Imperial College
London

Supply Chain Business Modelling

Gabriela Pereira

A thesis submitted for the degree of Doctor of Philosophy

of the University of London and Diploma of Imperial College London

Centre of Process Engineering Systems

Dept. of Chemical Engineering & Chemical Technology

South Kensington Campus, London, SW7 2AZ, UK

May 2012

ABSTRACT

The developed work is motivated by the hypothesis that the presented Supply Chain Business Model is a practical and comprehensive approach to support not only operational day-to-day business decisions, but most importantly strategic and long term decisions that may define the success and the longevity of a business.

Conceptually, the Business Supply Chain Model developed in this thesis replicates the behaviour and decision making of the different agents in a supply chain, and an Optimisation Module determines the optimised parameters that maximise the overall business profit, whatever scenario it may be. In the optimisation module, a Genetic Algorithm was used to determine the best equation parameters for each individual agent that optimise the overall supply chain profit. Furthermore, several business case-scenarios are presented and the findings highlighted. These case-scenarios prove that: the HC model is robust when subjected to predictable or unpredictable causes of variability; the bullwhip effect can be reduced significantly by applying GA as the optimisation tool; the improvement of profits needs to be evaluated at a global scale, independently of the individual agents' profit; impact of supply shortages in the SC ; retail expansion analysis; delivery patterns change impact in profitability; impact of sourcing decisions in the SC profitability; model suitability for seasonal vs. non-seasonal products.

The SC Modelling framework generic and globalising approach means that is easily applied and transposed to any other business realities and it can be easily changed to reflect other SC scenarios. The costing model associated means that, at any point in the network, all costs and profits can be easily measured. For the first time the shelf-life of a product captured and losses of product due to BBE dates, quantified. In this model the optimisation methodology runs parallel to the developed simulation tool, so the optimisation should be only run for new scenarios.

ACKNOWLEDGMENTS

"Your work is going to fill a large part of your life, and the only way to be truly satisfied is to do what you believe is great work. And the only way to do great work is to love what you do..."

Steve Jobs

I have been fortunate to study a subject that I find extremely interesting, and I have been lucky to be able to discuss about such an inspirational brand as Hotel Chocolat, which both customers and staff are truly passionate about. This has been a long and challenging journey only made possible by the support of so many people who encouraged, helped and supported me throughout the course of these part-time studies.

Initial thanks must be given to Hotel Chocolat, and specially Hotel Chocolat CEO, Mr Angus Thirlwell, for allowing me the opportunity to undertake these studies alongside my regular day-to-day employment and allowing me to use the company's data and information.

My thank you to my parents, brother and friends who emotionally supported and encouraged me for the past few years. My special thanks to the one without whom this work would have being impossible and to whom I am eternally grateful.

Finally, I would especially like to thank Professor Nilay Shah, my supervisor at Imperial College, for his invaluable support and advice throughout the course of my studies and for being always an optimist.

DECLARATION OF ORIGINALITY

I hereby declare that the dissertation "Supply Chain Business Modelling" is substantially my own unaided work. All information derived from published literature and other sources is acknowledged and referenced within the text.

Gabriela Pereira

TABLE OF CONTENTS

FIGURES

TABLES

NOMENCLATURE

An alphabetic list of acronyms and abbreviations used in this thesis:

BBE – Best-Before-End

BOM – Bill Of Material

B2B – Business-To-Business

COO – Country Of Origin

DC – Distribution Centre

DEDS – Discrete-Event Dynamic Systems

ERP – Enterprise Resource Planning

FDA – Food And Drug Administration

FIFO – First-In-First-Out

FMCG – Fast Moving Consumer Goods

GA – Genetic Algorithm

GCC – Cooperation Council for the Arab States of the Gulf

HC – Hotel Chocolat Ltd

ICT – Information and Communication Technology

IT – Information Technologies

KPI – Key Performance Indicator

LFL – Like-for-Like

MOQ – Minimum Order Quantity

MRP – Manufacturing Resource Planning

OOS – Out-Of-Stock(s)

RSP – Retail Selling Price

SCOR – Supply Chain Operations Reference

S&OM – Sales and Operations Management

S&OP – Static Sales and Operations Planning

SD – System dynamics

SC – Supply Chain

SCM – Supply Chain Management

SCN – Supply Chain Network

SKU(s) – Stock Keeping Unit(s)

Std – Standard

T – Scheduling period

t – Weekly period breakdown

TSD – Time-Series Decomposition

UK – United Kingdom

US – United States of America

VMI – Vendor Managed Inventory

WIP – Work In Progress

WO – Manufacturing Works-Order

1. INTRODUCTION

This first chapter is an introduction and overview to the content of this work. Attention has been given to the background and explained the key reasons that motivated me to do this research work and also states the main objectives set. The presented work aims to bring a new approach and deepen the knowledge on Business Modelling through the development of a factual and simplified Optimisation Simulation Model that accurately replicates the main business processes, allowing better management decisions for the global supply chain to achieve higher profit levels.

1.1. MOTIVATION AND OBJECTIVES

In a world where the reality of modern business is becoming increasingly complex and demanding, Business Modelling brings a scientific based approach and methods to structure problems, reduce complexity, improve understanding and deliver practical solutions in different business areas, namely: internal processes, operational efficiencies, quality of products and services, as well as future business investments (for example, expansion of product or service portfolios to reach new markets or customers, etc.). The aim in every business is to manage the supply-chain to create value for the customer at an acceptable cost, and managing this has been a key challenge for many operational professionals. The main challenge for any business professional is to have a holistic overview of the business processes and be able to understand the overall business pulleys and levers that guarantee maximum profit whilst maintaining good product availability, interesting product offers, excellent service levels, etc.

The reality shows that most companies have hundreds, if not thousands, of product lines making it almost impossible to analyse individual product and guarantee the optimum decision making and the best application of the marketing instruments. In addition to the number of stock keeping units (SKUs), the reality becomes even more complicated by the fact that many companies have multiple distribution channels: multiple retail stores, manufacturers, suppliers and distributors, and different type of sales channel specificities (for example mail order). What is more, prices, promotions, and local deals offered by specific retail stores adds further complexity so scientific computer-based optimisation tools are absolutely essential. Also, in terms of strategic business changes many professionals initiate those changes without fully understanding the impact on the overall business, so the most important aim for this work is to bring a practical approach to Business Modelling. New information technologies have made massive amounts of data available for such analyses but there are still few statistical and optimisation tools that provide comprehensive, easily implementable, scientifically sound, and powerful results.

Traditional SCs were viewed as a simple sequential system, like a flow line, with raw materials entering at one end and finished goods reaching the consumer at the other end through the following entities: end users, retailers, distribution, manufacturing, and suppliers (Chan & Chan, 2005). Each entity in such a system has little or even no information receives from the others, except the immediate predecessor. Nowadays, it is demanded that SCs should response quickly and collaboratively, therefore, they should be dynamic, flexible, responsive, and even re-configurable to suit for the fast changing environment. In most of the reviewed literature, authors refer to the supply chain as a whole and refer to the overall business benefits, but their area of research is mostly focused on a specific area or subject within a supply chain scenario. Furthermore, even the scientific work which considers global SCs, is not substantiated with real data as the different elements of the supply chain belong to different companies (therefore hard to get that information).

From the literature review of numerous scientific articles about Business Modelling, Supply Chain Modelling and Optimisation tools (presented in greater detail in Chapter 2), but it becomes clear that even though frequently mentioned the benefits and potential of the modelling (analytical approach), it is difficult to find a pragmatic modelling/ optimisation/ simulation approach of Business Modelling. It seems that practical modelling techniques are being applied mostly by private business and consultancy companies.

Most of the reported research was working on the localised problems instead of integrated problems. In other words, only few researchers view SC is an integrated network and hence the SC problem should be considered as a whole instead of localised decision-making problem only. Multi-agent system approach is proposed for solving problems in a supply chain network (Chan & Chan, 2005). Recently, Agent Based Modelling has become one of the favoured approaches of Supply Chain Modelling and Simulation for supply chain simulation and optimisation.

Gjerdrum et al. (2001) applied the multi-agent modelling techniques to simulate and control a simple demand driven supply chain network system. The approach combined optimization with agent-based simulation to access the performance of the supply chain. The tactical decision making and control policy was determined by the agent system while the scheduling problem was solved by a numerical optimization program. Policies such as reorder point, reorder quantity and lead time are tested using this approach to determine the appropriate policy to reduce operating cost while maintaining a high level of customer order fulfilment. Akanle & Zhang (2008) proposed a methodology for optimising supply-chain configurations to cope with customer demand over a period of time; a multi-agent system is used to model resource options available in a supply chain as well as dynamic changes taking place at the resources and their operational environment. Labarthe et al. (2007) propose an agent modelling framework for the modelling and simulation of such Supply Chains to facilitate their management.

The agent based capabilities combine to enable a unique set of functionalities associated with multi-agent systems that supports autonomous decision making, procedurally-correct behaviours and automatic performance of knowledge work along the supply chain. However, there are factors which drive the primary agent supply chain limitations. Even though most agent applications are implemented using object-oriented techniques and expert system development methods, for which considerable guidance and expertise exists, the autonomous, distributed, collaborative nature of multi-agent systems in the supply chain presents design challenges not encountered in most applications that comprise this experience base. Following these authors footsteps, agent-based modelling was also the first approach used to build the HC Supply Chain model, but this methodology proved to be harder to add further layers of complexity and rules, so that approach was replaced with a time-series method.

This work presents a comprehensive (Supply Chain) Business Modelling building approach, which reflects with great deal of accuracy of the business reality. The operational and financial benefits of various levels of supply chain integration are highlighted. Special emphasis is given to integration of SC Financial Flows because it is an interesting measure of how the entire supply chain is performing, against individual department or function performance. The integration of SC financial flows is also becoming a common topic in literature; (Bowersox, 1997) suggests that the creation of time and location benefits not only requires sharing the information to allow suitable business agreements with that purpose, but also requires the existence of a suitable environment for financial transactions.

The main objective of this research is the development of a comprehensive approach to a real supply-chain to support not only operational day-to-day decisions, but most importantly strategic long-term decisions, through the analysis of the existing approaches (chapter 3 and 4) and the proposal of new approaches (chapters 5 and 6). The motivation of this work is to build a comprehensive Supply Chain Business Model which:

1. Reflects orders, materials, and financial flows.

2. Highlights the operational issues and financial benefits.
3. Allows analysis of the impact of new business strategies in the overall profit.
4. Allows better management decisions for the global supply chain to achieve higher profit levels.

The supply chain business structure will be analysed to optimise overall effectiveness and profit, and the SC modelling will also enable the analysis of the impact of new business strategies, as mentioned above.

This study covers the overall supply chain from the external suppliers to retail and customers, supported by factual data, including all the business characteristics and constraints. Figure 1 represents a schematic of the overall approach for this work.

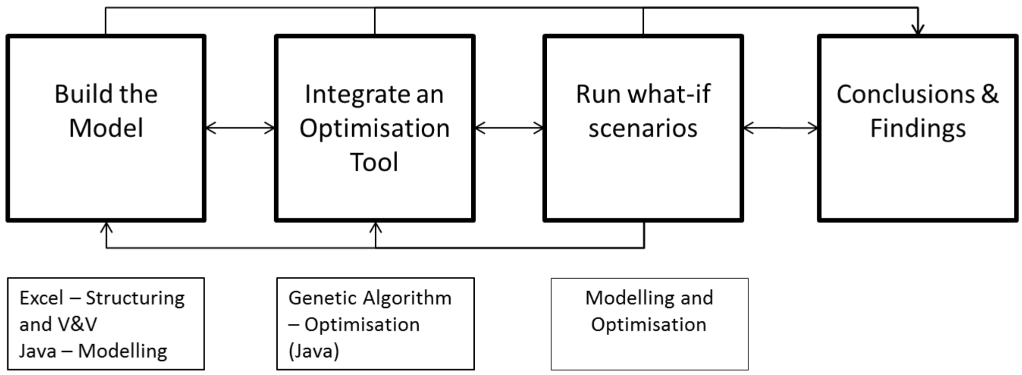

Figure 1 Overall work structure approach.

The general approach will be as follows:

(1) Defining and structuring the supply chain conceptual model and the information needed for the supply chain simulation.

(2) The business model was structured in Excel and finally built in Java and simulations to replicate the behaviour and decision making of the different agents in the supply chains and optimise the parameters to maximise the overall business profit. Excel is used to verify & validate (V&V) the Java simulation results.

(3) Simulate the supply chain behaviour.

(4) Analyse and validate the simulation models and establishing robust optimal solutions using a Genetic Algorithm.

(5) One of the main objectives of this thesis is to prove the applicability of the developed model in different Supply Chain Business scenarios, and determine how the same model reacts to changes to the reality (e.g. perturbations to the real data).

(6) From the analysis of the different scenarios, main findings are underlined and conclusions are presented, and other future work developments highlighted.

Conceptually, the Business Supply Chain Model developed in this work replicates the behaviour and decision-making of the different agents in the supply chain, and an optimisation module determines which parameters maximise the overall business profit for different business scenario. In the optimisation module, a Genetic Algorithm was used to determine the best equation parameters for each individual agent which optimises the overall supply chain profit: minimum stock holding for each individual agent but still securing maximum availability, decreasing BBE date losses, and improving forecasting and ordering accuracy.

Because of the interest of Hotel Chocolat in this project, the utilisation of the research results appears to be guaranteed. The presented business case-study, which is being used to build the supply chain model, is interesting due to its characteristics, namely:

1. It is a **food supply chain** which, in general, is one of the most demanding supply chains, due to the perishable nature of the product, but also due to fluctuations in time, quality and quantity.
2. The majority of the **supply chain elements belong to the same company**, meaning that Hotel Chocolat is a vertically integrated business. Being a manufacturer, distributor and retailer, is very unusual, so this scenario supports the different approaches referred to in the literature, namely:
    - Information sharing - special emphasis given to the information sharing between the different elements in the supply chain, including inventory, sales, demand forecast, order status, product

planning, logistics, production schedule, etc. (usually summarised into three types: product information, customer demand and transaction information, and inventory information);

▫ Collaboration and coordination between the elements in the Supply Chain, to reduce the bullwhip effect;

3. This is a **multi-channel** company: the majority of its sales come from its sales channels comprising high street shops (currently 52 stores), mail order business (which is how the company made its start) and also wholesale partners and franchising (in UK, US and Middle East).
4. While the **Business Commercial Strategy** of this company is set up as a **fashion brand** approach, the **Operational Business Centre** side of the business needs to comply with the challenges and complexity of a **food product**. Similarly to fashion markets, HC has similar challenges to overcome underpinned by the overall company strategy due to an ever changing product offer which guarantees freshness in terms of product range and growing customer interest: shorter life, high volatility, low predictability and high level of impulse purchasing, etc.

The mains points of originality of this work are as follows:

- At any point in time and in the SC network all costs and profits can be easily traced. Most of the other approaches only focus on material flows and information flows, this work focus on cash fluxes.
- Shelf-life of a product is included in supply chain modelling and product losses due to BBE are captured and quantified.
- Optimisation methodology runs parallel to the developed simulation tool, and optimisation parameters change accordingly to accommodate to a new business scenario.
- While most approaches focus on optimising the individual agents' profits (e.g. manufacturing efficiency or the transporter efficiency), for the first time the approach was optimising the overall business profit.

- Due to the fact that the company is vertically integrated, there was access to data in all points of the supply chain and that data was included in model simulation. Normally not possible as the SC agents belong to different companies.
- The impact of sampling is quantified in terms of business profit and impact on the business.
- A supplier scenario evaluation is presented as a tool to choose the best supplier option.

1.2. Thesis Structure

This first chapter is an introduction to the content of this work. Attention has been given to the background and explained the key reasons that motivated this research work and also states the objectives set and the main contributions achieved. This thesis is organised as follows:

- Chapter 2 is a compilation of relevant literature on Business Modelling and Supply Chain Management. Concepts of supply chain and supply chain management are presented and the competitive advantages and importance of linking supply chain to overall business strategy is discussed. A detailed overview of Supply Chain Business Modelling and Supply Chain Simulation is presented and the benefits of these approaches discussed. Special focus is given to Food Supply Chains and the added complexity described. This chapter finishes showing an interesting approach to breaking down the individual elements (agents) in the supply chain for modelling and simulation purposes, which is an introduction to chapter 3.
- Chapter 3 describes in great detail the business case and all the agents that constitute the Supply Chain Network – these agents will form the structural base of the model to be built. The modelling concepts, the model details, structure, considerations, and assumptions are explained. Furthermore the equations that enabled the implementation of the model in Java are presented and discussed.

The first simulations were executed to validate and verify the model assumptions by comparison with published literature, and further simulations were executed with product shelf life considerations.

- In Chapter 4 a Genetic Algorithm approach is used to determine the different equation optimisation parameters (α, β, θ, q) for each individual agent that optimise the overall business profit: through minimum stock holding possible for each individual agent and still guaranteeing full availability, reduced Best-Before-End date (BBE) losses, forecasting and ordering accuracy. In this chapter, the implementation of the genetic algorithm in Java and the cost equations are presented, as well as the fitness function which corresponds to the overall profit equation.
- Chapter 5 demonstrates the applicability of the developed model in different business scenarios. While in previous chapters the focus was on building and optimising the business model which reflects the reality of the HC supply chain, in this chapter the same SC business model is subjected to different case-scenarios, namely: costs in being out-of-stock vs. costs of product write-offs; decisions on stock locations; model suitability for a seasonal only product; impact of marketing sampling campaigns and price promotions in the Supply Chain; the impact of international retail expansions in the overall business.
- Chapter 6 follows the same approach as the previous chapter but the focus is on the impact the raw material suppliers have on the supply chain, therefore there is an extension to the Supply chain network created in the previous chapters. This case-study becomes extremely interesting as the shortage of one component affects the supply chain to an almost chaotic state and the impact of this shortage is quantified in terms of lost profit.
- Chapter 7 concludes all the work, research contributions and presents suggestions for future research.

The appendix chapter provides the java language program used to build the Business Model as well as methodology details used in different chapters.

2. LITERATURE REVIEW

In a competitive global market, companies are continuously searching for new strategies and business models to improve their performance and making sure that they are as efficient as they can be in order to remain more competitive in the ever-evolving global political and economic conjectures (Shen & Norrie, 1998). Business modelling is gaining increasing importance in the rapid development of new economic models resulted of global operations and realities. The ability to utilize advanced computing technology to model, analyse and simulate various aspects of ever-changing businesses has made an important impact on the way nowadays businesses are structured and run. With the current global trading reality, it has become important that all businesses carefully validate their business objectives, requirements, and strategies through a careful process of formal business modelling. It is important for effective enterprise decision making to have clear, concise business models that allow the extraction of critical value from business processes and specify the rules to be globally enforced (Holsapple et al., 2001).

2.1. Business Modelling

The business model concept is becoming increasingly popular within Information Systems, management and strategy literature. It is used within many fields of research, including both traditional strategy theory and in the emergent body of literature on e-business. However, the concept is often used independently from theory, meaning model components and their interrelations are relatively obscure (Hedman & Kalling, 2003). These authors propose a generic business model that includes the following causally related components, starting at the product market level: (1) customers, (2) competitors (3) offering, (4) activities and organisation, (5) resources, (6) supply of factor and production inputs and (7) process component, to cover the dynamics of the business model over time and the cognitive and cultural constraints that managers have to cope with. Osterwalder & Pigneur (2002) define business model as “describes the rationale of how an organisation creates, delivers and captures value”. They define nine blocks to build a business model namely: customers segments (the organisation serves one or more customer segments), value propositions (satisfying customers’ needs with value propositions), channels (the value propositions are delivered to the customers through communication, distribution and sales channels), customers relationships (to establish and maintain with each customer segment), revenue streams (results from value propositions successfully offered to customers), keys resources (assets required to offer and deliver the described elements), key partnerships activities (some activities are outsourced and resources acquired outside the enterprise), cost structure (the business model elements result in the cost structure). A Business Model is defined, by Fox et al. (2000), as a representation, both definition and description, of the structure, processes, resource and information of an identifiable business or other organisational system. The goal of a business model is to: achieve model-driven enterprise design and operation; provide an object library that is a shareable, reusable representation of supply chain information and knowledge; define the objects in a precise manner so that it is consistently applied across domains and interpreted by users; support supply chain tasks by enabling the answering of questions that are not explicitly represented in the model; support model visualisation that is intuitive, simple and consistent.

Martínez-Olvera (2009) presents an interesting literature review by several authors who worked in the past, in the area of business models.

Business Modelling refers to the representation of a system that can be studied in order to better understand the behaviour of the actual system itself and to make predictions about the future. It involves developing a model of a system and carrying out experiments on it, through:

- The construction of a symbolic model which describes system operations.
- Dividing the system into smaller components and combining them in their natural and logical order; analysing the effect of their (component) interactions on one another.
- Studying various specific alternatives with reference to performance of the model and choosing the best one.

The main objective of developing and analysing a business process model is to determine which elements in the process are the value generators or that bring value to the overall business network. A business model shows how business elements are related to each other and how they interact, so the main objective is to provide a clear picture of how the business currently operates and to determine future modifications in order to improve current processes.

There are several reasons for doing Business Modelling, namely (Trkman et al., 2007): to facilitate communication about the business; deepen the knowledge of existing business as the models can be used to clearly define the overall organisation or processes; show the structure of an innovated business; experiment with a new business concept; benchmarking with other companies; identify potential business opportunities; creating suitable information systems, acting as a basis for engineering requirements when a particular information system is being designed; improving the current business structure and operation as it shows a clear picture of the current business state identify the changes required to improve the business; to experiment with new business concepts and to study the implications of changes for the business structure or operation; identifying outsourcing opportunities (the core parts of a business system can be identified and other less important can be delegated to external suppliers). In all of these objectives, the model becomes the basis for the action plan, where the developed model becomes a sketch of a possible development for the business.

2.2. SUPPLY CHAIN BUSINESS MODELLING

Inevitable evolution means that companies need to constantly adjust their business models to changes in their environment. According to Weiss & Amyot (2005), a good approach to evolving business models strikes a balance between capitalising on new opportunities, and preserving investments in existing business processes. Business models are represented in terms of actors and their dependencies, which correspond to value flows between the actors, and those value' flows can subsequently be refined into business process activities. The approach gives business managers a tool for the systematic and incremental evolution of business model alternatives for their organisations and it allows them to model the strategic options available to them, and the conditions for their successful application.

Adopting new business approaches to supply chain integration promises not only incremental improvements in efficiency, but also whole new approaches to conduct business, and even new business models and opportunities not previously possible. SCM should result in the choice of a supply chain scenario, e.g., an internally consistent view on how a supply chain should look in terms of production and distribution processes and their coordination. This is not an easy task, because of a great variety of policies, conflicting objectives, and the inherent uncertainty of the business environment (Alfieri & Brandimarte, 1997).

A typical supply chain involves multiple semi-autonomous parties, who may have several, possibly conflicting, objectives which inhibits the full integration of a logistics chain (Van der Zee & Van der Vorst, 2005). SCM requires, among others, the alignment of partner strategies and interests, high intensity of information sharing, collaborative planning decisions and shared information tools. Even when there is a strong partnership among logistics nodes, in practice, there are potential conflict areas, such as local versus global interests, and a strong reluctance of sharing common information on collaborative planning (Terzi & Cavalieri, 2006; Hambuch, 2004). SCM requires trust and in-depth insight into each other's processes, which is difficult, since the widely followed competitive model suggests that companies will lose power, and therefore the ability to control profits, as suppliers or customers gain knowledge (Barratt & Oliveira, 2001).

An active communication, coordination and cooperation between all members are essential ingredients for the success of supply chain modelling. Even more since the complexity of the system and the solution space in terms of the number of alternative chain scenarios are significant. Basically, the members' efforts should result in credible models, active support in the search for better solutions, and the acceptance of solutions. In order to do so, high demands are set on model transparency and completeness. Transparency refers to the insight into model components and their workings, whereas completeness addresses a full overview of design parameters. This results in the following demands, classified by model elements and their relationships, model dynamics, user interface, and ease of (re)use (Van der Zee & Van der Vorst, 2005):

1. *Model elements and relationships* —supply chains assume an integrated approach to physical transformation, data processing, and decision-making. Especially, the allocation of control policies to specific supply chain members, and relationships, such as hierarchy and coordination, deserve explicit attention as decision variables. This requires the explicit notion of actors, roles, control policies, processes, and flows in the model;

2. *Model dynamics* —control of dynamic effects within the supply chain (e.g. stock levels, lead times and product quality) is an important issue given the many parties involved. Therefore, the logistics of control, e.g. the timing and execution of decision activities, should be explicit. This requires the ability to determine system state, calculate the values of multiple performance indicators at all times, and even more important, allocate performance indicators to the relevant supply chain stages;

3. *User interface* —active and joint participation of the supply chain partners in the simulation study is required: first, as a means to create trust in the solution and among the parties involved, so there is a better chance of acceptance of the outcomes of the study; second, the quality of the solution may be improved. This refers to model correctness as well as the quality of the chain scenario. Clearly, it is almost impossible for the analyst to have all relevant information on chain dynamics. Therefore, the contribution of the problem owner in terms of alternative solutions is vital to the success of the project;

4. *Ease of modelling scenarios* —given the complexity of the supply chain, the large number of conceivable scenarios, and the wishes and requirements of the problem owners, "what-if" analysis should be transparent. This concerns both the choice of building blocks and the time required for tailoring them to the right format for model adoption. Another demand is model reuse, because of the combination of volatile business environments and the major modelling efforts required. Reusable models may help to increase the speed of modelling and analysing alternative scenarios, while reducing costs of decision support.

According to Chan & Chan (2005), modelling methods in SCs can be divided into two major categories: analytical and simulation approaches. Simulation studies are not a dominant tool for SC studies, however, the number of papers that were employed simulation as the modelling method has been increasing in the past few years. This may be due to the fact that the operations and networks in SC are becoming more complicated. The analytical approach is not powerful enough to model complex scenarios as SC issues can be generated in different areas of the business which means multi-criterion decision-making problems -therefore simulation models are more appropriate to solve complex SC problems. Another observation is that often uncertainties are not taken in consideration. Since uncertainty is non-deterministic in nature, it is different to model it in mathematical equations - none of the reported analytical models research has taken uncertainties into consideration. This is one of the major weaknesses in using analytical models to solve SC problem; while uncertainty is a very important ingredient in SC problems, not many researchers take this into consideration. This is, of course, related to the nature of uncertainty –non-deterministic. Simulation would be a better approach as compared with analytical approach to solve SC problems with uncertainties. Furthermore, simulation can handle complex performance measures. It can also be employed to model complicated systems. The most important factor may be that simulation can take uncertainty into account as mentioned previously.

The use of simulation as a tool for analysing and evaluating supply chain strategies gained growing attention in recent years. Many computer-based models developed in the field of supply chain management use system dynamics, an approach for modelling and simulating systems with the help of ordinary differential equations (Parunak et al., 1998).

Fox et al. (2001) define SC modelling as “network of intelligent software modules that together dynamically manage the supply chain. Each module is an expert at its task, thereby optimising its goals, coordinates its decisions with other modules, thereby optimising supply chain wide goals, and quickly responds to changes in cooperation with other modules”. Four technologies are having a significant impact on the achievement of this vision namely the Internet/Web, Intelligent Agents, Constraint Directed Reasoning and Enterprise Models/Ontologies.

By modelling system-wide SC networks, different SC problems, such as production planning, coordination, order distribution, etc., can be integrated and solved simultaneously so that the solution is beneficial to all entities in the network in a long-term base. This can be accomplished by employing distributed, or decentralised, concepts. This paradigm normally deals with the problem through decomposition, aggregation and feedback mechanisms. Entities within the network should have their own preferences and characteristics (Chan & Chan, 2005). Manufacturing and logistics are currently providing the motivation to perform research on multi-agent systems, which is one of the distributed problem-solving techniques. One of the motivations to use a distributed modelling paradigm is the effects of globalisation and increasing outsourcing activities, which lead to the fact that nowadays SCs consist of many independent entities that are usually separated geographically. Each entity has its own operational strategies and business decision-making algorithms, e.g., they are autonomous in nature.

Traditional centralised systems are not able to meet rapidly changing customer requirements, which is another motivation to use a distributed problem-solving method. In addition, business practice has been shifting to task-oriented methods. In this connection, control in an SC network is no longer centralised in nature. Mabert & Venkataramanan (1998) presented some future research directions in solving supply chain problems and one of them is greater development of linked detail planning and control models for integrated firms, with inclusion of all stages of the chain.

When a supply chain is to be implemented successfully we are faced with different problems (Sauer & Appelrath, 2003), namely:

• Because of the elements involved (marketing, distribution, planning, manufacturing, purchasing) typically act independently as there is not a single, integrated plan for the whole organisation, instead there are as many plans as businesses. Additionally, the organisations have their own objectives and these are often conflicting.

• Not only is the efficient materials flow important within the supply chain, more important are the flow of information and the coordination of the interacting business entities of several independent companies.

• There is a massive exchange of information which presupposes that the companies involved can trust each other, which means that the information provided has to be correct, actual and complete.

• Good coordination of business processes, personnel, and information systems among the different entities in the SC is required, which is sometimes difficult as they utilise different operational conventions, specific constraint and objectives, and misaligned incentives.

• As the scope of coordination increases, the notion of centralised coordination breaks down at a point where the system complexity reaches its limit, and some form of decentralised coordination with local complexity through decomposition, aggregation and feedback mechanisms is appropriate.

Different entities in a supply chain operate subject to different sets of constraints and objectives. However, these entities are highly interdependent when it comes to improving performance of the supply chain in terms of objectives such as on-time delivery, quality assurance, and cost minimisation. As a result, the performance of any entity in a supply chain depends on the performance of others, and their willingness and ability to coordinate activities within the supply chain (Swaminathan et al., 1998).

Despite these shortcomings, it is still natural to employ distributed problem-solving tools to describe SC networks due to their complexity and the motivations as discussed above.

2.3. Supply Chain Management

The SCM concept has gained increased attention due to the emergence of flexible manufacturing, implementation of information technologies, increased demand for product variety and degree of customisation, shorter product lifecycles, increased global competition combined with local product customisation and preferences, increased outsourcing, and turbulent market conditions. As previously mentioned, customers' requirements for flexibility, agility, cost efficiency and product variety enforce companies to reconfigure their supply chains and to focus more on collaboration with external partners (Naylor et al., 1999; Mikkola & Skjott-Larsen, 2004). Supply chains need to be agile as they are influenced by external factors, such as money rate changes, delays in the goods deliveries, and failures of production and or transportation (Christopher, 2000). Businesses and supply chains sets are redefining their complementary capabilities and their coordination so as to achieve dynamic equilibrium levels between efficiency and innovation (Rai, 2001). Due to the increasing complexity of supply chain systems (which is the result of changes in customer preferences, the globalisation of the economy and the stringent competition among companies as mentioned before), these decisions are often far from optimum.

A Supply Chain is a "dynamic process and involves the constant flow of information, materials, and funds across multiple functional areas both within and between chain members" (Chopra & Meindl, 2001). Although there are several definitions for Supply Chain (for example, (Lummus & Vokurka, 1999); (Van der Vorst J. G., 2000); Chan & Chan, 2005), it is generally referred as "a network of organisations that are involved through upstream and downstream linkages in the different processes and activities that produce value in the form of products and services in the hands of the ultimate consumer" (Christopher, 1992). The objective of supply chains is to respond more accurately and quickly to customer demand and to keep the size of the inventory necessary to respond to changing customer demands to a minimum possible (Sauer & Appelrath, 2003).

The Council of Supply Chain Management Professionals defines Supply Chain Management as "the planning and management of all activities involved in sourcing and procurement, conversion and all logistics management activities. It also includes coordination and collaboration with channel partners, which can be suppliers, intermediaries, third-party service providers, and customers". In essence, Supply Chain Management integrates supply and demand management within and across companies. Supply Chain Management is an integrating function which primary responsibility is linking major business functions and business processes within and across companies into a cohesive and high-performing Business Model. It includes all of the logistics management activities referred to previously, as well as manufacturing operations, and it drives coordination of processes and activities with and across marketing, sales, product design and development, finance and information technology" (CSCMP , 2012).

There is no procedure to plan a supply chain transformation successfully. Anderson et al. (1997) have defined seven fundamental guidelines to supply chain management, to enhance profitable growth and customer satisfaction. An extensive literature review by Van der Vorst & Beulens (2002) identifies a generic list of SCM redesign strategies to facilitate the redesign process and to attain joint supply chain objectives, as follows:

1. Redesign the roles and processes performed in the supply chain (e.g., reduce the number of parties involved, reallocate roles, and eliminate non value-adding activities).
2. Reduce customer order lead times (e.g., implement ICT systems for information exchange and decision support, increase manufacturing flexibility).
3. Create information transparency (e.g., establish an information exchange infrastructure in the supply chain and exchange demand/supply/inventory or WIP information, standardise product coding).
4. Synchronise logistical processes with consumer demand (e.g., increase execution frequencies of production and delivery processes, decrease lot sizes).
5. Coordinate and simplify logistical decisions in the supply chain (e.g., coordinate lot sizes, eliminate human intervention and introduce product standardisation/ modularisation).

SCM requires, among other things, the alignment of partner strategies and interests, high intensity of information sharing, collaborative planning decisions, and shared information technology tools – this subject will be addressed in further detail in chapter 6. These requirements often represent major obstacles inhibiting the full integration of a supply chain (Van der Zee & Van der Vorst, 2005). Ideally, the agreement of a chain scenario has to be reached based on the evaluation of the consequences of KPIs (Key Performance Indicators) for the supply chain, given the restrictions set by the available resources (Beamon & Chen, 2001). The choice of supply chain performance indicators will typically reflect a balancing of investment, operational costs and customer service in terms of on-time delivery and product quality. Lummus & Vokurka (1999) say that companies who have achieved supply chain integration success report lower investments in inventory, a reduction in the cash flow cycle time, reduced cycle times, lower material acquisition costs, higher employee productivity, increased ability to meet customer requested dates (including short-term increases in demand), and lower logistics costs. To begin managing across the entire supply chain, the following guidelines need to be considered:

1. Link supply chain strategy to overall business strategy, to align supply chain initiatives to business objectives.

2. Identify supply chain goals and develop plans to assure every process is individually capable of meeting supply chain goals.

3. Develop systems to listen to signals of market demand and plan accordingly, including changes in ordering patterns and changes in demand due to customer promotions.

4. Manage the sources of supply by developing partnerships with suppliers to reduce the costs of materials and receive materials as needed.

5. Develop customised logistics networks tailored to each customer segment.

6. Develop a supply chain information systems strategy that can support decision making at all levels of the supply chain and offers a clear view of the flow of products.

7. Adopt cross-functional and cross-business performance measures that link every aspect of the supply chain and include both service and financial measures.

Different authors discuss the advantages of Supply Chain Management. Chang & Makatsoris (2001) and Stevenson (2005) identify the benefits of SCM as follows:

1. Throughput improvements —around better coordination of materials and capacity.
2. Cycle time reduction —considering constraints as well as other alternatives in the supply chain to reduce cycle time.
3. Inventory cost reductions —demand and supply visibility and the ability to know when to buy materials based on the customer demand, logistics, capacity and other materials build together.
4. Optimised transportation —optimising logistics and transport loads.
5. Increase order fill rate —real-time visibility across the supply chain (alternate routings, alternate capacity) enables to increase order fill rate.
6. Analysis of the supply chain management —to predict propagation of disturbances downstream (Forester or bullwhip effect).

7. Increase customer responsiveness and loyalty —understanding the capability to deliver based on availability of materials, capacity and logistics.

The Supply Chain Council (2005) has a holistic approach for the design and analysis of supply chains: Supply Chain Operations Reference, or SCOR – which was aimed to be used as a cross-industry framework for supply-chain management. The SCOR model provides a standard way of viewing a supply chain, a common set of manipulable variables and a set of accepted metrics for understanding the dynamic behaviour of supply chains.

Functional application suites, workflow applications, business process reengineering (BPR) tools, intranets and enterprise resource planning (ERP) applications are illustrations of technology initiatives launched to streamline internal process design and their information sharing practices. New channels for product, multiple alternatives for outsourcing and creation of an entirely virtual organisation within and outside the existing, traditional ways of doing business confound our ability to decide what the best strategy for business improvement is – this reality has forced companies to redefine their business models so as to improve the extended enterprise performance (Barnett & Miller, 2000). The focus has been on improving the extended enterprise transactions. This reality motivates the development of a framework that allows companies to identify the appropriate supply chain strategy depending on product characteristics (Levi & Levi, 2000).

According to Papazoglou et al (2000) "integrated value-chain organisations seek to streamline their processes and improve customer service through greater connectivity between both business processes and key operational systems. An important business objective of strategic alliances with suppliers, channel partners and service providers is to eliminate supply chain discontinuities that produce delays and waste. Enterprises can only become an effective link in a leading value chain by reconceptualising the company as a collection of business operations and processes, by reshaping corporate structures around modern business processes and by making their internal processes align with and support the integrated value chain. This requires that new business models are created to offer a new way to deliver value to customers. New business models are needed and currently emerging". A Business Model is defined as an organisation's core logic for creating value in a sustainable way on a long term basis (Anderson & Lee, 2001).

In system theoretical modelling the models can be either static or dynamic. Systems can also be divided into dynamic and static systems. According to Laurikkala et al. (2002), in a static system the system elements and the system itself do not change over time in relation to the environment, and a dynamic system changes the environment constantly and is also changed by the environment. Another way to define is:

dynamic model —systems where current state is influenced not only by current inputs but also history.

static model —the change in the input has a direct influence in the output and do not include delays or time constants.

If the systems under consideration are dynamic in nature, the descriptive equations are usually differential ones. In modelling complex dynamic systems one first defines the system and its components. Then, the mathematical model is formulated and necessary assumptions are listed. Differential equations describing the model are written and solved for desired output variables. After that the solutions and assumptions are examined, reanalysed or designed. The transfer function of a system or element represents the relationship describing the dynamics of the system.

According to Sterman (2000), system dynamics (SD) is a computer-aided approach for analysing and solving complex problems. System dynamics has been used for understanding and forecasting behaviour of the markets, establishing a structural framework for decision making, challenging industrial assumptions, shortening delivery times, improving customer service quality, and discovering new strategies. System dynamics has been applied to logistics and supply systems (). Multi-stage models for supply chain design and analysis can be divided into the following categories (Beamon, 1998): deterministic analytical models; stochastic analytical models; simulation models; economic models. In deterministic analytical models, the variables are known and specified. In stochastic analytical models, at least one of the variables is unknown and is supposed to follow a particular probability distribution. Complex dynamic models are usually suitable for simulation (Yang et al., 2004).

2.4. Supply Chain Simulation

Supply chain management is a demanding and complicated task due to its broad scope and the strong connectedness of its objects and issues. In order to make theoretical investigations of supply chains feasible and to support decision-making in real world supply chains, simulation models are used (Schieritz & Groβler, 2003). To use simulation for SC analysis is not a hard decision. Discrete event simulation models can handle stochastic behaviour throughout the SC and by doing that, queuing situations and other phenomena dependent upon uncertainty in operation and transportation times can be evaluated (Persson & Olhager, 2002). Supply chain simulation is an effective way to allow industrial partners to analyse the state of their supply chains. In supply chain simulations, the manufacturing activities needed to produce a product, as well as the associated information flows necessary to support the manufacturing process, are evaluated along with the logistical concerns of getting the right materials to the right place (supplier, factory, transportation system, warehouse, etc.) at the right time (Qiao & Riddick, 2004). Planners increasingly turn to simulation models to build confidence and consensus in selecting operational investments to improve or protect market share, revenue, and profit for global businesses (Marquez, 2004). Simulation provides an effective pragmatic approach to detailed analysis and evaluation of supply chain design and management alternatives (Swaminathan et al., 1998). Such experiments can be conducted without being confronted with real world consequences. They make investigations possible and useful, when in the real world situation such experimentation would be too costly or, for ethical reasons, not feasible, or where the decisions and their consequences are too broadly separated in space or in time.

Other reasons for the use of simulations are the possibility to replicate the initial situation, and the opportunity to investigate extreme conditions without risk. Simulation not only shows clearly how the business process is executed but can also correct design errors before they reveal themselves to be harmful.

In practice, simulation is a method that is relatively often used when compared with other quantitative models. Several reasons may explain this popularity: generally no mathematical sophistication is needed, multiple responses are natural in simulation: in SCM, these responses may be the fill rate or service percentage, stocks including work in progress or WIP, sales, etc. (Kleijnen, 2005).

Simulation allows the study of the dynamic characteristics of systems and the analysis of 'what-if' scenarios. Different scenarios are analysed in terms of completion times, costs and critical paths (Franken et al., 2000). Typically, simulation models have been applied only to specific sections of the business, leading to specialised applications (e.g. manufacturing, logistics, business processes, etc.) focused on analysing/ optimising their area of speciality, usually in isolation from other areas. Enterprise simulation has emerged as an attempt to overcome this shortfall by focusing on determining the impact across all the significant elements of the company's internal systems and even inclusion of necessary external elements like suppliers and customers (Tang et al., 2004).

By definition, a simulation model has the following three characteristics (Kleijnen, 2005):

i. It is a quantitative, mathematical, computer model.

ii. It is a dynamic model; i.e., it has at least one equation with at least one variable that refers to at least two different points in time.

iii. This model is not solved by mathematical analysis; instead the time paths of the dependent variables (outputs) are computed - given the initial state of the simulated system, and given the values of the exogenous (input) variables. Simulation does not give a 'closed form' solution. Instead, the simulation analysts experiment with different input values and model structures, to see what happens to the output (sensitivity analysis).

Kleijnen & Smits (2003) distinguishes four simulation types for SCM:

(i) Spreadsheet simulation —used to implement material resource planning (MRPII) and recently vendor managed inventory (VMI);

(ii) System dynamics (SD) —study of fluctuations in the demand by final customers up the SC. Reaction of four links (namely, retailer, wholesaler, distributor, and factory) to deviations between actual and target inventories;

(iii) Discrete-event dynamic systems (DEDS) simulation —DEDS simulation has the following two characteristics: it represents individual events and it incorporates uncertainties. This simulation type is already part of the MRP/ ERP toolbox for quantifying the costs and benefits of strategic and operational policies (ERP);

(iv) Business games —it is relatively easy to simulate technological and economic processes, but it is much more difficult to model human behaviour. A solution is to let managers themselves operate within the simulated 'world', which may consist of a SC and its environment. Such an interactive simulation is called a business or management game. The ability to manage the tactical and operational levels of the supply chain so that the timely dissemination of information, accurate coordination of decisions, and management of actions among people and systems is achieved ultimately determines the efficient, coordinated achievement of enterprise goals.

The recent trends towards out-sourcing of many business processes, and modularisation of supply-chains, result in a growing need for review and revision of traditional supply chain management (SCM) tools. Such tools generally model decisions as being made by one centralised decision maker, rather than as a decentralised negotiation and decision-making process. At the same time, analytical models are limited in their ability to model complex, multi-firm, multi-dimensional relationships. Recently, the emerging field of complexity science gained interest in the modelling and simulation of supply chains leading to a number of agent-based supply chain models (Beetz et al., 2000). Multi-agent system design meshes well with modelling supply chain networks, as it inherently assumes that agents have their own goals, which may be anywhere from pure self-interest to cooperative, thus allowing more freedom of analysis compared to traditional simulation or analytical tools (Mullen et al., 2004).

Vieira & Cesar Junior (2005) defined the following advantages in using simulation in supply chains, namely:

- Simulation assists the understanding of the entire process and characteristics of the supply chain by means of graphic presentations.
- Capacity to capture data for analysis: users may model unexpected events in certain areas and understand the impact of these to the supply chain.
- Can diminish drastically the risk inherent to changes in planning: users may test several alternatives before making the change to planning.
- Investigate the impact of changes due to a greater demand for components of the supply chain.
- Investigate the impact of some innovations within the supply chain, of eliminating an existing infrastructure or adding a new one within the supply chain; of strategic operational changes to the supply chain, such as process, location and use of new facilities, the fusion of two supply chains or the impact of the separation of some components of the supply chain, and of manufacturing products inside the company, and also of the impact of creating new suppliers or subcontracting some processes.
- Investigate relations between suppliers and other components of the supply chains to rationalise the number and size of order lots, using as a basis the total of costs, quality, flexibility and responsibilities;
- Investigate opportunities to diminish the varieties of product components and standardise them throughout the supply chains.

According to the same authors, simulation also brings disadvantages:

- A good simulation model may become expensive and take several months to develop, especially when the data is difficult to obtain.

- Simulation results are often difficult to interpret. Since models attempt to capture the variability of systems, it is common to find difficulties in determining when an observation found during an execution is due to any significant relation in the system or to random processes built in the model.

2.5. Food Supply Chains

Companies increasingly see themselves as part of a supply chain that has to compete against other supply chains (Christopher, 1992). This holds true especially in food supply chains because of shelf life constraints of food products and increased consumer attention for safe, environment/ animal-friendly production methods (Boehlje et al., 1995) and lately, new ethical trading policies. In recent years, Western-European consumers have become more demanding on food attributes such as quality guarantees, integrity, safety, diversity, and associated information services. At the same time, companies in the food industry are acting more and more on a global scale. This is reflected by company size, increasing cross-border food products and international cooperation, and partnerships. Global competition together with the advances in information technology have stimulated industrial partners to pursue a coordinated approach to establish more effective and efficient supply chains (Van der Vorst et al., 2005).

The food industry is becoming an interconnected system with a large variety of complex relationships. Each firm belongs to at least one supply chain in the network, i.e., it usually has multiple suppliers and customers. Supply chains are complex systems due to the presence of multiple (semi)autonomous organisations, functions, and people within a dynamic environment (Kleijnen & Van der Vorst, 2005).

The type of product usually determines the best paradigm to adopt in a supply chain network (Fisher, 1997). The make-and-sell paradigm is well-suited for the production of articles of wide consumption, characterised by a very long life cycle, a stable demand pattern, and low differentiation. On the other hand seasonal products or perishable goods, which become quickly obsolete, have a short life cycle and need a sense-and-respond supply chain paradigm (Blackburn & Scudder, 2009; Donselaar et al, 2006). The great diffusion of the internet and the rapid evolution of web technologies have enabled consumers to enjoy new kinds of purchasing experiences, among which the most remarkable is customisation. Customisable products are characterised by high differentiation, which prevents companies from stocking all the possible product variations in their warehouse. Even if the total demand from the market is relatively stable, the real challenge is to predict the correct product mix. This new category of products needs a solution that combines the benefits from both of the above-mentioned approaches. The new "assemble-to-order" paradigm provides that product components are manufactured and stocked in great quantities, so to exploit economy of scale and decrease production, warehousing and transportation costs. The components are to be assembled into final products on the basis of the orders coming from the consumers, thus improving customer's satisfaction and reducing response time between the receipt of an order and its completion (Yang, 2000). The role of the information coming from the consumer end of a supply chain is more important in this SCN paradigm than in all the others, because in this case it directly affects the manufacturing process of the final product and thus involves several nodes of the business network. As the mass customisation phenomenon is becoming more and more widespread, we have another notable piece of evidence of the fundamental role played by information sharing in supply chain networks (Verdicchio & Colombetti, 2002). Because of the perishable nature of food products, there is a high premium in food production networks on stable collaboration, as it is vital for industrial producers to contract suppliers to guarantee the supply of raw materials in terms of the right volume, quality, place, and time. Furthermore, they coordinate the timing of the supply of goods with suppliers, to match capacity availability.

Van der Vorst (2000) and Van der Vorst & Beulens (2002) define the following characteristics as the cause of fluctuations in time, quality and quantity:

1. Demand —consumer demand fluctuates, due in part to seasonal patterns (e.g. weather) and changes in consumer preferences resulted in requests for different products (larger assortments), which impacts the need for shelf space in retail outlets;
2. Product —perishability of products led to a need for air-conditioned or refrigerated transportation and restricted storage time to prevent quality decay. Furthermore, packaging characteristics (such as materials used and the number of products packed together) influence product handling time;
3. Process —fluctuations in process outcomes and production times, due to variable process yield and scrap-rates. Shelf life constraints for raw materials, intermediates and finished products, and quality decay while products throughout the supply chain. Necessity for lot traceability due to quality and environmental requirements and product responsibility;
4. Supply —food products are characterised by natural variations in quality, seasonal patterns and yield. The supply of goods in the case studies was sometimes hampered by bad weather conditions or traffic congestion, resulting in uncertainty concerning the timing, quantity and quality of supply. As a result there is a chance of product shrinkage and stock-outs in retail outlets when best before dates have passed and/or unacceptable product quality levels are attained;

To these factors is added the growing competitive threat from increasingly powerful global retailers. Fearne et al. (2001) defined a range of responses that FMCG companies consider as mechanisms to combat that competitiveness, namely:

- Innovation —companies must continue to innovate, if they are to provide a food product range that is identifiably different and more desirable than the retailers' own label offer. However, innovation should not be simply product related. Firms should seek to innovate in everything that they do - supply chain management, IT, customer management, demand management and, even in entrepreneurship;
- Cost Leadership —the challenge is to identify what the customer value and is willing to pay for. Then, to focus on the key drivers of cost advantage -viz. economies of scale, and of learning, production techniques,

product design, input costs, capacity utilisation, and managerial/organisational efficiency - and their minimisation, keeping customer requirements firmly in sight;

- Diversify Sectors within Existing Markets —there are other routes to the consumer; for example, there is continued growth in the foodservice sector. In history, food service has not been well-serviced itself;

- Explore New Distribution Channels —new routes to the consumer are emerging which bypass the main line grocery sector. The much-discussed internet shopping, and/or the more generic home delivery sector, offers substantial potential for the future. Even if these routes capture a small share of the overall grocery market, they can generate both significant value and volume;

- Seek to develop long-term partnerships —partnership development provides major companies substantial opportunities to gain “category captain” status, and garner the competitive advantage that such a position can bring. The emergence of supply chain management as a source of competitive advantage and the growing emphasis on the development of strategic supply chain partnerships within the global food industry is a relatively recent phenomenon. It is also one of the challenges which retailers, food manufacturers and all the other intermediaries in the food supply chain have to tackle.

Another response to add to the previous list of responses is sourcing. As industry increasingly recognises the significant contribution of suppliers’ performance and their resource support to buyers’ business strategies, sourcing has accordingly enlarged its scope of analysis to one of supplier relationship management to include buyers’ proactive tasks of supplier development and just-in-time purchasing, as well as the management of buyer-seller interactions for strategic goals.

Enterprises recognise that they need an integration of design, engineering, manufacturing, and distribution systems and processes to succeed. With respect to their supply chains, enterprises are faced with the urgent requirement to develop supply chain strategies that allow them to stay ahead of the competition and to achieve their growth and profitability objectives. In fact, a responsive supply chain is increasingly a requirement just to maintain existing levels of business activity (Kubota et al., 1999; Kleijnen

& Smiths, 2003). In today's global market, managing the entire supply chain becomes a key factor for the successful business.

Based on an extended multi-disciplinary multi-industry literature review (in SCM, logistics management, business process re-engineering, marketing and operational research journals), Van der Vorst & Beulens (2002) identified a generic list of SCM redesign strategies to facilitate the redesign process and attain joint supply chain objectives:

- Redesign the roles and processes performed in the supply chain —reduce the number of parties involved, reallocate roles such as inventory control, and eliminate non-value-adding activities (e.g. stock keeping).
- Reduce lead times —implement information and communication technology (ICT) systems for information exchange and decision support, increase manufacturing flexibility or reallocate facilities.
- Create information transparency —establish an information exchange infrastructure in the supply chain and exchange information on demand/ supply/ inventory or work-in-process (WIP), standardise product coding.
- Synchronise logistical processes with consumer demand —increase execution frequencies of production and delivery processes, decrease lot sizes.
- Coordinate and simplify logistical decisions in the supply chain —coordinate lot sizes, eliminate human interventions, introduce product standardisation and modularisation.

World-class organisations now realise that non-integrated manufacturing processes, non-integrated distribution processes and poor relationships with suppliers and customers are inadequate for their success. They realise the impact of an organisation's plan on the other areas of the supply chain. The impact of an organisation's plan on the whole supply chain is unpredictable before its execution. Simulation permits the evaluation of operating performance prior to the execution of a plan. In the practical application of this concept, the development of a simulation model for supply chain management has become a necessity (Chang & Makatsoris, 2001). Simulation tools are often used for supporting decision-making on supply chain (re)design, building on their inherent modelling flexibility. However, food supply chains set some specific requirements to simulation models (Van der Vorst et al., 2005).

2.6. Supply Chain Modelling Goals and Demands

When designing a model for a supply chain, one starts by defining a real supply chain and its business objects. The design of a supply chain has an essential influence on how a manufacturer of complex products organise and coordinate the stream of innovative products through platform and architectural design strategies *vis-a-vis* sourcing, manufacturing and distribution strategies (Mikkola & Skjott-Larsen, 2004). Then, an input-output analysis is carried out and the conceptual models are formulated (Laurikkala et al., 2002). After that the quantitative phase starts, dealing with more technical problems e.g. development and analysis of mathematical and simulation models and control theory techniques. Figure 2 shows the research structure of supply chain modelling (Laurikkala & Pajarre, 1999).

