Management of a high mix production system with interdependent demands: modeling of stochastic demands and the concept of virtual profit as a decomposition tool

by

Maria Carolina Serra
Laurea in Ingegneria Automatica (2007)
Sapienza University of Rome

Submitted to the Department of Mechanical Engineering
In Partial Fulfillment of the Requirements for the Degree of

Master of Engineering in Manufacturing

at the

Massachusetts Institute of Technology

September 2009

Signature of Author...
Department of Mechanical Engineering
August 7, 2009

Certified by...
Stanley B. Gershwin
Senior Research Scientist, Department of Mechanical Engineering
Thesis Supervisor

Accepted by...
David E. Hardt
Chairman, Committee on Graduate Students

Management of a high mix production system with interdependent demands: modeling of stochastic demands and the concept of virtual profit as a decomposition tool

by

Maria Carolina

Serra

Submitted to Department of Mechanical Engineering

on August 18, 2009, in Partial Fulfillment

of the Requirements for the Degree of

Master of Engineering in Manufacturing

ABSTRACT

An optimized framework for the inventory control of a high mix production system has been designed in order to guarantee the optimal mix of items in stock in presence of correlated demands. The Virtual Profit concept was developed to measure the criticality of an item in presence of correlated demands. The introduction of the Virtual Profit in the optimization problem allowed the problem to be decomposed and the optimal control parameters to be computed separately. Demands were modeled based on the stochastic properties of the historical demand so that simulations could be performed using statistically generated orders. The simulations provided a validation of the proposed technique showing that, with the same size of inventory, considering the Virtual Profits instead of the real profits improves the quality of the solution, especially when low levels of inventory are kept.

Thesis Supervisor: Stanley B. Gershwin

Title: Senior Research Scientist of Mechanical Engineering

Acknowledgments

I gratefully acknowledge Instron Corporation for giving me the opportunity to realize this project. I thank all the workers at the Norwood facility, for their help, trust and friendship. In particular I whish to express my gratitude to my company supervisor, Kerry Rosado, for his generous and unfailing support.

I would like to express my appreciation to my thesis advisor, Professor Stanley Gershwin, for his understanding, support and guidance throughout the writing of this thesis.

Finally I want to thank my project group mates, Alberto, Diego and Samarth, for sharing this invaluable experience with me.

Index

Acknowledgments 5
Index 6
1. Introduction 9
  1.1 Instron Corporation as a Research Environment 9
  1.2 Background 12
  1.3 Significance of the Problem 14
    1.3.1 Significance of the Project 14
    1.3.2 Significance of the Study 15
  1.4 Review of Prior Instron Projects 16
2. Problem statement 18
  2.1 Project Objectives 18
  2.2 Designing the Optimal Inventory Policy 19
3. Literature Survey 23
  3.1 Introduction 23
  3.2 The (Q,r) Policy 23
  3.3 Correlated demand and the inventory management problem 24
    3.3.1 Correlated demand and job-fill rate 24
    3.3.2 Correlated demand and joint replenishment 25
    3.3.3 Previous work with different assumptions 26
  3.4 Customer defection 26
  3.5 Simulation 28
  3.6 Conclusion 29
4. Methods 30
  4.1 Choosing the right methods 30
  4.2 Main steps followed 31
  4.3 Explanation of the tasks 33
    4.3.1 Individual demand analysis or Pareto analysis 33
    4.3.2 Correlated demand analysis and Comparison 34
    4.3.3 Lead time, holding costs and space constraints 35
    4.3.4 Customer satisfaction 35
    4.3.5 Inventory levels 36
    4.3.6 Optimization 37
    4.3.7 Simulation 38
  4.4 Data collection methods and IBS 38
  4.5 MATLAB implementation and reusability 40
    4.5.1 The need for a tool 40
    4.5.2 Reusability 40
    4.5.3 Frequency of stock determination 41
    4.5.4 Matlab implementation, reusability and flexibility 42
5. Analysis of the Correlation 43
  5.1 Introduction 43
  5.2 Methods 44
    5.2.1 The Optimization Problem 44

5.2.2 The Virtual Profit 45
5.2.3 The Pareto Analysis 49
5.3 Results 51
5.3.1 Virtual Profit and Pareto distribution 51
6. Simulation of the Demand 55
6.1 The demand simulator 55
6.2 Validation of the simulated demand 60
7. The Control Policy 63
7.1 The Q,r Policy 63
7.1.1 Definitions and Type I Service Level 63
7.1.2 Expected inventory level 65
7.1.3 Type II Service Level 67
7.2 Finished goods inventory control problem 68
7.3 The Proposed Policy 70
7.4 Reorder Quantity Q 72
7.5 Reorder Level R 72
7.5.1 Implemented policy for OTC 72
7.5.2 Implemented policy for Systems 73
7.5.3 Sum rule 74
7.6 Simulation Results 76
7.6.1 Validation of the Policy 76
7.6.2 Robustness to Variations 76
7.6.3 Comparison between results using Virtual Profit and using profit 79
7.7 Implementing a Tool 80
8. Results and discussion 84
8.1 Raw materials inventory 84
8.1.1 Results 84
8.1.2 Discussion 86
8.2 Finished goods inventory 87
8.3 Simulation 90
8.3.1 Validation 90
8.3.2 Robustness analysis 91
9. Recommendations 93
9.1 Introduction 93
9.2 Discussion 95
9.2.1 Updating inventory levels 95
9.2.2 Shift in demand 96
9.2.3 Dividing the analysis 96
9.2.4 Lead times accuracy and negotiation 97
9.2.5 Product categories 98
9.2.6 Warning messages 98
9.2.7 New products and substitutions 98
9.2.8 Selecting the best solution 99
9.2.9 Using and adjusting the recommended quantities 100
10. Future work 101
10.1 Lead time variability 101

10.2 Manufacturing constraints ... 101
10.3 Include back orders in the simulation ... 102
10.4 Include part level into the simulation ... 102
10.5 Q,r policy using Poisson distributed demand ... 102
10.6 Category-wise Optimization ... 103
Appendix ... 104
References ... 124

1. Introduction

1.1 Instron Corporation as a Research Environment

Founded in 1946, Instron® is the recognized worldwide market leader in the materials testing industry, holding more than 50% of the market share. The company has various products with all of them sharing production lines. The products cover the following areas of testing: fatigue, tension, compression, flexure, hardness, impact, torsion, spring, test analysis, structural and custom testing. Within each of these categories, many combinations of machines and accessories (hereafter called systems) are possible according to the customer's requirement. That is, all the testing equipment can be customized by the customer. Thus, even the same requirement of two customers may not result in the same order.

Such market behavior forces Instron to keep multiple product lines which further translate into a high inventory, low output factory floor. Thus, Instron serves more than a 'job-shop' volume but at the same time maintains a flexible manufacturing facility to produce highly customized products in minimum time. This issue is clearly visible in the accessories business of the electromechanical division. This area of the production line has the maximum variability and hence is an effective bottleneck. It is well known in the inventory management industry that rather than high demand, it is the variability that is the real reason behind the difficulty in managing service levels [1]. Thus, it is very important to make this area ready for such variability. This can only happen if the right mix of accessories is available at the right time, in the right quantities and at the right place.

Variability is not the only concern while dealing with the inventory in the EM accessories business. There are other intricacies involved which make the problem more challenging. For example, not only can the finished goods be sold as part of a system but they can also be sold as individual after sales parts (hereafter called as OTC - Over The Counter products). Secondly, each system has to wait until all the items in it are available and only then it can be shipped.

Thus, in the case of a system order, there is dependence of demand between these items, and they cannot be viewed as separate entities.

Figure 1.1 - 5800 Series System

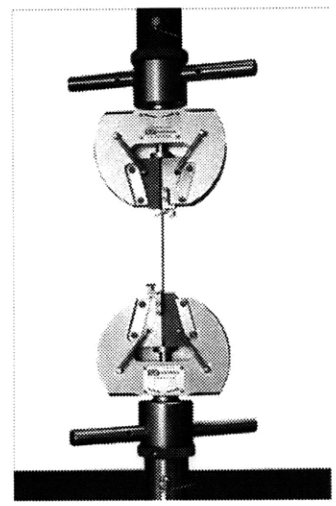


Figure 1.2 – A Wedge Action Grip

The figures above show some products offered by the Electro-Mechanical business. Figure-1.1 shows a 5800 Series System. It includes a double column machine with accessories- grips and computers. Orders comprising this whole package i.e. machine with accessories is called a system order. Figure1.2 shows a similar grip. These grips (and other accessories) can also be sold separately from the whole system and such orders are the Over the Counter (OTC) orders. Figure 1.3 shows different accessories that can be a part of the system.

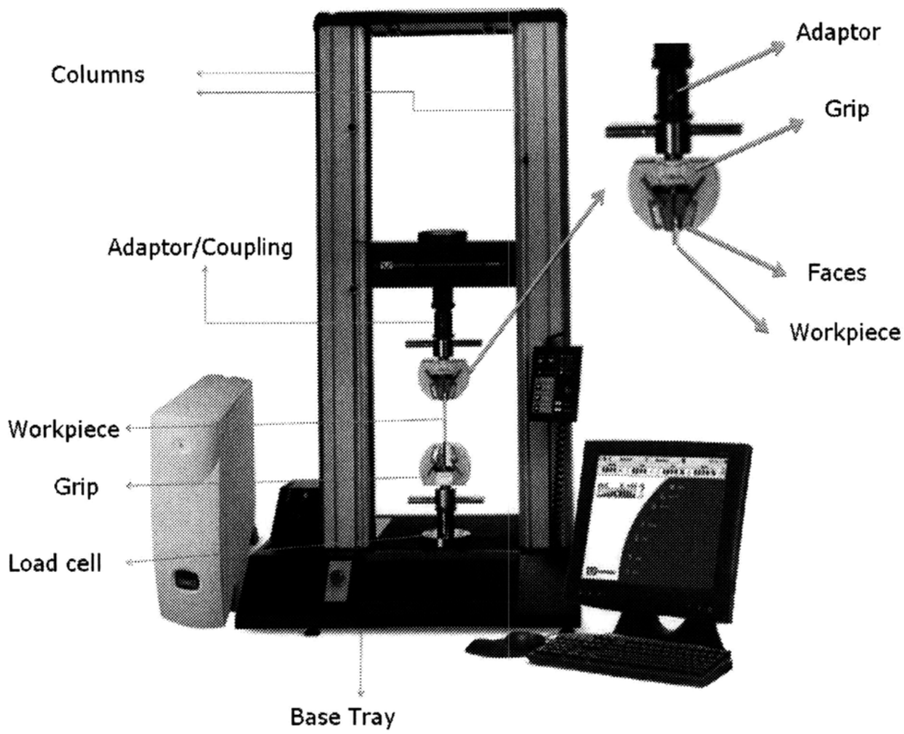


Figure 1.3 - An Instron Two Column Machine with Accessories

Currently, Instron holds an inventory value of $4million in the EM area with inventory control based on Distribution By Value (classifying products into categories A, B, C and D according to their cost) and ITW's policy of having a maximum of 2 months-on-hand demand. However, many aspects are neglected while determining their inventory policy- such as demand variability, percentage of lost sales, holding costs and customer expectations. Thus, there is a certain opportunity to scientifically determine the inventory levels taking all the significant factors into account and improving the customer satisfaction by fulfilling more orders as well as minimizing inventory levels.

1.2 Background

Generally, Instron's finished products can be classified into two categories: systems and OTC. In the electromechanical division, both of these categories exist and share a common inventory. Due to the demand variability, the management has decided not to base the inventory control on predicted demand but to switch to a pull production strategy in order to allow production to reorder parts only when finished goods are "pulled" away from the system. The physical implementation of pull production is achieved through the use of Kanban[2] cards for some of the purchased parts and components, and by not stocking some inventory items at all. However, Kanban is not available at the level of final finished goods level yet and has been implemented only 40% at the part level. Some finished goods are currently being replenished according to minimum level reports being generated through the internal inventory management system. This means that the goods are replenished only when a report is run and hence they are more prone to inaccuracies. Some other goods are being replenished by visually seeing on the floor if the quantity drops below a minimum mark, triggering a development order by the area manager. This method too can be inaccurate.

No records for lost orders are kept. Thus, it becomes difficult to determine which item causes the order to be lost. The available data shows only the orders which were fulfilled and hence, it acts as a barrier in determining the optimum inventory level since the actual demand will be underestimated.

The suppliers can also overlap i.e. one item can be bought from two different suppliers. This complicates the case further since there will be two lead times for the same product.

Finally, the final lead time to customer is also hard to determine due to a system audit which takes place on certain products and takes about a day to complete.

Figure 1.4 – The finished goods stocking area

Currently, sometimes during peak demand, the factory floor gets clogged with the unfinished machines. Also, occasionally, when a whole order is made to wait longer, it gets cancelled, even if just one item was not available.

The markets of OTC and system orders have their own special requirement. While on the system side, the customers are more relenting and are willing to accept larger lead times, the OTC market is more demanding. The customers prefer expedited delivery since they are just waiting for one component in their system. Hence, the OTC market is very competitive.

The system market is usually more relaxed as Instron machines are expensive and they come as a capital purchase for their customers. Hence, the customers understand the large lead times for the machines. For a capital purchase, customers themselves need time to get the money sanctioned from their own organization. This too helps to mitigate the dissatisfaction due to high lead times. For the same reason, Instron starts building the product as soon as it gets an order. However, it does not ship it until all the bills have been cleared.

The systems market helps the OTC market by ensuring that customers buy only Instron accessories which are specifically designed for their machines. Easily available cheaper duplicates require extra adaptors and are not backed up with warranty. Despite this, customers want lowest possible lead times in the OTC products.

1.3 Significance of the Problem

The significance of the project for Instron and the contribution of this study to the literature in the field of inventory management are shown in the following paragraphs.

1.3.1 Significance of the Project

The number of items and parts concerning the Assembly Department is around 1000. In this situation, a great waste of time and money can easily be caused by overstocking. On the other hand Instron's responsiveness to customer demand is identified as an important goal in order to maintain competitive advantage. The optimization of the control parameters is thus critical at the accessories area at Norwood. In order for the strategy to remain optimal in the future, the control parameters must be adjustable accordingly to variations in the product line and in the demand. It is also important that the proposed inventory strategy is easy to apply for the planners and the workers of the facility to properly control the stocking of so many items. The impact of proposing an effective inventory control strategy consists in improved production efficiency and better competitiveness on waiting times which is especially important for the OTC market.

1.3.2 Significance of the Study

This work considers the case of low-volume high-mix inventory systems where customer orders may require several different products (i.e., high customization between products and hence demand between different products is correlated) and the shipment of those items cannot be split. The time delay seen by the customer is the performance measure of concern and the customer impatience is modeled and taken into account: whenever one or more items belonging to an order are backlogged, the customer is quoted a waiting time which is as long as the slowest item's lead time. As the waiting time increases, a customer is less prone to make the order. A continuous review model is proposed using historical sales data rather than using forecasted demand.

Interdependent demands frequently arise in real life multi-item inventory systems. The dependencies of demands for different inventory items may be implied by product options or kits. When the manufacturing lead times for some accessories are long or when customer order assembly time is small, the configuration of a proper mix of items is critical to ensure their availability with the desired probability and avoid order fulfillment delays. Ignoring correlation in the demand when present may lead to two possible consequences: stocking more than necessary or not being able to provide the desired service level. It is demonstrated by R. Zangh that this assumption leads to an overestimate of the total time delay when items are actually correlated [2].

Unfortunately most inventory models on time delay in the literature assume one-item orders. The resources available in the literature which consider interdependence in the inventory planning can be split in two main categories:

- Studies about joint replenishment take advantage of the correlation of the demand to minimize the ordering or setup costs and transportation costs. Unfortunately these techniques are not useful when items are provided by many suppliers. As described in the Introduction, for what concerns the case studied here, accessories are both manufactured in-house as well as ordered from a large number of outsider suppliers.
- A small number of studies describe similar problems but under different conditions. In particular some of them assume that parts belonging to the same order can be shipped

separately to the customer if some item is not immediately available. Other works consider other inventory control models.

1.4 Review of Prior Instron Projects

In the past ten years three MIT graduate students have completed research internships at Instron working on inventory control and operations management. The theses of D. Wheeler, G. Caterino and H. T. Nguyen are outlined below.

The purpose of Wheeler was to optimize the EM grip inventory by applying queuing theory, optimization techniques, supply chain rationalization and simulation models [3]. In particular the author, together with a project improvement team, achieved a thirty-percent reduction of the inventory for the 56 EM grips belonging to the Instron product line at that time. They implemented a pull production in the grip assembly job shop by setting up stock shelves for finished goods and components within the shop from which the parts were removed to fill the orders. When the level of finished goods drops below a specified quantity (the reorder quantity) the mechanic is signaled to replenish it. Moreover as the components to build the grips, which are drawn from the bins on the shelves, drop below the reorder point, the planner receives a signal and replenishment orders are placed. Reorder quantities and lot sizes for the finished grips and some components were provided by the Economic Order Quantity (EOQ) and the continuous review (Q,r) models. These models were applied on the most significant components which had been identified by applying the Distribution By Value (DBV) technique [4]. Items were classified as belonging to three different Classes (A, B and C). The most valuable components (Class A and B) were placed under the Q,r control policy, while reorder quantities and reorder points for items belonging to Class C were set respectively to one year's supply and six months' supply for each item.

The second thesis objective was to improve the responsiveness and flexibility of the assembly process applying elements of Lean Manufacturing [5]. With the use of Kanban control in assembly, daily production schedules based on demand rate and decision rules to guide the work process, the assembly throughput times have been reduced by 40% on average in the final assembly operations. Changes to the physical assembly environment have been made in order to increase flexibility of the output. The author proposes an inventory policy to coordinate in-house inventory levels with manufacturing demand and improve the coordination with external suppliers. The policy, similarly to Wheeler's work, is based on a (Q,r) model and DBV and is tailored on a small number of finished good items (three selected product families). Its application on a pilot process showed a 15% reduction in the required floor space for an equivalent manufacturing output.

Nyugen in his work has tried to improve the service level by implementing lean initiatives in the plant [6]. Root cause and Value chain analysis were carried out in the plant to find opportunities for improvement. A material replenishment model was proposed that would help the company effectively pull parts from the suppliers. Lot sizes were determined according to extended economic order model quantities adjusted using Lagrange multiplier to account for multiple parts being manufactured at the same time. For the inventory control, continuous review policy is proposed for the EM business so that low safety stock can be kept and probability of stock out can be reduced.

In the next sections, the problem has been cleared defined qualitatively and quantitatively. Literature review for the work has been summarized in the next section. It highlights all the text that was helpful in understanding and interpreting the problem better. Next, the methodology to study the problem has been introduced which introduces the thought process used to develop the approach and then the steps that were followed, how data was collected and how it was interpreted. Finally, the problem was solved using the method highlighted in the above mentioned section and results obtained. These results after proper validation are discussed in the results section with some recommendations.

2. Problem statement

The project goal, shared among the four group members' theses, is the definition and implementation of an inventory control framework for the EM accessories stored in the Norwood facility. The result of this work is enabling the inventory planners of the Configuration Department to stock the optimal mix of accessories in order to guarantee a satisfactory service level to the customers and minimize the inventory cost.

2.1 Project Objectives

The project specifications provided by Instron are listed below.

1) Analyze the accessory level offerings based on customer demand and sales volume.
2) Determine finished goods inventory level for each accessory.
3) Develop and implement an internal finished goods replenishment model based on a *pull* strategy.
4) Coordinate with Supply Chain group to insure Kanban quantities support for the finished goods model.
5) Identify and procure any needed tooling.
6) Determine and implement any layout changes.
7) Measure and monitor results.
8) Make it visual and involve factory employees.
9) Identify key performance indicators.

2.2 Designing the Optimal Inventory Policy

In order to meet the specifications, the problem has been modeled and its critical elements have been identified.

A first challenge for this project comes from the large amount of accessories to control: more than 800 finished goods concern the Configuration Department and include grips, fixtures, faces, extensometers, couplings, adaptors, computers etc.

Some of them are assembled in the Norwood facility, while some of them are purchased parts or assemblies. The large number of components that constitute each finished item and the large number of vendors that supply Instron represent a further source of complexity for the analysis.

In the previous theses performed at Instron, a simplification of the large amount of parts considered was provided by Distribution By Value (DVB) and 80/20 techniques, which are described in Chapter 4, allowing the authors to focus on the most significant ones in terms of value or profitability. Since the 80/20 analysis is a currently widely used and appreciated tool within the company, the team decided to adopt it to perform an analysis of the demand, measuring volumes and profits.

As described in the Introduction, demand has two components: Systems and OTC. This allows the problem to be split in two separate analyses.

For OTC accessories customers expect immediate shipment. Since the OTC market is more sensitive to competitiveness, an effective control strategy is critical to provide customers with a satisfactory service.

The Systems market, instead, is characterized by longer waiting times expected by the customers and less external competitiveness. However all the parts of the machine must be shipped together, with rare exceptions, and if a part is missing the order is delayed. In fact most of the times customers cannot perform their tests if a part is missing, and in every case splitting the shipment of an order is costly and not desired by the company. In 2008 no more than 4% of the Systems orders got split and this percentage is meant to decrease.

