DECISION MAKING IMPLICATIONS FOR A SELECTED ECHELON IN THE BEER GAME

by

Mert Edalı

B.S., Industrial Engineering, Yıldız Technical University, 2011

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Industrial Engineering
Boğaziçi University
2014

DECISION MAKING IMPLICATIONS FOR A SELECTED ECHELON
IN THE BEER GAME

APPROVED BY:

Assist. Prof. Hakan Yaşarcan (Thesis Supervisor)	………………………
Prof. Yaman Barlas	………………………
Prof. Aslı Sencer	………………………

DATE OF APPROVAL: 15.01.2014

ACKNOWLEDGEMENTS

First and foremost, I would like to thank Assist. Prof. Hakan Yaşarcan, my thesis supervisor, for his guidance and support throughout this study. Without his foresight and comments, this thesis would not exist.

I would like to thank Prof. Yaman Barlas and Prof. Aslı Sencer for taking part in my thesis jury and providing valuable feedback.

I would like to express my deepest gratitude to my mother, Seyhan Parla, for her guidance and support throughout my life.

I thankfully acknowledge the support of TUBITAK - The Scientific and Technological Research Council of Turkey during my master.

This research is supported by a Marie Curie International Reintegration Grant within the 7th European Community Framework Programme (grant agreement number: PIRG07-GA-2010-268272) and also by Bogazici University Research Fund (grant no: 6924-13A03P1).

ABSTRACT

DECISION MAKING IMPLICATIONS FOR A SELECTED ECHELON IN THE BEER GAME

The beer production-distribution game, in short “The Beer Game”, is essentially a board game and it simulates a four echelon supply chain consisting of a retailer, wholesaler, distributor, and factory. During the game, every participant in a group of four is responsible for one of these four echelons and manages the associated inventory by placing orders. The aim of the game is to minimize the accumulated total cost obtained by the participants of a group managing each echelon. In this thesis, a mathematical model that is an exact one-to-one replica of the board version is constructed. The main aim of this thesis is to develop an understanding about how one should control an echelon in The Beer Game in the presence of identically controlled echelons; we assume that only the participant managing the echelon of concern behaves different than the rest of the group. We are specifically interested in the case where the echelons other than the selected one sub-optimally manage their individual inventories or backlogs. There can be two objectives: (i) the minimum cost for the echelon of concern can be obtained, (ii) the minimum group total cost can be obtained by optimizing the decision parameters of the selected echelon. Accordingly, we optimize the parameters of the *anchor-and-adjust heuristic*, which is the control policy used in this study, for the selected echelon by keeping the decision parameters constant for the rest of the three positions. We obtain different instances of the anchor-and-adjust ordering policy by optimizing *stock adjustment time* and by optimizing *desired inventory* of the selected echelon. In general, the group total cost can be decreased by allowing an increase in the total cost of the selected echelon. Unexpectedly, we obtained the lowest group total costs for the wholesaler when we minimized the group total cost by sacrificing the objective of minimizing the cost of the echelon of concern.

ÖZET

BİRA DAĞITIM OYUNUNDA SEÇİLEN BİR TEDARİK ZİNCİRİ KADEMESİ İÇİN KARAR OLUŞTURMA TAVSİYELERİ

Bira Üretim-Dağıtım Oyunu, kısaca "Bira Oyunu", temelde masa üzerinde oynanan bir oyundur ve bir perakendeci, bir toptancı, bir dağıtıcı ve bir fabrikadan oluşan dört kademeli bir tedarik zincirinin benzetimini yapmaktadır. Oyunda, dört kişilik bir grubun her bir üyesi, bu dört tedarik zinciri kademesinin birinden sorumludur ve kendi kademesine ait envanteri verdiği siparişlerle kontrol etmektedir. Oyunun temel amacı, her bir kademeyi yöneten katılımcıların ortaya çıkardığı toplam grup maliyetini enküçüklemektir. Bu tezde, oyunun masa versiyonunun tam bir eşdeğeri olan bir matematiksel model kurulmuştur. Bu tezin temel amacı, benzer şekilde kontrol edilen tedarik zinciri kademelerinin varlığında, bir kademenin kontrolünün nasıl yapılması gerektiği konusunda anlayış geliştirmektir; sadece üzerine odaklanılan kademenin diğer kademelerden farklı davranış sergileyebildiği kabul edilmiştir. Özellikle, odaklanılan kademe dışındaki kademelerin kendi envanter veya birikmiş siparişlerini optimalin altında yönettikleri durumla ilgilenilmiştir. Burada iki farklı amaç olabilmektedir: (i) odaklanılan kademenin toplam maliyeti enküçüklenebilir, (ii) odaklanılan kademenin karar parametreleri eniyilenilerek tüm grubun toplam maliyeti enküçüklenebilir. Bu doğrultuda, bu tezde kontrol politikası olarak kullanılan *çapala-ve-düzelt sezgiseli*'nin parametreleri, diğer kademelerin karar parametreleri sabit tutularak odaklanılan kademe için eniyilenmiştir. Odaklanılan kademe için *stok düzeltme süresi* ve *hedef envanter* seviyesi eniyilenerek, çapala-ve-düzelt sezgiseli için farklı politika durumları elde edilmiştir. Genel olarak, tüm grubun maliyeti, odaklanılan kademenin maliyetinde artışa izin verilerek düşürülebilmektedir. Beklenmedik bir şekilde, odaklanılan kademenin toplam maliyeti yerine tüm grubun toplam maliyeti enküçüklendiğinde, en düşük grup maliyet değerleri toptancı seviyesinde elde edilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

α_s	1 / Stock Adjustment Time
β	Weight of Supply Line
θ	Smoothing Factor
$\| \ \|$	Rounding Function

LIST OF ACRONYMS/ABBREVIATIONS

B	Backlog
D	Distributor
EECD	Expected End-Customer Demand
EI	Effective Inventory
ENDCD	End-Customer Demand
EO	Expected Order
F	Factory
I	Inventory
I*	Desired Inventory
IA	Inventory Adjustment
IO	Incoming Orders
ITI1	In-Transit Inventory 1
ITI2	In-Transit Inventory 2
mdt	Mailing Delay Time
O	Orders
plt	Production Lead Time
PSR	Production Start Rate
R	Retailer
RL	Reinforcement Learning
S	Shipment
S*	Desired Stock
sat	Stock Adjustment Time
SD	System Dynamics
SL*	Desired Supply Line
SLA	Supply Line Adjustment
st	Shipment Time
t	Time
TC	Total Cost
TI	Temporary Inventory

ubc	Unit Backlog Cost
uihc	Unit Inventory Holding Cost
W	Wholesaler
WIPI1	Work in Process Inventory 1
WIPI2	Work in Process Inventory 2
wsl	Weight of Supply Line

1. INTRODUCTION

The beer production-distribution game, in short “The Beer Game”, is a board game and was first introduced by Jay Forrester's System Dynamics (SD) research group of the Sloan School of Management at the Massachusetts Institute of Technology in the 1960s. The Beer Game is an application of SD modeling and simulation methodology, which is widely used in management education and aims to give an experience to the participants about the potential dynamic problems that can be encountered in supply chain management, such as oscillations and amplification of oscillations as one moves from downstream towards upstream echelons (Akkermans and Vos, 2003; Barlas, 2002; Chen and Samroengraja, 2000; Forrester, 1961; Forrester, 1971; Größler *et al.*, 2008; Sterman, 2000). A detailed description of the original beer game, which is widely played by numerous people with different educational backgrounds and is also used in scientific studies, is given in Sterman (1989) and Croson and Donohue (2006).

We list some of the work on The Beer Game to give an idea about the range of studies based on the game. Jacobs (2000) introduced the internet-based version of The Beer Game and reported that this version of the game significantly reduced the time required to play the game. According to him, the main reason for this difference is that in the board version of the game, participants manually keep the records of inventories and backlogs and calculate the total cost, but in the internet-based version of the game, the game software takes care of these calculations and does so in a faster and more accurate manner compared to human participants.

Day and Kumar (2010) used mobile phones to run the game and reported improved accuracy and speed due to the automated calculations. Independent from Jacobs' (2000) work, Samur *et al.* (2004 and 2005) developed a multi-player computerized version of The Beer Game. They first present verification runs to demonstrate that their model correctly represents the board game. Then they conclude that participants who played the board game were more successful than those who played the computerized version of the game. The potential causes for this result included the slower pace of progress of the board version, which gives more time to think about the order quantity, and the relatively more realistic environment of the board version.

Steckel *et al.* (2004) examined the effect of reduced cycle times and the effect of shared point-of-sale (POS) information among the supply chain members in The Beer Game. They reported that reduced cycle times lead to reduced costs, which is an expected result. They further reported an interesting result that the benefit of POS information sharing depends on the customer demand pattern. Chaharsooghi *et al.* (2008) proposed a reinforcement learning (RL) model for ordering policies in supply chains and used The Beer Game model as an experimental platform. Mosekilde and Laugesen (2007) conducted an extensive bifurcation analysis and showed that The Beer Game can produce complex dynamics. Thomsen *et al.* (1991) also showed that it is possible to obtain complex dynamics from The Beer Game, including hyperchaos.

The Beer Game is a four echelon supply chain consisting of a retailer, wholesaler, distributor, and factory; there is an inventory control problem for each one of these echelons. During the game, every participant in a group of four is responsible for one of the four echelons and manages the associated inventory by placing orders. A supply chain can be modeled as a series of connected stock management structures. Therefore, the structure of the game consists of four cascading stock management problems. The orders flow from downstream echelons towards upstream echelons and cases of beer flow in the opposite direction. The aim of the game is to minimize the accumulated total cost obtained by the participants of a group managing each echelon. The accumulated cost generated by each individual echelon is calculated at the end of the game by adding up all inventory holding and backlog costs obtained at the end of each simulated week (Sterman, 1989).

The main aim of this thesis is to develop an understanding about how one should control an echelon in The Beer Game in the presence of identical group members; we assume that only the participant managing the echelon of concern behaves different than the rest of the group. In this thesis, "identical group members" means that those group members have the same identical instance of the anchor-and-adjust ordering policy. We are specifically interested in the case where the echelons other than the selected one sub-optimally manage their individual inventories/backlogs. There can be two objectives: (i) the minimum cost for the echelon of concern can be obtained, (ii) the minimum group total cost can be obtained by optimizing the decision parameters of the selected echelon. The motivation for this study is the expectation that there can be a significant difference in the control for these two different objectives especially when the other three group members control their echelons in a suboptimal way.

To answer the research question, decision making processes of the computer simulated decision makers should be represented in the model. In his paper, Sterman (1989) reported and analyzed the results of 11 Beer Game trials. Sterman suggested a stock control ordering policy, namely the *anchor-and-adjust heuristic*, to be used in managing the level of a stock. According to the results reported in that paper, the proposed heuristic was a good representation of the participants' decision making processes. Therefore, we represent the decision making processes of the computer simulated participants (i.e., the echelon of concern and the rest of the three echelons) using the anchor-and-adjust heuristic. In this thesis, the parameters of the anchor-and-adjust heuristic are called "decision parameters" and the variables of the same heuristic are called "decision variables". We optimize the parameters of the anchor-and-adjust heuristic for the selected echelon by keeping the parameters of the anchor-and-adjust heuristic constant for the rest of the three positions. We carry out this optimization process for each one of the four echelons of the game, selecting them one by one. After we select an echelon for this optimization process, we change the relative weight given to the supply line compared to the stock in the control decisions of the other three echelons and obtain optimum parameter values of the anchor-and-adjust heuristic for the selected echelon. We extend the simulation experiments by changing the final simulated time. As a result, we obtain optimum parameter values of the anchor-and-adjust ordering policy for each echelon.

To carry out this research, a mathematical model that is an exact one-to-one replica of the original board version of The Beer Game was needed. Moreover, we decided to use a model that had equations organized and executed in exactly the same order as the 'five steps' of the board game. We believed that such a model would facilitate the verification of our results and also potentially contribute to the analysis and understanding of the board game. We were not able to find a computer model in the literature that provided such an exact replica of the board game. Therefore, we first constructed a generic mathematical model based on the descriptions of The Beer Game provided in Sterman (1989), which is given in full detail in Chapter 2, before we conduct the rest of the study. The difficulties faced in the model construction process are also mentioned in that chapter.

In Chapter 3, we give the descriptions of the *stock adjustment time* (*sat*), *weight of supply line* (*wsl*), *desired inventory* (I^*), and *smoothing factor* (θ), which are the main decision parameters of the anchor-and-adjust heuristic. We explain the potential results of the different values of these parameters on the dynamics of effective inventory, which is the difference between inventory and backlog level. In addition, we present the parameter settings for the experiments.

Although there are some modified versions of The Beer Game, the traditional Beer Game is still widely used in scientific studies. In the traditional (original, standard) Beer Game, there is a step-up increase in the end-customer demand from four cases to eight cases in week five. All the echelons are forced to give orders equal to four cases of beer in the first four simulated weeks. The weekly end-customer demand information is available only to the retailer. Moreover, each echelon knows only the order quantity of its own customer. The factory is the producer of the beer and it has an unlimited production capacity. There is also no production cost of beer. The duration of the game is set to 36 weeks. Note that the participants are informed that the game would last 50 weeks to prevent potential end-of-game effects. In addition, the group members do not collaborate, they are assumed to be decentralized. In Chapter 4, we use these standard settings of The Beer Game to conduct our experiments. In Chapter 5, we increase the final time of our simulation experiments to 144 weeks to validate or invalidate the results obtained from the standard beer game setting. In Chapter 6, the effect of sub-optimal control of a selected single echelon is investigated.

2. MATHEMATICAL MODEL OF THE BEER GAME

In his paper entitled "*Advancing the Art of Simulation in the Social Sciences*", Axelrod (1997) reports problems that were encountered in replicating simulation models described in other published work. According to Axelrod, some of the replication problems are caused by ambiguities, gaps, and errors in the model descriptions. Despite all the details provided by Sterman (1989), a significant effort was required to obtain a one-to-one mathematical replica of the board version of the game, and we experienced difficulties similar to the ones experienced by Axelrod (1997): (i) Sterman provided equations for the general stock management task, which can form a basis in obtaining The Beer Game equations. However, the exact equations for The Beer Game are not present in Sterman's paper, except for the ordering equation. (ii) There is an ambiguity in the tie-breaking rule used in rounding the values of the orders. Hence, we are forced to assume a tie-breaking rule in rounding the values. (iii) Expectation formation is assumed to be performed informally by a participant in his mind and, therefore, is not listed among the five steps of The Beer Game. However, in the mathematical model, the decision making process is also captured as a part of the model and, therefore, we have to determine its place among the steps of the game. (iv) There is an error regarding the conceptualization of the delay durations.

Although, The Beer Game is an application of SD methodology, a one-to-one SD model of the game cannot be directly obtained because the order of calculations followed in the game and the order of calculations followed in SD methodology will not match unless the order of calculations in the corresponding SD model is carefully altered by introducing additional variables to the model. This mismatch also contributes to the difficulty in obtaining a complete mathematical model of the game.

Axelrod (1997, pp. 20-21) noted that: "*Replication is one of the hallmarks of cumulative science. It is needed to confirm whether the claimed results of a given simulation are reliable in the sense that they can be reproduced by someone starting from scratch.*" To ease the simulation replications of our model, we provide an R code (R, 2013) of the mathematical model presented in this chapter. According to Axelrod (1997), validity, usability, and extendibility are the three goals of a simulation model. Accordingly, in Section 2.2, we shortly explain how the code given in Appendix A can be used in experimentation and how it can be used to create a single or multi-player beer game on a computer. In Sterman (1989), the anchor-and-adjust heuristic formulation that is suggested to be used in decision making and the anchor-and-adjust heuristic formulation that is used in modeling the participant behavior are slightly different. In Section 2.3, we clarify the issue about the conceptualization of the delay durations, which is important for the verification of the model that we developed. In Section 2.4, the model verification section, we explain the differences between the two formulations, provide updated equations for the anchor-and-adjust heuristic formulation that is used in modeling the participant behavior, execute the corresponding R code given in Appendix B with the optimum benchmark parameter values given by Sterman, and obtain the exact same benchmark cost values presented in Sterman (1989).

2.1. The Structure and the Equations of the Game

To conduct this research, we first constructed a mathematical model of The Beer Game based on a figure of the board game (see Figure 2.1 in this thesis), equations, the five steps of the game, and descriptions given in Sterman (1989).

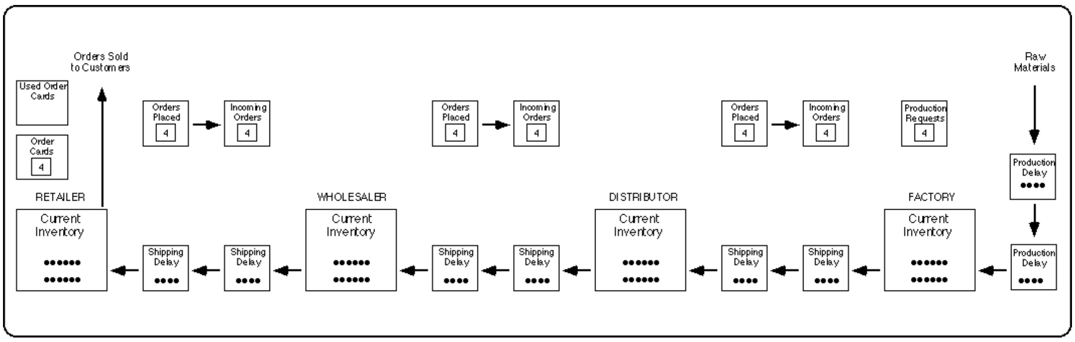

Figure 2.1. The Board of The Beer Distribution Game (Sterman, 1989).

2.1.1. Parameters and Initial Values of the Mathematical Model

$$sat_i = 1 \quad [week] \quad for \quad i = R, W, D, F \tag{2.1}$$

$$mdt_i = 1 \quad [week] \quad for \quad i = R, W, D \tag{2.2}$$

$$st_i = 2 \quad [week] \quad for \quad i = W, D, F \tag{2.3}$$

$$plt = 2 \quad [week] \tag{2.4}$$

Where *sat* ($1/\alpha_S$ in Sterman, 1989) stands for the *stock adjustment time, mdt* stands for the *mailing delay time, st* stands for the *shipment time*, and *plt* stands for the *production lead time*. *R, W, D,* and *F* stand, respectively, for the retailer, wholesaler, distributor, and factory echelons (Figure 2.1). Note that *sat, wsl, θ,* and *I** are the decision parameters (Equations 2.1, 2.5, 2.6, and 2.10). The different sets of values of these parameters represent different instances of the anchor-and-adjust ordering policy. For the equivalency of the anchor-and-adjust ordering policy and order-up-to-S policy, see Appendix C.

$$wsl_i = 1 \quad [dimensionless] \quad for \quad i = R, W, D, F \tag{2.5}$$

$$\theta_i = 0.2 \quad [1/week] \quad for \quad i = R, W, D, F \tag{2.6}$$

wsl (*β* in Sterman, 1989) stands for the *weight of supply line* and *θ* (also *θ* in Sterman, 1989) stands for the *smoothing factor* used by each echelon in the game in forming expectations using the simple exponential smoothing forecasting method.

- *Stock adjustment time* (*sat*) determines the intended time to close the gap between the desired level of the stock and the current stock itself. In The Beer Game, *sat* represents the number of weeks in which a decision maker wants to bring his current inventory level to the desired level. Smaller values of *sat* results in aggressive corrections while higher values correspond to mild corrections.

- *Weight of supply line* (*wsl*) represents the relative importance given to the supply line compared to the main stock. In other words, *wsl* is the fraction of supply line considered in the control decisions (i.e., order decisions). When *wsl* is taken as one, the main stock and its supply line will be effectively reduced to a single stock that cannot oscillate (Barlas and Ozevin, 2004; Sterman, 1989 and Chapter 17 in 2000; Yasarcan and Barlas, 2005a and 2005b). However, a zero value of *wsl* means that supply line is totally ignored in decision-making process and it may potentially create an unstable stock behavior.

- *Desired inventory* (I^*) is another parameter of the anchor-and-adjust heuristic and it simply represents the target inventory level. In The Beer Game, the cost function is asymmetric; *unit backlog cost* is \$1.00/(case·week) while *unit inventory holding cost* is \$0.50/(case·week). Therefore, it is usually less costly to have a positive on-hand inventory than having a backlog. Comparatively speaking, a better control decreases the requirement for large values of I^* while a worse control increases this requirement.

- *Smoothing factor* (θ) is the main parameter of exponential smoothing forecasting method and it represents the weight given to recent observations in the forecasting process. Although smoothing-factor is one of the parameters of the anchor-and-adjust heuristic, its optimization is out of the scope of this thesis. Theoretically, θ can take a value between 0 and 1. A zero value of θ means no corrections in the forecasted values. On the other hand, when it is taken as one, the exponential smoothing method will be equivalent to a naive forecast. It may not be practical to use a randomly selected *smoothing factor* value, even if that value fall in the theoretical range. According to Gardner (1985), the *smoothing factor* of a simple exponential smoothing forecasting method should be between 0.1 and 0.3 in practice. As a reasonable value, we suggest using a *smoothing factor* of 0.2 in forecasting, which is the middle point of the range suggested by Gardner (1985). This value of *smoothing factor* also falls in the range of 0.01 and 0.3 that is suggested by Montgomery and Johnson (1976). Therefore, θ is taken as 0.2 for all the echelons of the game.

$$ENDCD_t = \begin{Bmatrix} 4, & t < 5 \\ 8, & t \geq 5 \end{Bmatrix} \quad [case/week] \tag{2.7}$$

In the equation above, *ENDCD* stands for the *end-customer demand*. To save space, the unit *case* is used to represent a case of beer.

$$EECD_0 = 4 \quad [case/week] \tag{2.8}$$

$$EO_{i,0} = 4 \quad [case/week] \quad for \quad i = R, W, D \tag{2.9}$$

EECD stands for the *expected end-customer demand,* which is assumed to be calculated by the retailer. *EO* represents *expected orders* calculated by the wholesaler, distributor, and factory echelons based on the orders they receive from their respective customers (i.e., the retailer's orders received by the wholesaler, the wholesaler's orders received by the distributor, and the distributor's orders received by the factory). A time index of zero is the initial value of that variable at the beginning of the simulation.

$$I_i^* = 0 \quad [case] \quad for \quad i = R, W, D, F \tag{2.10}$$

$$SL_{R,0}^* = EECD_0 \cdot (mdt_R + st_W) \quad [case] \tag{2.11}$$

$$SL_{W,0}^* = EO_{R,0} \cdot (mdt_W + st_D) \quad [case] \tag{2.12}$$

$$SL_{D,0}^* = EO_{W,0} \cdot (mdt_D + st_F) \quad [case] \tag{2.13}$$

$$SL_{F,0}^* = EO_{D,0} \cdot plt \quad [case] \tag{2.14}$$

*I** represents the *desired inventory,* and *SL** stands for the *desired supply line*.

$$B_{i,0} = 0 \quad [case] \quad for \quad i = R, W, D, F \tag{2.15}$$

$$I_{i,0} = 12 \quad [case] \quad for \quad i = R, W, D, F \tag{2.16}$$

$$ITI1_{i,0} = 4 \quad [case] \quad for \quad i = R, W, D \tag{2.17}$$

$$WIPI1_0 = 4 \quad [case] \tag{2.18}$$

$$ITI2_{i,0} = 4 \quad [case] \quad for \quad i = R, W, D \tag{2.19}$$

$$WIPI2_0 = 4 \quad [case] \tag{2.20}$$

Equations 2.15 through 2.20 represent initial backlogs, initial inventories, and initial in-transit inventories (i.e., the values of the state variables at week zero). *ITI2* (*in-transit inventory 2*) represents the shipping delay box just before the inventory box, and *ITI1* (*in-transit inventory 1*) represents the shipping delay box before that (see Figure 2.1 on page 6). The value of *ITI1* belonging to an echelon is shifted to *ITI2* of the same echelon after one simulated week. *ITI2* is added to the inventory (*I*) or subtracted from the backlog (*B*) after a week. Likewise, *WIPI1* and *WIPI2* stand for work-in-process inventories. *WIPI1* is the work-in-process inventory of the factory that will be shifted to *WIPI2* after a week and that will eventually reach to the factory's inventory. *WIPI2* is the work-in-process inventory that will be added to the factory's inventory (I_F) or subtracted from the backlog (B_F) after a week.

$$O_{i,1} = 4 \quad [case/week] \quad for \quad i = R, W, D \tag{2.21}$$

$$PSR_1 = 4 \quad [case/week] \tag{2.22}$$

$$IO_{i,1} = 4 \quad [case/week] \quad for \quad i = W, D, F \tag{2.23}$$

O_i stands for *orders* that are placed by echelon *i* (retailer, wholesaler, and distributor). *PSR* stands for the *production start rate*, which is the production order given by the factory itself. IO_i stands for the *incoming orders* (the box right to the box of orders placed in Figure 2.1) that are received by echelon *i*. The purchase orders (*O*) placed by the retailer, wholesaler, and distributor and the production orders (*PSR*) given by factory at week *t* are placed for week (*t* + 1). *Orders* placed at week (*t* + 1) by the retailer, wholesaler, and distributor become the *incoming orders* (*IO*), respectively, for the wholesaler, distributor, and factory at week (*t* + 2). Therefore, the *end-customer demand*

(*ENDCD*), which is given by Equation 2.7, *orders* (*O*), and *production start rate* (*PSR*) have no value at week zero, but they are assigned a value for the first time at week 1.

$$TC_{i,0} = 0 \quad [\$] \quad for \quad i = R, W, D, F \tag{2.24}$$

$$uihc = 0.5 \quad [\$/(week \cdot case)] \tag{2.25}$$

$$ubc = 1 \quad [\$/(week \cdot case)] \tag{2.26}$$

TC, *uihc*, and *ubc* stand for the *total cost* generated by an echelon, the *unit inventory holding cost*, and the *unit backlog cost*, respectively.

The remaining model equations are given in an order based on the steps of the game presented in Sterman (1989). This sequence should strictly be followed while performing calculations to ensure an accurate representation of the board version of The Beer Game (Figure 2.1).

After initializing the board, the game facilitator declares the time as week 1, and the game starts from Step 1. After the completion of all steps (at the end of Step 5), the facilitator adds one week to the current time, declares it, and the game continues repeating the same process.

2.1.2. Step 1: Receive Inventory and Advance Shipping Delays

In The Beer Game, cases of beer flow from right to left (i.e., from the upper echelon to the lower) and orders flow from left to right (i.e., from lower echelon to the upper). In the first step of the game, the in-transit inventory (work-in-process inventory for the factory) that is immediately to the right of an inventory is added to the inventory by the participants. After that, the contents of the rightmost in-transit inventory boxes are shifted to the near-right in-transit inventory boxes, and the contents of the rightmost work-in-process inventory box are shifted to the near-right work-in-process inventory box. As a result, the rightmost in-transit inventory boxes and *WIPI1* become empty.

$$TI_{i,t} = I_{i,t-1} + ITI2_{i,t-1} \quad [case] \quad for \quad i = R,W,D \tag{2.27}$$

$$TI_{F,t} = I_{F,t-1} + WIPI2_{t-1} \quad [case] \tag{2.28}$$

In the above equations, *TI* 's are temporary calculation variables used to represent the temporary values of inventories. In this study, the equality sign is used in assigning values to parameters and variables, and it does not imply a mathematical equality. Therefore, *TI*'s are not a must. However, by adding *TI*'s, we aim to prevent the potential ambiguity that may be observed in Equations 2.43-2.46. Otherwise, one would see the same *inventory* (*I*) variable on the left and right hand sides of the same equation.

$$ITI2_{i,t} = ITI1_{i,t-1} \quad [case] \quad for \quad i = R,W,D \tag{2.29}$$

$$WIPI2_{t} = WIPI1_{t-1} \quad [case] \tag{2.30}$$

$$ITI1_{i,t} = 0 \quad [case] \quad for \quad i = R,W,D \tag{2.31}$$

$$WIPI1_{t} = 0 \quad [case] \tag{2.32}$$

Equations 2.31 and 2.32 are redundant in the sense that the exclusion of these equations will not prevent the correct simulation of the mathematical model, but we present them in order to follow the same exact process as in the board version of the game.