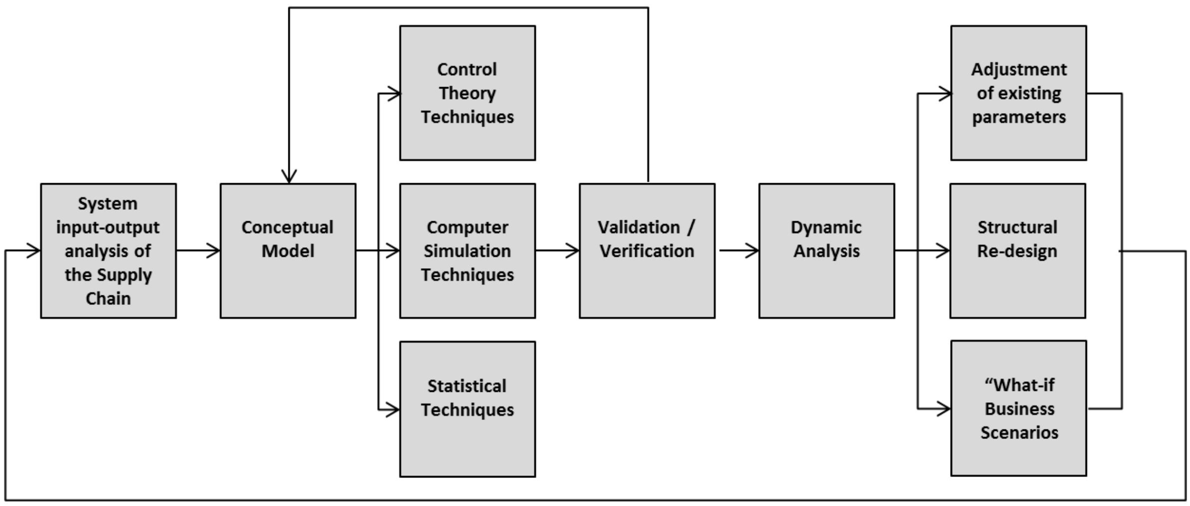

Figure 2 Research Structure for Supply Chain Modelling (*source: Laurikkala & Pajarre, 1999).*

This study will be carried out in the following phases:

1. Problem Structuring —information collection and understanding the business rules and constraint.
2. Process Mapping —use a process mapping techniques to reflect business processes (as-is or to-be). The output of this stage will be used as the basis for a Business Process Model.
3. Prototyping —once constraints are understood, a prototype model will be built. The model will demonstrate the capability of the prototype model.
4. Optimisation —the optimisation model will look to find the best solution under any number of constraints. These can be in a whole host of business areas such as Production Planning, Resource Allocation, Scheduling, Distribution Planning, Dispatching, Cutting problems and Stock Management.
5. Simulation —based on the process flow diagram, the simulation models provide a visual picture of the process and any possible changes overtime. The simulation represents the best method to explore the move from the present business conditions ("as-is") to the most effective "to-be" configurations in the future, but without the cost of implementing them.
6. Data analysis —modelling provides data that is associated with a number of related activities such as data-mining, statistics and forecasting which complement the modelling approach and provide the quantitative rigor.
7. Scenarios —used to illustrate the problems involved in a risk free environment. The role-playing business simulations, generally based on a validated Business Model, have many advantages over other techniques as it can prove catalysts to change, provoking discussion and encouraging assessment.

Recently (2005), Van der Zee & Van der Vorst proposed a modelling framework for Supply Chain Simulation framework in terms of an agent view on supply chain parties and their activities. The main characteristics of the modelling framework are presented:

- Model elements and relationships —their approach is founded on three key concepts: agents, jobs, and flows. Agents model supply chain entities (e.g. planners, production departments, distribution systems, etc.) are autonomous objects that are assigned decision-making intelligence. All chain activities are defined as jobs, including activities related to decision making. Specialisation of agents depends on the type of flow dealt with, that is: material, information, financial. They model the information required for steering and coordinating activities of subordinate agents in terms of the exchange of flow items.
- Model Dynamics —model dynamics is realised by job execution. The dynamic behaviour of the chain processes is captures by modelling the supply chain as a network of agents, jobs, and flows with precedence relationships; the jobs can be triggered by multiple causes and have processing times that depend on the entities processed and process capacity.
- User Interface —adopts basic logistic terminology and makes use of recognisable building blocks. Together with explicit choice and representation of decision variables appealing to the imagination of the problem owners, the visibility of the supply chain processes will in this way be improved.
- Ease of modelling scenarios —choose an object-oriented format for the modelling framework. This is motivated by the advantages it brings for modelling. Apart from the modelling efficiencies in terms of reuse, readability, and maintainability, object-orientation facilitates a natural one-to-one mapping of real world concepts to modelling constructs.

Swaminathan et al. (1998) present a very interesting approach to multi-agent application in Supply Chain Modelling. They classify different elements in the supply chain into two broad categories:

1. Structural elements —modelled as agents, are involved in actual *production* and *transportation* of products, they correspond to agents and control elements correspond to the control policies in the framework.

2. Control elements —facilitate production and transportation of products within the supply chain, helping to coordinate the flow of products in an efficient manner with the use of messages they are classified into Inventory Control, Demand Control, Supply Control, Flow Control and Information Control elements. The choice of appropriate control elements is the objective of problems related to supply chain contracts and supply chain coordination.

   Inventory control elements - they control flow of materials within the supply chain. They are mainly of two types:

(i) Centralised Control: control the inventory at a particular production element while taking into account the inventory levels in the supply chain as a whole. According to this policy, inventory control is applied while considering the total inventory upstream.

(ii) Decentralised Control: control inventory at a particular production element by considering inventory levels at that entity in the supply chain. Typical examples are- order-up to or base stock policy, MRP based ordering (with no information about inventory status at other agents) and (Q, R) or (s, S) policy.

   Demand Control - the demand process within a supply chain is sustained through actual data and forecasts which are modelled as messages in the framework. Orders contain information on - types of products which are being ordered, the number of products that are required, the destination where the product has to be shipped, and the due date of the order. Two important demand control elements are:

(i) Marketing Element: provides a mechanism that can trigger additional demand for products. Increase in demand could be seasonal, random or permanent. This element allows us to capture marketing strategies that might be used in the supply chain;

(ii) Forecast Element: in a "Push" system, forecast evolution plays a very important role because manufacturing decisions are based on demand forecasts, as forecast inaccuracy leads to greater mismatches between products demanded and products produced, and as a result leads to higher inventory costs. In a “Pull” system, products are built-to-order, but forecast accuracy still plays an important role in materials procurement and capacity planning.

Supply Control - dictates terms and condition for delivery of the material once orders have been placed, contain information on the price of the material, length of the contract, volume to be purchased over the contract period, penalty for defaulting, lead-time to receive the product once the final order has been placed, the amount of flexibility that the buyer has in terms of updating demand forecasts over time and types of information control that could be used.

Flow control elements - coordinate flow of products between production and transportation elements. Two types of flow control elements are:

(i) Loading Element: Loading Elements control the manner in which the transportation elements are loaded and unloaded;

(ii) Routing Element: Routing elements control the sequence in which products are delivered by the transportation element;

Information Control - essential for coordination within the supply chain. Two types of information flow are:

(i) Directly Accessible: refers to the instantaneous propagation of information (e.g., information on inventory levels, capacity allocations, machine breakdowns etc. at other production elements or the routes to be taken by other transportation elements);

(ii) Periodic: periodic information updates, in the form of messages, are sent by different production and transportation elements to indicate changes in business strategy, price increase, introduction of new services or features in the products, introduction of new production elements, etc.

A basic set of message classes define the types of interactions that can take place within the network. They recognise three broad categories of message classes, each associated with the simulation of a specific type of flow through the supply chain:

Material flows —relate to delivery of goods by one agent to another. The processing semantics associated with material delivery messages minimally dictate adjustment to inventories but can also trigger messages relevant to other supply chain flows as well as local processing activities. Material delivery messages can be either sent directly by a supplier agent to a consumer agent or may involve an intermediate transportation agent;

Information flows —model exchange of information between supply chain agents. It includes request for goods messages (flow of demand), capacity information, demand-forecast information and supply-related information;

Cash flows —movement of capital through the supply chain. This category includes a payment message sent by customer agents to their supplier upon delivery of goods.

The objective of the current chapter was to present concepts on business modelling, supply chain and supply chain management, as well as the competitive advantages and importance of linking supply chain to overall business modelling strategy. In chapter 3 further concepts on business modelling are presented, and the case-study details which allow the development of the model finally described. This allows the formulation of the business model and the optimisation of scenarios in subsequent chapters (4, 5 and 6).

3. THE BUSINESS MODEL

Depending on the nature of the business, product mix, and overall business strategy, an organisation will define the nature of its supply chain: reactive or functional, or both. Decisions about sourcing and procurement, product lifecycle, product design, replenishment, supply chain network design, etc., are made accordingly with that strategy. Business planning decides when (and where) to manufacture, store and distribute based on demand, SC constraints, profit margins across different products, location, etc., with the objective of maximising profit, as well as increasing the efficiency of their operations (manufacturing and distribution). This could include measures like maximising gross profit margin, return on inventory invested (meaning balancing the cost of inventory at all points in the supply chain with availability to the customer), minimising total operating expenses (transportation, inventory and manufacturing), or maximising gross profit of products distributed through the supply chain, still delivering products to customers at the lowest total cost and highest profit. In this chapter the principles and approach that have allowed building a generic Supply Chain Model are described. The assumptions and the structuring principles that allowed building this SC model are presented.

This model was used to maximise overall business profit for a company with an annual turnover of 60 million pounds, which is at the same time a manufacturer, distributer, wholesaler and retailer, where the majority of products (circa of 95%) are sold through its own channels: e-commerce (in this work frequently referred as mail order) and fifty stores in UK. The 5% of the sales corresponds to business-to-business (B2B) sales to another retail partners, the US franchising and the Middle-East franchising. The overall business structure is an interesting example of a vertically integrated medium-sized company -

where the key-players in are part of the same organisation - therefore there is a visibility in terms of factual data which facilitates the decision making, as all the parties aim to achieve overall company profit rather than individual business benefits. In the context of supply chain research, this is an unusual case-study where manufacturing, distributer and retailer belong to the same company therefore with an overall data access.

The main objective is to bridge between theory and practice with real numerical verified evidence, which is normally hard access in normal circumstances, as the all of the identities belong to different companies. In terms of overall approach, the current work has similarities in terms of approach to the work developed by:

- Truong & Azadivar (2003), but their main focus was on the optimisation module rather than a study on the supply chain itself. Their research develops a hybrid optimisation approach to address the Supply Chain Configuration Design problem. Their approach combines simulation, mixed integer programming and genetic algorithm.
- Daniel & Rajendran (2005) studied the performance of a single-product serial supply chain operating with a base-stock policy and to optimise the inventory (i.e. base stock) levels in the supply chain so as to minimise the total supply chain cost (TSCC), comprising holding and shortage costs at all the installations in the supply chain. A genetic algorithm (GA) is proposed to optimise the base-stock levels with the objective of minimising the sum of holding and shortage costs in the entire supply chain.
- O'Donnell et al. (2009) concluded that forecasting techniques combined with the optimisation process of the GA can be used to reduce the bullwhip effect by determining the optimal ordering policies for members. This method can be combined with any type of time-series forecasting, for example, exponential smoothing or moving average and can be used to forecast daily/ weekly/ monthly time periods. As the majority of companies use some form of forecasting technique, it would be beneficial for companies to combine these techniques with optimisation tools, such as GAs, as this

will further reduce the bullwhip effect and cost across the chain. This approach means that a smaller quantity of recent demand data needs to be retained. Companies typically retain substantial amounts of historical data, which could be used for forecasting purposes and other forms of analysis.

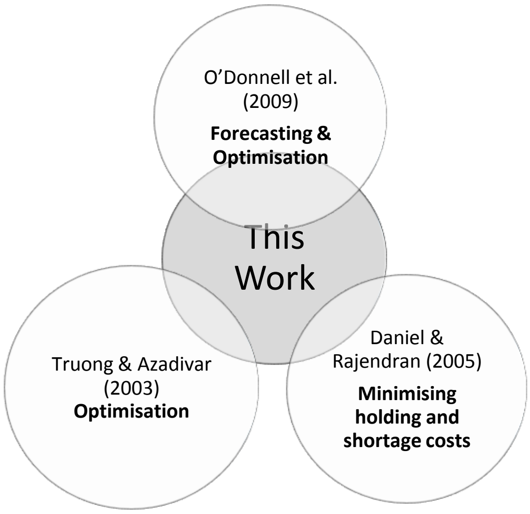

Figure 3 Thesis similarities in terms of approach to other authors' work.

In terms of similarities, the developed work captures all of the above approaches. Using forecasting tools to schedule the manufacturing agent and to determine the right ordering pattern (to guarantee the best stock allocation), the objective of the overall SC activity was to achieve maximum product availability to potentialise sales, thru keeping low the stock levels and stock holding costs, and therefore achieving maximum profit. The optimisation approach used by some of these authors was a Genetic Algorithm (GA), which was also used in this work and which will be described in greater detail in chapter 4. Similarly to Daniel & Rajendran's work, this study comprises a single-product supply chain operating with a base-stock policy and to optimise the inventory (i.e. base stock) levels in the supply chain so as to minimise the total supply chain cost, comprising holding and shortage costs in the supply chain. Despite the fact that the approach of these authors was the basis of this work, and this thesis uses similar methodology as the building blocks for structuring and building the SC model, that was only the starting point to run different business scenarios and quantify the impact on SC and profit resulted of topics like: impact of unpredictable demand and predictable variability, inventory location changes and deliveries pattern changes, global expansion, model suitability for a seasonal product, supplier scenarios, etc. The optimised ordering parameters were determined individually for each one of these SC scenarios and the impact of each scenario in the business profitability determined. The research involved building the Supply Chain Model using Java, with a Repast interface, which mimics the reality in terms of forecasting requirements for both manufacturing and replenishment over a period of three years.

3.1. The Case-Study

Hotel Chocolat, a UK chocolate retailer, was founded over fifteen years ago by two British entrepreneurs. Starting as an online and catalogue-based business, the last years have welcomed a series of expansions, namely fifty retail stores in UK, Ireland, US and Europe. And the Chocolate Tasting Club has attracted well over 100,000 regular members, who enjoy a brand new selection of exciting, artisan chocolates every month. Authentic, wholesome ingredients are the brand ethos that has been fundamental at Hotel Chocolat. A position strengthened by the historic West Indian cocoa plantation, Rabot Estate, where they grow fine flavour cocoa. This step allowed HC to make a unique and exciting connection between the consumer and the cocoa plantation. A set of initiatives underpin a unique ethical policy that currently has three specific focuses; working with cocoa communities in Ghana, the revitalisation of a newly acquired cocoa plantation in St Lucia as well as buying several tonnes of Fair Trade chocolate per year.

"Less Sugar More Cocoa" is the Hotel Chocolat product strategy and the brand mantra. The cheapest ingredient in chocolate recipes is sugar and the most expensive is cocoa. The fact that that many chocolates have been getting sweeter and sweeter over the years underlines this fact. It is tempting for industrial chocolate makers to keep tweaking a little bit more sugar in every year until you reach today's situation where the most popular brands of milk chocolate have less than 20% cocoa. That means that the biggest ingredient by far is sugar, which creates a craving for more and leads to unsatisfying ups and downs. In such a competitive market place, it is the excellence of the all-natural ingredients one of the main factors that differentiates Hotel Chocolat from other chocolate brands.

Besides the excellence of ingredients, the packaging, product concepts and overall product proposal make this product truly aspirational. The product is also supported by great messaging and storytelling, great store displays, excellence of customer service and the overall customer experience which has made Hotel Chocolat the British brand most likely to be recommended to friends and family in the UK. A research, carried out by Bain & Co surveyed 6,000 British shoppers in June 2011 and covered 350 brands across the shopping spectrum, using Net Promoter® Score (NPS), a common measure of customer loyalty. Overall, Hotel Chocolat placed fourth in the Top 10 Most Advocated Brands and was the only British brand to make the list – finishing behind such illustrious names as Kerastase, Mercedes and Apple iPhone. In the chocolate sector Hotel Chocolat finished in first place.

As mentioned previously, there are several reasons that make Hotel Chocolat an interesting case-study. Some of the reasons related to the overall company strategy and mission, organisation set up, unique product positioning and general offer, customer service, etc. While the Business Commercial Strategy of this company is set up as a fashion brand approach, the Operational Business Centre side of the business needs to comply with the challenges and complexity of a food product: perishable nature of the product, but also fluctuations in time, quality and quantity, challenging lead times, product handling requirements, etc. Similarly to fashion markets (Christopher et al., 2004), HC has similar challenges to overcome underpinned by the overall company strategy due to an ever changing product offer which guarantees freshness in terms of product range and growing customer interest, namely:

1. Shorter life cycles —products are designed to represent a period in time or season, but also the latest fashion trends and customers preferences. This also has an impact in terms of the product development cycles, which need to become shorter in order to capture the latest trends and fashions, but also to guarantee that the brand and product offer is ahead of the competition.

2. High volatility —customers' preferences are volatile due to reasons outside the control of fashion retailers (for example the influence of celebrity, media, etc.), so it is up to the retailer to follow consumer trends and sometimes be bullish enough to be a trend setter, in the sense of new product offers, new concepts, new packaging, etc.

3. Low predictability —high volatility naturally decreases the ability to forecast sales. With an ever changing products range this means:

- An increasing difficulty to accurately forecast demand of each stock-keeping unit (SKU), which means greater variability. The commercial team is, understandably, less confident to commit upfront to the totality of the forecasted quantities, which means extra pressure in terms of manufacturing and fulfilment if the product performance is as planned or above the planned quantities. In this scenario the supply chain needs to become agile and more reactive - this is hard to manage internally (when the same company manages its own manufacturing and sales channels) but this issue is exacerbated when as company needs to comply with agreed service levels with third party partners.
- Hard to determine the impact of the new SKUs on the existing proposed range and also on the core product offer (non-seasonal offer).

4. High level of impulse purchasing —consumers place high hedonic value to fashion goods and therefore there is an instant need to purchase it.

3.2. The Supply Chain Network

Modelling a complex business requires the collection of multiple views and each view is a simplified description of a business from a particular perspective. To describe a specific business view, several diagrams are usually used and complemented with textual descriptions (Trkman et al., 2007). For the efficient and effective integration of business processes in the supply chain first of all existing processes have to be fully understood. For any logic system there are a number of variables that describe the state of the system at any point in time. Constructing an appropriate mathematical model (deterministic/ stochastic) which translates a complex, real-life system is a difficult subject in itself. Only once the correct business entities are defined in the physical supply chain, it becomes easier to design a modelling approach for the Supply Chain Model.

The Hotel Chocolat Supply Chain comprises the procurement and transport of raw materials (ingredients and packaging) from vendors (from all over the world depending on the type of component) to the unique manufacturing site (UK based), where they are transformed into the final product. Procurement activities are undertaken in tandem with the scheduling of various production processes (continuous rather than discrete) within the processing plant. Finished products are then stored in the distribution centre, where consolidation takes place prior to distribution to retail stores, corporate customers and mail order customers. Recently, with the St Lucia cocoa, the supply chain has become more complex as it includes the cocoa plantation, growing crops, processing, production, storage and distribution. Hotel Chocolat is a multi-channel company, as customers can purchase HC products on e-commerce, in HC owned retail shops or in the stores from the wholesale partner in UK.

The Hotel Chocolat Supply Chain is presented in Figure 4. This picture illustrates the basic structure of the supply chain, which combines the ‘push’ of value-added material flow, from the stock downstream right through to retail, with the ‘pull’ of information from the final consumer upstream right through to the production of raw materials.

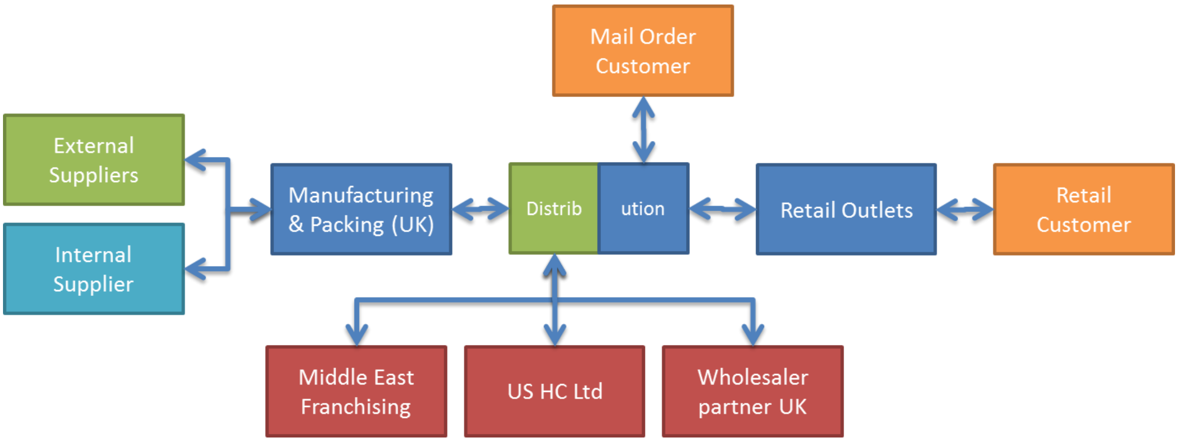

Figure 4 Hotel Chocolat Supply Chain Network.

The agent types presented in the Hotel Chocolat Supply Chain network are as follows:

1. Direct Customers (orange) —corresponds to mail order customers and retail customers —HC owned.
2. Indirect Customers (red) —business partners selling the HC product, working in term of approach to franchising, therefore the customers demand is indirect —independent companies to HC.
3. Fulfilment (darker blue) —this corresponds to the production, storage, distribution and selling agents —HC owned with the exception of the mail order distribution.
4. Suppliers (green and lighter blue) —corresponds to the suppliers that feed both manufacturing and the distribution agent —these are normally external suppliers, with the exception of the St Lucia Cocoa production (internal supply). This cocoa production with origin in St Lucia is only 1-2% of the total HC cocoa consumption due to its low yield as it is only a small island; the majority of the cocoa comes from West Africa.

Each one of these agents is described in greater detail below.

• **Direct Customers** —generates product demand. For this modelling two different types of customers (agents) are considered:

Retail Customer Agent —this agent corresponds to customers that generate demand to the retailer agent for a mix of products. In general the demand corresponds to the shopping patterns on the retail stores owned by HC.

Mail Order Customer Agent —this agent corresponds to customers that generate demand to the distribution centre through e-commerce or online shopping (www.hotelchocolat.co.uk), or through a catalogue (direct mailing) and through the Customer Service team. The CS's objective is not only to deal with any customers' queries and issues, as well as selling product.

Note that the product Retail Selling Price (RSP) is the same in all sales channels whatever they are owned or not by HC (price policy). Some international factors, such as exchange rates, taxes, and duties are not fully described by the existing model.

The reasons why two types of customers were considered are as follows (rather than considering an overall agent): distinct patterns of demand, different types of shelf life requirements and lead times, different fulfilment methods, etc. Furthermore, when a product is out-of-stock (OOS) in mail order, that does not mean that those sales are lost (there is enough stock to fulfil the existing orders, just not the future orders, which can be produced by the time that these need to be despatched); separating these two agents allows adding the complexity of the model and run as-if scenarios in a later stage. None of the mail order and catalogue orders is fulfilled at store level (store-based fulfilment), as the informatics and process systems in place cannot cope with this level of complexity, whereas other multi-channel retailers offer the service of ordering online and collecting in store.

• **Retail(er) Agent** —the different HC retail shops which stocks all (or the majority) of product lines and operate under an inventory control policy (in average four weeks' cover for each product). The state of the retailer agents is determined by the inventories associated with each of the products and the outstanding orders from the retail customer. Marketing elements (e.g. multi-buy offers, purchase discounts, product sampling, etc.) are used, when possible, to control demand generated by customers. The main focus is minimising stock-outs, so a (R, s, nQ) ordering policy, with a review period *R* between orders (normally a week), a dynamic reorder point *s* and orders the multiple *n* of a fixed case size. There are generally three hierarchical levels of retail inventory policy establish by the merchandising team for each store: assortment (deciding which products should be stocked), allocation (how much shelf space to give each product in the assortment), and replenishment (when and how much to reorder), which are determined beforehand depending on store grading (area of the store, turnover and store demographics); this function is normally managed by the merchandising team which sits under the commercial team.

• **Wholesale UK Agent** —this agent is a UK high-street retailer but is a company external to HC. This agent's retail sites correspond to department stores, where the HC product range shares the same retail space as other high street chocolate brands, as well as being in direct sales competition with other gift products. Even though the majority of sales comes from retailing, this company has a well-established direct commerce chain, but the product offer is more limited and with higher price points. There is a collaborative seasonal forecasting and weekly orders and fulfilment from their central distribution centre. This agent manages the overall demand and stock inventory of all products for all retail sites, and operates under an inventory control policy: weeks' cover for each product. The main focus is both minimising stock-outs and overstocks (equal importance). There is also an agreement in terms of shelf life from receipt.

• **Middle East Franchising Agent** —corresponds to an external company to HC, but all the retail sites look and operate in a similar way to the UK sites, and follows quite strict brand guidelines. There is a collaborative forecasting and replenishment planning activity between the two companies as well a minimum agreed shelf life from depot. In terms of the material flows, there are monthly despatches (at peak demand times with increased frequency) from the HC distribution centre to the franchising partner's global distribution centre. Once the product arrives at the global warehouse by aeroplane (one week lead time), it is then organised into weekly drops to the different stores in different countries in refrigerated vehicles.

• **US Hotel Chocolat Agent** —this is the Hotel Chocolat presence in the US. Despite being part of the same group, is treated as an external company. In terms of service, it is viewed in similar way to a franchising partner. Similarly to the UK business model, there are two product channels: retailing shops and e-commerce (with an HC-US site www.hotelchocolat.com). In terms of forecasting and replenishment strategy, this is done by the merchandising team in UK in collaboration with the team in US. Depending on the time of the year and stock levels in US, the product is despatched from the distribution centre in UK to the US subcontracted warehouse – this hub deals with the fulfilment of online orders and distribution to the different retail sites. There is also a minimum shelf life policy between the two sites.

• **Distribution Centre** (DC) —is involved in shipping products from the manufacturing plant to the retailer, mail order customer, middle east franchising, US HC, wholesaler agents. There are five direct links to the distribution agent, therefore it is a quite complex agent, as it needs to follow and fulfil different types of product/ distribution rules (for example: BBE dates, minimum order quantities, lead times, etc.).

In the distribution centre final products come in from the manufacturing and supplier plants (depending on type of product). They are unloaded and stored in the storage area.

At HC, the DC also holds the components and WIP that feed the manufacturing plant which is HC owned. Besides all the finished products, the DC also receives the raw materials and components to be assembled by the manufacturing agent who then packs them into finish product. These components are ordered to suppliers and picked to the manufacturing orders as a result of the Works Orders placed by the planning team who, in its turn, fulfils the merchandising team's request for each product. The merchandising team determines the quantity, the date and shelf life required of the product that they need to fulfil the overall product demand. The production planning function corresponds to scheduling the merchandising product orders in the most economic and efficient way in order to take full advantage of the installed manufacturing capacity.

When orders arrive from the subsequent agents, relevant products are removed from the storage area and are sent to the appropriate loading dock where they are loaded and sent to the destination. The product is packed and shipped to the customer if it is available as finished goods inventory, otherwise the order is added to a queue (for the particular product) according to its priority (if the priority of all the orders are same then it is FIFO, first-in-first-out, otherwise it needs to obey to a minimum shelf life requirements). When the product is fulfilled by the distribution centre, the order is removed from the queue and product is packed and shipped to the customer. Orders may be placed for multiple products in which case the processing becomes more complicated. The main focus is reducing the cycle time for the delivery of a customer order, minimising stock-outs and minimising stock losses due to shelf life.

In this work the logistics are not treated as a separate agent to the distribution centre. The distribution agent function is to store product, and fulfil the demand and getting the product to the next agent down on the supply chain. The transport to retail shops and wholesaler is done by HC owned transportation vehicles. In terms of mail order, the fulfilment is done using a parcel logistics company, while the distribution to other continents is a combination of HC vehicle and air logistics.

• **Manufacturing Agent** —the manufacturing plant is where components are assembled into finished product and chocolates are manufactured. HC does not manufacture the chocolate itself (more information below), instead it converts this chocolate into bars and filled chocolates, which are then packed into the final packaging, outered and then sent to the warehouse facility.

The main focus is on optimal procurement of components (particularly common components) and on efficient management of inventory and manufacturing process. Each product has an associated Bill-Of-Materials (BOMs). Manufacturing is based on both a “Pull" and “Push" mechanism (pull, the product is made only when an order from the merchandising team is received for it, and push, the products are built based on demand forecast). The majority of products are built based on demand forecast. The merchandising team uses inventory control elements for managing the inventory, contracts with downstream entities for supply control, flow control elements for loading and unloading products, forecast elements for propagating demand forecasts to the downstream entity and may use information control elements with other entities in the supply chain.

Part of the company strategy is to be a British Chocolatier with its manufacturing premises based in the UK (whereas other retailers to move production to countries with lower labour costs). Despite the higher costs in manufacturing, this results in extensive and complex supply chains, and consequentially, to long lead times for product due to the large geographical distance between sourcing and selling markets, but currently the company size justifies that approach. The benefits of being a manufacturer, are speed-to-market and responsiveness, there are also other factors that are not tangible resulting of that approach which is reflected in customer preference for the brand (made in UK) and being recognised as a British Chocolatier.

As chocolate demand is quite seasonal, both the manufacturing and distribution agents increase the respective throughput by increasing the number of production and picking shifts using temporary staff (circa 3 months/year).

• **External Suppliers** —supplier agents supply parts (e.g. raw materials, chocolate or packaging) to the manufacturing plant. These suppliers are normally categorised as follows:

1. Chocolate manufacture —despite the fact that the majority of chocolate grades used in HC products be bespoke to HC, it is not manufactured by HC. The chocolate grades development (meaning types of cocoa beans, percentage of cocoa solids, percentage of sugar and milk, vanilla, etc.) is steered by HC but it is sourced to one of the worldwide leaders of chocolate manufacturing. These chocolate grades are normally supplied in 25Kg bags of callets or in liquid format using trucks that can hold up to 2 ton of chocolate in liquid form.

2. Ingredients Suppliers —all raw materials necessary to make a chocolate and that complements the chocolate grades, namely: nut paste, fruit concentrates, nuts, cream, natural colours, etc.
3. Packaging Suppliers —suppliers that source all types of packaging items which can be boxes, trays, outers, labels, wraps, foil, bags, ribbon, etc., in order to pack the product (to final packaging or WIP). There is in general a clear company's green approach to packaging: 98% of the packaging used by HC is sourced in UK or Europe and the majority of that packaging is recyclable or FSC[1] (in case of the paper board). Only packaging that needs a lot of handwork is then sourced from the Far East.

For all of these suppliers, business contracts determine the lead-times, flexibility arrangements, and information-sharing, and agreed turn-around times and inventory.

In the first stage these agents are not incorporated in the overall process – it will be added to the overall process when exploring the "what-if scenarios". More information about suppliers to manufacturing will be described in Chapter 6, where the impact of suppliers on the SC will be investigated.

• (**Internal) Supplier** (Plantation) —this supplier corresponds to the cocoa production with origin in St Lucia which is only 1-2% of the total HC cocoa consumption due to its low yield as it is only a small island. This supplier has its own supply chain for production. It will be modelled as a single agent because the parent organisation has no direct control on their internal operations. This supplier supplies parts to the manufacturing plant. Their operation is characterised by the contracts with main company in terms of lead-times, cost-sharing and information-sharing.

[1] The Forest Stewardship Council (FSC) is a non-profit organization that sets certain high standards to make sure that forestry is practiced in an environmentally responsible and socially beneficial manner. If a product is labelled as "FSC Certified," it means that the wood used in the piece and the manufacturer that made it met the requirements of the Forest Stewardship Council.

The focus of supply chain management is explicitly on processes – finding the most effective and efficient way of adding value – with the aim of generating cross-functional solutions to the many complex problems associated with meeting consumer requirements effectively and at minimal cost. Cross-functionality occurs within the organisation (e.g. sales, marketing, logistics, logistics, production planning and production combining to reduce inventory levels whilst maintaining customer service levels) or between organisations (e.g. raw material suppliers) in a way that optimises short-term storage, transport optimisation and processing capacity.

Even though chocolate is not as demanding in terms of temperature control as in other businesses, the company policy means the stock in store and warehouse needs to be kept ideally at a temperature of 16°C. Failing to meet these temperature requirements, the properties of chocolate means that the product starts to lose some of its visual appeal (glossiness), even though it is still perfectly good and safe to be consumed.

3.3. Building the Supply Chain Model

The Hotel Chocolat Supply Chain network was structured and tested using firstly Excel (acting as the base for structuring and implementation in Java) and then compared with the literature, for the Validation & Verification of the model. The final model built in Java with output interface in Repast. In Java, the different agents have been defined (Manufacturing, Distribution, Middle East Franchising, US Hotel Chocolat, Wholesaler, Mail Order Customer, Retailer, Mail Order Customer, Retail Customer), as well as the information, material and cash flows will be identified to address the plan process and the relationship between different processes and the different elements

As mentioned previously, this SC model will be single-product operating with a base-stock policy and to optimise the inventory. This single-product is a core year-around line with well-established demand; due to the nature of the business the demand is variable with clear seasonal demand peaks, so a time-series forecasting method is applied to forecast the manufacturing requirements. At the first stage, best before end (BBE) dates limitations are not considered in the building the model, as Excel cannot cope with this type of data, but further on when the Java model is implemented the stock losses due to BBE are built into the model.

The considered analysis period corresponds to three years (*T*) and a weekly period breakdown (*t*). In later chapters this weekly period breakdown will be also named as *tick*, frequently used in computer science, which in this scenario corresponds to a week. The best solution of the order level at each stage is calculated based on the identical *t*. The optimal scheduling at each period *t* is also used as the baseline for the product allocation periods at the channels and distribution systems. The suitable numbers of allocation units from distribution to the different channels, manufacturing to distribution centre, and intermediate agents to end customers are determined based on this optimal scheduling period, *t*.

To build the model the following assumptions were made:

- This model will use a single-product line operating with a base-stock policy.
- The presented product demand corresponds to a year-round core line product with a well-known sales pattern.
- A periodic weekly review policy is used to keep track of inventory at each stage at all times.
- The model assumes a four-stage inventory for a single product, which consists of manufacturing, distribution, intermediate agents (Middle East Franchising, Wholesaler, US, Mail Order and Retail Outlets) and final customers (Mail Order customers and Retail customers).
- The customers' demand is known and independent of stock availability at each channel.
- None of the scenarios to be presented considers product demand substitution. This subject was researched by Rajaradam & Tang (2001) and Yücela et al. (2009), so in this case no availability means lost sales, which in real life may not be totally correct, but due to lack of specific data is hard to quantify, therefore no stock means lost sales. It is general knowledge that, depending on the product

and on the reason for the purchase in retail stores, the substitution is greater. For example if a customer goes in store to buy a gift then, they have a price or concept in mind, and they search for a product to fit those requirements, so the substitution rate is much higher; in other hand, if the customer goes to buy their favourite product and it is not available, then the product substitution is almost zero.

- The safety stock level of each intermediate is a positive stochastic quantity (q_t).
- The manufacturing agent takes one week to send the product to distribution.
- If an incoming order cannot be completely fulfilled due to a lack of available items in the inventory, that quantity will be transferred to backorders, with the exception of the retailer and mail orders agents where the customer order non fulfilments correspond to lost sales (backorders are always zero). As mentioned, at this stage no product substitution is considered.
- The intermediate agents are supplied products from a single DC.
- At the distribution agent the material flows are split to satisfy the downstream orders from the subsequent agents; this is done through a set of priority rules applied by the distribution agent. Table 1 shows the priority in terms of orders fulfilment in the Distribution Agent, where the highest priority has the smallest number.

Member SC	Distribution Fulfilment Priority (*)	HC owned
Retail Customer	NA	yes
Retailer	5	yes
Mail Order Customer	4	yes
US Hotel Chocolat	3	yes
Wholesaler	2	No
Middle East Franchising	1	No
Distribution	NA	yes
Manufacturing	NA	yes

(*) (1=greatest priority, 5=least priority)

Table 1 Prioritising orders fulfilment in the Distribution Agent.

- The only losses of product in the supply chain are due to Best-before-End (BBE) losses and no other reason.

3.3.1. Components of the Supply Process

Establishing the structure of a model comprises determining the system boundaries, identifying the entities, attributes and activities of the system, and defining the values and the attributes that define the relationships involved in the activities.

The Beer Distribution Game is a well-known and commonly used supply chain analysis tool widely used to illustrate human decision-making and the concepts of supply chain management (Sterman J. , 1989). The traditional game is normally played by four players, representing four agents: a retailer, a wholesaler, a distributor and a manufacturer. Each individual faces a decision-making challenge involving how they manage their current stock inventories. Each participant in the game seeks to minimise their total cost by managing their inventories in the face of uncertain demand. It has been shown that this simple game provides complex and often non-linear dynamics due to feedbacks and time delays. Sterman's ordering principles state that game participants should place sufficient orders to: (i) Satisfy Expected Demand, (ii) Adjust Inventory Levels, (iii) Adjust for Orders Currently in the Supply Line.

In this work we consider a serial single-product distribution system of four levels similar to the one presented by Sterman (1989) and Mosekilde et al. (1991). Figure 5 is a schematic representation of the different flows in each of the ordering agents. The green arrows correspond to material flows from upstream to downstream, while the orange arrows correspond to the information flows in the opposite way to the material flows; for example, for the retailer agent, the upstream would correspond to the distribution agent, while the downstream would correspond to retail customer agent. The red arrow corresponds to the stock losses due to products which BBE dates is no longer suitable to be sold to the customer at full price (BBE dates are less than the minimum shelf-life required, which is in most cases four weeks before the best-before-end date). After this date it triggers a discount promotion to increase the demand and trying to sell-through that stock.

As mentioned in chapter 1, one of the objectives of this this work is not only capture the material and information flows, but also to determine the cash flows between the different agents, so that at any point in the supply chain each activity has a monetary value associated to it. In chapter 4 the costs flows will be explained in greater detail as those costs will be fundamental for the SC optimisation analysis. The materials and information flows will be detailed in this chapter.

Each agent makes ordering decisions based only on locally available information: inventory vs. desired inventory; what is on the pipeline, meaning which orders have been placed to the agent upstream and which shipments are in-transit from the upstream agent; current demand; previous decisions made by the agent. The logic sequence at each stage is as follows:

1. Get items demanded (downstream order list) and received (upstream shipment list).
2. Determine the item to supply to downstream and adjust the inventory —stock, backorders and supply.
3. Determine the new orders and update the orders in the pipeline —new order.
4. Update inventory.
5. Update items in shipment and update pipeline for this stage.

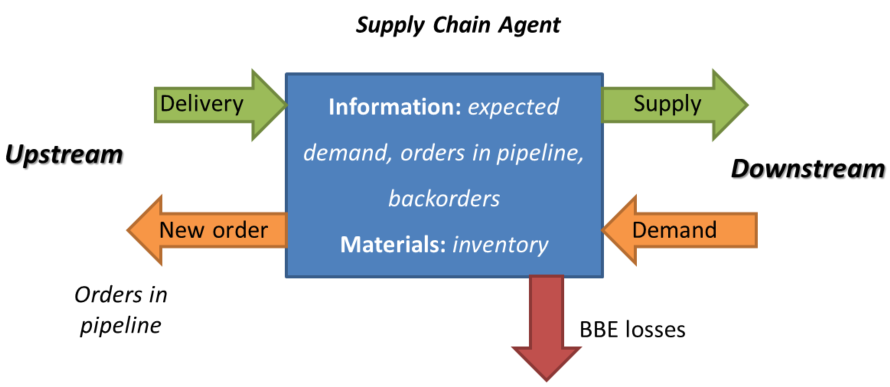

Figure 5 Schematic Representation of the different flows in each of the ordering agents.

3.1.1.1. DEMAND FORECASTS, STOCKS AND SUPPLY

The following equations representation and formalise an intuitive ordering policy based on the principles discussed by (Moseklide et al., 1991). First of all the orders must be non-negative, i.e.:

$$O_t = Max(0, IO_t) \quad (1)$$

where O_t is the new order and IO_t denotes the indicated order rate and the subscript *t* indicates that the value of the variable is considered at time *t*.

With the exception of the Manufacturing agent, forecasts for each SKU are generated by the replenishment system based on each agent needing to adjust their demand projections in each period. The forecasting technique considers the trend as well as the series average ignoring the trend will cause the forecast to always be below (with an increasing trend) or above (with a decreasing trend) actual demand.

At each time weekly period breakdown (*t*), the agents weight the current demand and the expected demand for this period to estimate the demand for the next period:

$$ED_t = \theta . D_{t-1} + (1 - \theta) . ED_{t-1} \quad (2)$$

where ED_t and ED_{t-1} are the expected demand at time *t* and *t-1*, respectively. D_t is the customer's demand at time *t.* θ is the parameter controlling the rate at which expectations are updated (smoothing parameter between 0 and 1).

The demand for chocolate is highly dependent on the weather and has a highly erratic demand profile, so sales and stock levels are monitored frequently. Forecasts are adjusted to account for the latest weather reports, seasonality, national holidays (for example, Christmas, New Year, Easter), special occasions (such as Mothers' Day, Valentines' Day, Chinese New Year, etc.) and events (particularly sporting events such as the Football World Cup or Olympics), as they change people's shopping habits in terms of the size of their purchase and/or the mix of products. These forecasts can also change depending on promotions and marketing activities such as the product being in the power bay, multi-offer promotions and media activity. Finally the local store manager has the opportunity to indicate that a local event is likely to increase the demand for specific product categories.

The order rate to the distribution centre for all the ordering agents are defined by the equation:

$$IO_t = ED_t + \alpha.[q - (Stc_t - BO_t) - \beta.OP_t] \quad (3)$$

where IO_t is the indicated order, BO_t are the backorders, Stc_t is the stock, OP_t are the orders in pipeline at time t. α is the moving average constant (parameter between 0 and 1), β is the relative weight attached to the pipeline vs. stock discrepancies from desired levels (between 0 and 1), q is the measure of the desired inventory relative to the desires supply line (always ≥0).

$$OP_t = OP_{t-1} - RS_{t-1} + O_{t-1} \quad (4)$$

where OP_{t-1} are the orders in the pipeline at time *t-1*, RS_{t-1} are the shipment received from the distribution agent, O_{t-1} is the new order to the distribution at period *t-1*.

To determine what is supplied to fulfil the existing customer demand at any agent the following equation was used:

$$S_t = Min[(D_t + BO_t), (Stc_{t-1} + RS_t)] \quad (5)$$

where Stc_{t-1} is the stock at *t-1*, RS_t is the shipment received from the distribution agent for the period *t.*

For both the Retail Customer and Mail Order Customer agents, it is considered that their demand is fulfilled if there is stock in the retailer and in the distribution centre respectively, with no time delays (meaning that all the orders are fulfilled within the week). For the retailer, if stock is not available to fulfil customer demand this means sales losses - as product substitution is not considered, or delays in fulfilling those orders (e.g., the customer going back to the store to full demand). Furthermore, in real circumstances, the lack of repeated availability means that the customers will not return back to the store to look for their favourite product(s), which will reflect in future sales loss but that loss is hard to quantify.

In this scenario for the retailer only agent, no stock means sales loss so there are no backorders, so BO_t (retailer) equals to zero, so the quantity to supply is given by the following equation:

$$S_t = Min\ [D_t, (Stc_{t-1} + RS_t)] \tag{6}$$

The backorders for all agents is determined by the following equation. This equation also corresponds to the lost sales for the mail order customer and retail customer agents:

$$BO_t = BO_{t-1} + D_{t-1} - S_{t-1} \tag{7}$$

where BO_{t-1} are the backorders quantities at *t-1.*

The stock at any point for each tick *t* is determined by the following equation:

$$Stc_t = Max[0, Stc_{t-1} - S_t - BBE\_L_t + RS_t] \tag{8}$$

Where Stc_{t-1} is the stock at period *t* -1, S_t is the quantity supplied at period *t,* BBE_L_t is stock losses when the product reaches its best-before-end date, RS_t corresponds to received shipment at period *t.*

The distribution agent has the same equations as the previous equations but the tasks of Receive Orders, Supply and Backorders are split into five corresponding to subsequent elements (Retailer Agent, Mail Order Customer Agent, US Hotel Chocolat Agent, Wholesaler Agent, and Middle East Franchising Agent).

From the analysis of the results of Moseklide et al. (1991), when a model starts with zero stock, the model takes much longer to achieve to a normal equilibrium, as the system has to work much harder to get the necessary stock levels (Figure 6).

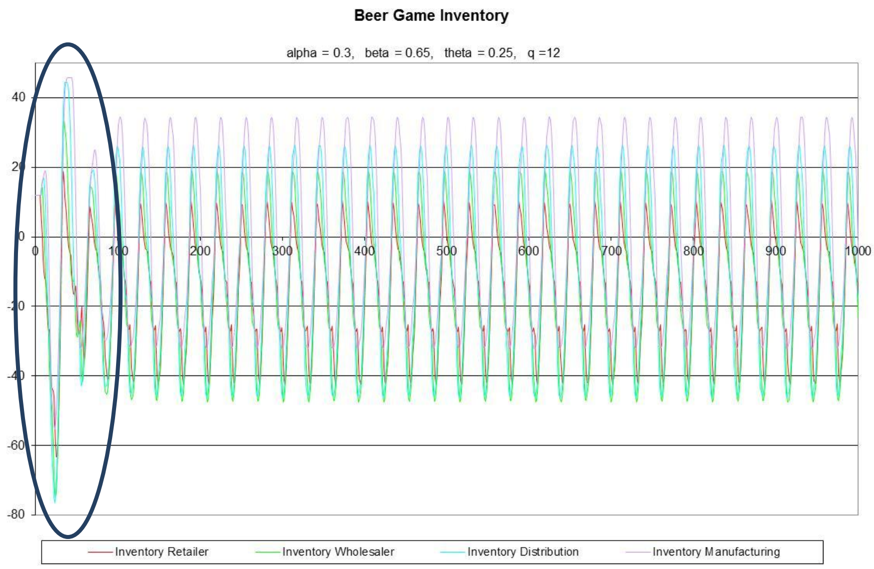

Figure 6 Beer Game Inventory Plot (*Source*: Moseklide et al., 1991).

Considering this fact, the following stocks were considered for each one of the SC agents, which corresponds to roughly four weeks of demand, at *t* equal to three. The reason why it is considered t = 3 as the starting point is that the manufacturing agent starts to produce at t = 1 (2 weeks before the stock needs to be available in the warehouse to fulfil the downstream orders) and to avoid the time lag that the system naturally needs to reach a position where the warehouse can start to fulfil orders.

```
inventoryRetailer.set(3, 2500);
inventoryMailOrder.set(3, 1350);
inventoryUS.set(3, 60);
inventoryWholesale.set(3, 600);
inventoryMiddleEast.set(3, 90);
```

3.1.1.2. Manufacturing Forecasting & Planning

For the Manufacturing Agent there are specific equations for both the stock levels and manufacturing scheduling. The manufactured quantities are determined using a Time-Series Decomposition forecast methodology (T_f), and the manufacturing happens one week before the stock is required in the warehouse.

$$M_t = Max\ (0, F_{t+1}) \tag{9}$$

The previous equation shows that manufacturing starts to produce a week earlier the forecasted required quantity, so that the stock is available to satisfy the initial orders:

$$Stc_t = Max(0, M_{t-1}) \tag{10}$$

where M_t and M_{t-1} are the manufactured quantities at time t and time *t-1* respectively, and F_{t+1} is the forecasted quantity to be produced by the manufacturer at period *t*+1.

The forecasted quantity (F_t) is determined using the Time-Series Decomposition (TSD) method. The procedure which decomposes a series into a seasonal component is a combined trend and cycle component, and an "error" component. The procedure is an implementation of the ratio-to-moving-average method. It takes the time series of sales data and breaks it into trend (*T*), seasonal (*S*), cyclical (*C*), and irregular (*I*) components. Each of these components is modelled separately and then combined to make the forecast.

$$X_t = f\ (S_t, T_t, C_t, R_t)$$

where X represents the level of sales, S_t the seasonal component at period *t,* T_t the trend component at time *t*, C_t the cyclical component at time *t*, R_t the random component at time *t*

The most common time series decomposition model is multiplicative, called the ratio of actual to moving averages, such as:

$$X_t = S_t\ .T_t\ .C_t\ .R_t$$

Multiplicative models are more prevalent with economic series since most seasonal economic series have seasonal variations which increase with the level of the series.

Theoretically the process of deseasonalising the data has useful results as the de-seasonalised data allows seeing better the underlying pattern in the data, it provides measures of the extent of seasonality in the form of seasonal indexes and it provides us with a tool in projecting what one quarter's (or month's) observation may portend for the entire year. The figure below corresponds to the time plot of number of units sold of a specific year around product for each month during a three year period (blue line) and the deseasonalised manufacturing (red line).

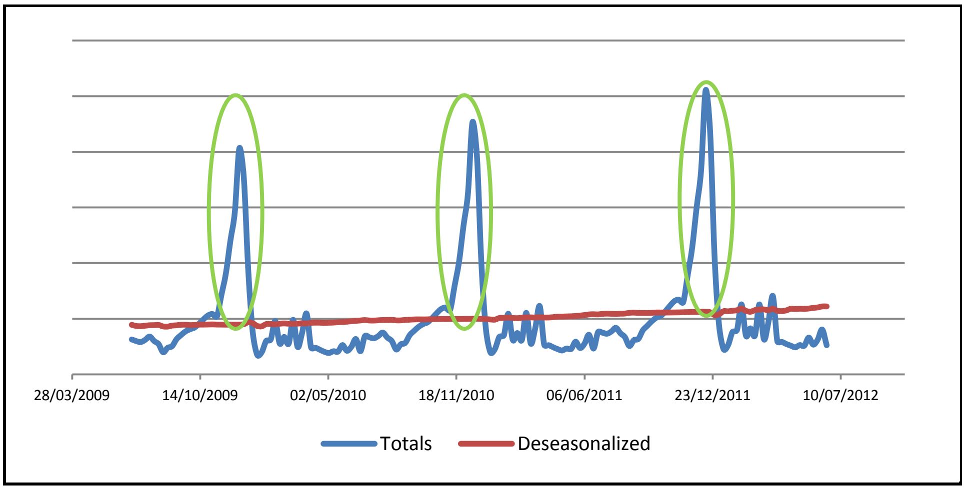

Figure 7 Time plot of number of units sold of a specific year around product for each month.

3.1.1.3. TRANSPORT LEAD-TIMES

The orders generated take one week to arrive to retail stores as per a weekly delivery cycle, but in reality in peak times (e.g. Easter and Christmas) further "top-up" deliveries may be scheduled, depending on the demand, local delivery restrictions (some stores are in residential areas and large trucks are not allowed to go there and/or delivery times may be restricted at certain periods of the day), number of deliveries per day (which may change during the week) and preferred delivery patterns. In the mail order sub-system the demand can be satisfied in the following 24 hours, while the wholesale partner, depending on time of the year and volume of delivery, takes usually 1 to 2 weeks to arrive in store. The Middle East franchising agent takes a week to reach the stores, as it is air shipped and for the US, depending on the volumes of the delivery, products can be air shipped or sea freight, so a two weeks lead time is considered.

For the Retailer, Mail Order, Middle East Franchising and Distribution Agent there is one week lead time, between the supply up the chain and receiving the shipment:

$$RS_t = S_{t-1}$$

for the Wholesaler UK Agent and US Hotel Chocolat Agent, the lead time is two weeks instead:

$$RS_t = S_{t-2}$$

3.1.1.4. PRODUCT TYPE AND SHELF LIFE CODE

Hotel Chocolat has two types of product types that divide the same merchandising space in well-defined sections: core product lines (basic or classic products offered at all times if not in all stores, in most of the stores and online) and seasonal lines (products that are currently in fashion or are available in a specific season, therefore the availability period is quite limited). All products are perishable (chocolate) and the shelf life can be between 4 and 12 months, which impacts on both the physical replenishment process and how the replenishment order is generated. As these products have a long shelf life and considerable demand volumes, a stock of product is kept in store and the depth of that stock will vary depending on the projected sales curve.

3.3.2. VALIDATION & VERIFICATION

With the objective of validating the presented balance equations and verify if the implementation of the coding in Java has been done correctly, the same equations were used to simulate the same case-scenario presented in the Beer Game (Moseklide et al., 1991). Considering the following equation parameters: α = 0.3, β = 0.15, θ = 0, q = 17, the results match the results presented by Moseklide (Figure 8), therefore the equations have been validated and verified.

