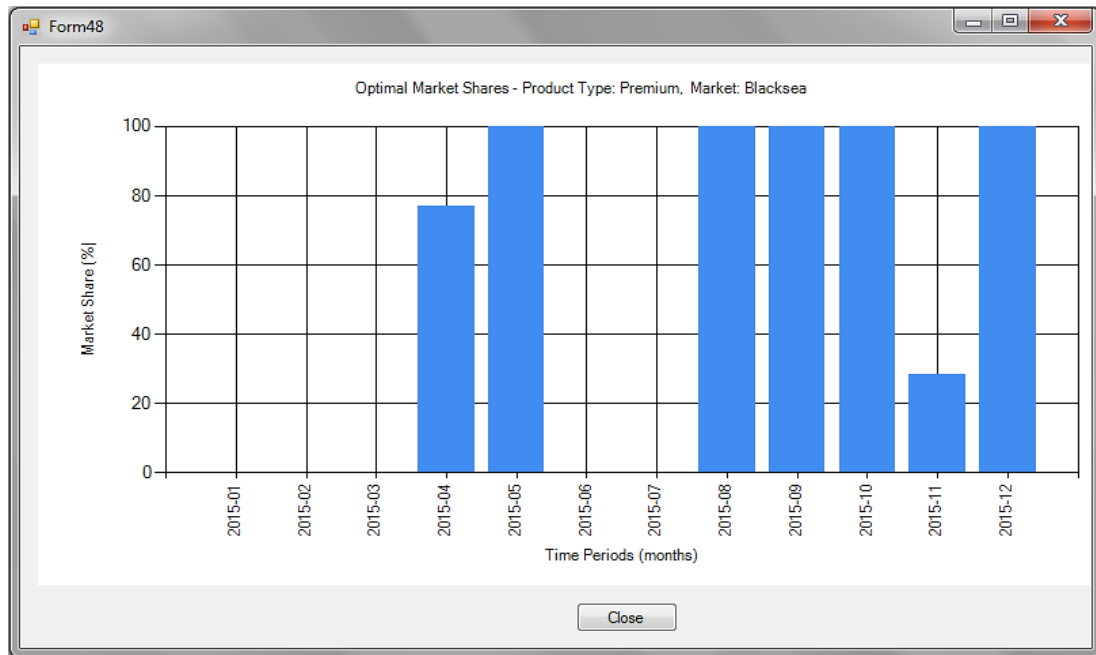


Figure 28. Optimal market share sample graph

On the transportation quantity graph screen which is given in Figure 29, the interacting user selects the factory, market and product type. Clicking on View Transportation Quantity Graph button gives optimal transportation quantities which are plotted for all time periods for the selected factory, market and product type, i.e. if the selected factory is Bursa, the selected market is Central Anatolia and the product type is basic, then the resulting graph is plotted as seen in Figure 30.

Optimal Plan Graphs

Select a factory: Gebze
 Select a market: Aegean
 Select a product type: Basic

View Transportation Quantity Graph

Close

Figure 29. Transportation quantity graphs screen

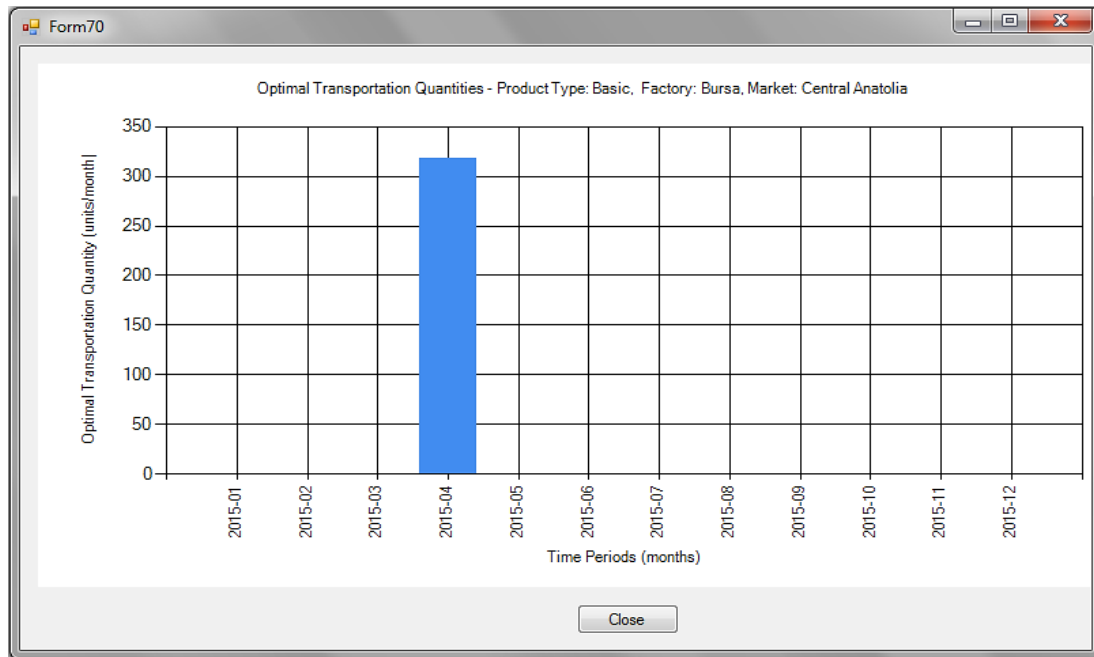


Figure 30. Optimal transportation quantity sample graph

Total revenue, total production cost, total transportation cost, total inventory holding cost, total stockout cost, total cost and optimal total profit is shown in a screen with a different layout resembling a balance sheet, as shown in Figure 31.

Category	Value	Unit
Total Revenue	+ 1,052,262	\$
Total Costs	- 166,316	\$
Total Production Cost	- 104,500	\$
Total Transportation Cost	- 54,766	\$
Total Inventory Holding Cost	- 7,050	\$
Total Stockout Cost	- 0	\$
Total Profit	+ 885,946	\$

Figure 31. View optimal objective values screen

All items are represented with dollars as unit at the end of the values in Figure 31. Here, total cost is calculated as the sum of total production cost, total transportation cost, total inventory holding cost, and total stockout cost. Optimal total profit is obtained by subtracting total cost from total revenue.

4.3.3 What-if analysis

After analyzing the optimal plan, the interacting user may go back using the Back button to the previous form, exit from the application using the Exit button, or change problem input parameters by clicking on What-If button. What-If analysis provides the users to change the desired input parameters, and resolve the model in different settings by making temporary changes in the database. The original input parameters and output results are stored in the database during the process without any changes. This is established by using temporary data tables instead of making the changes in the actual data tables.

When the user clicks on the What-If button or the Exit button, a message box that indicates a warning is shown asking the user to save the results of the current optimal solution to the database or not, with the buttons Yes, No, and Cancel. The warning message box can be seen in Figure 32.