While the OTC market can be analyzed considering individual profits and volumes for every item, an accurate model of the Systems demand should take into consideration the inter-correlation among products. This suggests that the demand analysis for systems should also account for the importance of an accessory as purchased together with critical items. The *Virtual Profit* is an index based on combined profits developed by the team to model the inter-dependence of the demands and it is presented in the paragraph 4.3.2.

Since the waiting time expectations for the two markets are different, the inventory levels for the same items must satisfy the two demands. The problem can be thus decomposed in two analyses for the different markets. Once both the stocking quantities are set for both demands a risk pooling strategy can be implemented by aggregating those results.

For both the markets, once the 80/20 analysis has provided a measurement of the criticality of the items within the product list, the proper inventory control policy for the items must be identified. Constraints to this project are given by the fact that the Norwood stocking capacity is limited and the inventory allowed by the Instron management is less than 2 MOH[1] (Months on Hand) for every item. Thus in order to maximize the customer satisfaction and so the profit, the basic strategy is implementing two different control policies for two different classes of accessories:

- The most critical items will be assembled or purchased to stock so that high service levels will be achieved.
- The less profitable items will be assembled or purchased to order, minimizing their inventory costs.

However the optimal division between items deserving to be stocked and items that will be made to order needs to be found. Another parameter to be set is the desired *Type I service level*, or percentage of customers that will be immediately served, for the first class items. Wheeler [3] suggests to favor the "80" items (those items that concur to the 80% of the total profit/volume or

[1] Months on Hand = 12 (Average Inventory Value on Hand / COGS)

Correlation) and provide them with 0.95 Type I service level. Unfortunately there are two reasons why this is only a suboptimal solution:

- The 80/20 curves usually show one or more steps in the distribution of volumes or profits, so that the division between most important and less important items is quite clear. This is also valid if the quantity measured is the Correlation. However the step does not necessarily occur at the 80% of the cumulative profit: its position can vary depending on the situation. Setting the threshold at 80% would lead only to a suboptimum.
- The 0.95 Type I service level was set accordingly to the Instron management which found it reasonable. However assigning a constant service level for all the make to stock parts is certainly not the optimal strategy.

This issue can be addressed designing an optimization problem which would allow splitting the items in the two classes in an optimal manner, setting at the same time the service levels for the for the first class items.

There are several factors that the problem must take into account. Firstly storing parts has a cost in terms of space, handling and cash blocking, in general referred to as *holding cost*, which has to be minimized. Moreover there are items which are more worthy to be stored than others because give a larger profit (on their own or being sold with other items). In order to consider the described issues the stock level for each item i will be determined by maximizing the expected total profit generated by that item. A model of the expected total profit is given by the expected revenue minus the expected total costs.

The expected revenue for each product can be found by multiplying its unit cost by its expected sales $E(S_i)$, which are a function of the demand rate and the number of items in stock. Note that the past and future expressions of the demand are not available since the sales lost because of the waiting time quoted are not registered and forecast is not used at Instron. Historical sales are the only information that can be found. For the purpose of this project we assume that the expected demand is equal to the past sales. The effects of this assumption are mitigated by the pull strategy that (Q,r) represents causing the actual demand to drive the inventory control once the control parameters are chosen.

Moreover, since customers are willing to wait a variable amount of time if the parts are not immediately available, sales are also function of the *delay acceptability* w_i, or the percentage of customers that would still buy the item if it is not in stock.

Currently, the production lot sizes or reorder quantities are determined based on their value and historical demand without taking into consideration the lead times. Though suppliers have a negotiated contract with the company, they are usually supportive of the lot size requirements. In order to guarantee the selected service levels to the customers, one of the components of the solution consists in making sure that these quantities are enough to satisfy the demand over lead time with satisfactory probability.

Finally, the raw materials control is evaluated. Based on the finished goods production, the raw materials inventory control has to be synchronized and the parts have to be available with high probability. An optimized policy is proposed in order to guarantee the necessary support to the finished goods replenishment model. The optimized policy requires knowing the suppliers' replenishment lead times; this requires data collection and accuracy. The raw materials control is evaluated by comparison with the current policy.

The resulting optimal strategy is evaluated in its costs and benefits: a simulation tool is designed in order to test and validate the control policy and compare it with the current situation.

In order for the finished good inventory policy to be implemented and utilized by the Instron workers in the future, the control parameters must be periodically computed and adjusted. For this reason the analytical tools used for this work are designed for reusability and robustness, as well as easiness of use and compatibility with the data and tools available at Instron. The tools must take into consideration adjustments for new products introduced in the product line and for dismissed ones. In fact the introduction of a new series of accessories with a partial substitution of some old one has occurred this year and can occur again in the future. The implementation of the strategy in the Configuration Department, including the physical arrangement of the stock bins and the Kanban cards, and the training of the workforce are part of this work, are part of this work, in order to guarantee that the strategy is correctly understood and continued.

3. Literature Survey

3.1 Introduction

Since our first contact with the problem, it was clear to us that its set of features and objectives made it a very particular challenge. The theory we learnt from classes and from Simchi-Levi et al. 2000 [7] guided us to the choice of a (Q,r) policy but the standard set of assumptions used to determine the parameters Q and R did not fit our problem. In particular the correlation between the demand of the various products, the fact that many items could be sold both alone and in a system order, the fact that a system order cannot be shipped unless all the items are available and the fact that customers have different expectations on acceptable lead times for different items required a new approach to solve the problem. Many of these challenges are somehow considered in literature but often with a different objective and anyway, to our best knowledge, they have been never considered together. In 3.2 we briefly discuss the vast literature about the (Q,r) policy which constitutes the basis of our work; in 3.3 we present papers which faces the demand correlation issue; in 3.4 we argue about the usage of some papers regarding the customers' expectation issue; in 3.5 some references about simulation are presented.

3.2 The (Q,r) Policy

In those cases in which the inventory is reviewed continuously (in opposition to periodically) a heuristic control policy which has been well-studied in the last several decades is the so called "Q,r" (sometimes also named r,Q or in other ways). The basic idea is that whenever the number of items held in inventory drops to or below r an amount of Q units of goods is issued to replenish the system. Hadley and Whitin 1963 [8] present an exact solution to the problem when there is a known penalty cost assessed on each unit backordered and they provide, under some assumptions, two approximate iterative heuristic solutions.

During the following decades the Q,r policy has been extensively explored in literature, many of the original assumptions have been relaxed and many of its properties proved.

In particular important convexity results are given in Zipkin 1986 [9] and Federgruen and Zheng 1992 [10] and the existence of such results justify the research of optimum values. Also, interesting convexity results are proved in Wang and Li 2007 [11] for the discrete demand and inventory case.

3.3 Correlated demand and the inventory management problem

3.3.1 Correlated demand and job-fill rate

Demand correlation among different items and its effect on inventory policies is a key aspect of this work. Even though it is common in real-life multi-item inventory systems, this phenomenon has not received a large attention in the existing inventory literature. We were able to find some papers related to the problem we are facing but none of them could directly be used in this case either because they pose different objectives or they are firmly based on a set of assumptions which does not apply to Instron case.

One of the first papers to focus on similar problems is Smith and Chambers 1980 [12]. In such work in fact it is introduced for the first time the concept of "job-fill" (in opposition to "part-fill") rate criterion in this context. The paper deals with the determination of the appropriate collection of parts to be carried out to repair a machine. As in our case if only one part is missing the order cannot be completed (the machine cannot be repaired). In that case the cost associated with not being able to complete a given job due to unavailable parts is related to a longer downtime for the machine (the repairer has to go back to the warehouse and return on site again), in our case it is tied to the customer unsatisfaction and the resulting risk of losing the order. Such problem was already known at the time as the "fly away kit problem" or the "submarine provisioning problem", however these previous papers traded off shortages against part-fill rate instead of order-fill rate. Smith and Chambers [12] is then an interesting article but doesn't consider all the issues present in our case because the correlation is not considered as the failures of different part types is assumed to be independent. However, other than for the "job-fill" rate criterion, [12] is very useful to us also for a theorem about the importance of ranking the items before considering an optimization problem.

Using Smith and Chambers' "Job-fill" rate criterion, Zhang 1999 [13] studies the expected time delay in multi-items inventory systems. In such paper the demand is assumed to be correlated across items and customer satisfaction is measured by the time delays seen by the customers. As a result, an exact expression for the expected total time delay is derived. Also, it is shown that when items are actually correlated, assuming items are independent leads to an overestimate of the total time delay. This however assumes that the parts can be sold separately if some of them are not in stock. In this sense it is shown that demand correlation is in fact an opportunity that should be exploited. In our case, because an order cannot be shipped unless all the parts are available, the demand correlation is an issue.

3.3.2 Correlated demand and joint replenishment

The point of view presented in [13] is common to many other papers that deal with correlated demand. In fact many papers who consider demand correlation are focused on joint replenishments policies such as Liu et Yuan 2000 [14], Feng et al. 2007 [15] and Tsai et al. 2009 [16]. In particular [14] specifically considers the can-order policy for a two-item inventory system with correlated demands. Unfortunately joint replenishment doesn't specifically help with the problems that Instron want to solve in its EM department and, even though it can still be beneficial, its usage would add a large amount of complexity and would allow very small benefits, if any. In fact, as regards items manufactured outside the company Instron has a very large number of suppliers and buys from each of them a very small amount of different products. Moreover, as regards items manufactured inside the company, very small setup costs are involved and the assembly is mostly make-to-order. In other words in the papers which focus on joint replenishment the objective is reaching a balance between ordering costs, storage costs and stockout costs while in our case ordering costs are not significant. The same considerations about joint replenishments also apply to [15] and [16]. Specifically, [15] formulates the problem as a Markov Decision Process and focuses on joint replenishment and correlated demand, proposing a moving boundary based policy and comparing it to other control policies. Tsai et al. [16] instead propose a clustering algorithm to deal with demand correlation which is similar to a first possible solution, later abandoned, that we considered to solve our problem. Such paper claims that it is difficult to define the demand correlation between items, especially when the number of items increases and for this reason a clustering algorithm is proposed. Such algorithm is used to find an optimal clustering result which is used to determine the parameters of a can-order policy in presence of joint replenishment. The result is then tested through simulation and sensitivity analysis, two steps that are fundamental also in our approach.

3.3.3 Previous work with different assumptions

As said the literature which deals with correlated demand is relatively small and a good part of it is focused on joint replenishment which is not useful in our case. However, some papers are closely related in their intent to our work, although not directly applicable due to different assumptions. Hausman et al. 1998 [17] has very similar problem statement to our as it is said that the objective is to "configure a proper combination of component item inventories so that availability of component items is ensured at pre-specified levels to avoid order fulfillment delays". Unfortunately this paper considers a periodic review order-up-to policy and so is not compatible with continuous replenishment. Anyway the paper contains some very interest ideas and some theorems and lemmas which can be considered also in our case. Very close to our objective is also Wang et Hu 2008 [18] which studies the application of a (Q,r) policy with budget constraints and optional components. The budget constraints, at least in the way they are formulated in [18], are not of primary concern in our case but the approach proposed is still very interesting. Unfortunately, two of their assumptions are not verified in our case: it is not true that the payment is due at the time an order is placed (but this problem could be overcome) and most importantly it is not true that the customer will purchase a system without optional components when the optional components are out of stock. Optional components are in fact, in the majority of cases, necessary to use the Instron machine and no one would buy a machine without them.

3.4 Customer defection

In this work, the effect of customer impatience (or *defection*) on the inventory performance is studied. Two main contributions on this field are used as references: Gershwin et al. 2009 [19] and Veatch [20]. The main reason why this work investigates the customer impatience is that the number of orders filled (in literature *Type II Service level*) depends on how many customers would wait for a product if it were not in stock. In particular, the number of filled orders is the sum of the number of orders filled immediately plus the number of orders completed because the customers decided to wait and not to cancel the order once they were quoted a lead time.

In [19], a manufacturing firm that builds a product to stock in order to meet a random demand is studied. If a product is not in stock and orders cannot be met, customers are quoted a lead time that is proportional to the backlog, based on the production time. In order to represent the customers' response to waiting, a *defection function* - the fraction of customers who choose not to order as a function of the quoted lead time - is introduced. The defection function is then used to obtain the optimal production policy, which results in a hedging point form. One family of defection functions is studied, a sigmoid function of the form:

$$B(x) = \frac{1}{1 + e^{\gamma(x-\eta)}} \tag{3.1}$$

This expression for the defection function is then used to model the system behavior, and will also be used in this work. However, an additional important conclusion is that numerical results suggest that there is limited sensitivity to the exact shape of B(x). Furthermore, the precision of the defection function is limited by the intrinsic approximate nature of what it models, i.e. the customer impatience.

In [20] the same production model, in which the customer is quoted a lead time depending on production time and backlog, is presented as a "nuanced model" of customer behavior, compared to the two extreme models of complete backordering and lost sales, where all the customers either wait or not. One particular production model is considered: a continuous one-part-type, single machine model with Markov modulated demand and deterministic production is considered. Using this particular model, the impact of customer impatience is shown to be captured by one quantity, the mean sojourn time in the backlog states. As in [19], the optimal quantity has hedging point form.

Based on the particular model considered, Veatch shows that the effect of customer impatience can be captured by the only mean sojourn time in backlog, and this simplifies the problem of obtaining an optimal production policy. Given that the effect of customer impatience is captured by the above mentioned quantity, in fact, other simpler customer behavior models can be used, and still the optimal policy is reached.

This thesis analyzes a different model: only some of the products are produced in the factory floor, while most of them are ordered from suppliers. Moreover, the replenishment lead time is random and constraints on the reorder quantities have to be considered. Thus, the assumptions made in [19] and [20] are not valid any more, and the optimization problem is different. Moreover, the two papers do not present any attempt to shape the defection function in the actual industrial application. However, the analyzed work gave some useful insight into the modeling of customer impatience. The suggested sigmoid form is used in this work, and the limited sensitivity to the exact shape of the function is considered. Finally, this thesis considers the use of company-wide surveys in order to shape the defection function to the needed precision level.

3.5 Simulation

Simulation has been used as a validation tool in this work. Monte Carlo is one of the simulation techniques used to validate our results. The principle behind Monte Carlo simulation is that the behavior of a statistic in random samples can be accessed by the empirical process of drawing lots of random samples and observing the behavior [21]. However, care has been taken while generating customer demand. Truncated normal distribution is used to generate demand since it should not go negative in the cases when the coefficient of variation is high [22]. Coakley and Carpenter 1983 [23] have used Monte Carlo simulation to predict final system behavior when it cannot be directly predicted from the inventory models. They validate the model before running the simulation using constant values and matching them with theoretical results. Finally, they use the simulation results to analyze different conditions such as relaxing theoretical constraints and getting the inventory levels.

Jung et al. 2004 [24] have presented a method to determine safety stock levels, which further effect the customer satisfaction levels (service levels), using a computational framework for planning scheduling applications in realistic supply chains. They use simulations to optimize their results when faced with improving customer satisfaction, holding costs and production constraints. Inside the computation for optimization, repeated simulation of the supply chain takes place over the planning horizon, each with given Monte Carlo demand samples. Then, within each of these simulations, a series of planning and scheduling optimization problems are solved.

Grange [25] in his paper pays particular focus to demand distributions of slow moving items. He finds out the misidentifying demand distributions can have a detrimental effect on the fill rate leading to high and lower rates depending on over and under estimation of right tails. He also adds that multi-SKU inventory compensates misidentification by reallocating investment relative to the costs and expected demands of all the SKUs. We have thus, taken particular care in finding out the demand distribution in our case, as highlight in the methods section.

3.6 Conclusion

The problem this work deals with is a particular one and a solution tailored for this case cannot be found in literature. Not many authors focused on demand correlation in multi-items inventory systems and many of them consider a rather different set of assumptions thus being allowed to see it as an opportunity to be exploited using joint replenishment. A few papers which consider a similar problem statement are still not applicable to our case because they differ in some fundamental assumption such as periodic inventory review and optional nature of accessories. Also as regards the customer impatience issue the papers analyzed do not provide a univocal methodology to be used in our practical case but they contain very interesting ideas and results. Simulation was also found to be frequently used both as tool to find a solution and as tool to validate the result found with another method.

In conclusion our problem requires a new solution in order to deal with all its features but the existing literature constitutes a fundamental basis to our work with its ideas, theorems, reasoning and methods.

4. Methods

4.1 Choosing the right methods

The goals of this project are described in detail in chapters 1 and 2. One sentence summarizes them effectively: "having the right mix of products on the shelves at the right time". As mentioned before, this involves searching an optimal inventory control and production policy by considering all the products together, especially taking into account the system orders, thus the correlation among items' demands.

The significant number of items involved and the differences in their supply chains added high levels of complexity to the project. Not only do we want to have the correct "mix" on the shelf, but the implementation of the derived policies will differ depending on the product's type and supply chain. Furthermore, using one's own judgment on each SKU would not provide the company with a repeatable strategy. For these reasons, general and parametric methods always have to be used.

In addition to the optimal policies, important results of the project come from the analysis phase (demand analysis, correlation analysis, customer defection, 80/20). The produced documents, indeed, are important in providing the manufacturing, sales and marketing departments with sources of data which allow effective strategic planning. As an example, knowing which products are often sold together in the last two years, could suggest marketing already customized systems (composed of the products often sold together); if this operation is successful, the company could focus its investment in the inventory for a limited number of products, holding less risk associated with other products. Moreover the results of the analysis performed by the team and provided to the company find an application in the identification of products to discontinue because of their scarce profitability and importance within the product list.

What is more, in each sub-issue addressed by this thesis, the purpose is not only identifying the optimum (optimal inventory control policy, optimal replenishment levels) but also proposing the so called *"good enough"* solution. As widely happens in manufacturing and operations management, in fact, the application of systematically searched optimal policies holds a level of complexity that is not worth the investment. For instance, considering the optimal replenishment methods, agreeing with the suppliers on the optimal reorder quantities for a product could not be feasible or could involve additional investment, and using a QR policy implemented with Kanban cards, that are already used, would be more easily and quickly implementable than different policies that could guarantee a relatively small increase in expected profit.

In conclusion, the work described in this thesis is meant to produce data analysis reports and suitable solutions for the inventory control policies of a significant number of products. This chapter describes the steps that are undertaken in building the analysis reports, in designing the control policies and in collecting the necessary data for the policies to be implemented. The methods used in each step are briefly described in the following paragraphs and then explained in more details in the following chapters.

4.2 Main steps followed

Figure 4.1 shows the main steps involved in the project. Every independent task is represented by a blue filled circle, while the developed software tools are represented by smoothed rectangles. The arrows indicate task scheduling requirements. As an example, let us consider the following tasks: comparison, individual demand analysis and correlated demand analysis. In order to perform the comparison task, the results from the individual and correlated demand analysis are necessary; thus, these two tasks need to be finished in order for the comparison task to be performed. The diagram is a modified version of the PERT diagram which does not show the duration of the tasks.

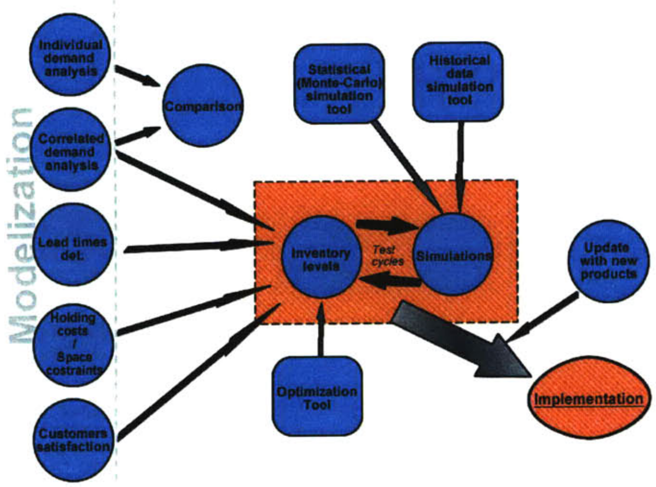


Figure 4.1 - Tasks Diagram

As previously mentioned, the main outcome of the project consists in data analysis reports and recommendations for inventory control policies. The most important reports are obtained in the steps *Individual demand analysis*, *Correlated demand analysis* and *Comparison.* In these three steps, demand analysis of all the involved products is performed, at first simply by volume and profit, and then considering how they correlate to each other. Finally the results are collected in a Comparison report, meant to underline the importance of the correlation. The step *Inventory level* involves designing the control policies, while the performance of these policies are estimated in the step *Simulations* and implemented in the step *Implementation.* The importance of these two final steps is highlighted by the orange box in the diagram.

The left side of the diagram shows the steps needed in modeling the system. In order to design the inventory control policies, the following information is needed: lead times for each product, profit and correlation analysis, holding costs, space constraints and a model of the customer satisfaction. All this information builds the model of the system, used to find the optimal solutions.

The remaining part of this chapter describes the goals of each task, the approach to it and the methods used.

4.3 Explanation of the tasks

4.3.1 Individual demand analysis or Pareto analysis

This task involves analyzing the orders placed in 2008 and 2009. The list of orders, together with the associated quantities and prices, is used to perform a demand analysis based on both profit and volume. The purpose of this analysis is to find the most important products and the least profitable ones. The results are useful to the company in showing the updated data on volume and profit made by the products during the last two years.