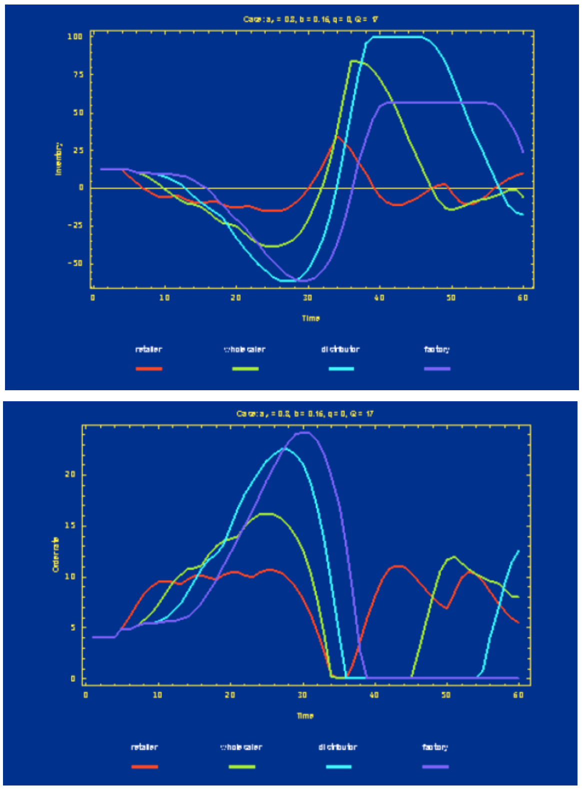

Figure 8 Beer Game Orders (left) and Inventory Graphs for α = 0.3, β = 0.15, θ = 0, q = 17.

As mentioned previously, the objective of the simulation of the Hotel Chocolat Supply Chain in Excel is to structure the model to be modelled in Java, well as support the principles and equations used to build the model as well as validating and verifying the model built and comparing it with literature results. In the first stage, the agents that generate demand (Retail Customer, Mail Order Customer, US Hotel Chocolat, Wholesaler, Middle East Franchising) were considered constant with the same demand before *t* = 5 (5 seconds) and different types of demand from tick 5, as per Table 2:

Order Rate	Tick < 5	Tick > = 5
RetailCustomerAgent.java	4	8
MailOrderCustomerAgent.java	4	6
USHotelChocolatAgent.java	4	10
WholesalerAgent.java	4	15
MiddleEastFranchisingAgent.java	4	20

Table 2 Orders values for the agents generating the demand.

The Figure 9 corresponds the orders and stock for each agent in Excel. At week 147, both orders and shipments reach stability. If demand does not change, the deterministic system will continue forever in equilibrium: orders received equals the orders sent at every stage, and the customer demand is always satisfied. The same changes were incorporated in Java and the results are similar to Excel, therefore the implementation in Java has been successful.

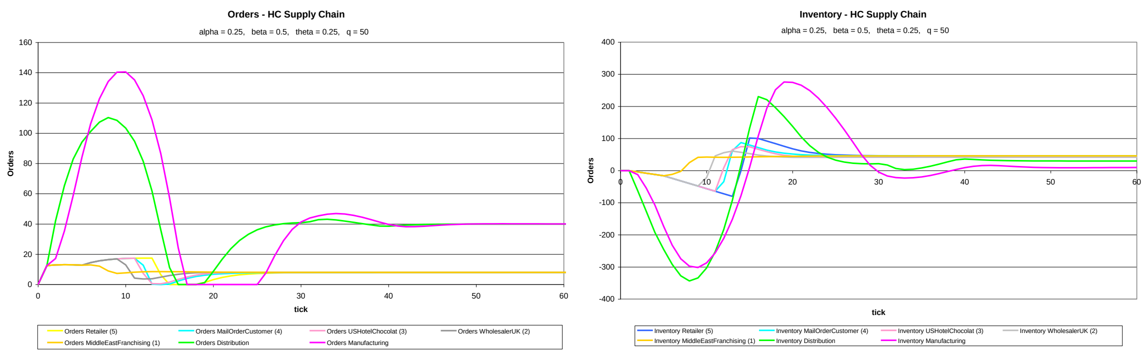

Figure 9 Simulation of the variation in inventory (left) and the corresponding order rate (right) for each of the different agents in the distribution supply chain. The calculations were performed for α = 0.25, β = 0.5, θ = 0.25, q = 50 (constant

demand). The numerical simulation reproduces the amplification and phase shifts characteristic of the hand simulations.

3.3.3. Implementation in Java

As mentioned previously, all the products considered in this study are perishable. The shelf life of a product is measured in days, counting from the day it is produced until the product becomes unacceptable for consumption or obsolete (Donselar et al., 2006). This end date relates to the date mentioned on the product. In Excel it is difficult to define quantities with different BBE dates for the same time period *t*. In Java that issue is overcome by the use of arrays. The first Java simulation stage was performed with no BBE losses considerations, so that the model output in Java could be verified and validated with the model built in Excel. For the presented model, the parameters were defined as per Table 3. In Chapter 4 these equations' parameters will be optimised; at this stage these parameters fit a purpose and are generic.

Parameters	**Retailer**	**Mail Order**	**US HC**	**Wholesaler**	**GCC**
α	0.25	0.25	0.25	0.25	0.25
β	0.5	0.5	0.5	0.5	0.5
θ	0.5	0.5	0.5	0.5	0.5
q	450	300	22	410	19

Table 3 Non-optimised equation parameters.

Figure 10 is an example of the Java output. The results are similar to the excel simulation so the built model in Java has been validated.

i);	demandRetailCustomer	expectedOrderRetailer	newOrderRetailer	ordersInPipelineRetailer	supplyRetailCustomer	backordersRetailer	stockRetailer	inventoryRetailer	receiveShipmentRetailer	demandMailOrderCustomer	expectedOrderMailOrder	newOrderMailOrder	ordersInPipelineMailOrder	supplyMailOrderCustomer	backordersMailOrder	stockMailOrder
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	2500	0	0	0	0	0	0	0	0
4	466	0	0	0	466	0	2500	2500	0	661	0	0	0	661	0	1350
5	661	21	15	0	661	0	2034	2034	0	362	595	678	0	362	0	689
6	663	49	46	15	663	0	1373	1373	15	315	385	705	678	315	0	327
7	701	76	76	46	701	0	725	725	46	395	322	274	705	395	0	690
8	704	104	107	76	146	0	70	70	76	521	388	110	274	521	0	1000
9	531	131	137	107	107	558	0	-558	107	474	508	507	110	474	0	753
10	461	149	157	137	137	982	0	-982	137	487	477	767	507	487	0	389
11	482	163	172	157	157	1306	0	-1306	157	160	486	708	767	160	0	409
12	678	177	188	172	172	1631	0	-1631	172	106	192	0	708	106	0	1016
13	551	199	212	188	188	2137	0	-2137	188	269	115	0	0	269	0	1613
14	691	215	230	212	212	2500	0	-2500	212	349	254	0	0	349	0	1349
15	658	236	253	230	230	2979	0	-2979	230	480	340	112	0	480	0	1000
16	670	255	274	253	253	3407	0	-3407	253	542	466	698	112	542	0	520
17	842	273	294	274	274	3824	0	-3824	274	362	534	1088	698	362	0	90
18	792	298	321	294	294	4392	0	-4392	294	437	379	525	1088	437	0	426
19	790	320	346	321	321	4890	0	-4890	321	531	431	30	525	531	0	1077
20	817	341	369	346	346	5359	0	-5359	346	557	521	217	30	557	0	1071
21	866	362	392	369	369	5830	0	-5830	369	588	553	741	217	588	0	544
22	766	384	416	392	392	6327	0	-6327	392	640	585	1048	741	640	0	173
23	686	401	435	416	416	6701	0	-6701	416	529	635	940	1048	529	0	274
24	1066	414	449	435	435	6971	0	-6971	435	764	540	346	940	764	0	793
25	1536	443	481	449	449	7602	0	-7602	449	865	742	482	346	865	0	969
26	2223	491	534	481	481	8689	0	-8689	481	1137	853	1086	482	932	0	450
27	2803	568	619	534	534	10431	0	-10431	534	1387	1109	1886	1086	1086	205	0
28	4197	667	729	619	619	12700	0	-12700	619	2044	1359	2289	1886	1886	506	0
29	3007	823	902	729	729	16278	0	-16278	729	2020	1976	2990	2289	2020	664	0
30	340	920	1009	902	340	18556	0	-18556	902	3063	2016	2631	2990	2990	395	0
31	305	894	980	1009	0	17994	0	-17994	0	1010	2959	3714	2631	951	468	0
32	454	868	952	1989	0	18299	0	-18299	0	56	1203	1506	5394	56	527	0

Figure 10 Java simulation output.

Subsequently, further complexity was included to the model. As chocolate is a product with shelf life, a BBE parameter at every stage of the process needs to be considered at each weekly period breakdown (*t*). The biggest challenge of the BBE implementation in Java was to convert the data from an integer (quantity) for both orders and shipments, to an array of integers (BBE date, quantity) for the upstream data for the stock, so the all the balance equations needed to be reviewed to reflect this. Similarly to Excel, both orders and shipments data contains quantities information transferred between the agents, but all the shipments also hold information about shelf life (meaning minimum shelf life to sell, in days). In Java, the overall architecture for each agent is as follows:

- Demand, Orders, Manufacturing Forecasting: array {quantity}
- Shipments, Stock, Stock BBE Losses, Production Manufacturing: array {quantity; BBE date}

The BBE losses correspond to quantities with BBE date greater than 25 weeks.

For the manufacturing agent

```
ArrayList<Integer>forecastingManufacturing = newArrayList<Integer> ();
ArrayList<ArrayList<ShipmentData>>stockManufacturing =
newArrayList<ArrayList<ShipmentData>> ();
ArrayList<ArrayList<ShipmentData>>productionManufacturing =
newArrayList<ArrayList<ShipmentData>> ();
```

For the other agents

```
ArrayList<Integer>demandCustomer = newArrayList<Integer> ();
ArrayList<Integer>expectedOrder = newArrayList<Integer> ();
ArrayList<Integer>newOrder = newArrayList<Integer> ();
ArrayList<Integer>ordersInPipeline = newArrayList<Integer> ();
ArrayList<ArrayList<ShipmentData>>supplyCustomer = newArrayList<ArrayList<ShipmentData>>
();
ArrayList<Integer>backorders = newArrayList<Integer> ();
ArrayList<ArrayList<ShipmentData>>stock = newArrayList<ArrayList<ShipmentData>> ();
ArrayList<ArrayList<ShipmentData>>receiveShipment = newArrayList<ArrayList<ShipmentData>>
();
ArrayList<Integer>stockBBELosses = newArrayList<Integer>();
```

The diagram below is an overview of Components of the SC Business Model and specifies the agents, its relationships and the types of outputs from the simulation.

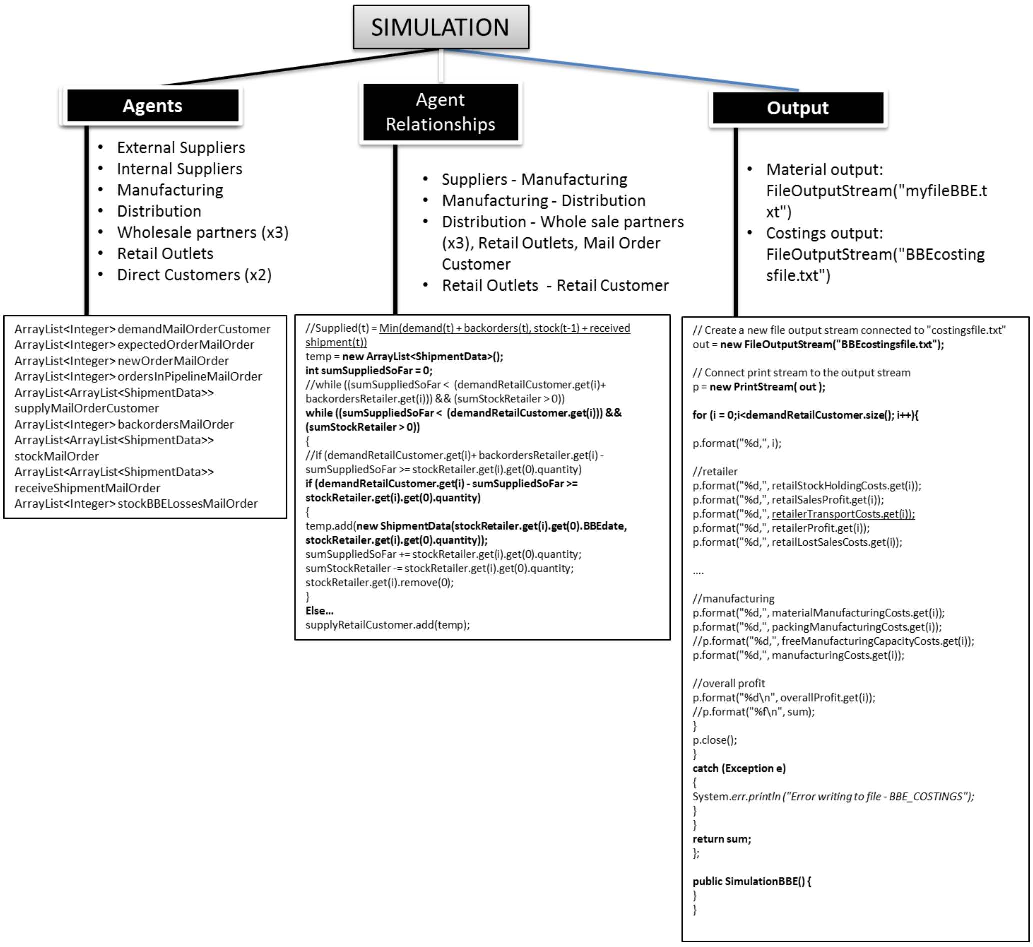

Figure 11 Components of the SC Business Model.

Once the model was built, the next step was to determine which equation parameters (α, β, θ, q) optimise the overall SC performance. In the following chapter a Genetic Algorithm is the proposed method to determine the parameters for optimisation of the business profit. The concepts behind the Genetic algorithm are detailed and the implementation in Java presented. Furthermore the financial flows are finally described (these are specific for this model in particular).

4. GA OPTIMISATION METHOD

Numerical simulation complements the traditional empirical and experimental approaches to research since they provide effective ways for organising existing data, focus experiments through hypothesis generation, identify critical areas where data are missing, and allow virtual experimentation when real experiments are impractical or just too expensive. Numerical simulations can also be used to forecast short and long-term consequences of particular choices of parameters and/or initial conditions in real experiments (de Oliveira et al. 1999, Stauffer et al. 2006).

In this chapter a Genetic Algorithm (GA) approach was used to determine the different equations parameters (α, β, θ, q) for each individual agent that Optimise the overall HC Supply Chain Profit: through minimum stock holding for each individual agent, guaranteeing availability, decrease BBE date losses and improving forecasting and ordering accuracy. From the different literature reviews, GA works well in problems with high complexity levels and with high levels of solution possibilities and constraints, as it does not try to find the exact solution, but rather tries to find the approximate neighbourhood of the optimal solution.

4.1. GENETIC ALGORITHM BASIC CONCEPTS

A genetic algorithm (GA) is a search technique used in computing to find exact or approximate solutions to optimisation and search problems. Originated by John H. Holland in the 1960s, is based on the principles of natural selection, this stochastic search algorithm is able to explore good solutions quickly on large and complicated search spaces. The power of the algorithms comes from the mechanism of evolution, which allows searching through a huge number of possibilities for solutions, and its representations and operations simplicity in the GAs is another feature that makes this methodology so popular.

Firstly, a set of individuals are chosen, corresponding to a set of possible solutions. In each generation (iteration), the best members of the population generate new members (sons) from a set of operators. At the end of each iteration the worst sons are removed, forming the population for the next generation with the remaining ones. The aim is to find the values of the parameters so that the overall output of the system is optimum. The decision variables are discrete in nature and the problem at hand is a typical combinatorial optimisation problem. The strength of GA lies in its ability to manipulate many parameters simultaneously, thus making it attractive and suitable for applications as an optimisation meta-heuristic in a wide variety of disciplines (Gonsalves *et al.*, 2007). For many realistic problems, such as manufacturing, communication and network design, GA can often find good solutions (near-optimal) in relatively short search period (Ding *et al.*, 2005).

The basic form of an optimising meta-heuristic can be framed as follows (Reeves, 2003):

As sum a discrete search space χ. Further, assuming that the optimisation problem is one of maximisation, the general problem is to find:

$$x, x \in \chi \ldots\ldots\ldots\ldots\ldots (1)$$

So as to maximise:

$$f(x) \ldots\ldots\ldots\ldots\ldots (2)$$

Subject to:

$$g(x) \leq 0 \ldots\ldots\ldots\ldots\ldots (3)$$

$$h(x) = 0 \ldots\ldots\ldots\ldots\ldots (4)$$

where x is a vector of decision variables, f is the objective function, and, g and h represent the inequality and equality constraints, respectively.

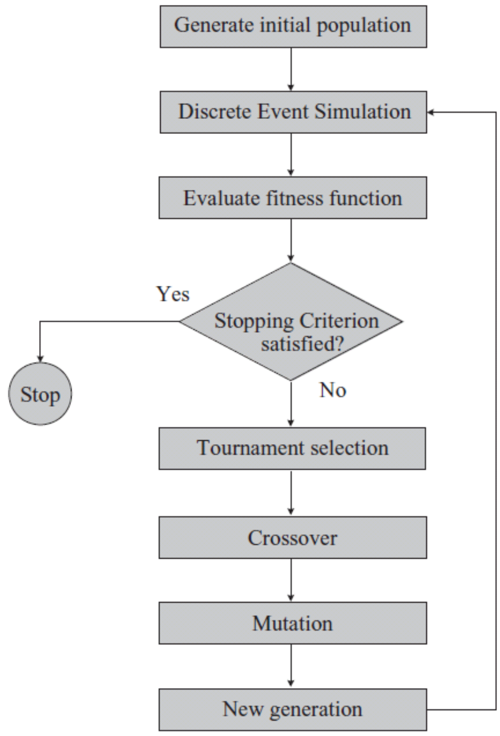

Figure 12 Outline of the Genetic Algorithm (*Source:* Gonsalves *et al.*, 2007).

In the following subsections, the bounds on the decision variables are described, then the objective function and finally the constraints under which the objective function is to be minimised.

A typical genetic algorithm requires two conditions to be defined: a genetic representation of the solution domain and a fitness function to evaluate the solution domain. The procedure of a genetic algorithm is as follows:

1. Generate random populations of feasible solutions for the problem.

2. Evaluate the fitness function of each solution (which, depending on the type of fitness, a function to maximise or minimise).

3. Create a new population by repeating the following step until the population is complete:

- select two parent solutions from a population based on their fitness value (the highest is the fitness the most likely is to be selected);

- crossover the parents to form new spring solutions;

- place new offspring in the new population;

4. Use new generated population for a further run of the algorithm.

5. Evaluate the new chromosomes.

6. If the terminate condition is satisfied, return the best solution; else go to condition 2.

The following parameters can be manipulated to get the desired results in GA:

- *Crossover probability* (Cp) —indicates how often the crossover will be performed. If this value is zero then the new generation is made from exact copies of chromosomes from old population, the higher the value of this parameter more offspring are made by the crossover and vice versa.
- *Mutation probability* (Mp) – indicates how often parts of the chromosome will be mutated. If there is no mutation the offspring generated just after the crossover or directly copied without any change. If the mutation is performed, than the parts of chromosome are changed. The higher this constant, the higher is the number of times the chromosomes are mutated and vice-versa.
- *Population size* (P) – indicates how many chromosomes are in the population. If there are too many chromosomes, the algorithm may slow down.
- *Generations* (G) – indicates how many generations will be in the population.

The pseudo code of a basic genetic algorithm is shown on Figure 13.

```
t – actual generation; d – criterion to finish the algorithm; P – population
{ t := 0;                              //counter
Starts_population(P,t);                //starts a population on n individuals
Evaluation (P,t);                      //evaluates individuals fitness
Repeat until (t = d)                   //tests criteria (duration, fitness, and so on.)
        { t := t + 1;                  //increases the counter of generations
        Selection_of_parent s (P,t);   //selects the couples for crossover
        Recombination (P, t);          //accomplishes selected couples crossover
        Mutation (P, t);               //disturbs the group generated by crossover
        Evaluation (P,t);              //evaluates new fitness
        Survive (P, t)                 //selects survivors
        }
}
```

Figure 13 Basic pseudo code of a Genetic Algorithm (Source: Ferrolho & Crisóstomo, 2005).

Genetic algorithms start with a population *P* of *n* individuals, where each individual codifies a solution to the problem. The evaluation of each individual's performance is based on a function of fitness evaluation. The best ones will tend to be the progenitors of the following generation, therefore transmitting their features to the next generations.

There are advantages and disadvantages to the use of GA. The advantages relate to how bad generation proposals are discarded from consideration and hence not affecting negatively the end solution, it can quickly evaluate a large solution area and very complex and loosely defined problems. Some of the disadvantages relate to the fact that GA does not evolve towards a good solution but evolves away from a bad one so, if not modelled correctly, can get caught in a local optimal solution; the number of iterations can be very high to arrive at near optimum solution, if the problem is not defined correctly; it is very difficult to achieve an optimal solution in very complex scenarios. Moreover, according to Ding et al. (2005), the differences between GA and other traditional methods are mainly:

- GA uses an encoding of the control variables, rather than the variables themselves.

- GA searches from one population of solutions to another, rather than from individual to individual. It is a great advantage for searching noisy spaces littered with local optima (instead of relying on a single configuration to search through the space).
- GA uses only objective function information to guide itself through the solution space, not derivatives. Once GA knows the value of 'goodness' about a configuration, it can use this to continuing to approach the optimum.
- GA is probabilistic in nature, not deterministic. This is a direct result of the randomisation techniques used by Gas, which is not the case for most existing methods.
- One of the most attractive advantages of using GA as a design tool is its ability to find solutions to problems in a way completely free of preconceptions about what is possible and what is not.

4.2. GA AS SC OPTIMISATION TOOL

The majority of supply chain approaches relates to optimisation of ordering and stock levels and thereby efficient supply chain management. The genetic algorithm methodology has been used with different approaches, namely on supply chain optimisation and to determine the best business practices.

Truong & Azadivar (2003) combine simulation, mixed integer programming and a genetic algorithm. The genetic algorithm provides a mechanism to optimise qualitative and policy variables, and the mixed integer programming model reduced computing efforts by manipulating quantitative variables. Finally simulation is used to evaluate performance of each supply chain configuration with non-linear, complex relationships and under more realistic assumptions.

Vergara*et al.* (2002) developed an evolutionary algorithm (EA) for optimal synchronisation of supply chains, using the economic delivery and scheduling model and analyse supply chains dealing with multiple-components. The EA is shown to be much faster at solving large problems than an enumeration procedure and exhibits robust behaviour when tested on a variety of different problem parameters.

Optimisation is the methodology for improving the quality and desirability of a product or product concept. Jeonga et al. (2002), for building a generic forecasting model applicable to SCM, proposed a linear causal forecasting model and its coefficients determined using the proposed genetic algorithms (GA), canonical GA and guided GA (GGA). Gonsalves et al. (2007) used a Genetic Algorithm for the operational optimisation of collaborative systems, where the cost function to be optimised was to find the minimum value of the cost function under operational constraints. Senouci & El-Rayes (2009) presented a model with multi-objective optimisation that provides new and unique capabilities including generating and evaluating optimal/ near-optimal construction resource utilisation and scheduling plans that simultaneously minimise the time and maximise the profit of construction projects. Sourirajana et al. (2009) explored the use of GAs to solve the Single Product Network Design Model with Lead time and Safety Stock Considerations.

O'Donnell et al. (2006) employed a genetic algorithm (GA) to reduce the bullwhip effect and cost in the MIT beer distribution game. The GA is used to determine the optimal ordering policy for members of the SC. The paper shows that the GA can reduce the bullwhip effect when facing deterministic and random customer demand combined with deterministic and random lead-times. This paper examined the effect of sales promotion on the ordering policies and shows that the bullwhip effect can be reduced, even when sales promotions occur in the SC. The same authors, (O'Donnell et al., 2009) studied the detrimental effect of promotions on the supply chain, one of the main causes of the bullwhip effect, and a genetic algorithm was proposed to reduce these negative effects. GAs were used to dampen the impact of the bullwhip effect and can be used to assist supply managers in predicting reorder quantities along the supply chain.

Daniel & Rajendran (2005) studied the performance of a single-product serial supply chain operating with a base-stock policy and to optimise the stock levels in the supply chain so as to minimise the total supply chain cost (TSCC), comprising holding and shortage costs at all the installations in the supply chain. The effectiveness of the proposed GA (in terms of generating base-stock levels with minimum TSCC) is compared with that of a random search procedure. In addition, optimal base-stock levels are obtained through complete enumeration of the solution space and compared with those yielded by the GA. It is found that the solutions generated by the proposed GA do not significantly differ from the optimal solution obtained through complete enumeration for different supply chain settings, thereby showing the effectiveness of the proposed GA.

Lu et al.'s (2009) research presents an extension to the genetic algorithm approach to reducing the bullwhip effect by investigating the individual efficient or responsive strategy for each member in different online supply chains. Four types of supply chain structure, by positioning the decoupling point, were investigated to determine if the genetic algorithm (GA) can help find optimal ordering policy and lead time for each member and, at the same time, reduce the impact of the Bullwhip effect and total mean cost across the online supply chain. They showed that the optimal supply chain structure that presents better performance on both the total lead time and the mean cost should be employed.

Vijayalakshmi et al. (2011) attempt to design an Intelligent Forecasting Engine which uses a combination of forecasting techniques. The design was based on the use of Genetic Algorithms, for selecting the best methods to combine for forecasting. Radhakrishthinan et al. (2010 – 2 articles) wrote several articles where GA was used for minimising the total supply chain cost through the reduction of holding and shortage cost in the entire supply chain: stock optimisation in the supply chain is distinctively determined to achieve minimum total supply chain cost. Azadeh et al. (2010) address the successful application of GA-simulation to simulation model optimisation and design, through the stochastic behaviour of their supply chain system. Zhu et al. (2011) used a genetic algorithm is to solve the supplier's replenishment model, where experiment results demonstrated the feasibility and the effectiveness of the replenishment strategy. Priya & Iyakutti (2011) presented an approach to optimise the reorder level (ROL) in the manufacturing unit taking consideration of the stock levels at the factory and the distribution centres of the supply chain, which in turn helps the production unit to optimise the production level and minimising the stock holding cost. A genetic algorithm is used for the optimisation in a multi-product, multi-level supply chain in a web enabled environment: the prediction of optimal ROL enables the manufacturing unit to overcome the excess/ shortage of stock levels in the upcoming period.

The main objective of this section is to highlight the potential of Genetic Algorithms as an optimisation tool for different types of supply chain realities. The field of application of this methodology is immense as described previously: optimise the production levels, minimise stock holding costs, optimise supplier's replenishment and ordering policies, and optimise forecasting methods, optimise synchronisation of supply chains, optimal scheduling plans, reduce these negative effects of marketing promotions, just to name a few. From the literature review the article of greater interest is of O'Donnell et al. (2006) who proved that by using historical data, the optimal ordering policy for an SC could be found by employing GAs. According to these authors, by employing the ordering policies determined to be optimal, the bullwhip effect in SCs will be reduced.

In this work a genetic algorithm (GA) is proposed to determine the equation parameters that optimise the ordering levels, and minimising the holding and shortage costs in the entire supply chain, maximising overall profit. Simulation will then be used to evaluate the base-stock levels and the ordering parameters generated by the GA. The proposed GA is evaluated with the consideration of a variety of supply chain settings in order to test for its robustness of performance across different supply chain scenarios. Both the optimisation and simulations are performed in Java and the output data is then analysed in Excel.

4.3. Genetic Algorithm & The Fitness Function

As mentioned previously, the objective of the implementation of GA is to determine the different equation parameters (α, β, θ, q) for each individual agent, with the objective of optimising the overall HC supply chain profit: through minimising stock holding for each individual agent, guaranteeing availability, decreasing BBE date losses and improving forecasting and ordering accuracy.

As mentioned in Chapter 3, the SC analysis considers a distribution agent which delivers to the different channels and receives products from the manufactory agent. Stock management of the distributor is a priority policy. Each one of the five channels orders independently and, if there is stock in the distribution, the non-fulfilled orders are kept as backorders which will be fulfilled for each channel as soon as there is stock in the distribution, in the following order: Middle East Franchising (1), Wholesaler (2), US Hotel Chocolat (3), Mail Order (4) and Retail Outlets (5). The delivery time is stochastic and, in general equal to 2 or 1 (weeks) depending on the agent. There are two direct end customers (mail order customer and retail customers) and the other customers are seen as business partner customers. The goal is to determine the target inventories and the ordering quantities to the distribution centre in order to fulfil the overall customer demand, in order to minimise the expected stock and distribution costs in a finite planning horizon.

The problem was structured around:

1. Optimal ordering quantities to keep up with demand at Middle East Franchising, Wholesaler, US, Mail Order and Retail Outlets (channels);
2. Optimal stock levels to keep up with demand at Middle East Franchising, Wholesaler, US, Mail Order and Retail outlets;

The objective is to find the optimal policy for each agent in order to maximise the overall supply chain profit. In this context, the objective of the GA is to determine which genes (parameters) maximise the profit. The fitness function corresponds to the profit, therefore the main objective is to achieve the maximum value for that profit.

In general, the profit equals to the product sales minus the incurring costs (transport costs, holding costs, BBE losses, etc.). In Table 4, the different costs types for each individual agent are defined.

Agent	**Retailer | Mailorder**	**GCC | Wholesaler | US**	**Distribution**	**Manufacturing**
Costs	Transport Costs BBE losses Stock Holding Costs	Transport Costs Stock availability penalty	Transport Costs BBE losses Stock Holding Costs	Material Costs Packing & Handling Costs
Sales	Product Sales	Product Sales	NA	NA
Profit	Sales – Costs	Sales – Costs	0	0

Table 4 Structuring the profit for each agent (from the HC perspective).

Note that in this case in particular and to simplify the calculations, as all agents belong to the same company, so product sales values only exist for agents with end users, so for the manufacturer and distributor sales values were not considered, (just costs). In reality, in most of supply chain cases this is not the case, so there are specific profit levels for each agent. In most companies all of the different agents from the supply chain are considered as different cost centres with individual costs and profits; in terms of cost accounting it keeps the cost accounts clearer and it allows to each agent to keep a measure of its individual (budget) performance.

Solving any optimization problem begins by its modelling, which consists of translating the problem into mathematical language, starting with the definition of variables, identification of the fitness function (objective) and identification of restrictions in order to obtain a model that allows an objective resolution of the problem. The following equations define the money flows for all the agents in the HC supply chain. Note that the type of costs and values are specific to this supply chain.

$$Fitness\ Function\ for\ GA\ analysis = Profit = \tag{11}$$

$$= \sum_{t=1}^{T} (P_t retailer + P_t mailorder + P_t US + P_t wholesaler + P_t GCC - C_t distribution - C_t manufacturing)$$

where *T* is time horizon and *t* is time period, and *P* corresponds to individual profit for the ordering agents (Retailer, Mail Order, US Hotel Chocolat, Wholesaler UK and Middle East Franchising), and *C* corresponds to the costs for the distribution and manufacturing agents.

Retailer and mail order profit

$$P_t retailer|mailorder = SP_t - CT_t - CBBE\_L_t - CSH_t =$$

$$P_t retailer|mailorder = \mathrm{S_t}.\frac{\mathrm{RSP}}{1.2} - \mathrm{PC}\ - \mathrm{S_t}.\mathrm{Cut} - PC.BBE\_L_t - Stc_t.CH \tag{12}$$

where S_t are the sales at time period *t*, SP_t Sales Profit at time period *t*, CT_t are the transport costs at time period *t*, $CBBE\_L_t$ the BBE Losses Costs, CSH_t stock holding costs, RSP the retail selling price, *PC* the product cost, S_t is the shipment, *Cut* is the unit transportation cost, BBE_L_t are the BBE losses, $Stc_t$is the stock at period *t* and *CH* the cost of holding per item.

GCC, Wholesaler, US agents profit

$$P_t GCC|Wholesaler|US = SP_t - CT_t + SAP_t = S_t.\frac{RSP}{1.2}.AS\% - S_t.Cut + BO_t.Pen \tag{13}$$

where SAP_t is the stock availability penalty at time *t*, *AS%* the agreed sale ratio %, BO_t are the backorders quantity at period *t* and Pen is the penalty cost per product, which corresponds to the penalty that the distribution has to pay to these partners if it misses or any delays in any deliveries.

Distribution Agent costs

$$CD_t = CT_t + CSH_t + CBBE\_L_t \tag{14}$$

where $CSH_t = Stc_t.CH$ where Stc_t is the stock at period *t* and *CH* the cost of holding per item.

Manufacturing agent costs

$$CM_t = CMat_t + CProd_t = \ UPC.M_t + \ PHC.M_t \tag{15}$$

where $CMar_t$ corresponds to the material costs and $CProd_t$ the packing & handling costs. UPC is the unit production cost for plant per unit period, M_t is the manufactured quantity at period, PHC is the Packing & Handling Costs at period *t* per item, and M_t is the number of items produced.

The GA was implemented in Java, and the implementation relates to an individual product line with well-known demand pattern. For the implementation of GA we considered a Population size of 156 $[P_0 \ \cdots P_{155}]$ and 200 Generations $[G_0 \ \cdots G_{199}]$. The Crossover probability (Cp) and the Mutation probability (Mp) are, respectively, 0.7 and 1. The table below represents the considered population, interval values:

Genes	Parameters	Interval values
G0	$\alpha_{retailer}$	[0, 1]
G1	$\beta_{retailer}$	[0, 1]
G2	$\theta_{retailer}$	[0, 1]
G3	$q_{retailer}$	[0, 1, 2, 3, ...]
G4	$\alpha_{mail\ order}$	[0, 1]
G5	$\beta_{mail\ order}$	[0, 1]
G6	$\theta_{mail\ order}$	[0, 1]
G7	$q_{mail\ order}$	[0, 1, 2, 3, ...]
G8	α_{US}	[0, 1]
G9	β_{US}	[0, 1]
G10	θ_{US}	[0, 1]
G11	q_{US}	[0, 1, 2, 3, ...]
G12	$\alpha_{wholesale}$	[0, 1]
G13	$\beta_{wholesale}$	[0, 1]
G14	$\theta_{wholesale}$	[0, 1]
G15	$q_{wholesale}$	[0, 1, 2, 3, ...]
G16	$\alpha_{middle\ east}$	[0, 1]
G17	$\beta_{middle\ east}$	[0, 1]
G18	$\theta_{middle\ east}$	[0, 1]
G19	$q_{middle\ east}$	[0, 1, 2, 3, ...]

Table 5 Genes Characterisation.

Realistically, the number of generations required to reach the true optimum is unknown at the beginning of the analysis. Traditional genetic algorithms seem to require prior knowledge of the convergence characteristics of the problem. With less intensive analyses this problem is typically overcome by having a huge population and simply running the analysis out many more generations than required.

In Java the randomisation and mutation is achieved using the following code:

α, β, θ	*Randomisation:* `this.setGene(2, m_rand.nextDouble());` *Mutation:* `this.setGene(2, this.getGene(2) + m_rand.nextGaussian()* 0.01);`
q	*Randomisation:* `this.setGene(3, m_rand.nextInt(1000));` *Mutation:* `this.setGene(3, this.getGene(3) + m_rand.nextInt(21)-10);`

Table 6 Randomisation and mutation coding in Java.

Demand information in 155 consecutive periods (*T*) for the different agents (Middle East Franchising, Wholesaler, US Hotel Chocolat, Mail Order and Retail Outlets) were generated and calculated by GA to provide the stock control policies of each agent. Both the orders and shipments were assumed to be integer values. 30845 solutions from the scheduling periods are compared and the combination, which represents the greatest value of profit, is selected as the stock control policy of the system as per the following schematics:

$$\begin{bmatrix} Fitness\ Function & \boldsymbol{G_0} & \cdots & \boldsymbol{G_{199}} \\ \boldsymbol{P_0} & F_{(P_0,G_0)} & \cdots & F_{(P_0,G_{199})} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{P_{155}} & F_{(P_{155},G_0)} & \cdots & F_{(P_{155},G_{199})} \end{bmatrix} \rightarrow \begin{bmatrix} Gen & Pop & \alpha_{\text{retailer}} & \cdots & Q_{\text{middle east}} \\ 0 & 0 & & \cdots & \\ 0 & \vdots & \vdots & \cdots & \vdots \\ 0 & 155 & & \cdots & \\ 1 & 0 & & \cdots & \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 199 & 155 & & \cdots & \end{bmatrix}$$

4.4. GA Results

The purpose of the GA approach is to determine the best parameters that optimise overall supply chain profits (fitness function) maintaining good levels of stock availability, wherever possible. As discussed in Chapter 3, the manufacturing scheduling was based on the TSD method and the BBE losses in the supply chain were incorporated in the coding. In this scenario the best solution (greater profit margin, which corresponds to the greater fitness function) converges at generation 192 and population 22, with a fitness function of £946,213. Figure 14 shows GA iterations and the optimal solution.

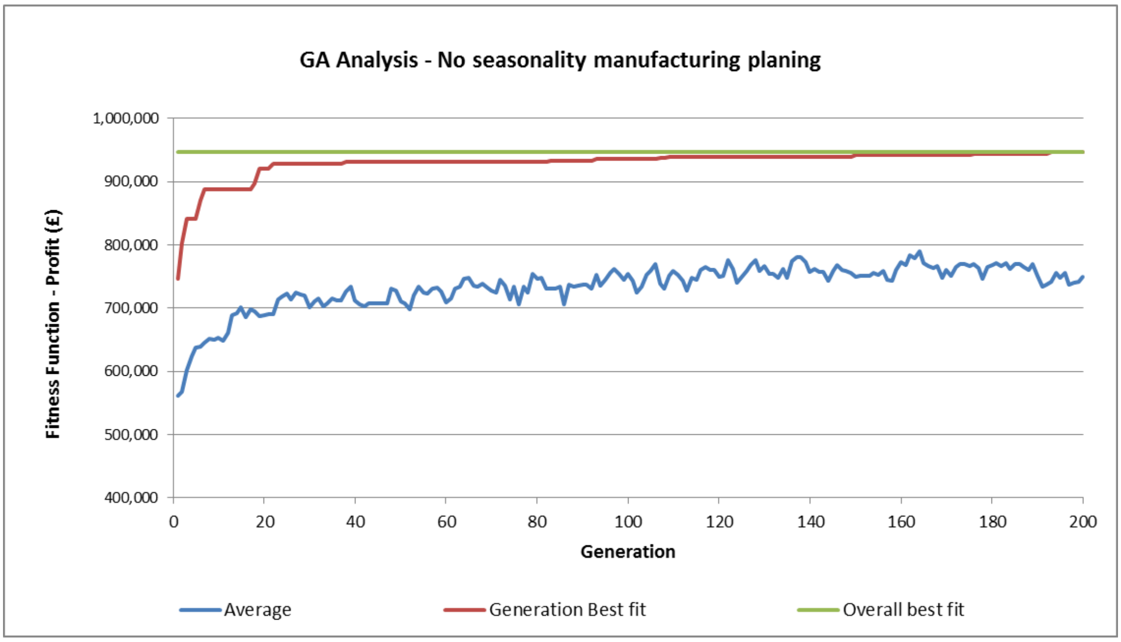

Figure 14 GA iteration and Optimal Solutions with BBE considerations and TSD method for manufacturing scheduling.

The determined parameters from the GA simulation are presented below which will allow us to determine the optimal ordering levels, stock holding and shipment quantities for all agents.

	Retailer	Hcmailorder	USHotelChocolat	Wholesaler	MiddleEastFranchising
α	0.026	0.768	0.296	0.959	0.781
β	0.193	0.780	0.682	0.004	0.788
θ	0.005	0.953	0.754	0.385	0.300
q	473	478	472	965	307

Table 7 Optimisation parameters: GA analysis.

Analysing the Java Output results of the SC simulation for a three year period and based on the data for one SKU only, the optimisation of the supply chain reflects no product losses in the system due to shelf life issues. Actually these losses correspond to lost sales (circa 35 thousand units) due to stock availability in both the retailer and mail order agents during a five weeks period of the year, corresponding to the Christmas period (Figure 15). During these five weeks, there is a 50% growth week-on-week, the optimisation algorithm makes the decision to pass most of the lost sales to the mail order agent even though there is enough stock in the distribution agent to cover that demand.

Figure 15 Overall Supply Chain Sales Losses – TSD manufacturing planning.

The reason for these results is that the forecast for each agent is dependent on both recent demand and forecasted values, and the minimum stock levels and speediness of delivery are not able to cope with such a sudden increase in the demand, as highlighted in Figure 16. These results show that the manufacturing agent production scheduling method needs to be reviewed in order to start to produce ahead of the demand peaks that emerge during the Christmas trading period. At any other time of the year, the presented model is able to adjust and react to other sales increases but for that specific three weeks period ordering and replenishment for both the Retailer Agent and Mail Order Agent needs to be done in a proactive fashion. Similarly to what happens in businesses where the product demand is highly dependent on events, the ordering process needs to be different so that the stock is made available much before the demand becomes apparent, so that the sales potential is maximised. This subject will be further addressed in Chapter 5.

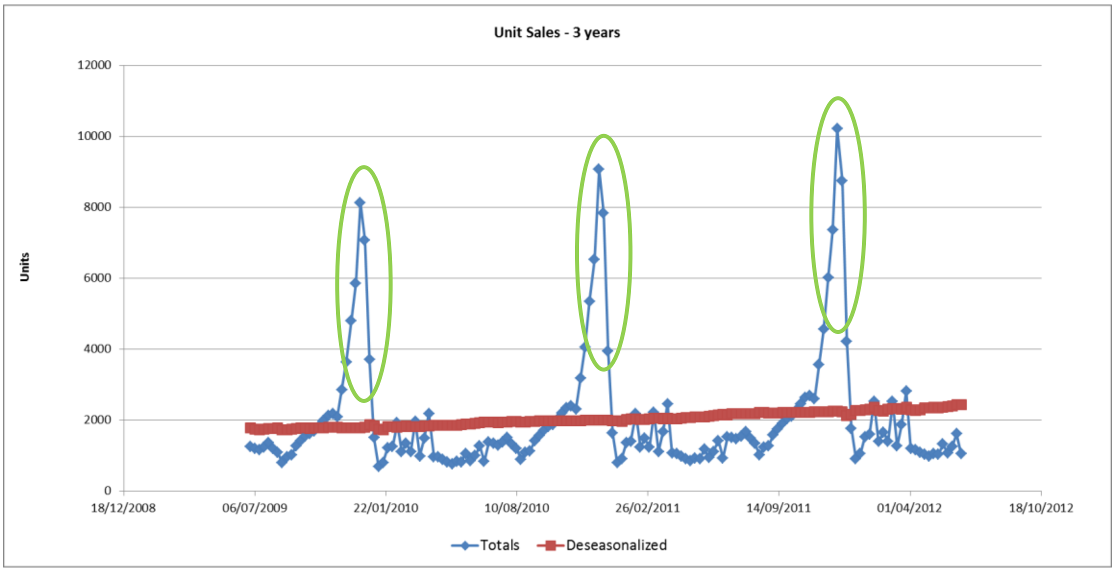

Figure 16 Time plot of number of week units sold for a three years period.

For this case in particular, due to the sudden increase in demand mainly for both the retailer and mail order agents (the other agents demand in much smaller in comparison) during 5 weeks of the year corresponding to the Christmas period (green circles on Figure 7), means that applying the TSD methodology for across the 52 weeks x 3 years period produces poor results for the manufacturing scheduling. The fact that manufacturing produces stock at a linear rate throughout the year means that when the Christmas arrives, there is not enough stock available to be able to cope with high levels of demand, so there is a backlog of orders that are not fulfilled due to the lack of availability (Figure 15). After this trading peak, the opposite is the issue, where the rate of sales decreases but the levels of production are still quite high, when it is too early to start to produce for the next peak as the company will face BBE losses. This issue was addressed by dividing the year into two different trading seasons, independent of each other, and two distinct time series decompositions applied to two different seasons as per the graph below (Figure 17, red line):

AW – Autumn-Winter which corresponds to the period from July to December.

SS – Spring-Summer which corresponds to the period from January to June.

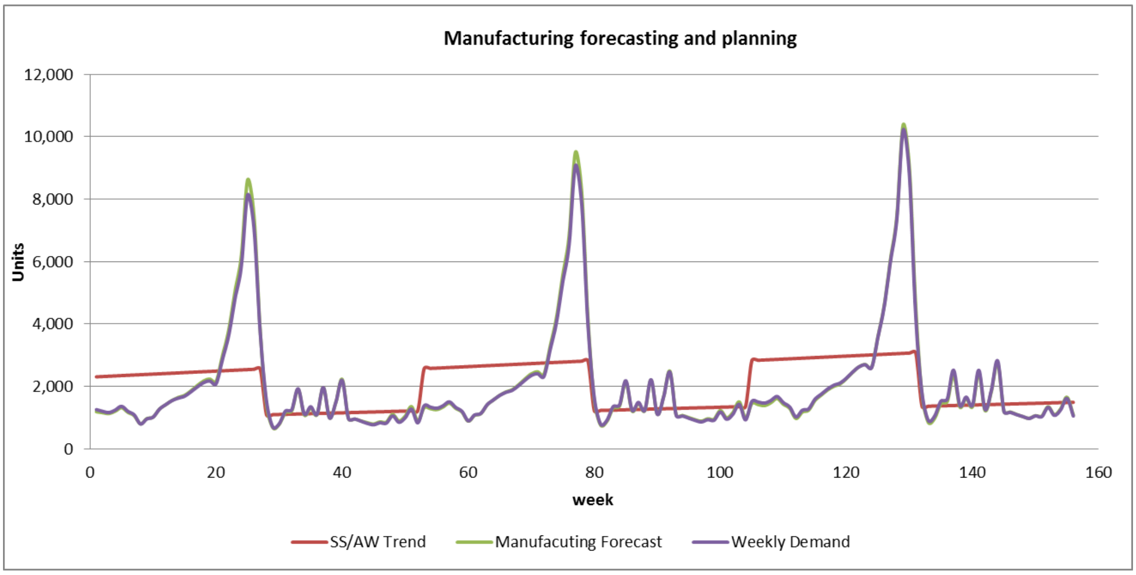

Figure 17 Manufacturing planning using time series decomposition (with seasonality vs. non-seasonality).

The split of the cycle in two different data treatments addressed the previous issues, and the measures of the forecast accuracy are good (low BIAS and MAPE —see Appendice II for further details), but the forecasts produced by this type of analysis need to be always be treated with caution, as changing conditions and changing seasonal factors will have an huge impact on long term forecasting.

For this manufacturing planning, the global fitness curve is £1,122k and the solution converges at generation 197 and at population 28. Figure 18 shows GA iterations and the optimal solution.

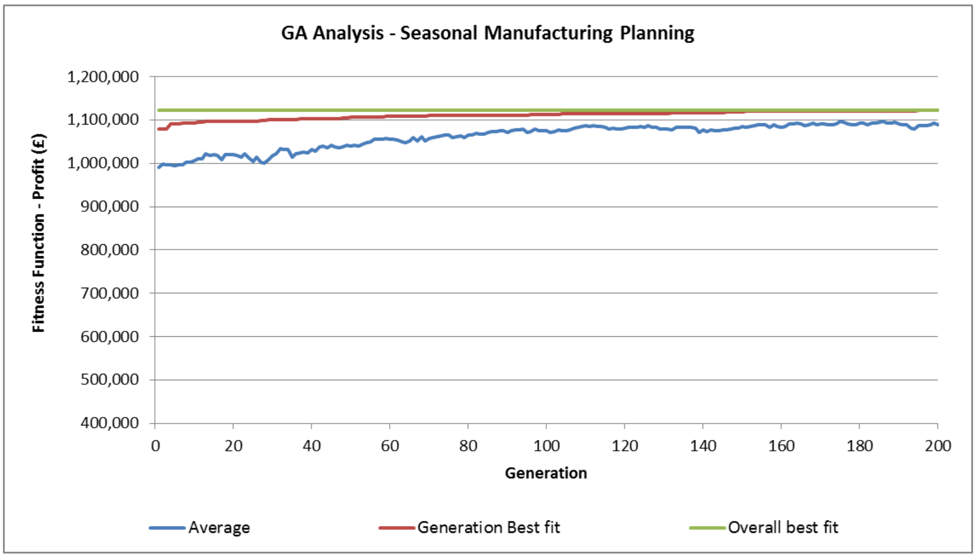

Figure 18 GA iteration and Optimal Solutions with BBE considerations and TSD method seasonal decoupling for manufacturing scheduling.

The optimal ordering levels, stock holding and shipment quantities for all the agents, results from the GA simulation are presented below. The presented results correspond to the results considering BBE losses in the supply chain:

	Retailer	Hcmailorder	USHotelChocolat	Wholesaler	MiddleEastFranchising
α	0.825	0.880	0.133	0.053	0.162
β	0.831	0.262	0.456	0.263	0.295
θ	0.953	0.992	0.242	0.287	0.080
q	163	342	107	998	137

Table 8 Optimisation parameters – GA analysis with seasonal decoupling for manufacturing scheduling.

With this manufacturing forecasting method, there is a considerable increase in the overall profitability due to a reduction of lost sales to 6 thousand units, so it is proven that this is a more suitable manufacturing planning tool than the previous Time Series Decomposition described in Chapter 3. Future simulations described in future chapters will consider this approach to plan the manufacturing activities.

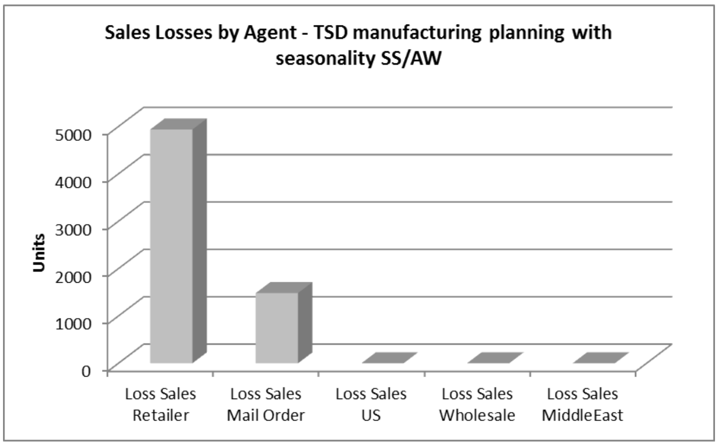

Figure 19 Overall Supply Chain Sales Losses – TSD seasonal decoupling manufacturing planning.

4.5. Robustness Analysis

According to Brown *et al*. (2000), a genetic algorithm follows in general the following structure: an initial population is created by randomly generating individuals, and this action is repeated until best individual created or maximum number of generations, which is achieved by:

1. Assigning a fitness value to each member of the current population.

2. Selecting a pool of individuals that will act as parents, using the fitness function as selection criteria (minimise or maximise the fitness function, whatever might be the case).

3. Matting a group of parents to create offspring.

4. Combining the offspring and the current population to create a new population.

The GA operation is a function which has as input a starting population and some random seeds and outputs a set of successive populations. The difficulty in setting up this structure is in choosing the right level of genericity. Too generic and the framework becomes trivial, lacking sufficient complexity to support a meaningful theory. Too specific and the properties of the framework will not be those of a sufficiently wide class of GAs.

The GA approach examines only a very tiny fraction of the possible solutions for the larger problems, yet still yields optimal or near-optimal solutions. The robustness of the GC is evaluated determining the fitness functions for different seeds. In this case in particular the full design was replicated eleven times. Figure 20 and Figure 21 show the results of eleven GA runs, and the correspondent fitness values for each seed and for each generation and the best fitness function for each seed, respectively. The seeds for the random numbers controlling the generation are different; consequently all runs start with the different initial population.

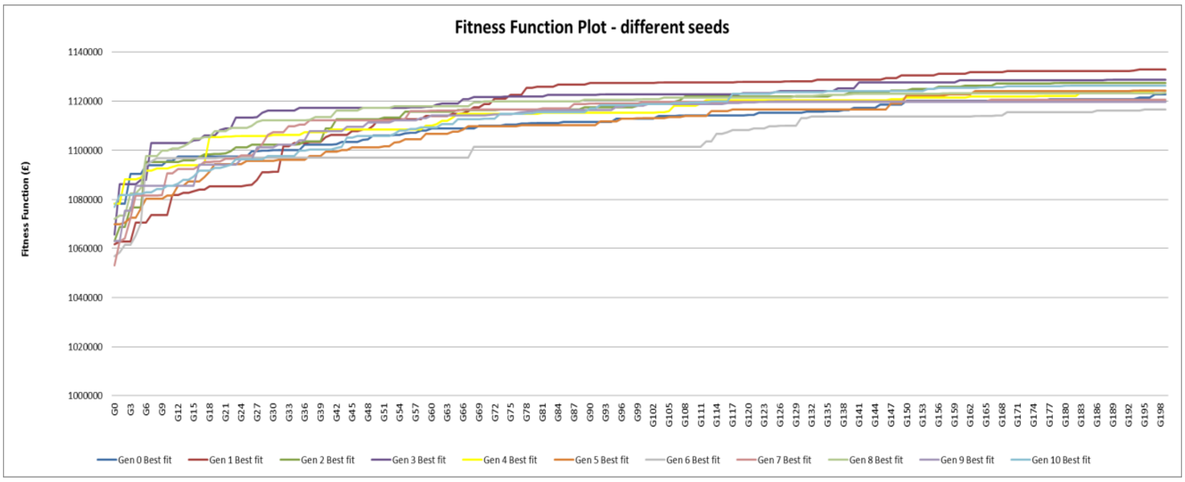

Figure 20 Plot of the fitness Function values for each seed.