2.1.3. Step 2: Fill Orders

In this step, each echelon calculates the amount of beer to be shipped to its customer (i.e., shipments from retailer to end customer, from wholesaler to retailer, from distributor to wholesaler, and from factory to distributor) by considering incoming orders from the customer, the backlog of orders, and the inventory of that echelon. After calculating *shipments* (*S*), each participant, except for the retailer, puts *shipments* to the boxes on their near left (i.e., $ITI1_R$, $ITI1_W$, $ITI1_D$, and *WIPI1* are updated).

$$S_{ENDC,t} = \begin{cases} TI_{R,t} \cdot (1/week), & TI_{R,t} \le B_{R,t-1} + ENDCD_t \cdot (1\,week) \\ (B_{R,t-1} + ENDCD_t) \cdot (1/week), & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.33}$$

$$S_{R,t} = \begin{cases} TI_{W,t} \cdot (1/week), & TI_{W,t} \le B_{W,t-1} + IO_{W,t} \cdot (1\,week) \\ (B_{W,t-1} + IO_{W,t}) \cdot (1/week), & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.34}$$

$$S_{W,t} = \begin{cases} TI_{D,t} \cdot (1/week), & TI_{D,t} \le B_{D,t-1} + IO_{D,t} \cdot (1\,week) \\ (B_{D,t-1} + IO_{D,t}) \cdot (1/week), & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.35}$$

$$S_{D,t} = \begin{cases} TI_{F,t} \cdot (1/week), & TI_{F,t} \le B_{F,t-1} + IO_{F,t} \cdot (1\,week) \\ (B_{F,t-1} + IO_{F,t}) \cdot (1/week), & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.36}$$

The shipment variables are flow variables in essence, and the unit of these variables is $[case/week]$. However, we use stock variables that have $[case]$ as their unit in calculating the shipment variables. Therefore, corrections in the units are necessary. Accordingly, $(1/week)$ and $(1\,week)$ are used in the shipment equations (Equations 2.33-2.36). These corrections have no effect on the numerical values, but they correct the units. To easily comprehend the issue, consider a car that traveled for one hour and covered 50 miles in this journey. In such a case, the average speed of the car during that one hour would be 50 miles per hour. Although the units of the distance covered by the car and its average speed are different, their numerical values are not. Note that, for similar reasons, we also use the same type of correction in many of the remaining equations.

$$ITI1_{i,t} = S_{i,t} \cdot (1\,week) \quad [case] \quad for \quad i = R, W, D \tag{2.37}$$

$$WIPI1_t = PSR_t \cdot (1\,week) \quad [case] \tag{2.38}$$

2.1.4. Step 3: Record Inventory or Backlog on the Record Sheet

After filling orders, participants either count their inventories if they have chips representing the cases of beer in their inventory boxes or calculate the backlogs if they fail to satisfy the totality of the past backlogs and the current orders received from their customers. They record the inventory or backlog on their record sheets. This is represented by the equations below:

$$B_{R,t} = B_{R,t-1} + \left(ENDCD_t - S_{ENDC,t}\right)\cdot\left(1\,week\right) \quad [case] \tag{2.39}$$

$$B_{W,t} = B_{W,t-1} + \left(IO_{W,t} - S_{R,t}\right)\cdot\left(1\,week\right) \quad [case] \tag{2.40}$$

$$B_{D,t} = B_{D,t-1} + \left(IO_{D,t} - S_{W,t}\right)\cdot\left(1\,week\right) \quad [case] \tag{2.41}$$

$$B_{F,t} = B_{F,t-1} + \left(IO_{F,t} - S_{D,t}\right)\cdot\left(1\,week\right) \quad [case] \tag{2.42}$$

$$I_{R,t} = TI_{R,t} - S_{ENDC,t}\cdot\left(1\,week\right) \quad [case] \tag{2.43}$$

$$I_{W,t} = TI_{W,t} - S_{R,t}\cdot\left(1\,week\right) \quad [case] \tag{2.44}$$

$$I_{D,t} = TI_{D,t} - S_{W,t}\cdot\left(1\,week\right) \quad [case] \tag{2.45}$$

$$I_{F,t} = TI_{F,t} - S_{D,t}\cdot\left(1\,week\right) \quad [case] \tag{2.46}$$

2.1.5. Expectation Formation

Expectation formation is assumed to be performed informally by a participant in his mind and, therefore, is not listed among the five steps of The Beer Game. Sterman (1989) modeled the expectation formation process using the simple exponential smoothing method (see Equation 9 in Sterman, 1989). This process is reflected by the equations given below:

$$EECD_t = EECD_{t-1} + \theta_R\cdot\left(ENDCD_t - EECD_{t-1}\right)\cdot\left(1\,week\right) \quad [case/week] \tag{2.47}$$

$$EO_{R,t} = EO_{R,t-1} + \theta_W\cdot\left(IO_{W,t} - EO_{R,t-1}\right)\cdot\left(1\,week\right) \quad [case/week] \tag{2.48}$$

$$EO_{W,t} = EO_{W,t-1} + \theta_D \cdot \left(IO_{D,t} - EO_{W,t-1}\right) \cdot \left(1\,week\right) \quad \left[case/week\right] \tag{2.49}$$

$$EO_{D,t} = EO_{D,t-1} + \theta_F \cdot \left(IO_{F,t} - EO_{D,t-1}\right) \cdot \left(1\,week\right) \quad \left[case/week\right] \tag{2.50}$$

2.1.6. Step 4: Advance the Order Slips

Orders (*O*) placed by the retailer, wholesaler, and distributor become *incoming orders* (*IO*), respectively, for the wholesaler, distributor, and factory after a week.

$$IO_{W,t+1} = O_{R,t} \quad \left[case/week\right] \tag{2.51}$$

$$IO_{D,t+1} = O_{W,t} \quad \left[case/week\right] \tag{2.52}$$

$$IO_{F,t+1} = O_{D,t} \quad \left[case/week\right] \tag{2.53}$$

2.1.7. Step 5: Place Orders

In the board version of The Beer Game, participants place orders and production requests in this last step. According to Sterman (1989), the decision making process of participants can be represented by using the anchor-and-adjust decision heuristic. The equations below (Equations 2.54-2.68) are all part of this heuristic, which finally results in orders and production requests. The retailer, wholesaler, and distributor place orders (Equations 2.65-2.67) and the factory decides on the production requests (Equation 2.68). *SL** stands for the *desired supply line*, *EI* stands for the *effective inventory*, *SL* stands for the *supply line*, *SLA* stands for the *supply line adjustment*, *IA* stands for the *inventory adjustment*, and *PSR* stands for the *production start rate*. The anchor of the anchor-and-adjust heuristic is the expected loss from the stock, which is $EECD_t$ in Equation 2.65, $EO_{R,t}$ in Equation 2.66, $EO_{W,t}$ in Equation 2.67, and $EO_{D,t}$ in Equation 2.68.

$$SL^*_{R,t} = EECD_t \cdot (mdt_R + st_W) \quad \left[case\right] \tag{2.54}$$

$$SL^{*}_{W,t} = EO_{R,t} \cdot (mdt_W + st_D) \quad [case] \tag{2.55}$$

$$SL^{*}_{D,t} = EO_{W,t} \cdot (mdt_D + st_F) \quad [case] \tag{2.56}$$

$$SL^{*}_{F,t} = EO_{D,t} \cdot plt \quad [case] \tag{2.57}$$

$$EI_{i,t} = I_{i,t} - B_{i,t} \quad [case] \quad for \quad i = R,W,D,F \tag{2.58}$$

$$SL_{R,t} = IO_{W,t+1} \cdot (1\, week) + B_{W,t} + ITI1_{R,t} + ITI2_{R,t} \quad [case] \tag{2.59}$$

$$SL_{W,t} = IO_{D,t+1} \cdot (1\, week) + B_{D,t} + ITI1_{W,t} + ITI2_{W,t} \quad [case] \tag{2.60}$$

$$SL_{D,t} = IO_{F,t+1} \cdot (1\, week) + B_{F,t} + ITI1_{D,t} + ITI2_{D,t} \quad [case] \tag{2.61}$$

$$SL_{F,t} = WIPI1_t + WIPI2_t \quad [case] \tag{2.62}$$

$$SLA_{i,t} = wsl_i \cdot (SL^{*}_{i,t} - SL_{i,t}) / sat_i \quad [case/week] \quad for \quad i = R,W,D,F \tag{2.63}$$

$$IA_{i,t} = (I^{*}_i - EI_{i,t}) / sat_i \quad [case/week] \quad for \quad i = R,W,D,F \tag{2.64}$$

$$O_{R,t+1} = \begin{cases} 4, & t < 5 \\ \begin{cases} \| EECD_t + IA_{R,t} + SLA_{R,t} \|, & \| EECD_t + IA_{R,t} + SLA_{R,t} \| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.65}$$

$$O_{W,t+1} = \begin{cases} 4, & t<5 \\ \begin{cases} \| EO_{R,t} + IA_{W,t} + SLA_{W,t} \|, & \| EO_{R,t} + IA_{W,t} + SLA_{W,t} \| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.66}$$

$$O_{D,t+1} = \begin{cases} 4, & t<5 \\ \begin{cases} \| EO_{W,t} + IA_{D,t} + SLA_{D,t} \|, & \| EO_{W,t} + IA_{D,t} + SLA_{D,t} \| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.67}$$

$$PSR_{t+1} = \begin{cases} 4, & t<5 \\ \begin{cases} \| EO_{D,t} + IA_{F,t} + SLA_{F,t} \|, & \| EO_{D,t} + IA_{F,t} + SLA_{F,t} \| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \quad \left[\frac{case}{week}\right] \tag{2.68}$$

All equations belonging to the five steps of the game (Equations 2.27-2.69) are calculated for the current simulated week, except for the equations for *incoming orders* (2.51-2.53), *orders* (2.65-2.67), and *production start rate* (2.68). The equations for *incoming orders*, *orders*, and *production start rate* are calculated for the next simulated week. In the board version of the game, *orders* can only be integers. The rounding function ($\| \ \|$) used in Equations 2.65-2.68 (and also Equations 2.72-2.75) reflects this aspect of the game. In rounding the values, we assume that the "round half away from zero" tie-breaking rule is used.

In the original game, the accumulated total cost of each echelon is calculated at the end of the game from the record sheet kept by the participant managing that echelon. In the mathematical model, the *total cost* is calculated at the end of each simulated week using the equation given below:

$$TC_{i,t} = TC_{i,t-1} + \left(uihc \cdot I_{i,t} + ubc \cdot B_{i,t}\right) \cdot \left(1\, week\right) \quad [\$] \quad for \quad i = R, W, D, F \tag{2.69}$$

After this final step, the simulated time is increased by one week and announced to the participants. The game continues by repeating the whole process starting from Step 1 until Step 5 of week 36 is completed. After the simulation ends, the main performance measure, which is *group total cost* (*GTC*), is calculated.

$$GTC = TC_{R,36} + TC_{W,36} + TC_{D,36} + TC_{F,36} \quad [\$] \tag{2.70}$$

2.2. R Code of the Mathematical Model as an Experimental Platform

The R code of the mathematical model is given in Appendix A, which is ready to be executed. As mentioned before, Equations 2.1, 2.5, 2.6, and 2.10 are decision parameters, and their different values represent different instances of the anchor-and-adjust ordering policy. Hence, one can change the values of these parameters in the code and re-execute it to obtain the resulting cost values. Furthermore, by simply writing the name of a variable, the weekly values of that variable can easily be obtained. In this way, the code serves as an experimental platform. To create different settings for a simulation experiment, one can also change the end-customer demand pattern given by Equation 2.7, the values of other parameters, and the initial values of the stock variables, but those changes will imply a diversion from the original setting of The Beer Game.

Simulation experiments described in the previous paragraph can also be conducted in another programming environment by re-writing the code using that programming language. One can also develop a single or multi-player beer game using a programming environment that supports user interface creation. In that case, the respective order equation should be removed from the code, and a user or multiple users would be asked to insert values for orders of that(those) echelon(s) instead.

2.3. A Discussion on Acquisition Lags

In The Beer Game, the acquisition lag is the summation of the *mailing delay time* and *shipment time* for the retailer, wholesaler, and distributor, and it is directly equal to the *production lead time* for the factory. In the game, orders placed by an echelon (i.e., the retailer, wholesaler, or distributor) at week t will reach the inventory of that echelon at week (t + 4) given that the supplier of that echelon has sufficient inventory to fulfill the order. For the factory, orders given at week t will be received at week (t + 3). Therefore, Sterman (1989) states many times in his paper that the acquisition lag for the retailer, wholesaler, and distributor is at least 4 weeks, and it is always 3 weeks for the factory. However, we claim that orders placed at Step 5 of week t are for week (t + 1) (see

Equations 2.21, 2.22, 2.65-2.68, and 2.72-2.75). Therefore, slightly changing the game, by placing orders at the beginning of Step 1 of week (*t* + 1) instead of placing them at the end of Step 5 of week *t*, will make no difference. Accordingly, in our mathematical model, the acquisition lags used in calculating the values of the *desired supply line* are 3, 3, 3, and 2 weeks for the retailer, wholesaler, distributor, and factory, respectively (see Equations 2.2-2.4, 2.11-2.14, and 2.54-2.57). In the desired supply line equations (2.11-2.14 and 2.54-2.57), which correspond to Equation 7 in Sterman (1989), using acquisition lags of 4 weeks (for the retailer, wholesaler, and distributor) and 3 weeks (for the factory) instead of 3 weeks (for the retailer, wholesaler, and distributor) and 2 weeks (for the factory) will create a steady-state error in the dynamics.

2.4. Verification of the Mathematical Model

After constructing a one-to-one model of The Beer Game, we entered the optimal decision parameter values suggested by Sterman (1989) into our model. These parameter values are 0, 1, and 1 for θ (*smoothing factor*; also θ in Sterman, 1989), *sat* (*stock adjustment time*; $1/\alpha_S$ in Sterman, 1989), and *wsl* (*weight of supply line*; β in Sterman, 1989), respectively, for all echelons. The other decision parameter given by Sterman is S', which is defined as I^* (*desired inventory*; S^* in Sterman, 1989) plus *wsl* times SL^* (*desired supply line*; SL^* in Sterman, 1989); see Equation 2.71 and the unnumbered S' equation in Sterman (1989, p. 334). Sterman gives the optimal values of S' as 28, 28, 28, and 20 for the retailer, wholesaler, distributor, and factory echelons, respectively.

$$S_i' = I_i^* + wsl_i \cdot SL_i^* \quad [case] \quad for \quad i = R, W, D, F \tag{2.71}$$

In our mathematical model, the SL^* values are dynamically updated as the expected orders from the customers change. Thus, S' should also be a variable. However, Sterman uses constant SL^* and S' values. If the S' value is used instead of separate I^* and SL^* values and if it is a constant, the order equations (Equations 2.65-2.68) become as follows:

$$O_{R,t+1} = \begin{cases} 4, & t < 5 \\ \begin{cases} \left\| \dfrac{EECD_t + \begin{pmatrix} S'_R - EI_{R,t} \\ - wsl_R \cdot SL_{R,t} \end{pmatrix}}{sat_R} \right\|, & \left\| \dfrac{EECD_t + \begin{pmatrix} S'_R - EI_{R,t} \\ - wsl_R \cdot SL_{R,t} \end{pmatrix}}{sat_R} \right\| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \left[\frac{case}{week}\right] \quad (2.72)$$

$$O_{W,t+1} = \begin{cases} 4, & t < 5 \\ \begin{cases} \left\| \dfrac{EO_{R,t} + \begin{pmatrix} S'_W - EI_{W,t} \\ - wsl_W \cdot SL_{W,t} \end{pmatrix}}{sat_W} \right\|, & \left\| \dfrac{EO_{R,t} + \begin{pmatrix} S'_W - EI_{W,t} \\ - wsl_W \cdot SL_{W,t} \end{pmatrix}}{sat_W} \right\| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \left[\frac{case}{week}\right] \quad (2.73)$$

$$O_{D,t+1} = \begin{cases} 4, & t < 5 \\ \begin{cases} \left\| \dfrac{EO_{W,t} + \begin{pmatrix} S'_D - EI_{D,t} \\ - wsl_D \cdot SL_{D,t} \end{pmatrix}}{sat_D} \right\|, & \left\| \dfrac{EO_{W,t} + \begin{pmatrix} S'_D - EI_{D,t} \\ - wsl_D \cdot SL_{D,t} \end{pmatrix}}{sat_D} \right\| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \left[\frac{case}{week}\right] \quad (2.74)$$

$$PSR_{t+1} = \begin{cases} 4, & t < 5 \\ \begin{cases} \left\| \dfrac{EO_{D,t} + \begin{pmatrix} S'_F - EI_{F,t} \\ - wsl_F \cdot SL_{F,t} \end{pmatrix}}{sat_F} \right\|, & \left\| \dfrac{EO_{D,t} + \begin{pmatrix} S'_F - EI_{F,t} \\ - wsl_F \cdot SL_{F,t} \end{pmatrix}}{sat_F} \right\| > 0 \\ 0, & otherwise \end{cases}, & otherwise \end{cases} \left[\frac{case}{week}\right] \quad (2.75)$$

After simulating our model with the optimum θ, *sat*, *wsl*, and S' values using the order equations (Equations 2.72-2.75), we obtained the exact same benchmark cost values reported by Sterman, which supports our claim that our model is an exact representation of The Beer Game. Note that, the conceptual error regarding the acquisition lags has no effect on the model used by Sterman in optimizing the parameters because S' is a constant; it is not dynamically calculated during the optimization runs. The R code of the mathematical model that is modified with and for the ordering equations (Equations 2.72-2.75) is given in Appendix B.

3. PROBLEM DEFINITION AND EXPERIMENTAL PARAMETERS

The main aim of this thesis is to develop an understanding about how one should control an echelon in The Beer Game for the different types of behaviors that can be shown by the other three group members, which we assume to have the same identical instance of the anchor-and-adjust ordering policy; only the participant managing the echelon of concern behaves different than the rest of the group. There can be two objectives: (i) the minimum cost for the echelon of concern can be obtained, (ii) the minimum group total cost can be obtained by optimizing the decision parameters of the selected echelon. The motivation for this study is the expectation that there can be a significant difference in the control for these two different objectives especially when the other three group members control their echelons in a suboptimal way.

We represent the decision making processes of the computer simulated participants (i.e., the echelon of concern and the rest of the three echelons) using the anchor-and-adjust heuristic. We optimize the parameters of the anchor-and-adjust heuristic for the selected echelon by keeping the parameters of the anchor-and-adjust heuristic constant for the rest of the three positions. We carry out this optimization process for each one of the four echelons of the game, selecting them one by one. After we select an echelon for this optimization process, we change the relative weight given to the supply line compared to the stock in the control decisions of the other three echelons and obtain optimum parameter values for the selected echelon. As a result, we obtain optimum parameter values for each echelon.

According to our observations, the dynamics obtained in 36 week simulations can remain incomplete that might have an effect on the optimum *sat* and I^* values and, as a result, the optimum values may not be valid in the long term. To eliminate these potential unwanted effects, a longer simulation time is selected and all experiments are re-conducted.

3.1. The Parameters of the Anchor-and-Adjust Heuristic

In this subsection, we present important decision parameters of the anchor-and-adjust heuristic and explain the meaning of different values of these parameters. Note that anchor-and-adjust heuristic is equivalent to order-up-to-S policy only for a selected set of values of these parameters (see Appendix C).

- *Stock adjustment time* (*sat*) determines the intended time to close the gap between the desired level of the stock and the current stock itself. In The Beer Game, *sat* represents the number of weeks in which a decision maker wants to bring his current inventory level to the desired level. Smaller values of *sat* results in aggressive corrections while higher values correspond to mild corrections.

- *Weight of supply line* (*wsl*) represents the relative importance given to the supply line compared to the main stock. In other words, *wsl* is the fraction of supply line considered in the control decisions (i.e., order decisions). When *wsl* is taken as one, the main stock and its supply line will be effectively reduced to a single stock that cannot oscillate (Barlas and Ozevin, 2004; Sterman, 1989 and Chapter 17 in 2000; Yasarcan and Barlas, 2005a and 2005b). However, a zero value of *wsl* means that supply line is totally ignored in decision-making process and it may potentially create an unstable stock behavior.

- *Desired inventory* (I^*) is another parameter of the anchor-and-adjust heuristic and it simply represents the target inventory level. In The Beer Game, the cost function is asymmetric; *unit backlog cost* is \$1.00/(case·week) while *unit inventory holding cost* is \$0.50/(case·week). Therefore, it is usually less costly to have a positive on-hand inventory than having a backlog. Comparatively speaking, a better control decreases the requirement for large values of I^* while a worse control increases this requirement.

- *Smoothing factor* (θ) is the main parameter of exponential smoothing forecasting method and it represents the weight given to recent observations in the forecasting process. Although smoothing-factor is one of the parameters of the anchor-and-adjust heuristic, its optimization is out of the scope of this thesis. Theoretically, θ can take a value between 0 and 1. A zero value of θ means no corrections in the forecasted values. On the other hand, when it is taken as one, the exponential smoothing method will be equivalent to a naive forecast. It may not be practical to use a randomly selected *smoothing factor* value, even if that value fall in the theoretical range. According to Gardner (1985), the *smoothing factor* of a simple exponential smoothing forecasting method should be between 0.1 and 0.3 in practice. As a reasonable value, we suggest using a *smoothing factor* of 0.2 in forecasting, which is the middle point of the range suggested by Gardner (1985). This value of *smoothing factor* also falls in the range of 0.01 and 0.3 that is suggested by Montgomery and Johnson (1976). Therefore, θ is taken as 0.2 for all the echelons of the game.

3.2. The Parameter Settings for the Three Identically Controlled Echelons

In our experiments, as we mentioned before, we focus only one of the echelons and assume that the control parameters for the other three echelons are identical. We do not experiment with the values of *sat* and I^*; the values that we use in all the experiments are *sat* = 1 (i.e., they aim to close the gap between their own desired inventory and current inventory in one week) and $I^* = 0$ (i.e. they aim to carry zero net inventory) for all the three echelons. The reason for selecting $I^* = 0$ is that if *EI* is zero for an echelon in a simulated week, that echelon produces no costs in that week. We use *wsl* as the experimental parameter assigning it the same value for all the three echelons during each set of simulation experiments for the optimization runs.

The reason for selecting *wsl* as the experimental parameter lies in the power of this parameter in creating different types of dynamics. The Beer Game is a discrete-time simulation. However, if The Beer Game were a continuous-time simulation, carrying out additional experiments by changing the values of *sat*, and I^* of the three identical echelons would not bring additional behavioral richness. In such a case, The Beer Game would consist of four continuous-time stock management tasks. To demonstrate the power of *wsl* as an experimental parameter, we focus only one of the isolated stock management tasks. In Figure 3.1, the different types of dynamics that can be generated by simply changing the value of *wsl* are presented for a simple stock management task in continuous time:

- Line 1 – Unstable oscillations around the desired level (*wsl* = *a*)
- Line 2 – Marginally stable oscillations around the desired level (*wsl* = *b*)
- Line 3 – Stable oscillations around the desired level (*wsl* = *c*)
- Line 4 – Stable approach to the desired level (*wsl* = 1)
- Line 5 – Overdamped (i.e., comparatively slow) approach to the desired level (*wsl* > 1)

When supply line delay is discrete, the values of *a*, *b*, and *c* will be different for the different parameter settings, but their relative values will always satisfy the following relationship:

$$0 \le a < b < c < 1 \tag{3.1}$$

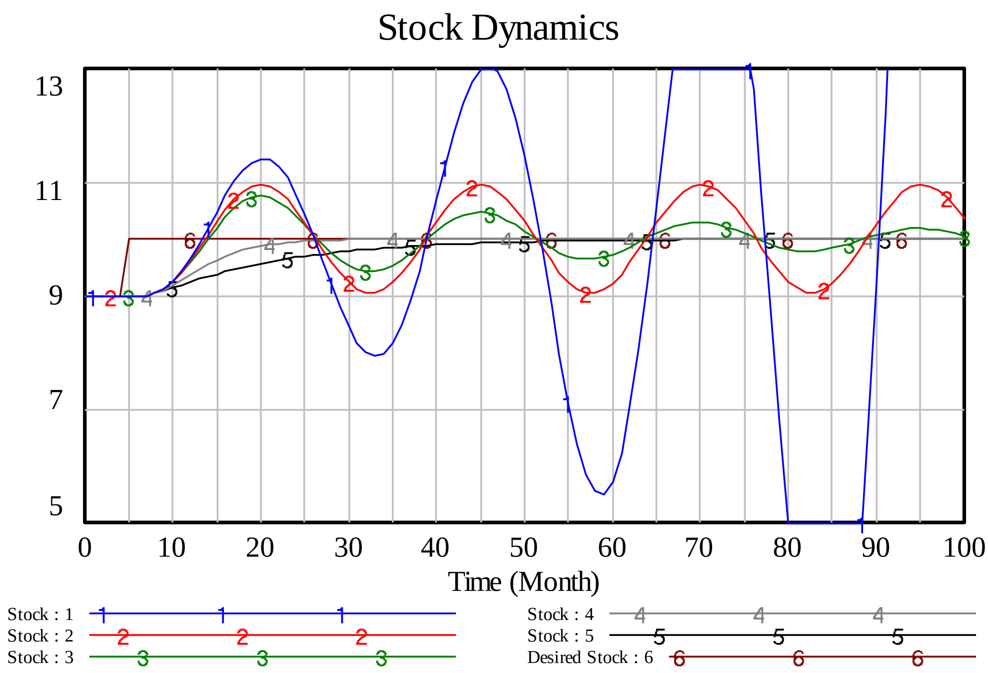

Figure 3.1. Stock dynamics with respect to different values of *wsl*.

It is possible to generate the aforementioned dynamics by assigning different values to *wsl* in a continuous-time model. However, in a discrete-time model, only *wsl* values between zero and one are meaningful. A *wsl* value greater than one cannot be used in a discrete-time model when *sat* is equal to one unit time because *supply line adjustment time*, which is sat/wsl, becomes less than one in this case. Note that, *supply line adjustment time* is an original parameter of the anchor-and-adjust heuristic. In our model, we prefer to replace *supply line adjustment time* with sat/wsl because the inclusion of *Weight of Supply Line* facilitates both the design of experiments and analysis of the results of those experiments. A *supply line adjustment time* value less than one unit time can potentially generate unexpected and meaningless behaviors similar to having a *stock adjustment time* value less than one unit time. In order to include the overdamped behavior in our experiments, we take *wsl* as unity for the three identical echelons and set their *sat* values to a value greater than one week. However, the results of the experiments with these settings show that the overdamped behavior of the other three echelons does not generate results different than the case where *wsl* and *sat* of the other three echelons are taken as unity and one week, respectively. Therefore, the runs for the overdamped behavior of the three identical echelons are excluded from this thesis.

As a side note, the level of the aforementioned experimental power of *wsl* would be lower if the acquisition lag (i.e., $mdt_R + st_W$, $mdt_W + st_D$, $mdt_D + st_F$, and *plt* for the retailer, wholesaler, distributor, and factory echelons, respectively) was lower and/or if the *sat* value was higher.

3.3. The Parameter Settings for the Selected Echelon

The optimum value of *wsl* is equal to unity for a discrete-delay single-stock-management problem (Mutallip, 2013). We assume that the echelon of concern uses this optimum value of *wsl* because, in our case, *wsl* equal to unity would either be optimum or very close to it. Moreover, when *wsl* is taken as unity for the echelon of interest, his inventory and supply line will be effectively reduced to a single stock (i.e., inventory), thus, he can adjust both his *supply line* and *inventory* only changing *stock adjustment time* values. However, adjusting *wsl* values instead of *stock adjustment time* has an effect only on *supply line adjustment* term in the control decisions.

There are two cases:

- We optimize *stock adjustment time* of the selected echelon by keeping *I** of the same echelon constant at zero cases.
- We optimize *I** of the selected echelon by keeping *stock adjustment time* of the same echelon constant at one week.

In all of the above optimization cases, optimizations are carried out with respect to other three participants' control behaviors (i.e., for the different *wsl* values they use). As we mentioned before, the echelon of interest makes optimizations under two different objectives: (i) The echelon of interest optimizes his own cost, (ii) the echelon of interest optimizes the group total cost. The aim in these experiments is to observe the differences and similarities of parameter and cost values under these two different objectives.

3.4. Simulation Experiments

The summary of the simulation experiments conducted is presented in Table 3.1. In each set of experiment, *wsl* values of the three identically controlled echelons changes from 0 to 1 with a step of 0.1 resulting in 176 (16×11) sub-set of experiments. For each one of these sub-sets, we carry out two different cases of optimization experiments for the echelon of concern:

- Case 1: optimize *sat*
- Case 2: optimize *I**

Table 3.1. Simulation experiments.

Set of Experiment	Final Time (weeks)		Echelons				Objectives	
	36	144	R	W	D	F	Own	*GTC*
1	X		X				X	
2	X		X					X
3	X			X			X	
4	X			X				X
5	X				X		X	
6	X				X			X
7	X					X	X	
8	X					X		X
9		X	X				X	
10		X	X					X
11		X		X			X	
12		X		X				X
13		X			X		X	
14		X			X			X
15		X				X	X	
16		X				X		X

4. RESULTS FOR THE STANDARD BEER GAME SETTING

In this chapter, we present observations for the standard setting of The Beer Game, which correspond to the set of experiments from one to eight (Table 3.1). We are interested in the decision making parameters of the echelon of concern and the related costs for the different control profiles (i.e., unstable oscillations, marginally stable oscillations, stable oscillations, and stable approach) of the three other echelons and for the two different objectives (i.e., we minimize the total cost of echelon of concern and we minimize *group total cost*).

4.1. Optimizing Stock Adjustment Time

In the optimization experiments, we assign integer values to *sat* of the selected echelon ranging from 1 week to 16 weeks. In addition, we also assign infinity to it, which implies no aimed corrections for the stock and supply line of that echelon. Then, we select a *sat* value that generates the minimum cost. In each sub-section, we present three different tables:

(i) The first table displays parameters and related cost values when we minimize the cost of selected echelon. First column gives *wsl* values used by the other three echelons**.** Second column gives optimum *sat* values for the echelon of concern. In the third column, we give *GTC* values. In the fourth, fifth, sixth, and seventh columns, we present individual total cost values of the echelons.

(ii) The second table displays parameters and related cost values when we optimize *group total cost* for the echelon of concern.

(iii) The third table compares the results presented in the first and second tables. First column gives *wsl* values used by the other three echelons**.** The second column gives optimum *sat* values when we minimize the cost of echelon of concern. The third column presents optimum *sat* values when we minimize *group total cost* by trying different *sat* values for the echelon of concern. In the fourth column, we present the percent increase in the total cost value of the echelon of concern when we change the objective from minimizing the cost of echelon of concern to minimizing *GTC*.

Finally, in the fifth column, we present the percent decrease in *GTC* when the we change the objective from minimizing the cost of echelon of concern to minimizing *GTC*.

4.1.1. Observations at the Retailer Echelon

When we minimize the retailer's total cost, the optimum value of *sat* for the retailer becomes one week for all *wsl* values used by the other three echelons (Table 4.1). This means that the retailer should close the gap between his *desired inventory* and inventory in one week.

Table 4.1. Optimum *sat* and corresponding cost values when we optimize the retailer's total cost.

wsl_W wsl_D wsl_F	Optimum sat_R (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	20816	605	3353.5	9369	7488.5
0.1	1	12668.5	585	2121.5	5753.5	4208.5
0.2	1	8248.5	561	1448.5	3632.5	2606.5
0.3	1	5666.5	536	1008	2425	1697.5
0.4	1	4263.5	530	771	1734.5	1228
0.5	1	3173	544	635	1149.5	844.5
0.6	1	2507.5	520	560	762	665.5
0.7	1	2236.5	530	527.5	625.5	553.5
0.8	1	1957.5	505	511	530	411.5
0.9	1	1567.5	456	425.5	399	287
1.0	1	1428.5	445	400	341	242.5

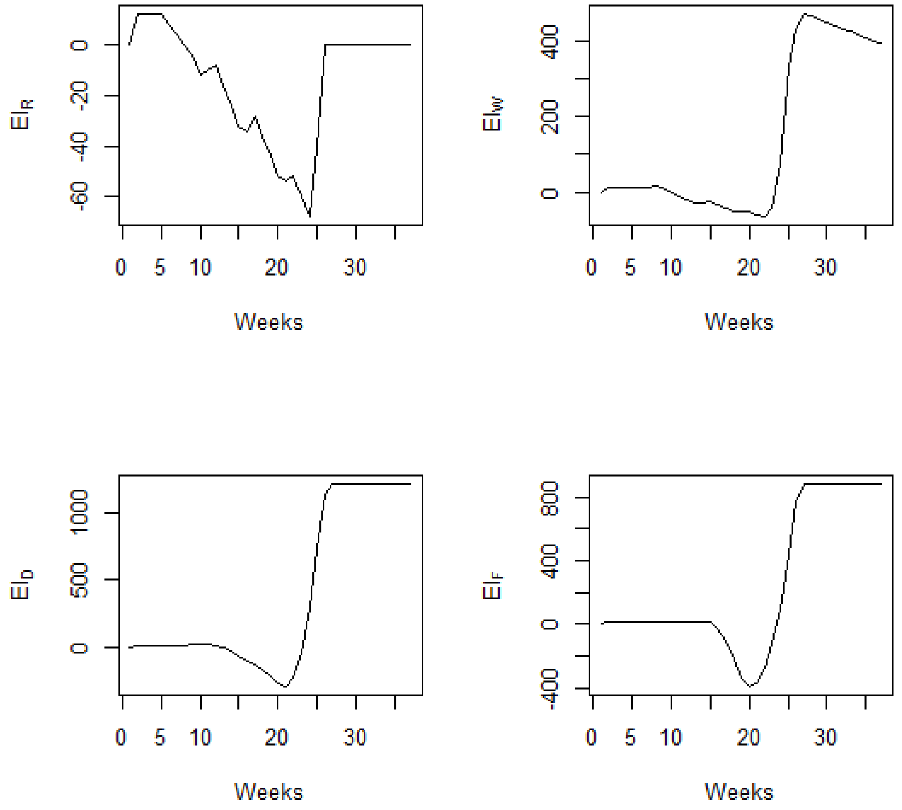

Figure 4.1. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F} = 0$ and $sat_R = 1$.

When we optimize *group total cost* by trying different *sat* values for the retailer, the optimum value of *sat* becomes infinity for *wsl* values between 0.0 and 0.6. This means that the retailer makes no adjustments in this range; he only gives orders equal to the expected value of *end-customer demand*. For *wsl* equals 0.7 and 0.8, the optimum *sat* becomes 16 weeks, which corresponds to mild adjustments (see Table 4.2).

Table 4.2. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the retailer.

wsl_W wsl_D wsl_F	Optimum sat_R (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	∞	15114.5	933	2470	6524	5187.5
0.1	∞	9266	913	1562.5	3973	2817.5
0.2	∞	5936.5	880	1035.5	2417.5	1603.5
0.3	∞	4196.5	862	720.5	1585.5	1028.5
0.4	∞	3298	860	585	1080	773
0.5	∞	2699.5	871	489	700.5	639
0.6	∞	2399	859	467.5	539.5	533
0.7	16	2129	708	465.5	534	421.5
0.8	16	1891	692	423	437.5	338.5
0.9	1	1567.5	456	425.5	399	287
1.0	1	1428.5	445	400	341	242.5

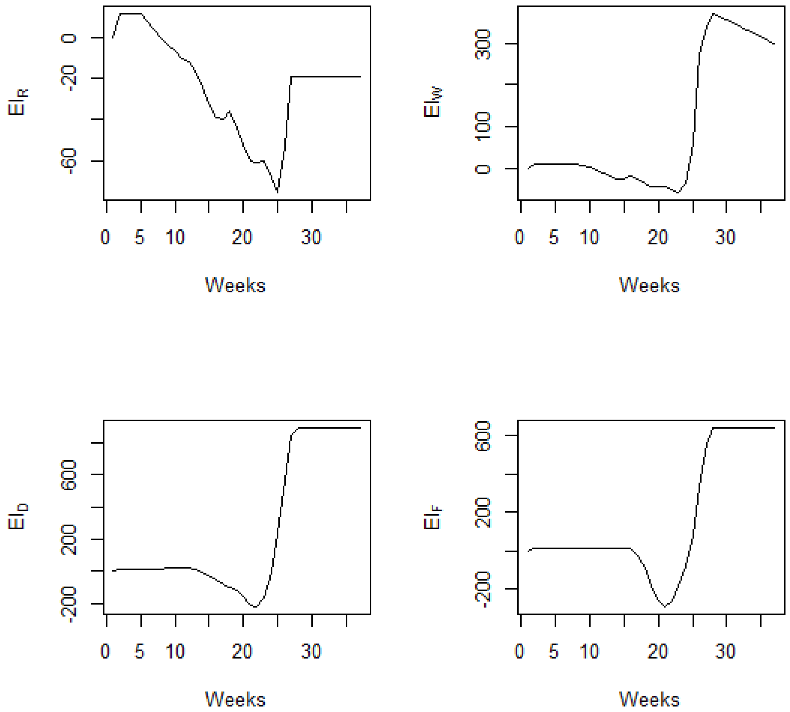

Figure 4.2. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F} = 0$ and $sat_R = \infty$.