The *Pareto principle* (also called *80/20 principle*) is a heuristic principle that is often applied in analyzing profit and volume in operations management (the Pareto analysis). Applied to profit, it states that about 80% of the profit of a company is made by only 20% of the products it sells. The products belonging to that 80%, which are the most profitable ones, are called the *80s*, while the remaining products are the *20s*.

For the purpose of this analysis, the products are divided in six different categories: grips, fixtures, faces, coupling and adapters, compression anvils and anvil sets and other accessories. The first step of the analysis involves summing up the profits made by each product in all the orders and determining the total quantity shipped in each year. A report has been given to the supervisor, in which the most profitable items were identified through the Pareto analysis. In addition to this, the least profitable items were highlighted in the report: all those products which belong to the bottom 1% of the profit or were sold at most twice. This result is important to identify items eligible to be discontinued. However it does not provide a measurement of their criticality within the product list. The Correlation analysis, described in 4.3.1, provides a more accurate result.

For a more expanded discussion of the Pareto analysis, please see Chapter 5 in Diego Palano's master's thesis [26].

4.3.2 Correlated demand analysis and Comparison

As mentioned earlier, the design of the optimal policy is complex because it has to encompass a very high number of different accessories that are often sold together in the system orders (when customers buy a machine and choose a set of accessories with it). Moreover, the above mentioned individual demand analysis is less accurate than necessary because it does not take into account the system orders.

As an example, two products X and Y can be considered. If X is an "80" item and Y is one of the lowest profit items, the individual demand analysis would suggest holding less inventory for item Y or even making it to order. By considering the system orders, however, we could find out that product Y is often sold together with X, and is less profitable because it is discounted or relatively less important. Holding lower inventory levels for item Y would then be a losing strategy, because it would block the orders of X and create additional profit loss.

In this project, the correlation between different products is considered in designing the control policies. The goal is obtaining a profit indicator which quantifies the profit made by each product if in stock, or quantifies the loss realized by not having it in stock for a given period of time. A MATLAB function, using the IBS reports with all the orders of 2007 and 2008, calculates how many times each product is sold with any other item and quantifies this expected profit.

New profit indicators were obtained considering the correlation, and a new analysis report was generated (step *Comparison*). This report shows what are the most profitable items and what are the ones which are still in the bottom 1% of the profit after considering the correlation. As mentioned in paragraph 4.3.1, this report completes the analysis of the items to be discontinued, together with the 80/20 report.

For a more expanded discussion of the correlation analysis, please see Chapter 5.

4.3.3 Lead time, holding costs and space constraints

These three steps involved data collection, which is necessary to design the control policies. The data collection methods, including holding costs and space constraints, are further explained in chapter 4.4 of this thesis.

By working with the supply chain managers and using the IBS tracking system, at first we tried to obtain a list of lead time values for all the products involved in the project. The term "lead time" was used in a more general sense, indicating replenishment lead times for purchased parts, manufacturing run time for manufactured or assembled parts, and collecting time for catalog numbers that actually are a kit of items. In general, the term, lead time, indicated the total time needed for a product to be again on the shelf when required.

4.3.4 Customer satisfaction

In order to maximize the expected profit, the loss for a part not being on the shelf has to be quantified. Let consider the case, however, in which one particular SKU is not on the shelf. The customer would learn that a particular product was not on the shelf and that the total waiting time would be n weeks. Would he still go on with the order? And what if the order request was actually for a system including that product?

In general, there will always be a number of customers who will still buy a product even if the order cannot be fulfilled from stock and a longer waiting time is quoted. This percentage depends on the product and on the type of order, and is a function of the quoted waiting time. This function is referred to as "customer defection". The literature background about customer defection is discussed in chapter 3.

Obtaining this quantity from the data or in any rigorous way is not feasible due to the following reasons:

- Lack of hard data about lost sales
- Customers have different interests, priorities, concerns
- Other reasons (human behavior, complex products interdependence)

Thus, a reasonable estimate is obtained through a survey directed to the sales people, who work on orders with the customers. The starting expression of the customer defection function is a sigmoid, as discussed from the literature, and the function is further shaped by asking general questions and looking for ranges of values through the survey. This function represents the percentage of customers still willing to wait depending on the waiting time that can be offered on one particular item.

For a more expanded discussion of the customer defection analysis, please see Chapter 6 in Diego Palano's master's thesis [26].

4.3.5 Inventory levels

This task involves designing the production and inventory control policies for both finished goods and raw materials.

Two main types of policies are used: make to stock and make to order. The less profitable items will hold lower service levels or be made to order, while for the remaining products stock levels are determined. The choice of the MTO or MTS policy for each item is based on optimizing the profit, and is described in *4.3.6*.

The most suitable make to stock inventory control policy is the QR policy (or *reorder quantity*). One reason is that the inventory at Instron has always been managed through two quantities: the so called *minimum quantity*, corresponding to the safety stock, and the reorder quantity. Even if these quantities were obtained with rules of thumb, they are used to set a safety stock level and reordering when the levels go below the minimum quantities. Moreover, an increasing number of parts are being managed by *Kanban cards*, which is an automatic inventory replenishment method. When the inventory level reaches a minimum quantity, the corresponding card is put on a board and it will automatically trigger the order of a predetermined release quantity from the suppliers. This system is easily updatable once the new optimal values for Q (reorder quantity) and R (reorder point) are derived.

The reorder quantities are determined in such a way that they cover the demand over lead time with a probability of 99.87%, still satisfying eventual constraints on the lot sizes. The optimal reorder points, on the other hand, are calculated from the lead times, the average demand, the values of Q and the desired service levels. While lead times and average demand are obtained in the data collection phase, the service levels represent our degrees of freedom in designing the policy. For the finished goods inventory, these levels were chosen by optimizing the profit, as described in *4.3.6*. The raw materials inventory control, instead, is designed in such a way that the service levels are always high, in order to support the finished goods production.

For a more expanded discussion of the raw materials inventory control, please see Chapter 7 in Diego Palano's master's thesis [26].

4.3.6 Optimization

The available degrees of freedom in designing the FG inventory control policy are given by the service level corresponding to each item (Type I service level, defined as the percentage of time the inventory for a certain item will not be empty, thus being able to meet demand) and whether each product will be made to stock or made to order (MTS or MTO).

These choices are determined by solving an optimization problem. The goal function is the total expected profit, defined as total expected profit coming from sales minus the inventory holding costs. The total expected profit coming from sales is calculated considering the correlation between products in the same orders (as described in *4.3.2*), while the inventory annual holding costs per item are multiplied by the expected inventory levels in the QR policy.

The result of the optimization tool, implemented in Matlab, is a list of optimal service levels for all the items. If the optimal service level for a particular product is lower than a certain limit than the final suggestion for it will be a make to order policy.

For a more expanded discussion of the finished goods policies optimization, please see Chapter 5 in Alberto Facelli's master's thesis [27].

4.3.7 Simulation

An important step in studying the optimal control policies is the simulation phase. It allows us to test the designed strategy in order to check its feasibility and to estimate its performance measures (actual service level obtained, months on hand of average inventory).

The simulation tools are used both as design aid and as final performance measurement that helps in selling the proposed recommendations. The simulations are implemented in two different ways: at first simulating random demand with a discrete probability distribution with the actual mean and standard deviation (plus intra-quarter growing average), then by using the actual historical data. The former tests the policy for robustness with a more general background; the latter shows a comparison between the results of the proposed policy and the current one.

For a more expanded discussion of the simulation of the proposed policies, please see Samarth Chugh's master's thesis [28].

4.4 Data collection methods and IBS

Most of the tasks undertaken in modeling the system involved hard data collection from the databases of the company. Referring to the diagram in picture 1, these tasks are:

- Individual demand analysis;
- Correlated demand analysis;
- Lead times determination;
- Holding costs / Space constraints;
- Customer satisfaction;
- Historical data simulation tool;
- Update with new products.

The holding costs are obtained from the operations manager and head of manufacturing and through some financial research on cost of capital; the space constraints are estimated talking to the managers and exploring the factory floor. The information about the new products (new item numbers, discontinued items, updated demand forecast) was obtained from to the engineers in charge of the corresponding projects.

The model of customer satisfaction is firstly defined based upon literature and suggestions from the operations management. Then, the model is shaped and refined through a company-wise survey, filled by the sales department and the field engineers, who are the ones involved in the customer satisfaction aspect of sales.

All the remaining tasks involve collecting data from Instron's databases:

- previous years' sales
- product types
- inventory locations
- costs and prices
- replenishment lead times
- manufacturing run times and set-up times
- current reorder points and quantities

The necessary information is collected through IBS. IBS is an Instron database management system that tracks all the information associated with orders and products. For each order placed by customers, IBS contains order number, dates, quoted lead times, standard costs, gross price, discounts and a number of other entries. For each product, IBS contains item number, bill of materials, information about suppliers and planners, current inventory levels and limited inventory level history, lead time and a number of other entries.

IBS is used in all the departments in the company. The sales people, when dealing with customers, use IBS to get the expected lead times, to check what is available in stock, to check prices and costs and to handle orders. The employees working in the factory floor update it when parts arrive from suppliers, when products are shipped, when changes are made to the orders, when WIP inventory is used and a part is assembled and in several other cases. Moreover, all the other employees often use IBS to get required information for analysis purposes or to update it.

In order to collect the needed data, reports are automatically generated by IBS. IBS can be queried with a list of items or orders, and the required information is written on Excel spreadsheets. The result is that every analysis or manipulation which starts from the generated spreadsheets can be easily repeated and updated by using the same type of queries.

4.5 MATLAB implementation and reusability

4.5.1 The need for a tool

The goal of the project at Instron is not only to provide a numerical solution to the problem of which control policy and which parameters should be used. Also, a fundamental goal is to provide a long term solution framework, so that, year after year and quarter after quarter, a new numerical solution can be computed and used. In fact one has to consider that every product has a certain life cycle and that the demand for each of them changes over time. Therefore, it is clear that the "determination of the right mix" is not something that can be determined once. On the contrary, a regular update of the safety stocks levels and inventory control policies parameters is necessary.

For this reason, since the beginning of the project the research team focused on creating a tool that could be used in the research and that then Instron could use in the future to make the calculations and update the policies regularly.

4.5.2 Reusability

The way we see the solution framework is depicted in Table 4.1. On a periodic basis (the choice of the frequency is discussed briefly in the next paragraph) Instron personnel will update the inventory levels. In order to do this, they will export all the relevant past sales data from IBS (the ERP software they are currently using) to an Excel file using a template that we built in IBS. Then, in a similar way, a list containing the lead times, the lot sizes and other information regarding the items will be extracted from IBS. Finally these XLS files will be put into the same folder as our software tool (an EXE file) and by just running it a solution will be computed.

The output will be composed of three files. The first one is an Excel file containing the information that should be used for the Kanban cards, that is to say the reorder quantities and the reorder point that has just been determined. The second file is a *Correlation report* that is to say a description of the items that were most often sold together which is useful for Instron personnel to understand the demand and what drove the suggested inventory levels. Finally, the third output is an *80/20 report* in which the items are divided by category and ranked by their virtual profit. Also this report will help to explain to the people the re-order quantities determined by the tool and it will also suggest which items can be suppressed without losing, both directly and indirectly, much profit.

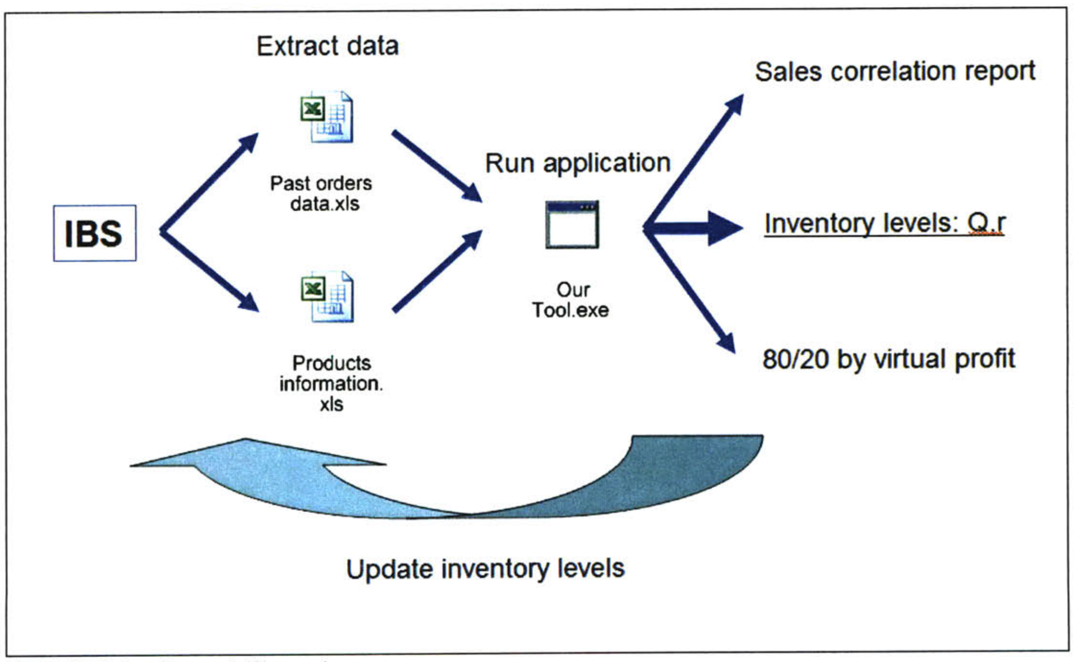


Table 4.1 – Reusability scheme

4.5.3 Frequency of stock determination

There is a trade-off in the frequency with which the inventory levels should be re-computed. In fact, on one hand the higher the frequency with which the inventory is re-determined, the best the inventory levels will theoretically perform because they will use the most recent demand information. On the other hand, re-determining the levels involves a certain effort from Instron staff and represents a cost that can balance the advantage of using more recent data. To determine the new levels in fact some data has to be gathered as described above and the computation has to be started. Then the resulting suggested reorder quantities has to be compared with the ones currently in use. If an "R" needs to be updated, then the Kanban card currently used for that item must be reprinted and substituted on the bin.

As seen, a trade-off exists and the correct time does determine new levels depend on the effort necessary to physically update the inventory levels. As a first guess, we think a frequency of 3 or 6 months seems reasonable, unless some of the determining factors (the demand or the lead times for example) will at some point drastically change.

4.5.4 Matlab implementation, reusability and flexibility

The tool described above is built in the Matlab environment and then compiled as an executable file. The choice was suggested by our familiarity with such environment and its power and abundance of mathematical functions. As regards the part of the code which deals with data crunching a C code would have probably been faster but in such a language the optimization part would have been harder to code and, overall, the time required to build the tool and test it would have been much longer. Because in our case the quickness with which the tool was to be built is very important while the computation time required for every run is not particularly significant (as seen the tool is going to be run a few times per year), the choice of Matlab seems to be the best one.

Moreover, Instron owns many Matlab licenses for other reasons so such software is and will be available to the company without any added cost. This is an important issue because, even though we want to give an "easy to use" – "black box" solution, we also want to provide the source code that could be checked and modified in the future and while to run the exe file Matlab is not necessary, to modify the source code is.

5. Analysis of the Correlation

5.1 Introduction

This chapter presents the contribution of the author to the project, the modeling and analysis of the correlations between products. The focus of this thesis is hence directed to the study of the interdependencies of the demands for the Systems market as opposed to the independency of the OTC orders.

Since the correlation in the demands has an effect on the inventory dynamics, not taking it into consideration while designing an inventory management framework for the Norwood facility would only provide suboptimal solutions. When, in fact, more than one product is required by the customer in the same order, an effective inventory strategy should guarantee the simultaneous availability of all these items.

Virtual Profit, a measurement of the correlation between products is here presented for the first time. This quantity has been derived by the author in collaboration with the other team members and Prof. S. Gershwin, the advisor of this work [1].

The applications of the Virtual Profit in the project include:

- Representing a model of the demand to be integrated in the optimization problem which is used to determine the inventory level for every item and defined by A. Facelli [2]
- Making possible for the company to periodically perform a Pareto analysis of the products which takes into account their relationships, in order to identify the most critical ones.

The first application represents the foundation of the derivation of the Virtual Profit, and is presented in more detail in paragraph 5.2.2 while the other application, which follows from a collateral but interesting property of the Virtual Profit, is described in section 5.2.3.

The validation of this approach, and thus of the strategy proposed by the team, takes place in the simulation of the strategy compared to an analogue approach which does not consider the interdependencies in the demands, and to the current policy used in Norwood. Both historical sales data (which are interpreted as historical demand) and statistically generated demands are used. A model of the demand is needed as an input for the policy simulator presented by S. Chugh [3].

The derivation of the demand model is presented and, in turn, validated in Chapter 6.

Finally the implementation of a tool for the finished goods replenishment optimization is presented in Chapter 7. The tool has been built to allow Instron to perform the computation of the control parameters for the proposed policy as frequently as it is needed.

5.2 Methods

5.2.1 The Optimization Problem

The Virtual Profit is a mathematical quantity derived in order to enable the team to optimally determine the stock levels for the finished goods considering the Systems market. As mentioned in Chapter 2 in fact, because of the differences between Systems (or machines) and OTC (Over the Counter) demands, the two markets can be analyzed separately and the solution of the problem can be split into two parts.

Two approaches to the optimization problem for Systems are presented briefly here and are described in detail in Facelli [2]: the Virtual Profit has an important role in the definition of the second one.

In both versions of the optimization problem, the goal is to maximize the expected total profit generated by all the items i in order to determine their service levels, which are, in other words, the variables of the optimization. The profit is composed of two terms: the profit coming from the sales and the inventory holding costs which are subtracted to it. The sales profit, in turn, is obtained as a theoretical maximum profit, the one reachable if all the products were always in stock, multiplied by a percentage of fulfilled sales. The sales profit is thus a key quantity for the optimization problem and another way to describe it is as the opposite of the total lost that the company would suffer not having in stock the items analyzed.

The first approach is complex because it presents only one objective function in which all the profits are evaluated together. In fact, because of the correlation of the demands, one product's sales profit not only depends on that item's service level, but also on some other accessories' service levels. This means that whether or not each item is in stock affects other items' profits. For this reason, all the items profits' appear together in the objective function and the optimization is done on all of them at the same time. This causes the solution to be extremely time-consuming and even unfeasible when the items to evaluate are several hundreds, like the ones considered for this project.

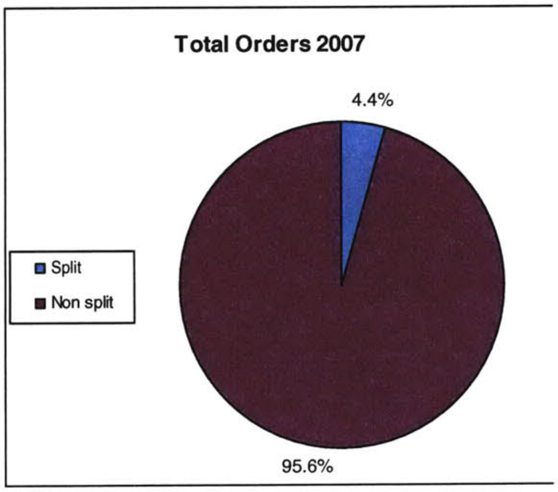
The second approach is intended to provide a simplified version of the first one, where the objective function is split and all the variables are analyzed separately, resulting in a much improved agility of the computation. In order to achieve this simplification without losing the information about the inter-relationship of the variables, the theoretical maximum profit for each item is substituted with an aggregated quantity which has in it all the information about the correlation of the demand. In other terms, the sales profit for every product i results from the multiplication of two factors:

- A quantity representing how much profit the company would lose not having item i in the product list at all
- The percentage of fulfilled orders containing item i

5.2.2 The Virtual Profit

The Virtual Profit (V_i) is a scalar quantity and is defined for every finished product i in a set of items Ω. The primary goal for which it is designed is to provide an answer to the question "Without modifications in the demand, how much money would the company lose if item i was not in the product list?"

The basic hypothesis underlying the concept of Virtual Profit is that no order can be shipped if not all the included components are ready. In the Systems orders reality this is generally true because customers are often not able to use a machine without the accessories. As shown in Figures 5.1a and 5.1b, only a small percentage of the orders gets split because the customers may already own Instron products and, when there is a delay in the production of some item, they may decide to ask for splitting the shipment, in order to get the ready parts and start using them.



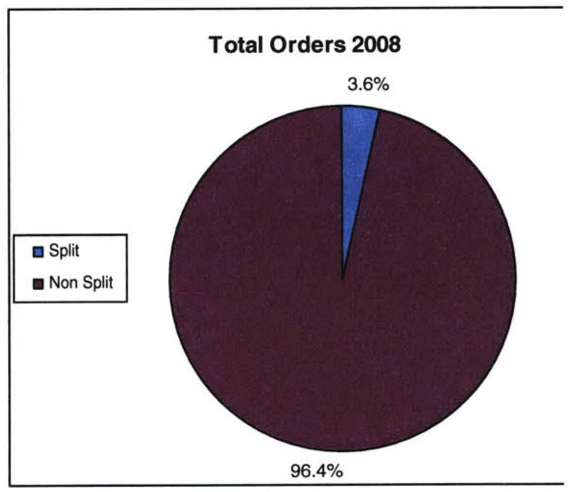


Figure 5.1a - Percentage of split orders in 2007

Figure 5.1b - Percentage of split orders in 2008

However, since splitting orders represents an extra cost for the company and it is meant to be avoided in the future, the assumption is reasonable.