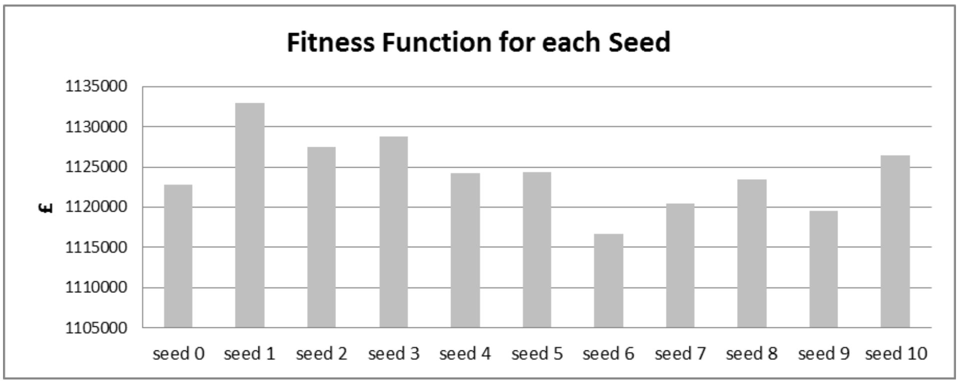

Figure 21 Plot of the overall fit values for each seed.

At the beginning of genetic search, there is a widely random and diverse population and crossover operator tends to perform widespread search for exploring all solution space. As the high fitness solutions develop, the crossover operator provides exploration in the neighbourhood of each of them. The results suggest that the seed one generates the best results in terms of optimisation, but all Gas returned optimal or near-optimal solutions on every run regardless of problem instance, problem size, or random number seed.

4.6. CONCLUSIONS

The presented GA model corresponds to a global Supply Chain Optimisation as it assumes a cooperative relationship among all stages of the supply chain in order to minimise the total operational costs of the chain as a whole, and increase global profit.

The level experiments represented a sizable number of CPU and many execution hours: eight hours for each optimisation. It is encouraging that the results appear to be applicable to a wide class of optimisation problems.

An alternative approach to the optimization of GA's would be to enable the GA to modify its own parameters dynamically during the search. However, for many optimization problems the number of evaluations which can be performed in a reasonable amount of time would not allow the GA enough evaluations to modify its search techniques to any significant degree. Therefore, the experiments described above are important in that they identify approximately optimal parameter settings for the two performance measures considered. The data also suggests several new trade-offs among the control parameters which may lead to further theoretical insights concerning the behaviour of genetic algorithms.

The effectiveness of the GA as an optimisation model is proved and the presented model will be used as a decision support system for stock allocation, manufacturing planning and stock distribution across the supply chain. Furthermore, the GA optimisation will be used in future simulations (Chapters 5 and 6) when trying to determine the impact of different factors in the supply chain and how the SC parameters interact in order to continue to guarantee the maximum profit levels.

So far the analysis refers only to one product line, but in a real life scenario the GA optimisation would be applied across multiple product lines, and the optimisation parameters determined individually for each product line. Moreover, for each product the parameters optimisation would be reviewed as often as required depending whether the demand for the product is well known or not or whether there are any other factors affecting the demand. For products that are highly seasonal, each product would have two sets of parameters, one for Spring-Summer and other for Autumn-Winter.

Shah (2004) mentioned the future challenges in supply chain are broad and complex, and provide fertile ground for research, which can be categorised under three headings: improvements to existing processes; improvements to the strategic decision-making process; future scenarios. In previous chapters the prime objective was on building and optimising the Business Model which reflects the reality of the HC supply chain, but it becomes interesting to see how the same model reacts to changes in business reality, so that the developed model can support decision making in different case-scenarios or decisions, as discussed in chapter 1. It becomes fascinating to monitor how business changes might impact on product availability, stock levels, BBE losses, ordering and supplying, logistics, etc., and still maximise overall business profit when the business model is subjected to different Supply Chain Scenarios.

5. SUPPLY CHAIN SCENARIO ANALYSIS

This chapter proves the suitability of the developed model to review different supply chain scenarios. So far, the developed model has supported core lines stock replenishment decisions with the objective of maximising profit, but we are also interested in how the model will react to business changes namely: sales patterns due to product promotions, sampling campaigns, different logistic decisions, etc. Moreover, in this chapter attention is given to the costs of being out-of-stock versus costs of product write-offs and other factors or situations that might affect overall profit namely, options between holding more stock at store level and decreasing the transport costs or vice-versa, analysis on a product with different seasonality, the impact of sampling and promotions in the supply chain. As mentioned previously, none of the scenarios to be presented considers product demand substitution (by another product), so these results will be the worst case scenario in terms of sales losses.

5.1. DEMAND VARIABILITY IN THE SUPPLY CHAIN

Supply chain instability or variability is often described as the bullwhip effect, the tendency for variability to increase at each level of a supply chain, as one move from customer sales to production (Lee et al. 1997, Chen et al. 2000). While amplification from stage to stage is important, supply chain instability is a more subtle phenomenon. The economy and the networks of supply chains embedded within it, is a complex dynamic system and generate multiple modes of behaviour. These include business cycles (oscillation), amplification of orders and production from consumption to raw materials (the bullwhip effect), and phase lag (shifts in the timing of the cycles from consumption to materials). Boom and bust dynamics in supply chains are often worsened by phantom orders - orders customers place in response to perceived shortages in an attempt to gain a greater share of a shrinking pie (Sterman, 2000; Gonçalves, 2002; Gonçalves & Sterman, 2005). There are several causes for the “The Bullwhip Effect”, namely promotional sales, inflated orders, demand forecast, long cycle times, increased safety stock, reduced service levels, inefficient allocation of resources, increased transportation costs, forecasting errors, overreaction to backlogs, lead time (of information-orders and of material) variability, delay times for material and information flow, batch ordering (large orders result in more variance), price fluctuations, free return policies, inflated orders, etc.

In any supply chain scenario, the main objective is to have the right levels of stock to fulfil the ongoing demand and minimise out of stock situations in order to maximise profit. For many companies, demand variability is increasing as actual customer demand is becoming less consistent therefore is harder to forecast resulting in several issues, mostly related to stock levels and customer service, which can have a significant impact on the top and bottom profit lines. The reason for variation in demand and supply may be predictable or unpredictable.

According to (Chopra & Meindl, 2001), a firm must decide how to handle predictable variability through capacity management, inventory management, subcontracting and using backlogs. Whereas the variations in demand can be managed using short-time price discounts and trade promotions, supply of product can be controlled combining production capacity and inventory. Capacity can be managed using any one or more of the following: flexible and/or seasonal workforce; subcontracting; combination of flexible and dedicated capacity; flexible manufacturing; standardisation and variety reduction; Inventory build-up during off-season. Out of the seven items listed, the first five are related to capacity management and the last two items are related to inventory management. By modifying capacity or managing inventory the output of the firm can be controlled. Change in capacity should result in optimising the profit.

The following approach will help to clarify the cost implications of sudden increases and decreases in demand as well as what is the best strategy for each case-scenario. Later on in this chapter the implications of scheduled increases of demand due to marketing promotions will be analysed, as well as the impact of business decisions in the supply chain.

5.1.1. Unpredictable Variability in the Supply Chain

There are natural circumstances, out of any company's control, that impact on the ongoing product demand. Those circumstances can lead to increment on the demand which may lead to under stock situations, so that sales opportunity is not being potentialised, or the opposite situation, where the lack of demand causes overstocks, which may be hard to manage in the case of food products. The main SC challenge is coping with variability in demand, which requires a balance between the stock levels and demand fluctuations.

The following case-scenarios relate to the same product and same pattern of demand described in previous chapters.

5.1.1.1. Random Demand Increase

In this scenario the objective is to determine if it is more economically beneficial to hold more stock in order to cope with random increase in demand or lose part of those sales and expect the overall system to adjust to this event. In reality, there are several reasons for relatively small and random increment in demand; for this case-study in particular, these are the main factors which increases or decreases demand:

- *Weather* is one of the most important factors influencing demand. In the case of chocolate sales weather has a significant impact on a week-to-week basis: if cold the demand increases, whilst in warm weather the demand decreases greatly. For example, customers do not purchase chocolate in warm conditions but also if it is, for example, too rainy the high-street footfall is affected but the mail order sales are increased;
- *Economic factors* for example an increase in disposable income due to for example higher wages and lower taxes;
- *Advertising, media and social networks* can increase brand exposure and loyalty to the goods and therefore an increase demand;
- Expectations of future *price increases*;
- Examples of *changes in other prices*, the impact of prices increases of some products might mean that customers will transfer their purchase to another product;
- *Non-planned product promotion or marketing* as a reaction to a competitor offer;
- A *change in the prices of related goods* (complementary goods or substitutes[2]), for example the main competitors to chocolate gifting are flowers gifting, so an increase in flower prices might impact on chocolate gifting;
- *Increase on the product exposure* in the shelves (retail) or the homepage (web), or change in the merchandising strategy;
- *Speed of change in the marketplace*, which means historical data is not necessarily a good predictor of future demand;
- *Shorter product lifecycles*.

---

[2]A good that causes an increase in the demand for another good when its price increases is called a "substitute good". A good that causes a decrease in the demand for another good when its price increases is called a "complementary good".

From the scenarios presented leading to a demand increase, two scenarios will be studied:

(i) Press coverage and advertising leading to an increase in demand for the specific product line;

(ii) How another product line out of stock impacts on this line through product substitution, one of the aspects not considered so far.

Considering a 52 weeks' time period (*T* = year), the demand has been manually changed during a period where demand is quite stable, meaning where no increase in demand was expected, as follows:

A. At week 38 the retail demand had a 66% increase on the same week last year (LFL = 66%) due to the fact that the product was displayed on the store windows following a press coverage on the Sunday Times.

B. At week 27 this product was used as a product replacement in the mail order chain after the seasonal version ran out, which meant that 800 extra units were sold.

These case-studies were analysed in two different scenarios:

(i) No changes to safety stock (determined in the GA optimisation in chapter 4).

(ii) Increase the safety stock (q was increased from 163 to 326 units in scenario A and q was increased from 342 to 682 units in scenario B, which corresponds to the average demand for the period).

The results are shown in Table 9.

52 weeks	*No changes to safety stock*		*Increase the safety stock*		
Case	*Overall profit*	*Backorders*	*Overall profit*	*Backorders*	*Agent*
A	£386k	34	£381k	302	*Retailer*
B	£378k	793	£375k	838	*Mail Order*

Table 9 Model output for increase on demand scenarios.

In scenario A, despite the 66% increase on the same period the previous year, the existing safety stock is almost enough to fulfil demand and manage the demand increase through the supply chain without major out-of-stock situations (34 backorders) - this is true even without changing the production plan. In real life, within 8 weeks the manufacturing agent will to be able to re-establish the stock levels, therefore no backorders will exist therefore is no need to increase the safety stock. In these circumstances it isn't surprise that increasing the safety stock will considerately affect the profitability.

In scenario B, the increase of demand corresponds to a period where there is already a natural uplift in demand therefore is not so easy to fulfil with extra product requirements causing an out-of-stock situation (793 units) so, even if the mail order safety stock is increased, it places further strain on the supply chain and interestingly increases the backorders affecting not only the profitability, as there are higher stock holding costs, but also a higher probability of more out-of-stocks in the longer term.

These results show that lower safety stocks levels mean higher profitability, despite the increase in out-of-stock situations. These results support Ketzenberg et al.'s (2000) conclusions, which demonstrated that excessive inventory levels impede profitability. Traditionally, lower inventories meant lower costs and lower service, but on the contrary, they proved that lost sales were lower than expected, with the lost sales of all the heuristics being minimal. In reality, another aspect to consider is that excessive inventory holding levels affects profitability through a less obvious route: by crowding out other categories of goods. A retailer with limited shelf space must face the trade-off of putting fewer categories out for sale against holding inventories of current products.

5.1.1.2. Random Demand Decrease

The same way that weather and economic factors may increase demand, they can also cause the opposite effect, thus causing demand reduction. There are other aspects that have a negative impact on demand: e.g. decrease in customers' preference and product expectations, decrease in income, increase in price, impact of seasonal products in the sales mix, etc.

Unquestionably, besides economic factors, weather conditions are one of the most important factors affecting trade. Considering the December 2010 trade, which normally corresponds to the busiest period of the year for chocolate sales due to Christmas, it becomes interesting to determine how the weather conditions affect the overall profitability for this core line. Considering the annual demand (over a 52 weeks' time period), the demand has been manually manipulated for a period where an increase in demand was expected, which never end up occurring as customers could not travel to stores and the mail order deliveries could not be delivered to the customers.

According to Retail official figures, as customers struggled to get to the shops, December 2010 UK retail sales volumes dropped 0.8% from the previous month. Assuming similar decreases for both the retailer and mail order agents for the two weeks before Christmas, this corresponds to 6,408 units of lost sales for this specific SKU. Because the product is produced beforehand, waiting for the Christmas trading peak, this naturally corresponds to an overstock situation. The impact on the other agents was not taken in consideration as they are small in comparison the sales loss in the retailer and mail order agents, and those losses correspond to the Wholesaler partner.

	Real Values		Reduced Demand		Decrease	
week	*Retail*	*Mail Order*	*Retail*	*Mail Order*	*Retail*	*Mail Order*
13/12/2010	4743	2146	2000	1500	-58%	-30%
20/12/2010	3398	2121	1500	1000	-56%	-53%
Totals	8141	4267	3500	2500	-4641	-1767

Table 10 Demand decrease scenario output values.

In normal circumstances for a core product line, if the shelf life of the product allows (which is the case), the sudden decrease in demand means that future manufacturing orders will be cancelled so that the stock levels return to the normal levels: there higher stock holding and handling costs, but this is marginal compared with having to make considerable price reductions (product markdowns) to sell the product. If the product has a short shelf life, or if it is a highly seasonal line, this means a straight price reduction (typically 50-70%) and massive profit losses as most of the times the retailers just want to recover the product cost.

In this specific scenario, even if the future manufacturing orders are not cancelled, the simulation shows that there are small losses due to BBE (673 units in the wholesaler agent), but the overall profit is decreased due to increase in stock holding costs, etc. from £383,666 to £297,279 (so overall £86,387 profit loss). If the future manufacturing orders are cancelled this corresponds to £331,837 profit so £ 51,829 profit loss. These results are quite interesting in the sense that even a poor Christmas performance means, as long as the self-life permits, the supply chain will be able to recover from a poor season, even though the bottom line profit is affected (and there is nothing that businesses can do to change that situation, just minimise it).

For seasonal lines this scenario would be different and affecting profitability greatly. As seasonal products generally have a limited lifespan, losses in the same scenario could decrease the bottom line profit by at least £10k per week per line. This is one of the main reasons why many companies prefer to invest in non-seasonal lines, as the risk is much lower and demand is much simpler to fulfil, besides the fact that the development costs of a core line can be amortised in a larger period of time, and the volumes for a seasonal line are smaller when compared to a core ongoing line. Well-planned strategies for risk management should be in place in case overstocking happens: it should take into account options for stock clearance, discounts and exploring other potential markets (if possible).

In this sense is important that businesses communicate effectively with customers on seasonal product lines to make sure that the seasonal stock is sold-out before the end of the season or it becomes outdated. This communication with the customers should convey a sense of urgency and an aspiration message to generate the feeling of “need to be the first to have”, or limited editions messages or “while the stocks lasts” in the mind of the customer.

These conclusions support the idea that businesses need not be afraid of running out of seasonal lines and keep their seasonal budget spend quite tight. When a product performs particularly well and it gives an early indication of good performance, than a reactive/ agile supply chain should be able to fulfil the extra demand indicated by those early trends (when possible, as customers are leaving to later and later their buy on seasonal events like Valentines, Easter and Christmas). To have a seasonal offer keeps the customers interested and gives them an excuse to visit the shops looking for these "new" products and buy some of the existing core products – this is an intangible effect impossible to measure. Furthermore in the chocolate market, customers seem to buy seasonal products alongside their core product favourites and impulse products because it is a season or because they are purchasing as a specific gift, so most of the times there is not a cannibalisation in terms of sales with the core lines.

From the factors that affect demand, not all the factors affect the bottom line profit and demand in the same way: for example the weather or an economic decline affects more the demand than for example a price increase, so it would be interesting to determine how those different reasons impact the overall the profit. In the actual economic circumstances, such study would be very relevant to any business.

5.1.2. Predictable Variability in the Supply Chain

In simple terms, predictable variability corresponds to changes in demand that can be predicted. Therefore if the changes in demand are roughly known, the company can maximise its profitability by responding properly to the variations in demand and supply. Predictable variability has a great impact on the company operations so it is important for marketing and operations to coordinate their efforts and plan for predictable variability together well before the peak demand is required. This coordination allows companies to pre-empt predictable variability and come up with a response that maximises profit say

(Chopra & Meindl, 2001). Examples of predictable variability include marketing activities like price promotions and price reductions, product sampling, free delivery, etc., to name a few. In the next section some of these scenarios are explored.

5.1.2.1. Impact of Price Promotions

There several reasons why companies make sales price promotions, independently of whether they are directly profitable or not: increase market share (manufacturer), to boost store footfall (retailer), and strategic considerations (for example to maintain a good relationship with an important channel partner) or other reasons like the prisoner dilemma situation: if everyone uses sales promotions, you have to follow suit, even if it is clear that it would be better to refrain from using them (Wierenga & Soethoudt, 2009). Whatever the motivation for sales promotions, their profitability should always be an important concern: if pricing and promotion are combined the objective of maximising profit may be achieved. This makes it crucial that companies in a supply chain coordinate both their forecasting and planning efforts, only then are profits maximised (Chopra & Meindl, 2001).

Despite pricing decisions, being one of the most important strategies in meeting the demand, it results sometimes in lower profitability, as changing the demand pattern may change the cost to be incurred in making the product. According to Donnovan (2003), many companies conduct sales promotions to increase the product uptake without understanding the impact on supply pipeline in both quantitative and qualitative terms. A common complaint from the manufacturing side of the business, and a common reason for severe demand distortions that cause supply chain oscillations, are unforecasted and "unknown" sales promotions. These "unplanned" sales promotions create excess costs which border on the incalculable, so how do sales or price promotions impact demand patterns, cost and margins? This is the challenge to be addressed in the following section, where we try to quantify the impact of marketing activity in both supply chain and overall business profit.

To determine the efficiency of a promotional activity in terms of sales increase as well as the impact in the supply chain profit, the ideal situation would be to analyse the same product sales with and without promotion activity in similar selling circumstances; however this is almost an impossible achievement due to the different nature of variables in question. Analysing past promotion effectiveness has some value for obtaining a general sense for what works and what doesn't work: one can observe how a particular promotion did in the past and feel comfortable that the exact same promotion will perform similarly in the future under the exact same conditions. But conditions do change, and no two promotions are exactly the same so to accurately predict the impact of specific promotions under unique conditions, companies need predictive modelling tools that can simulate the impact of hypothetical promotions and design promotions that generate the highest returns. With predictive modelling tools, different promotion tactics and strategies can be simulated in order to optimise promotion plans in order to maximise volume, revenue, profit, etc.

The developed SC model described in previous chapters allows the determination of how different scenarios will impact in the supply chain, so it becomes interesting to simulate the promotional activity and its impact on the SC.

At HC, discounts for ongoing lines are very unusual, and triggered in most cases by BBE date discount (30 or 50% price reduction). The most common (planned) multi-offers happen on seasonal product lines. Analysing real product sales data for an item that is on sale for a period of ten weeks, in the last two weeks of sale the price of the item is reduced to £12.5/unit (normally £14/unit) when the customer buys "3 or more" (therefore a 12% discount per product on multi-buy purchases). In this scenario, the multi-buy offer uptake is well known corresponding to 46% - this corresponds to 40% sales growth is then. The curves of increased product demand quantity and increased profit vs. price discount rate are shown in Figure 22.

Figure 22 The curves of increased product demand quantity and increased profit vs. price discount rate for a seasonal product line.

In this scenario we were able to dissociate the impact of the promotional activity and the normal increase in demand per comparison on sales of the previous year as there was no multi-buy, therefore the impact of the promotion is well known. The same base results and assumptions will be applied to the study of the effect of the price promotion on a core line sold normally at £20. Considering a multi-buy discount "*Buy 2 or more for £18 each*" for a period of 4 weeks corresponding to the weeks before Mothers' Day for both Mail Order and Retailer agents (for other agents the same promotion did not apply), the overall product demand increases from 4,854 to 6,795 units, and the system readjusts as follows for the mail order and retailer agents.

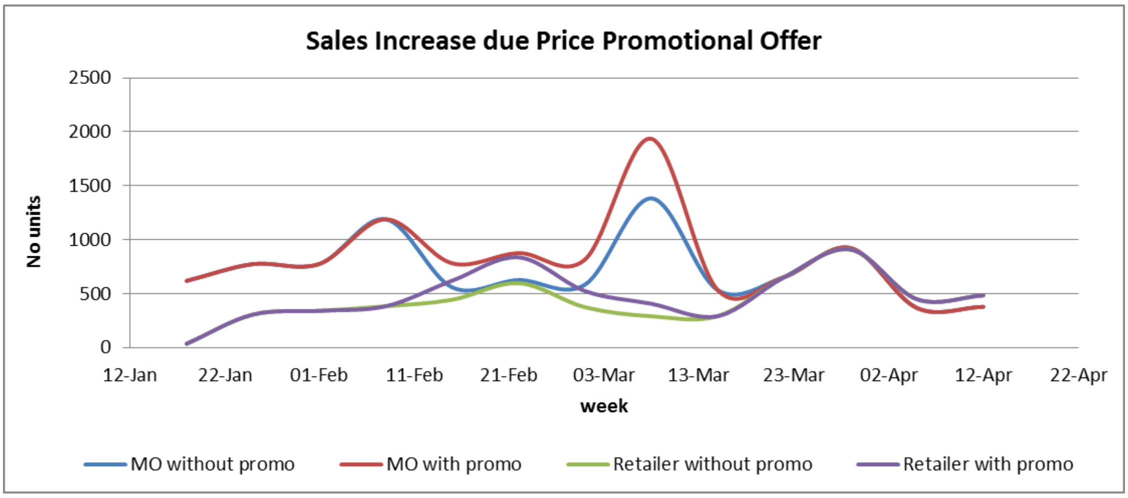

Figure 23 Mail order and retailer agents demand increase due to promotional offer.

Keeping the same optimisation parameters determined in the previous chapter (Table 7) the Java coding for the retailers and mail order profit calculation was changed as follows:

```
//promotional analysis
if ((k>=37) && (k<=40)){
sumRetailSalesProfit += (int)java.lang.Math.round(sumRetailSales * ((
(0.47*productRSPpromo + (1-0.47)*productRSP)*(1-vat)) - productCost));
}
else{
sumRetailSalesProfit += (int)java.lang.Math.round(sumRetailSales * (( productRSP*(1-vat))
- productCost));
}
retailSalesProfit.add((int)sumRetailSalesProfit);
```

Considering a period of a year, the following table shows the overall profits of all agents and overall supply chain system with and without price discount coordination. As expected, the profit increases for the retailer and mail order agents due sales increase and marginal reduction of stock holding costs; curiously, for the distribution agent there are also interesting savings due to stock holding cost reduction. In this case the manufacturing agent's planned production is robust to cope with such an increase of demand without compromising future demand, therefore the manufacturing costs don't change.

Agent	Profit with promotion	Profit without promotion	Difference
retailerProfit	£392k	£380k	£13
mailOrderProfit	£335k	£328k	£7k
usProfit	£4k	£4k	0
wholesalerProfit	£124k	£124k	0
middleEastProfit	£7k	£7k	0
distributionCosts	£3K	£4K	-£1K
manufacturingCosts	£543K	£543K	0
overallProfit	*£315K*	*£296K*	*£19K*

Table 11 Profit increase values due to price promotion for a specific SKU (one year trade), with standard GA parameters.

These results prove the supply chain ordering and supply robustness, in the sense that they prove that the built model can cope with promotional activity (planned and unplanned) without impacting on product availability and overall profit. The existing parameters allow the supply chain to continue to the ordering process without aggravation in terms of availability or SC costs increases. These results prove that the developed model is able to cope with predictable and planned price promotions without changes in the overall parameters and planned production, but it will be interesting to verify how these results would vary if the GA optimised the predicted variation in demand: would the overall profit increase even further? If so, by how much? Running the GA analysis (methodology described in chapter 4), the parameters are as shown in Table 12.

	Retailer	Hcmailorder	USHotelChocolat	Wholesaler	MiddleEastFranchising
α	0.496	0.884	0.455	0.120	0.248
β	0.206	0.028	0.632	0.013	0.856
θ	0.805	0.978	0.989	0.061	0.071
q	561	23	981	999	911

Table 12 Optimisation parameters for a price promotion scenario (GA analysis).

By incorporating these parameters in the simulation show that in reality the profits increase by £26,675, due to the increase of the mail order and retailer profit, but also the increase of the Middle East and US profit (even though there is not sales increase) due to availability increase; for the wholesale agent, the reverse happens and the profit decreases. Table 13 simply compares the profit for a planned and unplanned price promotion.

	Profit with price promotion with std parameters	Profit with price promotion and GA Optimisation	Difference
retailerProfit	£391K	£395k	£4k
mailOrderProfit	£335K	£338k	£3k
usProfit	£4K	£20k	£16k
wholesalerProfit	£124K	£116k	-£8k
middleEastProfit	£7k	£18k	£11k
distributionCosts	£3k	£2	-£1k
manufacturingCosts	£543k	£543	0
overallProfit	£315k	£342k	£27k
Profit growth	**6.32%**	**14.42%**	

Table 13 Profit values due to price promotion for a specific SKU (one year trade), with and without optimised GA parameters.

Two main conclusions can be drawn from these results. When a predicted variation in demand is scheduled, it is worthwhile reviewing the ordering and supply policies (optimisation parameters) in order to maximise the overall product profit; from those parameters review, individual channels (agents) profit may increase or even decrease, but the overall supply profit will increase. Independently of the fact the SC parameters are optimised or not, the main conclusion is that promotions work when the objective is to increase sales for a specific product, but the potential sales can be optimised using specific ordering parameters - the main objective of this analysis which was to prove that planned promotional activities can achieve better profit margins has been attained. There are other factors in the price and promotion optimisation methodology that are left to ascertain namely:

- *Price:* type of size of the discount and the role of price and how that affects the profitability.
- *Type of product*: the product affects equally the performance of a promotion, for example a product that is niche may not generate customer interest even if the price promotion is quite substantial.

- *Cannibalisation*: does the 40% promotion uptake have any cannibalisation in the sales of other products or if it is just a genuine sales uplift.
- *Seasonality*: for a seasonal event where the customer usually buys for multiple people (e.g., Christmas, Easter), the uptake on the product multi-offer is bigger the closer to that event, so in that case it makes even more sense to utilise this methodology should be applied to different promotion activities and the values analysed as well the impact in the supply chain.

5.1.2.2. Impact of Sampling Campaigns

In-store sampling is frequently used as a promotional technique designed primarily to show to the potential customer the product benefits whether they are taste related, performance, etc., hoping that a positive experience will be converted to a product related purchase(s). The sampling campaign acts in most cases at an emotional level, therefore most of the purchases are impulse purchases (not planned), making in most cases the impact of sampling campaigns even harder to quantify than a price promotion. Frequently the sampling is organised by a specialist company, rather than by the staff of the manufacturer or retailer, but in this case in particular the sampling are performed by the retail members of staff who take this opportunity to engage with the customer and talk about the product.

The chart below shows and example of a five weeks promotional activity for different products, the uptake on product related sales and product RSP (2011 data).

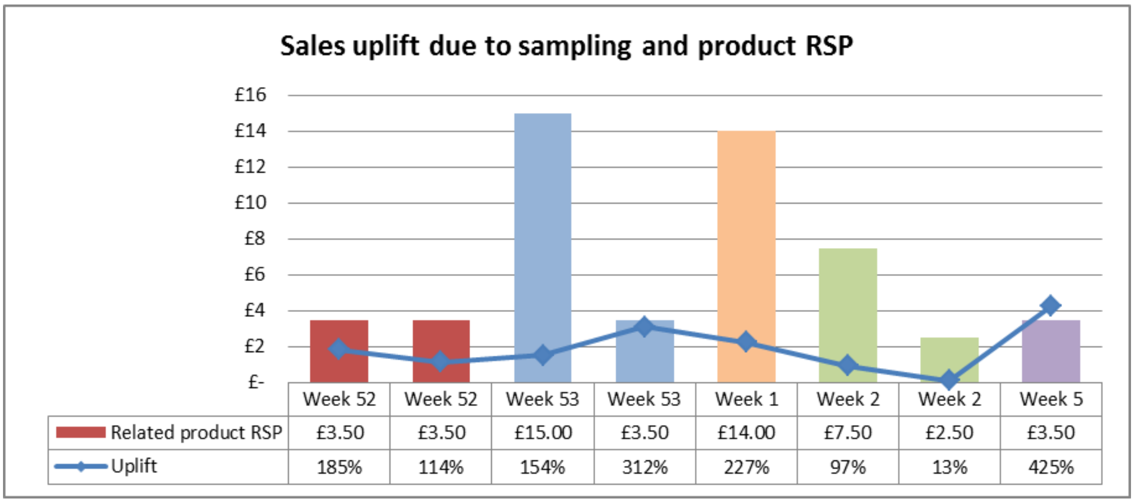

Figure 24 Sales uplift due to sampling and how that relates to product price.

The following assumptions/ conclusions are food related sampling activity in store - the same may not be necessarily applicable for a non-food sampling. In general, the success of the sampling campaign normally depends on the type of the product sampled, meaning the broader appeal the sample is in terms of target audience, higher is the conversion rate of the sampling to purchase. The largest uplift in terms of product is when customers have a preconceived idea that they would not like something but they change their opinion after tasting. This was the case on week 5 where a specific chocolate flavour was tasted and the uplift corresponds to 425%. When sampling something which relates to similar products with different price points (depending on the packaging type, bigger quantity per pack, etc.), the customer will not always opt for the lower price points (week 2 observations), even though the biggest uplifts come from lower price points (weeks 53 and 5).

The sales uplift quantities were converted into the number of samples available to be sampled to breakeven, by deduction of the product cost (sales uplift → extra income → price sample → quantities to sample → quantity per store). This information is interesting in the sense that allows to: (*1*) identify which products are more suitable to guarantee successful sampling campaigns, (*2*) determine the maximum sampling quantities by store to breakeven in terms of costs, (*3*) plan future sampling campaigns, as a similar exercise can be done to determine if a campaign will be successful or not. The quantity per store required to breakeven will be a good indicator if a campaign has been successful or not, but it does not show the sample impact in other products sales (harder to measure).

Week	***w52***	***w 52***	***w 53***	***w 53***	***w 1***	***w 2***	***w 2***	***w 5***
Uplift	185%	114%	154%	312%	227%	97%	13%	425%
Sales uplift	656	1768	256	1272	250	86	141	2276
Related product RSP	£3.50	£3.50	£15.00	£3.50	£14.00	£7.50	£2.50	£3.50
Quantity/store	546	1,473	913	1,060	1,561	574	314	3,556

Table 14 Sampling campaign performance indicators.

As a matter of fact, there are sampling campaigns where the final objective is not purely to increase the sales or even breakeven in terms of costs. Sampling is viewed as promotional exercise to first of all to engage with the potential customer and tell the story about that specific product and the store/company in general, increase the footfall (leading to sales on other products), launch a campaign for a new product as it can reduce consumers' apprehension about buying a new product or introduce them to products they were unfamiliar before. Although the potential benefits of in-store sampling are well known, there are not many literature reviews on the impact of sampling in the supply chain, as companies are usually reluctant to provide information which would enable that promotional technique performance to be assessed, so the impact of the sampling in the supply chain is still quite an unexplored area of research.

Ideally the planned sampling would always go according to plan and no disruption would happen to the SC but, as mentioned previously from observation of Figure 24, the sales uplift resulting from the sampling is quite difficult to determine beforehand. In a sampling scenario, when the planned rate-of-sales is not achieved, two opposite extreme situations may happen:

Out-of-stock – when the planned sampling performance is over the achieved sales rate, causing sale losses, included the amount of time between ordering a product and receiving it and the consequent disruptions between sales, order, receipts, restocking, manufacturing, etc.

Over-of-stock – when the planned sampling performance is under the achieved sales rate, causing excess of stock, which it is mostly located in the retailer agent, as stock was allocated to cover the increase in demand. In this scenario, depending on the quantity leftover after the sampling and the product shelf life, future manufacturing orders are cancelled and stock may be recalled from stores to feed other channels, leading to further expenses in managing this situation throughout the SC. If the shelf life of the product does not support this, normally the retailer discounts the product to increase sales rate and deplete that way the excess of stock.

Whatever the scenario might be, it is essential that the supply chain is prepared to deal with these two extreme realities and there is a contingency plan that mitigates the risks of a sampling campaign. It is fundamental that the stock is in right place so the sampling campaigns need to be perfectly planned and communicated to all the elements of the supply chain (internal or external), so that everyone understands the actions required to minimise these risks. Depending on the period that the sampling campaign lasts for the following approaches are possible:

1) if adopting an optimistic forecast, commit in terms of production to part of that forecast and then plan for the manufacturing and distribution to be reactive to fulfil the existing demand;

2) if adopting a pessimistic forecast, commit to the totality of the forecast and then plan for the manufacturing and distribution to be reactive to fulfil the extra demand;

3) if the manufacturing/ distribution is not reactive enough for whatever reason (for example lead time of the components, manufacturing capacity, etc.), commit to the optimist forecast and use promotional techniques to deploy that stock (price promotion, multi-buy, etc.).

Once again the impact of sampling activity in the HC supply chain was analysed. The period chosen for this simulation were the 2 weeks in beginning of April for the retailer agent only (normally the impact of the sampling in store in other channels is almost negligible), and similar to the approach described in the previous section was applied. The HC supply chain was subjected to three different scenarios:

(i) promotional sampling according to planed promotional plan (scenario A, 200% above normal sales target);

(ii) over performance of the sampling campaign against planned (Scenario B, 400%);

(iii) under performance of the sampling campaign against planned (Scenario C, 87%).

In terms of the optimisation ordering and supply, the parameters used are as shown in Table 7. Furthermore, for the first simulations there was no increment in the manufactured quantities, and the second simulations the manufacturing increases its production output by 200%, 3 weeks before the stock is required. The results are shown in Table 15. The results show surprisingly that for all three scenarios not altering the original manufacturing plan shows better profit results, as the current stock levels are once again enough to support fluctuations in demand without having OOS – once again it is proven that the built model is robust enough to cope with variations in demand.

	WITHOUT MANUFACTURING ADJUSTMENT			MANUAL MANUFACTURING ADJUSTMENT		
	Total without planned Promotion (scenario A)	Over performance (scenario B)	Under performance (scenario C)	Total with planned Promotion (scenario A)	Over performance (scenario B)	Under performance (scenario C)
Total sales	3561	4306	2719	3561	4306	2719
Retail Profit	£396k	£404k	£387k	£396k	£403k	£387k
Overall Profit	£11k	£19k	£1,696	£1,874	£9,915	£-7,215

Table 15 Impact of sampling campaign scenarios in the retailer agent.

The current parameters seem to be able to cope with predicted variations, but in order to improve the profitability even further the GA can be applied for all three scenarios. Following the same procedure described in the previous sections, the GA was determined for the scenario A. The results are shown in Table 16. These results show again that by optimising this model the increase in the overall profit increases by 158%, based only on the right ordering quantities.

	Profit with promotion with GA parameters	Profit with promotion with std parameters	**Difference**
retailerProfit	£408	£396k	£12k
mailOrderProfit	£30	£36k	-£6k
usProfit	£18k	£4,244	£13k
wholesalerProfit	£105k	£114k	-£9k
middleEastProfit	£13k	£7k	£6k
distributionCosts	£2k	£3k	-£1k
manufacturingCosts	£544k	£543k	0
overallProfit	£28k	£11k	£17k

Table 16 Profit values due to sampling promotion for a specific SKU (one year trade), with and without optimised GA parameters.

As a general rule, promotions should not be performed in a period where the demand is already quite high and the manufacturing capacity is already fulfilled with current demand as (1) in order to fulfil the future demand manufacturing will have to start to produce much earlier on which leads to increase in the stock holding costs, (2) it is harder to react if there is an over performance which might lead to out-of-stock situations, (3) if it is a period where sales are quite high, the fact that there is a product in promotion might detract from other products, causing an overstock situation for other products. Interestingly retailers plan their biggest promotional campaigns at trading peaks partially because other retailers are doing the same.

5.1.3. Conclusions on Variability

The previous two sections prove the robustness of the HC model when subjected to different causes of variability: predictable or unpredictable. In all cases, the profit is further improved when the predicted demand is optimised using the optimisation tool that is the GA, even though the standard parameters are also able to generate good results. For that reason, one can conclude that predicting the variability, and changing the ordering and supply parameters, better global profit levels can be achieved.

The results also show that the improvement of profits needs to be evaluated at a global scale independently of the individual agents' profit maximisation, meaning that for some agents the optimisation may mean that they will achieve lower individual profits (see Table 13 and Table 16). This is an opposite scenario to what happens in reality, where the individual agents try to achieve the maximum profit individually as they belong to different businesses/ companies, but if managers adopted these techniques there would be the opportunity to reduce overall SC costs significantly and increase overall profit. Observing the results we can also conclude that, because the wholesaler partner achieves the worst profit per product, the prioritisation is given to the other channels despite service penalties when there is an (non-predicted) increase in demand. In long term this is not a sustainable position to be, as it may jeopardise the business relationship with that channel, and possible a business partner loss, so it is essential to get the planning right first time around and predict possible causes; otherwise it needs to be business decision to give the stock to that channel and have a hit in terms of global profit, to keep those business relations in good terms.

It has been proven that forecasting techniques combined with the optimisation process of the GA can be used to reduce the bullwhip effect, by determining the optimal ordering policies for members and increase overall profitability. These conclusions support (Chopra & Meindl, 2001)'s solutions to deal with predictable variability in practice:

- Coordinate planning across enterprises in the supply chain
- Take predictable variability into account when making strategic decisions
- Pre-empt, do not just react to, predictable variability

Ideally the same approach would be applied to analyse the different types of promotion strategy (e.g. price discounts, multi-buy offers, or other types of promotions like free gifts with purchase, or discount vouchers, etc.) gaining a complete and accurate understanding of optimal price-discount rate and how sales promotions affect demand, so that information of which sales promotion policies work for each product, and how to pre-empt those variations is properly collected. Furthermore, companies perform also sampling campaigns simultaneously with price promotions to promote a certain product, so these cross-promotional techniques need also to be studied in detail.

From this analysis, the company could determine which type of promotion strategy can significantly increase demand and how to manage it to capitalise in terms of profit. Inserting various promotional strategies into demand and increasing it accordingly within the SC, the GA must be able to find the optimal ordering policy to reduce the bullwhip effect and cost across the entire SC.

5.2. Stock Location & Deliveries Pattern

Stock management and stock location are two important and close activities that can have an impact on supply chain costs. There is enormous focus to make sure the stock is "in the right quantity, in the right place" to guarantee maximum availability and maximise the sales potential. Furthermore, retailers were once the passive recipients of products allocated to stores by manufacturers in anticipation of demand; today, retailers are the controllers of product supply in reaction to known customer demand. They control, organise and manage the supply chain from production to consumption (Fernie et al., 2010). Deciding the quantity and the location of the stock (distribution centre or in store) needs to strike balance between costs and other less tangible factors. In the present business scenario, the main factors conditioning that decision are as follows:

- *Predictability of demand:* for products with low demand volatility, the risk of having the stock nearer the customer (in the SC chain) is less greater than for products with a less tracked sales history;
- *Required delivery time:* shorter delivery times mean greater capacity of reaction to demand and less probability of fulfilment delays and backorder costs;
- *Production and raw material supply lead times*: the longer the production time and raw material lead times the greater should be the stock holding requirements;
- *Holding costs*: the stock is likely to be in the location that is most cost effective;

- *Risk of obsolescence*: with respect to the products' shelf life and also the nature of products;
- *Product type and product mix*: the decision about the product will depend on whether the product is seasonal or an ongoing line, and how much percentage of stock is seasonal. Additionally, more SKUs add further complexity to the all supply. The seasonal aspects of a product will be approached in the subsequent section;
- *Transport costs*: the greater the transport costs, the likelihood of fewer deliveries;
- *Product related marketing activities*: as covered in previous sections, pricing policy and promotional activities of retailers indirectly influences the stock holding decision by influencing demand;
- *Cost of each item*: if the cost of each item is high, both the stock losses costs and the stock value is higher than if it was a lower cost item;
- *Space constraints*: depending on the stock volume and the storage space in each element of the supply chain;
- *Pre-defined service levels*: service agreements between the different elements of the supply chain;

As described in chapters 3 and 4, the built model considers an average weekly delivery to all channels following an order which depends on minimum stock holding quantities and recent demand. The overall profit (corresponding to the fitness function) is determined considering costs such as: stock holding costs, transport costs, product costs, manufacturing costs, etc. The objective of this sub-chapter is the determination of the impact on profit of the reduction of the deliveries pattern to every two weeks, and the increase of holding stock in store (retailer agent), and for the third party partners (US, Middle East and Wholesaler agents); the mail order agent will continue with a weekly supply as the despatch to the customer happens from the distribution centre. This analysis also does not take into consideration stock holding space constraints at store level and delivery handling quantities. Also the product is an ongoing line with well-known demand, and is not subjected to predicted or variable causes of demand.

In Java, the distribution agent was set up to make a distribution every two weeks to all the agents with the exception of the mail order agent as follows:

```
//delivery of stock only every two weeks

if (i % 2 == 0){
...
};
supplyDistributionRetailer.add(temp);
        }
else
        {
supplyDistributionRetailer.add(temp);
        };
```

Unsurprisingly the results from the GA optimisation show the stock holding quantities have considerably increased (Table 17) which has an large impact in terms of general profit: the sales profits go from £1,122,744 to £146,470 for a 3 years period (£976,274 reduction).

	Retailer	Hcmailorder	USHotelChocolat	Wholesaler	MiddleEastFranchising
α	0.648	1	1	0.431	0.023
β	0.485	0.014	0.029	0.247	0.053
θ	0.405	0.990	0.207	0.424	0.254
q	701	271	1000	882	990

Table 17 Optimisation parameters for a decrease deliveries scenario (GA analysis).

The output for the simulation for the retail agent is shown below – the last column in this table shows clearly that the retailer agent starts to receive the shipments from the distribution centre every two weeks (as mentioned previously these shipments happen normally in a weekly basis).

i	demandRetailCustomer	expectedOrderRetailer	newOrderRetailer	ordersInPipelineRetailer	sumSupplyRetailCustomer	backordersRetailer	stockBBELossesRetailer	sumStockRetailer	sumReceiveShipmentRetailer
0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	2500	0
4	466	0	0	0	466	0	0	2034	0
5	661	189	0	0	661	0	0	1373	0
6	663	380	375	0	663	0	0	710	0
7	701	495	826	375	701	0	0	9	0
8	704	578	655	1201	9	0	0	0	0
9	531	629	951	1856	0	695	0	0	0
10	461	589	956	2807	0	1226	0	0	0
11	482	537	903	3763	0	1687	0	0	0
12	678	515	909	4666	0	2169	0	0	0
13	551	581	1129	5575	128	2847	0	0	128
14	691	569	1076	6576	0	3270	0	0	0
15	658	618	772	7652	658	3961	0	714	1372
16	670	634	1411	7052	670	3961	0	44	0
17	842	649	1011	8463	842	3961	0	1	799
18	792	727	1023	8675	1	3961	0	0	0
19	790	753	777	9698	790	4752	0	714	1504
20	817	768	1483	8971	714	4752	0	0	0
21	866	788	1011	10454	866	4855	0	143	1009
22	766	820	1135	10456	143	4855	0	0	0
23	686	798	796	11591	686	5478	0	562	1248
24	1066	753	1257	11139	562	5478	0	0	0
25	1536	880	1316	12396	1089	5982	0	0	1089
26	2223	1146	1800	12623	0	6429	0	0	0
27	2803	1583	3112	14423	1714	8652	0	0	1714
28	4197	2077	3873	15821	0	9741	0	0	0

Table 18 Simulation output for the retailer agent for an increasing stock holding and decrease of number of deliveries.

These losses are verified across all channels and are motivated by:

i. The new minimum stock holding requirements mean that production needs to be producing much earlier in the process in order to be able to react to bigger orders which increases the backorders in the distribution agent and also affects the product shelf life;

ii. BBE date major stock losses - which was not verified in the previous scenario, as the supply chain is not able to react to changes in demand, when it reacts far too late leading to an overstock situation at store level, and products that will go out of date. To reverse this situation the same stock would have to be recalled from the stores and redistributed again, so increasing reverse logistics transport costs;

iii. Lost sales increase by 349% – these results prove that the proposed ordering model is not suitable as it will react too late to increase of demand, causing over stocks in a later stage.

In general, the change in term of patterns of delivery can only be considered when the pattern of demand is quite stable or the unpredictability in demand is then managed by setting safety stock levels, so that, for example, a distributor might hold two weeks of supply of an article with stable demand but double that amount for an article where the demand is more erratic. For that reason, the presented ordering pattern would need to be reviewed if, for whatever reason, the business decision is to reduce the number of deliveries, so the ordering method needs to change accordingly. In reality most companies with variable demand change the delivery schedule during the year to better suit the demand pattern so that they provide logistical support to replenish products quickly and hold minimal excess inventories at the store level. This also gives manufacturing the opportunity to increase the manufactured quantities and react if required by the different channels in order to minimise OOS situations. This also follows the principle of "diversifying as late as possible".

The present results highlight two points often discussed in the literature by many SC experts:

1. A supply chain is a complex network of elements with conflicting objectives —this case-study highlights one of the aspects of this problem: each business unit needs to consider its respective individual decisions and its impact on other areas of business and understand its contribution to total cost, customer service, and lead times, etc. Similar simulation models and approach are needed to evaluate dynamic decision rules for managing an inter-related series of supply chain processes and minimise risk. This simulation model also proves to be useful for measuring the bullwhip effect, as described previously.
2. Agility and adaptability —are competitive advantages reacting speedily to sudden changes in demand or supply, or changes in the markets or in the business strategies. An interesting article from Lee (2004) on supply chain agility reveals that companies whose supply chains became more efficient and cost effective did not gain a sustainable advantage over their rivals; during the same period surveys showed that consumer satisfaction with product availability fell sharply during the

same period. Only supply chains that are agile, adaptive and aligned provide companies with a competitive advantage.

To further this analysis, it would be interesting to study with greater detail different patterns of delivery for each channel and determine which scenarios would be most adequate delivery for different products and for the different times of the year, and different types of demand (conditioned by promotions, etc.), using the optimisation tools and knowledge developed in this work.

5.3. Global Expansion

5.3.1. Overview

As discussed in previous chapters, the longevity and the success of a business depend in great extent on the way its supply chain is able react to new global business realities. In the face of global competition, a company's success increasingly depends on how best it can design, manage, and restructure its SC to deal with product diversity, improve delivery reliability and timings, and also reduce system costs.

As the retail industry continues to experience consolidation, larger retailers enjoy scale economies that enable them to source raw material for and manufacturing of their products globally. Although global sourcing reduces manufacturing costs, it also increases the length and complexity of the firm's supply chain and the associated risks. These risks entail (1) country of origin (COO) issues, (2) the use of codes of conduct for suppliers, and (3) internet procurement auctions (Ganesana et al., 2009). Global sourcing makes it more difficult for firms to monitor the processes used to make the products they buy and assess the quality of those products (Roth et al., 2008). Only the companies that can manage the following external factors will stand a chance to be successful in an increasingly competitive environment:

- New information technology developments and access to new sources of information —which both companies and customers use to consider different choices;
- Speed of delivery and logistic challenges;
- Global economy and increasing global competition —customers can purchase goods anywhere in the world and have them delivered without having to leave their houses;
- Increasing customer requirements —for a wider product offer, better service levels and lower costs (made easy by price comparison at a click of a button). In this sense, service differentiation can play a big part in terms of keeping ahead of competitors.
- Product innovation, design and functionality —several companies have realised the importance of investing in R&D to try to identify customer requirements before the customers know themselves (following the Apple example);
- Products that fit the market needs, and it are aligned with the consumer trends as well as great promotional and marketing decoys;
- Increasing ethical and environmental concerns —fair-trade, organic, carbon footprint, clean technologies, sustainable sources, recyclable, etc., are common terms influencing customers' choices and buying patterns;
- World economic and social changes —recent America, Asia and Europe economic situation and relationships, as well as value of the dollar and euro;

Companies need to consider all of these aspects when exploring new markets or even new areas of business. As mentioned previously, in a globalisation context, only agile, flexible and fully integrated supply chains can support the ever changing business reality. The dynamic management of products, ideas, information and cash, alongside coordination and cooperation of all elements of the supply chain (customers, internal and external suppliers, production, distribution, etc.) can guarantee that the customer's needs are fulfilled, and profits are achieved and properly maximised. In this scenario a continual optimisation of the supply chain flows is important rather than one time calculations.

The objectives of the supply chain and the performance measurements need to be understood in order to build the most effective supply chain; performance measurements provide an approach to identify the success and potential of supply management strategies (Koprulu & Albayrakoglu, 2007). Meixell & Gargeya (2005), after extensive literature review, concluded that although most models resolve a difficult feature associated with globalisation, few models address the practical global supply chain design problem in its entirety. According to the same authors, the challenge for any particular industry is to strategically decide on those features that will be modelled, to keep the problem tractable, and thereby focus on the special structure of the practical setting; without a focus, the amount of data required are unnecessarily numerous and may be prohibitively time consuming. In the current model, most of the concerns are overcome due to the fact that HC is vertically integrated and the access to information is more straightforward.

With the ever changing customer needs means that product sales data patterns are ever changing, therefore it is essential that the supply chain can react also to those changes (downstream in the supply chain). Also new product requirements mean that a new focus needs to be on the upstream with (new) suppliers and the sourcing of raw material and new subcontractors (business partnerships) in order to deliver a product that fits customer needs and that is in alignment with the business values (upstream relationships) that are cost effective. Therefore the link of the supply chain to new product development, marketing and customer service is essential. The role of suppliers on the supply chain will be studied in the next chapter.

5.3.2. GLOBAL MODEL

Global operations are made even harder because supply chains need to adjust and operate in uncertain environments, so matching demand to supply is even more challenging. Furthermore, the bullwhip effect seems to be aggravated, by the fluctuation of stock and backorders even when levels of demand are quite stable. The ability to manage uncertainty and new supply practices become a positive advantage in a global scenario. The following approach represents an important link between model development and implementation in practice.

As mentioned in the first chapter, one of the objectives of the development of this generic SC model is the development of a comprehensive approach to a real supply chain to support not only operational day-to-day decisions, but most importantly strategic and also long term decisions. This section is a “what-if” analysis of the impact of a new business strategy: impact on the supply chain of the opening of ten stores in US within the next year. In general, part of the overall business strategic growth may come from abroad expansion to countries like America, in Europe, etc., so there is a need to be a focus on how the business should operate, and a requirement for high performing supply chains in its global markets: speedy, focused and reactive to customer needs, in order to support the business strategy and saving time and money, and achieving higher levels of customer satisfaction and finally the profit targets.

Considering that the US stores will have similar performance to the current store and weekly deliveries, the overall US supply chain demand can be considered to be ten times as the current demand and a weekly supply (as currently). So we need to understand the overall impact in manufacturing capacity, lead times, profitability, etc., and what operational considerations need to be taken. In this scenario the correspondent weight of the new US product sales (with ten stores) in the global product sales would be 5% (note that this is not weighted overall business sales, but specific for the product analysis, assuming also equivalent RSP to UK).

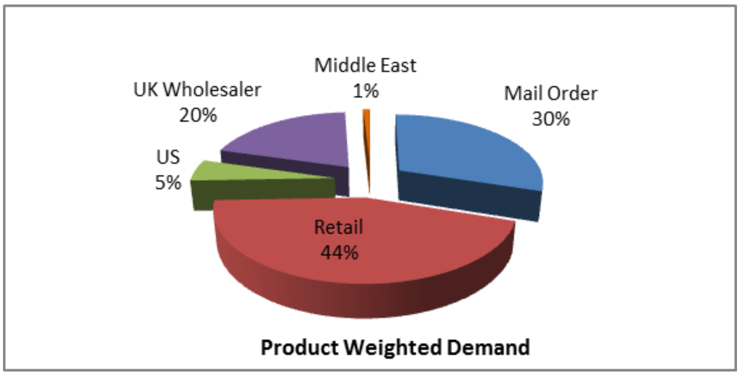

Figure 25 Weight of US product sales with the new ten stores in the overall product sales.

The forecasting demand has been updated to reflect the increase in demand from the US demand using the TSD methodology and the manufacturing agent produced quantities ranging from 318,453 to 408,254 units. In this scenario the optimised ordering parameters are:

	Retailer	Hcmailorder	USHotelChocolat	Wholesaler	MiddleEastFranchising
α	0.796	0.815	0.103	0.029	0.011
β	0.795	0.292	0.298	0.100	0.605
θ	0.992	0.646	0.317	0.489	0.441
q	52	44	727	998	523

Table 19 Optimisation parameters for a retail US expansion scenario (GA analysis).

Interestingly from the GA optimisation the minimum stock holding (green line in Figure 26) is quite high most of the year to be able to fulfil the peak for demand in week 30. This stock position does not contribute to any stock losses due to BBE (pink line in the graph below). Surprisingly the optimisation tool has chosen to keep the minimum stock holding for the Retailer and Mail Order agents and this choice is not at the expense of availability for these agents: the sales losses are very similar to the values presented in the Figure 19. For these reasons, the developed model is suitable to manage the US growth, but it would be interesting to study other ordering methods and compare them with these results.