After the relevant option is chosen in the warning message box, edit parameters for what-if analysis form is displayed which can be seen in Figure 33. The form is in the same layout with the view input parameters form, except small changes. As the form is loaded, temporary data tables are created using SQL queries in order to protect the original parameters in the database. The original input parameters are inserted into the temporary data tables with the aid of SQL syntaxes

embedded in the source code. What-if analysis allows the user to change any original parameter and resolve the NLP to see its impact on the optimal solution.

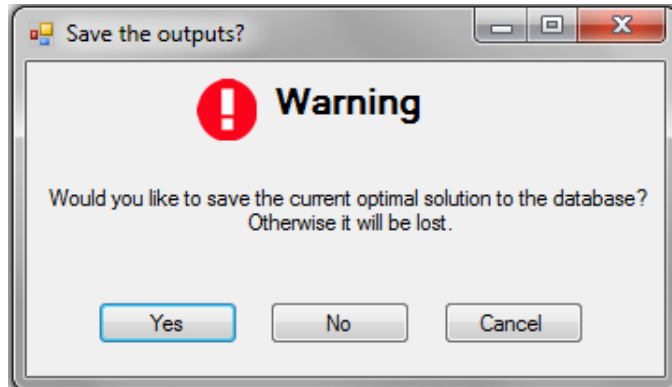


Figure 32. Warning message box

When the GAMS solver stops, view what-if optimal plan form is displayed that can be seen in Figure 34. It allows the user to view the results of the what-if analysis in the same layout of the view optimal plan form. The button with the data table icon brings up the table of what-if analysis output variables, and the button with the graph icon brings up the related graph screens to plot graphs.

This is the final step of DSS development as the model is solved with the original parameters of the database, and when the what-if analysis is completed the user can go back to previous forms or exit from the application. There is no save option as the changes done for what-if analysis are temporary.

PTP DSS - Edit Parameters for What-If Analysis

Edit Parameters for What-If Analysis

Click the related button to edit the parameters.

Factories

F₁
F₂
F₃
F_i

Markets

M₁
M₂
M₃
M_j

Unit Production Cost (\$/unit)
Unit Inventory Holding Cost (\$/unit)
Factory Capacity (units/month)

Initial Inventory Level (units/month)

Unit Transportation Cost (\$/unit)

Product Types

Time Periods

Attraction Sensitivity
Unit Stockout Cost (\$/unit)

Market Demand (units/month)

View Optimal Plan for What-If Analysis

Click the Resolve button to solve the problem and view the optimal plan.

Factories

F₁
F₂
F₃
F_i

Markets

M₁
M₂
M₃
M_j

Production Quantity (units/month)
Inventory Level (units/month)

Transportation Quantity (units/month)

Total Profit (\$)
Total Revenue (\$)
Total Production Cost (\$)
Total Transportation Cost (\$)
Total Inventory Holding Cost (\$)
Total Stockout Cost (\$)
Total Cost (\$)

Unit Price (\$/unit)
Attraction
Market Share (%)
Stockout Quantity (units/month)

Back Resolve Exit

Figure 33. Edit parameters for what-if analysis form

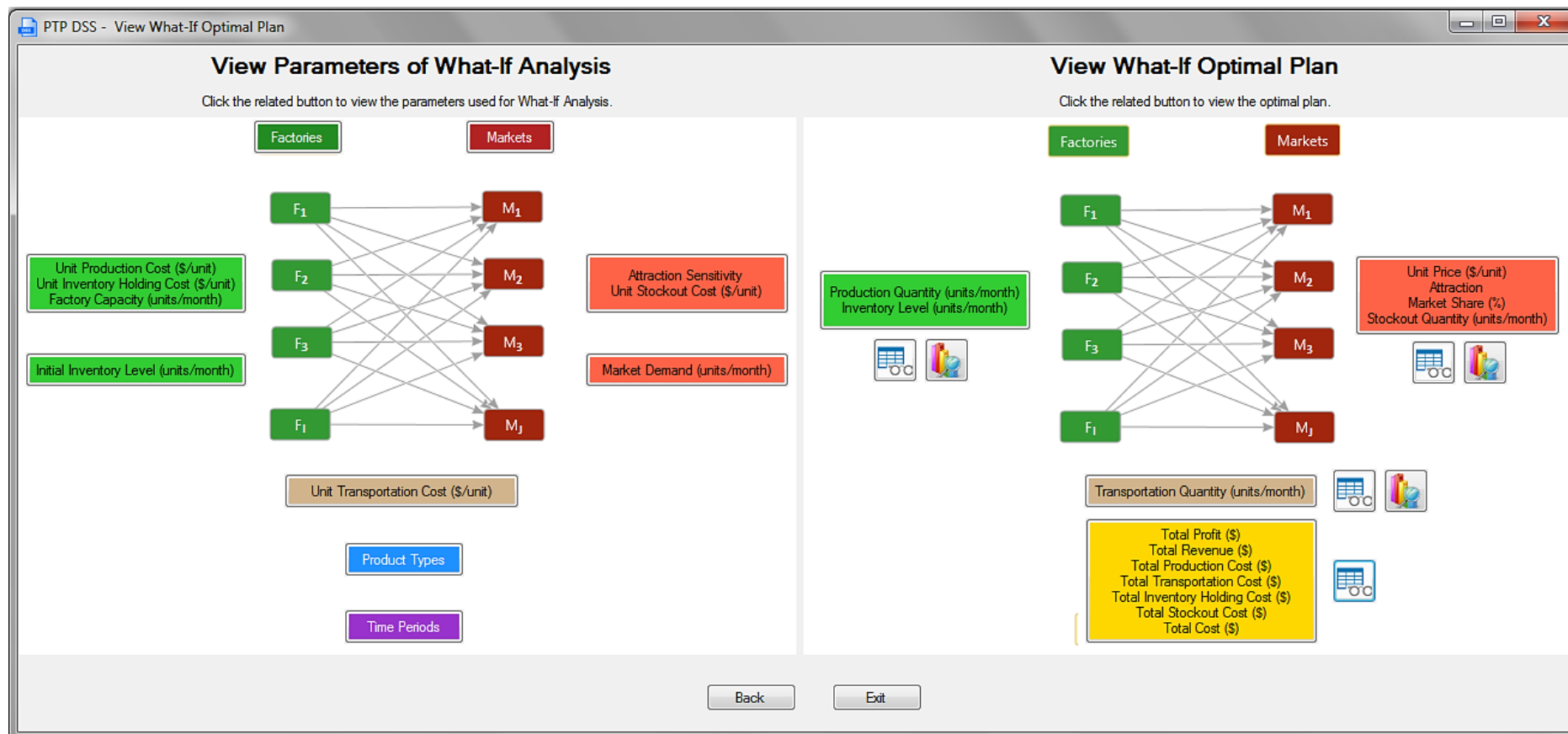


Figure 34. View what-if optimal plan form

CHAPTER 5

CONCLUSION

In this thesis study, a DSS is generated to provide an efficient, effective and flexible decision environment to optimize the production, transportation, and pricing decisions. An NLP model is developed to optimize the integrated production, transportation and pricing decisions. The model is studied in many ways, including several sensitivity analyses in both an uncapacitated setting and a capacitated setting in a single period, and scenario analyses in multiple periods. The originality of this thesis study is due to the fact that MSA models are used to determine the relationship between prices and demands in a cooperative competition environment.