Assuming the demand is known, the damage for the company of not having a product i available can be measured as the loss of all the demand orders containing that item. Demand orders are the ones obtained adding up the sales realized and the orders lost because items were not available on time. In other terms, the orders that are lost because of other reasons than the waiting time to the customers, such as an unacceptable price, are not considered.

As previously stated, demand forecasts are not used in Norwood for the inventory management. The team disposes of records of the past sales while records of the orders lost because of an unacceptable quoted waiting time or because of delays are not available. Therefore the demand orders that occurred in the past cannot be completely estimated and, for the purpose of this project, the past orders sales are directly interpreted as orders demand. This does not lead the accuracy of the analysis because only a small percentage of the orders is claimed to be lost because of availability and waiting time reasons and the past demand is not significantly different from the recorded sales.

At this point V_i, the Virtual Profit for an item i, can be mathematically formulated.

Defining:

- $j = 1,2,\ldots,n$ and $i = 1,2,\ldots,n$ where n is the total number of items of the product list.
- N_{ij} is the number of times item j has been sold in the same order with i. Note that j can be equal to i, and N_{ii} simply indicates the number of times item i has been sold. Note that N_{ij} can be zero when the corresponding items have never been sold together.
- p_j is the unit profit made by item j

The damage or the profit *virtually* lost because of the unavailability of item i can be expressed as:

$$V_i = \sum_{j=1}^{n} N_{ij} p_j$$

(1)

In other words V_i is the total profit made by the orders containing item i.

When no more than one of each item is sold in every order, a symmetric matrix M can be constructed, with elements N_{ij}, representing the frequencies with which every item is sold with others. The elements of the diagonal would indicate the number of orders containing the corresponding product and the sum of the elements of a given row i, all the elements N_{ij} with j varying for all the items in the product list Ω, excluding the element belonging to the diagonal N_{ii} would be less or equal than N_{ii}. This intuitively follows by the fact that one item is always sold in the same order with itself. A vector containing the Virtual Profits for every product ($\underline{V}$) would be simply obtained by multiplying the frequency matrix by a vector $\underline{p}$ containing the unit profits.

$$\underline{V} = M \cdot \underline{p} \qquad \textbf{(2)}$$

When instead quantities of products greater than one can be sold at a time, as for example cables or pins, the matrix would not be symmetrical. In this case, the derivation of the frequency matrix M might be tedious. A simple algorithm is presented in order to simplify this operation and reduce the computation time.

```
//Initialize

Virtual Profit = 0 (for all the items)

//Run

For current_order from 1 to total number of orders

    For item i from 1 to the total number of items in the product list

        If item i is in current_order

            Virtual Profit of item i = Virtual Profit of item i + profit generated by current_order.
```

Figure 5.2 – Pseudocode for the computation of the Virtual Profit

Note that the profit derived from the sale of a single product in a single order is counted in all the Virtual Profits of the items sold in that order. Because of this multiple-counting, the total damage of not having available more than one item cannot be calculated as the summation of their Virtual Profits but a different method must be adopted.

In fact, this is not the purpose for which the Virtual Profit is designed: because of its capability to measure the importance of a single item considering only indirectly the other items effects, it is used in the formulation of an optimization problem in which every item can be analyzed separately.

5.2.3 The Pareto Analysis

The Pareto principle and the 80/20 analysis have been presented in paragraph 4.3.1. The principle applies to the Instron products for both their profits and their sales volumes as demonstrated by Palano [4]. For instance, Figure 5.2 shows this concept for the grips sorted by percentage of the total volume. The graph, through the vertical bar dividing the most profitable items from the less profitable ones, illustrates how the 80% of the total volume is provided by a small percentage of grips. It also shows a long tail of items providing a negligible contribution to the total profit. In the graph, the grips are numbered from 1 to 87.

The Virtual Profit is a quantity which accounts for both the profitability and the sales volume of the item at the same time, because it increases in relation to the product's profit level and its sales frequency. Therefore it is reasonable to expect that the Pareto principle applies when the measured quantity is the Virtual Profit.

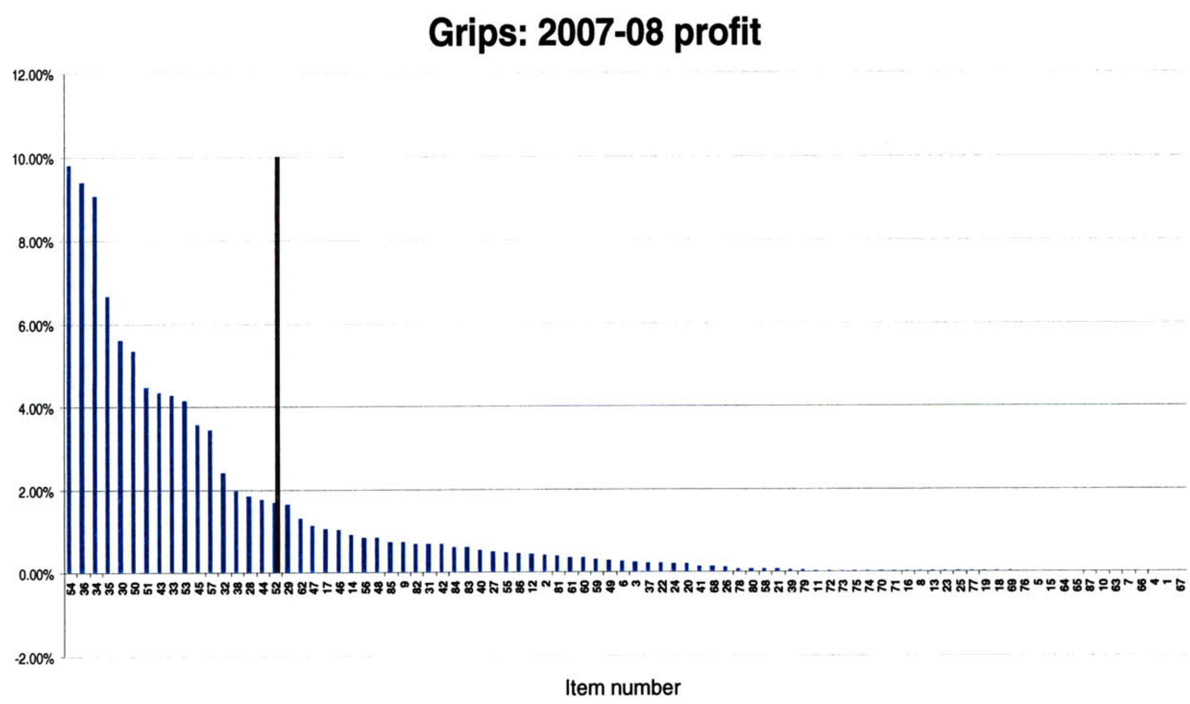


Figure 5.2 – 80/20 chart for grips by profit

Furthermore, from the way it is formulated, it is reasonable to expect that most of the items belonging to the "80s" in the profit analysis performed will remain in the "80s" if the quantity considered is the Virtual Profit.

Meaningful differences are found in those items which because of low prices or high costs have low unit profitability. They are ranked in a low position in the 80/20 chart by profit and a high one when the volume is considered: these analyses alone cannot provide a good interpretation of the importance of that item because it is not clear which value should be prevalent. In this case, the Virtual Profit provides an answer: the item will be ranking high if it is sold many times with top profitable items. This is to say, the Virtual Profit measures how much the product is critical within the products list.

The opposite can also occur: an item from the "80s" in the conventional 80/20 is dropped to the "20s" if it has low importance in the sales of other products.

As can be seen by Figure 5.2, many items do not provide a contribution to the total profit at all. Because there are costs associated with the complexity of the products list, it might be worth it to eliminate some items from it. A cost/benefit analysis of this operation is beyond the scope of this work, but the Virtual Profit provides a criterion for this selection from a manufacturing point of view: the items belonging to the corresponding 80/20 chart would be eligible to be discontinued without significantly affecting the sales profits. However the threshold value for this selection (e.g. the items making together less than the 1% of the profit) should be evaluated by the management team and the decision should be made according to the customer needs.

For safety reasons, when the three types of Pareto analyses described earlier are performed in order to decide which products to discontinue, the suggestion is to use the intersection of the sets of items in the tails. In this way, the items having an importance from whatever point of view do not risk being removed.

The results of the 80/20 analysis considering the Virtual Profits are shown in paragraph 5.3 and compared with the conventional ones determined by profits and volumes.

5.3 Results

5.3.1 Virtual Profit and Pareto distribution

As expected the Virtual Profits calculated for all the Instron accessories based on the 2007 and 2008 sales data and averaged over the two years are distributed following a Pareto Distribution. Figure 5.3 displays the 80/20 analysis of the grips performed using this quantity.

The 30% of the grips contribute to the 80% of the total Virtual Profit.

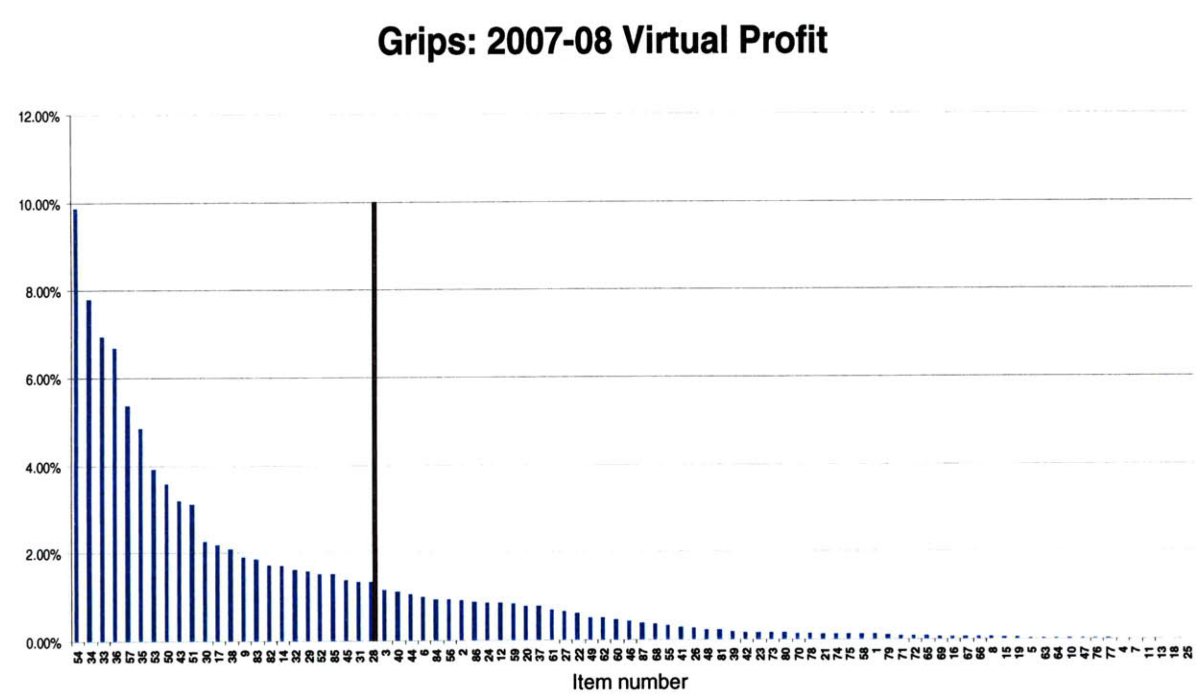


Figure 5.3 - 80/20 chart for grips by Virtual Profit

Moreover, the most profitable group in both the analysis by Virtual Profit and by profit presents many items in common. Table 5.1 provides a comparison of the two techniques, showing the first 28 items from two lists of item numbers respectively sorted by profit and by Virtual Profit. The "80" items are highlighted in green in both groups. As can be seen in the table, the number of "80s" in the analysis by Virtual Profit has increased by about 40%. As expected, most of the items have changed their position from one chart to the other. All the "80s" in the traditional analysis except one, item 44, are also considered "80s" in the one by Virtual Profit. Moreover the items 17, 9, 83, 82, 14, 29, 85 and 31 appear among the "80s" by Virtual Profit but not in the ones by profit.

Items by profit	Items by Virtual Profit
54	54
36	34
34	33
35	36
30	57
50	35
51	53
43	50
33	43
53	51
45	30
57	17
32	38
38	9
28	83
44	82
52	14
29	32
62	29
47	52
17	85
46	45
14	31
56	28
48	3
85	40
9	44
82	6
31	84

Table 5.1 – 80/20 table by profit and by Virtual Profit (top 30 items)

These differences are expected because the Virtual Profit, by taking into account the inter-relationships by products, attributes greater importance to items based on the frequency they are sold with the most profitable ones.

In order to better understand these differences, the sales data for the items mentioned above can be investigated. Table 5.2 reports, for some grips, the total number of time it has been sold, the products which have been sold in the same order with it and the number of times this occurred. For space reasons, only the three items that are sold most frequently with each grip are showed. The table, also referred to as Correlation Report, shows the records for the grips that from being "20s" in the traditional Pareto analysis become "80s" considering the Virtual Profit (in pink) and for the grip number 44, that becomes "20" from being "80" (in yellow).

Grip	tot times sold	sold with	# times	sold with	# times	sold with	# times
17	54	101	25	102	18	103	17
9	64	104	17	105	16	106	16
83	23	107	17	108	13	109	13
82	28	110	17	111	15	107	14
14	53	102	20	112	16	113	14
29	39	114	24	110	17	115	17
85	17	116	17	117	17	118	12
31	31	110	18	119	13	120	11
44	22	2501-093	21	2701-065	19	2701-004	8

Table 5.2 - Correlation Report for some grips

The items in the report are listed by decreasing value of Virtual Profit. The table shows that the number of times an item is sold, considered alone, is not relevant for the Virtual Profit. For example item 83 is sold less often than item 14 but ranks a higher position. The same is valid for item 44, which is out of the "80s" by Virtual Profit but has a greater sales volume than item 85.

What instead distinguishes the critical items, characterized by higher Virtual Profits, is the profitability of the items they are sold with.

This also suggests that the traditional 80/20 analysis by profit and volume are not sufficient, and that by measuring the Virtual Profits more information can be obtained.

6. Simulation of the Demand

6.1 The demand simulator

The testing of the framework for the inventory management proposed in this work is performed with the aid of a simulation tool described in Chugh [3]. The correctness of the policy and its parameters is proved using the historical sales from the last two years as input demand to the simulation tool. Its robustness to variations is tested on various sets of simulated orders. By statistically generating the orders, in fact, the framework can be tested for a large number of situations and its weaknesses can be discovered and adjusted.

Since the proposed framework presents original and heuristic aspects, such as the utilization of the Virtual Profit, the inclusion of the customer impatience in the determination of the optimal service levels, the simulation must be designed in order to properly test those aspects.

This work presents a simulated demand generator, implemented in Matlab, which provides the feed for the policy simulator. In particular, in order to enable the testing of the Virtual Profit technique, the statistically determined demand has to take into account that multiple items can be ordered at the same time. Moreover their arrival rate must follow a realistic probability distribution that can be evaluated using the previous years' sales history.

The proposed demand generator is split in two parts: one emulating the OTC demand and one for the Systems orders.

For the OTC market, orders are assumed to contain only one item. Even if this does not always happen in the reality, it is consistent with the fact that Over the Counter orders can be split and not all the items in the order are required to be available at the same time. The monthly demands for all the Instron accessories are simulated using the mentioned probability distributions and the OTC orders are generated containing these items. The simulation of Systems orders however, presents some non-trivial aspects such as the determination of how many and which items are sold in every order. In fact the number of items in each order is variable and the number of combinations of products that can be sold together is large, while, at the same time, not all the combinations are equally probable. Therefore in order to properly test the proposed replenishment framework and the use of the Virtual Profit, it would not be correct to select in a completely random manner a given number of items within the products list and use them to generate the orders.

The Systems orders generator is here presented in detail; for the other one an analogue but simplified reasoning can be done. It is designed to take into account that products are sold together with certain probabilities; in other words it considers the correlation.

The starting point to build the simulated orders is given by the frames. In fact all the Instron machines are built adding accessories to a frame so that it is completely customized. Every purchasing order made in 2007 or 2008 contains one of the frames sold by Instron.

The steps of the procedure to generate the fake orders for one year are listed below and then are described in detail:

1 For every month the demand for each frame is determined with the aid of a discrete and non-negative probability distribution resulting from the historical data. Each frame derived in this way is assigned to an order which is provided with an unambiguous name.

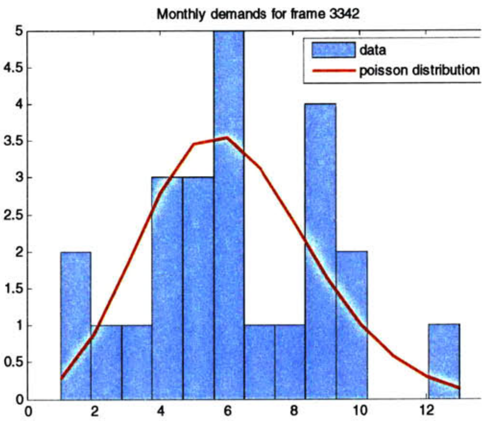
2 Dates are assigned to orders: within the considered month they are randomly selected using a uniform probability distribution.

3 Accessories are randomly assigned to every order accordingly to the probability of every item to be sold with the frame contained in the order.

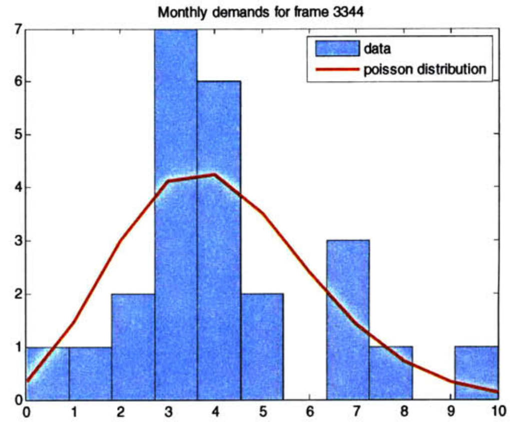
4 A quantity for every accessory is assigned in accordance to the average quantity in which the item has historically been sold.

In order to apply the described process, the collection of some data is required. Firstly the probability distribution for the monthly demand of every frame needs to be analyzed and modeled. Since the incoming demand cannot be negative (although it can be zero), the selected distribution should be non-negative. Moreover the accessories are required in an integer number, so the distribution should be discrete. For these reasons, the fitting of the Poisson distribution is tested with the sales data. In addition to the required features, Poisson is selected for its capability to express the probability of a number of events occurring in a fixed period of time, if these events occur with a known average rate and independently of the time since the last event. The events are in this case the arrivals of the orders in a month. Note that the assumption of independence between the arrivals of orders is reasonable in this case since the customers of a given frame don't interact with each other. The average arrival rates are thus derived for the orders received every month of 2007 and 2008 using the Matlab function *poissfit* which returns the maximum likelihood estimate (MLE)[2] of the parameter of the Poisson distribution. Figure 6.1 shows the comparisons between the historical results and the theoretical expression of the Poisson distribution fitted for those values. The frames presented are the one identified by the code F1, F2, F3, F4. The graphs show that the bell shape of the data is well represented by the fitted solution, and that their expected values and variances are identified. However the graphics also show that the fitting is not exact and the Poisson distribution is not able to represent exceptional peaks, bumps or depressed regions. This can be attributed to the small number of samples; since only two years are evaluated, the number of values available for each frame is 24.

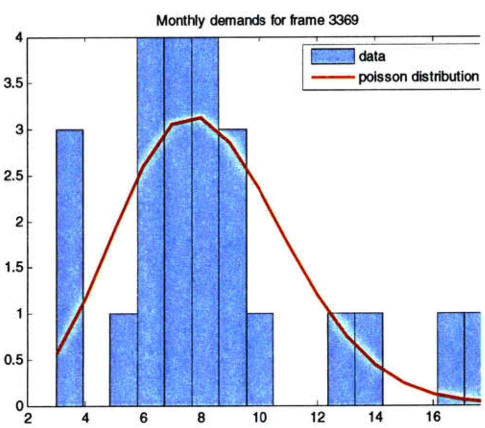
[2] The maximum likelihood estimate is a statistical technique which consists in fitting a statistical model for the data provided, and selecting the parameter value which gives the data the largest possible probability.