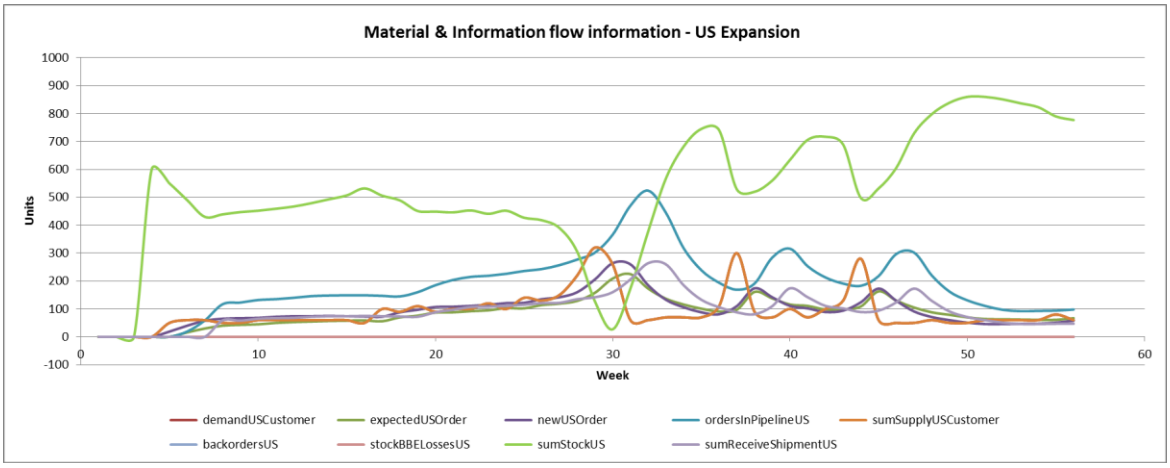

Figure 26 Material and information flows for the US after expansion.

These results seem to suit the US reality in the sense that slightly higher stock levels (in average 7 weeks, against the 5 UK weeks) are the right decision as: any random increase in demand can be fulfilled by the existing stock; if the demand is less than expected this is an ongoing core line (if it was a seasonal line or a line to be discontinued the stock levels would have to be tighter); any issues with deliveries or any other legal entities (like the FDA in US) or stock availability in UK are not reflected in the availability in US; the extra 2 weeks are reasonable considering the extra week required to fulfil an order. This average stock holding is established by the GA optimisation tool and it is the average between stock holding and demand for a year's period. These results also mean that stock holding and handling costs are quite higher that an UK store, but this fact is not reflected in the UK overall profit as the US business is dealt with as any other 3rd party partner, therefore it is up to the US team to decide about stock levels and patterns of delivery.

The presented results are a representation of the sales potential, stock holdings and reflective of the impact of the US expansion in the UK supply chain for only one SKU. Once all the parameters are fully understood, managers can plan all the variables in order to optimise overall profit through efficiency, therefore other scenarios are advisable, especially for other SKUs with different patterns of demand or seasonality. Also a multi-product scenario should be run, to gather more knowledge of supply chain costs and constrains (for example, overall stock holding throughout the year, patterns of delivery, capacity of reaction, legal aspects, overall SC costs, pricing, etc.). Similar approach could be applied to any retail expansion plan as this approach is generic, only needing the adjustment of all the variables (costs, gains) to each new reality.

5.4. MODEL SUITABILITY FOR A SEASONAL PRODUCT

The importance of product range innovation was highlighted several times in the course of this work. Variety is introduced due to the trend-oriented nature of the product and the rapid introduction of the new product options. To following a fashion model means that with this type of approach comes with highly unpredictable demand, and difficulty in forecasting, shorter selling seasons, high inventory costs, higher profit margins but also high obsolescence costs, and lower volumes per SKU. Businesses that invest in highly innovative products are synonymous with rapid change and, as a result, commercial success or failure in those markets is largely determined by the organisation's flexibility and responsiveness. Responsiveness is characterised by short time-to-market, the ability to scale up (or down) quickly and the rapid incorporation of consumer preferences into the design process (Christopher et al., 2004), and the chart below helps to visualise the impact of delayed product launches.

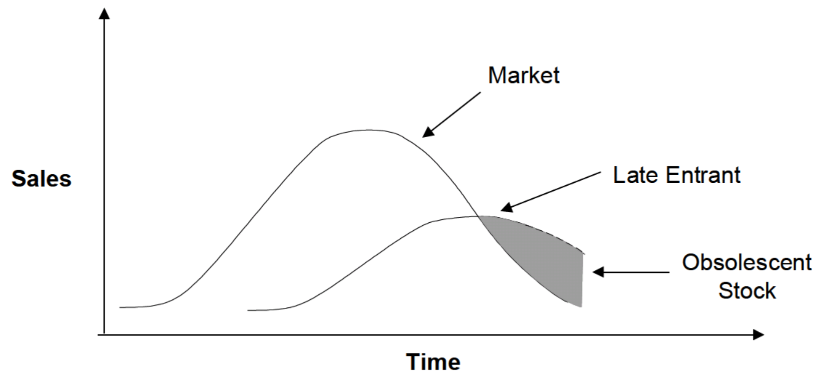

Figure 27 Impact of stock on delay product launches (*Source: Christopher et al., 2004).*

Until now, the overall approach focused on non-seasonal core product lines, but it is also interesting to determine how the existing model adapts to this different reality and how different the results would be for a non-seasonal line. With Christmas being "the" largest sales period in the year for chocolate purchases (both self-consumption and gifting), in this analysis we consider a product line which is belongs to the same product category to the product studied previously, but that it is typically a Christmas seasonal line and has overall demand as per the graph below. Week 25 in this graph corresponds to the Christmas trading week.

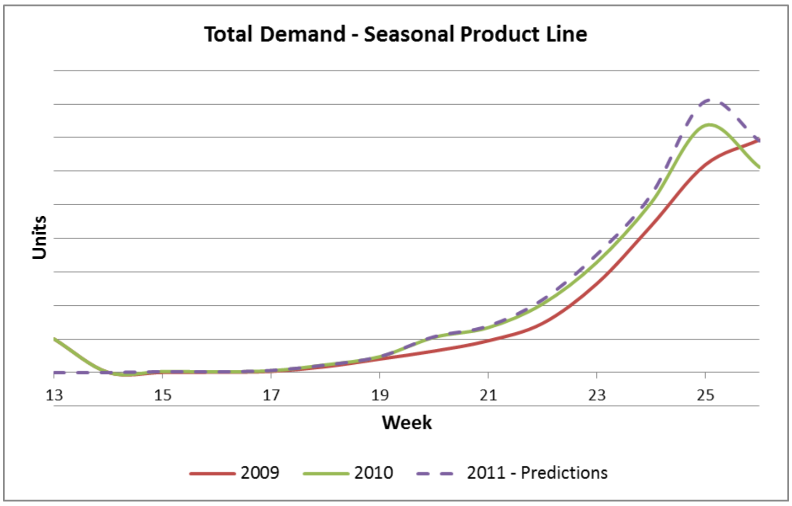

Figure 28 Demand for a seasonal line.

As mentioned, one of the biggest supply chain challenges is to be able to cope with high changes in demand due to the impact of seasonal fluctuations, trends, advertising and promotions, pricing strategies and so forth. These time-varying demand and cost parameters make it difficult to determine the most effective supply chain strategy, that is, the one that minimises system-wide costs and conforms to customers' requirements (Simchi-Levi et al., 2003). Seasonal forecasts are required in relation to raw material supply, production and manufacturing stock control and distribution and retailing and consumer demand as follows:

- *Raw material supply*: it will influence the potential volume and quality of raw materials from competitor imports from other countries, choice of suppliers, choice of sourcing location and time scales, contract arrangements, planning and requirements of raw materials.
- *Production and manufacturing:* prediction of timing, quality and quantity of raw materials, factory capacity management, production planning and scheduling.
- *Stock control and distribution:* planning and control of stocks, and stock replenishment strategy, planning of retail stocks, planning distribution, choice of distribution system.
- *Retailing and consumer demand:* planning promotional events, predicting consumer demand for quantities of given types of products, prediction of consumption patterns for demand of types of products, especially in relation to special seasonal events (e.g. bank holidays).

From the literature review it becomes clear that choosing the best method for the current pattern of demand is quite challenging due to the short sales period for the product (14 weeks), and there is a steep curve of demand in the last few weeks of trading that product which makes it challenging in terms of supply chain to fulfil such demand – this is an issue for both manufacturing and distribution.

In terms of manufacturing planning for this product, that planning was done using a different approach to the TSD forecasting methodology used in the previous chapters. Being a season specific product where the sales period correspond to 14 weeks up until Christmas day, the demand curve for this product is quite stable so the general demand curve for next year(including growth) is well known (2010 - 22%; 2011 - 24%). Generally due to manufacturing capacity constraints, manufacturers with seasonal products with well-known patterns and, shelf life permitting, starts much earlier than a core line – for this product in particular manufacturing starts to produce in September with six production slots even though the product starts to be sold in mid-October (Figure 29).

Figure 29 Manufacturing planning for a non-seasonal line.

The first four manufacturing works-orders are committed upfront by the retailers (red columns in the graph), and these volumes are just based on forecasted quantities. The stock starts to arrive to stores 2 weeks before the products are displayed in store, so the stock starts to arrive in store week 3 .The last manufacturing order (purple column in the graph) is confirmed at a later stage once the merchandisers analyse the current product sales performance against forecast: meaning that the size of the last production build will depend on the performance of the product therefore the size of the last build is "to be produced" nearer the date. Ideally this last production run is manufactured nearer the date of the event as possible (for Christmas that date would be weeks 17-18), considering that the time to send to stores (normally 1-2 weeks), meaning that the last production date would be week 15.

The same methodology described in the previous chapters was applied. The GA analysis results show that the solution converges after 100 iterations, at population 47, with a fitness function of £2,368,642. The results show that there is a huge gap between the three curves for the retailer agent demand, the retailer orders to the distribution and the fulfilment of these orders (Figure 30). As expected, this shows that the developed model is not robust for seasonal product lines as the model is quite reactive in the sense that the orders to the distribution centre are dependent on the current demand and the minimal stock levels in each agent. In essence, the model does not use enough predictive capability.

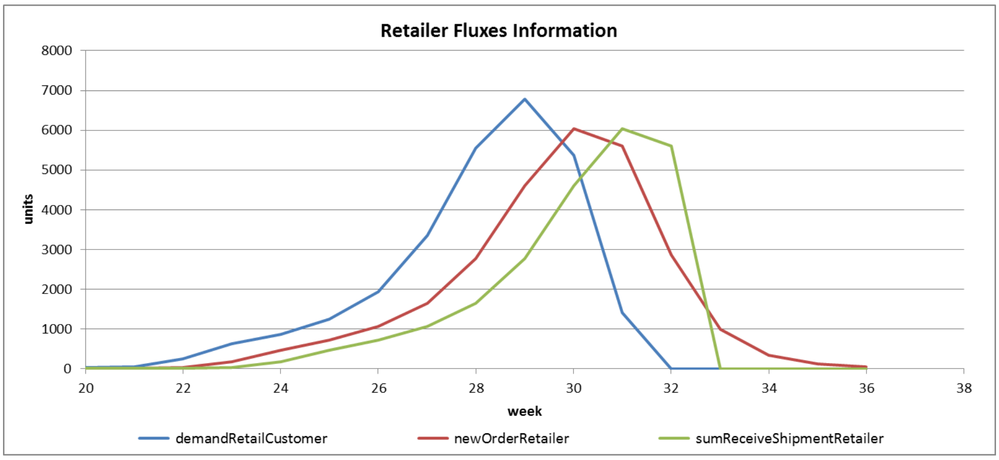

Figure 30 Retailer demand, retailer orders and the orders fulfilment plots.

Although the manufacturing agent plan schedules production much earlier on due to capacity constraints therefore having the stock ready to send out, the orders to the distribution are too late and not sufficient to fulfil that seasonal demand – in these circumstances it seems clear that the forecasting technique for each customer-facing agent needs to be different to the current model. As the demand curve is well known, and because the demand is ever growing, supply for agents such as the retailer, US and GCC need to be push rather than pull-oriented, meaning:

1. the stock needs to be already with the supply customer agents before there is any demand.
2. the quantity of stock to be sent to the customer facing agents should not depend on the current rate of sales or else they will not be able to react to increasing demand.

Bonney et al. (1999) present a very simple definition for a pull and push systems: *a pull system* is one in which the control information flow is in the opposite direction to the material flow, while a *push system* is one in which the control information flow is in the same direction as the material flow. So far in the presented model, a pull-based manufacturing system strives to synchronise production with consumption, which increases on-time delivery performance, reduces stock-outs and costly last-minute change orders; as orders arrive, material is pulled from the end of the final assembly line, which instantly sends an order to final assembly to produce more.

With the new demand reality, the distribution agent no longer waits for the orders from the lower levels; it needs instead to pull in the stock based on the predicted demand forecasted for the each consumer-facing agent so that the stock is available before the demand exists. In order to decide how much to push out to the lower layers, several traditional forecasting techniques have been considered: simple moving average, exponential smoothing, Holt's Method (double exponential smoothing), linear regression, Time-Series Decomposition for one cycle/year. All the equations for these methods are described in the Appendices and demand values plotted in Figure 31. From the observation of this graph it is clear that only two methods are suitable: Forecasted Average and the TSD method. The TSD methodology is described in the Chapter 3. In the Forecasted Average the quantity to send in each tick is given by the following equation:

$$F_t = \frac{(1 + \%g).\sum_{i=1}^{n} D_i}{n}$$

where *%g* is the percentage on growth year-on-year and in this case 30%, D_i is the demand for each week from previous year, F_t is the forecasted total demand for each tick and *n* is 14 weeks, as it corresponds of number of weeks that a Christmas product line is on sale.

Both methods are suitable to be used as ordering technique. Both methods should be created and implemented in Java to determine the best method (meaning the method that generates the largest profit).

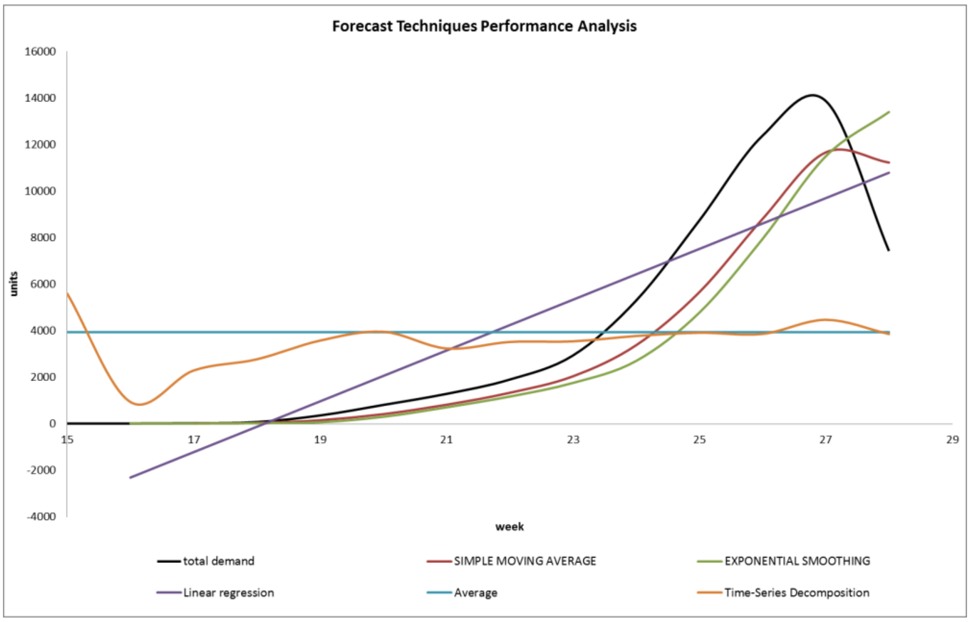

Figure 31 Forecasting techniques performance analysis for a seasonal specific product.

5.5. SUMMARY

In this chapter, simulation and scenario analysis were performed to validate an existing supply chain to identify the shortcomings and opportunities, identify the impact of changes in major demand changes on supply chain components, as well as the impact of new inventory strategies on the overall performance of a supply chain. This chapter was divided into three main areas:

The first part highlights the causes of variability or instability in the supply chain (Forrester effect), which may have a predictable or unpredictable nature, and what can be done to stabilise supply chains and improve their capacity of reaction, and ultimately achieve better profits. These SC scenarios were analysed, and its impact on stock levels, ordering levels, timings and profit studied:

- *Unpredictable variability* —impact of press coverage and product replacement in mail order and its influence the increase in demand. Also the impact of weather conditions affecting December trade and its influence the decrease in demand.
- *Predictable variability* – impact of price promotions and sampling campaigns in the increase of demand.

In both cases the robustness of the HC model was proven when subject to different causes of variability (predictable or not). In all cases, the profit is further improved when the predicted demand is optimised using the Genetic Algorithm optimisation tool, even though the standard parameters are able to generate good results. For that reason, one can conclude that predicting the variability, and changing the ordering and supply parameters, results in better global profit levels. It has also been proven that forecasting techniques combined with the optimisation process of the GA can be used to reduce the bullwhip effect, by determining the optimal ordering policies for members and increase overall profitability.

The second part presents two case-studies as an example of how the developed model can support business decision making and reduce investment risk:

1. *Impact of changes in finished goods delivery pattern.* The results show that in case of variable demand, greater agility to fulfil that demand pattern is essential to replenish products quickly and hold minimal excess inventories at the store level, giving also manufacturing the opportunity to increase the manufactured quantities and react if the different channels require in order to minimise OOSs, so in the current SC scenario the reduction of the number deliveries is not such a adequate decision.
2. *Impact of global expansion on the supply chain* and related emerging issues in supply chain globalisation. Global operations are made even harder because supply chains need to adjust and operate in uncertain environments, so matching demand to supply is even more challenging. Furthermore, the bullwhip effect seems to be aggravated by the fluctuation of inventory and

backorders even when the levels of demand are quite stable. The ability to manage uncertainty and new supply practices will become a positive advantage in a global scenario, so knowing how the supply chain needs to adjust to face the new reality is a step forward in that direction.

The third part proves the suitability of the developed method for a seasonal product line and chooses two forecasting ordering methods to order the materials ahead of demand for each member of the SC.

This approach intends to be a successful translation of the Hotel Chocolat supply chain network translating faithfully the flows of information and materials upstream and downstream of the HC supply chain network. This main objective of the current work is not only to develop a simulation model that reflects the reality of the HC supply chain but, above all, to extend the applicability of the developed model to different supply chain scenarios, and stay ahead of standard SC practices, allowing sounder business decisions in order to remain (globally) competitive. With this in mind, there is still a broad scope for further work concerning the modelling of SC; additional developments could potentially focus on the following subjects/ areas:

- Apply this methodology to another product line with different type of demand, to prove the model robustness and compare the simulation results with true results in order to validate the results of this study and highlight its potential;
- Create and implement in Java to determine the best ordering method for seasonal line(s);
- Study the price promotion analysis for a seasonal product, as many retailers do not have a full understanding of its potential on seasonal lines;
- Explore new manufacturing forecasting methods;
- Extend this approach to multi-products and understand the model's full potential. Simulate the overall SC behaviour for multi-product lines, so that the build model can simulate the overall supply chain behaviour;

- Place the supply chain network design under different kind of uncertainties and risks is an attractive research avenue with significant practical relevance;
- Investigate the impact of new and innovative ways of setting up and operating the supply chain;
- Investigate the impact of eliminating an existing and/or adding a new infrastructure component to an existing supply chain (for example, new distribution centre or new manufacturing site).
- Investigate the impact of changing operational strategies within a supply chain, due to major shifts in products, processes, location and use of facilities, etc.

To my knowledge, this is the first study that addresses the overall supply chain as a whole and supports the decision making with quantitative and qualitative data: results are validated against factual information. Furthermore, all the findings are described in great detail which allows an easy understanding of all the results, observations and presented conclusions. This approach tackled some of the most difficult global supply chain issues that are mentioned in literature but that are hard or almost impossible to substantiate or corroborate in normal circumstances. This is only possible as the company object of this study is vertically integrated and all the elements of the supply chain belong to the same company and there is sharing information across the different departments.

Hopefully the proposed modelling framework is the start of a new approach and generates new case-study examples and further developments, proving the potential of this approach in Supply Chain Management. However, further insights into the issues that arise from modelling future SC networks can be achieved by strengthening this modelling approach. Enhancing the modelling other SC networks will lead to more thorough assessments and to the development of other closely related strands of research.

In the next chapter, the impact of supplier performance on overall supply chain performance is quantified using the modelling approach.

6. IMPACT OF SUPPLIERS IN THE SC

There is still one remaining area of the global supply chain model that needs focus which has a huge influence on manufacturing and distribution tiers in the supply chain, and on the overall supply chain in general: the impact of suppliers on the supply chain performance, and in business in general. In recent years, and due to the complexity of the SCs, the collaboration between retailers and respective suppliers received new focus, as SC managers realised the competitive advantages of those relationships in the marketplace.

This chapter will address the importance of the supplier relationship in the overall SC performance as well the ordering requirements and lead times for the manufacturing agent. The relationships between suppliers and other critical components of a supply chain are investigated, by rationalising the number and size of supply points and the impact on total costs, responsiveness and overall profit. Some of the following lead times, exact links between the different agents in the supply chain and costs displayed do not correspond exactly to reality but they are instead usual values which will allow the visualisation and the exemplification of the impact of suppliers in the overall supply chain in terms of timings, availability and overall profits.

6.1. LITERATURE REVIEW

In today's environment, retailers must deal with increased competition both domestically and globally through both traditional and non-traditional channels. Changes in customers' expectations about product assortments and service, regulatory pressures for accurate data and business demands for "more for less" all drive efforts to deliver improved business performance and customer service. As a result, retailers look beyond their organisational boundaries to evaluate and integrate the resources and capabilities of their suppliers and customers and thus create superior value and a competitive advantage that they might sustain over time (Ganesan et al., 2009).

The response to fast-changing markets lies upstream of the organisation in the quality of supplier relationships. Often it is the lead time of in-bound suppliers that limits the ability of a manufacturer to respond rapidly to customer requirements. Similarly, new product introduction time can be dramatically reduced through the involvement of suppliers in the innovation process. Many companies have not recognised the competitive advantage that can be derived from closer relationships with key suppliers (Christopher, 2000). The supplier relationships represent some of the most important assets of a company and should thus be considered and treated with a similar logic to other types of investments. Exploiting some of the potential of a supplier requires that the operations team of the two companies become more closely integrated in the various facets of the relationship. This involves extensive and intense interpersonal interaction, coordination of various activities, and mutual adaptations of resources, which entails costs for both companies (Gadde & Snehota, 2000). Whether it is by coordination and integration of activities throughout the supply chain or by recognising the capabilities of immediate suppliers, understanding supply chain dynamics has a significant impact on performance. As the trend towards outsourcing and focusing on core competencies increases, organisations will be under greater pressure to effectively leverage supplier and customer relationships. The results demonstrate that doing so be a significant driver of a firm's success (Kannan & Tan, 2005). According to Spekman et al. (1999), companies with outstanding sourcing strategies appear to share two characteristics: they typically enjoy executive level commitment to building the organisation's sourcing capabilities, viewing sourcing as a cross functional capability that is linked to strategic and operational objectives while focusing on people and process; these organisations also relentlessly deploy these capabilities across the entire enterprise by creating and implementing an infrastructure of organisation, measures, and technology.

Chen & Paulraj (2004) consolidate various supply chain initiatives and factors to develop key SCM constructs conducive to advancing the field, with special interest the shift of the buyer–seller relationships to the use of a limited number of qualified suppliers. Reducing the number of primary suppliers and allocating a majority of the purchased material to a single source provides multiple benefits including: (1) fewer suppliers to contact in case of orders given on short notice, (2) reduced inventory management costs, (3) volume consolidation and quantity discounts, (4) increased economies of scale based on order volume and the learning curve effect, (5) reduced lead times due to dedicated capacity and work-in-process inventory from the suppliers, (6) reduced logistical costs, (7) coordinated replenishment, (8) an improved buyer–supplier product design relationship, (9) improved trust due to communication, (10) improved performance, and (11) better customer service and market penetration.

From personal experience, there are several (mutual) business benefits from a supplier-client co-operative relationship, namely: time, money and, above all, business relationship and trust between the customer (normally in the role as buyers of a company) and the supplier's base. It is the trust relationship that will have a greater impact also on the time and money saved whatever the business circumstance: new product development, quality issue with components or products, delay with deliveries, etc. The following business guidelines underpin the reasons why time-money-trust are key parameters:

Time – time saving can be achieved through different ways:

- Having fewer and more competent suppliers mean that there is time saving in avoiding the sourcing of new ones, having to audit them, and establishing rapport and understanding their capabilities.

- Decrease the time required to develop a new product or source a new material as the supplier understands and accommodate the business requirements.
- Decrease the time to react to any changes, whether they might be caused by development issues or by variations of demand or business requirements.

Money – besides the fact that "time is money" in the business sense, there are other aspects to consider, namely:

- Continuous improvement of supplied products, motivated by improved quality standards, new sourcing from the supplier's suppliers.
- Elimination of duplication of the tasks, as the supplier is closer to the customer requirements.
- Through sharing information on the long term business plans and how that affects them, as well as sharing forecasts, mean that they will be able to plan their future also.
- Any variation to demand (both positive and negative) sharing the information is both beneficial as if there is a decrease in demand the suppliers can also cancel material requirements with their suppliers. If there is a random increase in demand means that the supplier can react in order to produce the materials or product to fulfil that extra demand. Whatever the reality might be, that sharing of information lead to better profits for both the retailer and suppliers.
- Decrease the resources necessary to react to any changes whatever they might be caused by development issues or by variations of demand or business requirements.
- Early involvement in the product concept phase so they are also able to have an input first time so that they can also deliver an offer that is competitive and interesting to the customer.
- In general, working closely to the suppliers there is generally less wastage: due to better production practices where the production lines are run more efficiently as the supplier is not "just "trying to achieve what the customer wants, more clever purchase of materials, by the which is also reflected in the unit price.

Trust —a more open and productive relationship based on a solid relationship and commitment and information sharing:

- Frequent communication to resolve existing or potential issues and to develop new product ideas or best working practises.
- Share consumer data which in combination with the suppliers' capabilities can lead to radical and incremental innovation.
- Suppliers' ability to understand the business/ brand/ product requirements first time and work more closely to achieve better results.
- Suppliers' liking to be associated to a business/ brand/ product that does well as that shows and promotes their capabilities, creating also an image of trust with other business, bringing more potential business to them. This relationship is priceless in terms of marketing their capabilities and increasing their reputation in the market. The reverse is also verified, where the suppliers speak openly about the great business relationships with the customers they enjoy to work with.
- Long terms relationship, so any negotiation takes into consideration the long term commitment which is reflected in the price per unit.
- Furthermore, the suppliers also through their own competitor analysis within their area of expertise bring new materials, new components, and new processes, so they can also have a great input on product innovation.

There are also dangers in this type of "exclusive" relationship. Suppliers can become complacent in terms of the relationship, time and price so it is wise to have more on than one supplier for each product area/ expertise so that there is a price per unit benchmark, an alternative if for whatever reason the usual supplier cannot fulfil the orders, creating a healthy business competition. Many companies have a suppliers' day(s), where the different suppliers are invited to come along for a business day so suppliers know who are their competitors for the same potential business.

For Pishvaee & Torabi (2010), suppliers have to be managed to derive the maximum potential in the supply chain, and the selection of the supplier is the most critical task in supply management, so six strategic priorities were identified as the criteria, and the priority measures as the sub-criteria, and then an AHP-based model was formulated to select the best supplier. After finding the global priority weights, they can be used to determine the final composite priority weights of supplier occupying the last level of hierarchy. Chan & Chan (2005-b) applied the principle of cooperation to suppliers and manufacturers; five common supply chain models were built and tested with the aid of simulation. Various performance measures such as transportation cost, resources utilisation, inventory level, and order cycle time will be calculated for comparative indications. The methodology that is presented in this paper can be extended to any real life applications in SCM. The new competitive paradigm is that supply chain competes with supply chain and the success of any one company will depend upon how well it manages its supply chain relationships. Ding et al. (2005) presented a simulation optimisation methodology to make decisions on supplier selection. The methodology is composed of three basic modules: a genetic algorithm (GA) optimiser, a discrete-event simulator and a supply chain modelling framework. The GA optimiser continuously searches different supplier portfolios and related operation parameters. Corresponding simulation models are automatically created through an object-oriented process. After simulation runs, the fitness value of candidate supplier portfolio is derived from the estimations of key performance indicators (KPI). The fitness is returned to the GA to be utilised in searching the next prominent direction.

The following subsections address the role of suppliers in the supply chain and demonstrate the importance of the customer-supplier relationship to achieve good product availability and maximise profit, in a consumer reality where product preferences are ever evolving and shorter product life cycles make the fulfilment very challenging from a supply chain perspective. Furthermore the supply chain model described in previous chapters is extended to include the supplier agents, so an ordering methodology for packaging materials is presented as well the impact of suppliers in the supply chain is evaluated.

6.2. IMPACT OF SUPPLIERS TO THE SC

As mentioned in Chapter 3, in the HC supply chain there are two types of suppliers that feed materials to both the distribution centre and/ or manufacturing plant: external and internal. Both focus on low turn-around time and inventory and their operation is characterised by the supplier contracts which determine the lead-times, flexibility arrangements, and information-sharing. The image in below is an extension of the image shown in Figure 4: it shows in greater detail the front end of the HC supply chain, with all types of suppliers as well as the material flows details between all agents.

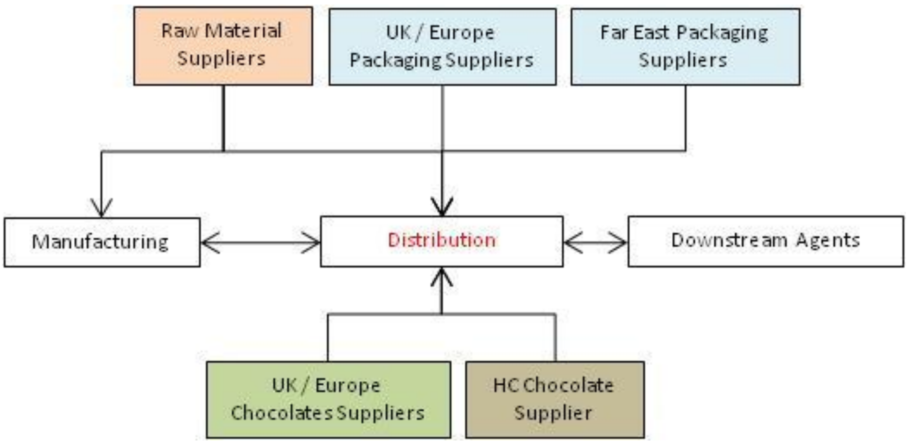

Figure 32 HC Supply Chain detail.

External suppliers can supply four types of components: (*i*) raw material to the manufacturer to manufacture individual chocolates (including chocolate as an ingredient); (*ii*) packaging components, which can be from Europe or the Far East; (*iii*) finished chocolates; (*iv*) finished product (as it is seen in store). Only the finished product is delivered directly to the distribution and then sent to downstream agents, so the manufacturing agent does not touch in that product. The other types of components are feed to the distribution agent who then supplies the manufacturer activity after a stock order, which after being processed are sent back as the finish product to the distribution centre and then sent out to downstream. The *internal supplier*, corresponding to the Plantation, works as an external supplier despite being part of the same company and it only supplies raw material, meaning cacao, to be processed to finished product by the manufacturer. It is modelled as a single agent because the parent organisation has no direct control on their internal operations and it has its own supply chain for production and distribution.

The relationship between the manufacturing agent and its suppliers is given by series of rules which translate that business relationship and agreed service levels (timings, quantities, quality standards, service levels, etc.). We assume that the inventories are managed according to an (R, s, S) policy (see inventory polices in Appendix III). In the (R, s, S) policy the inventory is checked at review moments, *R* time-units apart; only if the inventory position is at or below *s*, an order up to level *S* is placed. *R* is called the review period, *s* the reorder point. Orders are delivered with a fixed delay: the lead time *L*. Finally, backlogging of excess demand is assumed (Moors & Strijbosch, 2002). There is a periodic review single item single stage inventory system with stochastic demand. In each time period the system must order none or at least as much as a minimum order quantity.

In the developed model the following parameters and assumptions were considered:

- **Manufacturing Demand** (M) —deterministic, assumes that the demand follows a normal distribution with mean *D* and standard deviation *α*. This demand corresponds to the forecasted quantities to feed the manufacturing agent requirements.
- **Wastage** (w) – for each order placed for both materials and packaging is considered a 1% wastage which corresponds to the extra quantity ordered to be able to copy with damage, sampling, quality check, etc.
- **Minimum Order Quantity** (MOQ) —In terms of ordering from suppliers, elements like packaging are ordered less often due to the minimum production runs required therefore there is a MOQ, which normally corresponds to sales requirements for a semester. In terms of chocolates and raw material

there is also a MOQ but it is much smaller than the chocolate requirements, so that does not normally cause strategic decisions about stock holding. In terms of the plantation, whatever the harvest is, HC will buy that specific quantity as there is an exclusivity contract between the plantation in St Lucia and HC, and that quantity is normally used up until the next crop arrives. Table 20 shows the considered minimum order quantities (MOQs) for the different SC agents.

Agent	*MOQ (units)*
Far East Packaging Supplier	5,000
UK/Europe Packaging Supplier	5,000
HC chocolate Supplier	NA
UK/ Europe Chocolates Suppliers	10,000
Raw Material Suppliers	5,000

Table 20 MOQ from suppliers.

- **Lead times** (LT) – corresponds to the length of time between the placing an order for an stock item and receiving that stock. Table 21 shows the considered lead times between the different SC agents. This information will allow the updated supply chain model to include suppliers.

Agent	*Lead time (weeks)*
Far East Packaging Supplier	20
UK/Europe Packaging Supplier	4
HC cocoa supplier (plantations)	25
UK / Europe Chocolates Suppliers	4
Raw Material Suppliers	4

Table 21 Lead times (in weeks) from suppliers to fulfil their orders.

- **Safety Stock** (SS) at time t, is defined by the following expression:

$$SS_t = Maximum\ (M_{t+1} + M_{t+2} + M_{t+3} + M_{t+4}) - Average\ (M_{t+1} + M_{t+2} + M_{t+3} + M_{t+4})$$

where $M_{t+1/2/3/4}$ correspond to the manufacturing requirements for period t +1...4. The interval 4 is due to the lead time for the Europe Packaging Supplier. This stock is normally held in reserve as a cushion against uncertain demand (or usage) and replenishment lead time.

- (Re) **Order Point** (ROP) – quantity to which stock must fall in order to signal that an order must be placed to replenish an item. This parameters is defined as:

$$ROP_t = SS_t + \sum_{i=t+1}^{i=t+4} M_i$$

- **Delivery** (D_t) —it happens four weeks after the order: $D_t = O_{t-LT}$
- **Stock** (S_t) is the stock at time t, is defined as $S_t = S_{t-1} - M_t + D_t$ where D_t is the delivery at time t.
- **Projected Stock** (PS_t) corresponds to the virtual stock at any point in time and it is defined as

$PS_t = S_t + OS_t - \sum_{i=t+1}^{i=t+4} M_i$ where OS_t is the outstanding orders.

- **Order** (O_t) the ordering equation is as follows
- $$O_{t+1} = \begin{cases} 0 & ROP_t - PS_t \leq 0 \\ CEIL\left(\frac{ROP_t - PS_t}{MOQ}\right) \times MOQ & ROP_t - PS_t > 0 \end{cases}$$

where the CEIL function corresponds the rounding up to the nearest higher integer.

In this case supplier agent modules were tested and built in Excel. They could have been easily added to the HC supply chain in Java so that the ordering process could be calculated automatically each time and also any major change is requirements (due to increase or decrease on customers' demand) can be reflected also in terms of the component suppliers ordering.

The packaging scenario was analysed in terms of ordering pattern and ordering quantities with the objective of minimising total stock costs. For the finished product considered previously (core product line with known ongoing demand), each finished product corresponds 1 item from packaging from Europe, with lead times of 4 weeks. The re-ordering point (ROP) is calculated in a weekly basis, so it can align itself with the expected sales growth or decrease in demand for that specific product. The ordering pattern (new POS, O_t) and the stock levels (S_t) are shown below.

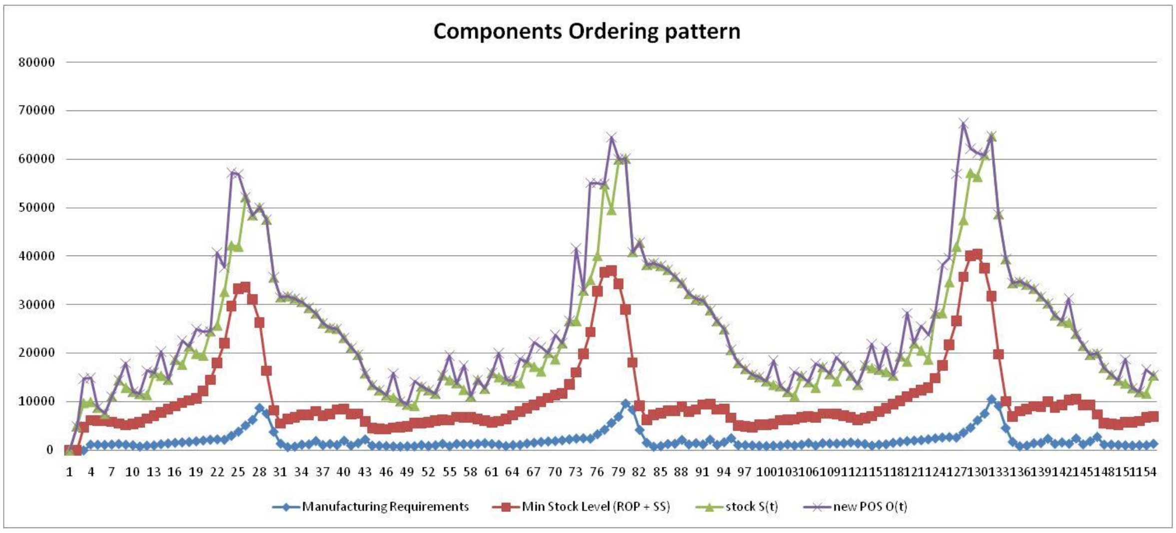

Figure 33 Packaging components for manufacture ordering pattern (units/week).

Consider a scenario where there is a four weeks' interruption in supply due to a delay in material necessary to build the packaging component. This means that the existing orders cannot be fulfilled, causing a delay in fulfilling the existing orders; after these four weeks the supplier will deliver the outstanding orders. Table 22 shows in the impact of supply delays in the supply chain; the magnitude of the impact on the supply chain will depend on when this interruption occurs. Interestingly, there are periods where a delay does not affect the manufacturing agent as the stock levels are able to cope with the non-predicted disturbance, which is the case of scenario 3.

supply shortage scenario	Delivery week	Delivery shortage quantity	Shortage to manufacturing
Scenario 1	22 + 23	10,000	0
Scenario 2	27 + 28	30,000	0
Scenario 3	25 ... 28	50,000	19712

Table 22 Interruption in supply scenarios and impact in manufacturing.

Considering the scenario 3, the component ordering scheduling reacts as per the graph below. Manufacturing requirements cannot be totally fulfilled during a three week period, corresponding to manufacturing orders of 19,909 units (weeks 25 to 28). This example of supply delay is quite extreme, and reflects the worst case scenario in terms of supplying affecting availability.

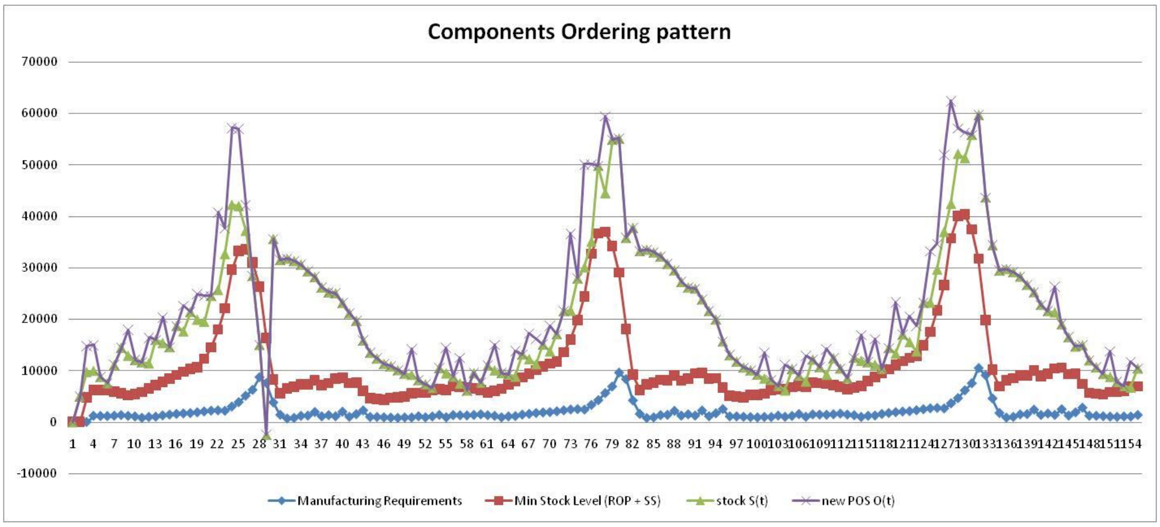

Figure 34 Packaging components for manufacture ordering pattern, after a two weeks delay (units/week).

Considering this shortage scenario, it becomes possible to quantity the impact of this delay on the supply chain overall profit using the same approach as in previous chapters. The same optimisation parameters (chapter 4) and the same manufacturing scheduling (chapter 3) were considered. The manufacturing agent manufactured quantities were reduced (weeks 25 to 28) to simulate the supply delay due to shortage of packaging components.

This supply disruption between the packaging supplier and the warehouse has the greatest impact on the supply chain, creating a “ripple effect” both downstream and upstream, as the different agents need to react to their orders not being fulfilled by the distribution agent due to the delay in manufacturing. The disruption between the warehouse the downstream only has an impact on availability and sales profit for all the agents. Despite a reduction in the distribution and manufacturing costs (£8,633 and £110,386, respectively), the overall profit decreases by £278,538, which is £73,533 less than the estimated sales loss (manufacturing quantity reduction 9712 multiplied by product profit margin).

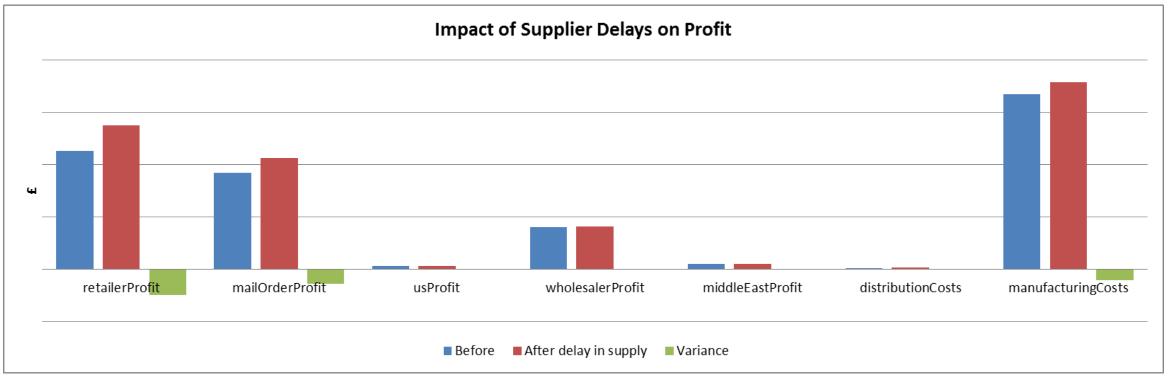

Figure 35 Impact of suppliers delays on profit.

This disruption to the manufacturing agent performance was due to supply shortages of raw materials, but other causes could have the same effect: delay in transportation of the materials from the supplier to the manufacturer (Wilson, 2007), manufacturing related constrains (Gong, 2008), non-conformance of the raw materials, forecasting errors, etc. Strategies to deal with supply disruptions need to be considered in the supply chain structure so that appropriate actions, risk management and benefits are widely understood and accepted.

This research has demonstrated the potential issue of a supply disruption and was able to quantify in monetary terms the impact of supply shortages in profitability. This study has been performed for one product line but it could have been easily extrapolated to more SKUs, helping to steer the risk management business plan in case of supply shortage of components or raw materials in general.

In chapter 5 we mentioned that global operations are made even harder because supply chains need to adjust and operate in uncertain environments, so matching demand to supply is even more challenging. The ability to manage uncertainty and new supply practices will become a positive advantage in a global scenario, or from a sourcing and purchasing perspective. One of the main questions that many Europeans companies question themselves when sourcing is what the best option is: cheaper products and longer lead times, or dearer products and shorter lead time. The next section tries to address the issue of sourcing bringing a pragmatic approach to supplier selection.

6.3. Supplier Scenario Evaluation

As discussed previously, companies face several supply chain challenges and more sourcing options due to globalisation, partly because many manufacturing companies moved outside the country to low labour cost countries in the Far-East (mainly China, but more recently India, Malaysia, Thailand, Vietnam, Bangladesh, etc.), especially in value-added items like packaging. McCullen & Towill (2000) showed that globalisation of supply chains compounds the logistical problem along three dimensions: replenishment level, time and distance. According to the theory of constraints (Goldratt, 2012), the outsourcing to the Far East has the attraction of lower purchase price per unit and can offer aggregation (economies of scale) of transportation costs.

There are significant risks that maybe overlooked when making the decision on sourcing as normaly have a substantial impact on the true cost of operating the supply chain and its ability to support the business, namely: responsiveness (capacity to react to fluctuations in demand, whatever they are increases or decreases); speed-to-market (ability to launch and capitalise on new product launches faster); price-quantities breakdowns (pressure to discount products with high stock levels); obsolescence (higher risk of slow moving stock at the end of the product life cycle); cost (high cost of storing larger volumes); dead stock (higher risk of having to write-off unusable or un-saleable stock); currency exchange (volatile exchange rates in growing economies); cash-flow (capital employed in goods in transit); risks of disturbances (single events have huge disruption on product availability). In this scenario, companies still are continuously challenged to quickly respond to changing market trends, to reduce long lead-times, to execute collaborative product development with suppliers, keep product quality standards, despite the greater physical distances, the cultural and work ethic differences and well as communication barriers.

The objective of this section is to investigate the viability of different sourcing and supply scenarios for value-added packaging, which corresponds normally to labour intensity of production. The aim is to evaluate the potential impact on manufacturing if the sourcing came from traditional European markets or from emerging markets like the Far East, and the potential risk if the sourcing were to move to a Far East Sourcing. Three sourcing options are presented:

- *Option 1* —presents the continuation of packaging sourcing from current European suppliers and current market distributions (lead limes normally 4-6 weeks).
- *Option 2* —presents a Far East sourcing scenario, where Hotel Chocolat tries to source packaging solutions from its low-cost overseas partners in the Far East and shipped to UK by sea in large containers (lead limes normally 12-16 weeks).
- *Option 3* —depicts a situation similar to Scenario 2, but the considered component could potentially be air shipped to its local distribution centre in Europe and is still able to derive margin (lead limes normally 6-8 weeks).

Options 2 and 3 are driven by similar factors, including suppliers, globalisation factors, manufacturing times and unit prices. All proposed sourcing options are currently alternative realities that many companies have to choose from, so the following approach with hopefully help to steer the business decision- making.

A scenario analysis allows strategic planning and long-term plans; it shows the drivers of change and helps to understand the levers and pulleys to take greater control of their situations and explore new opportunities. One approach is to highlight the various evaluation points to be used in the supplier comparison (price, quality standards, service levels, lead times, etc.), and internal stakeholders involved in the supplier selection decision agreed the importance of each point through weightings. This analysis creates a weighted supplier ranking that can be used to drive the final supplier selection decision making. The objective of this analysis is to develop a model based on costs, and the model should be sensitive to changes in market demand trends in order to evaluate the overall profitability under various component supplying scenarios. With this in mind, the following parameters help to characterise these different scenarios: lead times (time between order received and shipment), price per unit, MOQs, transport costs, handling and storage costs, legal/ duty costs, exchange rates. The following table quantifies all of the factors to be considered for the different scenarios.

	Scenario	Lead Times (weeks)	MOQ (units)	% discount (quant)	Flexibility	Price/ unit (£)	Transpor t costs(£)	Handling costs(£)	Legal costs(£)
1	EUROPE	4	5000	5%	60%	2	200	135	0
2	FAR-EAST BOAT	20	1000	10%	10%	0.48	187.5	135	41
3	FAR-EAST PLANE	6	1000	10%	30%	0.48	925	135	40

Table 23 Suppliers' Scenario Variables.

The best scenario will be the one that:

1. Minimises the lead times —as it requires lower levels of stock holding in the warehouse and allows better to variation on demand, therefore the supply chain can be much more reactive;
2. Lower MOQs —from a business cash-flow perspective this is a better option, as well as requires less storage capacity and less risk from a component ordering point of view;
3. Greater flexibility —this parameter relates to turn around in terms of timings and quantities and to changes in general;
4. Minimum price per unit —without compromising the quality of the packaging, the best price possible.

Considering that all of the components have the same weight in terms of the decision-making, it seems that sourcing of the packaging by plane (option 3) is the best option as retrieves the best overall score (Table 24). This scenario shows how favourable the conditions are of emerging Far East markets supply.

	Parameter	Lead Times	MOQ	% discount (quant)	Flexibility	Total Costs (£)	
	Scoring (0…1)	**1**	**1**	**1**	**1**	**1**	Weighted scoring
1	EUROPE	13%	71%	20%	60%	61%	**13%**
2	FAR-EAST BOAT	67%	14%	40%	10%	17%	**10%**
3	FAR-EAST PLANE	20%	14%	40%	30%	22%	**-3%**

Table 24 Suppliers' Scenario Evaluation.

Nevertheless due to recent world economic factors, companies are changing their sourcing approach and returning to traditional European markets due the importance placed on the lead times and flexibility as a way of adjusting to ever changing consumer demand and market requirements. Comparing scenarios 1 and 3 there is not much difference between the overall scores, meaning that both scenarios could be an alternative: 1% in score 3 against a scoring of 6% to scenario 1. Every newly proposed model should be subject to various analyses, to determine the importance of each factor in the overall choice of supplier; in very simplistic terms this means that if for example the lower MOQ and the price per unit are the most important factors then scenario 2 would be the most adequate. Table 25 shows the importance of each factor in the overall decision making, as it shows the dependence of the best scenario on the input (chosen) parameters, but above all shows the importance of an analysis of the factors shaping the decision making before considering any supplier choice.

Factor importance					SCENARIOS				
					1	2	3		
Lead Times	MOQ	% discount	Flexibility	Total Costs	EUROPE	FAR-EAST BOAT	FAR-EAST PLANE	Best Scenario	
1	1	1	1	1	13%	10%	-3%	-3%	**3**
2	0.5	0.5	1	1	11%	25%	4%	4%	**3**
5					13%	67%	20%	13%	**1**
	5				71%	14%	14%	14%	**2**
		5			-20%	-40%	-40%	-40%	**2**
			5		-60%	-10%	-30%	-60%	**1**
				5	61%	17%	22%	17%	**2**
1			2	2	3%	16%	1%	1%	**3**
	2			3	50%	23%	27%	23%	**2**
			2.5	2.5	1%	3%	-4%	-4%	**3**
2			1	2	18%	31%	11%	11%	**3**
2.5				2.5	37%	42%	21%	21%	**3**
3.5				1.5	28%	52%	21%	21%	**3**
	2.5		2.5		-13%	12%	2%	-13%	**1**
2		3			-7%	3%	-16%	-16%	**3**
2.5		2.5			-3%	13%	-10%	-10%	**3**
2.5				2.5	37%	42%	21%	21%	**3**

Table 25 Importance of each factor when choosing a supplier.

These are quantifiable parameters and these are the starting point for the analysis, but there are other non-quantifiable parameters that a company has to consider when evaluating these type of scenarios, namely: plant flexibility to produce a product specification (ability to produce a according to a requirement, artwork, material, innovation etc.), volume flexibility, responsiveness do changes (volumes, specifications, times, etc.), quality standards, traceability, service rate, delivery arrangements sustainable and ethically sourced resources/ materials, communication channels, payment terms, labour relations etc. Depending on the company and product type, some of the non-quantifiable parameters are as important as the price or lead times parameters.

This proposed approach should run separately to the main Supply Chain Model as to save time and make the SC model less busy in terms of data. Only when the developed scoring analysis show that there is not much difference between two scenarios, then both scenarios should be run in the main model in order to quantify the best option in terms of impact on the supply chain (lead times, MOQ, general costs, etc.) and determine the best overall profit scenario (same approach as this chapter previous section).