In MSA models, the attraction sensitivity parameter provides an upper bound on the maximum price that can be set for that product. As the attraction sensitivity increases, the product becomes less attractive since the maximum price that can be set for that product is smaller. The attractiveness of a product also depends on its unit costs. The profit potential of a product is the difference between its maximum price and its unit cost, and it plays a key role on the optimal solution and optimal profits. Noting that the profit potential can be increased either by decreasing the attraction sensitivity or by decreasing the unit costs, two sensitivity analyses are provided to observe the effect of changing these variables on the optimal profits. Obviously any attempts to improve these parameters bring several benefits in terms of profits obtained. The analyses provide the upper bounds on the costs of the investments to be made for these improvements.

In this thesis study, several managerial implications are provided on the optimal solution based on the relative profit potentials of the competing products. In

the optimal solution of our model, it is seen that prices are always at their upper bounds, or very close to it. Indeed, this is quite expected because in MSA models, the total market demand is exogenous, i.e., the total market size does not decrease when the average of the prices of the competing products increase. Hence, simultaneously increasing the prices of the competing products certainly results in increased revenue. Since the aim is to maximize the profits in this study, in the optimal solution prices are either at their upper bounds or very close to it.

It is also found that the allocation of the market between competing products is made in accordance to their profit potentials. The optimal solution tries to increase the market share of the product with higher profit potential as much as possible. If there is no capacity constraint, then the product with higher profit potential captures the whole market leaving no sales to its competing products which have less profit potentials. This conclusion is very apparent in single period case. When there is capacity restriction of the product with higher profit potential, then the market share of this product is limited by its capacity. The rest of the market is fulfilled by the product which has the second highest profit potential under its capacity restrictions.

In multiple periods case, the intertemporal effect is recognized on ranking the profit potentials. The optimal solution tends to allocate the highest market share to the product which has the highest profit potential among all other periods and products. Under capacity constraints, the model makes adjustments between periods and products to decrease the total costs of stockouts. Quite intuitively it is observed that as the relative capacities increase, the optimal total stockout quantities decrease resulting in higher total profits. .

The developed DSS (PTP-DSS) optimizes the decisions of production, transportation and pricing centrally in a cooperative competition environment. The

DSS aims to speed up the decision making process so the right decisions are given quickly in order to take action against business challenges. The DSS provides the users a simple user interface with brief explanations of the necessary functions and it provides well-structured visuals such as tables and graphs to present the optimal profits, optimal costs, optimal unit prices, optimal production quantities, optimal transportation quantities, optimal inventory levels, optimal stockout quantities, and optimal market shares.

Future studies may include warehouses being located in a different location rather than being located within the factory facility. In that case, the network model will turn into a transshipment model, requiring an additional index for the warehouses. With the addition of that aspect, the model can be adapted to many different real-life problems although the complexity of the model may increase due to the addition of a new index.

Another future work may be the implementation of the database and the DSS into a web-based application, so that the DSS becomes a web-based DSS. Then, the interacting users may have access to the DSS from the Internet adding flexibility to the system. The database can be reconfigured into a cloud computing system, i.e. Microsoft Azure provides tools for such an implementation.

Lastly, Business Intelligence (BI) can be implemented to the DSS which will provide OLAP data cubes with dimensions and measures. The interacting users would then view slices of the cube providing better organized view of tables, make calculations on the cube and use data mining tools to better estimate the attraction sensitivity parameters. SQL Server includes BI tools whereas the implementation is considered to be beyond the scope of this thesis study.

APPENDIX A

GAMS MODEL

```
SET          I          "Factory index"      /
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT DISTINCT Factory_Index FROM Factory"
O="factory.inc"
$include factory.inc
/;

SET          J          "Market index"       /
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT DISTINCT Market_Index FROM Market"
O="market.inc"
$include market.inc
/;

SET          K          "Product index"      /
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT DISTINCT Product_Index FROM Product"
O="product.inc"
$include product.inc
/;

SET          T          "Period index"       /
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT DISTINCT Period_Index FROM Period"
O="period.inc"
$include period.inc
/;

alias (K, KK);

PARAMETER    U(I,K,T) production cost ($)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT
Factory_Index,Product_Index,Period_Index,Prod_Cost FROM
FactoryProductPeriod" O="prodcost.inc"
$include prodcost.inc
/;

PARAMETER    H(I,K,T) inventory holding cost ($)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT
Factory_Index,Product_Index,Period_Index,Inv_Cost FROM
FactoryProductPeriod" O="invcost.inc"
$include invcost.inc
/;

PARAMETER    C(I,J,K,T) transportation cost ($)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT
Factory_Index,Market_Index,Product_Index,Period_Index,Trans_Cost
FROM FactoryMarketProductPeriod" O="transcost.inc"
$include transcost.inc
/;

PARAMETER    Q(J,T) demand (units)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT Market_Index,Period_Index,Demand FROM
MarketPeriod" O="demand.inc"
```

```

$include demand.inc
/;

PARAMETER          F(I,K,T) capacity (units)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT
Factory_Index,Product_Index,Period_Index,Capacity FROM
FactoryProductPeriod" O="capacity.inc"
$include capacity.inc
/;

PARAMETER          L0(I,K) initial inventory (units)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT Factory_Index,Product_Index,Initial_Inv FROM
FactoryProduct" O="initialinv.inc"
$include initialinv.inc
/;

PARAMETER          B(J,K,T) attraction sensitivity /
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT
Market_Index,Product_Index,Period_Index,Attr_Sensitivity FROM
MarketProductPeriod" O="attsens.inc"
$include attsens.inc
/;

PARAMETER          S(J,K,T) stockout cost ($)/
$call =sql2gms C="DSN=datasource;Data Source=datasource; Initial
Catalog=dss" Q="SELECT
Market_Index,Product_Index,Period_Index,Stockout_Cost FROM
MarketProductPeriod" O="stockout.inc"
$include stockout.inc
/;

SCALARS
TotRevenue          total revenue ($),
TotProd             total production cost ($),
TotTrans            total transportation cost ($),
TotInv              total inventory holding cost ($),
TotStockout         total stockout cost ($),
TotCost             total cost ($),
Relcap              relative capacity;

PARAMETER
D(J,K,T) stockout quantity (units);
RELCAP = SUM((I,K,T),F(I,K,T))/SUM((J,T),Q(J,T));

VARIABLES
Z "total profit ($)",
X(I,J,K,T) "transportation quantity (units)",
Y(I,K,T) "production quantity (units)",
L(I,K,T) "inventory level (units)",
P(J,K,T) "price ($)",
A(J,K,T) "attraction",
M(J,K,T) "market share";

POSITIVE VARIABLES X(I,J,K,T), Y(I,K,T), L(I,K,T), P(J,K,T),
A(J,K,T), M(J,K,T);

EQUATIONS
PROFIT define objective function,