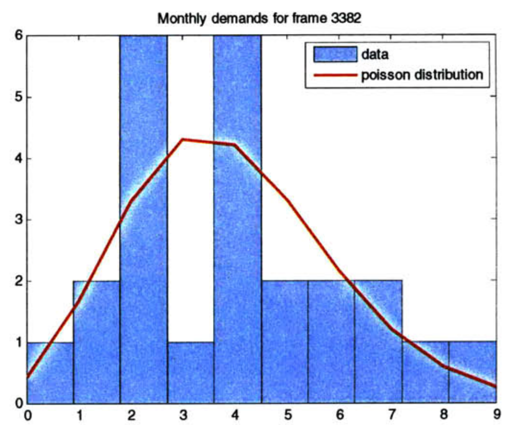
6.1a - Comparison of historical data and fitted distributions for frame F1



6.1b - Comparison of historical data and fitted distributions for frame F2



6.1c - Comparison of historical data and fitted distributions for frame F3



6.1d - Comparison of historical data and fitted distributions for frame F4

Unfortunately increasing the number of years considered in this estimation would not be of much help; year after year new products are introduced and old ones are discontinued. This influences a lot the trend of the demand for the entire products list and undermines the significance of the statistical analysis.

Referring to the fourth step of the orders generation procedure, another set of data which has to be defined is the probability of each item i to be sold together with each frame f. This information can be gathered by counting how many times i and f have been sold into the same order and building a matrix of frequencies containing these numbers. In the particular case that a quantity not greater than one of each item is sold in every order, this matrix coincides with the rows of M corresponding to the frames f, where M is the symmetric matrix presented in paragraph 5.2.2 and used to calculate the Virtual Profit.

The calculation of the number of times frame f has been sold, which is the total number of orders containing frame f, can be easily integrated in this computation by increasing the f-th element of a vector of counters by one every time an order is found in which frame f appears.

The definition of the matrix of frequencies and the mentioned vector are used in the computation of the probabilities of each item i to be sold with the frame f. These can be obtained by dividing each element of the f–th row of the matrix by the corresponding element of the vector of counters. The probability results to be zero when they have never been sold together and one when frame f has never been sold without the accessory i.

Getting back to the generation of the Systems orders, after an order has been assigned with a frame, an unambiguous name and a date, an imaginary dice is thrown for every existing accessory, generating a random number between zero and one; if this number is greater than the probability of the item i to be sold with the considered frame, item i is added to the order.

The quantity in which that accessory is present in the simulated order is given by the average quantity it appeared in the orders in the last two years. For instance there are items that are always sold in pairs or greater quantities, while for many others the quantity depends on the customer testing need. The choice to take the average of these numbers instead of using once again a statistical generation is based on the fact that it does not depend on the selected frame and that probabilities of an item to be ordered in a determined quantity when sold with every frame are equal.

The validation of the demand generator is provided in section 6.2; the Virtual Profit is calculated for the Systems orders generator, while for both the orders generated through it and the OTC demand emulator, the average values of the monthly demands are compared with the historical ones.

6.2 Validation of the simulated demand

In section 6.1 the statistical simulator of the demand is presented. In particular the generator of the Systems orders is shown as based on the correlation between items. This implies that if it is correctly designed, the Virtual Profits of the items calculated using the simulated data should be comparable with the historical ones. In order to validate the correctness of the generation algorithm and its Matlab implementation, the mentioned historical Virtual Profits for some of the accessories of the Grip Cell are compared in Figure 6.2 with the ones obtained by the average on 100 simulation runs.

By showing comparable values calculated on the two sets of data, the graph provides a validation of the simulation tool.

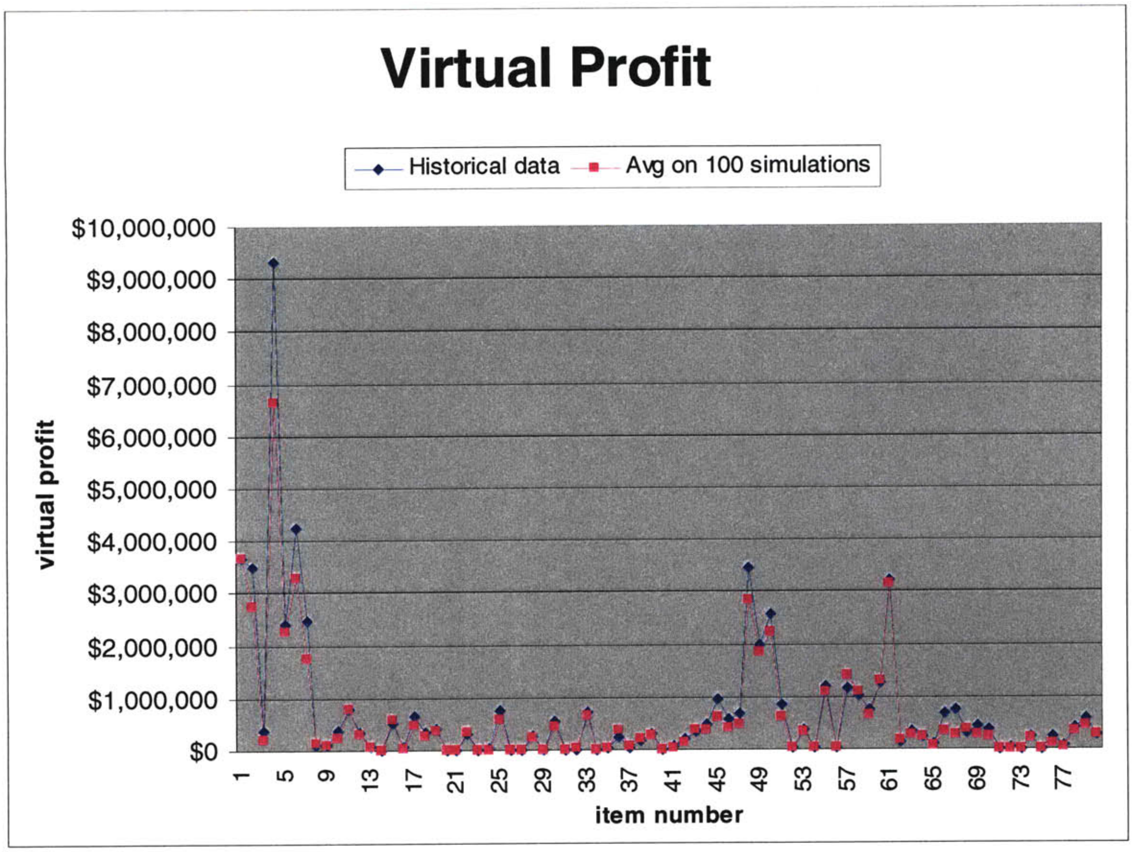


Figure 6.2 – Comparison of the Virtual Profits calculated on the historical and simulated data

Moreover the method directly derives the frames demand from their probability distributions while the accessories are assigned to orders based on the correlation they have with the frames. In other words, to determine the monthly number of accessories order, their probability distributions are not directly used.

In order to validate the demand simulator is thus useful to measure the accessories statistical properties and verify that they match with the historical values.

Figure 6.3 shows a comparison of the average monthly volumes of the demands for some of the accessories on the Grip Cell. The analysis verifies that the statistical parameters are significantly similar in the two cases.

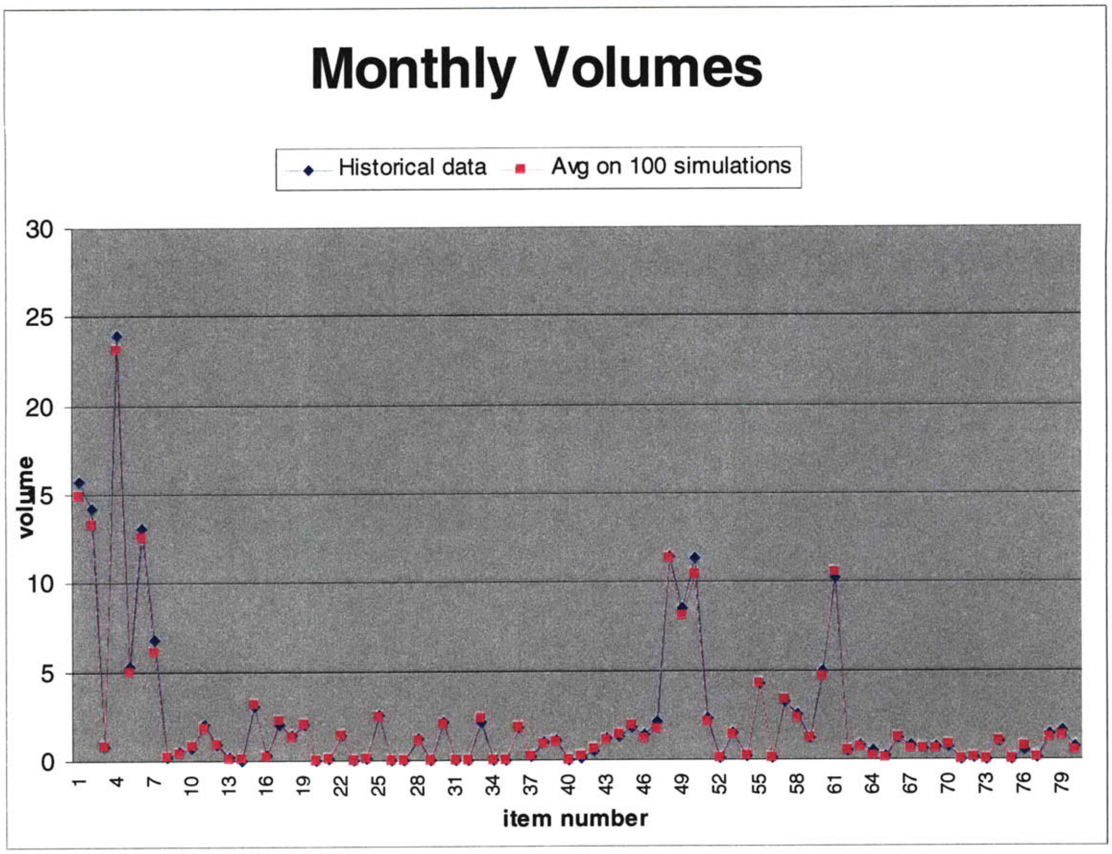


Figure 6.3 – Comparison of the average monthly demand calculated on the historical and simulated data

7. The Control Policy

7.1 The Q,r Policy

7.1.1 Definitions and Type I Service Level

The proposed approach is based on the Q,r or reorder point model, a continuous review inventory model. Hadley and Whitin (1963) [5] can be consulted as a reference.

This model requires the following basic assumptions as stated by Wang and Hu (2007) [6];

1. Demand comes from a stationary process with known mean μ and standard deviation σ.
2. Demand is independent in non-overlapping time increments
3. The demand over lead time does not exceed the reorder quantity Q
4. Lead time is constant (however extensions of the model exist where this assumption is relaxed)

A discussion of how these assumptions relate to the case studied can be found in the work of Facelli [2].

An internal order of constant size Q is placed whenever the inventory level drops to or below the reorder quantity R. The quantity Q can be determined using, for example, an EOQ (Economic Order Quantity) model which would consider the tradeoff between set-up or shipping cost and inventory holding cost. Note that in some cases the aforementioned tradeoff might not be evaluated because either the holding costs or the set-up/shipping costs are difficult to determine. Moreover constraints from the supplier about the order quantities might exist and limit the choice of Q. In any case, Q should be large enough so that the hypothesis 3 of the Q,r model is satisfied.

Given Q, R is determined in order to cover the lead-time demand with a certain probability. It is now convenient to introduce the concept of Type I service level which will be simply referred to as service level from now on:

The service level α is defined as the probability of not stocking out when there is an order event.

For a Q,r model, the probability of stocking out during the lead time is given by the probability that the demand x during the lead time L is larger than the reorder level R. Therefore, defining $f_L(u)$ as the probability density function of the demand over the lead time, the probability of stocking out can be written as

$$1-\alpha = \int_R^{\infty} f_L(x)dx \qquad (7.1)$$

If the mean of the demand is μ and its standard deviation is σ, the demand over the lead time has mean μL and standard deviation $\sigma\sqrt{L}$. The integral (7.1) can be solved under the assumption that the demand is characterized by a certain probability density function. In particular the demand is assumed to be normally distributed and this constitutes a fifth assumption to be added to the four ones for the Q,r model listed before.

Then the probability of stock out can be expressed as:

$$\int_{x=R}^{\infty} \varphi_{\mu L, \sigma\sqrt{L}}(x)dx = 1 - \int_{-\infty}^{R} \varphi_{\mu L, \sigma\sqrt{L}}(x)dx \qquad (7.2)$$

where $\varphi_{\mu,\sigma}$ represents a normal probability density function with mean μ and standard deviation σ.

Naming z the safety factor, let set:

$$R = \mu L + z\sigma\sqrt{L}$$

(7.3)

And substituting the expression for R in (7.3) into equation (7.2) we get

$$1 - \int_{-\infty}^{\mu L + z\sigma\sqrt{L}} \varphi_{\mu L, \sigma\sqrt{L}}(x)dx$$

(7.4)

Applying the definition of cumulative distribution function (abbreviated as *cdf*), and calling $\Phi_{\mu,\sigma}$ a normal cdf with mean μ and variance σ, (7.4) can be rewritten as

$$1 - \Phi_{\mu L, \sigma\sqrt{L}}(\mu L + z\sigma\sqrt{L})$$

(7.8)

Finally, using Φ to indicate a standard normal cdf, (7.8) becomes

$$1 - \Phi(z)$$

(7.9)

In conclusion, using (7.1) and (7.9) the **Type I** service level α using the Q,r policy can be expressed as

$$\alpha = \Phi(z)$$

(7.10)

7.1.2 Expected inventory level

The expected inventory level is one of the most important characteristics of any inventory systems. In particular, it is used in the proposed optimization to evaluate the holding cost of a given set of Q and R. Unfortunately, the exact expected inventory level is not easy to compute, hence several approximating procedures have been proposed in the literature. A discussion about some of these approximations and an attempt to identify the ''best'' one, has been discussed by Lau and Lau [7]. Such paper reports that the exact expression under deterministic lead times, non-negative R but otherwise fairly general conditions is:

$$\frac{1}{Q}\int_{R}^{Q+R}\left\{\int_{0}^{y}(y-x)f_L(x)dx\right\}dy$$

(7.11)

In [i.3] equation (7.11) is also modified to be applicable for both positive and negative values of R as

$$\frac{1}{Q}\int_{\max(0,R)}^{Q+R}\left\{\int_{0}^{y}(y-x)f_L(x)dx\right\}dy$$

(7.12)

Lau and Lau [7] propose their own approximating formula for (7.12) but also conclude that the most common approximation to the average inventory level given in [5] is more robust than the literature appears to imply. It is expressed as:

$$\frac{Q}{2}+R-\mu L$$

(7.13)

In this work (7.13) is used to evaluate the expected inventory level. As part of future work it might be interesting to evaluate which performances might be achieved using other approximating formulas in the framework presented in this work.

Finally, note that an equivalent expression of (7.13) can be obtained by substituting the expression of R (7.3) into (7.13):

$$\frac{Q}{2} + z\sigma\sqrt{L}$$

(7.14)

7.1.3 Type II Service Level

The concept of *Type II service level,* or *fill rate*, is defined as the percentage of demand that is immediately met from inventory.

For a Q,r model, the following approximation for the Type II service level can be made. Defining β as 1 minus the fraction of demand not met from inventory β can be expressed as:

$$\beta = 1 - \frac{\int_{x=R}^{\infty} (x-R) f_L(x) dx}{Q}$$

(7.15)

The integral in (7.15) is known as *partial loss function*.

Suppose the demand over the lead time has a normal distribution with mean μL and standard deviation $\sigma\sqrt{L}$. Setting R as (7.3) and following a reasoning similar to the one used for determining the Type I service level, the partial loss function can be expressed as:

$$\int_{x=R}^{\infty} (x-R) f_L(x) dx = \int_{\mu L + z\sigma\sqrt{L}}^{\infty} (x - \mu L - z\sigma\sqrt{L}) \varphi_{\mu L, \sigma\sqrt{L}}(x) dx$$

(7.16)

Considering the standard normal cdf and pdf (7.16) it can be written as:

$$\int_{x=R}^{\infty}(x-R)f_L(x)dx=\sigma\sqrt{L}[\varphi(z)-z(1-\Phi(z))]$$

(7.17)

Finally, using expression (7.17) in (7.15) we obtain:

$$\beta=1-\frac{\sigma\sqrt{L}[\varphi(z)-z(1-\Phi(z))]}{Q}$$

(7.18)

7.2 Finished goods inventory control problem

The Q,r policy assumes that a service level is chosen by the user and the corresponding reorder level R is computed. However in some cases, it is of interest to consider different service levels for different products taking into consideration their different importance and the holding cost to store them. In these cases, an optimization problem can be formulated, and by solving it, the service levels and, as a consequence, the reorder levels can be found.

The goal function to maximize in order to optimally determine the stock level for a given item i is the expected total profit generated by that item.

The expected total profit can be calculated as the expected revenue minus the expected total costs.

The expected revenue for each product can be found by multiplying its unit price times its expected sales $E(S_i)$, which are a function of the demand rate and the number of items in stock. The considered item and the ones that are ordered in the same orders must in fact all be in stock and available. Moreover, since some customers are willing to wait if the part is not in stock, sales are also function of the delay acceptability w_i, that is to say the percentage of customers that would still buy the item if it is not in stock and they have to wait the estimated waiting time.

As described in 6.1, Systems orders contain many items that cannot be shipped separately usually because they cannot be used without each other. Therefore another factor that influences the expected sales is the availability in stock and delay acceptability of the items that are ordered together with item *i*. The holding cost and the production cost represent all the costs considered for this problem. The expected holding cost is given by a unit holding cost h_i multiplied by the expected inventory $E(I_i)$ as defined in 7.1.2.

The production cost depends on the number of items produced, which can be greater than the number of items sold. However assuming that items are not perishable while they sit in stock and do not become obsolete (which are reasonable assumptions for the Instron products), when the number of produced items is greater than the number of sold items, they can be stored and sold in the next period (year or quarter). Therefore for those items there will not be a production cost in the following period.

In order to simplify the reasoning, the production cost is ascribed to the period in which the item is sold. This does not change the total production cost.

The unit profit from a sale p_i can be expressed as the difference of the unit price and the unit production cost. Since the standard values provided by Instron change year by year and may contain errors which are adjusted, introducing discounts or increment of price when the order is registered, the unit price and cost are determined considering their averages in the analyzed period and including discounts and other adjustments. This provides robustness to the method to errors and variations of price.

The expected total profit model for every item *i* can be written as

$$E(S_i)p_i - h_i E(I_i) . \tag{7.19}$$

Therefore the objective function becomes:

$$\max \sum_{i=1}^{I} E(S_i)p_i - h_i E(I_i)$$

(7.20)

The optimization problem has been developed by Facelli [2] who considered both a global and a decomposed optimization. The latter is shown to give results similar to the first one and it is based on the usage of the Virtual Profit. The decomposed optimization is shown in Facelli [2] to have an analytical solution. The resulting policy and its implementation are described in the following paragraphs.

7.3 The Proposed Policy

As previously discussed two types of order exists: OTC and Systems. The proposed approach to face this issue, in the *divide et impera* spirit, considers the OTC and Systems orders separately. Considering the generic *k-th* product, the sequence of operations is as follow:

1. The reorder quantity Q_k is determined considering the total demand (OTC plus Systems) as described in 7.4.
2. Considering the total profit P_k made by the item in OTC orders and its demand mean μ_k^{ϑ} and standard deviation σ_k^{ϑ} in OTC orders a reorder level R_k^{ϑ} is determined as defined in 7.5.1
3. Considering the virtual profit V_k made by the item in Systems orders (in place of P_k) and its demand mean μ_k^{ς} and standard deviation σ_k^{ς} in Systems orders a reorder level R_k^{ς} is determined as defined in 7.5.2.

4. The "partial" reorder levels R_k^{ς} and R_k^{ς} are summed up and rounded to an integer value following a heuristic rule as the one described in 7.5.3. This way a final value R is determined.