6.4. SUMMARY

Companies have sought to identify a limited number of “strategic” suppliers with whom they can work as partners through linked systems and processes. While the dangers of single sourcing need to be recognised, the advantages of having a network of key suppliers able to synchronise their production and deliveries with the requirements of the company are considerable (Christopher, 2000). It is not possible to create close relationships through process integration with multiple suppliers, therefore the supplier base needs to be rationalised. In order to rationalise is necessary to identify the suppliers that best fit the business needs, hence the factors that best translate the company’s requirements need to be firstly defined whether they are quantifiable factors (lead times, MOQ, costs) or they are not quantifiable factors but still needing to be considered as part of that analysis (flexibility, quality, service, changes in markets, etc.). Once those factors are identified, scenarios can be set up examining possible futures options, helping an organisation to identify the real options and be prepared by increasing the understanding of potential cost and value proposition of their various constituents.

This study developed a deterministic model based on accounting principles as a tool for evaluating and analysing the different supplier possibilities. A scenario development process is adopted as it offers the possibility of integrating various kinds of data in a consistent manner, and it can represent different supplier benefits and opportunities simultaneously, allowing a like-for-like comparison, before even starting the study of the impact of a supplier in the overall SC profit and costs.

Moreover this chapter was demonstrated the impact of raw material supply shortages on profitability. It has been performed for one product line but it could have been easily extrapolated to different types of raw material, helping to steer the risk management business plan in case of supply shortage of components/ raw material. This case-study becomes extremely interesting as the shortage of one component affects a multitude of SKUs leading to sales losses and the time lag for the supply chain establish equilibrium (going from a chaotic state).

The impact of on the financial viability of supply under different scenarios can also be simulated using the model developed in previous chapters. This chapter proves once more that simulation can improve the data upstream and downstream as it helps greatly to understand SC behaviours and reduce the bullwhip or Forrester effect.

Furthermore other scenarios can be simulated to take this study even further, namely exploring the potential suppliers' flexibility to manage uncertainty and complexity: delay in transportation of the materials from the supplier to the manufacturer, manufacturing related constrains, cancelation of orders, non-conformance of the raw materials, forecasting errors, company growth which can lead to supply issues, etc. Strategies to deal with supply disruptions need to be considered in the supply chain structure so that appropriate actions, risks management and benefits are widely understood and accepted.

7. CONCLUSIONS

In today's rapidly evolving world, companies need to constantly adjust their Business Models to changes in their environment in order to remain competitive and ahead of their competition. The developed work is motivated by the hypothesis that the presented Supply Chain Business Model is a practical and comprehensive approach to support not only operational day-to-day business decisions, but most importantly strategic and long term decisions that may define the success and the longevity of a business.

7.1. SUMMARY & FINDINGS

Conceptually, the Business Supply Chain Model developed in this work replicates the behaviour and decision-making of the different agents in the supply chain, and an optimisation module determines which parameters maximise the overall business profit for different business scenario. In the optimisation module, a Genetic Algorithm was used to determine the best equation parameters for each individual agent which optimises the overall supply chain profit: minimum stock holding for each individual agent but still securing maximum availability, decreasing BBE date losses, and improving forecasting and ordering accuracy.

In the presented work, circa 70% of the total time was invested in building the model. The biggest challenge was on building the model itself as the model was built from scratch in Java to try to mimic all the rules and equations around material, information and economic flows, and translate the behaviour of the overall business and its supply chain. The most interesting-challenging part of the modelling was restructuring the material and information flows to contain information about quantity and BBE dates (array), and finally the implementation of the GA algorithm in Java. Furthermore, in terms of time spent, even though most of the simulations running-time is almost negligible, the GA optimization module running-time is longwinded (circa 6 to 8 hours) every time it is run. Thankfully it was proven that the standard optimisation parameters provide good result levels in terms of the overall SC profitability. Hopefully due to generalist Business Modelling building approach, hopefully the developed methodology will be used in the study of other supply chain networks and many other supply chain scenarios.

This dissertation presents a framework for a SC Business Modelling, which due to its generic and globalising approach, can be easily applied and transposed to any other business realities, which makes this model quite versatile and broad that can be used to analyse different business scenarios similarly to the scenarios in chapters 5 and 6.

As mentioned previously in chapter 3, there are factors that made the presented case-study interesting due to both its complexity and range of applicability to other SC scenarios, namely:

- Special emphasis is given to the integration of SC Financial Flows because it is an interesting measure on how the entire supply chain is performing. The model highlights both the operational and, above all the financial benefits of various levels of supply chain decisions: in actual fact, all operations are quantified whether they are costs, sales, profit, etc., with the aim of overall profit optimisation.
- It is a food supply chain which, in general, is one of the most demanding supply chains, due to the perishable nature of the product, but also due to fluctuations in time, quality and quantity. Very few SC models were able to model the shelf life element when performing modelling so the presented model has managed to address that important factor.

- The majority of the supply chain elements belong to the same company, so it is a vertically integrated business. Being a manufacturer, distributor, wholesaler and retailer, is very unusual, so this scenario makes easier to make global optimisation. Also as it is a multi-channel company, due to the diversity of its customers' profiles there is extra complexity on the decision making. As a result, a holistic analysis is required to enhance overall system performance.
- Similarly to fashion markets, HC has similar challenges to overcome underpinned by the overall company strategy due to an ever changing product offer which guarantees freshness in terms of product range and growing customer interest: shorter life, high volatility, low predictability and high volatility, high level of impulse purchasing, etc., which has an huge impact on the decision-making. Low predictability naturally decreases the ability to forecast sales. With an ever changing product range comes an increasing difficulty to accurately forecast each SKU and hard to determine the impact of new SKUs in the existing proposed range and also on the core product offer (non-seasonal offer). There is also a high investment in product innovation, therefore there is a higher risk which needs to be managed.
- Furthermore, commercial aspects and commercial decisions have sometimes superseded the operational efficiencies, and the SC has been modelled in function of those commercial decisions.
- Scenarios were approached more in terms of profit rather than cost, with the SC reacting the best way to achieve the highest profit possible.
- The last point is one of the most important points from a Brand Positioning perspective: HC has similar challenges to overcome underpinned by the overall company strategy due to an ever changing product offer which guarantees freshness in terms of product range and makes the brand increasingly aspirational for the customer. For this reason the Supply Chain is characterised as a buyer-driven global chain, in which profits derived from a unique combinations of high-value research, design, sales, marketing, and merchandising aligning the offer (in terms of service and product) with market requirements.

- As addressed in chapter 5, the complexity to the supply chain is continuously growing as HC is becoming truly multichannel and multimarket business.

The main findings from the previous chapters can be resumed as follows:

- The HC model is robust when subjected to predictable or unpredictable causes of variability —in all cases, the profit is further improved when optimised using GA optimisation and the determination of new parameters, even though the standard parameters (first optimisation) are also able to generate good results.
- This study has proved that the bullwhip effect can be reduced significantly by applying GA as the optimisation tool (chapter 4) —it has shown that the GA optimisation has the capability to determine the optimal ordering policy for each member of the SC when facing predictable or non-predictable causes of variability (stochastic or random demand and lead times, promotional strategies occurring at any point in the time period).
- The improvement of profits needs to be evaluated at a global scale, meaning overall business, independently of the individual agents' profit maximisation —meaning that for some agents the optimisation may mean that they will achieve lower individual profit. This is an opposite scenario to what happens in reality, where the individual agents try to achieve the maximum profit individually as they belong to different companies.
- Impact of supply shortages in the SC —this model was able to quantify in monetary terms the impact of supply shortages in profitability.
- Retail Expansions Analysis —this model should be used to plan any retail expansion, as this approach will highlight any potential issues.
- Delivery patterns change —can only be considered when the pattern of demand is quite stable or the unpredictability in demand is managed by setting safety stock levels. Reviewed if, for whatever

reason, the business decision is to reduce the number of deliveries, so the ordering method needs to change accordingly.

- Scenario analysis —allows strategic planning and long-term plans, it shows the drivers of change and helps to understand the levers and pulleys to take greater control of their situations and explore new opportunities.
- Impact of sourcing decisions in the SC —this work shows a proposed approach to evaluate the potential impact on manufacturing, if the components were sourced from traditional European markets or from emerging markets like the Far East. Only when the developed scoring analysis shows that there is not much difference between two sourcing scenarios, then both scenarios should be run in the main model in order to quantify the best option in terms of impact on the supply chain (lead times, MOQ, general costs, etc.) and determine the best overall profit scenario.
- Seasonal vs. non-seasonal products —for seasonal products the forecasting technique for each customer-facing agent needs to be different to the current model: need to be push rather than pull-oriented.

7.2. Contribution of Research

Aside from the specific contributions of this work mentioned in previous chapters and more specifically in the what-if scenarios (chapters 5 and 6), which are specific for Hotel Chocolat, this work pretends to be a different take on the processes and methodologies that are followed/ implemented by this company and will hopefully help to steer some of the company future and present decision-making.

The far-reaching contributions of this work achieves with its proposed modelling framework can be summarised as follows:

1. **Costing model** - at any point in time and in the SC network all costs and profits can be easily traced. Most of the other approaches only focus on material flows and information flows, this work focus on cash flows.
2. **Shelf-life of a product is modelled in supply chain**. Besides the quantity parameter, all the material flows have a production date time (normally called batch number) parameter, and each agent is programmed to accept only stock which has batch number above a certain number. The stock that is rejected and product losses due to BBE data are then captured and quantified.
3. In normal circumstances, the simulation is not an optimisation tool, but in this work the optimisation methodology runs parallel to the developed simulation tool. **Optimisation methodology runs parallel to the developed simulation tool**, and optimisation parameters change accordingly to accommodate to a new business scenario.
4. While most approaches focus on optimising the individual agents' profits (e.g. manufacturing efficiency or the transporter efficiency), for the first time the approach was **optimising the overall business profit**, independently of the individual agent profit.
5. **Data Visibility** - due to the fact that the company is vertically integrated, there was access to data in all points of the supply chain and that data was included in model simulation. Normally not possible as the SC agents belong to different companies.
6. The **impact of sampling** is quantified in terms of business profit and impact on the business for the first time ever.
7. The impact, in the overall SC profitability, due to delays in components supply to the manufacturing agent is quantified. A **supplier scenario evaluation** format is presented as a tool to choose the best supplier option and the impact of delays in supply quantified.
8. The **Business Model** has been built implemented in Java, therefore has been **automated**, which means that any changes to the business scenario are easily implemented and the impact of any business reality easily studied.

The image below gives a general overview of the importance of this work from an academic versus business perspective. In general, businesses focus purely on the end result, while academic focus greatly of how the results are achieved and obviously the end result.

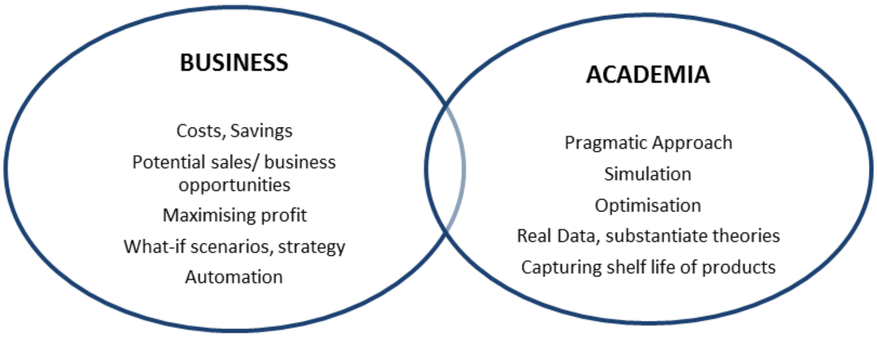

Figure 36 Contributions of this dissertation on business and academia.

This specific model has only been built to model this specific SC network, but it can be easily changed to reflect other SC scenarios, meaning a particular configuration of manufacturers, warehouses, retailers, etc. and it can help to estimate the costs associated in running that network and overall profits; As discussed, the different case studies were performed by making small changes to the main model, which proves the versatility of the built model. This model also can identify which chain dynamics may cause inefficiency and modify the structure to more closely align the individual objectives with global objectives. By tying the supply chain strategy to the overall company strategy, the objectives become process objectives rather than functional objectives. The model is a tool that can be easily modified permitting scenarios analysis, facilitating strategic planning and long-term plans. It shows the drivers of change and helps to understand the levers and pulleys to take greater control of emerging situations and explore new opportunities.

7.3. FUTURE WORK

Even though previous chapters of this thesis pointed to potential further actions, the fundamental work presented in this thesis points to several interesting and important areas for future research namely:

1. By testing the proposed modelling framework in one product example, the potential of the developed methodology and framework has been confirmed. However, further insights into the issues that arise from **modelling different types of product** can be achieved, strengthening the modelling tool. This methodology can be applied to core product lines with other curves of demand, seasonal and non-seasonal.
2. The SC modelling framework has been setup to **be applied simultaneously to multi-products** so that the ordering process and the decision making are automated and are able to react to any random changes in demand. Supply chain optimisation may include refinements at various stages of the product lifecycle, so that new, ongoing and obsolete items are optimised in different ways: for example seasonal merchandise and adaptations for different classes of products.
3. Enhancing the modelling framework will lead to more thorough assessments and to the development of other closely related strings of research. Therefore, future work concerning the modelling of **products with shorter shelf life** (for example four months, instead of six months).
4. Understand the **impact of different lead times in the supply chain** and optimise those timings whether they are for internal supply (similarly to chapter 5) or external (chapter 6).
5. From the factors that affect demand, not all the factors affect the bottom line profit and demand in the same way: for example the weather or an economic decline affects more the demand than for example a price increase, so it would be interesting to determine how those **different reasons**

impact the overall the profit. In the actual economic circumstances, such a study would be very relevant to any business (what-if scenarios).

6. Make the **SC modelling automated and part of the day-to-day decisions**. At the moment, and in the context of this thesis, the different parts of the modelling framework are not linked and the outputs of those simulations are exported to Excel for data analysis. Ideally this process would be automated so that companies could quickly and easily see the course of the dovetailed process chain.
7. In terms of taking the optimisation work further, it would be interesting to apply all of the principles to optimisation using the current GA model, to model the current supply chain under other different scenarios and compare the different GA results and how that impacts on the overall supply chain profit, namely:

- *Manufacturing facility optimisation:* under this scenario, the objective is to minimise the total cost incurred by the manufacturing facility only; the costs experienced by other facilities is ignored.

- *Decentralised optimisation:* this scenario optimises each of the supply chain components individually, and thus minimises the cost experienced by each level.

8. As mentioned previously, this model has only been built to model this specific SC network, but it can be easily changed to reflect other SC scenarios, meaning a particular configuration of manufacturers, warehouses, retailers, etc., so it would be extremely interesting if later research could **apply this methodology to other SC networks** and compare the results and general findings, with the results and conclusions presented in this work.
9. The study has addressed the practices of organisations only one tier upstream and downstream. As mentioned in chapter 6, truly integrated supply chains may consist of multiple organisations in a chain working together to bring the products to customers at the lowest cost possible in the

shortest time, so the **supplier strategy needs to be also incorporated in the HC supply chain** network.

10. There is a need to understand how **future strategies** will unfold and how organisational strategies will merge given different competitive objectives, so different scenarios analysis allows strategic planning and long-term plans. It shows the drivers of change and helps to understand the levers and pulleys to take greater control of emerging situations and explore new opportunities.
11. Explore the **impact of different corporate strategies in the business strategies of the SC chains**, and controlling their performance. The performance control down can be based on standardised reporting systems that focused on life-for-like sales growth, profit margins, overall net profit and return on capital.

In resume, this SC Modelling framework generic and globalising approach means that is easily applied and transposed to any other business realities and it can be easily changed to reflect other SC scenarios. Further case-studies can be easily performed by making small changes to the main model, which proves the versatility of the built model. The costing model associated means that, at any point in the network, all costs and profits can be easily measured. For the first time the shelf-life of a product captured and losses of product due to BBE dates measured. In this model the optimisation methodology runs parallel to the developed simulation tool, so the optimisation should be only run for new scenarios. It is also easy to identify which chain dynamics may cause inefficiency and modify the structure to more closely align the individual objectives with global objectives.

Hopefully the insights identified in this work will help channel research, and that the efforts in Supply chain Business Modelling are both forward-looking and practical. That was one of the main objectives when I first started this work. In closing, I hope there are further opportunities for the application of the developed methodology in future researches on Supply Chain Business Modelling.

BIBLIOGRAPHY

Akanle, O., & Zhang, D. (2008). Agent-based model for optimising supply-chain configurations. *INT J PROD ECON, 115 (2)*, 444-460.

Alfieri, A., & Brandimarte, P. (1997). Object-oriented modelling and simulation of integrated production/distribution systems. *Computer Integrated Manufacturing Systems, Vol 10*, 261-266.

Anderson, D. L., & Lee, H. (2000). *New supply chain business models – The opportunities and challenges.* Retrieved 2010, from http://anderson.ascet.com

Anderson, D., Britt, F., & Favre, D. (1997). The Seven Principles of Supply Chain Management. *Spring issue of Supply Chain Management Review, Supply Chain Management Review*.

Azadeh, A., Layegh, J., & Pourankooh, P. (2010). Optimal Model for Supply Chain System Controlled by Kanban under JIT Philosophy by Integration of Computer Simulation and Genetic Algorithm. *Australian Journal of Basic and Applied Sciences, Vol 4, No 3*, 370-378.

Barnett, M., & Miller, C. (2000). Analysis of the Virtual Enterprise using distributed Supply Chain Modelling and Simulation: an application of e-SCOR. *2000 Winter Simulation Conference*, (pp. 352-355).

Barratt, M., & Oliveira, A. (2001). Exploring the experiences of collaborative planning initiatives. *International Journal of Physical Distribution & Logistics Management, Vol 31, No 4*, 266-289.

Beamon, B. (International Journal of Production Economics, Vol 55). Supply chain design and analysis: models and methods. *1998*, 281-294.

Beamon, B., & Chen, V. (2001). Performance Analysis of Conjoined Supply Chains. *International Journal of Production Research, Vol 39, No 14*, 3195-3218.

Beetz, M., Schumacher, J., Cremers, A., Hellingrath, B., & Mazzocco, C. (2000). Perspectives on Plan-based Multiagent Systems for Distributed Supply Chain Management in the Steel Industry. *ECAI2000 Workshop on Agent Technologies and Their Application Scenarios in Logistics.*

Birtwistle, G., Siddiqui, N., & Fiorito, S. (2003). Quick response: perceptions of UK fashion retailers. *International Journal of Retail & Distribution Management, Vol 31, No 2*, 118 - 128.

Blackburn, J., & Scudder, G. (2009). Supply chain strategies for perishable products: The case of fresh produce. *Production and Operations Management, Vol 18, No 2*, 129-137.

Boehlje, M., Akridge, J., & Downey, D. (1995). Restructuring agribusiness for the 21st century. *Agribusiness, Vol 11, No 6*, 312-325.

Bonney, M., Zhang, Z., Head, M., Tien, C., & Barson, R. (1999). Are push and pull systems really so different? *International Journal of Production Economics, Vol 59, No 1-3*, 53-64.

Bowersox, D. J. (1997). Integrated supply chain management: A strategic imperative. *Council of Logistics Management 1997 Annual Conference.* Chicago, IL.

Brown, D., Garmendia-Doval, A., & McCall, J. (2000, May). A Genetic Algorithm Framework Using Haskell. *Proceedings of the 2nd Asia-Pacific Conference on Genetic Algorithms.* Hong Kong.

Chan, F., & Chan, H. (2005). Simulation modelling for comparative evaluation of supply chain management strategies. *The International Journal of Advanced Manufacturing Technology, Vol 25, No 9-10*, 998-1006.

Chan, F., & Chan, H. (2005). The future trend on system-wide modelling in supply chain studies. *International Journal Advanced Manufacturing Technology, Vol 25*, 820–832.

Chan, F., Tang, N., Lau, H., & Ip, R. (2002). A Simulation Approach in Supply Chain Management. *International Journal of Manufacturing Technology Management, Vol 13, No 2*, 117-122.

Chang, Y., & Makatsoris, H. (2001). Supply chain modelling using simulation. *International Journal of Simulation, Vol 2*, 24-30.

Chen, I., & Paulraj, A. (2004). Towards a theory of supply chain management: the constructs and measurements. *Journal of Operations Management, Vol 22, No 2*, 119–150.

Chopra, S., & Meindl, P. (2001). *Supply Chain Management: Strategy, Planning, and Operations.* Prentice Hall College.

Christopher, M. (1992). *Logistics and Supply Chain Management.* London: Pitman Publishing, U.K.

Christopher, M. (2000). The Agile Supply Chain: Competing in Volatile Markets. *Industrial Marketing Management, Vol 29*, 37–44.

Christopher, M., Lowson, R., & Peck, H. (2004). Creating agile supply chains in the fashion industry. *International Journal of Retail & Distribution Management, Vol. 32, No 8*, 367 – 376.

CSCMP. (2012). *CSCMP, Council of Supply Chain Management Professionals*. Retrieved 2012, from http://cscmp.org/aboutcscmp/definitions.asp

Daniel, J., & Rajendran, C. (2005). A simulation-based genetic algorithm for inventory optimisation in a serial supply chain. *International Transactions in Operational Research, Vol 12, No 1*, 101–127.

de Oliveira, M., de Oliveira, P., & Stauffer, D. (1999). *Evolution, Money, War, and Computers - Non-Traditional Applications of Computational Statistical Physics.* Stuttgart-Leipzig: Teubner.

Ding, H., Benyoucef, L., & Xie, X. (2005). A simulation optimisation methodology for supplier selection problem. *International Journal of Computer Integrated Manufacturing, Vol 18, No 2-3*, 210-224.

Donovan, M. (2007). *Supply Chain Management: Cracking The Bullwhip Effect - Part III.* Retrieved 2011, from http://www.supplychainmarket.com/article.mvc/Supply-Chain-Management-Cracking-The-Cracking-0001

Donselaar, K., Woensel, T., Broekmeulen, R., & Fransoo, J. (2006). Inventory control of perishables in supermarkets. *International Journal Production Economics Vol 104*, 462–472.

Fearne, A., Hughes, D., & Duffy, R. (2001). Concepts of collaboration – supply chain management in a global food industry. In *Food and Drink Supply Chain Management – Issues for the Hospitality and Retail Sectors, Butterworth-Heinemann* (pp. 55-89). Eastham, Sharples& Ball (Eds).

Fernie, J., Sparks, L., & McKinnon, A. C. (2010). Retail logistics in the UK: past, present and future. *International Journal of Retail & Distribution Management, Vol. 38, No 11/12*, 894 – 914.

Ferrolho, A., & Crisóstomo, M. (2005). Genetic Algorithms: concepts, techniques and applications. *WSEAS Transactions on Advances in Engineering Education, Vol 2, No 1*, 12-19.

Fisher, M. (1997). What is the right supply chain for your product? *Harvard Business Review, Vol. 75, No 2*, 105-116.

Fox, M., Barbuceanu, M., & Teigen, R. (2000). Agent-Oriented Supply-Chain Management. *The International Journal of Flexible Manufacturing Systems, Vol 12*, 165–188.

Fox, M., Barbuceanu, M., Beck, C., Davenport, A., & Gruninger, M. (2001). *Managing the Supply Chain – An AI Perspective.* Retrieved 2010, from Enterprise Integration laboratory: http://www.ise.ufl.edu/Supplychain/done/Day2/MarkFox/sld030.htm

Franken, H. M., Bal, R., Berg, H. V., Janssen, W., & Vos., H. (2000). Architectural design support for business process and business network engineering. *International Journal of Services Technology and Management, Vol 1, No 1*, 1-14.

Gadde, L.-E., & Snehota, I. (2000). Making the Most of Supplier Relationships. *Industrial Marketing Management, Vol 29, No 4*, 305-316.

Ganesan, S., George, M., Jap, S., Palmatier, R. W., & B.Weitz. (2009). Supply Chain Management and Retailer Performance: Emerging Trends, Issues, and Implications for Research and Practice. *Journal of Retailing, Vol 85, No 1*, 84-94.

Gjerdrum, J., Shah, N., & Papageorgiou, L. G. (2001). A combined optimization and agent-based approach to supply chain modeling and performance assessment. *Production Planning & Control, 12(1)*, 81-88.

Goldratt. (2012). *The Supply Chain and the Far East: The hidden costs.* Retrieved 2012, from http://www.goldratt.co.uk/articles/SC/fareast.html

Gong, Z. (2008). An economic evaluation model of supply chain flexibility. *European Journal of Operational Research, Vol 184, No 2*, 745–758.

Gonsalves, T., Baba, S., & Itoh, K. (2007). Genetic Algorithm for the Optimisation of Collaborative Systems. *Journal of Advanced Computational Intelligence and Intelligent Informatics, Vol 11, No 7*, 793-802.

Hambuch, P. (2004). Collaborative Planning, Forecasting and Replenishment – Collaboration along the Total Supply Chain. *Business Briefing: global purchasing and supply chain strategies*, pp. 54-63.

Hedman, J., & Kalling, T. (2003). The business model concept: theoretical underpinnings and empirical illustrations. *European Journal of Information Systems, 12*, 49–59.

Holsapple, C., Varghese, J., & Raghav, R. (2001). *Business Modelling, Multidisciplinary Approaches - Economics, Operational and Information.* Dordrecht: Kluwer Academic Publishers.

Jeonga, B., Junga, H.-S., & Park, N.-K. (2002). A computerised causal forecasting system using genetic algorithms in supply chain management. *Journal of Systems and Software, Vol 60, No 3*, 223-237.

Kannan, V., & Tan, K. (2005). Just in time, total quality management, and supply chain management: understanding their linkages and impact on business performance. *Omega, Vol 33, No 2*, 153-162.

Ketzenberg, M., Metters, R., & Vargas, V. (2000). Inventory policy for dense retail outlets. *Journal of Operations Management, Vol 18, No 3*, 303-316.

Kleijnen, J. (2005). Supply chain simulation tools and techniques: a survey. *International Journal of Simulation & Process Modelling, Vol 1, No 1/2*, (pp. 82–89).

Kleijnen, J., & Smits, M. (2003). Performance metrics in supply chain management. *Journal of the Operational Research Society, Vol 0*, 1–8.

Koprulu, A., & Albayrakoglu, M. (2007). Supply Chain Management in the Textile Industry: A Supplier Selection Model with the Analytical Hierarchy Process. *ISHAP 2007.* Chile.

Kubota, F., Sato, S., & Nakano, M. (1999). Enterprise modelling and simulation platform integrating manufacturing system design and supply chain. *IEEE International Conference on Systems, Man, and Cybernetics, Vol 4*, (pp. 511-515).

Labarthe, O., Espinasse, B., Ferrarini, A., & Montreuil, B. (2007). Toward a methodological framework for agent-based modelling and simulation of supply chains in a mass customization context. *Simulation Modelling Practice and Theory, 15 (2)*, 113–136.

Laurikkala, H., & Pajarre, E. (1999). Supply chain management with a strategic modelling and simulation tool for virtual supplies. *APMS*, 284-291.

Laurikkala, H., Vilkman, H., Ek, M., Koivisto, H., & Xiong, G.-Y. (n.d.). *Modelling and Control of Supply Chain with Systems Theory.* Retrieved 2010, from https://ae.tut.fi/projects/systema/julkaisut/Norddesign%20final.pdf

Lee, H. (2004, October). The triple-A supply chain. *Harvard Business Review*, pp. 102–112.

Levi, D., & Levi, E. (2000). *The Effect of e-Business on Supply Chain Strategy.* Retrieved from http://supplychain.mit.edu/innovation/docs/ecomm_final.pdf

Lu, J., Humphreys, P., McIvor, R., & Maguire, L. (2009). Employing Genetic Algorithms to Minimise the Bullwhip Effect in a Supply Chain. *Industrial Engineering and Engineering Management, IEEM 2009*, (pp. 1469–1473).

Lummus, R., & Vokurka, R. (1999). Defining supply chain management: a historical perspective and practical guideline. *Journal: Industrial Management & Data Systems*, 11–17.

Mabert, V., & Venkataramanan, M. (1998). Special research focus on supply chain linkages: challenges for design and management in the 21st century. *Decision Sciences Vol 29, No 3*, 537–552.

Marquez, A. C. (2004). Front-end, back-end and integration issues in virtual supply chain dynamics modelling. *International Journal of Logistics Systems and Management, Vol 1, No 1*, 38 - 63.

Martínez-Olvera, C. (2009). Benefits of using hybrid business models within a supply chain. *Int. J.ProductionEconomics120*, 501–511.

McCullen, P., & Towill, D. R. (2000). Manufacturing Agility and the Lean Supply Chain. *Proc. MIM 2000 Conf*, (pp. 424-431). Aston.

Meixell, M., & Gargeya, V. (2005). Global supply chain design: A literature review and critique. *Transportation Research Part E 41*, 531–550.

Mikkola, J. H., & Skjott-Larsen, T. (2004). Supply-chain integration: implications for mass customisation, modularisation and postponement strategies. *Production Planning & Control, Vol 15, No 4*, 352–361.

Moors, J., & Strijbosch, L. (2002). Exact fill rates for (R, s, S) inventory control with gamma distributed demand. *Journal of the Operational Research Society, Vol 53*, 1268-1274.

Moseklide, E., Larsen, E., & Sterman, J. (1991). Coping with Complexity: Deterministic Chaos in Human Decisionmaking Behavior. In *Beyond Belief: Randomness, Prediction, and Explanation in Science.* Boston: J. L. and A. Karlqvist. (Eds.), CRC Press.

Mullen, T., Sun, S., Levi, M., Bagby, J., Avasarala, V., & Yen, J. (2004). Implications of Agent-based Supply Chain Games. *IEEE SMC 2004, International Conference on Systems, Man and Cybernetics.* The Hague, The Netherlands.

Naylor, J. B., Naim, M. M., & Berry, D. (1999). Leagility: Integrating the lean and agile manufacturing paradigms in the total supply chain. *Internation Journal Production Economics, Vol 62, No 1-2*, 107-118.

O'Donnell, T., Humphreys, P., McIvor, R., & Maguire, L. (2009). Reducing the negative effects of sales promotions in supply chains using genetic algorithms. *Expert Systems with Applications, Vol 36, No 4*, 7827-7837.

O'Donnell, T., Maguire, L., McIvor, R., & Humphreys, P. (2006). Minimising the bullwhip effect in a supply chain using genetic algorithms. *International Journal of Production Research, Vol 44, No 8*, 1523-1543.

Osterwalder, A., & Pigneur, Y. (2002). An e-Business Model Ontology for Modeling e-Business. *Proceedings of the Bled Electronic Commerce Conference, June 17 -19.*

P. Priya, K. (2011). Web based Multi Product Inventory Optimisation using Genetic Algorithm. *International Journal of Computer Applications, Vol 25, No 8*, 23-28.

Papazoglou, M. P., Ribbers, P., & Tsalgatidou, A. (2000). Integrated value chains and their implications from a business and technology standpoint. *Decision Support Systems, Vol 29, No 4*, 323–342.

Parunak, H. V., Savit, R., & Riolo, R. L. (1998). Agent-Based Modelling vs. Equation-Based Modelling: A Case Study and Users' Guide. *Multi-agent systems and Agent-based Simulation (MABS'98)* (pp. 10-25). Springer.

Persson, F., & Olhager, J. (2002). Performance simulation of supply chain designs. *International Journal of Production Economics, Vol 77, No 3*, 231-245.

Pishvaee, M., & Torabi, S. A. (2010). A possibilistic programming approach for closed-loop supply chain network design under uncertainty. *Fuzzy Sets and Systems 161*, 2668 – 2683.

Qiao, G., & Riddick, F. (2004). Modelling Information for Manufacturing Supply Chain Simulations. *Proceedings of the 2004 Winter Simulation Conference*, (pp. 1184-1187).

Radhakrishnan, P., Prasad, V., & Jeyanthi, N. (2010). Predictive Analytics using Genetic Algorithm for Efficient Supply Chain Inventory Optimisation. *IJCSNS International Journal of Computer Science and Network Security, Vol 10 No 3*, 182 -187.

Radhakrishthinan, P., Gopalan, M., & N.Jeyan. (2010). Design of Genetic Algorithm Based Supply Chain Inventory Optimisation with Lead Time. *IJCSNS International Journal of Computer Science and Network Security, Vol 10, No 4*, 238-246.

Rai, A. (2001). Recalibrating Demand Supply Chains for the digital economy. *Netcentricity Symposium* (pp. 30-31). University of Maryland, College Park.

Ramanan, T., Ratnakumar, K., Vasudevan, A., Sarawgi, M., & Raj, M. (2010). Distribution planning in supply chain management. *Industrial Engineering and Engineering Management (IEEM)*, (pp. 1128-1132).

Reeves, C. (2003). Genetic Algorithms. In *Handbook of Metaheuristics.* Boston: F. Glover and G. A. Kochenberger (Eds.), Kluwer Academic Publications.

Reis, C., Paiva, L., Moutinho, J., & Marques, V. M. (2010). Genetic algorithms and sensitivity analysis in production planning optimization. *AIC'10/BEBI'10 Proceedings of the 10th WSEAS international conference on applied informatics and communications, and 3rd WSEAS international conference on Biomedical electronics and biomedical informatics* (pp. 246-251). Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS).

Roth, A., Tsay, A., Pullman, M., & Gray, J. (2008). Unraveling the Food Supply Chain: Strategic Insights from China and the 2007 Recalls. *Journal of Supply Chain Managemen*, 22–40.

Sauer, J., & H.-J.Appelrath. (2003). Scheduling the Supply Chain by Teams of Agents. *Hawai'i International Conference on System Sciences.* Big Island, Hawaii.

Schieritz, N., & Größler, A. (2003). Emergent Structures in Supply Chains - A Study Integrating Agent-Based and System Dynamics Modelling. *36th Hawaii International Conference on System Sciences (HICSS'03), Track3, Vol 3.*

Senouci, A., & El-Rayes, K. (2009). Time-Profit Trade-Off Analysis for Construction Projects. *Journal of Construction Engineering and Management, Vol 135, No 8*, 718-725.

Shah, N. (2004). Pharmaceutical supply chains: key issues and strategies for optimisation. *Computers and Chemical Engineering 28*, 929–941.

Shen, W., & Norrie, D. H. (1998). An Agent-Based Approach for Manufacturing Enterprise Integration and Supply Chain Management. *PROLAMAT '98 Proceedings of the Tenth International IFIP WG5.2/WG5.3 Conference on Globalization of Manufacturing in the Digital Communications Era of the 21st Century: Innovation, Agility, and the Virtual Enterprise*, (pp. 579-590). The Netherlands.

Simchi-Levi, D., Kaminsky, P., & Simchi-Lev, E. (2003). *Designing and managing the supply chain: concepts, strategies, and case studies.* McGraw-Hill/Irwin .

Sourirajana, K., Ozsenb, L., & Uzsoyc, R. (2009). A genetic algorithm for a single product network design model with lead time and safety stock considerations. *European Journal of Operational Research, Vol 197, No 2*, 599-608.

Spekman, R., Kamau, J., & Spear, J. (1999). Towards more effective sourcing and supplier management. *European Journal of Purchasing & Supply Management Vol 5, No 2*, 103-116.

Stauffer, D., Oliveira, S. M., Oliveira, P. M., & Martins, J. S. (2006). *Biology, Sociology, Geology by Computational Physicists.* Amsterdam: Elsevier.

Sterman, J. (1989). Modeling managerial behavior: misperceptions offeedback in a dynamic decision making experiment. *Manage. Sci., 35*(3), 321–339.

Sterman, J. (2000). *Business Dynamics. Systems Thinking and Modeling for a Complex World.* McGraw-Hill.

Stevenson, W. J. (2005). *Operations Management, 8th Edition.* The McGraw-Hill.

Swaminathan, J. M., Smith, S. F., & Sadeh, N. M. (1998). Modelling supply chain dynamics: A multi-agent approach. *Decision Sciences Decision, Vol 29, No 3*, 607-632.

Tang, N. K., Benton, H., Love, D., Albores, P., Ball, P., Mavbryde, J., et al. (2004). Developing an enterprise simulator to support electronic supply-chain management for B2B electronic business. *Production Planning & Control, Vol 15, No 6*, 572–583.

Tempelmeier. (2012, July 10th). *Inventory Management*. Retrieved 2012, from Advanced Planning: http://www.advanced-planning.eu/advancedplanninge-237.htm

Terzi, S., & Cavalieri, S. (2006). Proposal of a performance measurement system for the evaluation of scheduling solutions. *International Journal of Manufacturing Technology and Management, Vol 8, No 1/2/3*, 248 – 263.

Trkman, P., Indihar Štemberger, M., Jaklič, J., & Groznik, A. (2007). Process approach to supply chain integration. *Supply Chain Management - An International Journal, 12(2)*, 116-128.

Truong, T., & Azadivar, F. (2003). Simulation optimisation in manufacturing analysis: simulation based optimisation for supply chain configuration design. *WSC '03 Proceedings of the 35th conference on Winter simulation: driving innovation, Winter Simulation Conference*, (pp. 1268-1275).

Van der Vorst, J. G. (2000). *Effective food supply chains; generating, modelling and evaluating supply chain scenarios.* The Netherlands: Doctoral Dissertation, Department of Management Studies, Wageningen University.

Van der Vorst, J. G. (2000). Modelling and Simulation of Complex Supply Chains. *Proceedings of the Eleventh Annual Conference of the Production and Operations Management Society, POM-2000.* San Antonio, TX.

Van der Vorst, J. G., & Beulens, A. (2002). Identifying sources of uncertainty to generate supply chain redesign strategies. *International Journal of Physical Distribution and Logistics Management, Vol. 32, No. 6*, 409-430.

Van der Vorst, J. G., Tromp, S., & Van der Zee, D. (2005). A simulation environment for the redesign of food supply chain networks: modelling quality controlled logistics. *WSC '05 Proceedings of the 37th conference on Winter Simulation*, (pp. 1658 - 1667).

Van der Zee, D. J., & Van der Vorst, J. G. (2005). A Modelling Framework for Supply Chain Simulation: Opportunities for Improved Decision Making. *Decision Sciences, Vol 36, No 1*, 65-95.

Verdicchio, M., & Colombetti, M. (2002). Commitments for Agent-Based Supply Chain Management. *ACM SIGecom Exchanges, Vol 3, No 1*, 13-23.

Vergara, F., Khouja, M., & Michalewicz, Z. (2002). An evolutionary algorithm for optimising material flow in supply chains. *Journal Computers and Industrial Engineering archive, Vol 43, No 3*, 407 - 421.

Vieira, G. E., & César Júnior, O. (2005). A Conceptual Model for the creation of supply simulation models. *Proceedings of the 2005 Winter Simulation Conference*, (pp. 2619-2627).

Vijayalakshmi, M., Menon, R., Divecha, A., & Ravindran, R. (2010). Intelligent Sales Forecasting Engine Using Genetic Algorithms. *19th ACM conference on Conference on information and knowledge management CIKM 10* (pp. 1669-1672). ACM Press.

Weiss, M., & Amyot, D. (2005). Design and Evolution of e-Business Models. *7th IEEE International Conference on E-Commerce Technology (CEC 2005)*, (pp. 462-466). München, Germany.

Wierenga, B., & Soethoudt, H. (2009). Sales promotions and channel coordination - Original Empirical Research. *Journal of the Academy of Marketing Science, Vol 38, No 3*, 383-397.

Wilson, J., & Miller, R. (1998). Forecasting for the Small Retail Business: Operational Recommendations. *Mid-American Journal of Business, Vol 13, No 2*, 27-34.

Wilson, M. C. (2007). The impact of transportation disruptions on supply chain performance. *Transportation Research Part E: Logistics and Transportation Review, Vol 43, No 4*, 295-320.

Yang, R. (2000). *Supply chain management: Developing visible designing rules across organisations, Handbook on electronic commerce. E-Commerce and Foreign Trade.* New York: Springer-Verlag.

Yang, T., Lu, C., & Li, W. (2004). Study on agile supply chain management. *2004 IEEE Intenational Conference on Systems, Man and Cybernetics, Volume 7*, (pp. 10-13).

Yücela, E., Karaesmena, F., Salman, F., & Türkaya, M. (2009). Optimising product assortment under customer-driven demand substitution. *European Journal of Operational Research, Vol 199, No 3*, 759-768.

Yung, S. K., Yang, C. C., Lau, A. S., & Yen, J. (2000). Applying multi-agent technology to supply chain management. *Journal of Electronic Commerce Research, Vol. 1, No. 4*, 119-132.

Zhu, K., Wang, H., Kong, Y., & Li, S. (2011). Optimisation of the Replenishment Strategy for the Supplier Based on Genetic Algorithm. *International Journal of Business and Management, Vol 6, No 1*, 218-222.

APPENDICES

I - TIME-SERIES DECOMPOSITION (TSD)

The forecasted quantity (F_t) is determined using the Time-Series Decomposition (TSD). The procedure which decomposes a series into a seasonal component is a combined trend and cycle component, and an "error" component. The procedure is an implementation of the ratio-to-moving-average method. It takes the time series of sales data and breaks it into trend (T), seasonal (S), cyclical (C), and irregular (I) components. Each of these components is modelled separately and then put back together to make the forecast.

$$X_t = f(S_t, T_t, C_t, R_t)$$

where X represents the level of sales, S_t the seasonal component at period t, T_t the trend component at time t, C_t = cyclical component at time t, R_t = random component at time t.

The most common time series decomposition model is multiplicative, called the ratio of actual to moving averages, such as:

$$X_t = S_t . T_t . C_t . R_t$$

Multiplicative model is more prevalent with economic series since most seasonal economic series have seasonal variation which increases with the level of the series.

Time-Series Decomposition is effective at picking up patterns in the data. The challenge is to successfully project the patterns through the forecast horizon. This is generally fairly easy for the trend and seasonal patterns but is more difficult for the cyclical pattern.

If the cycle pattern is not important or if it can be projected with confidence, the method can also be used effectively for long-term forecasts. The particular advantage to this method is that it can be used often to identify turning points. Doing so is dependent on one's ability to correctly interpret when the cycle factor may turn up or down (Wilson & Miller, 1998).

<u>Multiplicative Decomposition</u>

$X_t = T_t . C_t . S_t . R_t$ is a demand model where X_t is the actual demand at period t.

First calculate a 12-month centred moving average for periods 7 onwards:

This moving average is equal to the trend × cycle, i.e. $MA_t = T_t . C_t$

For Multiplicative decomposition, the seasonally adjusted data are computed by dividing the original observation by the seasonal component. So, for every month from periods 7 onwards, calculate the seasonality × randomness:

$$S_t \times R_t = X_t / MA_t$$

Now, seasons repeat themselves. Assuming *t* is the month, then we can get an average seasonality for, say, August from:

$$S_8 = (S_8R_8 + S_{20}R_{20} + S_{32}+R_{32}) / 3 \text{ as we are considering 3 years' data}$$

Finally, the randomness is a useful indicator of the accuracy of the fit of the model. It is isolated by $R_t = X_t / (T_t \,.C_t \,.\, S_t)$

<u>Manufacturing Forecasting Model</u>

The series exhibits a number of peaks as per the graph below. This output suggests that the series has a periodic component as well as fluctuations that are not periodic, which is the typical for real-time series.

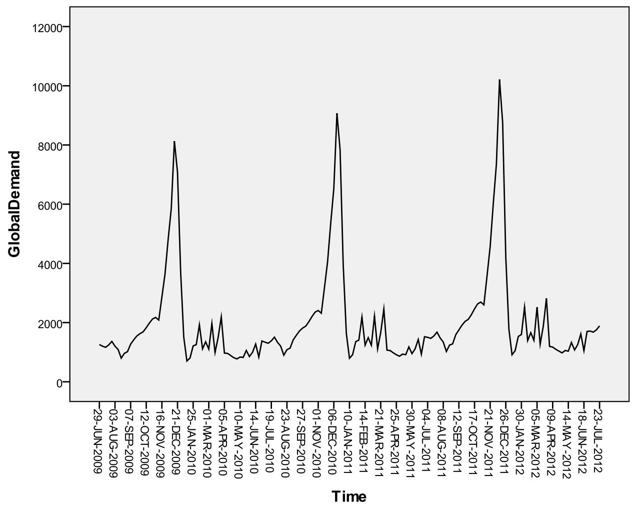

Figure 37 Product Demand for a 3 year window.

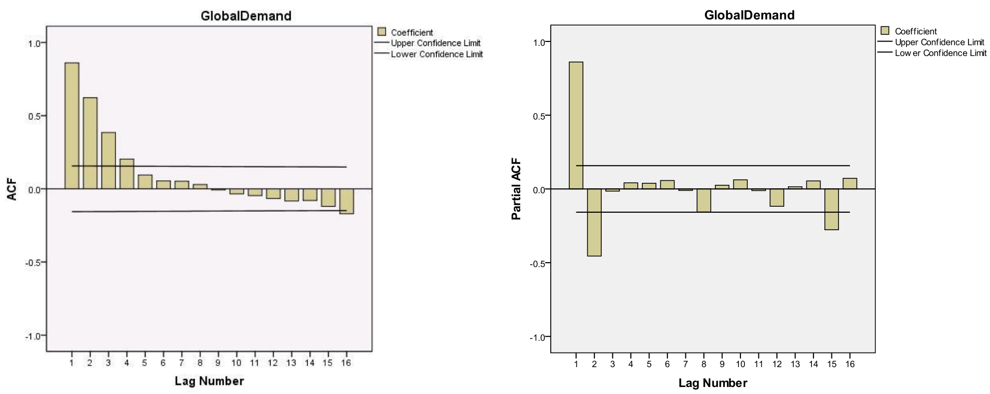

Figure 38 Autocorrelation and partial autocorrelation function time series of the demand.

The autocorrelation function shows a significant peak at a lag of 1 with a long exponential tail—a typical pattern for time series. The significant peak at a lag of 1 and 2 suggests the presence of an annual seasonal component in the data. Examination of the partial autocorrelation function will allow a more definitive conclusion.

Seasonal Factors

Series Name: Global Demand

Period	Seasonal Factor (%)
1	70.6
2	69.0
3	67.1
4	71.1
5	77.2
6	68.0
7	61.5
8	46.2
9	55.7
10	58.0
11	72.1
12	79.5
13	86.7
14	91.9
15	94.8
16	102.1
17	110.7
18	118.2
19	120.9
20	116.4
21	159.7
22	203.4
23	268.2
24	327.2
25	453.9
26	390.5
27	197.0
28	82.0
29	39.3
30	45.0
31	67.1
32	69.5
33	107.2
34	61.1
35	73.9
36	60.7
37	108.5
38	54.4
39	81.7
40	119.9
41	52.4
42	51.8
43	47.9
44	44.1
45	41.7
46	45.0
47	44.1
48	56.5
49	45.4
50	52.8
51	67.1
52	43.5

Table 26 SPSS Seasonal Decomposition Analysis Factors for the manufacturing agent.

II - Forecasting methods to determine future demand

Simple moving average -the forecast F_t is average of *n* previous observations or actuals D_t

$$F_{t+1} = \frac{1}{n}(D_t + D_{t-1} + \cdots + D_{t+1-n})$$

$$F_{t+1} = \frac{1}{n}\sum_{i=t+1-n}^{t} D_i$$

Exponential smoothing - new forecast is weighted sum of old forecast and actual demand, only 2 values (D_t and F_{t-1}) are required, compared with n for moving average. Parameters α are determined empirically. Simple Exponential Smoothing works well with data that is stationary and it needs to be adapted for data series which exhibit a definite trend and further adapted for data series which exhibit seasonal patterns.

$$F_t = aD_t + (1-a)F_{t-1}$$

Holt's Method: Double Exponential Smoothing – the ideas behind smoothing with trend: ``de-trend'' time-series by separating base from trend effects, smooth base in usual manner using α and smooth trend forecasts in usual manner using β.

Smooth the base forecast B_t

$$B_t = \alpha D_t + (1-\alpha)(B_{t-1} + T_{t-1})$$

Smooth the trend forecast T_t

$$T_t = \beta(B_t - B_{t-1}) + (1-\beta)T_{t-1}$$

Forecast k periods into future F_{t+k} with base and trend

$$F_{t+k} = B_t + kT_t$$

To determine the forecast accuracy:

Mean Forecast Error (MFE or Bias): Measures average deviation of forecast from actual

$$MFE = \frac{1}{n}\sum_{t=1}^{n}(D_t - F_t)$$

MFE needs to be as close to zero as possible (minimum bias). A large positive (negative) MFE means that the forecast is undershooting (overshooting) the actual observations.

Mean Absolute Deviation (MAD): Measures average absolute deviation of forecast from actual, meaning the absolute error

$$MAD = \frac{1}{n}\sum_{t=1}^{n}\left|D_t - F_t\right|$$

Positive and negative errors thus do not cancel out (as with MFE). MAD needs to be as small as possible for a good fit.

Mean Absolute Percentage Error (MAPE): Measures absolute error as a percentage of the forecast.

$$MAPE = \frac{100}{n}\sum_{t=1}^{n}\left|\frac{D_t - F_t}{D_t}\right|$$

Same as MAD, except this measures deviation as a percentage of actual data.

Standard Squared Error (MSE): Measures variance of forecast error.

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(D_t - F_t)^2$$

This method measures squared forecast error (error variance) and recognises that large errors are disproportionately more “expensive” than small errors.

	BIAS	MAD	MSE	MAPE
SS	5.48	34.49	1947.67	0.03
AW	-45.54	80.12	17466.30	0.03

Table 27 Forecasting accuracy measures for the Manufacturing planning using time series.

III - SUPPLIER ORDERING PROCESS- INVENTORY POLICIES

According to Tempelmeier (2012), inventory policies differ in two main aspects: the mechanism used to trigger replenishment orders and the decision rule that specifies the determination of the order size. The specific inventory policies are defined through the combination of the decision variables *s* (reorder point), *r* (review interval, order cycle), *q* (order quantity) and *S* (order level) as follows:

(s,q) policy

Under the (s,q) policy, the point in time at which replenishment orders are triggered, depends on the size of the reorder point s, whereas the order quantity q is constant over time. In the ideal (textbook) form of the (s,q) policy, the inventory position is continuously monitored. The inventory position is the sum of the inventory on hand plus the inventory on order minus the outstanding backorders (backlog). The inventory management system (or the inventory manager) acts according to the following decision rule: If at a review instant the inventory position has reached the reorder point s (from above), then launch a replenishment order of size *q*. In reality the inventory is not monitored continuously. In contrast, the replenishment decisions are made in discrete time intervals. In addition, often demand sizes are greater than one unit. Under these conditions, the analysis of the (s, q) policy as presented in many textbooks in false, as the so-called undershoot is neglected. Neglecting the undershoot usually results in significant over-estimation of the service level (under-estimation of the required safety stock).

(r,S) policy

If an (r,S) inventory policy is in effect, the points in time at which replenishment orders are released are determined through the review interval r. The inventory management system proceeds according to the following decision rule: In constant intervals of r periods launch a replenishment order that raises the inventory position to the target order level S. Obviously, the (r, S) policy is an inventory policy with periodic review. The order size at a time of a review depends on the demands and the development of the inventory observed in the preceding periods. If r=1, then this policy is called base-stock policy.

(s,S) policy

Under an (s,S) inventory policy, the points in time when an order is triggered are determined by the policy, i.e. through the reorder point s. However, the order quantity is now, similar to the (r, S) policy, a function of the inventory development over time. In the literature this policy is sometimes characterised with the help of a third parameter which specifies the length of the review interval r. In this notation the policy is

called (r, s, S) policy. In the case of r=0, continuous review is in effect. If demands arrive unit-sized, then the (r=0, s, S) policy is identical to the (s, q) policy with continuous review.
For the determination of the optimum safety stock under conditions of uncertainty the demand during the risk period plays a central role. The risk period is composed of the review period and the replenishment lead time.

Stochastic demand occurs within this time span that usually comprises several periods. In order to compute the parameters of an inventory policy, we must know the probability distribution of the demand during the risk period.