```

```

CAPACITY(I,K,T) capacity constraint,
DEMAND(J,K,T) demand constraint,
MSHARE(J,K,T) market share constraint,
MSHARE2(J,T) market shares add up to 1 constraint,
ATTRACTION(J,K,T) attraction constraint,
INVENTORY(I,K,T) inventory balance constraint;

PROFIT .. Z =E=
(SUM((J,K,T),P(J,K,T)*SUM((I),X(I,J,K,T))))-
((SUM((I,K,T),U(I,K,T)*Y(I,K,T)))+(SUM((I,K,T),H(I,K,T)*L(I,K,T)))+(
SUM((I,J,K,T),C(I,J,K,T)*X(I,J,K,T))))-
(SUM((J,K,T),S(J,K,T)*(Q(J,T)*M(J,K,T)-SUM((I),X(I,J,K,T)))));
CAPACITY(I,K,T) .. Y(I,K,T) =L= F(I,K,T) ;
DEMAND(J,K,T) .. SUM((I),X(I,J,K,T)) =L=
Q(J,T)*M(J,K,T);
MSHARE(J,K,T) .. M(J,K,T) =E=
A(J,K,T)/(SUM(KK,A(J,KK,T)+0.00000000000005));
MSHARE2(J,T) .. SUM(K,M(J,K,T)) =G=
0.9999999;
ATTRACTION(J,K,T) .. A(J,K,T) =E= 1-
B(J,K,T)*P(J,K,T);
INVENTORY(I,K,T) .. L(I,K,T) =E= L(I,K,T-
1)$ (ORD(T) GT 1)+L0(I,K)$ (ORD(T) EQ 1)+Y(I,K,T)-SUM(J,X(I,J,K,T));

MODEL MARKETSHARE /ALL/ ;
*Option decimals=8;
SOLVE MARKETSHARE USING NLP MAXIMIZING Z ;
D(J,K,T) = M.L(J,K,T)*Q(J,T)-SUM((I),X.L(I,J,K,T));
TotRevenue = SUM((J,K,T),P.L(J,K,T)*(SUM((I),X.L(I,J,K,T))));
TotProd = SUM((I,K,T),U(I,K,T)*Y.L(I,K,T));
TotTrans = SUM((I,J,K,T),C(I,J,K,T)*X.L(I,J,K,T));
TotInv = SUM((I,K,T),H(I,K,T)*L.L(I,K,T));
TotStockout = SUM((J,K,T),S(J,K,T)*D(J,K,T));
TotCost = TotProd + TotTrans + TotInv + TotStockout;

execute_unload
'results.gdx',i,j,k,t,x,y,l,p,a,m,z,TotRevenue,TotProd,TotTrans,TotI
nv,TotStockout,TotCost,u,h,c,q,f,l0,b,s,D,relcap;
execute 'gdxviewer.exe i=results.gdx csv=Z.csv id=Z';
execute 'gdxviewer.exe i=results.gdx csv=X.csv id=X';
execute 'gdxviewer.exe i=results.gdx csv=Y.csv id=Y';
execute 'gdxviewer.exe i=results.gdx csv=L.csv id=L';
execute 'gdxviewer.exe i=results.gdx csv=P.csv id=P';
execute 'gdxviewer.exe i=results.gdx csv=A.csv id=A';
execute 'gdxviewer.exe i=results.gdx csv=M.csv id=M';
execute 'gdxviewer.exe i=results.gdx csv=D.csv id=D';
execute 'gdxviewer.exe i=results.gdx csv=R.csv id=TotRevenue';
execute 'gdxviewer.exe i=results.gdx csv=PC.csv id=TotProd';
execute 'gdxviewer.exe i=results.gdx csv=TC.csv id=TotTrans';
execute 'gdxviewer.exe i=results.gdx csv=IC.csv id=TotInv';
execute 'gdxviewer.exe i=results.gdx csv=SC.csv id=TotStockout';
execute 'gdxviewer.exe i=results.gdx csv=TOTC.csv id=TotCost';
execute 'gdxviewer results.gdx';

```

APPENDIX B

PARAMETER TABLES

	Factory Index	Product Type Index	Initial Inventory Level (units/month)
▶	F1	P1	127
	F1	P2	11
	F2	P1	115
	F2	P2	32
	F3	P1	132
	F3	P2	1
	F4	P1	111
	F4	P2	94

	Market Index	Period Index	Market Demand (units/month)
▶	M1	2015-01	176
	M1	2015-02	699
	M1	2015-03	797
	M1	2015-04	170
	M1	2015-05	773
	M1	2015-06	584
	M1	2015-07	145
	M1	2015-08	709
	M1	2015-09	644
	M1	2015-10	704
	M1	2015-11	326
	M1	2015-12	589
	M2	2015-01	540
	M2	2015-02	887
	M2	2015-03	831
	M2	2015-04	533
	M2	2015-05	856
	M2	2015-06	754
	M2	2015-07	494

	Factory Index	Product Type Index	Period Index	Unit Production Cost (\$/unit)	Unit Inventory Holding Cost (\$/unit)	Factory Capacity (units/month)
▶	F1	P1	2015-01	5	2	314
	F1	P1	2015-02	5	2	367
	F1	P1	2015-03	2	1	194
	F1	P1	2015-04	4	1	346
	F1	P1	2015-05	4	2	378
	F1	P1	2015-06	6	3	498
	F1	P1	2015-07	2	4	401
	F1	P1	2015-08	5	1	494
	F1	P1	2015-09	4	3	104
	F1	P1	2015-10	4	4	282
	F1	P1	2015-11	6	2	455
	F1	P1	2015-12	7	1	353
	F1	P2	2015-01	3	1	399
	F1	P2	2015-02	4	4	117
	F1	P2	2015-03	3	4	202
	F1	P2	2015-04	4	1	223

	Factory Index	Market Index	Product Type Index	Period Index	Unit Transportation Cost (\$/unit)
▶	F1	M1	P1	2015-01	2
	F1	M1	P1	2015-02	1
	F1	M1	P1	2015-03	1
	F1	M1	P1	2015-04	3
	F1	M1	P1	2015-05	6
	F1	M1	P1	2015-06	4
	F1	M1	P1	2015-07	4
	F1	M1	P1	2015-08	5
	F1	M1	P1	2015-09	5
	F1	M1	P1	2015-10	2
	F1	M1	P1	2015-11	5
	F1	M1	P1	2015-12	1
	F1	M1	P2	2015-01	4
	F1	M1	P2	2015-02	6
	F1	M1	P2	2015-03	4
	F1	M1	P2	2015-04	6
	F1	M1	P2	2015-05	5
	F1	M1	P2	2015-06	4
	F1	M1	P2	2015-07	3