7.4 Reorder Quantity Q

The reorder quantity Q is often determined considering the trade off between holding cost and the set up (for manufactured parts) or ordering (for purchased parts) cost. A basic approach in this sense is represented by the EOQ model but many other models are discussed in literature and used in practice. In this case however the set up costs were either not available or not significant and the shipping costs hard to determine. When information about set up cost was available a lot size Q' was already determined and stored in IBS and such quantity is taken into consideration. In any case, Q is chosen large enough to satisfy with a high probability the demand over the lead time. So, Q_k is determined as

$$Q_k = \max\left\lfloor ceil\left(\mu_k L_k + 3\sigma_k \sqrt{L_k}\right), Q'_k \right\rfloor \tag{7.21}$$

7.5 Reorder Level R

7.5.1 Implemented policy for OTC

Reintroducing from now on the superscripts ϑ and ς to differentiate the variables relative to OTC and Systems orders, the implemented policy for OTC is

- If $\sigma_k^{\vartheta} < 0 \ \vee \ L_k \leq 0 \ \vee \ Q_k \leq 0 \ \vee \ P_k < 0 \ \vee \ w_k \notin (0,1)$
  - Give a warning
- Else if $P_k = 0$

- Make To Order Product

- Else if $\sigma_k^{\vartheta} = 0$
  - Make To Stock Product

    $R_k = \mu_k L_k$

- Else
  - If $\mathrm{Q_k c_k} \varepsilon \geq (1 - \mathrm{w_k})\mathrm{P_k}$

    Make To Order Product

  - If $\mathrm{Q_k c_k} \varepsilon < (1 - \mathrm{w_k})\mathrm{P_k}$

    Make To Stock Product

$$\mathrm{z_k}^{\vartheta} = \Phi^{-1}\left[1 - \frac{\mathrm{Q_k c_k} \varepsilon}{(1 - \mathrm{w_k})\mathrm{P_k}}\right],\ R_k^{\vartheta} = \mu_k^{\vartheta} L_k + z_k^{\vartheta} \sigma_k^{\vartheta} \sqrt{L_k}$$

(7.22)

7.5.2 Implemented policy for Systems

The resulting policy considering Systems orders is, similarly to 7.5.1

- If $\sigma_k^{\varsigma} < 0 \ \vee \ \mathrm{L_k} \leq 0 \ \vee \ \mathrm{Q_k} \leq 0 \ \vee \ \mathrm{V_k} < 0 \ \vee \ \mathrm{w_k} \notin (0,1)$
  - Give a warning

- Else if $\mathrm{V_k} = 0$

- Make To Order Product

- Else if $\sigma_k^{\varsigma} = 0$
  - Make To Stock Product

    $$R_k^{\varsigma} = \mu_k^{\varsigma} L_k$$

- Else
  - If $Q_k c_k \varepsilon \geq (1 - w_k) P_k$

    Make To Order Product

  - If $Q_k c_k \varepsilon < (1 - w_k) P_k$

    Make To Stock Product

$$z_k^{\varsigma} = \Phi^{-1}\left[1 - \frac{Q_k c_k \varepsilon}{(1 - w_k) P_k}\right],\ R_k^{\varsigma} = \mu_k^{\varsigma} L_k + z_k^{\varsigma} \sigma_k^{\varsigma} \sqrt{L_k}$$

(7.23)

7.5.3 Sum rule

Once R_k^{ϑ} and R_k^{ς} are determined these two quantities must be considered together to obtain a total R:

$$R_k = R_k^{\vartheta} + R_k^{\varsigma}$$

(7.24)

This is a conservative choice as it assumes that the demand for the *k-th* product in OTC orders is independent from its demand in Systems orders. Because equation (7.24) cannot be used if the resulting policy for OTC or Systems orders is Make To Order as either R_k^{ϑ} or R_k^{ς} is undefined and the reorder level to be used in practice must be an integer number, the following heuristic rule is used in this work.

- If ((Systems → MTO) AND (OTC → MTO))

  MTO product

- Else
  - If (Systems → MTO)

    MTS product with $R'_k = R_k^{\vartheta}$

  - If (Otc → MTO)

    MTS product with $R'_k = R_k^{\varsigma}$

  - Else

    MTS product with $R'_k = R_k^{\vartheta} + R_k^{\varsigma}$

  - $R_k = round(R'_k)$

7.6 Simulation Results

7.6.1 Validation of the Policy

In order to validate the proposed strategy, the results of 50 simulations for a policy corresponding to 1.2 theoretical MOH are compared with the current policy in Table 7.1.

	Current Policy	Proposed Policy
Lost Orders	20.94	1.5
MOH	4.52	1.18

Table 7.1 – Comparison between the current policy and the proposed one

The results of the 50 simulations are consistent with the results simulated using the historical demand and presented by Chugh [3]. Moreover the simulated average MOH is significantly similar to the theoretical one of 1.2 MOH.

7.6.2 Robustness to Variations

A test of robustness of the policy to drastic shifts in the demand is presented, so that the policy can be better understood and its parameters can be properly selected to take into account variations.

The policy corresponding to 1.2 MOH is tested for shifts in the total sales volumes of +10%, +20%, +30%, -10%, -20% and -30%.

Figure 7.1 summarizes the resulting average MOH for the 50 simulations performed for each shift in the demand.

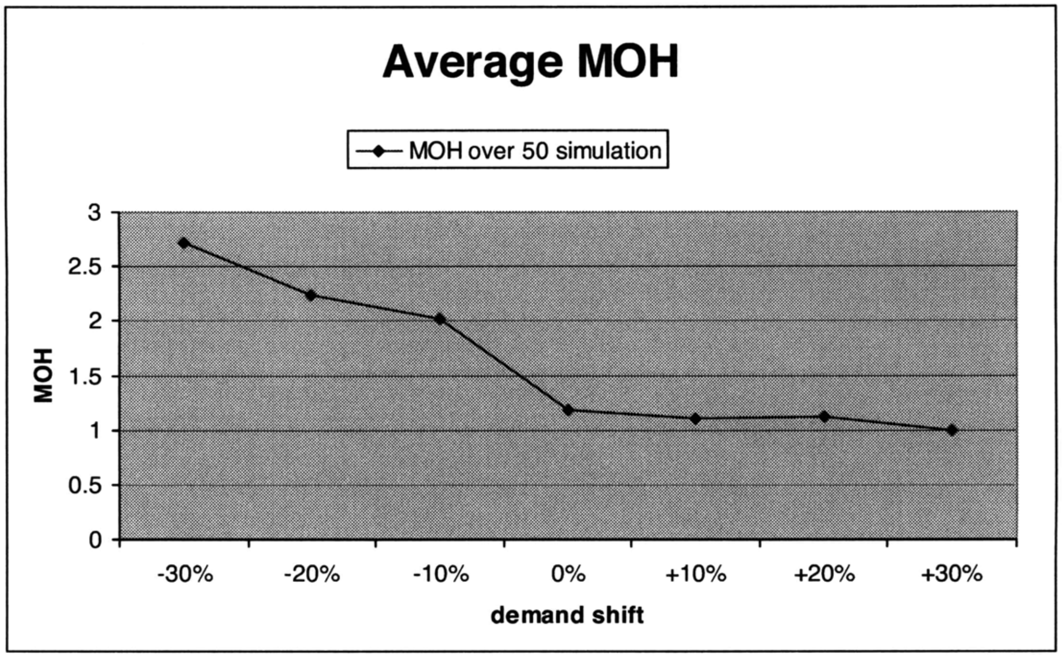


Figure 7.1 – Simulated average MOH vs. demand shift.

The policy shows a steep increase of the MOH as the demand decreases. Because of the drop in the demand, in fact, the replenishment of the inventory undergoes a slowdown, the inventory is more often filled at its maximum level, and the average months on hand increase significantly. On the other hand, when the sales volumes increases the months on hand only decrease by a small value. Because of the increase in the demand, the inventory gets replenished more often but more often it gets emptied. At the same time the number of lost sales increases because of the increased probability of stockout.

As can in fact be seen in Figure 7.2, which shows the percentage of sales that are lost when the demand shifts compared to the case in which there is no shift.

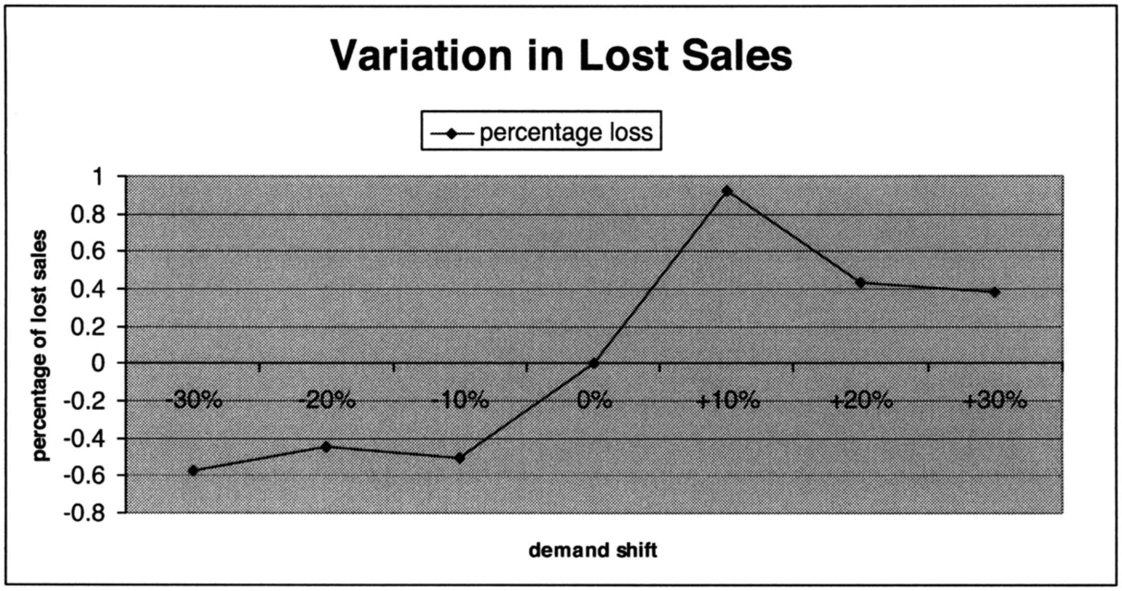


Figure 7.2 – Simulated lost sales (percentage of base case) vs. demand shift.

The graph shows that as soon as the demand has an increase, suddenly the number of lost sales becomes about the double, because the probability of stockout has increased. On the other hand, as the demand decreases the lost sales decrease by a less considerable quantity. Note that there will always be a percentage of lost orders, because of how the customer impatience curves are drawn, as described by Palano.

The results from Figure 7.1 and 7.2 suggest the importance of updating the levels as soon as a demand shift is detected: when an increase in the volumes occurs, the lost sales might rapidly increase, while when the opposite happens, the MOH quickly overcome the limit of 2.

7.6.3 Comparison between results using Virtual Profit and using profit

An important element that makes the decomposition of the optimization problem possible and allows using the policy proposed in 7.5 is the Virtual Profit. By using this quantity it is in fact possible to evaluate in a simple way the advantage of using a certain safety factor z when systems orders are considered. In order to prove the importance of the Virtual Profit, suppose now to use the policy described in 7.5.1 for both OTC and Systems orders. The results are shown in Figure 7.3.

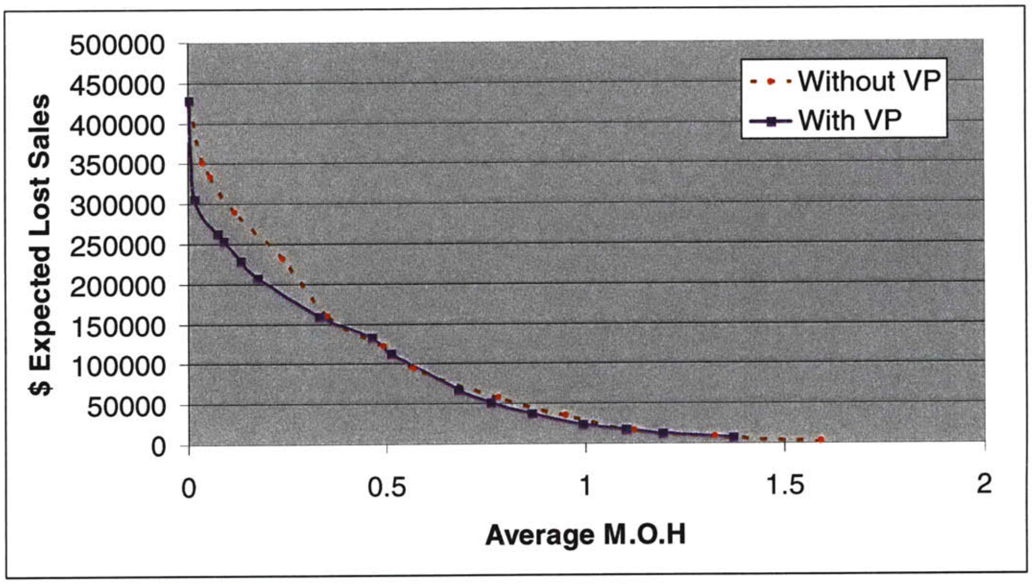


Figure 7.3 – Expected lost sales vs. MOH for the two policies.

Figure 7.3 shows that the considering the Virtual Profit in place of the total individual profit improves the quality of the solution. This means that with the same size of inventory, by using the Virtual Profit in the policy the value of expected lost sales can be reduced. This result is especially evident when the size of the inventory is small (less than 0.4 MOH) but is consistent for every amount of inventory considered. Moreover, note that in Figure 7.1 the effect of using the Virtual Profit is not fully visible because:

- Both OTC and Systems orders are considered at the same time. The use of Virtual Profit has effect on R_k^{ς} but not on R_k^{ϑ} which is the same in both cases.

The sum $R'_k = R_k^{\vartheta} + R_k^{\varsigma}$ is rounded to the closer integer. This means that a different value of R_k^{ς} result in a different control policy only if R'_k is rounded to a different integer.

7.7 Implementing a Tool

The purpose of the software provided is enabling the Configuration Department planners to recalculate the optimal parameters for the inventory control as frequently as needed, in an easy way and without waste of time. For these reasons the tool has to satisfy the specifications described in paragraph 4.6.

The inventory planner is provided with a tool consisting in an executable file compiled from a series of Matlab functions. The code for the implementation of the tool can be found in the Appendix. As showed in the use case diagram of Figure 7.4, the roles of the operator in the computation of the parameters can be split in two categories:

- The gathering of the information, including the extraction of the information from IBS, their integration and the reporting of exceptional events, such as the insertion of new items in the product list.
- The operation of the provided software and the interactive selection of the solution to implement.

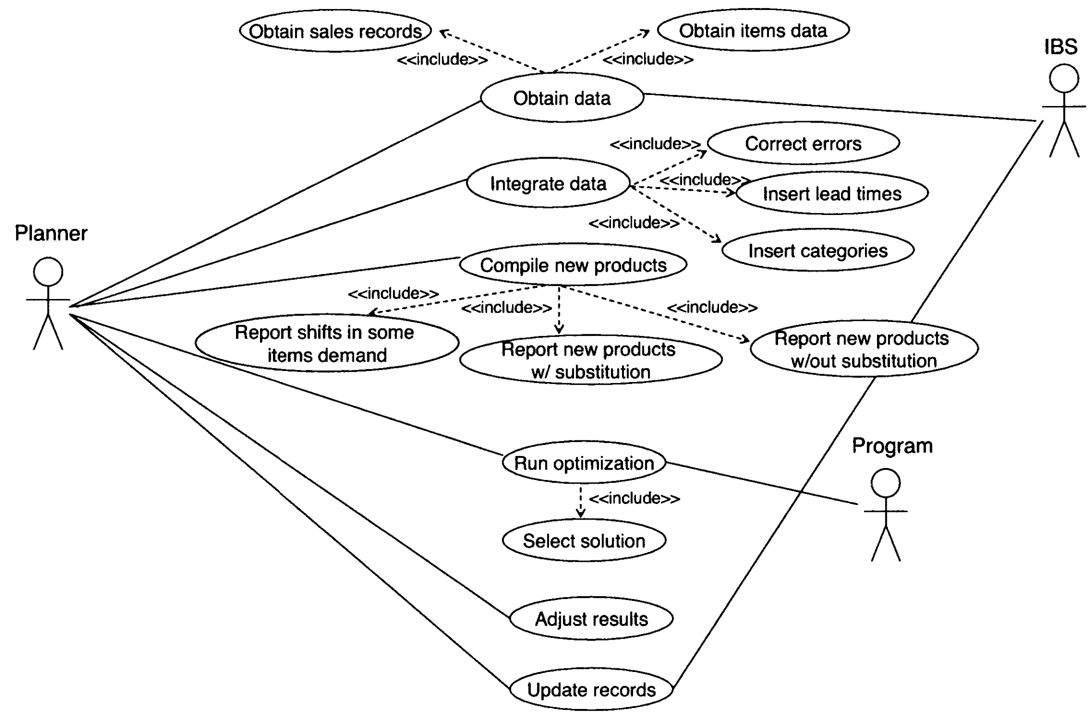


Figure 7.4 – Use Case diagram of the proposed tool for the control parameters optimization

The information required for the algorithm to operate concerns historical sales records, item records, and new products records. The firsts are used in order to calculate the Virtual Profit (for Systems orders) or total profit (for OTC orders), which are based on the past sales. They are also used to compute the statistical parameters of the demand, from which the reorder level (R) and the reorder quantity (Q) are calculated for the proposed policy. In particular a list containing the following IBS records is needed: order number, type of product, level of the order, order class, item number, quantity shipped, standard cost, net price, order entry date. A template has been created on IBS in order to simplify the data extraction and guarantee that no important information is missing. This operation does not need to be repeated more often than once per quarter.

The second type of information needed regards the items of interest: before the computation can start, the items to optimize must be selected, the information about their lead times, lot sizes and category extracted from IBS or manually inserted. When there are errors in the data downloaded from IBS, this is the moment to detect and correct them (a more detailed explanation about the data integration can be found in Chapter 9). Different groups of items to be optimized can be registered in different Excel spreadsheets. This operation can be performed only once, with the condition that the data are updated when major changes in the lead times or in the lot sizes occur.

The last type of information needed for the optimization regards the new products. Because at the beginning of one product's life no historical data are available, the program takes advantage of the sales forecasts, which are usually provided by the marketing team when new products are introduced. In particular, when an item substitutes an old one, the historical records of the replaced items can be exploited to extract information about the variability of the demand. When an item is introduced but no substitution takes place, an average value of variability is introduced by default so that the replenishment parameters can be computed.

At this point the operator is ready to run the optimization program. After providing information about the quantity of historical data to analyze he/she is asked to insert a forecast of increase or decrease in the sales, if available. As shown in section 7.6.2 in fact, as soon as a large demand shift is detected, the control parameters should be immediately recalculated. However it might be that the information for the last quarter is not yet entirely available, and so it is convenient to use the available historical data shifted by a quantity.

A picture is then printed to the screen, showing a plot of the lost sales versus the months on hand. The operator has the choice of the working point, keeping in mind that the months on hand can fluctuate following the variability of the demand.

The control parameters for the selected items are calculated and printed on an Excel file. The operator has the choice to modify them, depending on further constraints on the lot sizes or on the minimum levels. The Excel provided as an output by the software can be used to perform a sensitivity analysis: by changing the quantities, the displayed MOH and VOH change their values and the operator is aided in his/her choice. At this point the quantities can be printed on the Kanban cards and inserted on IBS as future lot sizes. Note that the number of items to be placed in the Kanban bag, or *Minimum Quantity*, is equal to $R+1$, because when the bag is opened the level reached is R.

8. Results and discussion

8.1 Raw materials inventory

As introduced in Chapter 4, purpose of the project was also to provide a raw materials inventory control policy supporting the finished goods inventory. The current policy is value-based: the parts are classified by financial value to the company (classes A, B, C and D) and the reorder quantities and levels only depend on the class. A Q,r policy with fixed service levels is proposed; the results are here summarized and discussed.

8.1.1 Results

In order to implement the Q,r policy for the raw materials inventory, some information is necessary. In particular, knowing the replenishment lead times negotiated with the supplier is fundamental. In this paragraph, the results of the Q,r policy are presented by comparison with the current value-based control policy. Firstly, the importance of the lead time is shown through a parametric comparison; then, the two policies are evaluated with the best current estimate of the lead times.

Figure 8.1 shows the difference that could be made by having more accurate information about the lead times. The graph on the top shows the expected inventory value on hand, while the graph below shows the average service levels. For the sole purpose of showing the differences as the lead times vary, the graphs are based on the assumption that the lead time is the same, and constant, for all the parts. The blue lines represent the current value-based policy, which does not consider lead times or the demand variability. The red lines correspond to the Q,r policy, implemented using also the lead times and variability information. Two examples are highlighted with vertical lines: a lead time of 4 days and a lead time of 18 days.

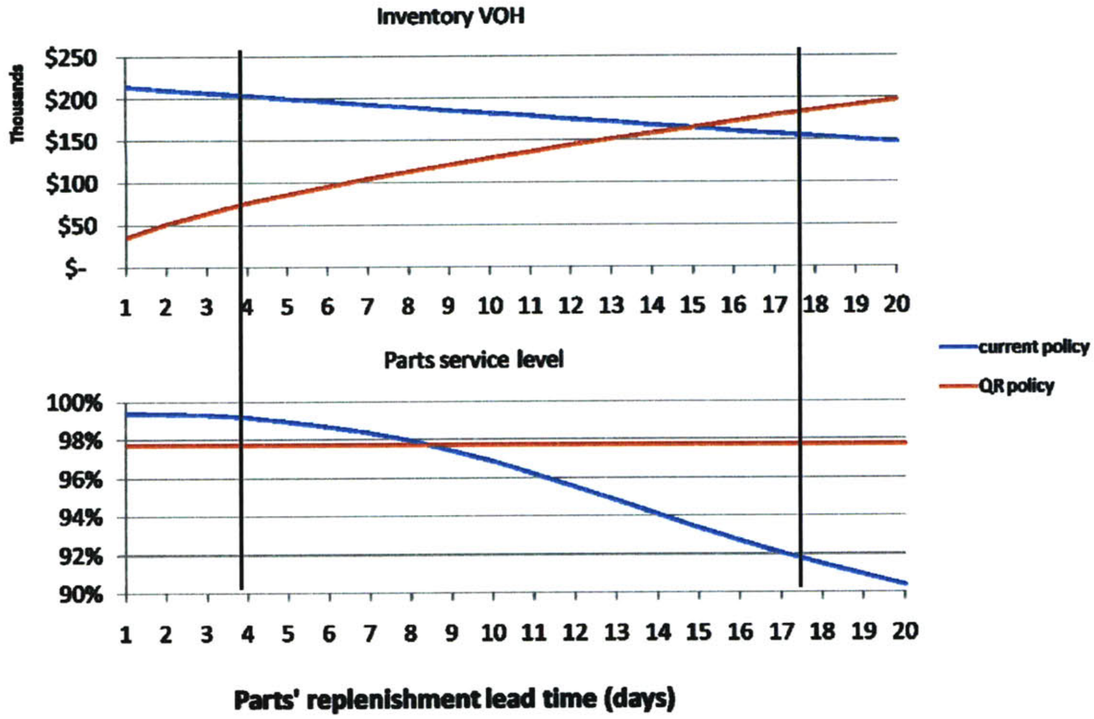


Figure 8.1 - Service level and Inventory VOH vs. lead time

If the suppliers ship more quickly than expected, and the lead time is shorter, both policies have high service levels because the shelves are replenished quickly. Being designed upon the shorter lead times, however, the Q,r policy manages to accomplish high service levels with low inventories. In the first example, indeed, the value on hand is reduced by three times.