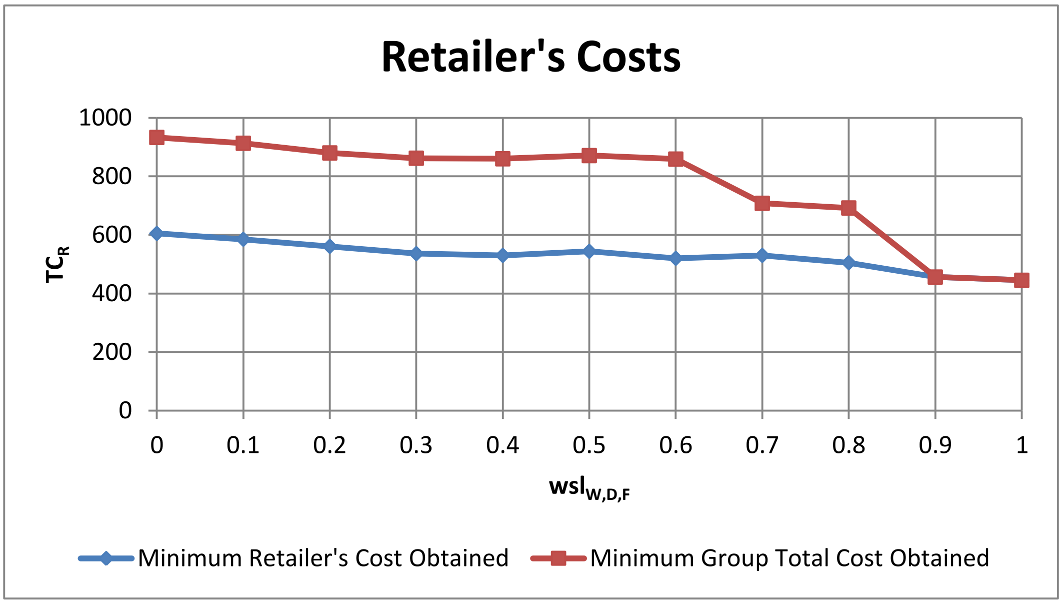

Figure 4.3. Retailer’s cost values for the two different objectives.

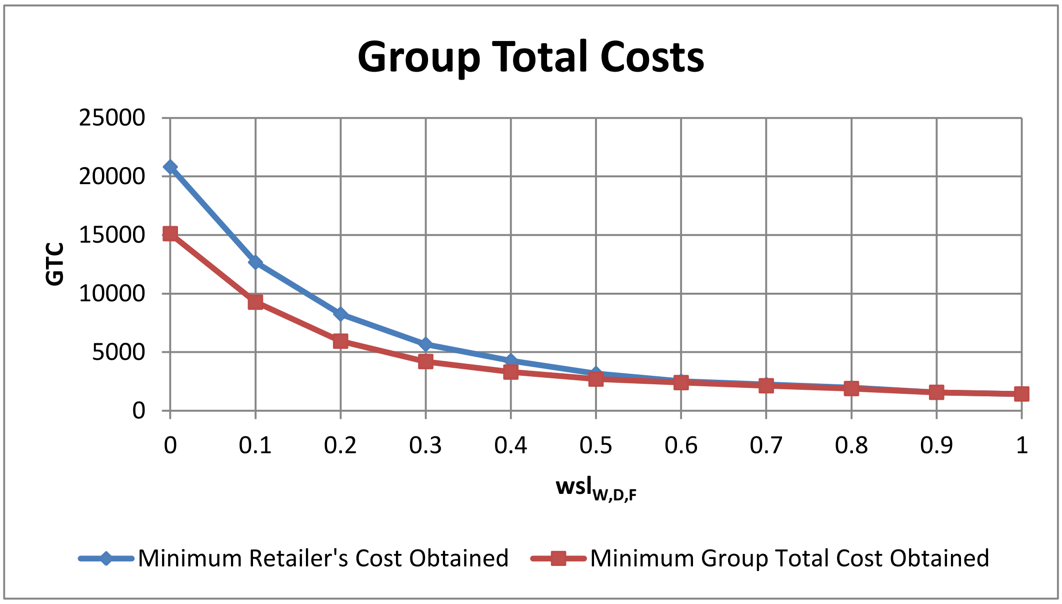

Figure 4.4. *Group total cost* values for the two different objectives.

In Figure 4.3, we observe that there is a difference in the retailer’s cost values for *wsl* between 0.0 and 0.8 under two different objectives (i.e., the minimum cost for the retailer can be obtained and the minimum *group total cost* can be obtained by optimizing the retailer’s *sat*). In Figure 4.4, one can observe a difference in the group total cost values too. We conclude that we can obtain lower *GTC* values (Figure 4.4) by sacrificing the objective of minimizing the retailer's total cost for *wsl* values less than or equal to 0.8. In both figures, one cannot observe a difference between the two lines of costs for *wsl* > 0.8.

Table 4.3. The percent changes in the optimum TC_R and in the optimum *GTC*.

wsl_W wsl_D wsl_F	sat_R for Obj. 1	sat_R for Obj. 2	Change in TC_R (%)	Change in *GTC* (%)
0.0	1	∞	54.21	-27.39
0.1	1	∞	56.07	-26.86
0.2	1	∞	56.86	-28.03
0.3	1	∞	60.82	-25.94
0.4	1	∞	62.26	-22.65
0.5	1	∞	60.11	-14.92
0.6	1	∞	65.19	-4.33
0.7	1	16	33.58	-4.81
0.8	1	16	37.03	-3.40
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

Table 4.3 presents the percent changes in the optimum retailer's total cost (TC_R) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the retailer's total cost" to "optimizing *group total cost* by optimizing the retailer's *sat*". For example, when *wsl* values for the wholesaler, the distributor and the factory are equal to zero, we can reduce *group total cost* by 27.39% and this reduction results in a 54.21% increase in the retailer's optimal total cost. The greatest reduction in the *group total cost* (-28.03%) is achieved for *wsl* = 0.2. We can reduce the group total cost by only 4.33% for *wsl* = 0.6. However, for the same *wsl* value, we need to increase the retailer's total cost by 65.19% to be able to achieve that relatively small improvement.

4.1.2. Observations at the Wholesaler Echelon

When we minimize the wholesaler's total cost, the optimum value of *sat* becomes equal to one week for all *wsl* values of the other three echelons (Table 4.4). In other words, taking *sat* as one week minimizes the wholesaler's total cost.

Table 4.4. Optimum *sat* and corresponding cost values when we optimize the wholesaler's total cost.

wsl_R wsl_D wsl_F	Optimum sat_W (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	24069	3570.5	3925	8328	8245.5
0.1	1	14013.5	2130	2502	4907.5	4474
0.2	1	9033.5	1467	1763.5	3073	2730
0.3	1	6212	1056	1372	1992.5	1791.5
0.4	1	4554.5	807	1124	1385	1238.5
0.5	1	3349	682.5	906	874	886.5
0.6	1	2666	592.5	763.5	690.5	619.5
0.7	1	2258.5	558	643	572.5	485
0.8	1	1991.5	509.5	565	507	410
0.9	1	1711	474.5	476	434.5	326
1.0	1	1428.5	445	400	341	242.5

When we optimize *group total cost* by trying different *sat* values for the wholesaler, we observe that making no stock and supply line adjustments minimizes *group total cost* until *wsl* is equal to 0.8. For *wsl* = 0.8, the optimum value of *sat* is 15 weeks. For *wsl* greater than or equal to 0.9, the optimum *sat* value becomes one week.

Table 4.5. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the wholesaler.

wsl_R wsl_D wsl_F	Optimum sat_W (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	∞	11785	2185	5331.5	1606.5	2662
0.1	∞	7081.5	1483	3081.5	1005.5	1511.5
0.2	∞	4781.5	1060.5	2133	732.5	855.5
0.3	∞	3727	812	1599	591	725
0.4	∞	3024.5	610	1238.5	506	670
0.5	∞	2430.5	530	972.5	427.5	500.5
0.6	∞	2142.5	544.5	832.5	368.5	397
0.7	∞	2013.5	630	841.5	283.5	258.5
0.8	15	1782.5	559	636.5	332.5	254.5
0.9	1	1711	474.5	476	434.5	326
1.0	1	1428.5	445	400	341	242.5

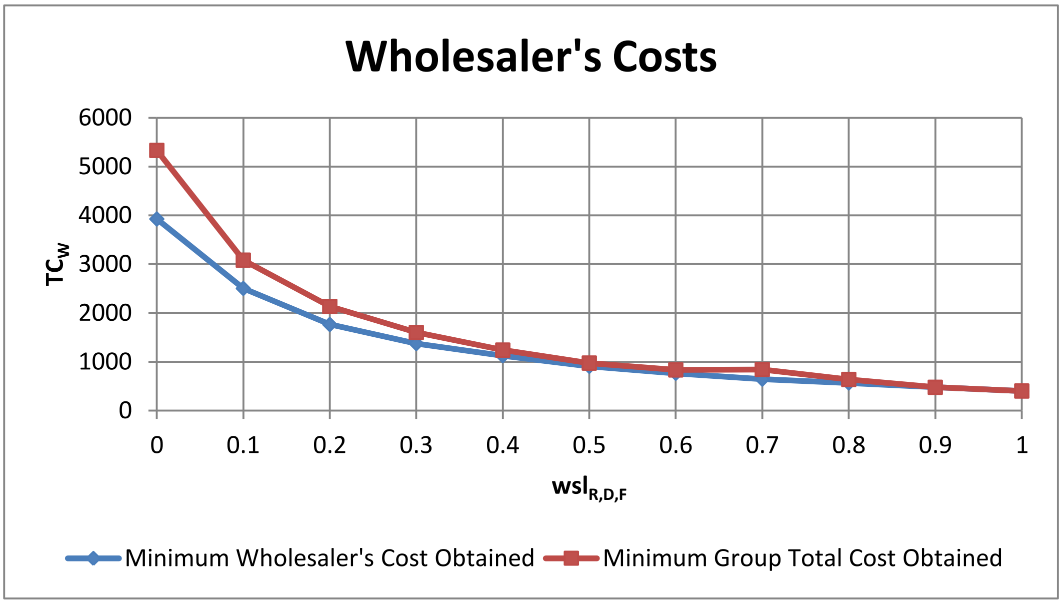

Figure 4.5. Wholesaler’s cost values for the two different objectives.

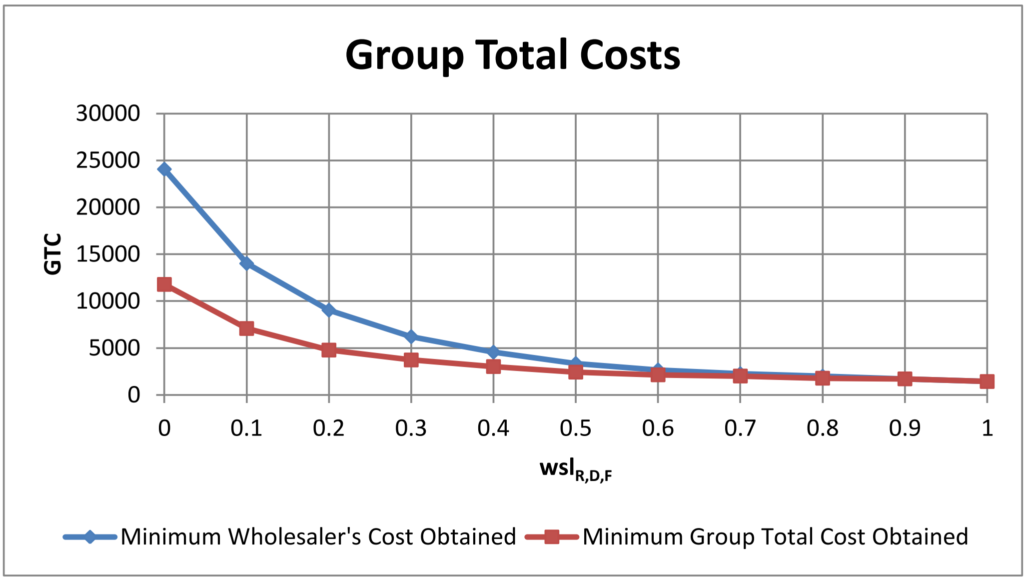

Figure 4.6. *Group total cost* values for the two different objectives.

In Figure 4.5, we observe that there is a difference in the wholesaler's cost values for *wsl* between 0.0 and 0.8 (i.e., the minimum cost for the wholesaler can be obtained and the minimum *group total cost* can be obtained by optimizing the wholesaler's *sat*). In Figure 4.6, one can observe a difference in the *group total cost* values too. We conclude that we can obtain lower *GTC* values by sacrificing the objective of minimizing the wholesaler's total cost for *wsl* values less than or equal to 0.8.

Table 4.6. The percent changes in the optimum TC_W and in the optimum *GTC*.

wsl_R wsl_D wsl_F	sat_W for Obj. 1	sat_W for Obj. 2	Change in TC_W (%)	Change in GTC (%)
0.0	1	∞	35.83	-51.04
0.1	1	∞	23.16	-49.47
0.2	1	∞	20.95	-47.07
0.3	1	∞	16.55	-40.00
0.4	1	∞	10.19	-33.59
0.5	1	∞	7.34	-27.43
0.6	1	∞	9.04	-19.64
0.7	1	∞	30.87	-10.85
0.8	1	15	12.65	-10.49
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

Table 4.6 presents the percent changes in the optimum wholesaler's total cost (TC_W) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the wholesaler's total cost" to "optimizing *group total cost* by optimizing the wholesaler's *sat*". For example, when *wsl* values for the retailer, the distributor and the factory are equal to zero, we can reduce *group total cost* by 51.04% and this reduction results in a 35.83% increase in the wholesaler's total cost. The greatest reduction in the group total cost (-51.04%) is achieved for *wsl* = 0.0. We can reduce the *group total cost* by only 10.49% for *wsl* = 0.8. However, for the same *wsl* value, we need to increase the wholesaler's total cost by 12.65% to be able to achieve that improvement.

4.1.3. Observations at the Distributor Echelon

When we minimize the distributor's total cost, the optimum value of *sat* for the distributor becomes one week for all *wsl* values of the other three echelons (Table 4.7).

Table 4.7. Optimum *sat* and corresponding cost values when we optimize the distributor's total cost.

wsl_R wsl_W wsl_F	Optimum sat_D (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	47432	3788.5	14884	14697	14062.5
0.1	1	23344.5	2279.5	7867	6976.5	6221.5
0.2	1	13525	1561.5	4594.5	4091	3278
0.3	1	8473.5	1106.5	2794.5	2625.5	1947
0.4	1	5732	838.5	1930	1804.5	1159
0.5	1	4148.5	705	1301	1310.5	832
0.6	1	3220	619.5	890	1045	665.5
0.7	1	2614.5	592	713.5	781	528
0.8	1	2221.5	538	622.5	609	452
0.9	1	1800.5	482.5	495	477	346
1.0	1	1428.5	445	400	341	242.5

When we minimize *group total cost* by trying different *sat* values for the distributor, the optimum value of *sat* becomes equal to infinity for *wsl* values between 0.1 and 0.7 (Table 4.8). Unexpectedly, the optimum value of *sat* is equal to one week for *wsl* = 0.0.

Table 4.8. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the distributor.

wsl_R wsl_W wsl_F	Optimum sat_D (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	47432	3788.5	14884	14697	14062.5
0.1	∞	20742.5	1741.5	4851.5	12464	1685.5
0.2	∞	11075	1205	3006	5996.5	867.5
0.3	∞	6463.5	876.5	1959.5	3136.5	491
0.4	∞	4472	703	1406	2037.5	325.5
0.5	∞	3318	595	1031.5	1430.5	261
0.6	∞	2671.5	524	807.5	1114.5	225.5
0.7	∞	2338	535.5	693	906.5	203
0.8	1	2221.5	538	622.5	609	452
0.9	1	1800.5	482.5	495	477	346
1.0	1	1428.5	445	400	341	242.5

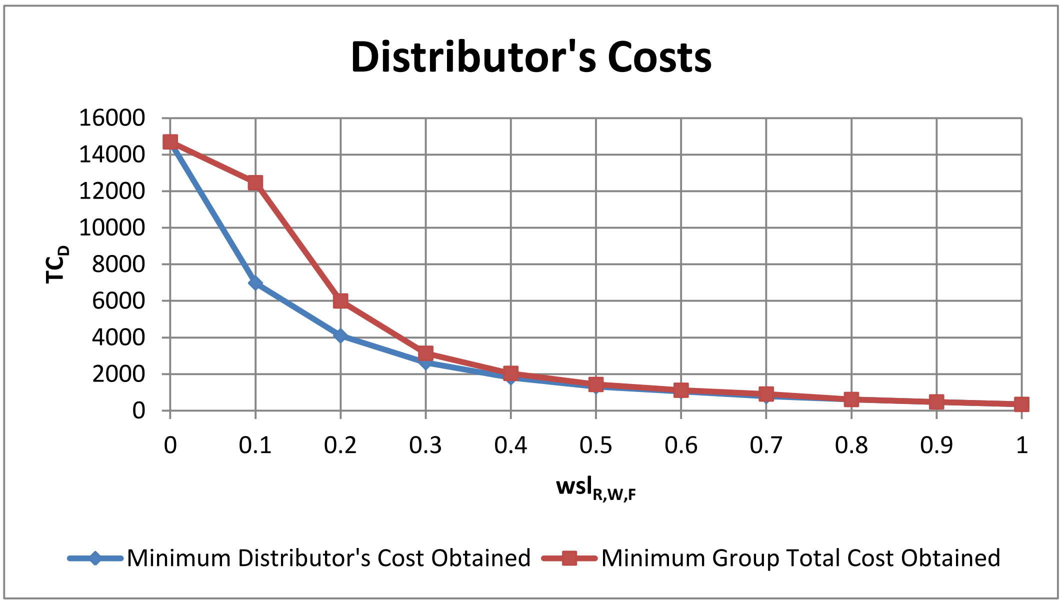

Figure 4.7. Distributor's cost values for the two different objectives.

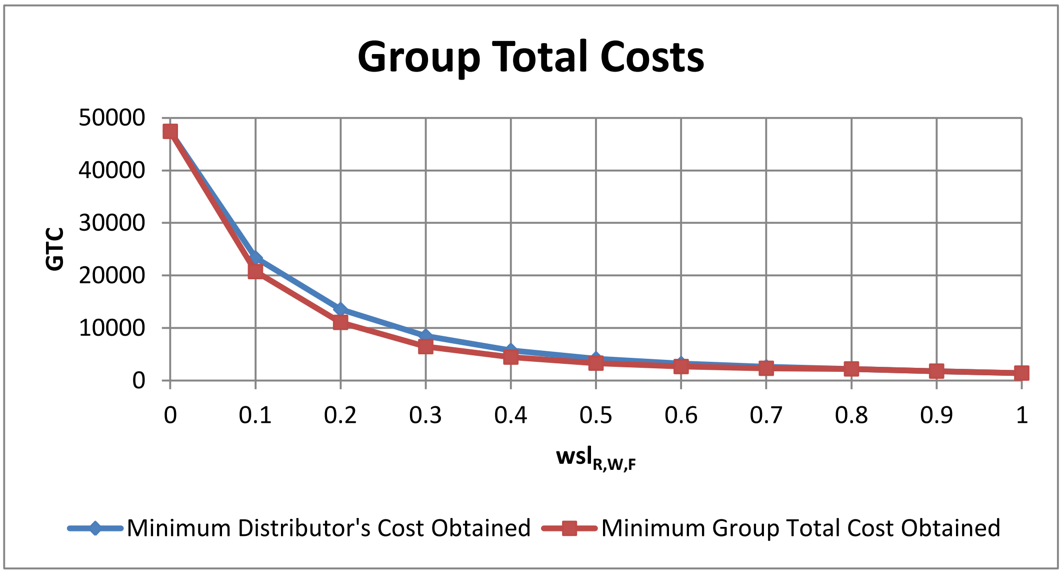

Figure 4.8. *Group total cost* values for the two different objectives.

In Figure 4.7, we observe that there is a difference in the distributor's cost values for *wsl* between 0.1 and 0.7 under two different objectives (i.e., the minimum cost for the distributor can be obtained and the minimum *group total cost* can be obtained by optimizing the distributor's *sat*). In Figure 4.8, one can observe a difference in the *group total cost* values too. We conclude that we can obtain lower *GTC* values by sacrificing the objective of minimizing the distributor's total cost for *wsl* values between 0.1 and 0.7.

Table 4.9. The percent changes in the optimum TC_D and in the optimum *GTC*.

wsl_R wsl_W wsl_F	sat_D for Obj. 1	sat_D for Obj. 2	Change in TC_D (%)	Change in *GTC* (%)
0.0	1	1	0.00	0.00
0.1	1	∞	78.66	-11.15
0.2	1	∞	46.58	-18.11
0.3	1	∞	19.46	-23.72
0.4	1	∞	12.91	-21.98
0.5	1	∞	9.16	-20.02
0.6	1	∞	6.65	-17.03
0.7	1	∞	16.07	-10.58
0.8	1	1	0.00	0.00
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

In Table 4.9, we present the percent changes in the optimum distributor's total cost (TC_D) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the distributor's total cost" to "optimizing *group total cost* by optimizing the distributor's *sat*". For example, when *wsl* values for the retailer, the wholesaler and the factory are equal to 0.1, we can reduce *group total cost* by 11.15% and this reduction results in a 78.66% increase in the distributor's optimal total cost. The greatest reduction in the group total cost (-23.72%) is achieved for *wsl* = 0.3.

4.1.4. Observations at the Factory Echelon

When we minimize the factory's total cost, the optimum value of *sat* becomes equal to one week for all *wsl* values of the other three echelons.

Table 4.10. Optimum *sat* and corresponding cost values when we optimize the factory's total cost.

wsl_R wsl_W wsl_D	Optimum sat_F (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	93486	3861	16105	41332	32188
0.1	1	40876.5	2346	8135	17659	12736.5
0.2	1	21776	1582.5	4785.5	8602.5	6805.5
0.3	1	12612.5	1163	2880.5	4730.5	3838.5
0.4	1	8030	875	1947	2827	2381
0.5	1	5410	713.5	1345	1768	1583.5
0.6	1	3879.5	630	920	1238	1091.5
0.7	1	2961	603.5	728.5	831	798
0.8	1	2371	545	635.5	642.5	548
0.9	1	1835.5	485	498.5	482	370
1.0	1	1428.5	445	400	341	242.5

When we minimize *group total cost* by trying different *sat* values for the factory, the optimum values of *sat* follow a different pattern. The optimum value of *sat* is one week when *wsl* values range from 0.0 and 1.0. Making smooth -almost no- adjustments especially for lower values of *wsl* minimizes *GTC* for the retailer, the wholesaler, and the distributor. However, making adjustments in a one-week period for $0.0 \leq wsl \leq 1.0$ minimizes *GTC* for the factory echelon.

Table 4.11. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the factory.

wsl_R wsl_W wsl_D	Optimum sat_F (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	93486	3861	16105	41332	32188
0.1	1	40876.5	2346	8135	17659	12736.5
0.2	1	21776	1582.5	4785.5	8602.5	6805.5
0.3	1	12612.5	1163	2880.5	4730.5	3838.5
0.4	1	8030	875	1947	2827	2381
0.5	1	5410	713.5	1345	1768	1583.5
0.6	1	3879.5	630	920	1238	1091.5
0.7	1	2961	603.5	728.5	831	798
0.8	1	2371	545	635.5	642.5	548
0.9	1	1835.5	485	498.5	482	370
1.0	1	1428.5	445	400	341	242.5

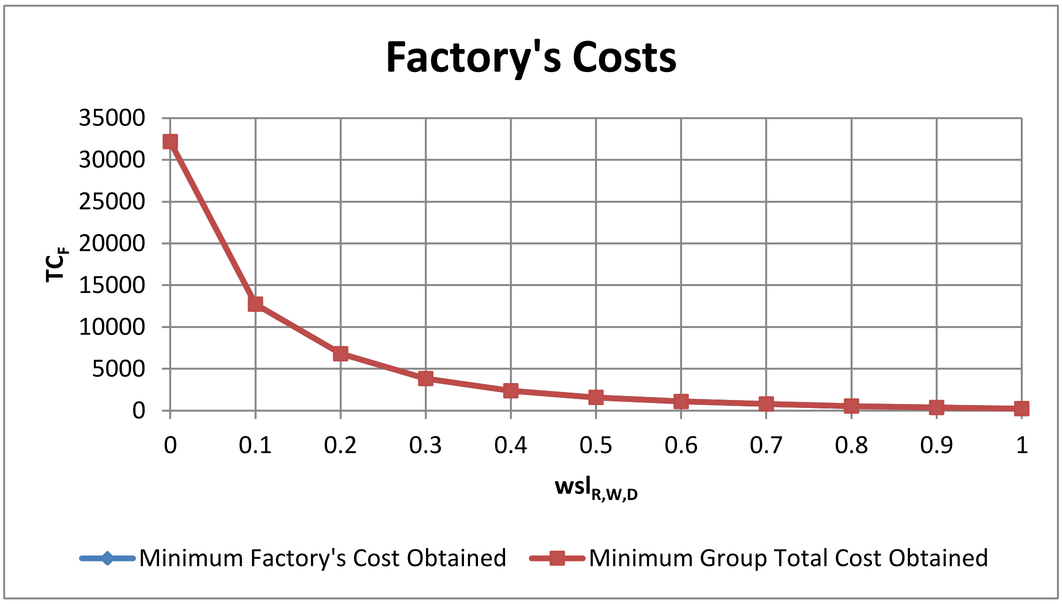

Figure 4.9. Factory’s cost values for the two different objectives.

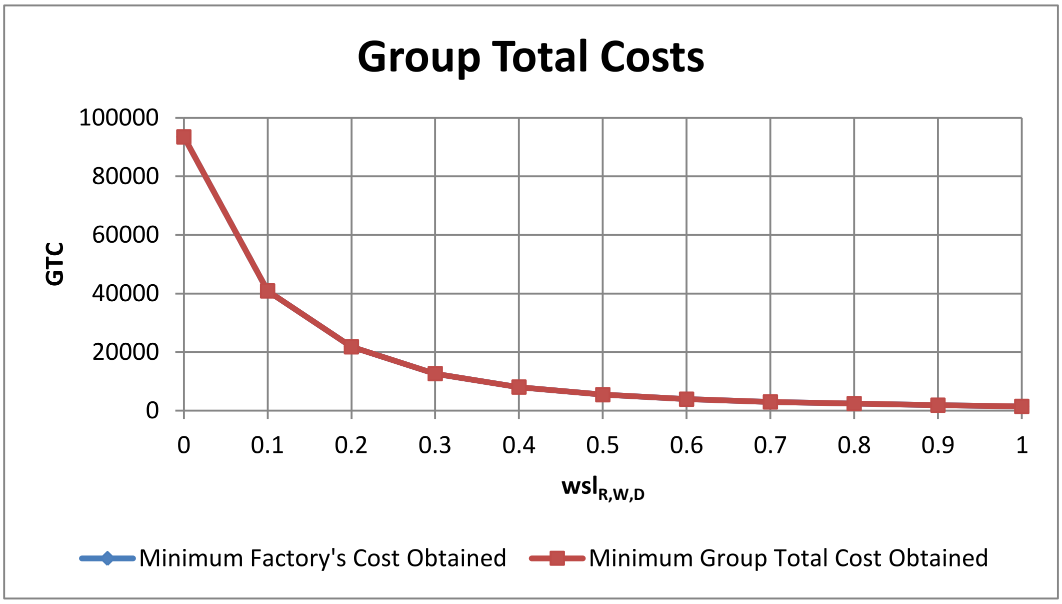

Figure 4.10. *Group total cost* values for the two different objectives.

For the factory echelon, the two different objectives have the same effect on costs for *wsl* values between 0.0 and 1.0.

Table 4.12. The percent changes in the optimum TC_F and in the optimum *GTC*.

wsl_R wsl_W wsl_D	sat_F for Obj. 1	sat_F for Obj. 2	Change in TC_F (%)	Change in GTC (%)
0.0	1	1	0.00	0.00
0.1	1	1	0.00	0.00
0.2	1	1	0.00	0.00
0.3	1	1	0.00	0.00
0.4	1	1	0.00	0.00
0.5	1	1	0.00	0.00
0.6	1	1	0.00	0.00
0.7	1	1	0.00	0.00
0.8	1	1	0.00	0.00
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

4.2. Optimizing Desired Inventory

Desired Inventory (*I**) is the target inventory level that a participant tries to maintain throughout the game. Normally, we expect *I** to be zero because if *EI* is zero for an echelon in a simulated week, that echelon produces no costs in that week. However, a positive *desired inventory* level might decrease the associated costs in the presence of non-optimal decision parameter values used by the other decision makers in the game. These decision parameters can be listed as *stock adjustment time* (*sat*) and *weight of supply line* (*wsl*). Non-optimal values of these parameters can potentially cause oscillations in the inventory and backlog levels. In The Beer Game, because of the asymmetry in the cost function, it is usually less costly to have a positive on-hand inventory than having a backlog.

In this sub-section, we present three different tables:

(i) The first table displays parameters and related cost values when our objective is to minimize the selected echelon's cost. First column gives *wsl* values used by the other three echelons**.** Second column gives optimum *I** values for the echelon of concern. In the third column, we give *group total cost* values. In the fourth, fifth, sixth, and seventh columns, we present individual total cost values of the echelons.

(ii) The second table displays parameters and related cost values when we optimize *group total cost* by trying different *I** values for the echelon of concern.

(iii) The third table compares the results presented in the first and second tables. First column gives *wsl* values used by the other three echelons**.** The second column gives optimum *I** values when we minimize the cost of the echelon of concern. The third column presents optimum *I** values when we minimize *group total cost* by trying different *I** values for the echelon of concern. In the fourth column, we present the percentage increase in the total cost value of the echelon of concern when we change the objective from minimizing the cost of echelon of concern to minimizing *GTC*. Finally, in the fifth column, we present the percentage decrease in *GTC* when we change the objective from minimizing the cost of echelon of concern to minimizing *GTC*.

4.2.1. Observations at the Retailer Echelon

When we minimize the retailer's total cost, the average optimum value of *I** for the retailer becomes 17.55 cases for $0.0 \leq wsl \leq 1.0$ (Table 4.13). In obtaining the optimum *I** values, we limit the search interval to [-50, 50] cases. We observe that the minimum *I** level is 15 cases and it is obtained when *wsl* is 0.3, 0.9, and 1.0. Besides this, the maximum *I** level is 22 cases and it is obtained when *wsl* is 0.4. Note that the optimum *I** values do not follow a regular pattern; there is no clear relationship between *wsl* values of the other three echelons and the optimum *I** values of the retailer.