IV – HC Supply Chain Simulation Code in Java

```
package hotelchocolatmodeling;

import hotelchocolatmodeling.ShipmentData;

import java.io.FileOutputStream;
import java.io.PrintStream;
import java.util.ArrayList;

//CONSIDERING BBE LOSSES

public class SimulationBBE {

/** define the different arrays */
// orders simply an array (quantity per tick)
//shipment an array within an array (different quantities with different BBE dates, per each tick)

/** RETAIL CUSTOMER AND RETAILER */
ArrayList<Integer> demandRetailCustomer = new ArrayList<Integer> ();// retail customer demand
ArrayList<Integer> expectedOrderRetailer = new ArrayList<Integer> ();//expected order from the retailer to distribution
ArrayList<Integer> newOrderRetailer = new ArrayList<Integer>();//new order from the retailer to distribution
ArrayList<Integer> ordersInPipelineRetailer = new ArrayList<Integer> ();//orders in pipeline from the retailer to distribution
ArrayList<ArrayList<ShipmentData>> supplyRetailCustomer = new ArrayList<ArrayList<ShipmentData>>();//retailer fulfills the retail customer demand
ArrayList<Integer> backordersRetailer = new ArrayList<Integer> ();//retailer backorders
ArrayList<ArrayList<ShipmentData>> stockRetailer = new ArrayList<ArrayList<ShipmentData>> ();//retailer stock
ArrayList<ArrayList<ShipmentData>> receiveShipmentRetailer = new ArrayList<ArrayList<ShipmentData>>();//retailer to receive shipment from distibution
ArrayList<Integer> stockBBELossesRetailer = new ArrayList<Integer>();//BBE stock losses in the retailer

/** MAIL ORDER (CUSTOMER)*/
ArrayList<Integer> demandMailOrderCustomer = new ArrayList<Integer> ();// mail order customer demand
ArrayList<Integer> expectedOrderMailOrder = new ArrayList<Integer> ();//expected order from the mail order to distribution
ArrayList<Integer> newOrderMailOrder = new ArrayList<Integer> ();//new order from the mail order to distribution
ArrayList<Integer> ordersInPipelineMailOrder = new ArrayList<Integer> ();//orders in pipeline from the retailer to distribution
ArrayList<ArrayList<ShipmentData>> supplyMailOrderCustomer = new ArrayList<ArrayList<ShipmentData>>();//retailer fulfills the mail order customer demand
ArrayList<Integer> backordersMailOrder = new ArrayList<Integer> ();//retailer backorders
ArrayList<ArrayList<ShipmentData>> stockMailOrder = new ArrayList<ArrayList<ShipmentData>>();//mail order stock
ArrayList<ArrayList<ShipmentData>> receiveShipmentMailOrder = new ArrayList<ArrayList<ShipmentData>>();//mail order to receive shipment from distibution
ArrayList<Integer> stockBBELossesMailOrder = new ArrayList<Integer>();//BBE stock losses in the mail order

/** HOTEL CHOCOLAT US */
ArrayList<Integer> demandUSCustomer = new ArrayList<Integer> ();// US customer demand
ArrayList<Integer> expectedUSOrder = new ArrayList<Integer> ();//expected order from US to distribution
ArrayList<Integer> newUSOrder = new ArrayList<Integer> ();//new order from US to distribution
ArrayList<Integer> ordersInPipelineUS = new ArrayList<Integer> ();//orders in pipeline from the US to distribution
ArrayList<ArrayList<ShipmentData>> supplyUSCustomer = new ArrayList<ArrayList<ShipmentData>>();//US fulfills the US customer demand
ArrayList<Integer> backordersUS = new ArrayList<Integer>();//US backorders
ArrayList<ArrayList<ShipmentData>> stockUS = new ArrayList<ArrayList<ShipmentData>>();//US stock
ArrayList<ArrayList<ShipmentData>> receiveShipmentUS = new ArrayList<ArrayList<ShipmentData>>();//US to receive shipment from distibution
ArrayList<Integer> stockBBELossesUS = new ArrayList<Integer>();//BBE stock losses in US

/** WHOLESALER UK */
ArrayList<Integer> demandWholesale = new ArrayList<Integer> ();// Wholesale customer demand
ArrayList<Integer> expectedWholesaleOrder = new ArrayList<Integer> ();//expected order from Wholesale to distribution
ArrayList<Integer> newWholesaleOrder = new ArrayList<Integer> ();//new order from Wholesale to distribution
ArrayList<Integer> ordersInPipelineWholesale = new ArrayList<Integer> ();//orders in pipeline from the Wholesale to distribution
ArrayList<ArrayList<ShipmentData>> supplyWholesale = new ArrayList<ArrayList<ShipmentData>>();//Wholesale fulfills the Wholesaler demand
```

```
ArrayList<Integer> backordersWholesale = new ArrayList<Integer>();//Wholesale backorders
ArrayList<ArrayList<ShipmentData>> stockWholesale = new ArrayList<ArrayList<ShipmentData>>();//Wholesale stock
ArrayList<ArrayList<ShipmentData>> receiveShipmentWholesale = new ArrayList<ArrayList<ShipmentData>>();//Wholesale to receive shipment from
distibution
ArrayList<Integer> stockBBELossesWholesale = new ArrayList<Integer>();//BBE stock losses in wholesale

/** MIDDLE EAST FRANCHASING */
ArrayList<Integer> demandMiddleEast = new ArrayList<Integer> ();// MiddleEast customer demand
ArrayList<Integer> expectedMiddleEastOrder = new ArrayList<Integer> ();//expected order from MiddleEast to distribution
ArrayList<Integer> newMiddleEastOrder = new ArrayList<Integer> ();//new order from MiddleEast to distribution
ArrayList<Integer> ordersInPipelineMiddleEast = new ArrayList<Integer> ();//orders in pipeline from the MiddleEast to distribution
ArrayList<ArrayList<ShipmentData>> supplyMiddleEast = new ArrayList<ArrayList<ShipmentData>>();//MiddleEast fulfills the MiddleEast customer
demand
ArrayList<Integer> backordersMiddleEast = new ArrayList<Integer>();//MiddleEast backorders
ArrayList<ArrayList<ShipmentData>> stockMiddleEast = new ArrayList<ArrayList<ShipmentData>>();//MiddleEast stock
ArrayList<ArrayList<ShipmentData>> receiveShipmentMiddleEast = new ArrayList<ArrayList<ShipmentData>>();//MiddleEast to receive shipment from
distibution
ArrayList<Integer> stockBBELossesMiddleEast = new ArrayList<Integer>();//BBE stock losses in MiddleEast

/** DISTRIBUTION */
//the disribution agent does not have demand, just receives what the production has produced and fulfills the coming orders
ArrayList<Integer> demandDistribution = new ArrayList<Integer> ();// sum of the total demand from downstream
ArrayList<ArrayList<ShipmentData>> supplyDistributionRetailer = new ArrayList<ArrayList<ShipmentData>>();//Distribution fulfilment to retailer
ArrayList<ArrayList<ShipmentData>> supplyDistributionMailOrder = new ArrayList<ArrayList<ShipmentData>>();//Distribution fulfilment to mail order
ArrayList<ArrayList<ShipmentData>> supplyDistributionUS = new ArrayList<ArrayList<ShipmentData>>();//Distribution fulfilment to US
ArrayList<ArrayList<ShipmentData>> supplyDistributionWholesale = new ArrayList<ArrayList<ShipmentData>>();//Distribution fulfilment to wholesale UK
ArrayList<ArrayList<ShipmentData>> supplyDistributionMiddleEast = new ArrayList<ArrayList<ShipmentData>>();//Distribution fulfilment to middle east
franchasing
ArrayList<Integer> backordersDistribution = new ArrayList<Integer>();//Distribution backorders
ArrayList<Integer> backordersDistributionRetailer = new ArrayList<Integer>();//Distribution backorders to retailer
ArrayList<Integer> backordersDistributionMailOrder = new ArrayList<Integer>();//Distribution backorders to mail order
ArrayList<Integer> backordersDistributionUS = new ArrayList<Integer>();//Distribution backorders to US
ArrayList<Integer> backordersDistributionWholesale = new ArrayList<Integer>();//Distribution backorders to wholesale UK
ArrayList<Integer> backordersDistributionMiddleEast = new ArrayList<Integer>();//Distribution backorders to middle east franchasing
ArrayList<ArrayList<ShipmentData>> stockDistribution = new ArrayList<ArrayList<ShipmentData>>();//Distribution stock
ArrayList<ArrayList<ShipmentData>> receiveShipmentDistribution = new ArrayList<ArrayList<ShipmentData>>();//Distribution to receive shipment from
manufacturing
ArrayList<Integer> stockBBELossesDistribution = new ArrayList<Integer>();//BBE stock losses in distribution

/** MANUFACTURING */
ArrayList<Integer> forecastingManufacturing = new ArrayList<Integer> ();//forecasting production requirements
ArrayList<ArrayList<ShipmentData>> stockManufacturing = new ArrayList<ArrayList<ShipmentData>>();//Manufacturing stock
ArrayList<ArrayList<ShipmentData>> productionManufacturing = new ArrayList<ArrayList<ShipmentData>>();//Manufacturing production

//defining real demand - this is from i = 4 till 156
int
mailOrderCustomerRealDemand[]={661,362,315,395,521,474,487,160,106,269,349,480,542,362,437,531,557,588,640,529,764,865,1137,1387,2044,202
0,3063,1010,56,35,306,340,383,443,597,374,290,291,652,909,448,484,376,208,216,238,291,353,263,266,346,276,694,380,331,415,547,498,511,168,11
1,282,366,504,569,380,459,558,585,617,672,555,802,908,1194,1456,2146,2121,3216,1061,59,37,321,357,402,465,627,393,305,306,685,954,470,508,39
5,218,227,250,306,371,276,279,363,290,729,399,348,436,574,523,537,176,117,296,384,529,597,399,482,586,614,648,706,583,842,953,1254,1529,2253
,2227,3377,1114,62,39,337,375,422,488,658,413,320,321,719,1002,494,533,415,229,238,263,321,390,290,293,381,305};
int
retailCustomerRealDemand[]={466,661,663,701,704,531,461,482,678,551,691,658,670,842,792,790,817,866,766,686,1066,1536,2223,2803,4197,3007,3
40,305,454,618,774,775,1190,558,626,583,1383,532,654,923,367,378,403,479,431,468,400,592,428,509,789,415,527,747,749,792,796,600,521,545,766
,623,781,744,757,951,895,893,923,979,866,775,1205,1736,2512,3167,4743,3398,384,345,513,698,875,876,1345,631,707,659,1563,601,739,1043,415,4
27,455,541,487,529,452,669,484,575,892,469,596,844,846,895,899,678,589,616,866,704,883,841,855,1075,1011,1009,1043,1106,979,876,1362,1962,2
839,3579,5360,3840,434,390,580,789,989,990,1520,713,799,745,1766,679,835,1179,469,483,514,611,550,598,511,756,547,650,1008,530};
int
usRealDemand[]={5,6,6,5,5,6,6,6,6,6,6,5,10,9,11,9,10,10,12,10,14,13,15,22,32,26,6,6,7,7,7,11,30,9,7,10,7,10,13,28,6,5,5,6,5,5,6,6,6,6,8,6,10,12,12,10,10,
12,12,12,12,12,12,10,20,18,22,18,20,20,24,20,28,26,30,44,64,52,12,12,14,14,14,22,60,18,14,20,14,20,26,56,12,10,10,12,10,10,12,12,12,12,16,12,20,24,
24,20,20,24,24,24,24,24,24,20,40,36,44,36,40,40,48,40,56,52,60,88,128,104,24,24,28,28,28,44,120,36,28,40,28,40,52,112,24,20,20,24,20,20,24,24,24,2
4,32,24};
```

```
int
wholesaleRealDemand[]={120,165,171,136,127,184,129,142,161,183,216,266,316,403,436,486,586,642,741,840,988,1186,1383,1581,1778,1976,296,18
2,177,135,116,119,297,92,112,126,266,144,161,287,137,84,95,112,113,116,117,97,147,208,119,130,132,181,189,150,139,202,142,156,177,201,238,293
,348,443,480,535,645,706,815,924,1087,1305,1521,1739,1956,2174,326,200,195,149,128,131,327,102,123,139,293,158,177,315,151,92,105,124,124,12
8,129,107,161,229,131,143,145,199,208,165,153,222,156,172,195,221,262,322,383,487,528,589,710,777,897,1016,1196,1436,1673,1913,2152,2391,35
9,220,215,164,141,144,360,112,135,153,322,174,195,347,166,101,116,136,136,141,142,118,177,252,144,157};
int
middleEastRealDemand[]={8,9,10,8,8,9,9,9,9,9,9,10,10,11,11,12,14,15,16,21,29,39,47,62,80,47,8,6,7,8,9,11,25,11,10,12,21,13,20,45,10,9,8,9,8,9,9,10,9,
9,13,9,16,18,20,16,16,18,18,18,18,18,18,20,20,22,22,24,28,30,32,42,58,78,94,124,160,94,16,12,14,16,18,22,50,22,20,24,42,26,40,90,20,18,16,18,16,18,
18,20,18,18,26,18,32,36,40,32,32,36,36,36,36,36,36,40,40,44,44,48,56,60,64,84,116,156,188,248,320,188,32,24,28,32,36,44,100,44,40,48,84,52,80,180,
40,36,32,36,32,36,36,40,36,36,52,36};
// forecasted values for the production at the manufacurer - these were done in excel - if i had more time this calculation could have been done in java
//SEASONAL TDS
//PINK
//int
forecastingTimeSeriesManufacturing[]={1189,1160,1133,1208,1322,1172,1065,800,969,1018,1272,1417,1553,1651,1716,1859,2026,2174,2235,2159,29
75,3807,5044,6181,8616,7466,3766,1381,681,786,1170,1222,1911,1084,1310,1093,1970,994,1497,2226,967,959,896,839,797,867,854,1102,894,1050,1
354,888,1323,1290,1259,1342,1468,1301,1182,888,1074,1129,1410,1569,1720,1827,1898,2056,2240,2402,2469,2384,3284,4200,5563,6816,9497,8226,
4148,1553,765,883,1313,1371,2143,1215,1468,1224,2204,1112,1674,2487,1080,1070,1000,936,888,966,951,1227,994,1168,1505,986,1456,1420,1386,
1477,1614,1430,1299,975,1180,1239,1547,1722,1886,2004,2080,2253,2453,2630,2703,2609,3593,4594,6083,7450,10378,8987,4531,1725,849,980,145
7,1520,2375,1346,1625,1355,2439,1229,1850,2749,1193,1182,1104,1032,980,1065,1048,1352,1095,1286,1656,1085,0,0};
//pink but with no manufacturing due to manufacturing breakages
int
forecastingTimeSeriesManufacturing[]={1189,1160,1133,1208,1322,1172,1065,800,969,1018,1272,1417,1553,1651,1716,1859,2026,2174,2235,2145,0,1
898,5044,6181,8616,7466,3766,1381,681,786,1170,1222,1911,1084,1310,1093,1970,994,1497,2226,967,959,896,839,797,867,854,1102,894,1050,1354,
888,1323,1290,1259,1342,1468,1301,1182,888,1074,1129,1410,1569,1720,1827,1898,2056,2240,2402,2469,2384,3284,4200,5563,6816,9497,8226,414
8,1553,765,883,1313,1371,2143,1215,1468,1224,2204,1112,1674,2487,1080,1070,1000,936,888,966,951,1227,994,1168,1505,986,1456,1420,1386,147
7,1614,1430,1299,975,1180,1239,1547,1722,1886,2004,2080,2253,2453,2630,2703,2609,3593,4594,6083,7450,10378,8987,4531,1725,849,980,1457,15
20,2375,1346,1625,1355,2439,1229,1850,2749,1193,1182,1104,1032,980,1065,1048,1352,1095,1286,1656,1085,0,0};

//define the different variables
protected double alpha_Retailer; protected double beta_Retailer; protected double theta_Retailer; protected double q_Retailer;
protected double alpha_HCmailorder; protected double beta_HCmailorder; protected double theta_HCmailorder; protected double q_HCmailorder;
protected double alpha_USHotelChocolat;protected double beta_USHotelChocolat; protected double theta_USHotelChocolat; protected double
q_USHotelChocolat;
protected double alpha_WholesaleUK; protected double beta_WholesaleUK;protected double theta_WholesaleUK;protected double q_WholesaleUK;
protected double alpha_MiddleEastFranchising; protected double beta_MiddleEastFranchising;protected double theta_MiddleEastFranchising;protected
double q_MiddleEastFranchising;
int minimumShelfLife = 25; //25 ticks - 25 weeks from production (6 months)
//int minimumShelfLife = 1000; //testing the program


// PRIORITIES
public int priorityRetailer = 5;
public int priorityMailOrder = 4;
public int priorityUS = 3;
public int priorityWholesalerUK = 2;
public int priorityMiddleEastFranchsing = 1;


public void setParameters(double alphaR, double betaR, double thetaR, double qR, double alphaMO, double betaMO, double thetaMO, double qMO,
double alphaUS, double betaUS, double thetaUS, double qUS, double alphaW, double betaW, double thetaW, double qW,
double alphaME, double betaME, double thetaME, double qME){


// FIXED CONSTANTS
// retailer constantes
alpha_Retailer = alphaR; beta_Retailer = betaR; theta_Retailer = thetaR; q_Retailer = qR;

// mail order constantes
alpha_HCmailorder = alphaMO; beta_HCmailorder = betaMO; theta_HCmailorder = thetaMO; q_HCmailorder = qMO;
```

```
// US constantes
alpha_USHotelChocolat = alphaUS; beta_USHotelChocolat = betaUS; theta_USHotelChocolat = thetaUS; q_USHotelChocolat = qUS;

//WholesalerUK constantes
alpha_WholesaleUK = alphaW;beta_WholesaleUK = betaW; theta_WholesaleUK = thetaW; q_WholesaleUK = qW;

// MiddleEastFranchising constantes
alpha_MiddleEastFranchising = alphaME; beta_MiddleEastFranchising = betaME; theta_MiddleEastFranchising = thetaME; q_MiddleEastFranchising =
qME;

};


public double calculateProfit(boolean outputData){


//for tick below 4

for (int i=0;i<4;i++){

/** RETAIL CUSTOMER AND RETAILER */
demandRetailCustomer.add(0);
demandMailOrderCustomer.add(0);
demandUSCustomer.add(0);
demandWholesale.add(0);
demandMiddleEast.add(0);
expectedOrderRetailer.add(0);
newOrderRetailer.add(0);//new order from the retailer to distribution
ordersInPipelineRetailer.add(0);//orders in pipeline from the retailer to distribution
supplyRetailCustomer.add(new ArrayList<ShipmentData>());//retailer fulfills the retail customer demand
backordersRetailer.add(0);//retailer backorders
receiveShipmentRetailer.add(new ArrayList<ShipmentData>());//retailer to receive shipment from distibution
stockBBELossesRetailer.add(0);//BBE stock losses in the retailer

/** MAIL ORDER (CUSTOMER)*/
expectedOrderMailOrder.add(0);//expected order from the mail order to distribution
newOrderMailOrder.add(0);//new order from the mail order to distribution
ordersInPipelineMailOrder.add(0);//orders in pipeline from the retailer to distribution
supplyMailOrderCustomer.add(new ArrayList<ShipmentData>());//retailer fulfills the mail order customer demand
backordersMailOrder.add(0);//retailer backorders
receiveShipmentMailOrder.add(new ArrayList<ShipmentData>());//mail order to receive shipment from distibution
stockBBELossesMailOrder.add(0);//BBE stock losses in the mail order

/** HOTEL CHOCOLAT US */
expectedUSOrder.add(0);//expected order from US to distribution
newUSOrder.add(0);//new order from US to distribution
ordersInPipelineUS.add(0);//orders in pipeline from the US to distribution
supplyUSCustomer.add(new ArrayList<ShipmentData>());//US fulfills the US customer demand
backordersUS.add(0);//US backorders
receiveShipmentUS.add(new ArrayList<ShipmentData>());//US to receive shipment from distibution
stockBBELossesUS.add(0);//BBE stock losses in US

/** WHOLESALER UK */
expectedWholesaleOrder.add(0);//expected order from Wholesale to distribution
newWholesaleOrder.add(0);//new order from Wholesale to distribution
ordersInPipelineWholesale.add(0);//orders in pipeline from the Wholesale to distribution
supplyWholesale.add(new ArrayList<ShipmentData>());//Wholesale fulfills the Wholesaler demand
backordersWholesale.add(0);//Wholesale backorders
receiveShipmentWholesale.add(new ArrayList<ShipmentData>());//Wholesale to receive shipment from distibution
stockBBELossesWholesale.add(0);//BBE stock losses in Wholesale

/** MIDDLE EAST FRANCHASING */
```

```
expectedMiddleEastOrder.add(0);//expected order from MiddleEast to distribution
newMiddleEastOrder.add(0);//new order from MiddleEast to distribution
ordersInPipelineMiddleEast.add(0);//orders in pipeline from the MiddleEast to distribution
supplyMiddleEast.add(new ArrayList<ShipmentData>());//MiddleEast fulfills the MiddleEast customer demand
backordersMiddleEast.add(0);//MiddleEast backorders
receiveShipmentMiddleEast.add(new ArrayList<ShipmentData>());//MiddleEast to receive shipment from distibution
stockBBELossesMiddleEast.add(0);//BBE stock losses in MiddleEast

/** DISTRIBUTION */
demandDistribution.add(0);
supplyDistributionRetailer.add(new ArrayList<ShipmentData>());//Distribution fulfilment to retailer
supplyDistributionMailOrder.add(new ArrayList<ShipmentData>());//Distribution fulfilment to mail order
supplyDistributionUS.add(new ArrayList<ShipmentData>());//Distribution fulfilment to US
supplyDistributionWholesale.add(new ArrayList<ShipmentData>());//Distribution fulfilment to wholesale UK
supplyDistributionMiddleEast.add(new ArrayList<ShipmentData>());//Distribution fulfilment to middle east franchasing
backordersDistribution.add(0);//Distribution backorders
backordersDistributionRetailer.add(0);//Distribution backorders to retailer
backordersDistributionMailOrder.add(0);//Distribution backorders to mail order
backordersDistributionUS.add(0);//Distribution backorders to US
backordersDistributionWholesale.add(0);//Distribution backorders to wholesale UK
backordersDistributionMiddleEast.add(0);//Distribution backorders to middle east franchasing
receiveShipmentDistribution.add(new ArrayList<ShipmentData>());//Distribution to receive shipment from distibution
stockBBELossesDistribution.add(0);//BBE stock losses in Distribution

/** MANUFACTURING */
forecastingManufacturing.add(0);//forecasting production requirements
stockManufacturing.add(new ArrayList<ShipmentData>());//Manufacturing stock
}



//create a local array which is an array of shipment data
ArrayList<ShipmentData> temp;

//START THE INPUT IN THE SYSTEM
for (int i=0; i<retailCustomerRealDemand.length;i++)
{
demandRetailCustomer.add(retailCustomerRealDemand[i]);
demandMailOrderCustomer.add(mailOrderCustomerRealDemand[i]);
demandUSCustomer.add(usRealDemand[i]);
demandWholesale.add(wholesaleRealDemand[i]);
demandMiddleEast.add(middleEastRealDemand[i]);
forecastingManufacturing.add(forecastingTimeSeriesManufacturing[i]);
}

forecastingManufacturing.add(0);


//define initial parameters for tick = 3
demandRetailCustomer.set(3, 0);
demandMailOrderCustomer.set(3, 0);// mail order customer demand
demandUSCustomer.set(3, 0);// US customer demand
demandMiddleEast.set(3, 0);// MiddleEast customer demand
demandWholesale.set(3, 0);// Wholesaler demand

stockRetailer.add(new ArrayList<ShipmentData>());//retailer stock i=0
stockRetailer.add(new ArrayList<ShipmentData>());//retailer stock i=1
stockRetailer.add(new ArrayList<ShipmentData>());//retailer stock i=2
temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData(3 + minimumShelfLife,2500));//retailer stock i=3
stockRetailer.add(temp);
```

```
stockMailOrder.add(new ArrayList<ShipmentData>());// stock i=0
stockMailOrder.add(new ArrayList<ShipmentData>());// stock i=1
stockMailOrder.add(new ArrayList<ShipmentData>());// stock i=2
temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData(3 + minimumShelfLife,1350));
stockMailOrder.add(temp);

stockUS.add(new ArrayList<ShipmentData>());// stock i=0
stockUS.add(new ArrayList<ShipmentData>());// stock i=1
stockUS.add(new ArrayList<ShipmentData>());// stock i=2
temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData(3 + minimumShelfLife,60));
stockUS.add(temp);

stockWholesale.add(new ArrayList<ShipmentData>());// stock i=0
stockWholesale.add(new ArrayList<ShipmentData>());// stock i=1
stockWholesale.add(new ArrayList<ShipmentData>());// stock i=2
temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData(3 + minimumShelfLife,600));
stockWholesale.add(temp);

stockMiddleEast.add(new ArrayList<ShipmentData>());// stock i=0
stockMiddleEast.add(new ArrayList<ShipmentData>());// stock i=1
stockMiddleEast.add(new ArrayList<ShipmentData>());// stock i=2
temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData(3 + minimumShelfLife,90));
stockMiddleEast.add(temp);

stockDistribution.add(new ArrayList<ShipmentData>());//Distribution stock i=0
stockDistribution.add(new ArrayList<ShipmentData>());//Distribution stock i=1
stockDistribution.add(new ArrayList<ShipmentData>());//Distribution stock i=2
stockDistribution.add(new ArrayList<ShipmentData>());//Distribution stock i=3


//THE FOLLOWING PIECE OF CODE MAKES SURE THAT THE MANUFACTURER STARTS TO PRODUCE 2 TICK BEFORE THE OTHER MEMBERS START TO OPERATE

//Manufacturing production
productionManufacturing.add(new ArrayList<ShipmentData>());//i=0
productionManufacturing.add(new ArrayList<ShipmentData>());//i=1
productionManufacturing.add(new ArrayList<ShipmentData>());//i=2


// ******************************************** need to populate the arraylists in the next two lines

temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData(3 + minimumShelfLife,forecastingTimeSeriesManufacturing[0])); //i=3
productionManufacturing.add(temp);

//previously was:
//productionManufacturing.add(new ShipmentData(0 + minimumShelfLife, 0));
//productionManufacturing.add(new ShipmentData(1 + minimumShelfLife, 0));
//productionManufacturing.add(new ShipmentData(, forecastingTimeSeriesManufacturing[0]));
//productionManufacturing.add(new ShipmentData(, forecastingTimeSeriesManufacturing[1]));

//Loop
```

```
for (int i=4; i < demandRetailCustomer.size(); i++) {


/******************** RETAIL CUSTOMER AND RETAILER ***********************************/

//Expected Order (t) = theta * Incoming Order (t-1) + (1 - theta) * Expected Order (t-1) - there is only quantity in this array
expectedOrderRetailer.add((int) java.lang.Math.round((theta_Retailer * demandRetailCustomer.get(i-1) + (1 - theta_Retailer) * expectedOrderRetailer.get(i-1))));

//ordersInPipeline(t) = ordersInPipeline(t-1) - ReceiveShipment(t-1) + New Order(t-1)
int sumReceiveShipmentRetailer = 0;
for(int j=0;j<receiveShipmentRetailer.get(i-1).size();j++){//for(int j=0;j<receiveShipmentRetailer.get(i-1).size();j++){
sumReceiveShipmentRetailer += receiveShipmentRetailer.get(i-1).get(j).quantity;
}
ordersInPipelineRetailer.add( (int)java.lang.Math.round (ordersInPipelineRetailer.get(i-1) - sumReceiveShipmentRetailer + newOrderRetailer.get(i-1)));


//Review the BBE losses in current array
temp = new ArrayList<ShipmentData>();
for(int m=0; m<stockRetailer.get(i-1).size(); m++){
temp.add( new ShipmentData(stockRetailer.get(i-1).get(m).BBEdate,stockRetailer.get(i-1).get(m).quantity));
}
//BBE losses
int sumBBElossesRetailer = 0;
for(int m=0;m<temp.size();m++){
if(i >= temp.get(m).BBEdate){
sumBBElossesRetailer += temp.get(m).quantity;
temp.remove(m);
m--;
}
}
stockBBELossesRetailer.add((int)sumBBElossesRetailer);


//1 week lead time for distribution to supply the retail stores - supply from distribution to retailer - BBE data + quantity in this array
ArrayList<ShipmentData> temp2 = new ArrayList<ShipmentData>();
for(int k=0;k < supplyDistributionRetailer.get(i-1).size(); k++) {
temp2.add(new ShipmentData(supplyDistributionRetailer.get(i-1).get(k).BBEdate, supplyDistributionRetailer.get(i-1).get(k).quantity)); // 1 week lead time from distribution to retailer
}
receiveShipmentRetailer.add(temp2);


// add the shipment to stock
for(int k=0;k < receiveShipmentRetailer.get(i).size(); k++) {
temp.add(new ShipmentData(receiveShipmentRetailer.get(i).get(k).BBEdate, receiveShipmentRetailer.get(i).get(k).quantity));
}
stockRetailer.add(temp);


int sumSuppliedLastTimeRetailer = 0;
for(int j=0;j<supplyRetailCustomer.get(i-1).size();j++){
sumSuppliedLastTimeRetailer += supplyRetailCustomer.get(i-1).get(j).quantity;
}
backordersRetailer.add(demandRetailCustomer.get(i-1)+ backordersRetailer.get(i-1) - sumSuppliedLastTimeRetailer);



int sumStockRetailer = 0;
for(int j=0;j<stockRetailer.get(i).size();j++){
sumStockRetailer += stockRetailer.get(i).get(j).quantity;
}
```

```
//Supplied(t) = Min(demand(t) + backorders(t), stock(t-1) + received shipment(t))
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFar = 0;
//while ((sumSuppliedSoFar <  (demandRetailCustomer.get(i)+ backordersRetailer.get(i))) && (sumStockRetailer > 0))
while ((sumSuppliedSoFar <  (demandRetailCustomer.get(i))) && (sumStockRetailer > 0))
{
//if (demandRetailCustomer.get(i)+ backordersRetailer.get(i) - sumSuppliedSoFar >= stockRetailer.get(i).get(0).quantity)
if (demandRetailCustomer.get(i) - sumSuppliedSoFar >= stockRetailer.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockRetailer.get(i).get(0).BBEdate, stockRetailer.get(i).get(0).quantity));
sumSuppliedSoFar += stockRetailer.get(i).get(0).quantity;
sumStockRetailer -= stockRetailer.get(i).get(0).quantity;
stockRetailer.get(i).remove(0);
}
else
{
//temp.add(new ShipmentData(stockRetailer.get(i).get(0).BBEdate, demandRetailCustomer.get(i)+ backordersRetailer.get(i) - sumSuppliedSoFar));
temp.add(new ShipmentData(stockRetailer.get(i).get(0).BBEdate, demandRetailCustomer.get(i) - sumSuppliedSoFar));
//sumStockRetailer -= demandRetailCustomer.get(i)+ backordersRetailer.get(i) - sumSuppliedSoFar;
sumStockRetailer -= demandRetailCustomer.get(i) - sumSuppliedSoFar;
//stockRetailer.get(i).get(0).quantity -= demandRetailCustomer.get(i)+ backordersRetailer.get(i) - sumSuppliedSoFar;
stockRetailer.get(i).get(0).quantity -= demandRetailCustomer.get(i) - sumSuppliedSoFar;
//sumSuppliedSoFar += demandRetailCustomer.get(i)+ backordersRetailer.get(i) - sumSuppliedSoFar;
sumSuppliedSoFar += demandRetailCustomer.get(i) - sumSuppliedSoFar;
}
};
supplyRetailCustomer.add(temp);


//New Order = Max(Indicated Order,0) = Order Rate(t) = Max [indicatedOrder(t) = expectedOrder(t) + alpha * [q - (Stock(t)-Backorders(t)) - beta
*ordersInPipeline(t)]],0]
newOrderRetailer.add( (int)java.lang.Math.round (Math.ceil(Math.max(0.0,(expectedOrderRetailer.get(i) + alpha_Retailer *
(q_Retailer - (sumStockRetailer - backordersRetailer.get(i)) - beta_Retailer * ordersInPipelineRetailer.get(i) ))))));


/************************************** MAIL ORDER (CUSTOMER)***************** *******************/

//Expected Order (t) = theta * Incoming Order (t-1) + (1 - theta) * Expected Order (t-1)
expectedOrderMailOrder.add((int)java.lang.Math.round(theta_HCmailorder * demandMailOrderCustomer.get(i-1) + (1-theta_HCmailorder) *
expectedOrderMailOrder.get(i-1)));


//ordersInPipeline(t) = ordersInPipeline(t-1) - ReceiveShipment(t-1) + New Order(t-1)
int sumReceiveShipmentMailOrder = 0;
for(int j=0;j<receiveShipmentMailOrder.get(i-1).size();j++){
sumReceiveShipmentMailOrder += receiveShipmentMailOrder.get(i-1).get(j).quantity;
}
ordersInPipelineMailOrder.add( (int) java.lang.Math.round(ordersInPipelineMailOrder.get(i-1) - sumReceiveShipmentMailOrder +
newOrderMailOrder.get(i-1)));


//Review the BBE losses in current array
temp = new ArrayList<ShipmentData>();
for(int m=0;m<stockMailOrder.get(i-1).size();m++){
temp.add( new ShipmentData(stockMailOrder.get(i-1).get(m).BBEdate,stockMailOrder.get(i-1).get(m).quantity));
}
//BBE losses
int sumBBElossesMailOrder = 0;
for(int m=0;m<temp.size();m++){
if(i >= temp.get(m).BBEdate){
```

```
sumBBElossesMailOrder += temp.get(m).quantity;
temp.remove(m);
m--;
}
}
stockBBELossesMailOrder.add((int)sumBBElossesMailOrder);


//1 week lead time for distribution to mail order - BBE data + quantity in this array
ArrayList<ShipmentData> temp3 = new ArrayList<ShipmentData>();
for(int j=0; j<supplyDistributionMailOrder.get(i-1).size();j++){
temp3.add(new ShipmentData(supplyDistributionMailOrder.get(i-1).get(j).BBEdate, supplyDistributionMailOrder.get(i-1).get(j).quantity));
}
receiveShipmentMailOrder.add(temp3);


// add the shipment to stock
for(int k=0;k < receiveShipmentMailOrder.get(i).size(); k++) {
temp.add(new ShipmentData(receiveShipmentMailOrder.get(i).get(k).BBEdate, receiveShipmentMailOrder.get(i).get(k).quantity));
}
stockMailOrder.add(temp);


int sumSuppliedLastTimeMailOrder = 0;
for(int j=0;j<supplyMailOrderCustomer.get(i-1).size();j++){
sumSuppliedLastTimeMailOrder += supplyMailOrderCustomer.get(i-1).get(j).quantity;
}
backordersMailOrder.add(demandMailOrderCustomer.get(i-1)+ backordersMailOrder.get(i-1) - sumSuppliedLastTimeMailOrder);


int sumStockMailOrder = 0;
for(int j=0;j<stockMailOrder.get(i).size();j++){
sumStockMailOrder += stockMailOrder.get(i).get(j).quantity;
}
//Supplied(t) = Min(demand(t) + backorders(t), stock(t) + received shipment(t))
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarMailOrder = 0;
while ((sumSuppliedSoFarMailOrder < (demandMailOrderCustomer.get(i))) && (sumStockMailOrder > 0))
{
if (demandMailOrderCustomer.get(i) - sumSuppliedSoFarMailOrder >= stockMailOrder.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockMailOrder.get(i).get(0).BBEdate, stockMailOrder.get(i).get(0).quantity));
sumSuppliedSoFarMailOrder += stockMailOrder.get(i).get(0).quantity;
sumStockMailOrder -= stockMailOrder.get(i).get(0).quantity;
stockMailOrder.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockMailOrder.get(i).get(0).BBEdate, demandMailOrderCustomer.get(i) - sumSuppliedSoFarMailOrder));
sumStockMailOrder -= demandMailOrderCustomer.get(i) - sumSuppliedSoFarMailOrder;
stockMailOrder.get(i).get(0).quantity -= demandMailOrderCustomer.get(i) - sumSuppliedSoFarMailOrder;
sumSuppliedSoFarMailOrder += demandMailOrderCustomer.get(i) - sumSuppliedSoFarMailOrder;
}
};
supplyMailOrderCustomer.add(temp);


//New Order = Max(Indicated Order,0) = Order Rate(t) = Max [indicatedOrder(t) = expectedOrder(t) + alpha * [q - (Stock(t)-Backorders(t)) - beta *ordersInPipeline(t)]),0]
newOrderMailOrder.add( (int)java.lang.Math.round (Math.ceil(Math.max(0.0,(expectedOrderMailOrder.get(i) + alpha_HCmailorder *
(q_HCmailorder - (sumStockMailOrder - backordersMailOrder.get(i)) - beta_HCmailorder * ordersInPipelineMailOrder.get(i) ))))));
```

```
/****************************************** HOTEL CHOCOLAT US **********************************************/

//Expected Order (t) = theta * Incoming Order (t-1) + (1 - theta) * Expected Order (t-1)
expectedUSOrder.add((int) java.lang.Math.round(theta_USHotelChocolat * demandUSCustomer.get(i-1) + (1-theta_USHotelChocolat) * expectedUSOrder.get(i-1)));

//ordersInPipeline(t) = ordersInPipeline(t-1) - ReceiveShipment(t-1) + New Order(t-1)
int sumReceiveShipmentUS = 0;
for(int j=0;j<receiveShipmentUS.get(i-1).size();j++){
sumReceiveShipmentUS += receiveShipmentUS.get(i-1).get(j).quantity;
}
ordersInPipelineUS.add((int) java.lang.Math.round(ordersInPipelineUS.get(i-1)- sumReceiveShipmentUS + newUSOrder.get(i-1)));


//Review the BBE losses in current array
temp = new ArrayList<ShipmentData>();
for(int m=0;m<stockUS.get(i-1).size();m++){
temp.add( new ShipmentData(stockUS.get(i-1).get(m).BBEdate,stockUS.get(i-1).get(m).quantity));
}
//BBE losses
int sumBBElossesUS = 0;
for(int m=0;m<temp.size();m++){
if(i >= temp.get(m).BBEdate){
sumBBElossesUS += temp.get(m).quantity;
temp.remove(m);
m--;
}
}
stockBBELossesUS.add((int)sumBBElossesUS);


//2 weeks lead time for distribution to supply the US -  BBE data + quantity in this array
ArrayList<ShipmentData> temp4 = new ArrayList<ShipmentData>();
for(int k=0;k < supplyDistributionUS.get(i-2).size(); k++) {
temp4.add(new ShipmentData(supplyDistributionUS.get(i-2).get(k).BBEdate, supplyDistributionUS.get(i-2).get(k).quantity)); //2 weeks lead time for distribution to supply the mail order customers
}
receiveShipmentUS.add(temp4);


// add the shipment to stock
for(int k=0;k < receiveShipmentUS.get(i).size(); k++) {
temp.add(new ShipmentData(receiveShipmentUS.get(i).get(k).BBEdate, receiveShipmentUS.get(i).get(k).quantity));
}
stockUS.add(temp);


int sumSuppliedLastTimeUS = 0;
for(int j=0;j<supplyUSCustomer.get(i-1).size();j++){
sumSuppliedLastTimeUS += supplyUSCustomer.get(i-1).get(j).quantity;
}
backordersUS.add(demandUSCustomer.get(i-1)+ backordersUS.get(i-1) - sumSuppliedLastTimeUS);



int sumStockUS = 0;
for(int j=0;j<stockUS.get(i).size();j++){
sumStockUS += stockUS.get(i).get(j).quantity;
```

```
}
//Supplied(t) = Min(demand(t) + backorders(t), stock(t) + received shipment(t))
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarUS = 0;
while ((sumSuppliedSoFarUS < (demandUSCustomer.get(i)+ backordersUS.get(i))) && (sumStockUS > 0))
{
if (demandUSCustomer.get(i)+ backordersUS.get(i) - sumSuppliedSoFarUS >= stockUS.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockUS.get(i).get(0).BBEdate, stockUS.get(i).get(0).quantity));
sumSuppliedSoFarUS += stockUS.get(i).get(0).quantity;
sumStockUS -= stockUS.get(i).get(0).quantity;
stockUS.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockUS.get(i).get(0).BBEdate, demandUSCustomer.get(i)+ backordersUS.get(i) - sumSuppliedSoFarUS));
sumStockUS -= demandUSCustomer.get(i)+ backordersUS.get(i) - sumSuppliedSoFarUS;
stockUS.get(i).get(0).quantity -= demandUSCustomer.get(i)+ backordersUS.get(i) - sumSuppliedSoFarUS;
sumSuppliedSoFarUS += demandUSCustomer.get(i)+ backordersUS.get(i) - sumSuppliedSoFarUS;

}
};
supplyUSCustomer.add(temp);


//Send Order = Max(Indicated Order,0) = Order Rate(t) = Max [indicatedOrder(t) = expectedOrder(t) + alpha * [q - Inventory(t) - beta *ordersInPipeline(t)]),0]
newUSOrder.add((int) java.lang.Math.round(Math.ceil(Math.max(0.0, expectedUSOrder.get(i) + alpha_USHotelChocolat *
(q_USHotelChocolat - (sumStockUS - backordersUS.get(i))- beta_USHotelChocolat * ordersInPipelineUS.get(i))))));


/***************************WHOLESALER ****************************/
//Expected Order (t) = theta * Incoming Order (t-1) + (1 - theta) * Expected Order (t-1)
expectedWholesaleOrder.add((int) java.lang.Math.round(theta_WholesaleUK * demandWholesale.get(i-1) + (1 - theta_WholesaleUK) * expectedWholesaleOrder.get(i-1)));

//ordersInPipeline(t) = ordersInPipeline(t-1) - ReceiveShipment(t-1) + New Order(t-1)
int sumReceiveShipmentWholesale = 0;
for(int j=0;j<receiveShipmentWholesale.get(i-1).size();j++){
sumReceiveShipmentWholesale += receiveShipmentWholesale.get(i-1).get(j).quantity;
}
ordersInPipelineWholesale.add((int) java.lang.Math.round(ordersInPipelineWholesale.get(i-1)- sumReceiveShipmentWholesale + newWholesaleOrder.get(i-1)));


//Review the BBE losses in current array
temp = new ArrayList<ShipmentData>();
for(int m=0;m<stockWholesale.get(i-1).size();m++){
temp.add( new ShipmentData(stockWholesale.get(i-1).get(m).BBEdate,stockWholesale.get(i-1).get(m).quantity));
}
//BBE losses
int sumBBElossesWholesale = 0;
for(int m=0;m<temp.size();m++){
if(i >= temp.get(m).BBEdate){
sumBBElossesWholesale += temp.get(m).quantity;
temp.remove(m);
m--;
}
}
stockBBELossesWholesale.add((int)sumBBElossesWholesale);
```

```
//2 weeks lead time for distribution to supply the wholesale - BBE data + quantity in this array
ArrayList<ShipmentData> temp5 = new ArrayList<ShipmentData>();
for(int j=0; j<supplyDistributionWholesale.get(i-2).size();j++){
temp5.add(new ShipmentData(supplyDistributionWholesale.get(i-2).get(j).BBEdate, supplyDistributionWholesale.get(i-2).get(j).quantity));
}
receiveShipmentWholesale.add(temp5);

for(int k=0;k < receiveShipmentWholesale.get(i).size(); k++) {
temp.add(new ShipmentData(receiveShipmentWholesale.get(i).get(k).BBEdate, receiveShipmentWholesale.get(i).get(k).quantity));
}
stockWholesale.add(temp);


int sumSuppliedLastTimeWholesale = 0;
for(int j=0;j<supplyWholesale.get(i-1).size();j++){
sumSuppliedLastTimeWholesale += supplyWholesale.get(i-1).get(j).quantity;
}
backordersWholesale.add(demandWholesale.get(i-1)+ backordersWholesale.get(i-1) - sumSuppliedLastTimeWholesale);



int sumStockWholesale = 0;
for(int j=0;j<stockWholesale.get(i).size();j++){
sumStockWholesale += stockWholesale.get(i).get(j).quantity;
}
//Supplied(t) = Min(demand(t) + backorders(t), stock(t) + received shipment(t))
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarWholesale = 0;
while ((sumSuppliedSoFarWholesale < (demandWholesale.get(i)+ backordersWholesale.get(i))) && (sumStockWholesale > 0))
{
if (demandWholesale.get(i)+ backordersWholesale.get(i) - sumSuppliedSoFarWholesale >= stockWholesale.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockWholesale.get(i).get(0).BBEdate, stockWholesale.get(i).get(0).quantity));
sumSuppliedSoFarWholesale += stockWholesale.get(i).get(0).quantity;
sumStockWholesale -= stockWholesale.get(i).get(0).quantity;
stockWholesale.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockWholesale.get(i).get(0).BBEdate, demandWholesale.get(i)+ backordersWholesale.get(i) -
sumSuppliedSoFarWholesale));
sumStockWholesale -= demandWholesale.get(i)+ backordersWholesale.get(i) - sumSuppliedSoFarWholesale;
stockWholesale.get(i).get(0).quantity -= demandWholesale.get(i)+ backordersWholesale.get(i) - sumSuppliedSoFarWholesale;
sumSuppliedSoFarWholesale += demandWholesale.get(i)+ backordersWholesale.get(i) - sumSuppliedSoFarWholesale;
}
};
supplyWholesale.add(temp);


//New Order = Max(Indicated Order,0) = Order Rate(t) = Max [indicatedOrder(t) = expectedOrder(t) + alpha * [q - Inventory(t) - beta
*ordersInPipeline(t)]),0]
newWholesaleOrder.add((int) java.lang.Math.round(Math.ceil(Math.max(0.0,(expectedWholesaleOrder.get(i) + alpha_WholesaleUK *
(q_WholesaleUK - (sumStockWholesale - backordersWholesale.get(i)) - beta_WholesaleUK * ordersInPipelineWholesale.get(i) ))))));



/****************** MIDDLE EAST FRANCHASING ******************************************************/

//Expected Order (t) = theta * Incoming Order (t-1) + (1 - theta) * Expected Order (t-1)
expectedMiddleEastOrder.add((int) java.lang.Math.round(theta_MiddleEastFranchising * demandMiddleEast.get(i-1) + (1 - theta_MiddleEastFranchising)
* expectedMiddleEastOrder.get(i-1)));
```

```
//ordersInPipeline(t) = ordersInPipeline(t-1) - ReceiveShipment(t-1) + New Order(t-1)
int sumReceiveShipmentMiddleEast = 0;
for(int j=0;j<receiveShipmentMiddleEast.get(i-1).size();j++){
sumReceiveShipmentMiddleEast += receiveShipmentMiddleEast.get(i-1).get(j).quantity;
}
ordersInPipelineMiddleEast.add((int) java.lang.Math.round(ordersInPipelineMiddleEast.get(i-1)- sumReceiveShipmentMiddleEast + newMiddleEastOrder.get(i-1)));


//Review the BBE losses in current array
temp = new ArrayList<ShipmentData>();
for(int m=0;m<stockMiddleEast.get(i-1).size();m++){
temp.add( new ShipmentData(stockMiddleEast.get(i-1).get(m).BBEdate,stockMiddleEast.get(i-1).get(m).quantity));
}
//BBE losses
int sumBBElossesMiddleEast = 0;
for(int m=0;m<temp.size();m++){
if(i >= temp.get(m).BBEdate){
sumBBElossesMiddleEast += temp.get(m).quantity;
temp.remove(m);
m--;
}
}
stockBBELossesMiddleEast.add((int)sumBBElossesMiddleEast);


//1 week lead time for distribution to supply to middle east
ArrayList<ShipmentData> temp6 = new ArrayList<ShipmentData>();
for(int k=0;k < supplyDistributionMiddleEast.get(i-1).size(); k++) {
temp6.add(new ShipmentData(supplyDistributionMiddleEast.get(i-1).get(k).BBEdate,supplyDistributionMiddleEast.get(i-1).get(k).quantity));
}
receiveShipmentMiddleEast.add(temp6);

// add the shipment to stock
for(int k=0;k < receiveShipmentMiddleEast.get(i).size(); k++) {
temp.add(new ShipmentData(receiveShipmentMiddleEast.get(i).get(k).BBEdate, receiveShipmentMiddleEast.get(i).get(k).quantity));
}
stockMiddleEast.add(temp);


int sumSuppliedLastTimeMiddleEast = 0;
for(int j=0;j<supplyMiddleEast.get(i-1).size();j++){
sumSuppliedLastTimeMiddleEast += supplyMiddleEast.get(i-1).get(j).quantity;
}
backordersMiddleEast.add(demandMiddleEast.get(i-1)+ backordersMiddleEast.get(i-1) - sumSuppliedLastTimeMiddleEast);


int sumStockMiddleEast = 0;
for(int j=0;j<stockMiddleEast.get(i).size();j++){
sumStockMiddleEast += stockMiddleEast.get(i).get(j).quantity;
}
//Supplied(t) = Min(demand(t) + backorders(t), stock(t) + received shipment(t))
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarMiddleEast = 0;
while ((sumSuppliedSoFarMiddleEast < (demandMiddleEast.get(i)+ backordersMiddleEast.get(i))) && (sumStockMiddleEast > 0))
{
if (demandMiddleEast.get(i)+ backordersMiddleEast.get(i) - sumSuppliedSoFarMiddleEast >= stockMiddleEast.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockMiddleEast.get(i).get(0).BBEdate, stockMiddleEast.get(i).get(0).quantity));
sumSuppliedSoFarMiddleEast += stockMiddleEast.get(i).get(0).quantity;
sumStockMiddleEast -= stockMiddleEast.get(i).get(0).quantity;
```

```
stockMiddleEast.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockMiddleEast.get(i).get(0).BBEdate, demandMiddleEast.get(i)+ backordersMiddleEast.get(i) - sumSuppliedSoFarMiddleEast));
sumStockMiddleEast -= demandMiddleEast.get(i)+ backordersMiddleEast.get(i) - sumSuppliedSoFarMiddleEast;
stockMiddleEast.get(i).get(0).quantity -= demandMiddleEast.get(i)+ backordersMiddleEast.get(i) - sumSuppliedSoFarMiddleEast;
sumSuppliedSoFarMiddleEast += demandMiddleEast.get(i)+ backordersMiddleEast.get(i) - sumSuppliedSoFarMiddleEast;
}
};
supplyMiddleEast.add(temp);

//Send Order = Max(Indicated Order,0) = Order Rate(t) = Max [indicatedOrder(t) = expectedOrder(t) + alpha * [q - Inventory(t) - beta *ordersInPipeline(t)]),0]
newMiddleEastOrder.add((int) java.lang.Math.round(Math.ceil(Math.max(0.0,(expectedMiddleEastOrder.get(i) + alpha_MiddleEastFranchising *
(q_MiddleEastFranchising - (sumStockMiddleEast - backordersMiddleEast.get(i)) - beta_MiddleEastFranchising * ordersInPipelineMiddleEast.get(i) ))))));



/******************************* DISTRIBUTION ***************************************************/

//total distribution demand
demandDistribution.add((int)(newOrderRetailer.get(i)+ newOrderMailOrder.get(i) + newUSOrder.get(i) + newWholesaleOrder.get(i) + newMiddleEastOrder.get(i)));

// no orders to the manufacturing

//Review the BBE losses in current array
temp = new ArrayList<ShipmentData>();
for(int m=0;m<stockDistribution.get(i-1).size();m++){
temp.add( new ShipmentData(stockDistribution.get(i-1).get(m).BBEdate,stockDistribution.get(i-1).get(m).quantity));
}
//BBE losses
int sumBBELossesDistribution = 0;
for(int m=0;m<temp.size();m++){
if(i >= temp.get(m).BBEdate){
sumBBELossesDistribution += temp.get(m).quantity;
temp.remove(m);
m--;
}
}
stockBBELossesDistribution.add((int)sumBBELossesDistribution);


// receive stock from manufacturing
ArrayList<ShipmentData> temp7 = new ArrayList<ShipmentData>();
for(int k=0;k < stockManufacturing.get(i-1).size(); k++) {
temp7.add(new ShipmentData(stockManufacturing.get(i-1).get(k).BBEdate,stockManufacturing.get(i-1).get(k).quantity)); // 1 week lead time
}
receiveShipmentDistribution.add(temp7);

// add the shipment to stock
for(int k=0;k < receiveShipmentDistribution.get(i).size(); k++) {
temp.add(new ShipmentData(receiveShipmentDistribution.get(i).get(k).BBEdate, receiveShipmentDistribution.get(i).get(k).quantity));
}
stockDistribution.add(temp);


//Distribution backorders to middle east franchasing - priotity 1
int sumSupplyDistributionMiddleEast = 0;
for(int j =0; j<supplyDistributionMiddleEast.get(i-1).size(); j++){
```

```
sumSupplyDistributionMiddleEast += supplyDistributionMiddleEast.get(i-1).get(j).quantity;
}
backordersDistributionMiddleEast.add(backordersDistributionMiddleEast.get(i-1) + newMiddleEastOrder.get(i-1)- sumSupplyDistributionMiddleEast);

//Distribution fulfilment to wholesale UK - priotity 2
int sumSupplyDistributionWholesale = 0;
for(int j =0; j<supplyDistributionWholesale.get(i-1).size(); j++){
sumSupplyDistributionWholesale += supplyDistributionWholesale.get(i-1).get(j).quantity;
}
backordersDistributionWholesale.add(backordersDistributionWholesale.get(i-1) + newWholesaleOrder.get(i-1)- sumSupplyDistributionWholesale);

//Distribution fulfilment to US - priority 3
int sumSupplyDistributionUS = 0;
for(int j =0; j<supplyDistributionUS.get(i-1).size(); j++){
sumSupplyDistributionUS += supplyDistributionUS.get(i-1).get(j).quantity;
}
backordersDistributionUS.add(backordersDistributionUS.get(i-1) + newUSOrder.get(i-1)- sumSupplyDistributionUS);

//Distribution fulfilment to mail order - priority 4
int sumSupplyDistributionMailOrder = 0;
for(int j =0; j<supplyDistributionMailOrder.get(i-1).size(); j++){
sumSupplyDistributionMailOrder += supplyDistributionMailOrder.get(i-1).get(j).quantity;
}
backordersDistributionMailOrder.add(backordersDistributionMailOrder.get(i-1) + newOrderMailOrder.get(i-1)- sumSupplyDistributionMailOrder);

//Distribution fulfilment to retailer - priority 5
int sumSupplyDistributionRetailer = 0;
for(int j =0; j<supplyDistributionRetailer.get(i-1).size(); j++){
sumSupplyDistributionRetailer += supplyDistributionRetailer.get(i-1).get(j).quantity;
}
backordersDistributionRetailer.add(backordersDistributionRetailer.get(i-1) + newOrderRetailer.get(i-1)- sumSupplyDistributionRetailer);

//Backorders(t) = - Min[0,inventory(t)]
backordersDistribution.add(backordersDistributionRetailer.get(i) + backordersDistributionMailOrder.get(i) + backordersDistributionUS.get(i)
+ backordersDistributionWholesale.get(i)+ backordersDistributionMiddleEast.get(i));


int sumStockDistribution = 0;
for(int j=0;j<stockDistribution.get(i).size();j++){
sumStockDistribution += stockDistribution.get(i).get(j).quantity;
}
//Distribution fulfilment to middle east franchasing - priority 1
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarDistributionMiddleEast = 0;
while ((sumSuppliedSoFarDistributionMiddleEast < (newMiddleEastOrder.get(i) + backordersDistributionMiddleEast.get(i))) && (sumStockDistribution > 0))
{
if (newMiddleEastOrder.get(i) + backordersDistributionMiddleEast.get(i) - sumSuppliedSoFarDistributionMiddleEast >= stockDistribution.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, stockDistribution.get(i).get(0).quantity));
sumSuppliedSoFarDistributionMiddleEast += stockDistribution.get(i).get(0).quantity;
sumStockDistribution -= stockDistribution.get(i).get(0).quantity;
stockDistribution.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, newMiddleEastOrder.get(i) + backordersDistributionMiddleEast.get(i) - sumSuppliedSoFarDistributionMiddleEast));
sumStockDistribution -= newMiddleEastOrder.get(i) + backordersDistributionMiddleEast.get(i) - sumSuppliedSoFarDistributionMiddleEast;
stockDistribution.get(i).get(0).quantity -= newMiddleEastOrder.get(i) + backordersDistributionMiddleEast.get(i) - sumSuppliedSoFarDistributionMiddleEast;
```

```
sumSuppliedSoFarDistributionMiddleEast += newMiddleEastOrder.get(i) + backordersDistributionMiddleEast.get(i) - sumSuppliedSoFarDistributionMiddleEast;
}
};
supplyDistributionMiddleEast.add(temp);



//Distribution fulfilment to wholesale UK - priotity 2
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarDistributionWholesale = 0;
while ((sumSuppliedSoFarDistributionWholesale < (newWholesaleOrder.get(i) + backordersDistributionWholesale.get(i))) && (sumStockDistribution > 0))
{
if (newWholesaleOrder.get(i) + backordersDistributionWholesale.get(i) - sumSuppliedSoFarDistributionWholesale >= stockDistribution.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, stockDistribution.get(i).get(0).quantity));
sumSuppliedSoFarDistributionWholesale += stockDistribution.get(i).get(0).quantity;
sumStockDistribution -= stockDistribution.get(i).get(0).quantity;
stockDistribution.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, newWholesaleOrder.get(i) + backordersDistributionWholesale.get(i) - sumSuppliedSoFarDistributionWholesale));
sumStockDistribution -= newWholesaleOrder.get(i) + backordersDistributionWholesale.get(i) - sumSuppliedSoFarDistributionWholesale;
stockDistribution.get(i).get(0).quantity -= newWholesaleOrder.get(i) + backordersDistributionWholesale.get(i) - sumSuppliedSoFarDistributionWholesale;
sumSuppliedSoFarDistributionWholesale += newWholesaleOrder.get(i) + backordersDistributionWholesale.get(i) - sumSuppliedSoFarDistributionWholesale;
}
};
supplyDistributionWholesale.add(temp);


//Distribution fulfilment to US - priority 3
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarDistributionUS = 0;
while ((sumSuppliedSoFarDistributionUS < (newUSOrder.get(i) + backordersDistributionUS.get(i))) && (sumStockDistribution > 0))
{
if (newUSOrder.get(i) + backordersDistributionUS.get(i) - sumSuppliedSoFarDistributionUS >= stockDistribution.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, stockDistribution.get(i).get(0).quantity));
sumSuppliedSoFarDistributionUS += stockDistribution.get(i).get(0).quantity;
sumStockDistribution -= stockDistribution.get(i).get(0).quantity;
stockDistribution.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, newUSOrder.get(i) + backordersDistributionUS.get(i) - sumSuppliedSoFarDistributionUS));
sumStockDistribution -= newUSOrder.get(i) + backordersDistributionUS.get(i) - sumSuppliedSoFarDistributionUS;
stockDistribution.get(i).get(0).quantity -= newUSOrder.get(i) + backordersDistributionUS.get(i) - sumSuppliedSoFarDistributionUS;
sumSuppliedSoFarDistributionUS += newUSOrder.get(i) + backordersDistributionUS.get(i) - sumSuppliedSoFarDistributionUS;
}
};
supplyDistributionUS.add(temp);

//Distribution fulfilment to mail order - priority 4
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarDistributionMailOrder = 0;
while ((sumSuppliedSoFarDistributionMailOrder < (newOrderMailOrder.get(i) + backordersDistributionMailOrder.get(i))) && (sumStockDistribution > 0))
{
```

```
if (newOrderMailOrder.get(i) + backordersDistributionMailOrder.get(i) - sumSuppliedSoFarDistributionMailOrder >= stockDistribution.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, stockDistribution.get(i).get(0).quantity));
sumSuppliedSoFarDistributionMailOrder += stockDistribution.get(i).get(0).quantity;
sumStockDistribution -= stockDistribution.get(i).get(0).quantity;
stockDistribution.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, newOrderMailOrder.get(i) + backordersDistributionMailOrder.get(i) - sumSuppliedSoFarDistributionMailOrder));
sumStockDistribution -= newOrderMailOrder.get(i) + backordersDistributionMailOrder.get(i) - sumSuppliedSoFarDistributionMailOrder;
stockDistribution.get(i).get(0).quantity -= newOrderMailOrder.get(i) + backordersDistributionMailOrder.get(i) - sumSuppliedSoFarDistributionMailOrder;
sumSuppliedSoFarDistributionMailOrder += newOrderMailOrder.get(i) + backordersDistributionMailOrder.get(i) - sumSuppliedSoFarDistributionMailOrder;

}
};
supplyDistributionMailOrder.add(temp);


//Distribution fulfilment to retailer - priority 5
temp = new ArrayList<ShipmentData>();
int sumSuppliedSoFarDistributionRetailer = 0;
while ((sumSuppliedSoFarDistributionRetailer < (newOrderRetailer.get(i) + backordersDistributionRetailer.get(i))) && (sumStockDistribution > 0))
{
if (newOrderRetailer.get(i) + backordersDistributionRetailer.get(i) - sumSuppliedSoFarDistributionRetailer >= stockDistribution.get(i).get(0).quantity)
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, stockDistribution.get(i).get(0).quantity));
sumSuppliedSoFarDistributionRetailer += stockDistribution.get(i).get(0).quantity;
sumStockDistribution -= stockDistribution.get(i).get(0).quantity;
stockDistribution.get(i).remove(0);
}
else
{
temp.add(new ShipmentData(stockDistribution.get(i).get(0).BBEdate, newOrderRetailer.get(i) + backordersDistributionRetailer.get(i) - sumSuppliedSoFarDistributionRetailer));
sumStockDistribution -= newOrderRetailer.get(i) + backordersDistributionRetailer.get(i) - sumSuppliedSoFarDistributionRetailer;
stockDistribution.get(i).get(0).quantity -= newOrderRetailer.get(i) + backordersDistributionRetailer.get(i) - sumSuppliedSoFarDistributionRetailer;
sumSuppliedSoFarDistributionRetailer += newOrderRetailer.get(i) + backordersDistributionRetailer.get(i) - sumSuppliedSoFarDistributionRetailer;
}
};
supplyDistributionRetailer.add(temp);


//Total supplied by distribution - Supplied(t) = Min(demand(t) + backorders(t), stock(t) + received shipment(t))
sumSupplyDistributionMiddleEast = 0;
for(int j=0;j<supplyDistributionMiddleEast.get(i).size();j++){
sumSupplyDistributionMiddleEast += supplyDistributionMiddleEast.get(i).get(j).quantity;
}

sumSupplyDistributionWholesale = 0;
for(int j=0;j<supplyDistributionWholesale.get(i).size();j++){
sumSupplyDistributionWholesale += supplyDistributionWholesale.get(i).get(j).quantity;
}
sumSupplyDistributionUS = 0;
for(int j=0;j<supplyDistributionUS.get(i).size();j++){
sumSupplyDistributionUS += supplyDistributionUS.get(i).get(j).quantity;
}
sumSupplyDistributionMailOrder = 0;
for(int j=0;j<supplyDistributionMailOrder.get(i).size();j++){
```

```
sumSupplyDistributionMailOrder += supplyDistributionMailOrder.get(i).get(j).quantity;
}
sumSupplyDistributionRetailer = 0;
for(int j=0;j<supplyDistributionRetailer.get(i).size();j++){
sumSupplyDistributionRetailer += supplyDistributionRetailer.get(i).get(j).quantity;
}




/************************************* MANUFACTURING *****************************************/

//Manufacturing production
temp = new ArrayList<ShipmentData>();
temp.add(new ShipmentData((i + minimumShelfLife),(int) java.lang.Math.round(forecastingManufacturing.get(i + 1)) ));
productionManufacturing.add(temp);

//probable this should be forecastingManufacturing.get(i+2)

//Manufacturing stock
temp = new ArrayList<ShipmentData>();
for(int j=0;j<productionManufacturing.get(i-1).size();j++){
temp.add(new ShipmentData(productionManufacturing.get(i-1).get(j).BBEdate,((int) java.lang.Math.round(productionManufacturing.get(i-
1).get(j).quantity))));
}
stockManufacturing.add(temp);
}



/** DETERMINE THE HOTEL CHOCOLAT OVERALL COSTINGS AND PROFIT */

//Product Details
double productRSP = 20; // Product Retail Selling Price
double vat = 0.20; // vat = 20%
double productCost = 0.35 * (productRSP * (1 - vat));
double handlingCost = 0.35 * productCost;
double materialCost = productCost - handlingCost;
//double productionCapacity = 2500;


ArrayList<Integer> retailStockHoldingCosts = new ArrayList<Integer> ();
ArrayList<Integer> retailSalesProfit = new ArrayList<Integer> ();
ArrayList<Integer> retailerBBELossesCosts = new ArrayList<Integer> ();
ArrayList<Integer> retailerTransportCosts = new ArrayList<Integer> ();
ArrayList<Integer> retailLostSalesCosts = new ArrayList<Integer> ();
ArrayList<Integer> retailerProfit = new ArrayList<Integer> ();
ArrayList<Integer> mailOrderStockHoldingCosts = new ArrayList<Integer> ();
ArrayList<Integer> mailOrderSalesProfit = new ArrayList<Integer> ();
ArrayList<Integer> mailOrderBBELossesCosts = new ArrayList<Integer> ();
ArrayList<Integer> mailOrderTransportCharges = new ArrayList<Integer> ();
ArrayList<Integer> mailOrderLostSalesCosts = new ArrayList<Integer> ();
ArrayList<Integer> mailOrderProfit = new ArrayList<Integer> ();
ArrayList<Integer> usDeliveryPenalty = new ArrayList<Integer> ();
ArrayList<Integer> usSalesProfit = new ArrayList<Integer> ();
ArrayList<Integer> usTransportCosts = new ArrayList<Integer> ();
ArrayList<Integer> usProfit = new ArrayList<Integer> ();
ArrayList<Integer> usBBELossesCosts = new ArrayList<Integer> ();
ArrayList<Integer> wholesalerDeliveryPenalty = new ArrayList<Integer> ();
```

```
ArrayList<Integer> wholesalerSalesProfit = new ArrayList<Integer> ();
ArrayList<Integer> wholesalerTransportCosts = new ArrayList<Integer> ();
ArrayList<Integer> wholesalerProfit = new ArrayList<Integer> ();
ArrayList<Integer> wholesalerBBELossesCosts = new ArrayList<Integer> ();
ArrayList<Integer> middleEastDeliveryPenalty = new ArrayList<Integer> ();
ArrayList<Integer> middleEastSalesProfit = new ArrayList<Integer> ();
ArrayList<Integer> middleEastProfit = new ArrayList<Integer> ();
ArrayList<Integer> middleEastBBELossesCosts = new ArrayList<Integer> ();
ArrayList<Integer> distributionStockHoldingCosts = new ArrayList<Integer> ();
ArrayList<Integer> distributionBBELossesCosts = new ArrayList<Integer> ();
ArrayList<Integer> distributionTransportCosts = new ArrayList<Integer> ();
ArrayList<Integer> distributionCosts = new ArrayList<Integer> ();
ArrayList<Integer> materialManufacturingCosts = new ArrayList<Integer> ();
ArrayList<Integer> packingManufacturingCosts = new ArrayList<Integer> ();
ArrayList<Integer> manufacturingCosts = new ArrayList<Integer> ();


for (int k=0; k<demandRetailCustomer.size(); k++) {

/** RETAILER */

//retailStockHoldingCosts.add((int)java.lang.Math.round(.01 * productRSP * stockRetailer.get(k).quantity));
int sumRetailStockHolding = 0;
int sumRetailStockHoldingCosts = 0;
for(int i =0; i<stockRetailer.get(k).size(); i++){
sumRetailStockHolding += stockRetailer.get(k).get(i).quantity;
}
sumRetailStockHoldingCosts += (int)java.lang.Math.round(.01 * productRSP * sumRetailStockHolding);
retailStockHoldingCosts.add((int)sumRetailStockHoldingCosts);


// retailSalesProfit.add( (int)java.lang.Math.round(supplyRetailCustomer.get(k).quantity * (( productRSP*(1-vat)) - productCost)));
int sumRetailSales = 0;
int sumRetailSalesProfit = 0;
for(int i =0; i<supplyRetailCustomer.get(k).size(); i++){
sumRetailSales += supplyRetailCustomer.get(k).get(i).quantity;
}
sumRetailSalesProfit += (int)java.lang.Math.round(sumRetailSales * (( productRSP*(1-vat)) - productCost));
retailSalesProfit.add((int)sumRetailSalesProfit);


retailerBBELossesCosts.add((int)(stockBBELossesRetailer.get(k)* productCost));    // retailer BBE losses


// retailerTransportCosts.add( (int)(0.095000 * receiveShipmentRetailer.get(k).quantity)); //  £0.10/item
int sumRetailerTransport = 0;
int sumRetailerTransportCosts = 0;
for(int i =0; i<receiveShipmentRetailer.get(k).size(); i++){
sumRetailerTransport += receiveShipmentRetailer.get(k).get(i).quantity;
}
sumRetailerTransportCosts += (int)java.lang.Math.round(sumRetailerTransport * 0.095000);
retailerTransportCosts.add((int)sumRetailerTransportCosts);


retailerProfit.add( (int)java.lang.Math.round(retailSalesProfit.get(k) - retailStockHoldingCosts.get(k) - retailerTransportCosts.get(k))); // - retailerBBELossesCosts.get(k)


//retailLostSalesCosts.add( (int)java.lang.Math.round((productRSP*(1-vat) - productCost)* (demandRetailCustomer.get(k) - supplyRetailCustomer.get(k).quantity)));
int sumRetailLostSales = 0;
int sumRetailLostSalesCosts = 0;
```

```
for(int i =0; i<supplyRetailCustomer.get(k).size(); i++){
sumRetailLostSales += supplyRetailCustomer.get(k).get(i).quantity;
}
sumRetailLostSalesCosts += (int)java.lang.Math.round((productRSP*(1-vat) - productCost)* (demandRetailCustomer.get(k) - sumRetailLostSales));
retailLostSalesCosts.add((int)sumRetailLostSalesCosts);


/** mail order */

// mailOrderStockHoldingCosts.add( (int)java.lang.Math.round(0.005 * productRSP * stockMailOrder.get(k).quantity));
int sumMailOrderStockHolding = 0;
int sumMailOrderStockHoldingCosts = 0;
for(int i =0; i<stockMailOrder.get(k).size(); i++){
sumMailOrderStockHolding += stockMailOrder.get(k).get(i).quantity;
}
sumMailOrderStockHoldingCosts += (int)java.lang.Math.round(0.005 * productRSP * sumMailOrderStockHolding);
mailOrderStockHoldingCosts.add((int)sumMailOrderStockHoldingCosts);



// mailOrderSalesProfit.add( (int)java.lang.Math.round(supplyMailOrderCustomer.get(k).quantity * (( productRSP * (1 - vat)) - productCost)));
int sumMailOrderSales = 0;
int sumMailOrderSalesProfit = 0;
for(int i =0; i<supplyMailOrderCustomer.get(k).size(); i++){
sumMailOrderSales += supplyMailOrderCustomer.get(k).get(i).quantity;
}
sumMailOrderSalesProfit += (int)java.lang.Math.round(sumMailOrderSales * (( productRSP*(1-vat)) - productCost));
mailOrderSalesProfit.add((int)sumMailOrderSalesProfit);


mailOrderBBELossesCosts.add( (int)( stockBBELossesMailOrder.get(k)* productCost));    // mail order BBE losses


// mailOrderTransportCharges.add( (int)java.lang.Math.round(1.5 * supplyMailOrderCustomer.get(k).quantity)); //  £1.5 transport profit /item
int sumMailOrderTransport = 0;
int sumMailOrderTransportCharges = 0;
for(int i =0; i<supplyMailOrderCustomer.get(k).size(); i++){
sumMailOrderTransport += supplyMailOrderCustomer.get(k).get(i).quantity;
}
sumMailOrderTransportCharges += java.lang.Math.round(1.5 * sumMailOrderTransport);
mailOrderTransportCharges.add((int)sumMailOrderTransportCharges);


mailOrderProfit.add( (int)java.lang.Math.round(mailOrderSalesProfit.get(k) - mailOrderStockHoldingCosts.get(k) + mailOrderTransportCharges.get(k))); //
- mailOrdererBBELossesCosts.get(k)


//mailOrderLostSalesCosts.add(      (int)java.lang.Math.round((productRSP*(1-vat)      -      productCost)      *      (demandMailOrderCustomer.get(k)      -
supplyMailOrderCustomer.get(k).quantity)));
int sumMailOrderLostSales = 0;
int sumMailOrderLostSalesCosts = 0;
for(int i =0; i<supplyMailOrderCustomer.get(k).size(); i++){
sumMailOrderLostSales += supplyMailOrderCustomer.get(k).get(i).quantity;
}
sumMailOrderLostSalesCosts      +=      (int)java.lang.Math.round((productRSP*(1-vat)      -      productCost)*      (demandMailOrderCustomer.get(k)      -
sumMailOrderLostSales));
mailOrderLostSalesCosts.add((int)sumMailOrderLostSalesCosts);


/** HOTEL CHOCOLAT US */
```

```
usDeliveryPenalty.add( (int)java.lang.Math.round(0.05 * productRSP * backordersUS.get(k))); //5% of RSP


// usSalesProfit.add( (int)java.lang.Math.round(receiveShipmentUS.get(k).quantity *(productRSP * 0.39))); //39% of RSP
int sumUSSales = 0;
int sumUSSalesProfit = 0;
for(int i =0; i<receiveShipmentUS.get(k).size(); i++){
sumUSSales += receiveShipmentUS.get(k).get(i).quantity;
}
sumUSSalesProfit += (int)java.lang.Math.round(sumUSSales * productRSP * 0.39);
usSalesProfit.add((int)sumUSSalesProfit);


usBBELossesCosts.add( (int)( stockBBELossesUS.get(k) * productCost));    // US BBE losses


//usTransportCosts.add( (int)java.lang.Math.round(0.178333333 * receiveShipmentUS.get(k).quantity)); //  £0.178333333/item
int sumUSTransport = 0;
int sumUSTransportCosts = 0;
for(int i =0; i<receiveShipmentUS.get(k).size(); i++){
sumUSTransport += receiveShipmentUS.get(k).get(i).quantity;
}
sumUSTransportCosts += java.lang.Math.round(0.178333333 * sumUSTransport);
usTransportCosts.add((int)sumUSTransportCosts);


usProfit.add( (int)java.lang.Math.round(usSalesProfit.get(k) - usDeliveryPenalty.get(k) - usTransportCosts.get(k)));



/**  WHOLESALER UK */
wholesalerDeliveryPenalty.add( (int)java.lang.Math.round(0.05 * productRSP * backordersWholesale.get(k))); //5% of RSP

//wholesalerSalesProfit.add( (int)java.lang.Math.round(receiveShipmentWholesale.get(k).quantity * (productRSP * 0.4))); //40% of RSP
int sumWholesalerSales = 0;
int sumWholesalerSalesProfit = 0;
for(int i =0; i<receiveShipmentWholesale.get(k).size(); i++){
sumWholesalerSales += receiveShipmentWholesale.get(k).get(i).quantity;
}
sumWholesalerSalesProfit += (int)java.lang.Math.round(sumWholesalerSales * (productRSP * 0.4));
wholesalerSalesProfit.add((int)sumWholesalerSalesProfit);


wholesalerBBELossesCosts.add( (int)( stockBBELossesWholesale.get(k)* productCost));    // US BBE losses


//wholesalerTransportCosts.add( (int)java.lang.Math.round(0.095 * receiveShipmentWholesale.get(k).quantity)); //  £0.10/item
int sumWholesalerTransport = 0;
int sumWholesalerTransportCosts = 0;
for(int i =0; i<receiveShipmentWholesale.get(k).size(); i++){
sumWholesalerTransport += receiveShipmentWholesale.get(k).get(i).quantity;
}
sumWholesalerTransportCosts += (int)java.lang.Math.round(sumWholesalerTransport * 0.095);
wholesalerTransportCosts.add((int)sumWholesalerTransportCosts);


wholesalerProfit.add( (int)java.lang.Math.round(wholesalerSalesProfit.get(k) - wholesalerDeliveryPenalty.get(k) - wholesalerTransportCosts.get(k)));


/** MIDDLE EAST FRANCHASING */
middleEastDeliveryPenalty.add( (int)java.lang.Math.round(0.05 * productRSP * backordersMiddleEast.get(k))); //5% of RSP
```

```
//middleEastSalesProfit.add( (int)java.lang.Math.round(receiveShipmentMiddleEast.get(k).quantity * (productRSP * 0.41))); //41% of RSP
int sumMiddleEastSales = 0;
int sumMiddleEastSalesProfit = 0;
for(int i =0; i<receiveShipmentMiddleEast.get(k).size(); i++){
sumMiddleEastSales += receiveShipmentMiddleEast.get(k).get(i).quantity;
}
sumMiddleEastSalesProfit += (int)java.lang.Math.round(sumMiddleEastSales * (productRSP * 0.41));
middleEastSalesProfit.add((int)sumMiddleEastSalesProfit);

middleEastBBELossesCosts.add( (int)( stockBBELossesMiddleEast.get(k) * productCost));    // US BBE losses
middleEastProfit.add( (int)java.lang.Math.round(middleEastSalesProfit.get(k) - middleEastDeliveryPenalty.get(k)));



/**  DISTRIBUTION */
distributionBBELossesCosts.add( (int)( stockBBELossesDistribution.get(k)* productCost));    // Distribution BBE losses

//distributionStockHoldingCosts.add( (int)java.lang.Math.round(0.005 * productCost * stockDistribution.get(k).quantity)); //5% of RSP
int sumDistributionStockHolding = 0;
int sumDistributionStockHoldingCosts = 0;
for(int i =0; i<stockDistribution.get(k).size(); i++){
sumDistributionStockHolding += stockDistribution.get(k).get(i).quantity;
}
sumDistributionStockHoldingCosts += (int)java.lang.Math.round(0.005 * productCost * sumDistributionStockHolding);
distributionStockHoldingCosts.add((int)sumDistributionStockHoldingCosts);


// distributionTransportCosts.add( (int)java.lang.Math.round(0.022500 * receiveShipmentDistribution.get(k).quantity)); //  £0.0225/item
int sumDistributionTransport = 0;
int sumDistributionTransportCosts = 0;
for(int i =0; i<receiveShipmentDistribution.get(k).size(); i++){
sumDistributionTransport += receiveShipmentDistribution.get(k).get(i).quantity;
}
sumDistributionTransportCosts += (int)java.lang.Math.round(sumDistributionTransport * 0.022500);
distributionTransportCosts.add((int)sumDistributionTransportCosts);

distributionCosts.add(    (int)java.lang.Math.round(distributionStockHoldingCosts.get(k)    +    distributionTransportCosts.get(k)));    //+
distributionBBELossesCosts.get(k)


/** MANUFACTURING */
// materialManufacturingCosts.add( (int)java.lang.Math.round(materialCost * productionManufacturing.get(k).quantity));
int sumMaterialManufacturing = 0;
int sumMaterialManufacturingCosts = 0;
for(int i =0; i<productionManufacturing.get(k).size(); i++){
sumMaterialManufacturing += productionManufacturing.get(k).get(i).quantity;
}
sumMaterialManufacturingCosts += (int)java.lang.Math.round(materialCost* sumMaterialManufacturing);
materialManufacturingCosts.add((int)sumMaterialManufacturingCosts);


// packingManufacturingCosts.add( (int)java.lang.Math.round(handlingCost * productionManufacturing.get(k).quantity));
int sumPackingManufacturing = 0;
int sumPackingManufacturingCosts = 0;
for(int i =0; i<productionManufacturing.get(k).size(); i++){
sumPackingManufacturing += productionManufacturing.get(k).get(i).quantity;
}
sumPackingManufacturingCosts += (int)java.lang.Math.round(handlingCost* sumPackingManufacturing);
packingManufacturingCosts.add((int)sumPackingManufacturingCosts);


manufacturingCosts.add( (int)java.lang.Math.round(materialManufacturingCosts.get(k) + packingManufacturingCosts.get(k)));
```

```
}

/** PROFIT */

ArrayList<Integer> overallProfit = new ArrayList<Integer> ();
for (int r=0; r<demandRetailCustomer.size(); r++) {
overallProfit.add( (int)java.lang.Math.round(retailerProfit.get(r) + mailOrderProfit.get(r) + usProfit.get(r) + wholesalerProfit.get(r)
+ middleEastProfit.get(r) - distributionCosts.get(r) - manufacturingCosts.get(r)));
}

double sum = 0.0;

for(int i =0; i<overallProfit.size(); i++){
sum += overallProfit.get(i);
}

if (outputData)
{
//EXPORT THE DATA

FileOutputStream out; // declare a file output object
PrintStream p; // declare a print stream object
int i;
try{
// Create a new file output stream connected to "myfileBBE.txt"
out = new FileOutputStream("myfileBBE.txt");

// Connect print stream to the output stream
p = new PrintStream( out );

for (i = 0;i<demandRetailCustomer.size(); i++){

p.format("%d,", i);

//*********************** retailer *****************
p.format("%d,", demandRetailCustomer.get(i));
p.format("%d,", expectedOrderRetailer.get(i));
p.format("%d,", newOrderRetailer.get(i));
p.format("%d,", ordersInPipelineRetailer.get(i));

int sumSupplyRetailCustomer = 0;
for(int j =0; j<supplyRetailCustomer.get(i).size(); j++){
sumSupplyRetailCustomer += supplyRetailCustomer.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyRetailCustomer);
p.format("%d,", backordersRetailer.get(i));
p.format("%d,", stockBBELossesRetailer.get(i));


int sumStockRetailer = 0;
for(int j =0; j<stockRetailer.get(i).size(); j++){
sumStockRetailer += stockRetailer.get(i).get(j).quantity;
}
p.format("%d,", sumStockRetailer);

int sumReceiveShipmentRetailer = 0;
for(int j =0; j<receiveShipmentRetailer.get(i).size(); j++){
sumReceiveShipmentRetailer += receiveShipmentRetailer.get(i).get(j).quantity;
```

```
}
p.format("%d,", sumReceiveShipmentRetailer);



//*********************** mail order *****************
p.format("%d,", demandMailOrderCustomer.get(i));
p.format("%d,", expectedOrderMailOrder.get(i));
p.format("%d,", newOrderMailOrder.get(i));
p.format("%d,", ordersInPipelineMailOrder.get(i));

int sumSupplyMailOrderCustomer = 0;
for(int j =0; j<supplyMailOrderCustomer.get(i).size(); j++){
sumSupplyMailOrderCustomer += supplyMailOrderCustomer.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyMailOrderCustomer);
p.format("%d,", backordersMailOrder.get(i));
p.format("%d,", stockBBELossesMailOrder.get(i));

int sumStockMailOrder = 0;
for(int j =0; j<stockMailOrder.get(i).size(); j++){
sumStockMailOrder += stockMailOrder.get(i).get(j).quantity;
}
p.format("%d,", sumStockMailOrder);

int sumReceiveShipmentMailOrder = 0;
for(int j =0; j<receiveShipmentMailOrder.get(i).size(); j++){
sumReceiveShipmentMailOrder += receiveShipmentMailOrder.get(i).get(j).quantity;
}
p.format("%d,", sumReceiveShipmentMailOrder);


//******************** US ********************
p.format("%d,", demandUSCustomer.get(i));
p.format("%d,", expectedUSOrder.get(i));
p.format("%d,", newUSOrder.get(i));
p.format("%d,", ordersInPipelineUS.get(i));

int sumSupplyUSCustomer = 0;
for(int j =0; j<supplyUSCustomer.get(i).size(); j++){
sumSupplyUSCustomer += supplyUSCustomer.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyUSCustomer);
p.format("%d,", backordersUS.get(i));
p.format("%d,", stockBBELossesUS.get(i));


int sumStockUS = 0;
for(int j =0; j<stockUS.get(i).size(); j++){
sumStockUS += stockUS.get(i).get(j).quantity;
}
p.format("%d,", sumStockUS);

int sumReceiveShipmentUS = 0;
for(int j =0; j<receiveShipmentUS.get(i).size(); j++){
sumReceiveShipmentUS += receiveShipmentUS.get(i).get(j).quantity;
}
p.format("%d,", sumReceiveShipmentUS);
```

```
//***************** Wholesale ***********************
p.format("%d,", demandWholesale.get(i));
p.format("%d,", expectedWholesaleOrder.get(i));
p.format("%d,", newWholesaleOrder.get(i));
p.format("%d,", ordersInPipelineWholesale.get(i));

int sumSupplyWholesale = 0;
for(int j =0; j<supplyWholesale.get(i).size(); j++){
sumSupplyWholesale += supplyWholesale.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyWholesale);
p.format("%d,", backordersWholesale.get(i));
p.format("%d,", stockBBELossesWholesale.get(i));

int sumStockWholesale = 0;
for(int j =0; j<stockWholesale.get(i).size(); j++){
sumStockWholesale += stockWholesale.get(i).get(j).quantity;
}
p.format("%d,", sumStockWholesale);

int sumReceiveShipmentWholesale = 0;
for(int j =0; j<receiveShipmentWholesale.get(i).size(); j++){
sumReceiveShipmentWholesale += receiveShipmentWholesale.get(i).get(j).quantity;
}
p.format("%d,", sumReceiveShipmentWholesale);


//********************* middle east *******************
p.format("%d,", demandMiddleEast.get(i));
p.format("%d,", expectedMiddleEastOrder.get(i));
p.format("%d,", newMiddleEastOrder.get(i));
p.format("%d,", ordersInPipelineMiddleEast.get(i));

int sumSupplyMiddleEast = 0;
for(int j =0; j<supplyMiddleEast.get(i).size(); j++){
sumSupplyMiddleEast += supplyMiddleEast.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyMiddleEast);

p.format("%d,", backordersMiddleEast.get(i));
p.format("%d,", stockBBELossesMiddleEast.get(i));


int sumStockMiddleEast = 0;
for(int j =0; j<stockMiddleEast.get(i).size(); j++){
sumStockMiddleEast += stockMiddleEast.get(i).get(j).quantity;
}
p.format("%d,", sumStockMiddleEast);

int sumReceiveShipmentMiddleEast = 0;
for(int j =0; j<receiveShipmentMiddleEast.get(i).size(); j++){
sumReceiveShipmentMiddleEast += receiveShipmentMiddleEast.get(i).get(j).quantity;
}
p.format("%d,", sumReceiveShipmentMiddleEast);


//**********************distribution*****************
p.format("%d,", demandDistribution.get(i));


int sumSupplyDistributionRetailer = 0;
for(int j =0; j<supplyDistributionRetailer.get(i).size(); j++){
```

```
sumSupplyDistributionRetailer += supplyDistributionRetailer.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyDistributionRetailer);

int sumSupplyDistributionMailOrder = 0;
for(int j =0; j<supplyDistributionMailOrder.get(i).size(); j++){
sumSupplyDistributionMailOrder += supplyDistributionMailOrder.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyDistributionMailOrder);

int sumSupplyDistributionUS = 0;
for(int j =0; j<supplyDistributionUS.get(i).size(); j++){
sumSupplyDistributionUS += supplyDistributionUS.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyDistributionUS);

int sumSupplyDistributionWholesale = 0;
for(int j =0; j<supplyDistributionWholesale.get(i).size(); j++){
sumSupplyDistributionWholesale += supplyDistributionWholesale.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyDistributionWholesale);

int sumSupplyDistributionMiddleEast = 0;
for(int j =0; j<supplyDistributionMiddleEast.get(i).size(); j++){
sumSupplyDistributionMiddleEast += supplyDistributionMiddleEast.get(i).get(j).quantity;
}
p.format("%d,", sumSupplyDistributionMiddleEast);

int sumSupplyDistribution = sumSupplyDistributionWholesale + sumSupplyDistributionMiddleEast + sumSupplyDistributionUS +
sumSupplyDistributionMailOrder + sumSupplyDistributionRetailer;

p.format("%d,", sumSupplyDistribution);

p.format("%d,", backordersDistributionRetailer.get(i));
p.format("%d,", backordersDistributionMailOrder.get(i));
p.format("%d,", backordersDistributionUS.get(i));
p.format("%d,", backordersDistributionWholesale.get(i));
p.format("%d,", backordersDistributionMiddleEast.get(i));
p.format("%d,", backordersDistribution.get(i));

p.format("%d,", stockBBELossesDistribution.get(i));


int sumStockDistribution = 0;
for(int j =0; j<stockDistribution.get(i).size(); j++){
sumStockDistribution += stockDistribution.get(i).get(j).quantity;
}
p.format("%d,", sumStockDistribution);

int sumReceiveShipmentDistribution = 0;
for(int j =0; j<receiveShipmentDistribution.get(i).size(); j++){
sumReceiveShipmentDistribution += receiveShipmentDistribution.get(i).get(j).quantity;
}
p.format("%d,", sumReceiveShipmentDistribution);


// *******************************************************
int sumStockManufacturing = 0;
for(int j =0; j<stockManufacturing.get(i).size(); j++){
sumStockManufacturing += stockManufacturing.get(i).get(j).quantity;
}
p.format("%d,", sumStockManufacturing);
```

```
int sumProductionManufacturing = 0;
for(int j =0; j<productionManufacturing.get(i).size(); j++){
sumProductionManufacturing += productionManufacturing.get(i).get(j).quantity;
}
p.format("%d,", sumProductionManufacturing);
p.format("%d\n", forecastingManufacturing.get(i));
}
p.close();
}
catch (Exception e)
{
System.err.println ("Error writing to file - BBE_SIMULATION");
}


try
{
// Create a new file output stream connected to "costingsfile.txt"
out = new FileOutputStream("BBEcostingsfile.txt");

// Connect print stream to the output stream
p = new PrintStream( out );

for (i = 0;i<demandRetailCustomer.size(); i++){

p.format("%d,", i);

//retailer
p.format("%d,", retailStockHoldingCosts.get(i));
p.format("%d,", retailSalesProfit.get(i));
p.format("%d,", retailerTransportCosts.get(i));
p.format("%d,", retailerProfit.get(i));
p.format("%d,", retailLostSalesCosts.get(i));

//mail order
p.format("%d,", mailOrderTransportCharges.get(i));
p.format("%d,", mailOrderStockHoldingCosts.get(i));
p.format("%d,", mailOrderSalesProfit.get(i));
p.format("%d,", mailOrderProfit.get(i));
p.format("%d,", mailOrderLostSalesCosts.get(i));

// US
p.format("%d,", usDeliveryPenalty.get(i));
p.format("%d,", usTransportCosts.get(i));
p.format("%d,", usSalesProfit.get(i));
p.format("%d,", usProfit.get(i));

//wholesaler
p.format("%d,", wholesalerDeliveryPenalty.get(i));
p.format("%d,", wholesalerTransportCosts.get(i));
p.format("%d,", wholesalerSalesProfit.get(i));
p.format("%d,", wholesalerProfit.get(i));

//middle east
p.format("%d,", middleEastDeliveryPenalty.get(i));
p.format("%d,", middleEastSalesProfit.get(i));
p.format("%d,", middleEastProfit.get(i));

//distribution
```

```
p.format("%d,", distributionStockHoldingCosts.get(i));
p.format("%d,", distributionTransportCosts.get(i));
p.format("%d,", distributionCosts.get(i));

//manufacturing
p.format("%d,", materialManufacturingCosts.get(i));
p.format("%d,", packingManufacturingCosts.get(i));
//p.format("%d,", freeManufacturingCapacityCosts.get(i));
p.format("%d,", manufacturingCosts.get(i));

//overall profit
p.format("%d\n", overallProfit.get(i));
//p.format("%f\n", sum);
}
p.close();
}
catch (Exception e)
{
System.err.println ("Error writing to file - BBE_COSTINGS");
}
}
return sum;


};

public SimulationBBE() {
}
}
```