On the other hand, if the lead times are longer, the only way to achieve high service levels is to have higher inventories. Thus, the proposed policy suggests inventory levels that are comparable or even higher than the current ones. The Q,r policy, on one hand, uses the information about lead times in order to maintain high service levels; the current policy, on the contrary, does not consider them, causing a significant percentage of lost orders (orders meaning grips to be assembled), as shown in the second example.

An estimate of the actual supplier replenishment lead times is obtained by talking with the purchasing department and described by Palano [26]. In this case, the lead times are different for each part. Table 8.1 shows a comparison of the results obtainable with the two policies based on this estimate. Moreover it provides an estimate of the savings that would be achieved by agreeing on shorter lead times with the suppliers.

Method	**Average inventory VOH**	**Parts service level**
ABCD - Division by value	$179,731	93.2%
Q,r – Knowing and using the lead times	$126,299 (-30%)	97.7%

Table 8.1 - Comparison of raw materials inventory control policies

As table 8.1 shows, only as a result of improving the accuracy of lead times, the Q,r policy would allow achieving high service levels at the same time cutting the costs by 30%. If, in addition, the purchasers obtain agreements for shorter lead times for the most valuable parts, the costs would further decrease.

8.1.2 Discussion

Based on the analysis proposed by Palano [26] and on the results here described, the current inventory policy, which is value-based and does not consider lead time and demand variability, can result in irregular inventory distribution, lower service levels and higher inventory value on hand. A simple Q,r policy is proposed, which gives better and more regular results.

In designing and optimizing the finished goods inventory control, the assumption that all the raw materials are always available is made. The designed Q,r policy achieves service levels of about 98% for each part. Thus, the above mentioned assumption can be still considered valid.

However, in order to implement the Q,r policy, the replenishment lead times are necessary. As a general consideration, the lead times are necessary to make sure that the service levels are high without wasting inventory. Thus, the lead times of every part should be tracked in the way described in section 9.1, and accurate information should be kept on the company databases. In addition, if the suppliers are flexible on the lead times, the Excel spreadsheets can be used in the decision process to determine the correct tradeoff between lead times and inventory value on hand.

8.2 Finished goods inventory

The policy proposed shows potential for a significant improvement in inventory control. Figure 8.2 shows a comparison between the proposed policy, a simple Q,r policy and the values of Q and R currently in use. Note that the term "simple Q,r" refers to a Q,r policy with an equal safety factor z for all the products. The figure shows the expected lost sales, due to products unavailability, versus the total expected inventory held. The amount of inventory held is measured in months on hand (MOH):

$$E[I]_{MOH} = \frac{\text{expected inventory value on hand}}{\text{average monthly demand}} = \frac{\sum_i c_i E[I_i]}{\sum_i c_i \mu_i} \qquad (8\text{-}1)$$

Where c_i is the unit cost of part i, $E[I_i]$ is its expected inventory level and μ_i is its average monthly demand.

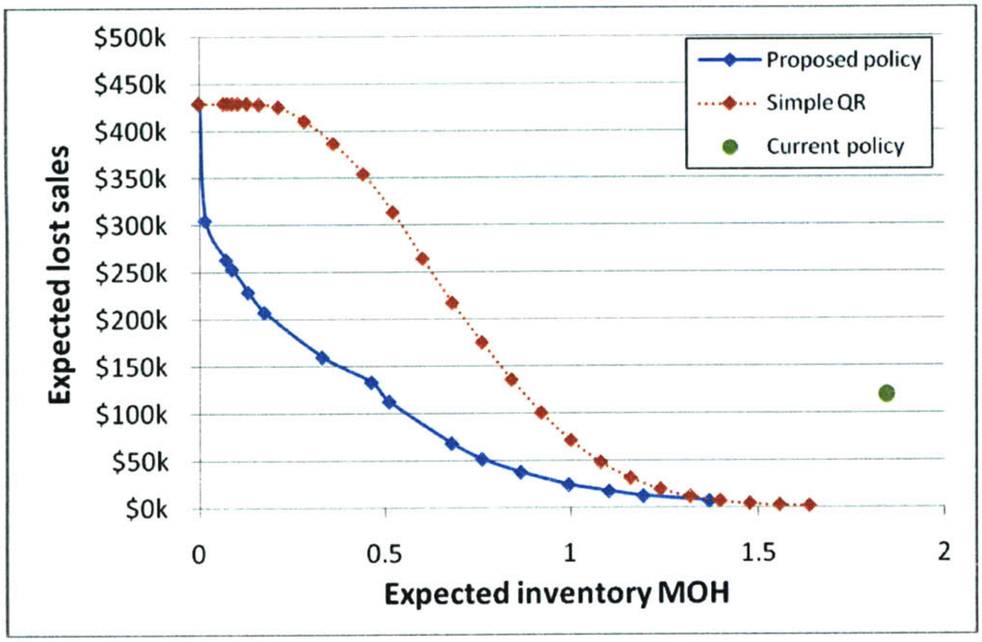


Figure 8.2 – Expected lost sales vs. Inventory MOH

As figure 8.2 shows, the proposed policy outperforms both the simple Q,r and the current policy. In particular, at the same level of expected loss sales given by the current policy, the proposed policy allows reducing the inventory from about 1.8 MOH to 0.5 MOH. From another point of view, with the amount of inventory currently held, the proposed policy would allow reducing the expected lost sales from about $120,000 per year to nearly zero.

In addition, Figure 8.2 shows that the proposed policy outperforms the simple Q,r policy. As one might expect, the difference increases as the size of the inventory gets smaller, while it decreases as larger inventory is considered. As a limit case, the value of lost sales achieved by the simple Q,r with 0.15 MOH is the same that would be obtained by a complete make to order (MTO) policy. With the proposed policy, instead, 0.15 MOH of inventory can halve the expected loss as compared to an MTO policy.

Figure 8.3 shows the expected lost sales value versus the value of the inventory on hand. As one can see from the graph, if a solution with 1.2 MOH is chosen (the penultimate point on the purple line) the inventory could be reduced from $240,000 to $157,000.

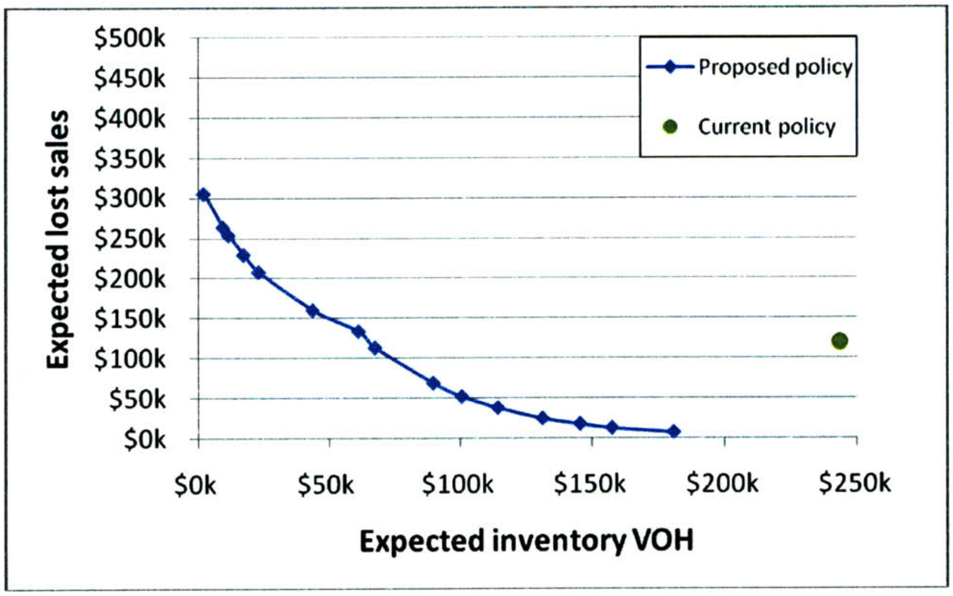


Figure 8.3 - Expected lost sales vs. Inventory VOH

Considering the trade-off between size of inventory and expected loss sales, a good compromise is a solution with an expected inventory of 1.2 MOH. This allows both reducing the amount of inventory and the expected loss sales. Moreover, a preliminary analysis of the maximum inventory levels shows that, with this solution, it is unlikely that the inventory levels measured at the end of one month will go above 2 MOH (considering the monthly demand variability). Table 8.2 shows a comparison between the proposed solution (with 1.2 MOH) and the current policy.

	Current Policy	Proposed Policy
Average value of Lost orders	$119,391	$12,453
Expected Inventory (MOH)	1.85	1.19
Expected Inventory (VOH)	$243,481	$157,411

Table 8.2 - Current policy vs. 1.2 MOH solution

8.3 Simulation

The aim of simulation is to validate the results of the optimization module and to test the robustness of the proposed policy. The simulation also helps to determine the advantage of considering correlation between the demands of items sold in systems as compared to neglecting them in the analysis as explained by Palano [26]. The simulation estimates the following performance measures: number of orders lost, their value, months on hand of inventory for every month simulated and dollar value of inventory for each simulated day.

8.3.1 Validation

The optimization model provides the right mix of products that should be available on the floor. To validate these results, the levels were simulated 50 times over two years, 2007 and 2008, and then compared with the projected results from the optimization.

Figure 8.5 shows the losses made for different optimized inventory levels as predicted from the optimization and the simulation, versus the inventory months on hand.

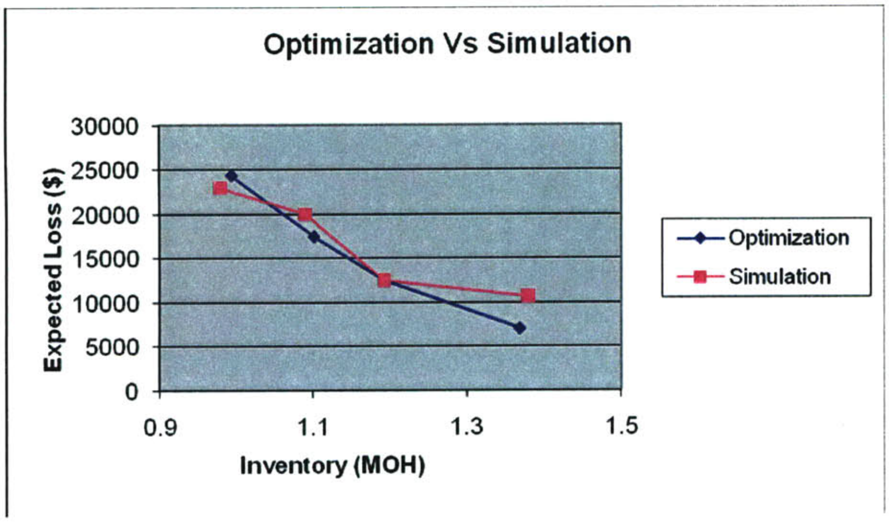


Figure 8.5 - Comparison between theoretical and simulated loss for different solutions of the proposed policy

As it can be seen in figure 8.5, the optimization and the simulation graphs show a similar behavior, supporting the correctness of the optimization model.

This curve led to the selection of a solution providing an average inventory level of 1.2 months on hand, as described in section 8.2.

8.3.2 Robustness analysis

By running the proposed inventory levels over statistical demand, the robustness of the proposed policy can be tested, as described in Chapter 7. The statistical demand is generated using the distribution of demand of each system and item over the previous two years. In the following example, the simulation is run 50 times for seven different values of shift in demand. The shift in demand, however, is not taken into account in calculating the proposed inventory levels. Figure 8.6 depicts the average inventory months on hand versus the shift in demand.

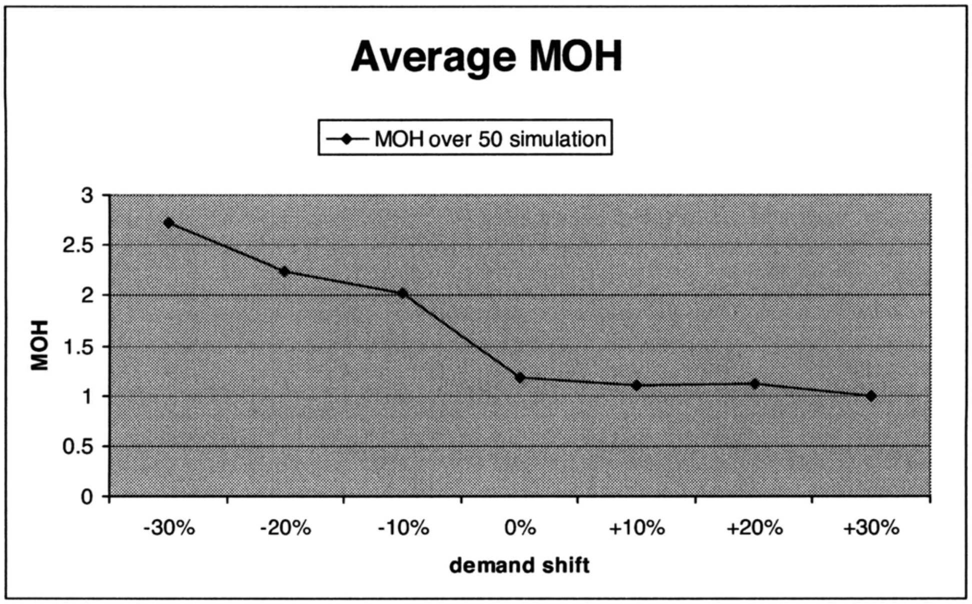


Figure 8.6 - Simulated average MOH vs. demand shift

As the demand decreases the proposed policy shows a steep increase in the MOH (above the limit of 2), while, when there is an increase in the volumes, the months on hand remain substantially stable but there is a considerable increase in the lost sales. This suggests the need for the inventory planner at Instron to update the control parameters as soon as a shift in the demand is detected, using the provided tools.

9. Recommendations

9.1 Introduction

As showed in section 8.2, the optimized control parameters result in a decrease of 35% in the inventory MOH. Moreover, it is estimated that extending the optimization to all the accessories in the Configuration Department would reduce the MOH by a similar percentage. Finally, as mentioned in section 8.1, the raw materials inventory policy provided would cut the parts inventory value on hand by 30% (or even 46% if shorter lead times are agreed with suppliers).

This represents a substantial motivation to extensively use the software provided, which allows computing the replenishment parameters for all the Instron accessories both at the finished goods and part levels, and integrate it into the Manufacturing Department procedures.

The following recommendations are made to the Instron workers in order to properly implement the proposed policy and allow improvements in the future:

- Compute the inventory levels for the raw parts using the proposed tool as frequently as possible
- Compute the inventory levels for the finished goods using the proposed tool as frequently as possible
- Keep the data on IBS updated as the accuracy of the solutions depend on the quality of available data
- Keep track of the lead times for both raw parts and finished goods
- Use the provided tool to evaluate the benefits of negotiating better lead times from the suppliers

9.2 Discussion

9.2.1 Updating inventory levels

In order to guarantee that the optimal mix of accessories is on the shelves, the inventory planners of the Configuration Department should periodically update the proposed inventory control framework using the most recent sales records available. The computation of the control parameters can be performed with the provided software.

The rapid changes that can occur in the demand, in fact, dictate the need to update the replenishment quantities as frequently as possible. On the other hand, changing the parameters implies a cost in terms of time: the time required to gather the data, run the executable file and insert the new values in IBS. This might imply negotiating new quantities with the suppliers, when agreements exist. Since it is common practice at Instron to update the IBS records at the beginning of every quarter, there is the opportunity to combine these operations and perform the computation every quarter, in time for the data of last quarter to be fully available.

A further decision to be taken by the software operator concerns the quantity of sales data to include in the analysis, for the statistical characterization of the demand and the computation of the Virtual Profit. One year is the minimum time interval that should be considered to properly estimate the variations. As the considered time period increases, the computation time increases as well. Moreover, since there is continuous variation in the product list and in the market, including older data in the analysis implies greater differences between the historical data and the current situation.

In order to minimize the run time and achieve accurate results, the sales records of the last four quarters should be used. As an example, if the analysis is performed in July, the planner should collect the data for the third and fourth quarters of the previous year and for the first and second quarter of the current year.

9.2.2 Shift in demand

As mentioned in Chapter 1, historical sales are used to estimate the future demand. While it is reasonable to assume that the relationships among products (the correlation) and the variations in the demands resemble the ones of the previous year, shifts in the average volumes can occur from one year to another. When a forecast of the shift is available, it should be entered in the command shell of the software, which is able to take this factor into consideration and to provide control parameters that fit the actual situation.

9.2.3 Dividing the analysis

In order for the information involved to be easily managed, the control parameters should not be optimized for all the items at the same time. In fact, because IBS does not currently provide all the quantities needed for the analysis, a manual integration of data is required. For example, the operator has to manually enter lead times for the items considered when not available and check for the accuracy of other parameters, such as unit costs and lot sizes, when unexpected results are detected. Moreover, the optimization of the part level replenishment quantities involves downloading the bill of materials for all the considered products and the complexity of this operation increases with the number of products. Therefore the items should be divided into groups sized so that the operator is comfortable with their management.

The division of the analysis in groups of items allows focusing on the accuracy of the inputted data which is critical for the correct performance of provided software. As an example, the inaccuracy of the lead times data provided by IBS can lead to store inadequate quantity of items.

Similarly, even if the simulation would be a closer representation of the factory floor since more items will be simulated, the run time would become large and the results difficult to interpret.

9.2.4 Lead times accuracy and negotiation

As demonstrated by Palano [26], the correct estimation of the replenishment lead times could lead to a saving of 30% in terms of VOH.

This suggests the need to improve the recording criterion for this type of data, which is currently based on many criteria. For some items that are on Kanban and for the parts that come from Binghamton (another Instron facility) the values of the lead times are known. However, for the majority of the items the lead time corresponds to the maximum lead time that can be tolerated from the supplier. As the cost and the yearly volume of one item increase, the less quantity can be stored for that item and the less time the company can wait for the supply to arrive. Also regarding the finished goods levels, lead times are missing on IBS for the parts assembled or reworked in the Norwood facility. For these parts, in fact, while setup time and run time are usually available, the time that elapses between the arrival of the order and the moment the product is ready is not recorded. The latter, however, is necessary for the computation of the optimal inventory levels.

Sufficiently accurate values can be obtained by using a new recording procedure and integrating it into IBS. Whenever an order is placed to the supplier, the purchasing agent should register the date and the supplier code, assigning a unique code to this record. The same identification number should be used in the receiving area to register the arrival date as soon as the order gets to the Norwood facility. In this way, by comparing the records with the same identification numbers, it is possible to track the lead times for all the items and suppliers so that they can be used in the computation of the inventory control parameters. When variability is present, the statistical distributions of the lead times can be evaluated. The availability of this type of data would potentially allow an extension of the optimization tools which consider stochastic lead times, as described in section 9.1.

As also showed in the raw materials control, a more drastic drop in the VOH can by achieved by negotiating shorter lead times with the suppliers. Whenever negotiation is possible, the supply chain planners should use the provided tool to evaluate the possible benefits of changing the lead times. In particular, they can compare the decrease in inventory value on hand with the eventual increase in purchasing cost.

9.2.5 Product categories

The category of a finished good (face, grip, fixture, etc.) is not stored by the IT system. However, as showed by Palano [26], the customer expectations differ for items belonging to different categories, and this record becomes important for the optimization tool. Right now such information can be found in the product catalog and in many other sources. However, keeping an updatable database or excel file with all the products divided by categories would help to easily identify this information and decrease the time necessary to gather the data needed for the finished goods optimization program.

9.2.6 Warning messages

For what concerns the information accuracy, the operator should take advantage of the warning messages displayed by the programs provided when unexpected results are detected. The user is provided with detailed instructions to follow when such events occur, and with the operating procedure for the calculation of the inventory control parameters. The detailed instructions are provided to the user with the software, and are not shown in this work.