Table 4.13. Optimum *I** and corresponding cost values when we optimize the retailer's total cost.

wsl_W wsl_D wsl_F	Optimum I_R^* (case)	GTC ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	20	23825	468.5	3817	10728	8811.5
0.1	18	13982	456.5	2287.5	6477	4761
0.2	16	8769.5	445	1505	3960	2859.5
0.3	15	5845.5	429.5	1017.5	2592.5	1806
0.4	22	4564.5	406.5	814.5	1879	1464.5
0.5	20	3163	381.5	641	1142.5	998
0.6	19	2419	375.5	556.5	780.5	706.5
0.7	17	2069	346.5	526.5	626.5	569.5
0.8	16	1752	326	489.5	517.5	419
0.9	15	1434	340.5	409	389.5	295
1.0	15	1263	343	370	315	235

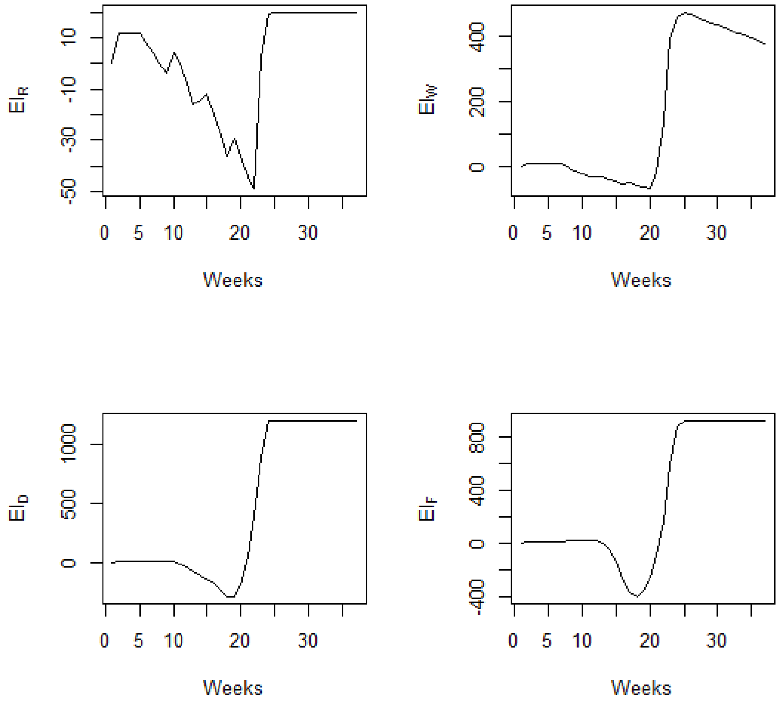

Figure 4.11. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F}$ = 0 and I^*_R = 20.

When we minimize *group total cost* by trying different *I** values for the retailer, the average optimum value of *I** for the retailer becomes -199.5 cases for 0.0 ≤ *wsl* ≤ 0.3 and 12.43 cases for 0.4 ≤ *wsl* ≤ 1.0 (Table 4.14). In obtaining the optimum *I** values, we limited the search interval to [-300, 300] cases. We observe that the minimum *I** level is -200 cases and it is obtained when *wsl* is 0.2 and 0.3. Besides this, the maximum *I** level is 17 cases and it is obtained when *wsl* is 0.8.

Although the need to carry backlog (i.e., negative *desired inventory* level) does not exactly correspond to the case where retailer makes no adjustments (i.e., *sat* = ∞), there still are similarities. By aiming to make its own net inventory negative, retailer aims to prevent the other three echelons' net inventories go below zero.

Table 4.14. Optimum I^* and corresponding cost values when we optimize *group total cost* by trying different I^* values for the retailer.

wsl_W wsl_D wsl_F	Optimum I_R^* (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	-199	4271.5	3336	337	322.5	276
0.1	-199	4270	3336	337	321	276
0.2	-200	4270	3338	334	322	276
0.3	-200	4269	3338	334	321	276
0.4	4	4072	489	773	1681.5	1128.5
0.5	12	3062.5	390.5	627	1095	950
0.6	13	2383	394	539.5	779	670.5
0.7	11	2045	375.5	509	617.5	543
0.8	17	1750.5	332.5	486	514.5	417.5
0.9	15	1434	340.5	409	389.5	295
1.0	15	1263	343	370	315	235

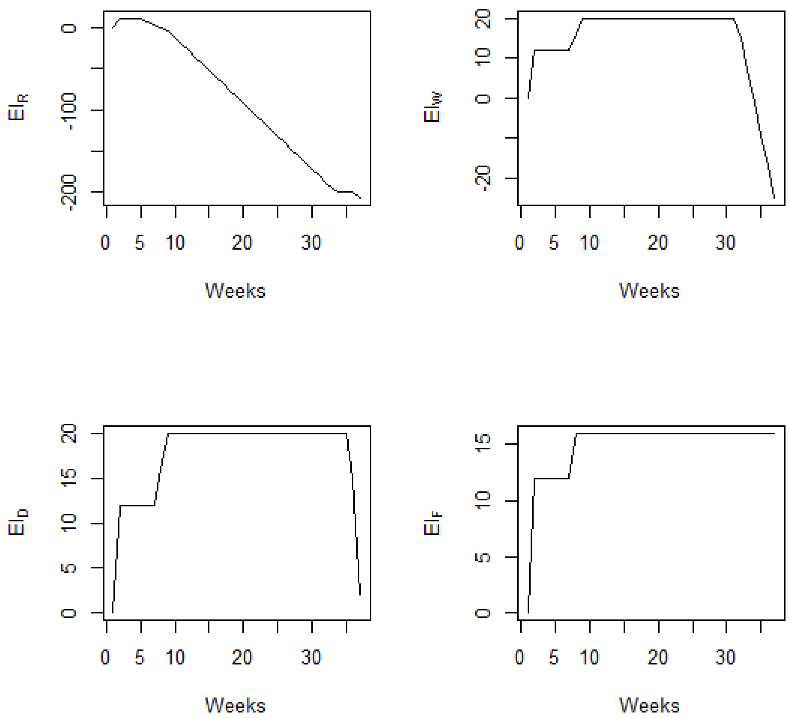

Figure 4.12. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F}$ = 0 and I_R^* = -199.

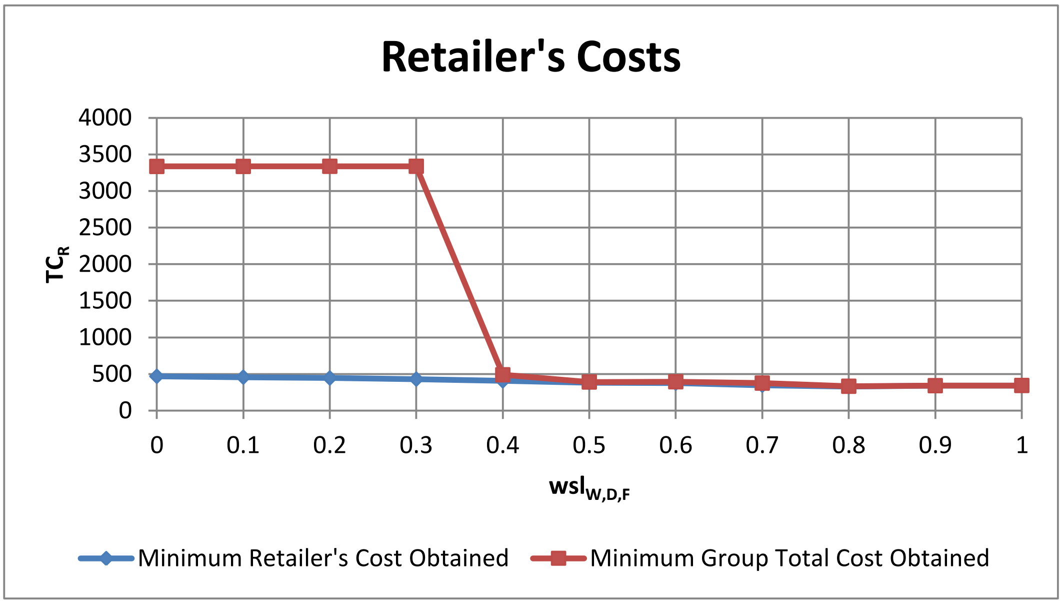

Figure 4.13. Retailer’s cost values for the two different objectives.

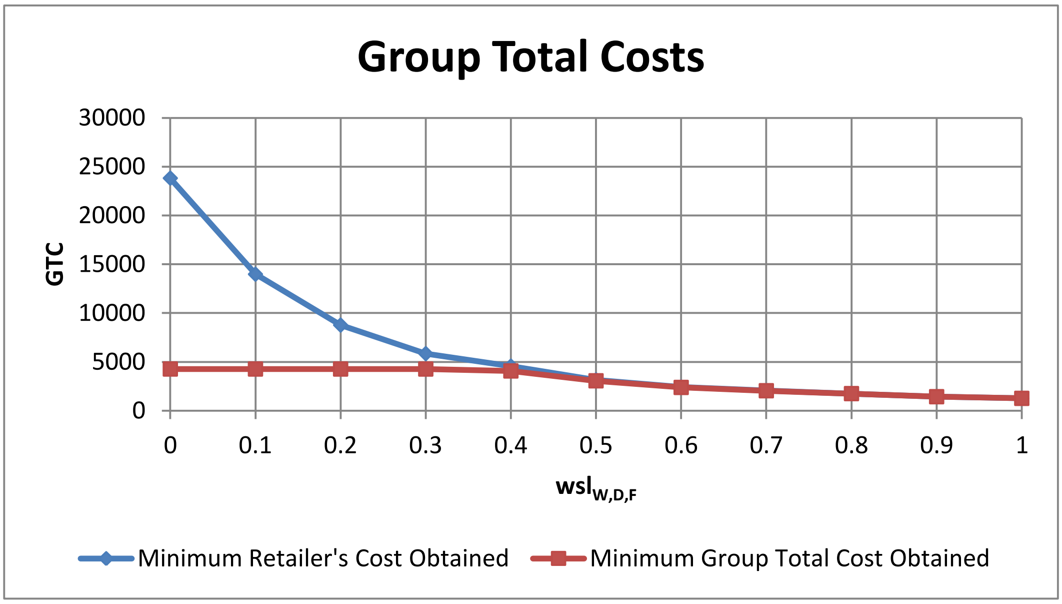

Figure 4.14. *Group total cost* values for the two different objectives.

In Figure 4.13, we observe that there is a significant difference in the retailer’s cost values for *wsl* between 0.0 and 0.3 under two different objectives (i.e., the minimum cost for the retailer can be obtained and the minimum *group total cost* can be obtained by optimizing the retailer’s I^*). In Figure 4.14, one can observe a significant difference in the *group total cost* values for the same range too. There are also relatively small differences that are not visible between the *group total cost* values obtained under the two different objectives (see Table 4.15). We conclude that we can obtain lower *GTC* values by sacrificing the objective of minimizing the retailer's total cost for *wsl* values between 0.0 and 0.3.

In Table 4.15, we present the percent changes in the optimum retailer's total cost (TC_R) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the retailer's total cost" to "optimizing *group total cost* by optimizing the retailer's *I**". For example, when *wsl* values for the wholesaler, the distributor, and the factory are equal to 0.0, we can reduce *group total cost* by 82.07% and this reduction results in a 612.06% increase in the retailer's total cost. The greatest reduction in the group total cost (-82.07%) is achieved for *wsl* = 0.0.

Table 4.15. The percent changes in the optimum TC_R and in the optimum *GTC*.

wsl_W wsl_D wsl_F	I_R^* for Obj. 1	I_R^* For Obj. 2	Change in TC_R (%)	Change in GTC (%)
0.0	20	-199	612.06	-82.07
0.1	18	-199	630.78	-69.46
0.2	16	-200	650.11	-51.31
0.3	15	-200	677.18	-26.97
0.4	22	4	20.30	-10.79
0.5	20	12	2.36	-3.18
0.6	19	13	4.93	-1.49
0.7	17	11	8.37	-1.16
0.8	16	17	1.99	-0.09
0.9	15	15	0.00	0.00
1.0	15	15	0.00	0.00

4.2.2. Observations at the Wholesaler Echelon

When we minimize the wholesaler's total cost, the average optimum value of *I** for the wholesaler becomes 30.73 cases for 0.0 ≤ *wsl* ≤ 1.0 (Table 4.16). In obtaining the optimum *I** values, we limit the search interval to [-50, 50] cases. We observe that the minimum *I** level is 21 cases and it is obtained when *wsl* = 1.0. Besides this, the maximum *I** level is 42 cases and it is obtained when *wsl* = 0.0.

When we minimize *group total cost* by trying different *I** values for the wholesaler, the average optimum value of *I** for the wholesaler becomes 40.1 cases for 0.0 ≤ *wsl* ≤ 1.0 (Table 4.17). In obtaining the optimum *I** values, we limit the search interval to [-300, 300] cases. We observe that the minimum *I** level is 28 cases and it is obtained when *wsl* is 0.7 and 1.0. Besides this, the maximum *I** level is 55 cases and it is obtained when *wsl* = 0.0 and 0.1.

Table 4.16. Optimum *I** and corresponding cost values when we optimize the wholesaler's total cost.

wsl_R wsl_D wsl_F	Optimum I_W^* (case)	GTC ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	42	7506	693.5	1276.5	1686	3850
0.1	41	5847	443	900	1481.5	3022.5
0.2	33	5010	356	661.5	1208	2784.5
0.3	31	3893.5	288	584	888.5	2133
0.4	31	2801	250.5	480	706	1364.5
0.5	31	2142.5	204.5	384	609.5	944.5
0.6	30	1648.5	148	371	466.5	663
0.7	30	1370	141.5	347.5	434	447
0.8	25	1244.5	177.5	350.5	379.5	337
0.9	23	1130	195	351.5	323	260.5
1.0	21	1045	203	337	288.5	216.5

Table 4.17. Optimum *I** and corresponding cost values when we optimize *group total cost* by trying different *I** values for the wholesaler.

wsl_R wsl_D wsl_F	Optimum I_W^* (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	55	4845	628	1420.5	1807	989.5
0.1	55	3742	436.5	1023	1420	862.5
0.2	54	3261.5	360	877.5	1087.5	936.5
0.3	52	2880.5	289	781.5	875	935
0.4	44	2598.5	269	622	708	999.5
0.5	37	2079	192	471	531.5	884.5
0.6	29	1633.5	148	378	461.5	646
0.7	28	1318.5	150	356.5	399.5	412.5
0.8	30	1156.5	135	362	349.5	310
0.9	29	1068.5	140	356	314	258.5
1.0	28	1004	142	343.5	286	232.5

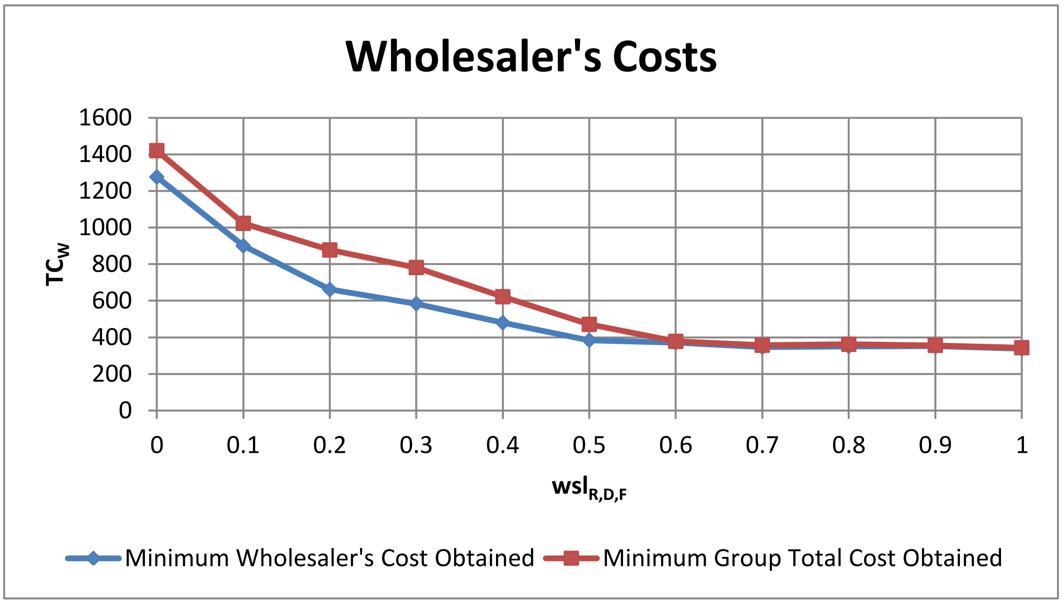

Figure 4.15. Wholesaler’s cost values for the two different objectives.

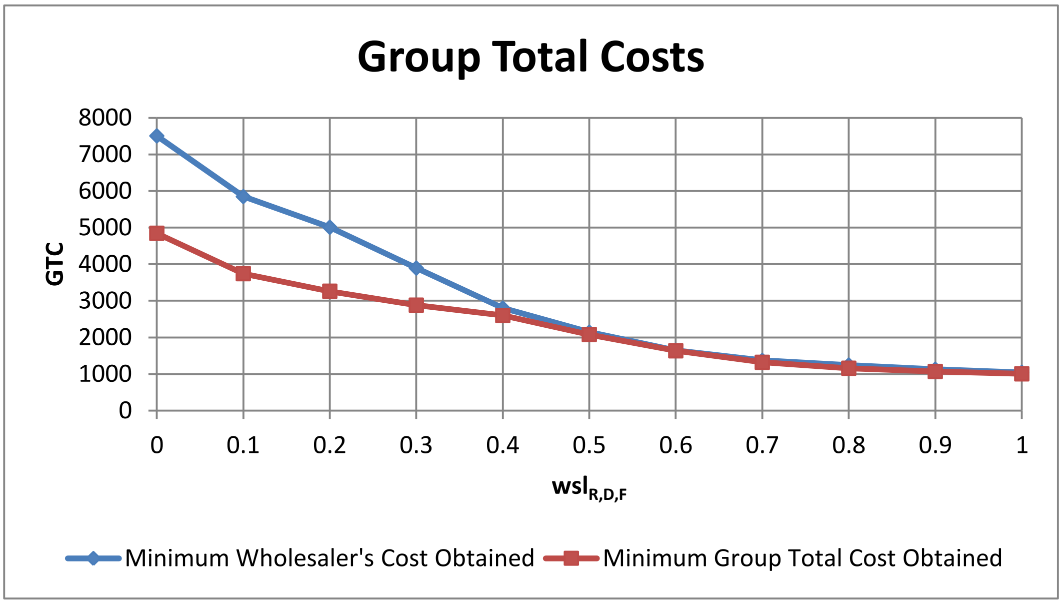

Figure 4.16. *Group total cost* values for the two different objectives.

In Figure 4.15, we observe that there is a difference in the wholesaler's cost values for *wsl* between 0.0 and 0.5 under two different objectives (i.e., the minimum cost for the wholesaler can be obtained and the minimum *group total cost* can be obtained by optimizing the wholesaler's *I**). In Figure 4.16, one can observe a significant difference in the *group total cost* values for *wsl* between 0.0 and 0.3. There are also relatively small differences that are not visible (see Table 4.18). We conclude that we can effectively obtain lower *GTC* values by sacrificing the objective of minimizing the wholesaler's total cost for *wsl* values between 0.0 and 0.3. In addition, we can also reduce *GTC* for $0.4 \leq wsl \leq 1.0$.

In Table 4.18, we present the percent changes in the optimum wholesaler's total cost (TC_W) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the wholesaler's total cost" to "optimizing *group total cost* by optimizing the wholesaler's *I**". For example, when *wsl* values for the retailer, the distributor, and the factory are equal to 0.0, we can reduce *group total cost* by 35.45% and this reduction results in a 11.28% increase in the wholesaler's optimal total cost. The greatest reduction in the group total cost (-36.00%) is achieved for $wsl = 0.1$.

Table 4.18. The percent changes in the optimum TC_W and in the optimum *GTC*.

wsl_R wsl_D wsl_F	I_W^* for Obj. 1	I_W^* for Obj. 2	Change in TC_W (%)	Change in GTC (%)
0.0	42	55	11.28	-35.45
0.1	41	55	13.67	-36.00
0.2	33	54	32.65	-34.90
0.3	31	52	33.82	-26.02
0.4	31	44	29.58	-7.23
0.5	31	37	22.66	-2.96
0.6	30	29	1.89	-0.91
0.7	30	28	2.59	-3.76
0.8	25	30	3.28	-7.07
0.9	23	29	1.28	-5.44
1.0	21	28	1.93	-3.92

4.2.3. Observations at the Distributor Echelon

When we minimize the distributor's total cost, the average optimum value of *I** for the distributor becomes 74.45 cases for $0.0 \leq wsl \leq 1.0$ (Table 4.19). In obtaining the optimum *I** values, we limit the search interval to [-300, 300] cases. We observe that the minimum *I** level is one case of beer and it is obtained when $wsl = 1.0$. Besides this, the maximum *I** level is 217 cases and it is obtained when $wsl = 0.0$.

When we minimize *group total cost* by trying different *I** values for the distributor, the average optimum value of *I** for the distributor becomes 105.64 cases for $0.0 \leq wsl \leq 1.0$ (Table 4.20). In obtaining the optimum *I** values, we limit the search interval to [-400, 400] cases. We observe that the minimum *I** level is 32 cases and it is obtained when *wsl* is 1.0. Besides this, the maximum *I** level is 372 cases and it is obtained when *wsl* is 0.0.

Table 4.19. Optimum *I** and corresponding cost values when we optimize the distributor's total cost.

wsl_R wsl_W wsl_F	Optimum I_D^* (case)	GTC ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	217	18059.5	1659	3877.5	4037	8486
0.1	145	11073.5	948.5	2405.5	2669.5	5050
0.2	114	7826	615.5	1606.5	2069.5	3534.5
0.3	89	5495.5	459.5	924	1583.5	2528.5
0.4	74	3903.5	360.5	597.5	1190	1755.5
0.5	58	2802.5	292.5	438	934.5	1137.5
0.6	48	2052	262.5	345	730.5	714
0.7	31	1746.5	303	340.5	558	545
0.8	27	1455	297	300.5	478	379.5
0.9	15	1460.5	374	357	415.5	314
1.0	1	1418.5	440	396	338	244.5

Table 4.20. Optimum *I** and corresponding cost values when we optimize *group total cost* by trying different *I** values for the distributor.

wsl_R wsl_W wsl_F	Optimum I_D^* (case)	GTC ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	372	16262	1697	3129.5	5876	5559.5
0.1	253	10696.5	978.5	1895.5	4067	3755.5
0.2	138	7791.5	637.5	1312	2351	3491
0.3	92	5490.5	462.5	897.5	1600	2530.5
0.4	73	3897	360	602.5	1192	1742.5
0.5	43	2730	285	440	1000	1005
0.6	43	2037.5	268	336.5	740.5	692.5
0.7	41	1575	226	249.5	583	516.5
0.8	37	1286.5	207.5	199	492	388
0.9	38	1118.5	190	162.5	480	286
1.0	32	1053	216	172.5	420.5	244

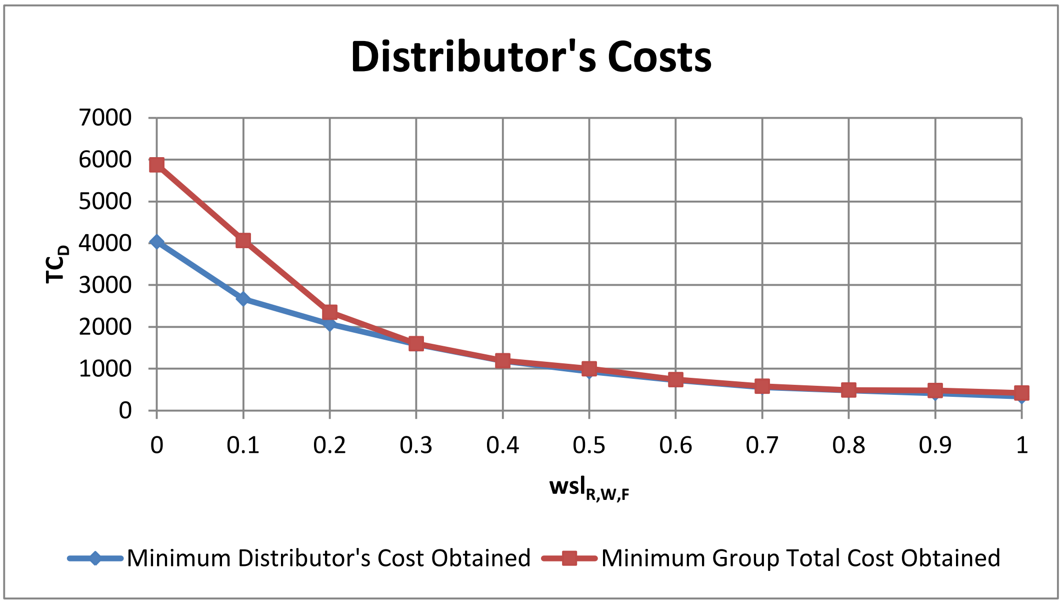

Figure 4.17. Distributor's cost values for the two different objectives.

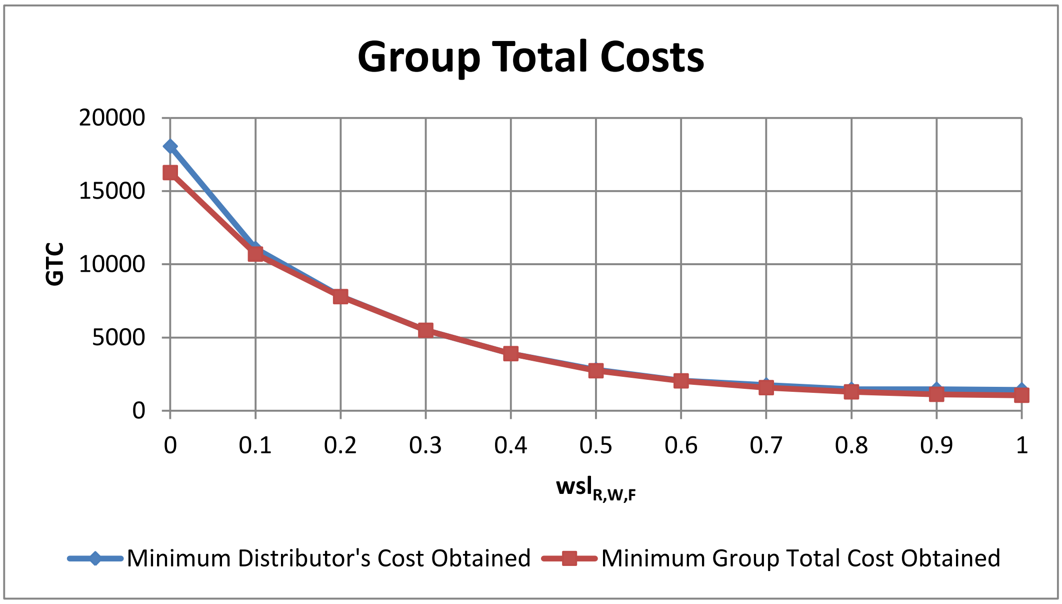

Figure 4.18. *Group total cost* values for the two different objectives.

In Figure 4.17, we observe that there is a significant difference in the distributor's total cost values for *wsl* between 0.0 and 0.2 under two different objectives (i.e., the minimum cost for the distributor can be obtained and the minimum *group total cost* can be obtained by optimizing the distributor's I^*). In Figure 4.18, one can observe some differences in the *group total cost* values. There are also relatively small differences that are not visible in the figure (see Table 4.21 for these differences). Different than the retailer and wholesaler echelons, we obtain significantly different *I** values for the distributor echelon under the two different objectives even for *wsl* values higher than or equal to 0.7.

Table 4.21. The percent changes in the optimum TC_D and in the optimum *GTC*.

wsl_R wsl_W wsl_F	I_D^* for Obj. 1	I_D^* for Obj. 2	Change in TC_D (%)	Change in GTC (%)
0.0	217	372	45.55	-9.95
0.1	145	253	52.35	-3.40
0.2	114	138	13.60	-0.44
0.3	89	92	1.04	-0.09
0.4	74	73	0.17	-0.17
0.5	58	43	7.01	-2.59
0.6	48	43	1.37	-0.71
0.7	31	41	4.48	-9.82
0.8	27	37	2.93	-11.58
0.9	15	38	15.52	-23.42
1.0	1	32	24.41	-25.77

In Table 4.21, we present the percent changes in the optimum distributor's total cost (TC_D) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the distributor's total cost" to "optimizing *group total cost* by optimizing the distributor's *I**". For example, when *wsl* values for the retailer, the wholesaler, and the factory are equal to 0.0, we can reduce *group total cost* by 9.95% and this reduction results in a 45.55% increase in the distributor's optimal total cost. Surprisingly, the greatest reduction in the group total cost (-25.77%) is achieved for *wsl* = 1.0.

4.2.4. Observations at the Factory Echelon

When we minimize the factory's total cost, the average optimum value of *I** for the factory becomes 331 cases for $0.0 \leq wsl \leq 0.4$ and -31 cases for $0.5 \leq wsl \leq 1.0$ (Table

4.22). In obtaining the optimum *I** values, we limit the search interval to [-1000, 1000] cases. We observe that the minimum *I** level is -74 cases and it is obtained when *wsl* = 0.6. Besides this, the maximum *I** level is 855 cases and it is obtained when *wsl* = 0.0.

Table 4.22. Optimum *I** and corresponding cost values when we optimize the factory's total cost.

wsl_R wsl_W wsl_D	Optimum I_F^* (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	855	48466	3136	10185.5	17762.5	17382
0.1	413	26289.5	1941.5	5356	9848.5	9143.5
0.2	224	16224.5	1240	3222	6209.5	5553
0.3	111	10421.5	797.5	2094	4035.5	3494.5
0.4	52	7037.5	604	1608.5	2643.5	2181.5
0.5	-66	5612	791	1429	1917	1475
0.6	-74	4175.5	719	1035.5	1437.5	983.5
0.7	-37	2992	627.5	789	945	630.5
0.8	-14	2390	559	634	664.5	532.5
0.9	5	1779.5	471	484	463	361.5
1.0	0	1428.5	445	400	341	242.5

When we minimize *group total cost* by trying different *I** values for the factory, the average optimum value of *I** for the factory becomes 228.55 cases for $0.0 \leq wsl \leq 1.0$ (Table 4.23). In obtaining the optimum *I** values, we limit the search interval to [-2000, 2000] cases. We observe that the minimum *I** level is 18 cases and it is obtained when *wsl* = 1.0. Besides this, the maximum *I** level is 1050 cases and it is obtained when *wsl* = 0.0.

Table 4.23. Optimum *I** and corresponding cost values when we optimize *group total cost* by trying different *I** values for the factory.

wsl_R wsl_W wsl_D	Optimum I_F^* (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1050	47684.5	3136	10359	15592.5	18597
0.1	502	25424	1941.5	5479.5	8493.5	9509.5
0.2	329	15678	1240	3350.5	4909.5	6178
0.3	252	10007.5	833	2131.5	2615.5	4427.5
0.4	127	6715	643.5	1289	2105.5	2677
0.5	67	4688.5	492	885	1540.5	1771
0.6	61	3466.5	457	651	945	1413.5
0.7	43	2739.5	449.5	580.5	659	1050.5
0.8	33	2097.5	442.5	498.5	480.5	676
0.9	32	1591	368	360	308.5	554.5
1.0	18	1325	370	325	266	364

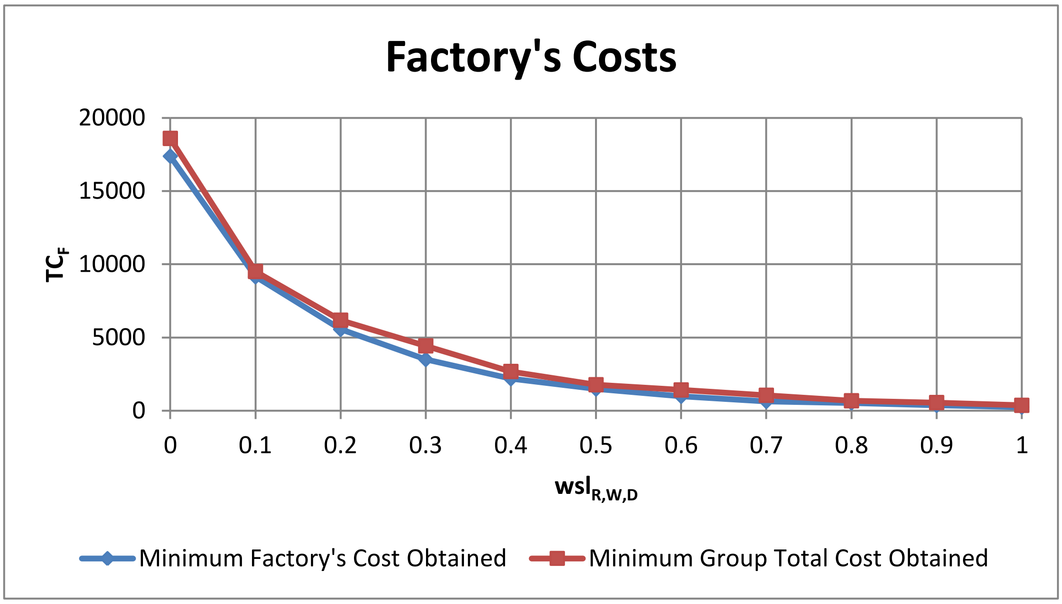

Figure 4.19. Factory’s cost values for the two different objectives.