APPENDIX C

VARIABLE TABLES

Production Quantity and Inventory Level Optimal Table

	Factory Index	Product Type Index	Period Index	Production Quantity (units/month)	Inventory Level (units/month)
▶	F1	P1	2015-01	314	114
	F1	P1	2015-02	367	87
	F1	P1	2015-03	194	0
	F1	P1	2015-04	346	346
	F1	P1	2015-05	378	724
	F1	P1	2015-06	0	719
	F1	P1	2015-07	401	0
	F1	P1	2015-08	494	494
	F1	P1	2015-09	104	159
	F1	P1	2015-10	282	0
	F1	P1	2015-11	0	0
	F1	P1	2015-12	0	0
	F1	P2	2015-01	399	410
	F1	P2	2015-02	117	0
	F1	P2	2015-03	202	202
	F1	P2	2015-04	223	425
	F1	P2	2015-05	369	621
	F1	P2	2015-06	157	779

Unit Price, Attraction, Market Share, Stockout Quantity Optimal Table

	Market Index	Product Type Index	Period Index	Unit Price (\$/unit)	Attraction	Market Share	Stockout Quantity (units/month)
▶	M1	P1	2015-01	32	0	0	
	M1	P1	2015-02	44	1E-05	100	
	M1	P1	2015-03	44	1E-05	100	
	M1	P1	2015-04	46.999	0	0	
	M1	P1	2015-05	46.999	1E-05	100	0
	M1	P1	2015-06	22	0	0	
	M1	P1	2015-07	29	0	0	
	M1	P1	2015-08	35.001	0	0	
	M1	P1	2015-09	33	1E-05	100	
	M1	P1	2015-10	50	1E-05	100	0
	M1	P1	2015-11	26	0	0	
	M1	P1	2015-12	28	0	0	
	M1	P2	2015-01	49	1E-05	100	
	M1	P2	2015-02	30	0	0	
	M1	P2	2015-03	30	0	0	
	M1	P2	2015-04	50	1E-05	100	0
	M1	P2	2015-05	27	0	0	
	M1	P2	2015-06	30	1E-05	100	0

APPENDIX D

SOURCE CODE OF VIEW OPTIMAL PLAN FORM

```
Public Class Form16
    Private Sub Form16_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim SQLCONN As New SqlClient.SqlConnection
        Dim SQLCMD As New SqlClient.SqlCommand
        SQLCONN = New SqlClient.SqlConnection("Server=TOSHIBA-
TOSH;Database=dss;Integrated security=True")
        SQLCONN.Open()
        SQLCMD = New SqlClient.SqlCommand("
TRUNCATE TABLE MarketProductPeriodR;
TRUNCATE TABLE FactoryMarketProductPeriodR;
TRUNCATE TABLE FactoryProductPeriodR;
TRUNCATE TABLE Objective;
ALTER TABLE FactoryProductPeriodR
DROP COLUMN Inventory_Level;

BULK INSERT FactoryProductPeriodR
FROM 'C:\Users\TOSHIBA\Documents\gamssdir\projdir\Y.CSV'
WITH
(
    FIRSTROW = 1,
    FIELDTERMINATOR = ',',
    ROWTERMINATOR = '\n'
);

ALTER TABLE FactoryProductPeriodR
DROP COLUMN Production_Qty;

CREATE TABLE Temp1
(Factory_Index      Varchar(50),
Product_Index Varchar(50),
Period_Index  Varchar(50),
Production_Qty      float  NULL);

BULK INSERT Temp1
FROM 'C:\Users\TOSHIBA\Documents\gamssdir\projdir\Y.CSV'
WITH
(
    FIRSTROW = 1,
    FIELDTERMINATOR = ',',
    ROWTERMINATOR = '\n'
);

CREATE TABLE Temp2
(Factory_Index      Varchar(50),
Product_Index Varchar(50),
Period_Index  Varchar(50),
Inventory_Level      float  NULL);

BULK INSERT Temp2
FROM 'C:\Users\TOSHIBA\Documents\gamssdir\projdir\L.CSV'
WITH
(
    FIRSTROW = 1,
    FIELDTERMINATOR = ',',
```

```

        ROWTERMINATOR = '\n'
    );

ALTER TABLE FactoryProductPeriodR
ADD Production_Qty float NULL;
ALTER TABLE FactoryProductPeriodR
ADD Inventory_Level float NULL;", SQLCONN)
        SQLCMD.ExecuteNonQuery()
        SQLCONN.Close()
        SQLCONN.Open()
        SQLCMD = New SqlClient.SqlCommand("UPDATE FactoryProductPeriodR SET
FactoryProductPeriodR.Production_Qty=(
    Select ROUND(Production_Qty,3)
    From Temp1
    Where Temp1.Factory_Index = FactoryProductPeriodR.Factory_Index
        And Temp1.Product_Index=FactoryProductPeriodR.Product_Index
        And Temp1.Period_Index=FactoryProductPeriodR.Period_Index);
UPDATE FactoryProductPeriodR SET FactoryProductPeriodR.Inventory_Level=(
    Select ROUND(Inventory_Level,3)
    From Temp2
    Where Temp2.Factory_Index = FactoryProductPeriodR.Factory_Index
        And Temp2.Product_Index=FactoryProductPeriodR.Product_Index
        And Temp2.Period_Index=FactoryProductPeriodR.Period_Index);
DROP TABLE Temp1;
DROP TABLE Temp2;", SQLCONN)
        SQLCMD.ExecuteNonQuery()
        SQLCONN.Close()
        SQLCONN.Open()
        SQLCMD = New SqlClient.SqlCommand("
ALTER TABLE MarketProductPeriodR
DROP COLUMN Attraction;
ALTER TABLE MarketProductPeriodR
DROP COLUMN Market_Share;
ALTER TABLE MarketProductPeriodR
DROP COLUMN Stockout_Qty;
BULK INSERT MarketProductPeriodR
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\P.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );
ALTER TABLE MarketProductPeriodR
DROP COLUMN Price;
CREATE TABLE Temp3
(Market_Index Varchar(50),
Product_Index Varchar(50),
Period_Index Varchar(50),
Price float NULL);

BULK INSERT Temp3
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\P.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );

CREATE TABLE Temp4
(Market_Index Varchar(50),