V – GA OPTIMISATION CODE IN JAVA

```
package hotelchocolatmodeling;

import hotelchocolatmodeling.SimulationBBE;
import java.io.FileOutputStream;
import java.io.PrintStream;
import java.util.Random;

public class GAsimpleBBE {

final int ELITISM_K = 2;
final int POP_SIZE = 154 + ELITISM_K; //size of population
Random m_rand = new Random();  // random-number generator
FileOutputStream out; // declare a file output object
PrintStream p; // declare a print stream object


//DEFINE EACH INDIVIDUAL

public class Individual {

public static final int SIZE = 20; //size of the variables
double[] genes = new double[SIZE];
double fitnessValue;
protected double alpha_Retailer;
protected double beta_Retailer;
protected double theta_Retailer;
protected int q_Retailer;
protected double alpha_HCmailorder;
protected double beta_HCmailorder;
protected double theta_HCmailorder;
protected int q_HCmailorder;
protected double alpha_USHotelChocolat;
protected double beta_USHotelChocolat;
protected double theta_USHotelChocolat;
protected int q_USHotelChocolat;
protected double alpha_WholesaleUK;
protected double beta_WholesaleUK;
protected double theta_WholesaleUK;
protected int q_WholesaleUK;
protected double alpha_MiddleEastFranchising;
protected double beta_MiddleEastFranchising;
protected double theta_MiddleEastFranchising;
protected int q_MiddleEastFranchising;

public Individual() {}

public double getFitnessValue() {
return fitnessValue;
}

public void setFitnessValue(double fitnessValue) {
this.fitnessValue = fitnessValue;
}

public double getGene(int index) {
return genes[index];
}
```

```
public void setGene(int index, double gene) {
this.genes[index] = gene;
}

public void randGenes() {
Random m_rand = new Random();
// for(int i=0; i<SIZE; ++i) {
this.setGene(0, m_rand.nextDouble()); //this makes sure that tetha, betha and alpha are between 0 and 1
this.setGene(1, m_rand.nextDouble());
this.setGene(2, m_rand.nextDouble());
this.setGene(3, m_rand.nextInt(1000));
this.setGene(4, m_rand.nextDouble());
this.setGene(5, m_rand.nextDouble());
this.setGene(6, m_rand.nextDouble());
this.setGene(7, m_rand.nextInt(1000));
this.setGene(8, m_rand.nextDouble());
this.setGene(9, m_rand.nextDouble());
this.setGene(10, m_rand.nextDouble());
this.setGene(11, m_rand.nextInt(1000));
this.setGene(12, m_rand.nextDouble());
this.setGene(13, m_rand.nextDouble());
this.setGene(14, m_rand.nextDouble());
this.setGene(15, m_rand.nextInt(1000));
this.setGene(16, m_rand.nextDouble());
this.setGene(17, m_rand.nextDouble());
this.setGene(18, m_rand.nextDouble());
this.setGene(19, m_rand.nextInt(1000));

}

// mutation
public void mutate() {
//for(int i=0; i<SIZE; ++i) {this.setGene(i, this.getGene(i)+ m_rand.nextGaussian()*0.01);}
this.setGene(0, this.getGene(0) + m_rand.nextGaussian()* 0.01); //this makes sure that tetha, betha and alpha are between 0 and 1
this.setGene(1, this.getGene(1) + m_rand.nextGaussian()* 0.01);
this.setGene(2, this.getGene(2) + m_rand.nextGaussian()* 0.01);
this.setGene(3, this.getGene(3) + m_rand.nextInt(21)-10);
this.setGene(4, this.getGene(4) + m_rand.nextGaussian()* 0.01);
this.setGene(5, this.getGene(5) + m_rand.nextGaussian()* 0.01);
this.setGene(6, this.getGene(6) + m_rand.nextGaussian()* 0.01);
this.setGene(7, this.getGene(7) + m_rand.nextInt(21)-10);
this.setGene(8, this.getGene(8) + m_rand.nextGaussian()* 0.01);
this.setGene(9, this.getGene(9) + m_rand.nextGaussian()* 0.01);
this.setGene(10, this.getGene(10) + m_rand.nextGaussian()* 0.01);
this.setGene(11, this.getGene(11) + m_rand.nextInt(21)-10);
this.setGene(12, this.getGene(12) + m_rand.nextGaussian()* 0.01);
this.setGene(13, this.getGene(13) + m_rand.nextGaussian()* 0.01);
this.setGene(14, this.getGene(14) + m_rand.nextGaussian()* 0.01);
this.setGene(15, this.getGene(15) + m_rand.nextInt(21)-10);
this.setGene(16, this.getGene(16) + m_rand.nextGaussian()* 0.01);
this.setGene(17, this.getGene(17) + m_rand.nextGaussian()* 0.01);
this.setGene(18, this.getGene(18) + m_rand.nextGaussian()* 0.01);
this.setGene(19, this.getGene(19) + m_rand.nextInt(21)-10);

//make sure the generated individuals are between the required intervals alpha/beta/theta [0,1] and q [0,1000]
double cteMin = 0.0;
double cteMax = 1.0;
double qMin = 0.0;
double qMax = 1000.0;

if (this.getGene(0) < cteMin) {this.setGene(0, cteMin);}
if (this.getGene(0) > cteMax) {this.setGene(0, cteMax);}
```

```
if (this.getGene(1) < cteMin) {this.setGene(1, cteMin);}
if (this.getGene(1) > cteMax) {this.setGene(1, cteMax);}

if (this.getGene(2) < cteMin) {this.setGene(2, cteMin);}
if (this.getGene(2) > cteMax) {this.setGene(2, cteMax);}

if (this.getGene(3) < qMin) {this.setGene(3, qMin);}
if (this.getGene(3) > qMax) {this.setGene(3, qMax);}

if (this.getGene(4) < cteMin) {this.setGene(4, cteMin);}
if (this.getGene(4) > cteMax) {this.setGene(4, cteMax);}

if (this.getGene(5) < cteMin) {this.setGene(5, cteMin);}
if (this.getGene(5) > cteMax) {this.setGene(5, cteMax);}

if (this.getGene(6) < cteMin) {this.setGene(6, cteMin);}
if (this.getGene(6) > cteMax) {this.setGene(6, cteMax);}

if (this.getGene(7) < qMin) {this.setGene(7, qMin);}
if (this.getGene(7) > qMax) {this.setGene(7, qMax);}

if (this.getGene(8) < cteMin) {this.setGene(8, cteMin);}
if (this.getGene(8) > cteMax) {this.setGene(8, cteMax);}

if (this.getGene(9) < cteMin) {this.setGene(9, cteMin);}
if (this.getGene(9) > cteMax) {this.setGene(9, cteMax);}

if (this.getGene(10) < cteMin) {this.setGene(10, cteMin);}
if (this.getGene(10) > cteMax) {this.setGene(10, cteMax);}

if (this.getGene(11) < qMin) {this.setGene(11, qMin);}
if (this.getGene(11) > qMax) {this.setGene(11, qMax);}

if (this.getGene(12) < cteMin) {this.setGene(12, cteMin);}
if (this.getGene(12) > cteMax) {this.setGene(12, cteMax);}

if (this.getGene(13) < cteMin) {this.setGene(13, cteMin);}
if (this.getGene(13) > cteMax) {this.setGene(13, cteMax);}

if (this.getGene(14) < cteMin) {this.setGene(14, cteMin);}
if (this.getGene(14) > cteMax) {this.setGene(14, cteMax);}

if (this.getGene(15) < qMin) {this.setGene(15, qMin);}
if (this.getGene(15) > qMax) {this.setGene(15, qMax);}


if (this.getGene(16) < cteMin) {this.setGene(16, cteMin);}
if (this.getGene(16) > cteMax) {this.setGene(16, cteMax);}

if (this.getGene(17) < cteMin) {this.setGene(17, cteMin);}
if (this.getGene(17) > cteMax) {this.setGene(17, cteMax);}

if (this.getGene(18) < cteMin) {this.setGene(18, cteMin);}
if (this.getGene(18) > cteMax) {this.setGene(18, cteMax);}

if (this.getGene(19) < qMin) {this.setGene(19, qMin);}
if (this.getGene(19) > qMax) {this.setGene(19, qMax);}
}

// evaluate fitness function
```

```
public double evaluate() {

SimulationBBE sim = new SimulationBBE();// CALL THE OBJECT SIM OF CLASS SIMULATION

double fitness = 0;

//calling for variables alpha + theta + betha + Q (similar to quadratic funtion a= this.getGene(0), b = this.getGene(1), c = this.getGene(2))
sim.setParameters(this.getGene(0), this.getGene(1), this.getGene(2), this.getGene(3), this.getGene(4), this.getGene(5), this.getGene(6), this.getGene(7), this.getGene(8), this.getGene(9), this.getGene(10), this.getGene(11), this.getGene(12), this.getGene(13), this.getGene(14), this.getGene(15), this.getGene(16), this.getGene(17), this.getGene(18), this.getGene(19));
fitness = sim.calculateProfit(false);
if (fitness<1)
fitness = Math.exp(fitness - 1);
this.setFitnessValue(fitness);

return fitness;
}
}

//DEFINE THE POPULATION

// population size
final static int MAX_ITER = 200;            // max number of iterations
final static double MUTATION_RATE = 1.0;    // probability of mutation
final static double CROSSOVER_RATE = 0.7;    // probability of crossover


Individual[] m_population;
double totalFitness;

public GAsimpleBBE() {

m_population = new Individual[POP_SIZE];

// init population
for (int i = 0; i < POP_SIZE; i++) {
m_population[i] = new Individual();
m_population[i].randGenes();
}

// evaluate current population
this.evaluate();

Individual[] newPop = new Individual[POP_SIZE];
Individual[] indiv = new Individual[2];

// current population
System.out.print("Total Fitness = " + this.totalFitness);
System.out.println(" ; Best Fitness = " + this.findBestIndividual().getFitnessValue());


FileOutputStream out; // declare a file output object
PrintStream p; // declare a print stream object

//EXPORT THE DATA

try{
// Create a new file output stream connected to "HotelChocolatGAtest.txt"
out = new FileOutputStream("HotelChocolatGAtestBBE.txt");

// Connect print stream to the output stream
p = new PrintStream( out );
```

```
// main loop
int count;
for (int iter = 0; iter < MAX_ITER; iter++) {
count = 0;

// Elitism
for (int i=0; i<ELITISM_K; ++i) {
newPop[count] = new Individual();
for (int j=0; j<Individual.SIZE; ++j) {
newPop[count].setGene(j, this.findBestIndividual().getGene(j));
}
count++;
}

// build new Population
while (count < POP_SIZE) {
// Selection
indiv[0] = this.rouletteWheelSelection();
indiv[1] = this.rouletteWheelSelection();

// Crossover
if ( m_rand.nextDouble() < CROSSOVER_RATE ) {
indiv = crossover(indiv[0], indiv[1]);
}

// Mutation
if ( m_rand.nextDouble() < MUTATION_RATE ) {
indiv[0].mutate();
}
if ( m_rand.nextDouble() < MUTATION_RATE ) {
indiv[1].mutate();
}

// add to new population
newPop[count] = indiv[0];
newPop[count+1] = indiv[1];
count += 2;
}
this.setPopulation(newPop);

// reevaluate current population
this.evaluate();

//genes of the best individual
System.out.print("Generation = " + iter);
System.out.print(" ; alpha_Retailer = " + this.findBestIndividual().getGene(0));
System.out.print(" ; beta_Retailer = " + this.findBestIndividual().getGene(1));
System.out.print(" ; theta_Retailer = " + this.findBestIndividual().getGene(2));
System.out.print(" ; q_Retailer = " + this.findBestIndividual().getGene(3));
System.out.print(" ; alpha_HCmailorder = " + this.findBestIndividual().getGene(4));
System.out.print(" ; beta_HCmailorder = " + this.findBestIndividual().getGene(5));
System.out.print(" ; theta_HCmailorder = " + this.findBestIndividual().getGene(6));
System.out.print(" ; q_HCmailorder = " + this.findBestIndividual().getGene(7));
System.out.print(" ; alpha_USHotelChocolat = " + this.findBestIndividual().getGene(8));
System.out.print(" ; beta_USHotelChocolat = " + this.findBestIndividual().getGene(9));
System.out.print(" ; theta_USHotelChocolat = " + this.findBestIndividual().getGene(10));
System.out.print(" ; q_USHotelChocolat = " + this.findBestIndividual().getGene(11));
System.out.print(" ; alpha_WholesaleUK = " + this.findBestIndividual().getGene(12));
System.out.print(" ; beta_WholesaleUK = " + this.findBestIndividual().getGene(13));
System.out.print(" ; theta_WholesaleUK = " + this.findBestIndividual().getGene(14));
System.out.print(" ; q_WholesaleUK = " + this.findBestIndividual().getGene(15));
```

```
System.out.print(" ; alpha_MiddleEastFranchising = " + this.findBestIndividual().getGene(16));
System.out.print(" ; beta_MiddleEastFranchising = " + this.findBestIndividual().getGene(17));
System.out.print(" ; theta_MiddleEastFranchising = " + this.findBestIndividual().getGene(18));
System.out.print(" ; q_MiddleEastFranchising = " + this.findBestIndividual().getGene(19));
System.out.print(" ; Total Fitness = " + this.totalFitness);
System.out.println(" ; Best Fitness = " + this.findBestIndividual().getFitnessValue());

for (int i = 0; i < POP_SIZE; i++) {
p.format("%d,", iter);
p.format("%d,", i);
p.format("%f,", this.m_population[i].getGene(0));
p.format("%f,", this.m_population[i].getGene(1));
p.format("%f,", this.m_population[i].getGene(2));
p.format("%f,", this.m_population[i].getGene(3));
p.format("%f,", this.m_population[i].getGene(4));
p.format("%f,", this.m_population[i].getGene(5));
p.format("%f,", this.m_population[i].getGene(6));
p.format("%f,", this.m_population[i].getGene(7));
p.format("%f,", this.m_population[i].getGene(8));
p.format("%f,", this.m_population[i].getGene(9));
p.format("%f,", this.m_population[i].getGene(10));
p.format("%f,", this.m_population[i].getGene(11));
p.format("%f,", this.m_population[i].getGene(12));
p.format("%f,", this.m_population[i].getGene(13));
p.format("%f,", this.m_population[i].getGene(14));
p.format("%f,", this.m_population[i].getGene(15));
p.format("%f,", this.m_population[i].getGene(16));
p.format("%f,", this.m_population[i].getGene(17));
p.format("%f,", this.m_population[i].getGene(18));
p.format("%f,", this.m_population[i].getGene(19));
p.format("%f\n", this.m_population[i].getFitnessValue());
}

}
p.close();

}
catch (Exception e)
{
System.err.println ("Error writing to file - SIMULATION_BBE");
}
// best indiv
Individual bestIndiv = this.findBestIndividual();

}

public void setPopulation(Individual[] newPop) {
// this.m_population = newPop;
System.arraycopy(newPop, 0, this.m_population, 0, POP_SIZE);
}

public Individual[] getPopulation() {
return this.m_population;
}

public double evaluate() {
this.totalFitness = 0.0;
for (int i = 0; i < POP_SIZE; i++) {
this.totalFitness += m_population[i].evaluate();
}
return this.totalFitness;
}
```

```
public Individual rouletteWheelSelection() {
double randNum = m_rand.nextDouble() * this.totalFitness;
int idx;
for (idx=0; idx<POP_SIZE && randNum>0; ++idx) {
randNum -= m_population[idx].getFitnessValue();
}
return m_population[idx-1];
}

public Individual findBestIndividual() {
int idxMax = 0;
int idxMin = 0;
double currentMax = 0.0;
double currentMin = 1.0;
double currentVal;

for (int idx=0; idx<POP_SIZE; ++idx) {
currentVal = m_population[idx].getFitnessValue();
if (currentMax < currentMin) {
currentMax = currentMin = currentVal;
idxMax = idxMin = idx;
}
if (currentVal > currentMax) {
currentMax = currentVal;
idxMax = idx;
}
if (currentVal < currentMin) {
currentMin = currentVal;
idxMin = idx;
}
}

//return m_population[idxMin];      // minimisation
return m_population[idxMax];        // maximisation
}

public Individual[] crossover(Individual indiv1,Individual indiv2) {
Individual[] newIndiv = new Individual[2];
newIndiv[0] = new Individual();
newIndiv[1] = new Individual();

int randPoint = m_rand.nextInt(Individual.SIZE);
int i;
for (i=0; i<randPoint; ++i) {
newIndiv[0].setGene(i, indiv1.getGene(i));
newIndiv[1].setGene(i, indiv2.getGene(i));
}
for (; i<Individual.SIZE; ++i) {
newIndiv[0].setGene(i, indiv2.getGene(i));
newIndiv[1].setGene(i, indiv1.getGene(i));
}

return newIndiv;
}


}
```