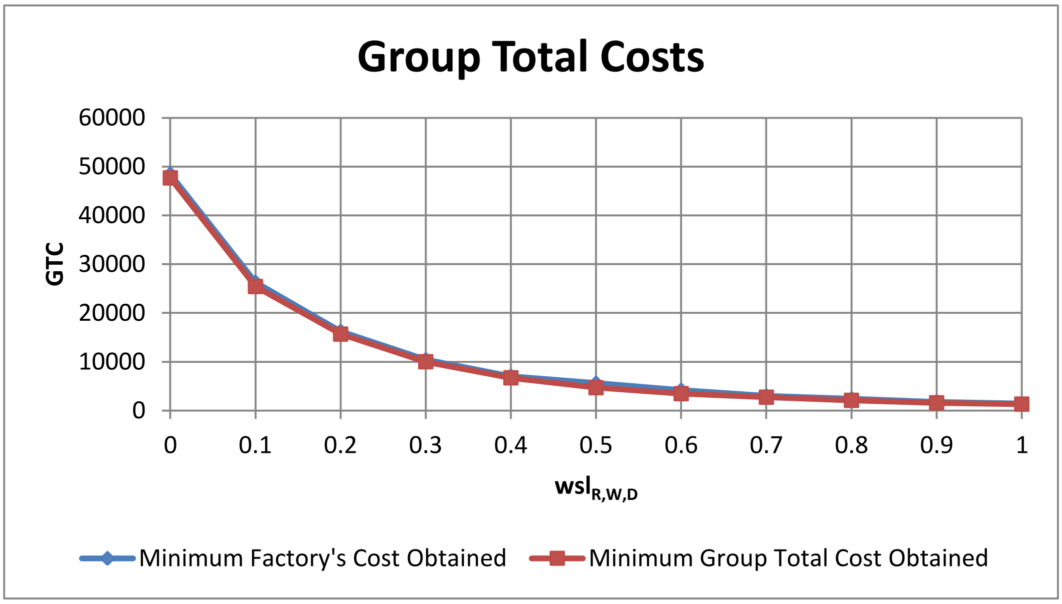

Figure 4.20. *Group total cost* values for the two different objectives.

In Figure 4.19 and Figure 4.20, we observe that there are some differences in the factory's total cost values and *GTC* values under two different objectives (i.e., the minimum cost for the factory can be obtained and the minimum *group total cost* can be obtained by optimizing the factory's *I**). We cannot change *GTC* values by optimizing the factory's *sat* value. However, we can reduce *GTC* by optimizing *I** values.

Table 4.24. The percent changes in the optimum TC_F and in the optimum *GTC*.

wsl_R wsl_W wsl_D	I_F^* for Obj. 1	I_F^* for Obj. 2	Change in TC_F (%)	Change in GTC (%)
0.0	855	1050	6.99	-1.61
0.1	413	502	4.00	-3.29
0.2	224	329	11.26	-3.37
0.3	111	252	26.70	-3.97
0.4	52	127	22.71	-4.58
0.5	-66	67	20.07	-16.46
0.6	-74	61	43.72	-16.98
0.7	-37	43	66.61	-8.44
0.8	-14	33	26.95	-12.24
0.9	5	32	53.39	-10.59
1.0	0	18	50.10	-7.25

In Table 4.24, we present the percent changes in the optimum factory's total cost (TC_F) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the factory's total cost" to "optimizing *group total cost* by optimizing the factory's *I**". For example, when *wsl* values for the retailer, the wholesaler, and the distributor are equal to 0.0, we can reduce *group total cost* by 1.61% and this reduction results in a 6.99% increase in the factory's optimal total cost. The greatest reduction in the group total cost (-16.98%) is achieved for *wsl* = 0.6.

4.3. Key Observations

In the previous sections of this chapter, the results obtained for each echelon are analyzed and presented in an isolated noncomparative fashion. In this section, a comparative analysis will be presented.

4.3.1. Key Observations in Stock Adjustment Time Optimization

When we minimize the retailer's, the wholesaler's, the distributor's, and the factory's total costs, *sat* equals one week becomes optimal (Table 4.25).

Table 4.25. Optimum *sat* values for the retailer, the wholesaler, the distributor, and the factory when we minimize their total costs by trying different *sat* values.

wsl	sat_R (week)	sat_W (week)	sat_D (week)	sat_F (week)
0.0	1	1	1	1
0.1	1	1	1	1
0.2	1	1	1	1
0.3	1	1	1	1
0.4	1	1	1	1
0.5	1	1	1	1
0.6	1	1	1	1
0.7	1	1	1	1
0.8	1	1	1	1
0.9	1	1	1	1
1.0	1	1	1	1

When we optimize *group total cost* by trying different *sat* values for the retailer, the wholesaler, and the distributor, we observe that making mild or no adjustments (i.e., they take *sat* as infinity.) for low values of *wsl* minimizes *group total cost*. This also means that they give orders equal to (or close to) the expected demands of their customers for these *wsl* values. However, taking *sat* as one week for *wsl* values between 0.0 and 1.0 for the factory minimizes *group total cost* (Table 4.26). Unlike the other echelons, behaving as aggressively as possible for inventory and supply line adjustments still minimizes *group total cost* for the factory when the objective changes from minimizing its echelon's total cost to minimizing *group total cost*. The main reason behind this behavior is the fact that the factory does not have a supplier. The retailer, the wholesaler, and the distributor can make their suppliers end with a backlog by making relatively aggressive corrections. However, the relatively aggressive corrections of the factory do not have such an impact.

Table 4.26. Optimum *sat* values for the retailer, the wholesaler, the distributor, and the factory when we minimize *group total cost* by trying different *sat* values.

wsl	sat_R (week)	sat_W (week)	sat_D (week)	sat_F (week)
0.0	∞	∞	1	1
0.1	∞	∞	∞	1
0.2	∞	∞	∞	1
0.3	∞	∞	∞	1
0.4	∞	∞	∞	1
0.5	∞	∞	∞	1
0.6	∞	∞	∞	1
0.7	16	∞	∞	1
0.8	16	15	1	1
0.9	1	1	1	1
1.0	1	1	1	1

In Table 4.27, we present *group total cost* values when we minimize *group total cost* by trying different *sat* values. The *sat* based optimizations yield the lowest *group total cost* values for most values of the *wsl* when the wholesaler is the echelon of concern. However, when *wsl* is equal to 0.9, the minimum *group total cost* value is achieved when the retailer is the echelon of concern. In addition, as *wsl* increases, the differences between maximum and minimum optimum *group total cost* values obtained for the different echelons of concern gets smaller. For example, for *wsl* equals to 0.0, the maximum *group total cost* value, which is equal to $93,486, is obtained when the factory is the echelon of interest and the minimum value, which is equal to $11,785, is obtained when the wholesaler is the echelon of interest. The difference between these cost values is $81,701. However, when *wsl* is 1.0, all the *group total cost* values become the same.

Table 4.27. *Group total cost* values when we minimize *group total cost* for the four different echelons of concern by trying different *sat* values.

wsl	*GTC* Minimized for the Retailer ($)	*GTC* Minimized for the Wholesaler ($)	*GTC* Minimized for the Distributor ($)	*GTC* Minimized for the Factory ($)
0.0	15114.5	**11785**	47432	93486
0.1	9266	**7081.5**	20742.5	40876.5
0.2	5936.5	**4781.5**	11075	21776
0.3	4196.5	**3727**	6463.5	12612.5
0.4	3298	**3024.5**	4472	8030
0.5	2699.5	**2430.5**	3318	5410
0.6	2399	**2142.5**	2671.5	3879.5
0.7	2129	**2013.5**	2338	2961
0.8	1891	**1782.5**	2221.5	2371
0.9	**1567.5**	1711	1800.5	1835.5
1.0	**1428.5**	**1428.5**	**1428.5**	**1428.5**

4.3.2. Key Observations in Desired Inventory Optimization

The optimum values of *desired inventory* (I^*) for all echelons of concern are affected from the *wsl* value of the three identically controlled echelons; the optimum I^* values for the distributor and the factory echelons are affected more than the optimum I^* values for the retailer and the wholesaler echelons. The optimum *desired inventory* (I^*) levels for the retailer, the wholesaler, the distributor, and the factory when we minimize their total costs are presented in Table 4.28.

Table 4.28. Optimum I^* values for the retailer, the wholesaler, the distributor, and the factory when we minimize their total costs by trying different I^* values.

wsl	I_R^* (case)	I_W^* (case)	I_D^* (case)	I_F^* (case)
0.0	20	42	217	855
0.1	18	41	145	413
0.2	16	33	114	224
0.3	15	31	89	111
0.4	22	31	74	52
0.5	20	31	58	-66
0.6	19	30	48	-74
0.7	17	30	31	-37
0.8	16	25	27	-14
0.9	15	23	15	5
1.0	15	21	1	0

The optimum *desired inventory* (I^*) levels for the retailer, the wholesaler, the distributor, and the factory when we minimize *group total cost* are presented in Table 4.29. This time, the optimum I^* values for the retailer are significantly affected from the changes in the *wsl* values like the distributor and the factory echelons, but the strength of the effect on the optimum I^* values for the wholesaler still is lower than the other three echelons.

Table 4.29. Optimum I^* values for the retailer, the wholesaler, the distributor, and the factory when we minimize *group total cost* by trying different I^* values.

wsl	I_R^* (case)	I_W^* (case)	I_D^* (case)	I_F^* (case)
0.0	-199	55	372	1050
0.1	-199	55	253	502
0.2	-200	54	138	329
0.3	-200	52	92	252
0.4	4	44	73	127
0.5	12	37	43	67
0.6	13	29	43	61
0.7	11	28	41	43
0.8	17	30	37	33
0.9	15	29	38	32
1.0	15	28	32	18

In Table 4.30, we present *group total cost* values when we minimize *group total cost* by trying different *desired inventory* levels. The *I** based optimizations yield the lowest *group total cost* values for most values of the *wsl* when the wholesaler is the echelon of concern. However, for *wsl* equals to 0.0, the minimum *group total cost* value is achieved when the retailer is the echelon of concern. In addition, as *wsl* increases, the differences between maximum and minimum optimum *group total cost* values obtained for the different echelons of concern gets smaller. For example, for *wsl* equals to 0.0, the maximum *group total cost* value, which is equal to $47,684.5 is obtained when the factory is the echelon of interest and the minimum value, which is equal to $4271.5, is obtained when the retailer is the echelon of interest. The difference between these cost values is $43,413. However, when *wsl* is 1.0, all the *group total cost* values become the same.

Table 4.30. *Group total cost* values when we minimize *group total cost* for the four different echelons of concern by trying different *desired inventory* levels.

wsl	*GTC* Minimized for the Retailer ($)	*GTC* Minimized for the Wholesaler ($)	*GTC* Minimized for the Distributor ($)	*GTC* Minimized for the Factory ($)
0.0	**4271.5**	4845	16262	47684.5
0.1	4270	**3742**	10696.5	25424
0.2	4270	**3261.5**	7791.5	15678
0.3	4269	**2880.5**	5490.5	10007.5
0.4	4072	**2598.5**	3897	6715
0.5	3062.5	**2079**	2730	4688.5
0.6	2383	**1633.5**	2037.5	3466.5
0.7	2045	**1318.5**	1575	2739.5
0.8	1750.5	**1156.5**	1286.5	2097.5
0.9	1434	**1068.5**	1118.5	1591
1.0	1263	**1004**	1053	1325

When we compare the *group total cost* values in Table 4.27 and Table 4.30, we observe that keeping *desired inventory* at a level different than zero in general gives a better instance of the anchor-and-adjust ordering policy compared to assigning a value to *sat* that is greater than one week. The only exception of this observation is obtained when the retailer is the echelon of concern; for some values of *wsl* (i.e., *wsl* = 0.3, *wsl* = 0.4, and *wsl* = 0.5), minimizing *group total cost* by obtaining optimum *sat* values gives better results than obtaining optimum *I** values.

5. RESULTS FOR THE EXTENDED SIMULATION TIME

In this chapter, we increase the final time of The Beer Game simulations from 36 weeks to 144 weeks. According to our observations, the dynamics obtained in 36 week simulations can remain incomplete that might have an effect on the optimum *sat* and *I** values and, as a result, the optimum values may not be valid in the long term. To eliminate these potential unwanted effects, a longer simulation time is selected and all experiments are re-conducted. The experiments and the corresponding results presented in Chapter 4 are important because The Beer Game is played for 36 simulated weeks. The experiments and the results in this chapter are important because they will either validate or invalidate the results presented in Chapter 4 for the long run.

5.1. Optimizing Stock Adjustment Time

In this section, we repeat the experiments mentioned in Section 4.1 with only a single change: the length of the simulation time is 144 weeks instead of 36 weeks.

5.1.1. Observations at the Retailer Echelon

Similar to the results obtained for the 36 week simulations reported in Section 4.1.1, when we minimize the retailer's total cost, the optimum value of *sat* for the retailer becomes one week for all *wsl* values, except for *wsl* = 0.7 (Table 5.1).

When we optimize *group total cost* by trying different *sat* values for the retailer, the optimum value of *sat* becomes infinity for *wsl* values between 0.0 and 0.3. This means that the retailer makes no adjustments in this range; he only gives orders equal to the expected value of *end-customer demand*. For 0.4 ≤ *wsl* ≤ 0.8, the retailer makes milder adjustments, except for *wsl* = 0.5 (see Table 5.2). Similar to the results obtained for the 36 week simulations, the retailer makes no adjustments for low values of *wsl*.

Table 5.1. Optimum *sat* and corresponding cost values when we optimize the retailer's total cost.

wsl_W wsl_D wsl_F	Optimum sat_R (week)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	1	132292	917	9045.5	67051	55278.5
0.1	1	59150.5	1051	4285.5	26627	27187
0.2	1	25987.5	1043	2547.5	10729.5	11667.5
0.3	1	13995.5	1075	1819	5383	5718.5
0.4	1	10311.5	1237	1759	3838	3477.5
0.5	1	8704.5	1369	1647	2959	2729.5
0.6	1	6410	1239	1399	1959.5	1812.5
0.7	2	3103	812	684	820.5	786.5
0.8	1	2534.5	654	642.5	680	558
0.9	1	1595.5	468	429.5	402	296
1.0	1	1432.5	447	402	341	242.5

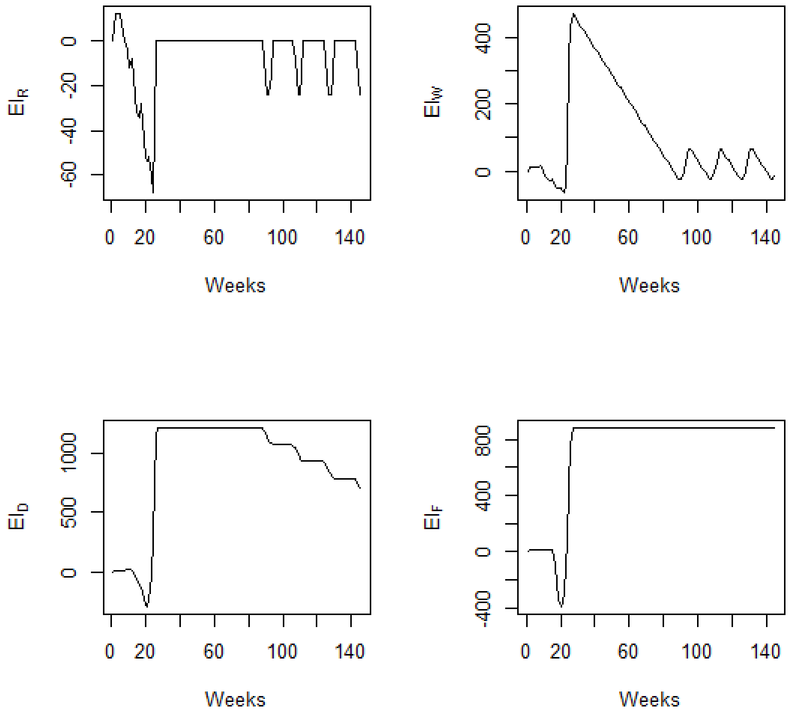

Figure 5.1. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F} = 0$ and $sat_R = 1$.

Table 5.2. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the retailer.

wsl_W wsl_D wsl_F	Optimum sat_R (week)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	∞	93127.5	3337	6377.5	43449.5	39963.5
0.1	∞	39964.5	3498	3516	16207.5	16743
0.2	∞	19323.5	3460	1993.5	7216.5	6653.5
0.3	∞	13455.5	3503	1598.5	3719.5	4634.5
0.4	8	9795.5	1679	1468.5	3154.5	3493.5
0.5	1	8704.5	1369	1647	2959	2729.5
0.6	4	5947.5	1394	1250	1693	1610.5
0.7	3	3076	832	685.5	800	758.5
0.8	3	2505.5	777	601	625.5	502
0.9	1	1595.5	468	429.5	402	296
1.0	1	1432.5	447	402	341	242.5

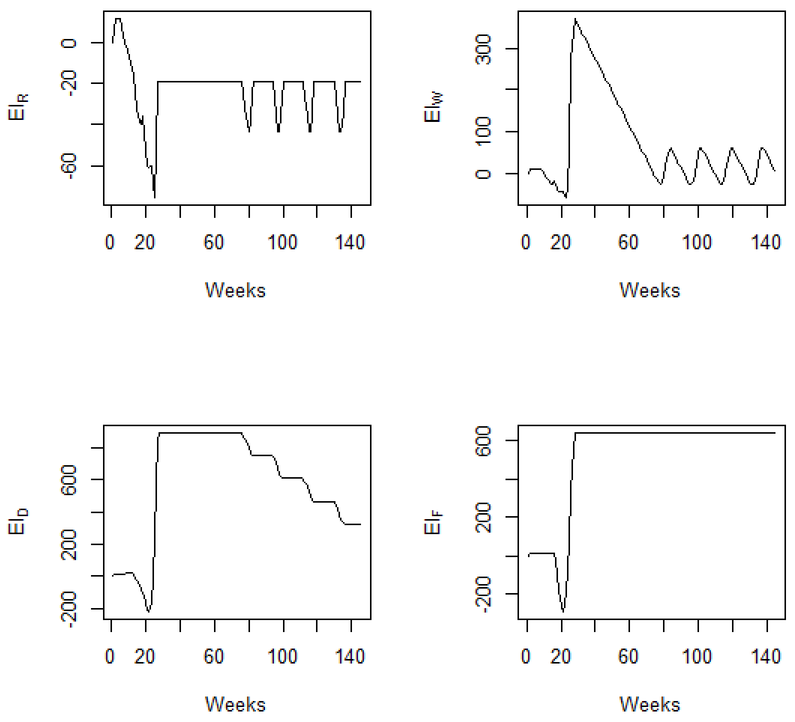

Figure 5.2. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F} = 0$ and $sat_R = \infty$.

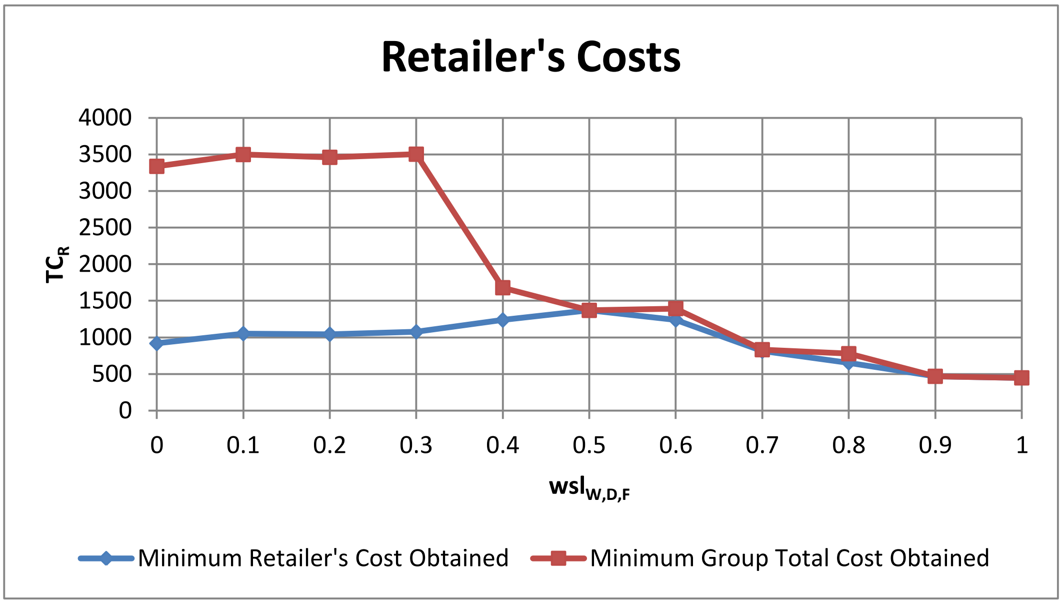

Figure 5.3. Retailer’s cost values for the two different objectives.

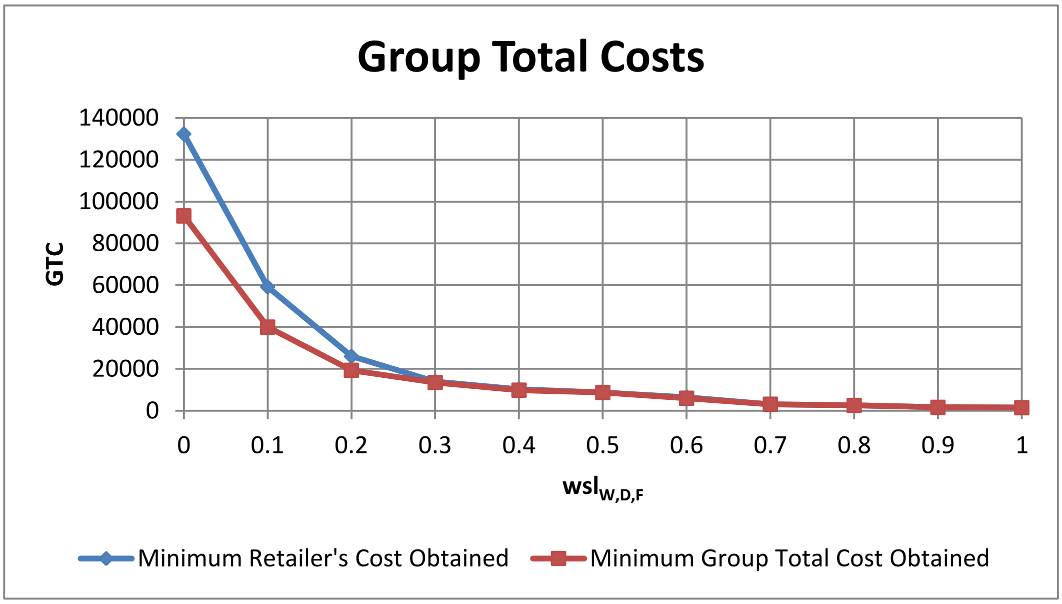

Figure 5.4. *Group total cost* values for the two different objectives.

In Figure 5.3, we observe that there is a difference in the retailer’s cost values for $0.0 \leq wsl \leq 0.4$, $wsl = 0.6$ and, $wsl = 0.8$ under two different objectives (i.e., the minimum cost for the retailer can be obtained and the minimum *group total cost* can be obtained by optimizing the retailer’s *sat*). In Figure 5.4, one can observe a difference in the *group total cost* values for $0.0 \leq wsl \leq 0.2$. We conclude that we can obtain lower *GTC* values by sacrificing the objective of minimizing the retailer's total cost for all *wsl* values, except for *wsl* = 0.5, *wsl* = 0.9 and, *wsl* = 1.0. Behaviorally, these results are no different than the results obtained for the 36 week simulations reported in Section 4.1.1.

Table 5.3. The percent changes in the optimum TC_R and in the optimum *GTC*.

wsl_W wsl_D wsl_F	sat_R for Obj. 1	sat_R for Obj. 2	Change in TC_R (%)	Change in GTC (%)
0.0	1	∞	263.90	-29.60
0.1	1	∞	232.83	-32.44
0.2	1	∞	231.74	-25.64
0.3	1	∞	225.86	-3.86
0.4	1	8	35.73	-5.00
0.5	1	1	0.00	0.00
0.6	1	4	12.51	-7.22
0.7	2	3	2.46	-0.87
0.8	1	3	18.81	-1.14
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

Table 5.3 presents the percent changes in the optimum retailer's total cost (TC_R) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the retailer's total cost" to "optimizing *group total cost* by optimizing the retailer's *sat*". For example, when *wsl* values for the wholesaler, the distributor and the factory are equal to zero, we can reduce *group total cost* by 263.90% and this reduction results in a 29.60% increase in the retailer's optimal total cost. The greatest reduction in the *group total cost* (-32.44%) is achieved for *wsl* = 0.1. These results are similar to the results obtained for the 36 week simulations reported in Section 4.1.1 with an exception; in order to obtain an improvement in *GTC* values for $0.0 \leq wsl \leq 0.3$, a higher percent increase in retailer's total cost is needed in the 144 week simulations.

5.1.2. Observations at the Wholesaler Echelon

When we minimize the wholesaler's total cost, the optimum value of *sat* becomes equal to infinity for *wsl* = 0.0. In addition, different than the results obtained for the 36 week simulations, we observe that making milder adjustments for *wsl* values between 0.2 and 0.8 minimizes the wholesaler's total cost (see Table 5.4). Note that in the 36 week simulations, taking *sat* as one week minimizes the wholesaler's total cost.

Table 5.4. Optimum *sat* and corresponding cost values when we optimize the wholesaler's total cost.

wsl_R wsl_D wsl_F	Optimum sat_W (week)	*TTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	∞	42889.5	11854	8772	7380	14883.5
0.1	1	62871	4893	6323	22449	29206
0.2	12	15482	2581	3946.5	2928.5	6026
0.3	2	16433.5	1773.5	2779	5142	6739
0.4	12	7681.5	1286	2158	1845.5	2392
0.5	11	6969	1172.5	1820.5	1692	2284
0.6	13	5275	1071.5	1506	1258	1439.5
0.7	14	3120.5	804.5	1061.5	641	613.5
0.8	10	2288.5	669.5	775.5	471.5	372
0.9	1	2035.5	576.5	568	502	389
1.0	1	1432.5	447	402	341	242.5

When we optimize *group total cost* by trying different *sat* values for the wholesaler, making no stock and supply line adjustments for *wsl* equal to 0.0 and 0.1 minimizes *group total cost*. For $0.2 \leq wsl \leq 0.9$, making mild adjustments minimizes *group total cost*. For *wsl* equal to 1.0, the optimum *sat* value becomes one week. Different than the results obtained for the 36 week simulations, making mild adjustments for $0.2 \leq wsl \leq 0.7$ minimizes *group total cost*. However, in the 36 week simulations, making no adjustments (i.e., taking *sat* as infinity.) minimizes *group total cost* in this range.

Table 5.5. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the wholesaler.

wsl_R wsl_D wsl_F	Optimum sat_W (week)	TTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	∞	42889.5	11854	8772	7380	14883.5
0.1	∞	24024.5	4479	7126.5	5550.5	6868.5
0.2	12	15482	2581	3946.5	2928.5	6026
0.3	13	9831.5	1708.5	3045	2108.5	2969.5
0.4	12	7681.5	1286	2158	1845.5	2392
0.5	10	6823.5	1229	1841	1628	2125.5
0.6	13	5275	1071.5	1506	1258	1439.5
0.7	14	3120.5	804.5	1061.5	641	613.5
0.8	14	2228.5	664.5	805.5	422	336.5
0.9	6	1866.5	585.5	600	397.5	283.5
1.0	1	1432.5	447	402	341	242.5

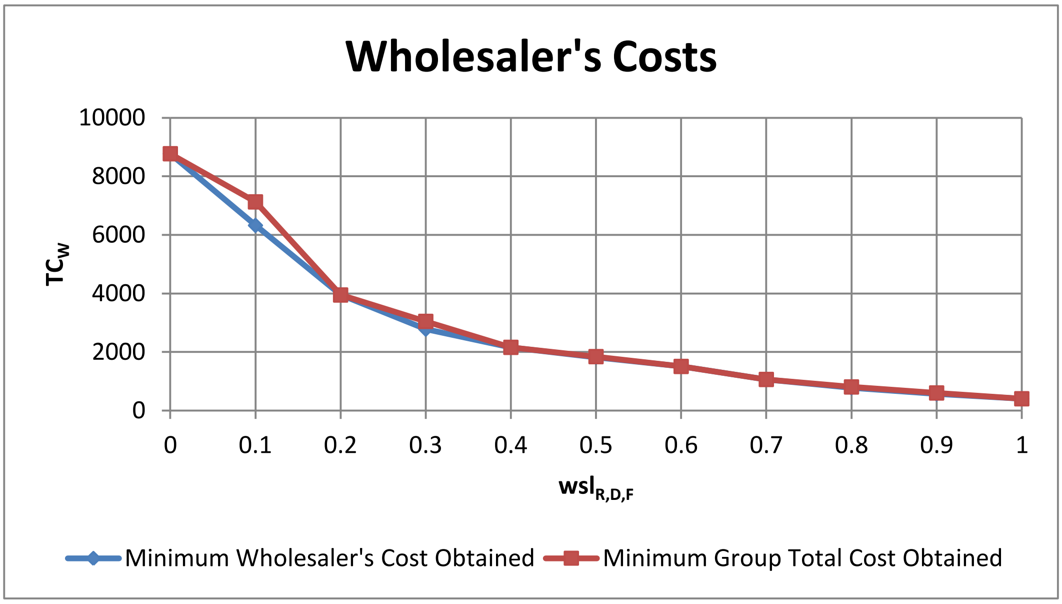

Figure 5.5. Wholesaler’s cost values for the two different objectives.

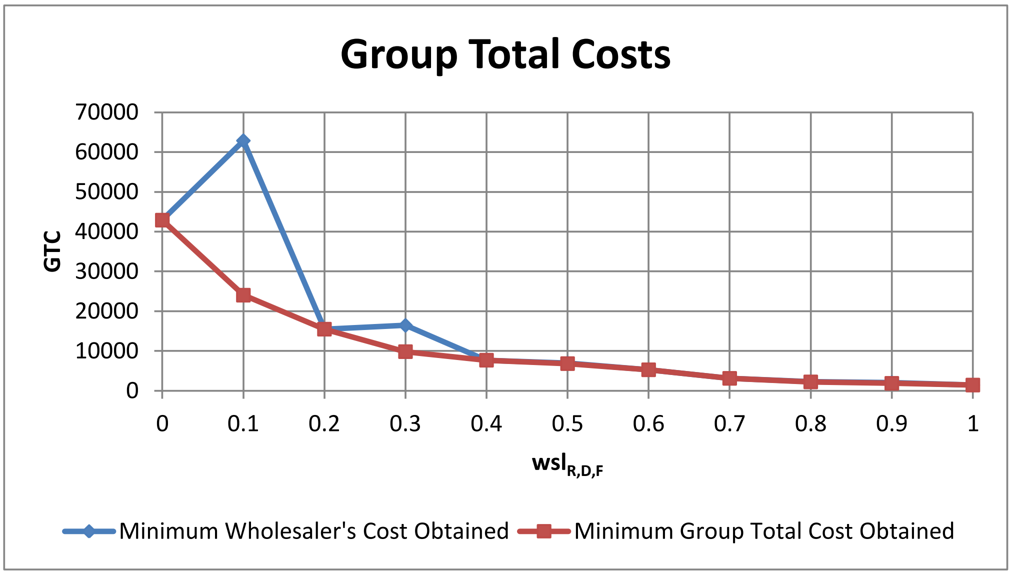

Figure 5.6. *Group total cost* values for the two different objectives.

In Figure 5.5, we observe that there are some differences in the wholesaler's cost values under two different objectives (i.e., the minimum cost for the wholesaler can be obtained and the minimum *group total cost* can be obtained by optimizing the wholesaler's *sat*). In Figure 5.6, one can also observe some differences in the *group total cost* values. We conclude that we can obtain lower *GTC* values by sacrificing the objective of minimizing the wholesaler's total cost for some values of *wsl* (see Table 5.6).