```

```

Product_Index Varchar(50),
Period_Index Varchar(50),
Attraction float NULL);

BULK INSERT Temp4
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\A.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);

CREATE TABLE Temp5
(Market_Index Varchar(50),
Product_Index Varchar(50),
Period_Index Varchar(50),
Market_Share float NULL);

BULK INSERT Temp5
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\M.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);

CREATE TABLE Temp15
(Market_Index Varchar(50),
Product_Index Varchar(50),
Period_Index Varchar(50),
Stockout_Qty float NULL);

BULK INSERT Temp15
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\D.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);

CREATE TABLE Temp13
(Market_Index Varchar(50),
Product_Index Varchar(50),
Period_Index Varchar(50),
Market_Share float NULL);

BULK INSERT Temp13
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\M.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);

ALTER TABLE MarketProductPeriodR
ADD Price float NULL;
ALTER TABLE MarketProductPeriodR
ADD Attraction float NULL;

```

```

ALTER TABLE MarketProductPeriodR
ADD Market_Share float NULL;
ALTER TABLE MarketProductPeriodR
ADD Stockout_Qty float NULL;"; SQLCONN)
    SQLCMD.ExecuteNonQuery()
    SQLCONN.Close()
    SQLCONN.Open()
    SQLCMD = New SqlClient.SqlCommand("UPDATE Temp13 SET
Temp13.Market_Share=(
SELECT ROUND(Market_Share,3) FROM Temp5
WHERE Temp5.Market_Index=Temp13.Market_Index
AND Temp5.Product_Index=Temp13.Product_Index
AND Temp5.Period_Index=Temp13.Period_Index);", SQLCONN)
    SQLCMD.ExecuteNonQuery()
    SQLCONN.Close()
    SQLCONN.Open()
    SQLCMD = New SqlClient.SqlCommand("UPDATE MarketProductPeriodR SET
MarketProductPeriodR.Price=(
SELECT ROUND(Price,3)
FROM Temp3
WHERE Temp3.Market_Index=MarketProductPeriodR.Market_Index
AND Temp3.Product_Index=MarketProductPeriodR.Product_Index
AND Temp3.Period_Index=MarketProductPeriodR.Period_Index);
UPDATE MarketProductPeriodR SET MarketProductPeriodR.Attraction=(
SELECT ROUND(Attraction,7)
FROM Temp4
WHERE Temp4.Market_Index=MarketProductPeriodR.Market_Index
AND Temp4.Product_Index=MarketProductPeriodR.Product_Index
AND Temp4.Period_Index=MarketProductPeriodR.Period_Index);
UPDATE MarketProductPeriodR SET MarketProductPeriodR.Market_Share=(
SELECT Market_Share=CONVERT(varchar(50),100.00*Market_Share) FROM Temp13
WHERE Temp13.Market_Index=MarketProductPeriodR.Market_Index
AND Temp13.Product_Index=MarketProductPeriodR.Product_Index
AND Temp13.Period_Index=MarketProductPeriodR.Period_Index);
UPDATE MarketProductPeriodR SET MarketProductPeriodR.Stockout_Qty=(
SELECT ROUND(Stockout_Qty,3)
FROM Temp15
WHERE Temp15.Market_Index=MarketProductPeriodR.Market_Index
AND Temp15.Product_Index=MarketProductPeriodR.Product_Index
AND Temp15.Period_Index=MarketProductPeriodR.Period_Index);
DROP TABLE Temp3;
DROP TABLE Temp4;
DROP TABLE Temp5;
DROP TABLE Temp13;
DROP TABLE Temp15;"; SQLCONN)
    SQLCMD.ExecuteNonQuery()
    SQLCONN.Close()
    SQLCONN.Open()
    SQLCMD = New SqlClient.SqlCommand("
BULK INSERT FactoryMarketProductPeriodR
FROM 'C:\Users\TOSHIBA\Documents\gamssdir\projdir\X.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);

ALTER TABLE FactoryMarketProductPeriodR
DROP COLUMN Trans_Qty;

CREATE TABLE Temp6

```

```

(Factory_Index      Varchar(50),
Market_Index      Varchar(50),
Product_Index Varchar(50),
Period_Index  Varchar(50),
Trans_Qty      float  NULL);

BULK INSERT Temp6
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\X.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
)
ALTER TABLE FactoryMarketProductPeriodR
ADD Trans_Qty float NULL;"; SQLCONN)
SQLCMD.ExecuteNonQuery()
SQLCONN.Close()
SQLCONN.Open()
SQLCMD = New SqlClient.SqlCommand("UPDATE FactoryMarketProductPeriodR
SET FactoryMarketProductPeriodR.Trans_Qty=(
SELECT ROUND(Trans_Qty,3)
FROM Temp6
WHERE Temp6.Factory_Index=FactoryMarketProductPeriodR.Factory_Index
AND Temp6.Market_Index=FactoryMarketProductPeriodR.Market_Index
AND Temp6.Product_Index=FactoryMarketProductPeriodR.Product_Index
AND Temp6.Period_Index=FactoryMarketProductPeriodR.Period_Index);
DROP TABLE Temp6;";, SQLCONN)
SQLCMD.ExecuteNonQuery()
SQLCONN.Close()
SQLCONN.Open()
SQLCMD = New SqlClient.SqlCommand("
ALTER TABLE Objective
DROP COLUMN Total_Revenue;
ALTER TABLE Objective
DROP COLUMN Total_Prod_Cost;
ALTER TABLE Objective
DROP COLUMN Total_Trans_Cost;
ALTER TABLE Objective
DROP COLUMN Total_Inv_Cost;
ALTER TABLE Objective
DROP COLUMN Total_Stockout_Cost;
ALTER TABLE Objective
DROP COLUMN Total_Cost;
CREATE TABLE Temp7
(Total_Profit float NULL);

BULK INSERT Temp7
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\Z.CSV'
WITH
(
FIRSTROW = 1,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);

CREATE TABLE Temp8
(Total_Revenue      float  NULL);

BULK INSERT Temp8
FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\R.CSV'
WITH

```

```

(
    FIRSTROW = 1,
    FIELDTERMINATOR = ',',
    ROWTERMINATOR = '\n'
);

CREATE TABLE Temp9
(Total_Prod_Cost          float  NULL);

BULK INSERT Temp9
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\PC.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );

CREATE TABLE Temp10
(Total_Trans_Cost        float  NULL);

BULK INSERT Temp10
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\TC.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );

CREATE TABLE Temp11
(Total_Inv_Cost          float  NULL);

BULK INSERT Temp11
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\IC.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );

CREATE TABLE Temp12
(Total_Stockout_Cost     float  NULL);

BULK INSERT Temp12
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\SC.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );

CREATE TABLE Temp14
(Total_Cost              float  NULL);

BULK INSERT Temp14
    FROM 'C:\Users\TOSHIBA\Documents\gamsdir\projdir\TOTC.CSV'
    WITH
    (
        FIRSTROW = 1,
        FIELDTERMINATOR = ',',
        ROWTERMINATOR = '\n'
    );