Table 5.6. The percent changes in the optimum TC_W and in the optimum *GTC*.

wsl_R wsl_D wsl_F	sat_W for Obj. 1	sat_W for Obj. 2	Change in TC_W (%)	Change in *GTC* (%)
0.0	∞	∞	0.00	0.00
0.1	1	∞	12.71	-61.79
0.2	12	12	0.00	0.0
0.3	2	13	9.57	-40.17
0.4	12	12	0.00	0.00
0.5	11	10	1.13	-2.09
0.6	13	13	0.00	0.00
0.7	14	14	0.00	0.00
0.8	10	14	3.87	-2.62
0.9	1	6	5.63	-8.30
1.0	1	1	0.00	0.00

Table 5.6 presents the percent changes in the optimum wholesaler's total cost (TC_W) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the wholesaler's total cost" to "optimizing *group total cost* by optimizing the wholesaler's *sat*". For example, when *wsl* values for the retailer, the distributor and the factory are equal to 0.3, we can reduce *group total cost* by 40.17% and this reduction results in a 9.57% increase in the wholesaler's optimal total cost. The greatest reduction in the group total cost (-61.79%) is achieved for *wsl* = 0.1.

5.1.3. Observations at the Distributor Echelon

Different than the results obtained for the 36 week simulations reported in Section 4.1.3, we observe that the optimum value of *sat* is equal to infinity for $0.0 \leq wsl \leq 0.3$ when we minimize the distributor's total cost (Table 5.7). Making mild adjustments for *wsl* values between 0.4 and 0.8 minimizes the distributor's total cost. Similar to the results obtained for the 36 week simulations, taking *sat* as one week for *wsl* = 0.9 and 1.0 minimizes the distributor's total cost.

Table 5.7. Optimum *sat* and corresponding cost values when we optimize the distributor's total cost.

wsl_R wsl_W wsl_F	Optimum sat_D (week)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	∞	257595.5	12296	197091	37482.5	10726
0.1	∞	78707.5	4302	54908	14774	4723.5
0.2	∞	36617	3409	17379	12942.5	2886.5
0.3	∞	14280.5	1737.5	4963	6223.5	1356.5
0.4	7	12204	1434	3339.5	4617.5	2813
0.5	2	9152.5	1359	2431	3129.5	2233
0.6	12	7072.5	1346.5	1963.5	2703.5	1059
0.7	7	3388.5	732	923	1181	552.5
0.8	14	3017	755	879	1024	359
0.9	1	2168	598	601	556.5	412.5
1.0	1	1432.5	447	402	341	242.5

When we minimize *group total cost* by trying different *sat* values for the distributor, the optimum value of *sat* becomes equal to infinity for *wsl* values between 0.0 and 0.3 (Table 5.8). For $0.4 \leq wsl \leq 0.8$, the optimum values of *sat* correspond to making mild adjustments. However, in the 36 week simulations, taking *sat* as infinity for $0.4 \leq wsl \leq 0.7$ minimizes *group total cost*.

Table 5.8. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the distributor.

wsl_R wsl_W wsl_F	Optimum sat_D (week)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	∞	257595.5	12296	197091	37482.5	10726
0.1	∞	78707.5	4302	54908	14774	4723.5
0.2	∞	36617	3409	17379	12942.5	2886.5
0.3	∞	14280.5	1737.5	4963	6223.5	1356.5
0.4	16	12007	1531.5	3430.5	5006	2039
0.5	2	9152.5	1359	2431	3129.5	2233
0.6	12	7072.5	1346.5	1963.5	2703.5	1059
0.7	7	3388.5	732	923	1181	552.5
0.8	14	3017	755	879	1024	359
0.9	1	2168	598	601	556.5	412.5
1.0	1	1432.5	447	402	341	242.5

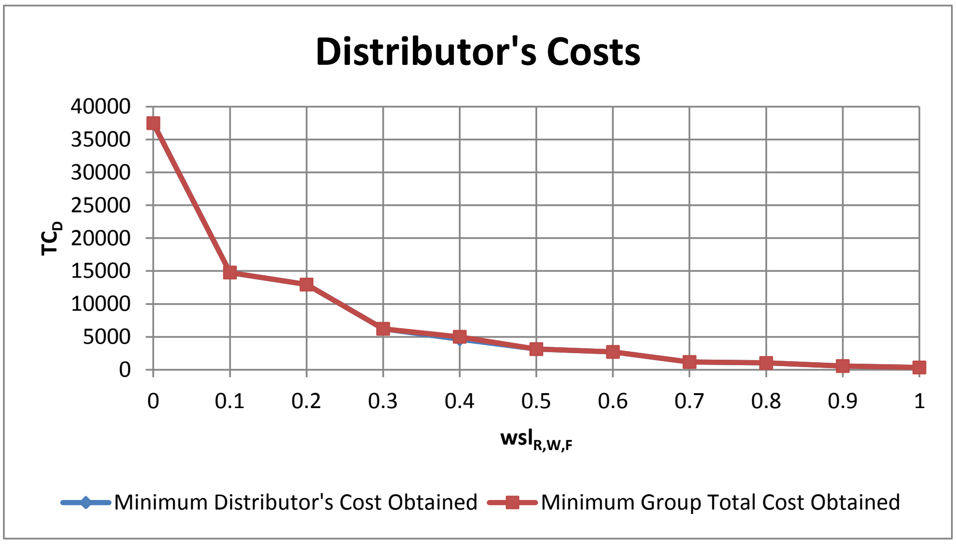

Figure 5.7. Distributor’s cost values for the two different objectives.

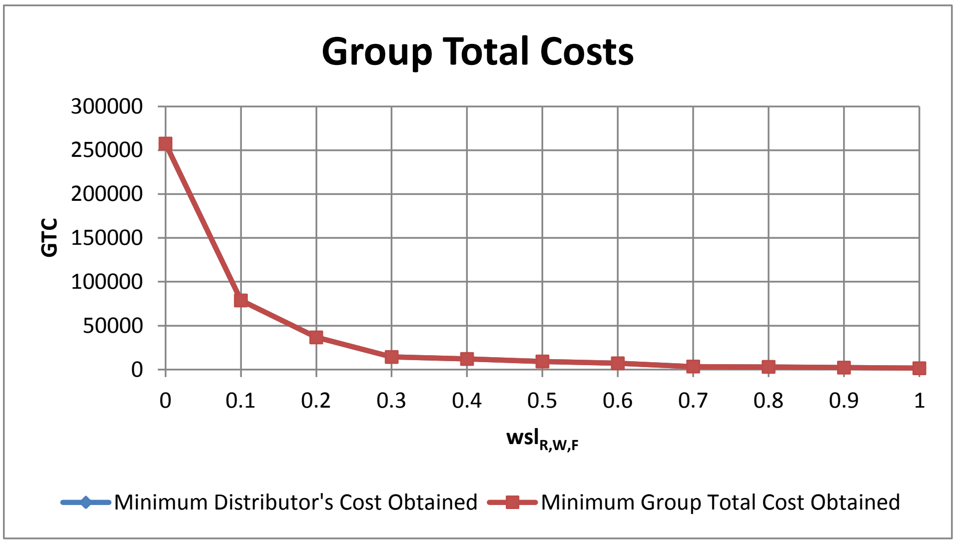

Figure 5.8. *Group total cost* values for the two different objectives.

In Figure 5.7 and Figure 5.8, we cannot see any difference in the wholesaler's cost values under two different objectives (i.e., the minimum cost for the distributor can be obtained and the minimum *group total cost* can be obtained by optimizing the distributor's *sat*). We conclude that we cannot reduce *GTC* by allowing an increase in the distributor's total cost, except for *wsl* = 0.4 (see Table 5.9).

Table 5.9. The percent changes in the optimum TC_D and in the optimum *GTC*.

wsl_R wsl_W wsl_F	sat_D for Obj. 1	sat_D for Obj. 2	Change in TC_D (%)	Change in GTC (%)
0.0	∞	∞	0.00	0.00
0.1	∞	∞	0.00	0.00
0.2	∞	∞	0.00	0.00
0.3	∞	∞	0.00	0.00
0.4	7	16	8.41	-1.61
0.5	2	2	0.00	0.00
0.6	12	12	0.00	0.00
0.7	7	7	0.00	0.00
0.8	14	14	0.00	0.00
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

In Table 5.9, we present the percent changes in the optimum distributor's cost (TC_D) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the distributor's total cost" to "optimizing *group total cost* by optimizing the distributor's *sat*". For example, when *wsl* values for the retailer, the wholesaler and the factory are equal to 0.4, we can reduce *group total cost* by 1.61% and this reduction results in an 8.41% increase in the distributor's optimal total cost. Note that we can decrease *GTC* by sacrificing the objective of minimizing the distributor's total cost only for *wsl* = 0.4.

5.1.4. Observations at the Factory Echelon

When we minimize the factory's total cost, the optimum value of *sat* becomes equal to infinity for *wsl* values between 0.0 and 0.4. Making mild adjustments for $0.6 \leq wsl \leq 0.8$ minimizes the factory's total cost. Note that, when the final simulated time is 36 weeks, taking *sat* as one week for all *wsl* values minimizes the factory's total cost.

Table 5.10. Optimum *sat* and corresponding cost values when we optimize the factory's total cost.

wsl_R wsl_W wsl_D	Optimum sat_F (week)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	∞	1580932.5	13742.5	226622	1160268.5	180299.5
0.1	∞	437124.5	4570.5	69860.5	311325	51368.5
0.2	∞	153825	2807	20300.5	110337	20380.5
0.3	∞	55926	1927.5	7393	36517.5	10088
0.4	∞	24726.5	1753	3992.5	11369.5	7611.5
0.5	1	12321.5	1476.5	2523	3923.5	4398.5
0.6	7	8645	1177	1751	2707	3010
0.7	16	6300.5	1066	1410.5	1789	2035
0.8	9	4354.5	908	1064	1164.5	1218
0.9	1	2322	628	633.5	586	474.5
1.0	1	1432.5	447	402	341	242.5

Similar to the results obtained for the 36 week simulations reported in Section 4.1.4, when we minimize *group total cost* by trying different *sat* values for the factory, the optimum value of *sat* becomes one week when *wsl* values range from 0.0 and 0.3. However, making milder adjustments for $0.6 \le wsl \le 0.8$ minimizes *group total cost*.

Table 5.11. Optimum *sat* and corresponding cost values when we optimize *group total cost* by trying different *sat* values for the factory.

wsl_R wsl_W wsl_D	Optimum sat_F (week)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	1	823558	14535	154463	446602	207958
0.1	1	290018.5	5486	55957	156493	72082.5
0.2	1	120964	2973.5	18609	62444	36937.5
0.3	1	49976	2276	7329	21033.5	19337.5
0.4	2	23623.5	1867.5	3919	8992	8845
0.5	1	12321.5	1476.5	2523	3923.5	4398.5
0.6	7	8645	1177	1751	2707	3010
0.7	16	6300.5	1066	1410.5	1789	2035
0.8	14	4253.5	879.5	1026	1119.5	1228.5
0.9	1	2322	628	633.5	586	474.5
1.0	1	1432.5	447	402	341	242.5

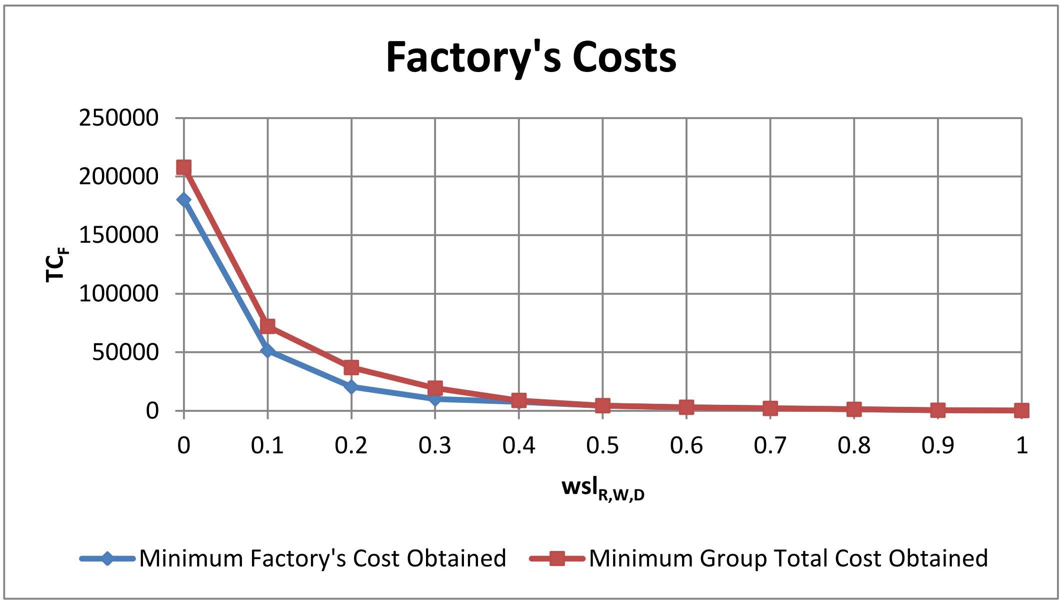

Figure 5.9. Factory’s cost values for the two different objectives.

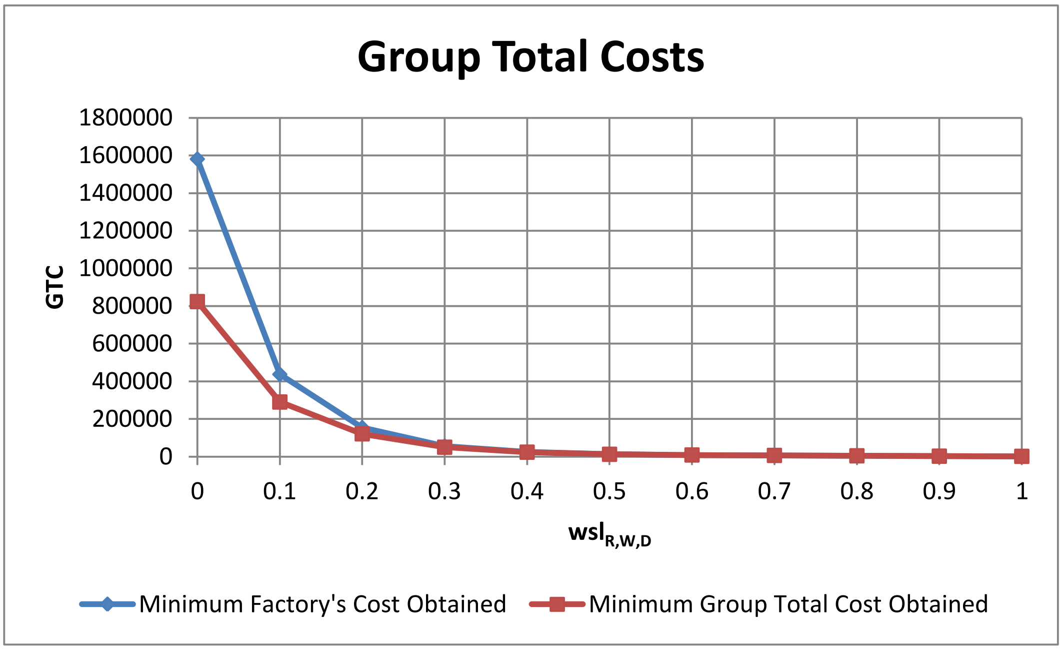

Figure 5.10. *Group total cost* values for the two different objectives.

Table 5.12. The percent changes in the optimum TC_F and in the optimum *GTC*.

wsl_R wsl_W wsl_D	sat_F for Obj. 1	sat_F for Obj. 2	Change in TC_F (%)	Change in *GTC* (%)
0.0	∞	1	15.34	-47.91
0.1	∞	1	40.32	-33.65
0.2	∞	1	81.24	-21.36
0.3	∞	1	91.69	-10.64
0.4	∞	2	16.21	-4.46
0.5	1	1	0.00	0.00
0.6	7	7	0.00	0.00
0.7	16	16	0.00	0.00
0.8	9	14	0.86	-2.32
0.9	1	1	0.00	0.00
1.0	1	1	0.00	0.00

Different than the results obtained for the 36 week simulations reported in Section 4.1.4, the two different objectives do not have the same effect on costs for all *wsl* values. In Figure 5.9, we observe that there are some differences in the factory's cost values under two different objectives (i.e., the minimum cost for the factory can be obtained and the minimum *group total cost* can be obtained by optimizing the factory's *sat*) when $wsl < 0.4$. In Figure 5.10, one can also observe some differences in the *group total cost* values. Different than the results obtained for the 36 week simulations, we conclude that we can reduce *GTC* while increasing the factory's total cost for $wsl \leq 0.4$ and $wsl = 0.8$ (see Table 5.12).

5.2. Optimizing Desired Inventory

In this section, we repeat the experiments mentioned in Section 4.2 with only a single change: the length of the simulation time is 144 weeks instead of 36 weeks.

5.2.1. Observations at the Retailer Echelon

When we minimize the retailer's total cost, the average optimum value of I^* for the retailer becomes 4.55 cases for $0.0 \leq wsl \leq 1.0$ (Table 5.13). In obtaining the optimum I^* values, we limit the search interval to [-50, 50] cases. We observe that the minimum I^* level is zero cases and it is obtained when *wsl* is 0.0, 0.9, and 1.0. Besides this, the maximum I^* level is 13 cases and it is obtained when *wsl* is 0.5. Similar to the results obtained for the 36 week simulations, the optimum I^* values do not follow a regular pattern; there is no clear relationship between *wsl* values of the other three echelons and the optimum I^* values of the retailer.

When we minimize *group total cost* by trying different I^* values for the retailer, the average optimum value of I^* for the retailer becomes -46.33 cases for $0.0 \leq wsl \leq 0.2$ and 6.38 cases for $0.3 \leq wsl \leq 1.0$ (Table 5.14). In obtaining the optimum I^* values, we limited the search interval to [-300, 300] cases. We observe that the minimum I^* level is -110 cases and it is obtained when *wsl* is 0.0. Besides this, the maximum I^* level is 26 cases and it is obtained when *wsl* is 0.7. Similar to the results obtained for the 36 week simulations reported in Section 4.2.1, the retailer aims to carry backlog for the low values of *wsl*.

Table 5.13. Optimum I^* and corresponding cost values when we optimize the retailer's total cost.

wsl_W wsl_D wsl_F	Optimum I_R^* (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	0	132292	917	9045.5	67051	55278.5
0.1	2	59432.5	1034.5	4290.5	26623	27484.5
0.2	7	25880.5	951.5	2467	10651	11811
0.3	6	14530	922.5	1824.5	5701	6082
0.4	11	8879	895	1476	3320	3188
0.5	13	8184.5	897	1625.5	2910.5	2751.5
0.6	1	4328	806.5	968.5	1315.5	1237.5
0.7	8	4079	752.5	1003.5	1219.5	1103.5
0.8	2	2485.5	622	632	675	556.5
0.9	0	1595.5	468	429.5	402	296
1.0	0	1432.5	447	402	341	242.5

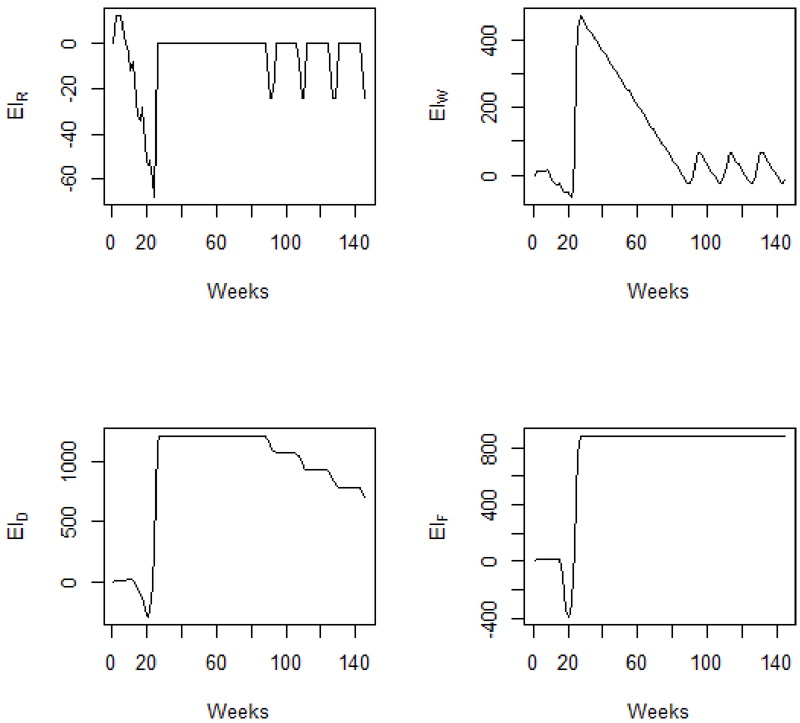

Figure 5.11. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F}$ = 0 and I_R^* = 0.

Table 5.14. Optimum I^* and corresponding cost values when we optimize *group total cost* by trying different I^* values for the retailer.

wsl_W wsl_D wsl_F	Optimum I^*_R (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	-110	119727	15083	7726	53540	43378
0.1	-18	56376.5	3507	4177.5	23836	24856
0.2	-11	24552	2484	2360	9841.5	9866.5
0.3	0	13995.5	1075	1819	5383	5718.5
0.4	11	8879	895	1476	3320	3188
0.5	10	8115.5	954	1616	2836	2709.5
0.6	1	4328	806.5	968.5	1315.5	1237.5
0.7	26	3874.5	1773.5	609	748	744
0.8	3	2368.5	627	589	629.5	523
0.9	0	1595.5	468	429.5	402	296
1.0	0	1432.5	447	402	341	242.5

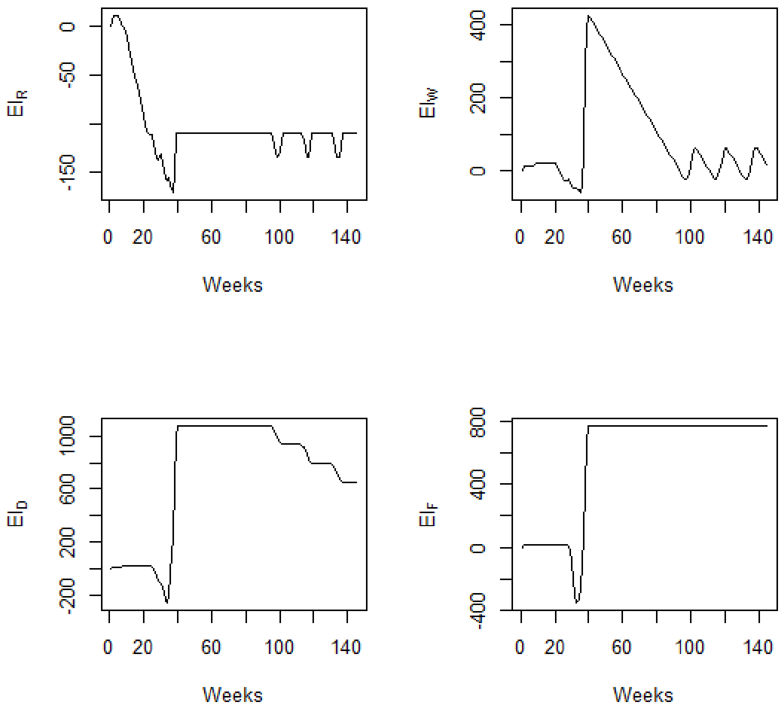

Figure 5.12. Dynamics of *EI* levels of the four echelons when $wsl_{W,D,F}$ = 0 and I^*_R = -110.

Although the need to carry backlog (i.e., negative *desired inventory* level) does not exactly correspond to the case where retailer makes no adjustments (i.e., *sat* = ∞), there still are similarities. By aiming to make its own net inventory negative, retailer aims to prevent the other three echelons' net inventories go below zero.

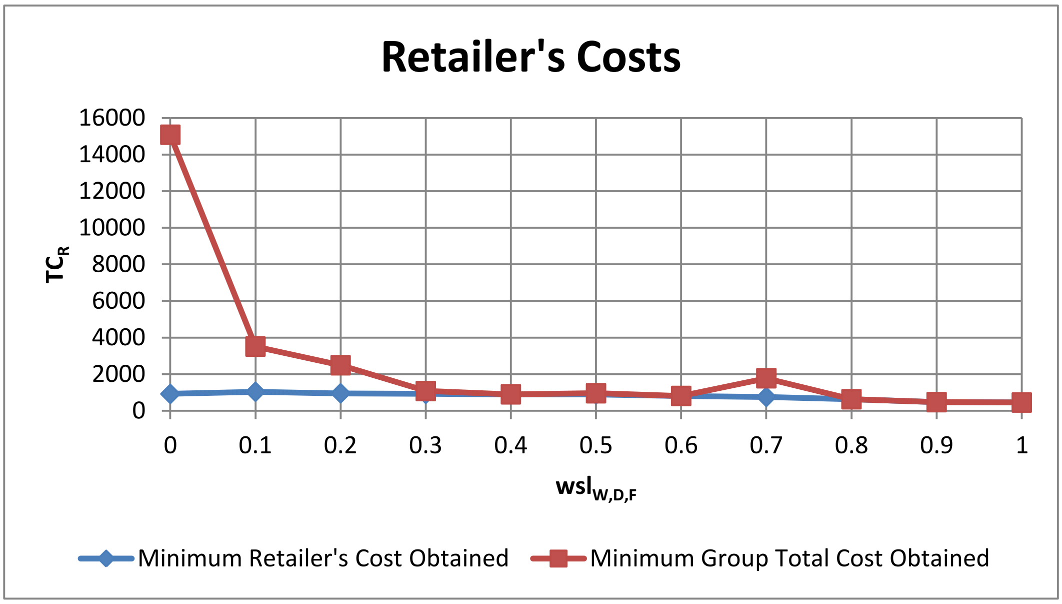

Figure 5.13. Retailer's cost values for the two different objectives.

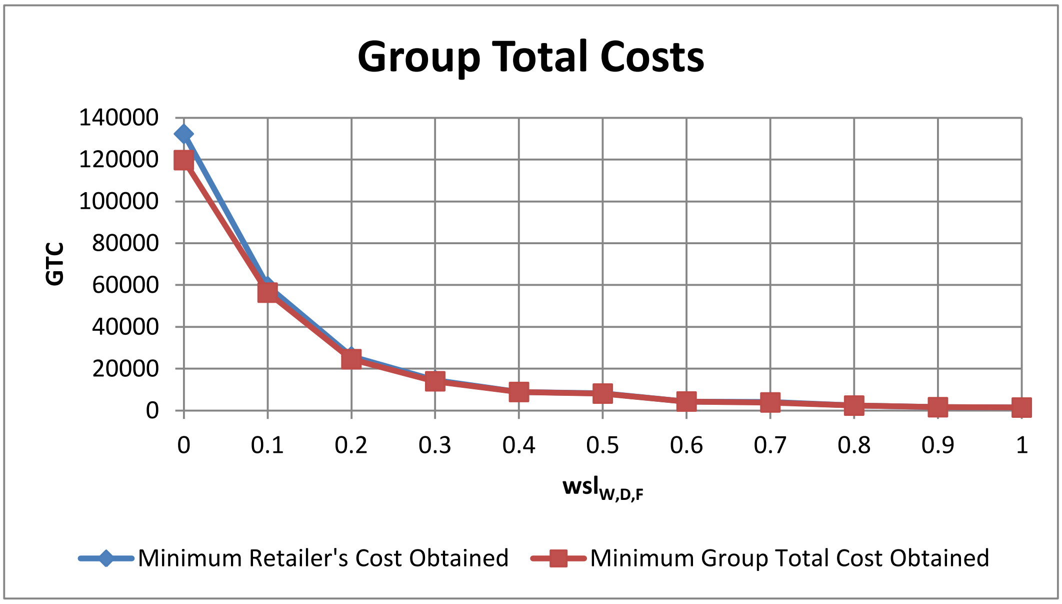

Figure 5.14. *Group total cost* values for the two different objectives.

In Figure 5.13, we observe that there is a significant difference in the retailer's cost values for *wsl* between 0.0 and 0.2 under two different objectives (i.e., the minimum cost for the retailer can be obtained and the minimum *group total cost* can be obtained by optimizing the retailer's *I**). In Figure 5.14, one can observe some differences in the *group total cost* values too. There are also relatively small differences that are not visible between the *group total cost* values obtained under the two different objectives (see Table 5.15).

Table 5.15. The percent changes in the optimum TC_R and in the optimum *GTC*.

wsl_W wsl_D wsl_F	I_R^* for Obj. 1	I_R^* for Obj. 2	Change in TC_R (%)	Change in GTC (%)
0.0	0	-110	1544.82	-9.50
0.1	2	-18	239.00	-5.14
0.2	7	-11	161.06	-5.13
0.3	6	0	16.53	-3.68
0.4	11	11	0.00	0.00
0.5	13	10	6.35	-0.84
0.6	1	1	0.00	0.00
0.7	8	26	135.68	-5.01
0.8	2	3	0.80	-4.71
0.9	0	0	0.00	0.00
1.0	0	0	0.00	0.00

In Table 5.15, we present the percent changes in the optimum retailer's total cost (TC_R) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the retailer's total cost" to "optimizing *group total cost* by optimizing the retailer's *I**". For example, when *wsl* values for the wholesaler, the distributor, and the factory are equal to 0.0, we can reduce *group total cost* by 9.50% and this reduction results in a 1544.82% increase in the retailer's optimal total cost. The greatest reduction in the group total cost (-9.50%) is achieved for *wsl* = 0.0. However, for this improvement, we have to increase the retailer's total cost by 1544.82%.

5.2.2. Observations at the Wholesaler Echelon

When we minimize the wholesaler's total cost, the average optimum value of *I** for the wholesaler becomes 15.36 cases for $0.0 \leq wsl \leq 1.0$ (Table 5.16). In obtaining the optimum *I** values, we limit the search interval to [-50, 50] cases. We observe that the minimum *I** level is 0 cases and it is obtained when $wsl = 1.0$. Besides this, the maximum *I** level is 40 cases and it is obtained when $wsl = 0.0$.

Table 5.16. Optimum *I** and corresponding cost values when we optimize the wholesaler's total cost.

wsl_R wsl_D wsl_F	Optimum I_W^* (case)	GTC (\$)	TC_R (\$)	TC_W (\$)	TC_D (\$)	TC_F (\$)
0.0	40	41315	2344.5	5220	11990	21760.5
0.1	34	31826	1744.5	3916	7741.5	18424
0.2	31	22128.5	1362	3080	5260.5	12426
0.3	26	16067.5	1054.5	2169.5	4295	8548.5
0.4	12	8134.5	916	1466	2002	3750.5
0.5	8	5957.5	809	1189	1587.5	2372
0.6	8	4847	722	1043	1370	1712
0.7	6	3858	677	905	1108	1168
0.8	3	2852.5	629.5	762	770.5	690.5
0.9	1	1926	528	567	475	356
1.0	0	1432.5	447	402	341	242.5

When we minimize *group total cost* by trying different *I** values for the wholesaler, the average optimum value of *I** for the wholesaler becomes 30.45 cases for $0.0 \leq wsl \leq 1.0$ (Table 5.17). In obtaining the optimum *I** values, we limit the search interval to [-100, 100] cases. We observe that the minimum *I** level is 0 cases and it is obtained when $wsl = 1.0$. Besides this, the maximum *I** level is 74 cases and it is obtained when $wsl = 0.0$.

Table 5.17. Optimum *I** and corresponding cost values when we optimize *group total cost* by trying different *I** values for the wholesaler.

wsl_R wsl_D wsl_F	Optimum I_W^* (case)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	74	32034.5	2434	7032.5	9303.5	13264.5
0.1	58	20425.5	1872	5151.5	5723	7679
0.2	57	13549.5	1453.5	4582.5	3460.5	4053
0.3	41	10884	1199.5	3125	2702.5	3857
0.4	28	7678	469	1717	1796	3696
0.5	27	5026.5	199.5	1603	1296.5	1927.5
0.6	30	3324.5	148	1976.5	492.5	707.5
0.7	10	2796	473	1077	638	608
0.8	8	2358.5	440.5	898.5	539	480.5
0.9	2	1920.5	501.5	594.5	470.5	354
1.0	0	1432.5	447	402	341	242.5

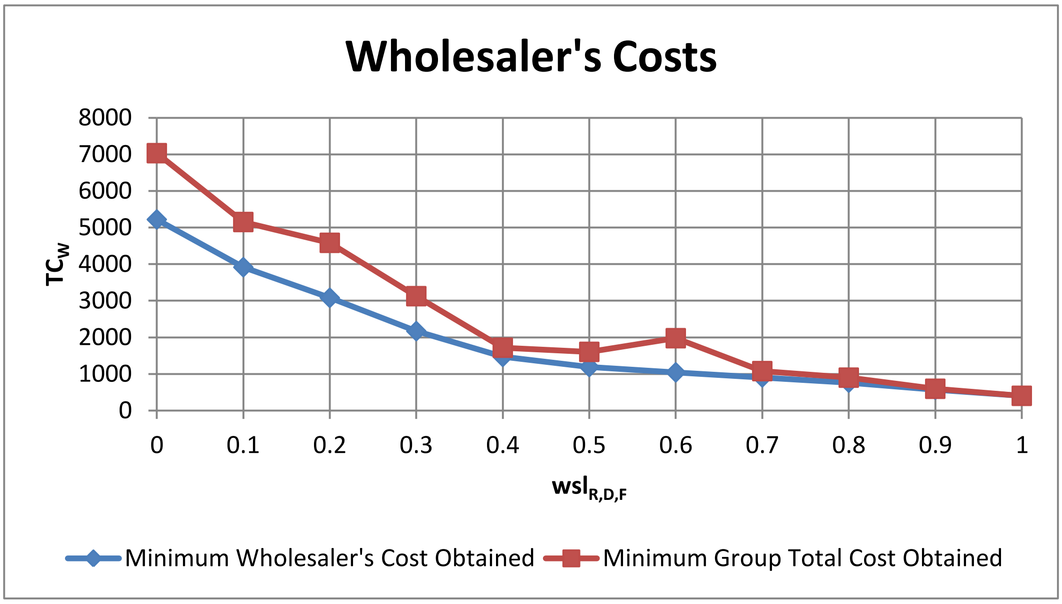

Figure 5.15. Wholesaler’s cost values for the two different objectives.

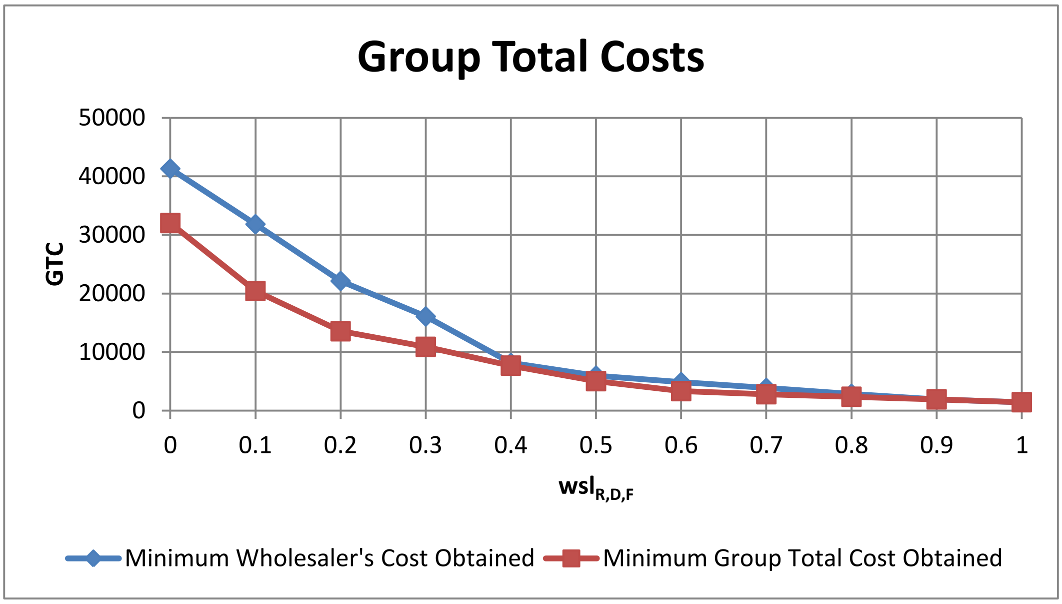

Figure 5.16. *Group total cost* values for the two different objectives.

In Figure 5.15, we observe that there is a difference in the wholesaler's cost values for *wsl* between 0.0 and 0.9 under two different objectives (i.e., the minimum cost for the wholesaler can be obtained and the minimum *group total cost* can be obtained by optimizing the wholesaler's *I**). In Figure 5.16, one can observe a difference in the *group total cost* values in the same range too. Similar to the results obtained for the 36 week simulations reported in Section 4.1.2, we conclude that we can effectively obtain lower *GTC* values by sacrificing the objective of minimizing the wholesaler's total cost for *wsl* values between 0.0 and 0.8.

In Table 5.18, we present the percent changes in the optimum wholesaler's total cost (TC_W) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the wholesaler's total cost" to "optimizing *group total cost* by optimizing the wholesaler's *I**". For example, when *wsl* values for the retailer, the distributor, and the factory are equal to 0.0, we can reduce *group total cost* by 22.46% and this reduction results in a 34.72% increase in the wholesaler's optimal total cost. The greatest reduction in the group total cost (-38.77%) is achieved for *wsl* = 0.2.

Table 5.18. The percent changes in the optimum TC_W and in the optimum *GTC*.

wsl_R wsl_D wsl_F	I_W^* for Obj. 1	I_W^* for Obj. 2	Change in TC_W (%)	Change in *GTC* (%)
0.0	40	74	34.72	-22.46
0.1	34	58	31.55	-35.82
0.2	31	57	48.78	-38.77
0.3	26	41	44.04	-32.26
0.4	12	28	17.12	-5.61
0.5	8	27	34.82	-15.63
0.6	8	30	89.50	-31.41
0.7	6	10	19.01	-27.53
0.8	3	8	17.91	-17.32
0.9	1	2	4.85	-0.29
1.0	0	0	0.00	0.00

5.2.3. Observations at the Distributor Echelon

When we minimize the distributor's total cost, the average optimum value of I^* for the distributor becomes 47.91 cases for $0.0 \leq wsl \leq 1.0$ (Table 5.19). In obtaining the optimum I^* values, we limit the search interval to [-300, 300] cases. We observe that the minimum I^* level is zero cases and it is obtained when $wsl = 0.9$ and 1.0. Besides this, the maximum I^* level is 229 cases and it is obtained when $wsl = 0.0$.

When we minimize *group total cost* by trying different I^* values for the distributor, the average optimum value of I^* for the distributor becomes 59.73 cases for $0.0 \leq wsl \leq 1.0$ (Table 5.20). In obtaining the optimum I^* values, we limit the search interval to [-300, 300] cases. We observe that the minimum I^* level is zero cases and it is obtained when $wsl = 1.0$. Besides this, the maximum I^* level is 281 cases and it is obtained when $wsl = 0.0$.

Table 5.19. Optimum *I** and corresponding cost values when we optimize the distributor's total cost.

wsl_R wsl_W wsl_F	Optimum I_D^* (case)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	229	92499	3382	13029	26078	50010
0.1	109	48283.5	2112	8158	15119.5	22894
0.2	93	25499.5	1416	4036.5	9472.5	10574.5
0.3	32	18417	1372.5	3830.5	6204	7010
0.4	19	11431	1295	2684.5	3915	3536.5
0.5	18	7308.5	954	1475.5	2506	2373
0.6	16	5438	831	1059	1777	1771
0.7	7	3294.5	621	756	1144	773.5
0.8	4	2860	640.5	709	876.5	634
0.9	0	2168	598	601	556.5	412.5
1.0	0	1432.5	447	402	341	242.5

Table 5.20. Optimum *I** and corresponding cost values when we optimize *group total cost* by trying different *I** values for the distributor.

wsl_R wsl_W wsl_F	Optimum I_D^* (case)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	281	77278	4447	13407	32186	27238
0.1	163	43569.5	2159	5903	15226.5	20281
0.2	84	24779.5	1615	4176.5	10072	8916
0.3	46	17945.5	1230.5	3002	6480.5	7232.5
0.4	19	11431	1295	2684.5	3915	3536.5
0.5	18	7308.5	954	1475.5	2506	2373
0.6	17	5109	751	991	1782.5	1584.5
0.7	9	3204	584	708	1247.5	664.5
0.8	17	2764	427.5	449	1416	471.5
0.9	3	2085.5	533.5	529	633	390
1.0	0	1432.5	447	402	341	242.5

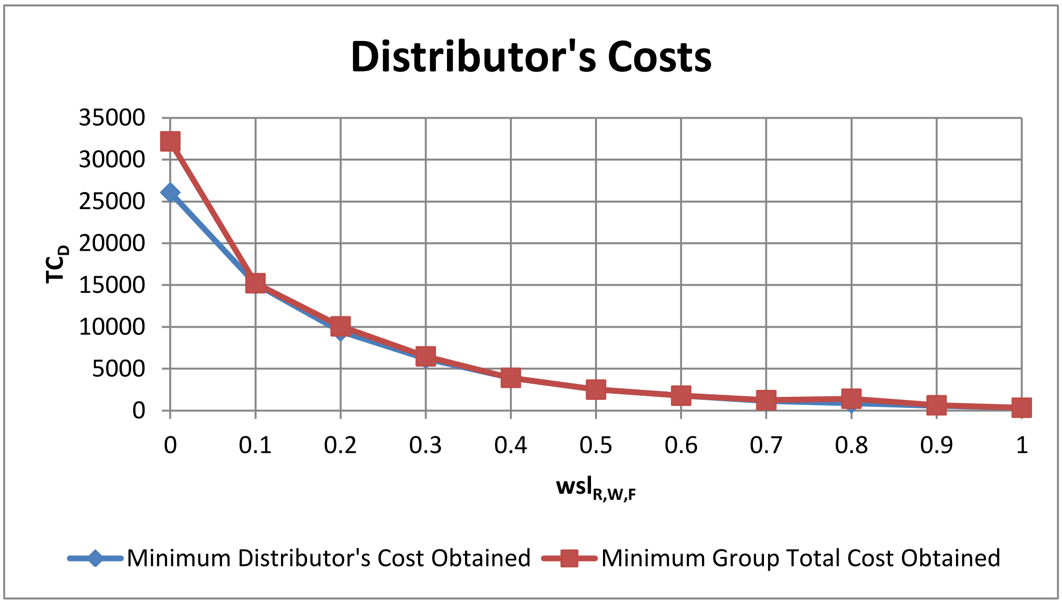

Figure 5.17. Distributor’s cost values for the two different objectives.

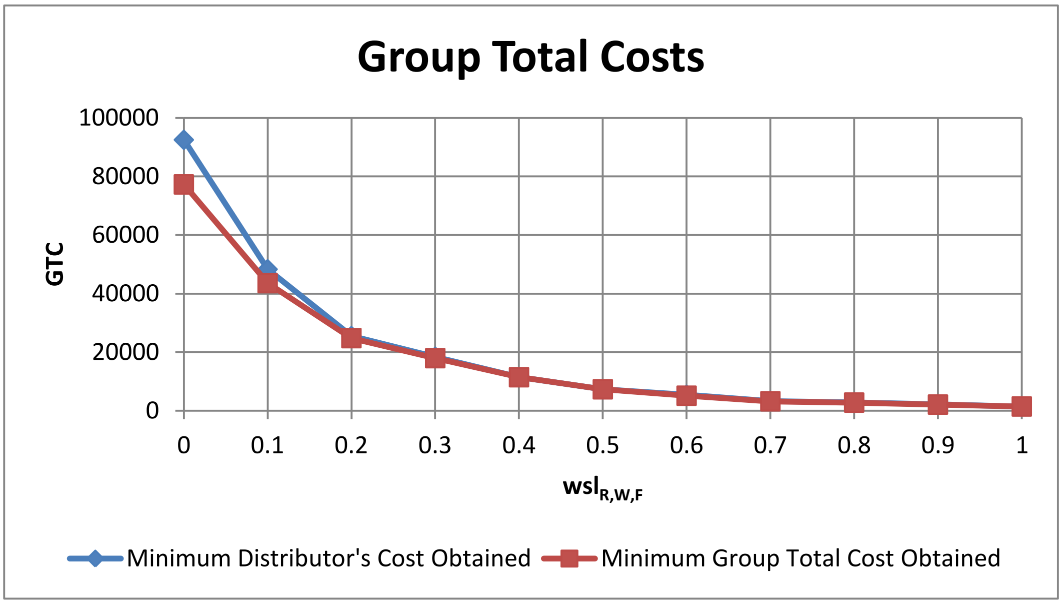

Figure 5.18. Group total cost values for the two different objectives.

In Figure 5.17, different than the results obtained for the 36 week simulations, we observe that there is not a significant difference in the distributor’s cost values for *wsl* between 0.0 and 0.2 under two different objectives (i.e., the minimum cost for the distributor can be obtained and the minimum *group total cost* can be obtained by optimizing the distributor’s I^*). In Figure 5.18, one can observe some differences in the *group total cost* values. There are also relatively small differences that are not visible (see Table 5.21).

Table 5.21. The percent changes in the optimum TC_D and in the optimum *GTC*.

wsl_R wsl_W wsl_F	I_D^* for Obj. 1	I_D^* for Obj. 2	Change in TC_D (%)	Change in GTC (%)
0.0	229	281	23.42	-16.46
0.1	109	163	0.71	-9.76
0.2	93	84	6.33	-2.82
0.3	32	46	4.46	-2.56
0.4	19	19	0.00	0.00
0.5	18	18	0.00	0.00
0.6	16	17	0.31	-6.05
0.7	7	9	9.05	-2.75
0.8	4	17	61.55	-3.36
0.9	0	3	13.75	-3.81
1.0	0	0	0.00	0.00

In Table 5.21, we present the percent changes in the optimum distributor's total cost (TC_D) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the distributor's total cost" to "optimizing *group total cost* by optimizing the distributor's *I**". For example, when *wsl* values for the retailer, the wholesaler, and the factory are equal to 0.2, we can reduce *group total cost* by 2.82% and this reduction results in a 6.33% increase in the distributor's optimal total cost. The greatest reduction in the group total cost (-16.46%) is achieved for *wsl* = 0.0.

5.2.4. Observations at the Factory Echelon

When we minimize the factory's total cost, the average optimum value of *I** for the factory becomes 482 cases for $0.0 \le wsl \le 0.1$ and -156.44 cases for $0.2 \le wsl \le 1.0$ (Table 5.22). In obtaining the optimum *I** values, we limit the search interval to [-1000, 1000] cases. We observe that the minimum *I** level is -826 cases and it is obtained when *wsl* is 0.2. Besides this, the maximum *I** level is 863 cases and it is obtained when *wsl* is 0.0.

Table 5.22. Optimum *I** and corresponding cost values when we optimize the factory's total cost.

wsl_R wsl_W wsl_D	Optimum I_F^* (case)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	863	366378	7951.5	72295	151436.5	134695
0.1	101	228079.5	3684	38294.5	120197.5	65903.5
0.2	-826	176903.5	4775	36587.5	123380	12161
0.3	-353	57582.5	2729	10651	39233	4969.5
0.4	-129	26304	2167	5298.5	13179	5659.5
0.5	-71	16797	1999.5	3857.5	6573.5	4366.5
0.6	1	9593.5	1451	2146	2923.5	3073
0.7	-2	8538	1693	2144.5	2450	2250.5
0.8	7	3661	803	909.5	908.5	1040
0.9	1	2187.5	592	591.5	541.5	462.5
1.0	0	1432.5	447	402	341	242.5

When we minimize *group total cost* by trying different *I** values for the factory, the average optimum value of *I** for the factory becomes 230.73 cases for $0.0 \leq wsl \leq 1.0$ (Table 5.23). In obtaining the optimum *I** values, we limit the search interval to [-2000, 2000] cases. We observe that the minimum *I** level is -1 case and it is obtained when *wsl* is 0.5. Besides this, the maximum *I** level is 1139 cases and it is obtained when *wsl* = 0.0.

In Figure 5.19 and Figure 5.20, we observe that there are some differences in the factory's cost values and *GTC* values under two different objectives (i.e., the minimum cost for the factory can be obtained and the minimum *group total cost* can be obtained by optimizing the factory's *I**).

Table 5.23. Optimum *I** and corresponding cost values when we optimize *group total cost* by trying different *I** values for the factory.

wsl_R wsl_W wsl_D	Optimum I_F^* (case)	*GTC* ($)	TC_R ($)	TC_W ($)	TC_D ($)	TC_F ($)
0.0	1139	342114	7951.5	72505	118639.5	143018
0.1	641	154266.5	4038	23676.5	50508.5	76043.5
0.2	385	75915.5	2406	9481.5	20270.5	43757.5
0.3	226	38031	1912.5	4875	9347	21896.5
0.4	70	20052.5	1308	2778.5	6442.5	9523.5
0.5	-1	12023	1396.5	2370	3849.5	4407
0.6	45	9239.5	1095.5	1597.5	2167.5	4379
0.7	18	7639.5	1338	1779	2004	2518.5
0.8	13	3528	691	771	773	1293
0.9	2	2184.5	578.5	579	531	496
1.0	0	1432.5	447	402	341	242.5

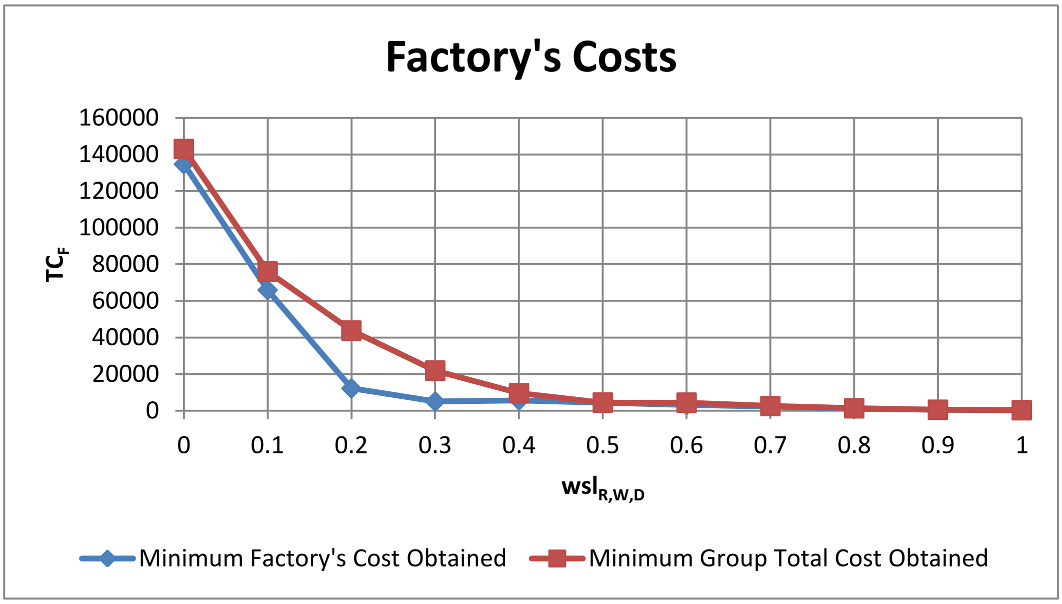

Figure 5.19. Factory's cost values for the two different objectives.

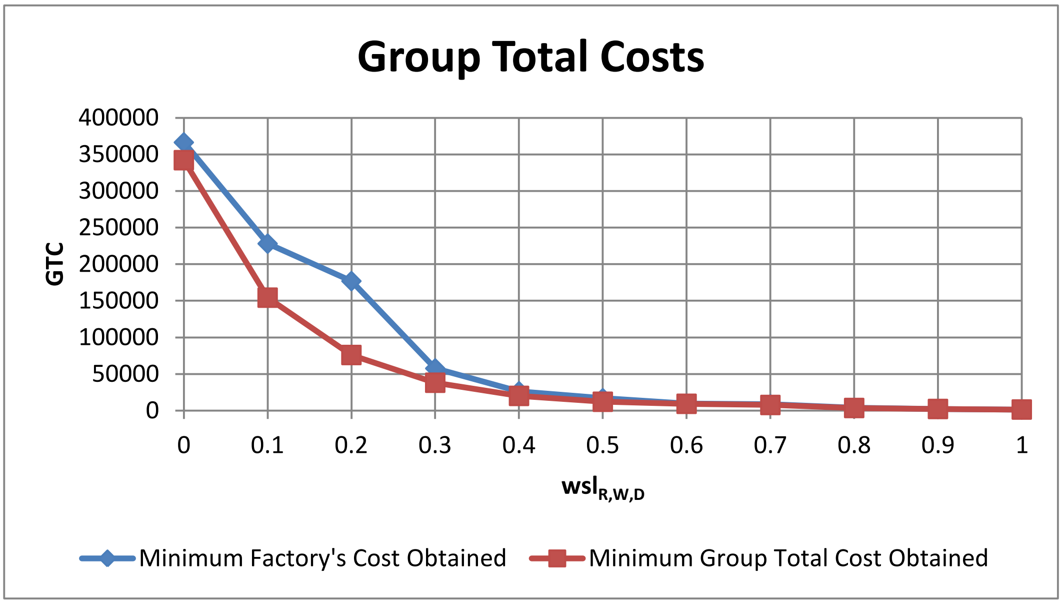

Figure 5.20. Group total cost values for the two different objectives.

Table 5.24. The percent changes in the optimum TC_F and in the optimum *GTC*.

wsl_R wsl_W wsl_D	I_F^* for Obj. 1	I_F^* for Obj. 2	Change in TC_F (%)	Change in *GTC* (%)
0.0	863	1139	6.18	-6.62
0.1	101	641	15.39	-32.36
0.2	-862	385	259.82	-57.09
0.3	-353	226	340.62	-33.95
0.4	-129	70	68.27	-23.77
0.5	-71	-1	0.93	-28.42
0.6	1	45	42.50	-3.69
0.7	-2	18	11.91	-10.52
0.8	7	13	24.33	-3.63
0.9	1	2	7.24	-0.14
1.0	0	0	0.00	0.00

In Table 5.24, we present the percent changes in the optimum factory's total cost (TC_F) and in the optimum *group total cost* (*GTC*) for each *wsl* value when we change the objective of the minimization problem from "optimizing the factory's total cost" to "optimizing *group total cost* by optimizing the factory's *I**". For example, when *wsl* values for the retailer, the wholesaler, and the distributor are equal to 0.0, we can reduce *group total cost* by 6.62% and this reduction results in a 6.18% increase in the factory's optimal total cost. The greatest reduction in the group total cost (-57.09%) is achieved for *wsl* = 0.2.

5.3. Key Observations

In the previous sections of this chapter, the results obtained for each echelon are analyzed and presented in an isolated noncomparative fashion. In this section, a comparative analysis will be presented.

5.3.1. Key Observations in Stock Adjustment Time Optimization

In the 36 week simulations, when we minimize the retailer's, the wholesaler's, the distributor's, or the factory's total costs, *sat* equals one week for all *wsl* values becomes optimal (see Table 4.25). However, when we increase the final simulated time to 144 weeks, we observe that taking *sat* as infinity for low values of *wsl* for the wholesaler, the distributor, and the factory becomes optimum. Moreover, making milder adjustments until *wsl* is 0.9 turns out to be optimum. (see Table 5.25). Similar to the results obtained for the 36 week simulations, taking *sat* as one week still minimizes the retailer's total cost, except for *wsl* = 0.7.

Similar to the results obtained for the 36 week simulations, when we optimize *group total cost* by trying different *sat* values for the retailer, the wholesaler, and the distributor, we observe that making mild or no adjustments (i.e., taking *sat* as infinity.) minimizes *group total cost* for low values of *wsl*. This also means that they give orders equal to (or close to) the expected demands of their customers for these *wsl* values. Different than the results obtained for the 36 week simulations, making milder adjustments for some values of *wsl* minimizes the factory's total cost (see Table 5.26). Unlike the other echelons, when we minimize *group total cost* for 0.0 ≤ *wsl* ≤ 0.3 for the factory, we observe that behaving as aggressively as possible for inventory and supply line adjustments in the 144 week simulations turns out to be optimal.

Table 5.25. Optimum *sat* values for the retailer, the wholesaler, the distributor, and the factory we minimize their total costs by trying different *sat* values.

wsl	sat_R (week)	sat_W (week)	sat_D (week)	sat_F (week)
0.0	1	∞	∞	∞
0.1	1	1	∞	∞
0.2	1	12	∞	∞
0.3	1	2	∞	∞
0.4	1	12	7	∞
0.5	1	11	2	1
0.6	1	13	12	7
0.7	2	14	7	16
0.8	1	10	14	9
0.9	1	1	1	1
1.0	1	1	1	1

Table 5.26. Optimum *sat* values for the retailer, the wholesaler, the distributor, and the factory when we minimize *group total cost* by trying different *sat* values.

wsl	sat_R (week)	sat_W (week)	sat_D (week)	sat_F (week)
0.0	∞	∞	∞	1
0.1	∞	∞	∞	1
0.2	∞	12	∞	1
0.3	∞	13	∞	1
0.4	8	12	16	2
0.5	1	10	2	1
0.6	4	13	12	7
0.7	3	14	7	16
0.8	3	14	14	14
0.9	1	6	1	1
1.0	1	1	1	1

In Table 5.27, we present *group total cost* values when we minimize *group total cost* by trying different *sat* values. Similar to the results obtained for the 36 week simulations, the *sat* based optimizations yield the lowest *group total cost* values for most values of the *wsl* when the wholesaler is the echelon of concern. In addition, for *wsl* equals 0.7 and 0.9, the minimum *group total cost* value is achieved when the retailer is the echelon of concern. Similar to the results obtained for the 36 week simulations, as *wsl* increases, the differences between maximum and minimum optimum *group total cost* values obtained for the different echelons of concern gets smaller.

Table 5.27. *Group total cost* values when we minimize *group total cost* for the four different echelons of concern by trying different *sat* values.

wsl	*GTC* Minimized for the Retailer ($)	*GTC* Minimized for the Wholesaler ($)	*GTC* Minimized for the Distributor ($)	*GTC* Minimized for the Factory ($)
0.0	93127.5	**42889.5**	257595.5	823558
0.1	39964.5	**24024.5**	78707.5	290018.5
0.2	19323.5	**15482**	36617	120964
0.3	13455.5	**9831.5**	14280.5	49976
0.4	9795.5	**7681.5**	12007	23623.5
0.5	8704.5	**6823.5**	9152.5	12321.5
0.6	5947.5	**5275**	7072.5	8645
0.7	**3076**	3120.5	3388.5	6300.5
0.8	2505.5	**2228.5**	3017	4253.5
0.9	**1595.5**	1866.5	2168	2322
1.0	**1432.5**	**1432.5**	**1432.5**	**1432.5**

5.3.2. Key Observations in Desired Inventory Optimization

Similar to the results obtained for 36 week simulations, the optimum values of *desired inventory* (I^*) for all echelons of concern are affected from the *wsl* value of the three identically controlled echelons; the optimum I^* values for the distributor and the factory echelons are affected more than the optimum I^* values for the retailer and the wholesaler echelons. The optimum *desired inventory* (I^*) levels for the retailer, the wholesaler, the distributor, and the factory when we minimize their total costs are presented in Table 5.28.

Table 5.28. Optimum *I** values for the retailer, the wholesaler, the distributor, and the factory when we minimize their total costs by trying different *I** values.

wsl	I_R^* (case)	I_W^* (case)	I_D^* (case)	I_F^* (case)
0.0	0	40	229	863
0.1	2	34	109	101
0.2	7	31	93	-826
0.3	6	26	32	-353
0.4	11	12	19	-129
0.5	13	8	18	-71
0.6	1	8	16	1
0.7	8	6	7	-2
0.8	2	3	4	7
0.9	0	1	0	1
1.0	0	0	0	0

Table 5.29. Optimum *I** values for the retailer, the wholesaler, the distributor, and the factory when we minimize *group total cost* by trying different *I** values.

wsl	I_R^* (case)	I_W^* (case)	I_D^* (case)	I_F^* (case)
0.0	-110	74	281	1139
0.1	-18	58	163	641
0.2	-11	57	84	385
0.3	0	41	46	226
0.4	11	28	19	70
0.5	10	27	18	-1
0.6	1	30	17	45
0.7	26	10	9	18
0.8	3	8	17	13
0.9	0	2	3	2
1.0	0	0	0	0

The optimum *desired inventory* (*I**) levels for the retailer, the wholesaler, the distributor, and the factory when we minimize *group total cost* are presented in Table 5.29. This time, the optimum *I** values for the retailer are significantly affected from the changes in the *wsl* values like the distributor and the factory echelons, but the strength of the effect on the optimum *I** values for the wholesaler still is lower than the other three echelons, similar to the results for 36 week simulations.

In Table 5.30, we present *group total cost* values when we minimize *group total cost* by trying different *desired inventory* levels. The *I** based optimizations yield the lowest *group total cost* values for most values of the *wsl* when the wholesaler is the echelon of concern However, for *wsl* equals to 0.9, the minimum *group total cost* value is achieved when the retailer is the echelon of concern. In addition, similar to the results obtained from 36 week simulation, as *wsl* increases, the differences between maximum and minimum optimum *group total cost* values obtained for the different echelons of concern gets smaller.

Table 5.30. *Group total cost* values when we minimize *group total cost* for the four different echelons of concern by trying different *desired inventory* levels.

wsl	*GTC* Minimized for the Retailer ($)	*GTC* Minimized for the Wholesaler ($)	*GTC* Minimized for the Distributor ($)	*GTC* Minimized for the Factory ($)
0.0	119727	**32034.5**	77278	342114
0.1	56376.5	**20425.5**	43569.5	154266.5
0.2	24552	**13549.5**	24779.5	75915.5
0.3	13995.5	**10884**	17945.5	38031
0.4	8879	**7678**	11431	20052.5
0.5	8115.5	**5026.5**	7308.5	12023
0.6	4328	**3324.5**	5109	9239.5
0.7	3874.5	**2796**	3204	7639.5
0.8	2368.5	**2358.5**	2764	3528
0.9	**1595.5**	1920.5	2085.5	2184.5
1.0	**1432.5**	**1432.5**	**1432.5**	**1432.5**

When we compare the cost values in Table 5.27 and Table 5.30, we observe that keeping *desired inventory* at a level different than zero in general gives a better instance of the anchor-and-adjust ordering policy compared to assigning a value to *sat* that is greater than one week.

6. A SINGLE ECHELON WITH A SUB-OPTIMAL CONTROL

In this chapter, we first report the benchmark group total cost obtained when all four echelons manage their inventories optimally (i.e., *sat* = 1 week and *wsl* = 1 for all echelons). Later, we obtain results by assigning zero to *wsl* of a selected echelon, which represents a sub-optimal control for that echelon, while keeping the rest of the parameters of the benchmark as they are. Sub-optimal inventory management of the distributor has a worse effect on the group total cost ($4,446.5 for 36 weeks and $10,986.5 for 144 weeks) compared to the effect of the factory ($1,837.5 for 36 weeks and $4,179.5 for 144 weeks), the effect of the wholesaler ($8,171.5 for 36 weeks and $26,643 for 144 weeks) is worse than the effect of the distributor, and the retailer has the worst effect ($12,866 for 36 weeks and $54,001 for 144 weeks) compared to all (see Table 6.1and Table 6.2).

The benchmark *group total cost* values are $1,428.5 for 36 weeks and $1,432.5 for 144 weeks. To obtain the benchmark *group total cost*, the main parameter setting that is explained in Chapter 3 is used. The experimental parameters are set as below:

- *I** = 0 cases for all echelons
- *sat* =1 week for all echelons
- *wsl* = 1 for all echelons

Table 6.1. Sub-optimal total cost values obtained by assigning zero to *wsl* of one of the echelons (for 36 weeks).

	$wsl_R=0$ $wsl_W=1$ $wsl_D=1$ $wsl_F=1$	$wsl_R=1$ $wsl_W=0$ $wsl_D=1$ $wsl_F=1$	$wsl_R=1$ $wsl_W=1$ $wsl_D=0$ $wsl_F=1$	$wsl_R=1$ $wsl_W=1$ $wsl_D=1$ $wsl_F=0$
TC_R (\$)	2845.5	534	491	455
TC_W (\$)	4225	2571.5	447	416
TC_D (\$)	3628.5	3011.5	1902.5	387
TC_F (\$)	2167	2054.5	1606	579.5
GTC (\$)	12866	8171.5	4446.5	1837.5
Percent increase in GTC	800.67	472.03	211.27	28.63

Table 6.2. Sub-optimal total cost values obtained by assigning zero to *wsl* of one of the echelons (for 144 weeks).