```

```

ALTER TABLE Objective
ADD Total_Revenue float null;
ALTER TABLE Objective
ADD Total_Prod_Cost float null;
ALTER TABLE Objective
ADD Total_Trans_Cost float null;
ALTER TABLE Objective
ADD Total_Inv_Cost float null;
ALTER TABLE Objective
ADD Total_Stockout_Cost float null;
ALTER TABLE Objective
ADD Total_Cost float null;"; SQLCONN)
    SQLCMD.ExecuteNonQuery()
    SQLCONN.Close()
    SQLCONN.Open()
    SQLCMD = New SqlClient.SqlCommand("INSERT INTO Objective
(Total_Profit)
    SELECT ROUND(Total_Profit,0)
    FROM Temp7;
UPDATE Objective SET Objective.Total_Revenue=(
    SELECT ROUND(Total_Revenue,0)
    FROM Temp8);
UPDATE Objective SET Objective.Total_Prod_Cost=(
    SELECT ROUND(Total_Prod_Cost,0)
    FROM Temp9);
UPDATE Objective SET Objective.Total_Trans_Cost=(
    SELECT ROUND(Total_Trans_Cost,0)
    FROM Temp10);
UPDATE Objective SET Objective.Total_Inv_Cost=(
    SELECT ROUND(Total_Inv_Cost,0)
    FROM Temp11);
UPDATE Objective SET Objective.Total_Stockout_Cost=(
    SELECT ROUND(Total_Stockout_Cost,0)
    FROM Temp12);
UPDATE Objective SET Objective.Total_Cost=(
    SELECT ROUND(Total_Cost,0)
    FROM Temp14);
DROP TABLE Temp7;
DROP TABLE Temp8;
DROP TABLE Temp9;
DROP TABLE Temp10;
DROP TABLE Temp11;
DROP TABLE Temp12;
DROP TABLE Temp14;"; SQLCONN)
    SQLCMD.ExecuteNonQuery()
    SQLCONN.Close()
    SQLCONN.Open()
    SQLCMD = New SqlClient.SqlCommand("TRUNCATE TABLE
FactoryMarketProductPeriodB;
TRUNCATE TABLE FactoryProductB;
TRUNCATE TABLE FactoryProductPeriodB;
TRUNCATE TABLE MarketPeriodB;
TRUNCATE TABLE MarketProductPeriodB;
Insert Into FactoryMarketProductPeriodB
SELECT * FROM FactoryMarketProductPeriod;
Insert Into FactoryProductB
SELECT * FROM FactoryProduct;
Insert Into FactoryProductPeriodB
SELECT * FROM FactoryProductPeriod;
Insert Into MarketPeriodB
SELECT * FROM MarketPeriod;

```



```

Insert Into MarketProductPeriodB
SELECT *          FROM          MarketProductPeriod;" , SQLCONN)
        SQLCMD.ExecuteNonQuery()
        SQLCONN.Close()
    End Sub

    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles
Button4.Click
        Form5.Show()
        Me.Hide()
    End Sub

    Private Sub Button3_Click(sender As Object, e As EventArgs) Handles
Button3.Click
        Form89.Show()
    End Sub

    Private Sub Button6_Click(sender As Object, e As EventArgs) Handles
Button6.Click
        Form11.Show()
    End Sub

    Private Sub Button7_Click(sender As Object, e As EventArgs) Handles
Button7.Click
        Form12.Show()
    End Sub

    Private Sub Button18_Click(sender As Object, e As EventArgs) Handles
Button18.Click
        Form13.Show()
    End Sub

    Private Sub Button21_Click(sender As Object, e As EventArgs) Handles
Button21.Click
        Form88.Show()
    End Sub

    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles
Button2.Click
        Form19.Show()
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
Button1.Click
        Form2.Show()
    End Sub

    Private Sub Button5_Click(sender As Object, e As EventArgs) Handles
Button5.Click
        Form8.Show()
    End Sub

    Private Sub Button8_Click(sender As Object, e As EventArgs) Handles
Button8.Click
        Form15.Show()
    End Sub

    Private Sub Button9_Click(sender As Object, e As EventArgs) Handles
Button9.Click
        Form9.Show()
    End Sub

```

```

        Private Sub Button11_Click(sender As Object, e As EventArgs) Handles
Button11.Click
            Form14.Show()
        End Sub

        Private Sub Button12_Click(sender As Object, e As EventArgs) Handles
Button12.Click
            Form10.Show()
        End Sub

        Private Sub Button13_Click(sender As Object, e As EventArgs) Handles
Button13.Click
            Form6.Show()
        End Sub

        Private Sub Button14_Click(sender As Object, e As EventArgs) Handles
Button14.Click
            Form7.Show()
        End Sub

        Private Sub Button15_Click(sender As Object, e As EventArgs) Handles
Button15.Click
            Form3.Show()
        End Sub

        Private Sub Button16_Click(sender As Object, e As EventArgs) Handles
Button16.Click
            Form4.Show()
        End Sub

        Private Sub Button10_Click(sender As Object, e As EventArgs) Handles
Button10.Click
            Form83.Show()
        End Sub

        Private Sub Button20_Click(sender As Object, e As EventArgs) Handles
Button20.Click
            Form84.Show()
        End Sub

        Private Sub Button25_Click(sender As Object, e As EventArgs) Handles
Button25.Click
            Form85.Show()
        End Sub

End Class

```

REFERENCES

- Ahn, H. S., Gümüő, M., & Kaminsky, P. (2007). Pricing and manufacturing decisions when demand is a function of prices in multiple periods. *Operations Research*, 55(6), 1039-1057.
- Ardalan, A. (1994). Optimal prices and order quantities when temporary price discounts result in increase in demand. *European Journal of Operational Research*, 72(1), 52-61.
- Assuncao, J. L., & Meyer, R. J. (1993). The rational effect of price promotions on sales and consumption. *Management Science*, 39(5), 517-535.
- Bell, D. E., Keeney, R. L., & Little, J. D. (1975). A market share theorem. *Journal of Marketing Research*, 136-141.
- Biswas, S., & Narahari, Y. (2004). Object oriented modeling and decision support for supply chains. *European Journal of Operational Research*, 153(3), 704-726.
- Bitran, G., & Caldentey, R. (2003). An overview of pricing models for revenue management. *Manufacturing & Service Operations Management*, 5(3), 203-229.
- Burwell, T. H., Dave, D. S., Fitzpatrick, K. E., & Roy, M. R. (1997). Economic lot size model for price-dependent demand under quantity and freight discounts. *International Journal of Production Economics*, 48(2), 141-155.
- Chan, L. M., Shen, Z. M., Simchi-Levi, D., & Swann, J. L. (2004). Coordination of pricing and inventory decisions: A survey and classification. In *Handbook of quantitative supply chain analysis* (pp. 335-392). Secaucus, NJ, Springer.
- Chen, X., & Simchi-Levi, D. (2004). Coordinating inventory control and pricing strategies with random demand and fixed ordering cost: The finite horizon case. *Operations Research*, 52(6), 887-896.
- Chen, X., & Simchi-Levi, D. (2012). Pricing and inventory management. *The Oxford handbook of pricing management*, 784-822.