	$wsl_R=0$ $wsl_W=1$ $wsl_D=1$ $wsl_F=1$	$wsl_R=1$ $wsl_W=0$ $wsl_D=1$ $wsl_F=1$	$wsl_R=1$ $wsl_W=1$ $wsl_D=0$ $wsl_F=1$	$wsl_R=1$ $wsl_W=1$ $wsl_D=1$ $wsl_F=0$
TC_R (\$)	9952	864	909	915
TC_W (\$)	12406.5	6794.5	865	870
TC_D (\$)	16185	9544	4278.5	817
TC_F (\$)	15457.5	9440.5	4934	1577.5
GTC (\$)	54001	26643	10986.5	4179.5
Percent increase in GTC	3669.70	1759.90	666.95	191.76

7. CONCLUSIONS

In this study, we first constructed a detailed mathematical model that represents and replicates the exact execution order of the steps of the original board version of The Beer Game. In Chapter 2, we state the difficulties that we faced in the construction process of such an exact one-to-one replica. One of the main difficulties is an error regarding the conceptualization of the delay durations as explained in detail in Section 2.3. We present the constructed model in full precision including necessary assumptions, explanations, and units for all parameters and variables in Section 2.1. To increase the usability of the model presented in this paper, we write an R code of the model that is given in Appendix A. For extendibility, in Section 2.2, we shortly discuss how the code given in Appendix A can be used in experimentation and how it can be used to create a single-player or multi-player beer game on a computer. In Section 2.4, we verify that the constructed mathematical model is a correct representation of The Beer Game.

We conduct experiments for two different objectives: (i) we minimize the total cost of the echelon of interest or (ii) we minimize the group total cost. Except for a few cases, these two objectives results in different instances of the anchor-and-adjust heuristic, especially when the three other echelons underweight their own individual supply lines. In general, our results show that, by optimizing the decision parameters for the echelon of concern, the group total cost value can be decreased when the group total cost is minimized instead of minimizing the cost of the selected echelon. This result implies that the echelon of concern can decrease the expected group total cost, which also includes the total cost of that echelon, by allowing an increase in its own expected total cost by changing the policy instance he uses. This new instance of policy will be different than the optimum policy instance obtained when all group members use the optimal policy instance.

Sub-optimal inventory management of the distributor has a worse effect on the group total cost compared to the effect of the factory, the effect of the wholesaler is worse than the effect of the distributor, and the retailer has the worst effect compared to all. We were thus expecting the retailer to be the most effective echelon in reducing the group total cost when we sacrifice the objective of minimizing the retailer's total cost. Unexpectedly, we obtained the lowest group total costs for the wholesaler when we minimized the group total cost by sacrificing the objective of minimizing the cost of the echelon of concern. Therefore, a group of people playing The Beer Game for the first time should place the group member who has the most experience and/or knowledge in managing inventories to the wholesaler position to be able to obtain the least group total cost.

We obtained different instances of the anchor-and-adjust ordering policy by optimizing with respect to *stock adjustment time* and with respect to *desired inventory* of the selected echelon. All echelons make adjustments in a week while minimizing their own costs, but the retailer, the wholesaler, and the distributor echelons tend to make mild or no adjustments while minimizing the group total cost; the factory does not change its behavior, it continues to make adjustments in a week. If the group members are not knowledgeable in managing inventories or have no experience as such, we suggest the participant in the retailer, in the wholesaler, or in the distributor position to make smooth adjustments. However, the participant in the factory position should behave as aggressively as possible in inventory and supply line adjustments. In optimizations with respect to *stock adjustment time*, the retailer is the second effective echelon after the wholesaler; the factory is ineffective in reducing the group total cost by sacrificing his own total cost. Making mild or no adjustments implies that the echelon of concern should give orders equal to (or close to) the expected demands of its customer. In other words, it can be said that it avoids adjusting its effective inventory and supply line towards their desired levels because any adjustments brings instabilities to the system. The selection of a high or infinite *stock adjustment time* value results in negative effective inventory level (i.e., backlog) for the echelon of interest. However, when the echelon of concern minimizes its own cost, it makes relatively more aggressive adjustments to bring its own inventory to the desired level, which results in costly oscillations in the other three echelons' inventories.

We were expecting the optimum values of *desired inventory* of a selected echelon to be zero because if *effective inventory* is zero for an echelon in a week, that echelon produces no costs in that week. Surprisingly, when the group members other than the echelon of concern sub-optimally manage their inventories, having a non-zero *desired inventory* can reduce costs for both the echelon of concern and the whole supply chain. If the group members are not knowledgeable in managing inventories or have no experience as such, we suggest the participant who is assumed to be knowledgeable to have a non-zero desired inventory regardless of his position. Moreover, in general, optimizing with respect to *desired inventory* level produces better results than optimizing with respect to *stock adjustment time* according to the results. Except for the retailer, an echelon of concern tends to have a higher desired inventory level when optimizing the group total cost, as compared to optimizing its own total cost. Although the factory was not effective in reducing the group total cost in optimizations for *stock adjustment time*, it is effective in optimizations for *desired inventory*. Unpredictably, in optimizations for *desired inventory*, the distributor is the second effective echelon after the wholesaler.

After conducting our experiments in the original setting of The Beer Game, we increased the final time of the simulation runs from 36 weeks to 144 weeks to observe the effect of final time on the results. Similar to the results obtained for the 36 week simulations, the wholesaler remained to be the most effective echelon in reducing the group total cost by sacrificing its own objective of minimizing its own echelon's cost. Different than the results obtained for the 36 week simulations, optimizing *desired inventory* level produces better results than optimizing *stock adjustment time* in lesser number of cases. If the group members are not knowledgeable in managing inventories or have no experience as such, we can still suggest the participant who is the knowledgeable one to keep a desired inventory level different than zero. Different than 36 week simulations, in optimizations for *stock adjustment time*, all echelons except for the factory start to make stronger corrections when optimizing the group total cost value. All echelons except for the retailer start to make milder corrections when optimizing their own total cost values compared to the results obtained for 36 week simulations. Relatively aggressive corrections create difficulties for the other echelons. We presume, there is no bad effect reflecting on the echelon making strong corrections in the 36 week simulations. However, this is not the case in 144 week simulations; the difficulties faced by the other echelons start to have some effect on the echelon making strong corrections, after some time. In 144 week simulations, when one week is assigned to *stock adjustment time* and unity is assigned to *weight of supply line* for all the echelons, the optimum *desired inventory* values are obtained as zero cases for all echelons and for both of the objectives, which supports the expectation that the long-term optimum values of *desired inventory* levels should all be zero.

We repeated our experiments by conducting multi-dimensional optimizations for both *stock adjustment time* and *desired inventory*. However, we were unable to obtain a significant interaction effect. We are planning to continue the research by repeating our experiments for different jump sizes in the end-customer demand. We are also planning to use varying demand in the experiments. Another idea for future research is to use different cost functions.

APPENDIX A: R CODE OF THE MATHEMATICAL MODEL (EQUATIONS 2.1-2.70)

```
# Appendix A: R code of the mathematical model (equations 1-
70)
# VARIABLE CREATION
# In this segment of the code, variables are created by
# assigning dummy values to them, which are not used in
# the simulation. This step is necessary in R.

endcd = 0

s_endc = 0
s_r = 0
s_w = 0
s_d = 0

ti_r = 0
ti_w = 0
ti_d = 0
ti_f = 0

o_r = 0
o_w = 0
o_d = 0
psr = 0

io_w = 0
io_d = 0
io_f = 0

ei_r = 0
```

```
ei_w = 0
ei_d = 0
ei_f = 0

sl_r = 0
sl_w = 0
sl_d = 0
sl_f = 0

sla_r = 0
sla_w = 0
sla_d = 0
sla_f = 0

ia_r = 0
ia_w = 0
ia_d = 0
ia_f = 0

# PARAMETERS AND INITAL VALUES OF THE MATHEMATICAL MODEL
# In order to ease the comparison of the mathematical
# model and the R code, the equation numbers are also
# provided in the R code. For example #1, #2, #3, and so
# on and so forth.

sat_r = 1       #1
sat_w = 1       #1
sat_d = 1       #1
sat_f = 1       #1

mdt_r = 1       #2
mdt_w = 1       #2
mdt_d = 1       #2
```

```
st_w = 2        #3
st_d = 2        #3
st_f = 2        #3

plt = 2         #4

wsl_r = 1       #5
wsl_w = 1       #5
wsl_d = 1       #5
wsl_f = 1       #5

theta_r = 0.2   #6
theta_w = 0.2   #6
theta_d = 0.2   #6
theta_f = 0.2   #6

# In R, an index of zero cannot be used in an array. For
# example, the first element of an array of variable x,
# is represented as x[1]. Therefore, t in R corresponds
# to t-1 in the mathematical model.

for(t in 2:5)   #7 (corresponds to t = 1 to 4
endcd[t] = 4    #  in the mathematical model)

for(t in 6:37)  #7 (corresponds to t = 5 to 36
endcd[t] = 8    #  in the mathematical model)

eecd = 4        #8
eo_r = 4        #9
eo_w = 4        #9
eo_d = 4        #9
```

```
di_r = 0        #10
di_w = 0        #10
di_d = 0        #10
di_f = 0        #10

dsl_r = eecd * (mdt_r + st_w)       #11
dsl_w = eo_r * (mdt_w + st_d)       #12
dsl_d = eo_w * (mdt_d + st_f)       #13
dsl_f = eo_d * plt                  #14

b_r = 0             #15
b_w = 0             #15
b_d = 0             #15
b_f = 0             #15

i_r = 12            #16
i_w = 12            #16
i_d = 12            #16
i_f = 12            #16

iti1_r = 4          #17
iti1_w = 4          #17
iti1_d = 4          #17

wipi1 = 4           #18

iti2_r = 4          #19
iti2_w = 4          #19
iti2_d = 4          #19

wipi2 = 4           #20

o_r[2] = 4          #21
```

```
o_w[2] = 4          #21
o_d[2] = 4          #21

psr[2] = 4          #22

io_w[2] = 4         #23
io_d[2] = 4         #23
io_f[2] = 4         #23

tc_r = 0            #24
tc_w = 0            #24
tc_d = 0            #24
tc_f = 0            #24

uihc = 0.5          #25
ubc = 1             #26

# START OF THE SIMULATION-FOR-LOOP

for(t in 2:37)  # (corresponds to t = 1 to 36
{               # in the mathematical model)

#####Step 1#####

ti_r[t] = i_r[t-1] + iti2_r[t-1]    #27
ti_w[t] = i_w[t-1] + iti2_w[t-1]    #27
ti_d[t] = i_d[t-1] + iti2_d[t-1]    #27

ti_f[t] = i_f[t-1] + wipi2[t-1] #28

iti2_r[t] = iti1_r[t-1]         #29
iti2_w[t] = iti1_w[t-1]         #29
iti2_d[t] = iti1_d[t-1]         #29
```

```
wipi2[t] = wipi1[t-1]        #30

iti1_r[t] = 0                #31
iti1_w[t] = 0                #31
iti1_d[t] = 0                #31

wipi1[t] = 0                 #32

#####Step 2#####

s_endc[t] = min(ti_r[t], b_r[t-1] + endcd[t])   #33
s_r[t] = min(ti_w[t], b_w[t-1] + io_w[t])       #34
s_w[t] = min(ti_d[t], b_d[t-1] + io_d[t])       #35
s_d[t] = min(ti_f[t], b_f[t-1] + io_f[t])       #36

iti1_r[t] = s_r[t]           #37
iti1_w[t] = s_w[t]           #37
iti1_d[t] = s_d[t]           #37

wipi1[t] = psr[t]            #38

#####Step 3#####

b_r[t] = b_r[t-1] + endcd[t] - s_endc[t]    #39
b_w[t] = b_w[t-1] + io_w[t] - s_r[t]        #40
b_d[t] = b_d[t-1] + io_d[t] - s_w[t]        #41
b_f[t] = b_f[t-1] + io_f[t] - s_d[t]        #42

i_r[t] = ti_r[t] - s_endc[t]       #43
i_w[t] = ti_w[t] - s_r[t]          #44
i_d[t] = ti_d[t] - s_w[t]          #45
i_f[t] = ti_f[t] - s_d[t]          #46
```

```
#####Expectation Formation#####

eecd[t] = eecd[t-1] + theta_r * (endcd[t] - eecd[t-1])
    #47
eo_r[t] = eo_r[t-1] + theta_w * (io_w[t] - eo_r[t-1])
    #48
eo_w[t] = eo_w[t-1] + theta_d * (io_d[t] - eo_w[t-1])
    #49
eo_d[t] = eo_d[t-1] + theta_f * (io_f[t] - eo_d[t-1])
    #50

#####Step 4#####

io_w[t+1] = o_r[t]  #51
io_d[t+1] = o_w[t]  #52
io_f[t+1] = o_d[t]  #53

#####Step 5#####

dsl_r[t] = eecd[t] * (mdt_r + st_w) #54
dsl_w[t] = eo_r[t] * (mdt_w + st_d) #55
dsl_d[t] = eo_w[t] * (mdt_d + st_f) #56
dsl_f[t] = eo_d[t] * plt            #57

ei_r[t] = i_r[t] - b_r[t]       #58
ei_w[t] = i_w[t] - b_w[t]       #58
ei_d[t] = i_d[t] - b_d[t]       #58
ei_f[t] = i_f[t] - b_f[t]       #58

sl_r[t] = io_w[t+1] + b_w[t] + iti1_r[t] + iti2_r[t]    #59
sl_w[t] = io_d[t+1] + b_d[t] + iti1_w[t] + iti2_w[t]    #60
sl_d[t] = io_f[t+1] + b_f[t] + iti1_d[t] + iti2_d[t]    #61
```

```
sl_f[t] = wipi1[t] + wipi2[t]                          #62

sla_r[t] = wsl_r * (dsl_r[t] - sl_r[t]) / sat_r     #63
sla_w[t] = wsl_w * (dsl_w[t] - sl_w[t]) / sat_w     #63
sla_d[t] = wsl_d * (dsl_d[t] - sl_d[t]) / sat_d     #63
sla_f[t] = wsl_f * (dsl_f[t] - sl_f[t]) / sat_f     #63

ia_r[t] = (di_r - ei_r[t]) / sat_r          #64
ia_w[t] = (di_w - ei_w[t]) / sat_w          #64
ia_d[t] = (di_d - ei_d[t]) / sat_d          #64
ia_f[t] = (di_f - ei_f[t]) / sat_f          #64

if(t <= 5)      # (corresponds to t <= 4
{               # in the mathematical model)
o_r[t+1] = 4    #65
o_w[t+1] = 4    #66
o_d[t+1] = 4    #67
psr[t+1] = 4    #68
}
else
{
o_r[t+1] = floor(max(eecd[t] + ia_r[t] + sla_r[t], 0) + 0.5)
    #65
o_w[t+1] = floor(max(eo_r[t] + ia_w[t] + sla_w[t], 0) + 0.5)
    #66
o_d[t+1] = floor(max(eo_w[t] + ia_d[t] + sla_d[t], 0) + 0.5)
    #67
psr[t+1] = floor(max(eo_d[t] + ia_f[t] + sla_f[t], 0) + 0.5)
    #68
}

# In the mathematical model, we assume that the values are
# rounded using the "round half away from zero" tie-breaking
```

```
# rule. However, the function "round" in R uses the "round
# half alternatingly" rule. After adding 0.5 to the values
# to be rounded, the "floor" function confirms to our assumed
# tie-breaking rule only in the presence of non-negative
# orders, which is guaranteed by the "max" function.

tc_r[t] = tc_r[t-1] + (b_r[t] + 0.5 * i_r[t])       #69
tc_w[t] = tc_w[t-1] + (b_w[t] + 0.5 * i_w[t])       #69
tc_d[t] = tc_d[t-1] + (b_d[t] + 0.5 * i_d[t])       #69
tc_f[t] = tc_f[t-1] + (b_f[t] + 0.5 * i_f[t])       #69
}

# END OF THE SIMULATION-FOR-LOOP

# TOTAL COSTS OBTAINED AT EACH ECHELON ARE REPORTED

tc_r[37]
tc_w[37]
tc_d[37]
tc_f[37]

# The above cost values correspond to the costs obtained
# at the end of the game (t = 36 in the mathematical model)

# THE GROUP TOTAL COST IS CALCULATED AND REPORTED

gtc = tc_r[37] + tc_w[37] + tc_d[37] + tc_f[37]     #70

gtc
```

APPENDIX B: R CODE OF THE MATHEMATICAL MODEL (EQUATIONS 2.1-2.9, 2.15-2.53, 2.58-2.62, 2.69-2.70, and 2.72-2.75)

```
# Appendix B: R code of the mathematical model
# (equations 1-9, 15-53, 58-62, 69-70, and 72-75)
# VARIABLE CREATION
# In this segment of the code, variables are created by
# assigning dummy values to them, which are not used in
# the simulation. This step is necessary in R.

endcd = 0

s_endc = 0
s_r = 0
s_w = 0
s_d = 0

ti_r = 0
ti_w = 0
ti_d = 0
ti_f = 0

o_r = 0
o_w = 0
o_d = 0
psr = 0

io_w = 0
io_d = 0
io_f = 0

ei_r = 0
```

```
ei_w = 0
ei_d = 0
ei_f = 0

sl_r = 0
sl_w = 0
sl_d = 0
sl_f = 0

# PARAMETERS AND INITAL VALUES OF THE MATHEMATICAL MODEL
# In order to ease the comparison of the mathematical
# model and the R code, the equation numbers are also
# provided in the R code. For example #1, #2, #3, and so
# on and so forth.

sat_r = 1       #1
sat_w = 1       #1
sat_d = 1       #1
sat_f = 1       #1

mdt_r = 1       #2
mdt_w = 1       #2
mdt_d = 1       #2

st_w = 2        #3
st_d = 2        #3
st_f = 2        #3

plt = 2         #4

wsl_r = 1       #5
wsl_w = 1       #5
wsl_d = 1       #5
```

```
wsl_f = 1       #5

theta_r = 0     #6
theta_w = 0     #6
theta_d = 0     #6
theta_f = 0     #6

# In R, an index of zero cannot be used in an array. For
# example, the first element of an array of variable x,
# is represented as x[1]. Therefore, t in R corresponds
# to t-1 in the mathematical model.

for(t in 2:5)   #7 (corresponds to t = 1 to 4
endcd[t] = 4    #  in the mathematical model)

for(t in 6:37)  #7 (corresponds to t = 5 to 36
endcd[t] = 8    #  in the mathematical model)

eecd = 4        #8
eo_r = 4        #9
eo_w = 4        #9
eo_d = 4        #9

b_r = 0         #15
b_w = 0         #15
b_d = 0         #15
b_f = 0         #15

i_r = 12        #16
i_w = 12        #16
i_d = 12        #16
i_f = 12        #16
```

```
iti1_r = 4       #17
iti1_w = 4       #17
iti1_d = 4       #17

wipi1 = 4        #18

iti2_r = 4       #19
iti2_w = 4       #19
iti2_d = 4       #19

wipi2 = 4        #20

o_r[2] = 4       #21
o_w[2] = 4       #21
o_d[2] = 4       #21

psr[2] = 4       #22

io_w[2] = 4      #23
io_d[2] = 4      #23
io_f[2] = 4      #23

tc_r = 0         #24
tc_w = 0         #24
tc_d = 0         #24
tc_f = 0         #24

uihc = 0.5       #25
ubc = 1          #26

s_prime_r = 28
s_prime_w = 28
s_prime_d = 28
```

```
s_prime_f = 20

# START OF THE SIMULATION-FOR-LOOP

for(t in 2:37)  # (corresponds to t = 1 to 36
{               # in the mathematical model)

#####Step 1#####

ti_r[t] = i_r[t-1] + iti2_r[t-1]    #27
ti_w[t] = i_w[t-1] + iti2_w[t-1]    #27
ti_d[t] = i_d[t-1] + iti2_d[t-1]    #27

ti_f[t] = i_f[t-1] + wipi2[t-1] #28

iti2_r[t] = iti1_r[t-1]         #29
iti2_w[t] = iti1_w[t-1]         #29
iti2_d[t] = iti1_d[t-1]         #29

wipi2[t] = wipi1[t-1]           #30

iti1_r[t] = 0               #31
iti1_w[t] = 0               #31
iti1_d[t] = 0               #31

wipi1[t] = 0                #32

#####Step 2#####

s_endc[t] = min(ti_r[t], b_r[t-1] + endcd[t])   #33
s_r[t] = min(ti_w[t], b_w[t-1] + io_w[t])       #34
s_w[t] = min(ti_d[t], b_d[t-1] + io_d[t])       #35
s_d[t] = min(ti_f[t], b_f[t-1] + io_f[t])       #36
```

```
iti1_r[t] = s_r[t]          #37
iti1_w[t] = s_w[t]          #37
iti1_d[t] = s_d[t]          #37

wipi1[t] = psr[t]           #38

#####Step 3#####

b_r[t] = b_r[t-1] + endcd[t] - s_endc[t]    #39
b_w[t] = b_w[t-1] + io_w[t] - s_r[t]        #40
b_d[t] = b_d[t-1] + io_d[t] - s_w[t]        #41
b_f[t] = b_f[t-1] + io_f[t] - s_d[t]        #42

i_r[t] = ti_r[t] - s_endc[t]        #43
i_w[t] = ti_w[t] - s_r[t]           #44
i_d[t] = ti_d[t] - s_w[t]           #45
i_f[t] = ti_f[t] - s_d[t]           #46

#####Expectation Formation#####

eecd[t] = eecd[t-1] + theta_r * (endcd[t] - eecd[t-1])
    #47
eo_r[t] = eo_r[t-1] + theta_w * (io_w[t] - eo_r[t-1])
    #48
eo_w[t] = eo_w[t-1] + theta_d * (io_d[t] - eo_w[t-1])
    #49
eo_d[t] = eo_d[t-1] + theta_f * (io_f[t] - eo_d[t-1])
    #50

#####Step 4#####

io_w[t+1] = o_r[t]  #51
```

```
io_d[t+1] = o_w[t]  #52
io_f[t+1] = o_d[t]  #53

#####Step 5######

ei_r[t] = i_r[t] - b_r[t]       #58
ei_w[t] = i_w[t] - b_w[t]       #58
ei_d[t] = i_d[t] - b_d[t]       #58
ei_f[t] = i_f[t] - b_f[t]       #58

sl_r[t] = io_w[t+1] + b_w[t] + iti1_r[t] + iti2_r[t]    #59
sl_w[t] = io_d[t+1] + b_d[t] + iti1_w[t] + iti2_w[t]    #60
sl_d[t] = io_f[t+1] + b_f[t] + iti1_d[t] + iti2_d[t]    #61
sl_f[t] = wipi1[t] + wipi2[t]                           #62

if(t <= 5)      # (corresponds to t <= 4
{               # in the mathematical model)
o_r[t+1] = 4    #72
o_w[t+1] = 4    #73
o_d[t+1] = 4    #74
psr[t+1] = 4    #75
}
else
{
o_r[t+1] = floor(max(eecd[t] + (s_prime_r - ei_r[t]
- wsl_r * sl_r[t])/sat_r, 0) + 0.5) #72
o_w[t+1] = floor(max(eo_r[t] + (s_prime_w - ei_w[t]
- wsl_w * sl_w[t])/sat_w, 0) + 0.5) #73
o_d[t+1] = floor(max(eo_w[t] + (s_prime_d - ei_d[t]
- wsl_d * sl_d[t])/sat_d, 0) + 0.5) #74
psr[t+1] = floor(max(eo_d[t] + (s_prime_f - ei_f[t]
- wsl_f * sl_f[t])/sat_f, 0) + 0.5) #75
}
```

```
# In the mathematical model, we assume that the values are
# rounded using the "round half away from zero" tie-breaking
# rule. However, the function "round" in R uses the "round
# half alternatingly" rule. After adding 0.5 to the values
# to be rounded, the "floor" function confirms to our assumed
# tie-breaking rule only in the presence of non-negative
# orders, which is guaranteed by the "max" function.

tc_r[t] = tc_r[t-1] + (b_r[t] + 0.5 * i_r[t])       #69
tc_w[t] = tc_w[t-1] + (b_w[t] + 0.5 * i_w[t])       #69
tc_d[t] = tc_d[t-1] + (b_d[t] + 0.5 * i_d[t])       #69
tc_f[t] = tc_f[t-1] + (b_f[t] + 0.5 * i_f[t])       #69
}

# END OF THE SIMULATION-FOR-LOOP

# TOTAL COSTS OBTAINED AT EACH ECHELON ARE REPORTED

tc_r[37]
tc_w[37]
tc_d[37]
tc_f[37]

# The above cost values correspond to the costs obtained
# at the end of the game (t = 36 in the mathematical model)

# THE GROUP TOTAL COST IS CALCULATED AND REPORTED

gtc = tc_r[37] + tc_w[37] + tc_d[37] + tc_f[37]     #70

gtc
```

APPENDIX C: THE RELATIONSHIP BETWEEN ANCHOR-AND-ADJUST ORDERING POLICY AND ORDER-UP-TO-S POLICY

The anchor-and-adjust ordering policy and the order-up-to-S policy are equivalent to each other for specific set of parameter values (Yasarcan, 2012).

The ordering equation of the anchor-and-adjust heuristic is as follows:

$$O = \hat{E} + \left(\frac{I^* - EI}{SAT} \right) + wsl \cdot \left(\frac{SL^* - SL}{SAT} \right) \tag{C.1}$$

O stands for the order quantity, $\hat{E}$ stands for the expected orders, *I** stands for *desired inventory*, *EI* stands for *effective inventory*, *wsl* stands for *weight of supply line*, *SL** stands for *desired supply line*, *SL* stands for *supply line*, and *SAT* stands for *stock adjustment time*. Note that, *EI* is equal to the difference between inventory and backlog level.

SAT and *wsl* are the two parameters that are not used in inventory management literature. When they are taken as one and unity respectively, Equation C.1 turns out to be:

$$O = \hat{E} + \left(I^* - EI \right) + \left(SL^* - SL \right) \tag{C.2}$$

Re-arranging the terms of Equation C.2 yields:

$$O = \left(\hat{E} + I^* + SL^* \right) - \left(EI + SL \right) \tag{C.3}$$

The order-up-to-S policy can be given as:

$$O = S - IP \tag{C.4}$$

where, *IP* is *inventory position* and *S* is the base stock level.

Comparing Equations C.3 and C.4 yields the following two equations:

$$IP = EI + SL \tag{C.5}$$

$$S = \hat{E} + I^{*} + SL^{*} \tag{C.6}$$

According to Equation C.5, *inventory position* has two terms: *effective inventory* and *supply line* (i.e., *pipeline inventory*). According to Equation C.6, base stock level consists of three terms. *SL** is the expected demand during *lead time* (*LT*) and it is equal to expected orders ($\hat{E}$) times *lead time* (*LT*). Therefore, Equation C.6 can be re-written as:

$$S = \hat{E} \cdot (LT + 1) + I^{*} \tag{C.7}$$

*I** can be replaced by *safety stock* (*SS*). Therefore, Equation C.7 can be re-written as:

$$S = \hat{E} \cdot (LT + 1) + SS \tag{C.8}$$

REFERENCES

Akkermans, H. and B. Vos, 2003, "Amplification in Service Supply Chains: An Exploratory Case Study from the Telecom Industry", *Production and Operations Management*, Vol. 12, No. 2, pp. 204-223.

Axelrod, R., 1997, "Advancing the Art of Simulation in the Social Sciences", *Complexity*, Vol. 3, No. 2, pp. 16-22.

Barlas, Y., 2002, "System Dynamics: Systemic Feedback Modeling for Policy Analysis", *Knowledge for Sustainable Development: An Insight into the Encyclopedia of Life Support Systems*, UNESCO/EOLSS, Paris and Oxford, pp. 1131-1175.

Barlas, Y. and M.G. Ozevin, 2004, "Analysis of Stock Management Gaming Experiments and Alternative Ordering Formulations", *Systems Research and Behavioral Science*, Vol. 21, No. 4, pp. 439-470.

Chaharsooghi, S.K., J. Heydari and S.H. Zegordi, 2008, "A Reinforcement Learning Model for Supply Chain Ordering Management: An Application to the Beer Game", *Decision Support Systems*, Vol. 45, No. 4, pp. 949-959.

Chen, F. and R. Samroengraja, 2000, "The Stationary Beer Game", *Production and Operations Management*, Vol. 9, No. 1, pp. 19-30.

Croson, R. and K. Donohue, 2006, "Behavioral Causes of the Bullwhip Effect and the Observed Value of Inventory Information", *Management Science*, Vol. 52, No. 3, pp. 323-336.

Day, J.M. and M. Kumar, 2010, "Using SMS Text Messaging to Create Individualized and Interactive Experiences in Large Classes: A Beer Game Example", *Decision Sciences Journal of Innovative Education*, Vol. 8, No. 1, pp. 129-136.

Forrester, J.W, 1961, *Industrial Dynamics*, Pegasus Communications, Waltham, MA.

Forrester, J.W., 1971, *Principles of Systems*, Pegasus Communications, Waltham, MA.

Gardner, E.S., 1985, "Exponential Smoothing: The State of the Art", *Journal of Forecasting*, Vol. 4, No. 1, pp. 1-28.

Größler, A., J.-H. Thun and P.M. Milling, 2008, "System Dynamics as a Structural Theory in Operations Management", *Production and Operations Management*, Vol. 17, No. 3, pp. 373-384.

Jacobs, F.R., 2000, "Playing the Beer Distribution Game over the Internet", *Production and Operations Management*, Vol. 9, No. 1, pp. 31-39.

Montgomery, D.C. and L.A. Johnson, 1976, *Forecasting and Time Series Analysis*, McGraw-Hill, New York.

Mosekilde, E. and J.L. Laugesen, 2007, "Nonlinear Dynamic Phenomena in the Beer Model", *System Dynamics. Review*, Vol. 23, No. 2-3, pp. 229-252.

Mutallip, A., 2013, *Deciding on Parameter Values of Anchor and Adjust Heuristic in Stock Management*, M.S. Thesis, Bogazici University.

R Core Team, 2013, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org/, accessed at July 2013.

Samur, M.G., E.C. Eren, O. Hancer and B.B. Tunakan, 2004, *Testing Simultaneous Multiple Decision Maker Characteristics in a Dynamic Feedback Environment*, B.S. Project, Bogazici University.

Samur, M.G., E.C. Eren, O. Hancer and B.B. Tunakan, 2005, "A Computerized Beer Game and Decision-Making Experiments", In: System Dynamics Society, *Proceedings of the 23rd International System Dynamics Conference*. Boston, MA.

Steckel, J.H., S. Gupta and A. Banerji, 2004, "Supply Chain Decision Making: Will Shorter Cycle Times and Shared Point-of-sale Information Necessarily Help?", *Management Science*, Vol. 50, No. 4, pp. 458-464.

Sterman, J.D., 1989, "Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment", *Management Science*, Vol. 35, No. 3, pp. 321-339.

Sterman, J.D., 2000, *Business Dynamics: Systems Thinking and Modeling for a Complex World*, Irwin/McGraw-Hill, Boston, MA.

Thomsen, J.S., E. Mosekilde and J.D. Sterman, 1991, "Hyperchaotic phenomena in dynamic decision making", in: E. Mosekilde, L. Mosekilde (Eds.), *Complexity, Chaos, and Biological Evolution*, Plenum Press, New York, pp. 397-420.

Yasarcan, H. and Y. Barlas, 2005a, "A Generalized Stock Control Formulation for Stock Management Problems Involving Composite Delays and Secondary Stocks", *System Dynamics Review*, Vol. 21, No. 1, pp. 33–68.

Yasarcan, H. and Y. Barlas, 2005b, "Stable Stock Management Heuristics when the Outflow is Proportional to the Control Stock", In: System Dynamics Society, *Proceedings of the 23rd International System Dynamics Conference*. Boston, MA.

Yasarcan, H., 2012, Unpublished Lecture Notes, IE-580 Stock Management in System Dynamics, Boğazici University.