- Chen, Z. L. (2004). Integrated Production and Distribution Operations. In *Handbook of quantitative supply chain analysis* (pp. 711-745). Secaucus, NJ: Springer.
- Cooper, L. G., & Nakanishi, M. (1988). *Market-share analysis: Evaluating competitive marketing effectiveness*. Boston: Kluwer Academic Publishers.
- Ekşioğlu, S. D., Romeijn, H. E., & Pardalos, P. M. (2006). Cross-facility management of production and transportation planning problem. *Computers & Operations Research*, 33(11), 3231-3251.
- Elmaghraby, W., & Keskinocak, P. (2003). Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. *Management Science*, 49(10), 1287-1309.
- Falkenberg, A. W. (1984). Modeling market share: A study of the Japanese and US performance in the US auto market. *Journal of the Academy of Marketing Science*, 12(4), 145-160.
- GDXVIEWER (n.d.), *GAMS Documentation Center*, Retrieved from <https://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Ftools%2Fgdxviewer%2Findex.html>
- Gen, M., & Syarif, A. (2005). Hybrid genetic algorithm for multi-time period production/distribution planning. *Computers & Industrial Engineering*, 48(4), 799-809.
- Gilbert, S. M. (2000). Coordination of pricing and multiple-period production across multiple constant priced goods. *Management Science*, 46(12), 1602-1616.
- Gnanendran, K., & Chien, Y. (2006). Integrating pricing and distribution decisions in multiple markets. *Journal of Business & Economics Research (JBER)*, 4(12), 81-90.
- Gümüş, M., & Kaminsky, P. (2010). The impact of substitution and intertemporal demand on coordinated pricing-inventory strategies. Retrieved from <http://ssrn.com/abstract=1753020>
- Gümüş, M., Kaminsky, P., & Mathur, S. (2011). The impact of product substitution and retail capacity on the timing and depth of price promotions: theory and evidence. Retrieved from <http://ssrn.com/abstract=1964325>

- Gupta, A., & Maranas, C. D. (2003). Managing demand uncertainty in supply chain planning. *Computers & Chemical Engineering*, 27(8), 1219-1227.
- Huh, W. T., & Janakiraman, G. (2008). (s, S) optimality in joint inventory-pricing control: An alternate approach. *Operations Research*, 56(3), 783-790.
- Ishii, K., Takahashi, K., & Muramatsu, R. (1988). Integrated production, inventory and distribution systems. *The International Journal Of Production Research*, 26(3), 473-482.
- Kalvelagen, E. (n.d.), SQL2GMS. *GAMS Documentation Center*, Retrieved from <https://www.gams.com/help/index.jsp?topic=%2Fgams.doc%2Ftools%2Fsql2gms%2Findex.html>
- Karkkainen, H., Elfvengren, K., Torkkeli, M., & Tuominen, M. (2001). Group decision support systems in industrial customer need assessment. In *Management of Engineering and Technology, 2001. PICMET'01. Portland International Conference on* (Vol. 1, pp. 94-vol). IEEE.
- Keskinocak, P., Goodwin, R., Wu, F., Akkiraju, R., & Murthy, S. (2001). Decision support for managing an electronic supply chain. *Electronic Commerce Research*, 1(1-2), 15-31.
- Kotler, P. (1971). *Marketing decision making: A model building approach*. New York: Holt, Rinehart and Winston.
- Kotler, P. (1984). *Marketing management: Analysis, planning, and control*, Fifth Edition, Englewood Cliffs, NJ: Prentice-Hall, Inc.
- Lambin, J. J. (1970). Optimal allocation of competitive marketing efforts: an empirical study. *Journal of Business*, 468-484.
- Leblanc, L. J., & Cooper, L. (1974). The transportation-production problem. *Transportation Science*, 8(4), 344-354.
- Mansouri, S. A., Gallea, D., & Askariazad, M. H. (2012). Decision support for build-to-order supply chain management through multiobjective optimization. *International Journal of Production Economics*, 135(1), 24-36.

- Microsoft Automatic Graph Layout. (n.d.). Retrieved January 26, 2016, from:
<http://research.microsoft.com/en-us/projects/msagl/>
- Naert, P. A., & Bultez, A. (1973). Logically consistent market share models. *Journal of Marketing Research*, 334-340.
- Park, Y. B. (2005). An integrated approach for production and distribution planning in supply chain management. *International Journal of Production Research*, 43(6), 1205-1224.
- Pol, A. & Ahuja, R. K. (2007). Principles of good database design. In *Developing web-enabled decision support systems*. (pp. 3-12). Belmont, MA: Dynamic Ideas.
- Pujari, N. A., Day, J. M., Huq, F., & Hale, T. S. (2008). A framework for an integrated distribution system optimization model. *International Journal of Logistics Systems and Management*, 4(5), 506-522.
- Schuur, P., Badur, B., & Sencer, A. (2015). Pricing competing products under production capacity restrictions: A market share approach based on linear attraction. Manuscript in preparation.
- Sharkey, T. C. (2011). Network flow problems with pricing decisions. *Optimization Letters*, 5(1), 71-83.
- Sharp, J. F., Snyder, J. C., & Greene, J. H. (1970). A decomposition algorithm for solving the multifacility production-transportation problem with nonlinear production costs. *Econometrica: Journal of the Econometric Society*, 490-506.
- Shetty, C. M. (1959). A solution to the transportation problem with nonlinear costs. *Operations Research*, 7(5), 571-580.
- Song, Y., Ray, S., & Boyaci, T. (2009). Technical note-optimal dynamic joint inventory-pricing control for multiplicative demand with fixed order costs and lost sales. *Operations Research*, 57(1), 245-250.
- Şeref, M. M. H., Ahuja, R. K. & Winston, W. L. (2007). The DSS development process. In *Developing spreadsheet-based decision support systems*. (pp.571-594). Belmont, MA: Dynamic Ideas.

- Wagner, H. M., & Whitin, T. M. (1958). Dynamic version of the economic lot size model. *Management Science*, 5(1), 89-96.
- Weiss, D. L. (1968). Determinants of market share. *Journal of Marketing Research*, 290-295.
- Whitin, T. M. (1955). Inventory control and price theory. *Management Science*, 2(1), 61-68.
- Yano, C. A., & Gilbert, S. M. (2004). Coordinated pricing and production/procurement decisions: A review. In *Managing business interfaces* (pp. 65-103). Boston, MA: Springer.
- Youssef, M. A., & Mahmoud, M. M. (1996). An iterative procedure for solving the uncapacitated production-distribution problem under concave cost function. *International Journal of Operations & Production Management*, 16(3), 18-27.