9.2.7 New products and substitutions

Whenever new products are released and their replenishment quantities have to be calculated, the operator should provide a table containing information about the new items. Two cases can be considered:

- If the new products directly substitute one or more items in the product list, those item should be indicated as well as the fraction of demand of the old product that would converge into the new one. This allows the program to estimate the Virtual Profit and the statistical parameters of the new products demands based on the old sales data.

- If the new products are added to the product list and no old item is substituted, no historical sales data can be used to estimate the Virtual Profit, and the control parameters should be evaluated based on the simple Q,r model, without considering the correlation among the new items and the rest of the product list. In this case the operator is asked to provide a forecast of the future sales. This data is used to estimate mean value and standard deviation of the demand, and the z-factors are set by default to a high value which is not necessarily the optimal one, which cannot be estimated without knowing the Virtual Profit, but matches the need for the company to provide a high service level when the new items are introduced to the market.

9.2.8 Selecting the best solution

The final step of the computation of the control parameters involves the selection of the desired solution. Different solutions are provided, each one involving a different value of average MOH, and the operator is asked to choose one of them. A graph, similar to the one showed in figure 9.1, is displayed in order to aid the selection. For all the different solutions, the loss of sales profits and the MOH are plotted in the same graph and, as described in Facelli [27], the higher is the MOH, the smaller loss is achieved.

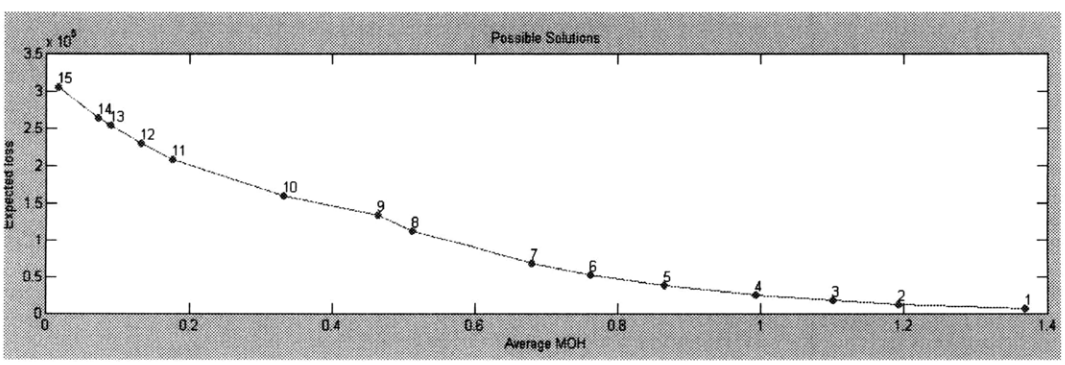


Figure 9.1 – Expected loss from sales vs. average inventory VOH for different solutions of the optimization.

When making this decision, the operator should consider that the displayed MOH is an average value and may fluctuate depending of the variability of the sales volumes. Because the Instron demand is subject to consistent fluctuations, the operator should not choose a value close to 2 MOH, which is the maximum value allowed at Instron. At the same time a small loss from sales should be achieved. This curve usually shows a flat tail, where for a little increase in the inventory cost only a little portion of sales is redeemed. The starting point of the flat tail can be considered a satisfactory solution.

9.2.9 Using and adjusting the recommended quantities

The output of this computation is a list of recommended minimum quantities and reorder quantities, which are the parameters used to build the Kanban cards. While the reorder quantity coincides with Q in the Q,r model, the minimum quantity is R+1. The reason for this is that the minimum quantity indicates the number of items contained in a bag; when the bag is opened to take one part the level R is reached and the order is placed.

At this point, the operator has the chance to modify the proposed quantities if constraints are present. For instance, constraints on the lot sizes exist. In addition, some items have to be ordered or assembled in lots that are multiples of some predetermined quantity. After the quantities are updated according to these constraints, a sensitivity analysis for the finished goods should be performed in order to evaluate the increase in the costs. The quantities can be directly modified in the Excel spreadsheet provided as output of the optimization tool, and the updated values of the theoretical MOH and VOH are showed. These quantities can be compared with the proposed ones and the choice must be taken accordingly.

A simulation can also be performed to observe the changes introduced by the adjusted quantities on the lost sales, value on hand and months on hand.

10. Future work

As discussed in the Results and Discussion section, Instron has potential for improving its operations management. The result of this work is reducing wastes in the inventory management. Some topics from this research, which can be further explored, are:

10.1 Lead time variability

Lead time variability is critical to every inventory policy. Variation in lead time can lead to unexpected stock outs or surges in inventory leading to increased costs and unsatisfied customers. This issue can be taken into account if the variation in lead time is known. If Instron Corporation keeps track of lead times as described in the recommendations section, the variability can be recorded and implemented inside the replenishment policy.

10.2 Manufacturing constraints

Manufacturing constraints are essential on a factory floor since mostly limited work force is available to accomplish tasks. Orders sometimes need to be rescheduled, or in the worst case lost, if manufacturing constraints and pending commitments are not taken into consideration while promising a lead time to a customer. Thus, while determining the finished goods and part levels, it is important to consider the manufacturing constraints since if these are not considered, unrealistic levels will be obtained. At the same time, the initial analysis has revealed that most of the manufacturing constraints are both independent and difficult to quantify.

Currently, final finished good levels are checked and compared by the inventory planning team before implementing. Also, the lead times have been increased to account for manufacturing constraints (Facelli [27]). However, the optimum method to implement this would be to consider the constraints inside the optimization and simulation itself. This will make the new inventory levels faster to implement and easily reusable.

10.3 Include back orders in the simulation

As discussed in Chugh [28], simulation has been developed on a simplified model of the manufacturing floor. Back orders have not been considered in the simulation and immediate order execution is being done. However, in reality, back orders will cause the orders to wait longer than required. Implementing back orders in the simulation is a complex process and needs the creation of a new database to keep track of them. Also, some orders are unexpectedly delayed due to incomplete payments, quality audits, etc. A more accurate picture can be obtained if back orders and manufacturing time is considered inside the simulation.

10.4 Include part level into the simulation

Currently the simulation tool only considers the finished goods level. The part level inventory has been determined directly under the condition that it has to be available with a very high probability whenever the finished goods need to be prepared. This, however, is an approximation and there is a miniscule probability that an order cannot be satisfied if a part level inventory is not available. Thus, it is required that a simulation be built which starts from the part level inventory, develops finished goods and finally executes the orders. This simulation will be a more accurate representation of the factory floor.

10.5 Q,r policy using Poisson distributed demand

As shown in Chapter 7, Instron's monthly demand for frames can be better approximated with a Poisson distribution [26]. The assumption of normally distributed and continuous demand fits well the reality if the average demand is large enough. However for many products at Instron the sales volume is limited and it might then be interesting to perform a similar analysis with a Q,r policy assuming Poisson distributed demand. An in depth study can provide detailed results on whether changing the demand distribution can lead to increased profits.

10.6 Category-wise Optimization

Optimization is a complex process to run every time. It gives the service levels for each item such that an optimal mix is obtained. However, having different service level for every item can lead to confusion while undergoing policy revisions and corrections. Currently, the factory floor operates on dividing the products into categories based on values having very high service levels for each item in every category.

An optimization framework can be implemented which can present discrete service levels for such categories. The benefit of using such a method is that not only will the manufacturing planners will have easy control and understanding over such a system but, also that the correct mix of products will be available while working within the same framework. However, this solution would be less optimal than the solution proposed in this work and its implementation may still be complex.

Appendix

The code use for the implementation of the optimization tool is the following.

Table A.1 – The parent script

```
%Correlation

%This script performs the analysis of the correlation and of the demand for
the System orders,
% the calculation of the total profit and the demand for the OTC orders

% IMPORTANT NOTES about the orders data spreadsheets:
% 1) starting and ending date for the Systems files must coincide with the
% ones for the OTC files
% 2) the files must be sorted by order numbers

%OUTPUT:
%ItemNumber - Correlation - LeadTime - LotSize - QuarterlyAvg -
%QuarterlyVar

close all; clear all;

%PARAMETERS:
%Columns of the excel spreadsheets with the sales data
COL.OrderNumber=1;
COL.ALevel=4;
COL.OrderClass=5;
COL.ItemNumber=6;
COL.OrderQty=7;
COL.QtyShipped=8;
COL.StdCost=9;
COL.ExtNetPrice=12;
COL.OrderEntry=13;
COL.PostCode=14;

%Columns of the excel spreadsheet with the items to optimize
C.LS=5;
C.LT=6;
C.Cat=7;

CATNUM=8; %number of categories
Cat=cell(CATNUM,1);
%category identifiers
Cat{1}='GR'; %grip
Cat{2}='FI'; %fixture
Cat{3}='FA'; %face
Cat{4}='CO'; %coupling or adapter
Cat{5}='AC'; %other accessory
Cat{6}='EX'; %extensometer
```

```
Cat{7}='GA'; %gauge
Cat{8}='CH'; %chamber

%tab names
Tab1='SBCLIENT'; %name of the spreadsheet with items to consider

% Ask how many files with sales records are available for the analysis
fprintf('How many files with orders will you insert?\n ');
fprintf('*** Note that the number of files for Systems \nmust be equal to the number of files for OTC ');
ny=input('*** \n\n>: ');

if mod(ny,2)
    fprintf('\n\n\n\n!!!Warning: the total number of files with sales records for \n');
    fprintf('Systems and OTC must be the same, and their sum must be EVEN!!!\n\n\n\n\n');
    beep
end

fileNameEm=cell(ny/2,1);
fileNameOtc=cell(ny/2,1);

% Determine all their names

for y=1:ny/2

    fprintf('Insert the name of the file #%d with EM orders\n (Example: newem2008.xls)\n',y);
    in=input('>: ', 's');
    fileNameEm{y}=in;

end

for y=1:ny/2

    fprintf('Insert the name of the file #%d with OTC orders\n (Example: newotc2008.xls)\n',y);
    in=input('>: ', 's');
    fileNameOtc{y}=in;

end

% Ask the name of the file with the items to consider
fprintf('\n\nInsert the name of the excel files with the items to consider (Example: items.xls)\n');
fileName1=input('\n>: ','s');

% Ask for the shift in the demand
fprintf('\nInsert the estimated percentage shift in the demand\n as compared to the inserted data ' );
fprintf('\n(example: if the demand increases by 50%% insert 50)\n');
```

```
fprintf('(example: if no shift is expected insert 0)\n');
big_shift=input('\n>: ');
big_shift=big_shift/100+1;

%Import data about New products and substitutions
[num_shift, txt_shift, raw_shift] = xlsread('Newproducts.xls',
'DemandShift');
[num_subst, txt_subst, raw_subst] = xlsread('Newproducts.xls',
'Substitutions');
clear num_shift txt_shift num_subst txt_subst

%%% Import the list of items we must consider
%Read Items into Matlab matrix (Data1)
[NData1, TXTData1, Data1] = xlsread(fileName1, Tab1);
N1=size(Data1,1);

%convert all item numbers to strings (character arrays)
for i=2:N1
      if isequalwithequalnans(Data1(i,1),{NaN})
          N1=i-1;
          break
      end

      if (iscellstr(Data1(i,1))==0)
          Data1{i,1}=num2str(NData1(i,1));

      else
          Data1{i,1}=char(TXTData1(i,1));

      end
end

%build a file containing the item number, the descriptio, the categories,
%the lot sizes and the lead times of the analyzed items
Items = cell(N1-1,1);
Category = cell(N1-1,1);
LeadTime = zeros(N1-1,1);
LotSize = zeros(N1-1,1);
Description = cell(N1-1,1);

%find the considered items within the index
for j=2:N1
    Items{j-1,1}=Data1{j,1}; %j-1 so that there is no title
end

for i=1:N1-1
    for c=1:CATNUM
        if strcmp(Data1(i+1,C.Cat),Cat(c))
            Category{i,1}=Cat{c};
            break
        else
            Category{i,1}='UN'; %Undefined category used if no category
                                 %is provided by the user
```

```
        end
    end

    if Data1{i+1,C.LT}==0 || isequalwithequalnans(Data1(i+1,C.LT),{NaN});
        Data1{i+1,C.LT}=0;
        fprintf('Warning: Lead time for Item %s is ZERO or MISSING\n',
Data1{i+1,1});
    end
    LeadTime(i,1)=Data1{i+1,C.LT};

    if isequalwithequalnans(Data1(i+1,C.LS),{NaN});
        Data1{i+1,C.LS}=0;
        fprintf('Warning: Lot size for Item %s is MISSING\n', Data1{i+1,1});
    end
    LotSize(i,1)=Data1{i+1,C.LS};

    if isequalwithequalnans(Data1(i+1,2),{NaN});
        Data1{i+1,2}='';
    end
    Description{i,1}=Data1{i+1,2};

end

%%% CORRELATION AN PROFIT ANALYSIS: START
index=[]; %create one index for all the items
costs=[];
profits=[];
count=[];
OmegaAll=[];
NWAll=[];
PrOrd=[];

for y=1:ny/2

    % Systems Analysis
    [VP, index, costs, profits, count, demandEm, quartersEm, OmegaEm,...
        NWEm,PrOrdSyst]=SystemsAnalysis(fileNameEm{y}, index, costs, profits,
count,...
        COL, raw_shift, raw_subst, big_shift, Items, Category,
LeadTime/(365/4));
    OmegaAll=vertcat(OmegaAll,OmegaEm);
    NWAll=vertcat(NWAll,NWEm);
    PrOrd=vertcat(PrOrd,PrOrdSyst);

    % OTC Analysis
    [TP, index, costs, profits, count, demandOtc, quartersOtc, OmegaOtc,...
        NWOtc,PrOrdOtc]=OtcAnalysis(fileNameOtc{y}, index, costs, profits,
count,...
        COL, raw_shift, raw_subst, big_shift, Items, Category,
LeadTime/(365/4));
    OmegaAll=vertcat(OmegaAll,OmegaOtc);
    NWAll=vertcat(NWAll,NWOtc);
    PrOrd=vertcat(PrOrd,PrOrdOtc);
```

```
    if size(quartersEm,1)~=size(quartersOtc,1)
        fprintf('\n*** Warning: the dates used for the extraction of the
files for System and Otc orders must COINCIDE ***\n');
    end

    %results collection
    if y==1
        SysQuartDemand=demandEm;
        OtcQuartDemand=demandOtc;
        VirtualProfit=VP;
        TotalProfit=TP;
        totQuarters=quartersEm;

    else
        SysQuartDemand=[[SysQuartDemand;zeros(size(demandEm,1)-
size(SysQuartDemand,1),size(SysQuartDemand,2))], demandEm];
        OtcQuartDemand=[[OtcQuartDemand;zeros(size(demandOtc,1)-
size(OtcQuartDemand,1),size(OtcQuartDemand,2))], demandOtc];
        VirtualProfit=[[VirtualProfit;zeros(length(VP)-
length(VirtualProfit),size(VirtualProfit,2))], VP];
        TotalProfit=[[TotalProfit;zeros(length(TP)-
length(TotalProfit),size(TotalProfit,2))], TP];
        totQuarters=[totQuarters;quartersEm];
    end

clear demandEm demandOtc VP TP
end

%the last index for the Systems is the less updated than the last one for the
%OTC
VirtualProfit=[VirtualProfit;zeros(length(index)-
length(VirtualProfit),size(VirtualProfit,2))];
SysQuartDemand=[SysQuartDemand;zeros(length(index)-
length(SysQuartDemand),size(SysQuartDemand,2))];

%calculate the mean values
avgCosts=costs./count;
avgProfits=profits./count;
VirtualProfit = sum(VirtualProfit,2)/(length(totQuarters)/4);
TotalProfit = sum(TotalProfit,2)/(length(totQuarters)/4);

%statistical analysis of the demand
SysQuartMu=mean(SysQuartDemand,2);
SysQuartVar=var(SysQuartDemand,0,2);
OtcQuartMu=mean(OtcQuartDemand,2);
OtcQuartVar=var(OtcQuartDemand,0,2);
TotQuartMu=mean(SysQuartDemand+OtcQuartDemand,2);
TotQuartVar=var(SysQuartDemand+OtcQuartDemand,0,2);

%%% END
```

```
%%% Skimming of the results

VProfit = zeros(N1-1,1);
TProfit = zeros(N1-1,1);
EMQuartMu = zeros(N1-1,1);
EMQuartVar = zeros(N1-1,1);
OTCQuartMu = zeros(N1-1,1);
OTCQuartVar = zeros(N1-1,1);
TOTQuartMu = zeros(N1-1,1);
TOTQuartVar = zeros(N1-1,1);
DemandOverLT = zeros(N1-1,1);
unitCost = zeros(N1-1,1);
unitProfit = zeros(N1-1,1);

%Include a categorization of items

for i=1:N1-1

    for x=1:length(index)
        if strcmpi(Items(i,1),index(x,1))
            VProfit(i,1)=VirtualProfit(x,1);
            TProfit(i,1)=TotalProfit(x,1);
            EMQuartMu(i,1)=SysQuartMu(x,1);
            EMQuartVar(i,1)=SysQuartVar(x,1);
            OTCQuartMu(i,1)=OtcQuartMu(x,1);
            OTCQuartVar(i,1)=OtcQuartVar(x,1);
            TOTQuartMu(i,1)=TotQuartMu(x,1);
            TOTQuartVar(i,1)=TotQuartVar(x,1);
            unitCost(i,1)=avgCosts(x,1);
            unitProfit(i,1)=avgProfits(x,1);
         end
    end
    DemandOverLT(i,1)=TOTQuartMu(i,1)*(Data1{i+1,C.LT}/(365/4));
end

%Compute set of possible solution
Q=[];
R=[];
MOH=[];
VOH=[];
ELoss=[];

setepsilon=[0.25;0.5;0.75;1;1.5;2;3;5;6;9;12;25;30;50;100;150;200;250;500];

for i=1:size(setepsilon,1)
    epsilon=setepsilon(i,1)/100;
    [z,Qcurr,Rcurr,MOHcurr,VOHcurr,EITot]=Optimization(Items, LeadTime,
unitCost, unitProfit, TOTQuartMu, ...
        TOTQuartVar, EMQuartMu, EMQuartVar, VProfit, OTCQuartMu, ...
        OTCQuartVar, TProfit, LotSize, Category,epsilon);

[MaxProfit,F,Hc]=ComputeELoss(z,Rcurr,Qcurr,unitCost,sqrt(TOTQuartVar),...
        totQuarters,epsilon,LeadTime/(365/4),OmegaAll,NWAll,PrOrd);
```

```
    Q=horzcat(Q, Qcurr);
    R=horzcat(R, Rcurr);
    VOH=vertcat(VOH, VOHcurr);
    MOH=vertcat(MOH, MOHcurr);
    ELoss=vertcat(ELoss, MaxProfit+F);
end


%Choose solution
[nsol]=LevelsChoice(ELoss,MOH,VOH);



PrintResults(Items,Description,LeadTime,R(:,nsol),Q(:,nsol),fileName1,unitCos
t, TOTQuartMu...
    ,TOTQuartVar, MOH(nsol), VOH(nsol));

% Ask for Kanban Cards printing
cchoice=[];
while (~strcmp(cchoice, 'Y') && ~strcmp(cchoice, 'N'))
    fprintf('\n\nDo you want to print Kanban Cards?\n' );
    fprintf('Type Y/N\n');
    cchoice=input('\n>: ', 's' );
end

if strcmp(cchoice, 'Y')
    PrintKanban(Items,Description,Q(:,nsol),R(:,nsol),LeadTime);
end

save('simulation_data', 'index', 'avgProfits');
```

Table A.2 – Function SystemsAnalysis.m

```
% Systems analysis
function [VirtualProfits, index, costs, profits, count, demand, quarters,
OmegaInt, NWInt,PrOrd]=SystemsAnalysis(fileName, index, costs, profits,
count, COL, raw_shift, raw_subst, big_shift, indexInt, Category, L)

%Import skim data from spreadsheets
[Regular, Otc] = ImportData2('EM',fileName,[]);

R=size(Regular,1);

%% Substitutions
% Demand shifts
shifted_prod={};
for i=2:size(raw_shift,1)
    if isequalwithequalnans(raw_shift(i,1),{NaN})
        break
    end
    shifted_prod(i-1)=raw_shift(i,1);
    shift_amount(i-1)=raw_shift(i,2);
end

% Substitutions
old_prod={};
for i=2:size(raw_subst,1)
    if isequalwithequalnans(raw_subst(i,1),{NaN})
        break
    end
    old_prod(i-1)=raw_subst(i,1);
    prob(i-1)=raw_subst{i,2};
    new_prod1(i-1)=raw_subst(i,3);
    new_price1(i-1)=raw_subst(i,4);
    new_cost1(i-1)=raw_subst(i,5);
    new_prod2(i-1)=raw_subst(i,6);
    new_price2(i-1)=raw_subst(i,7);
    new_cost2(i-1)=raw_subst(i,8);
end

for i=1:size(Regular,1)
    %correct qty shipped considering total mean shift
    Regular{i,COL.QtyShipped}=big_shift*Regular{i,COL.QtyShipped};
    Regular{i,COL.OrderQty}=big_shift*Regular{i,COL.OrderQty};
    Regular{i,COL.ExtNetPrice}=big_shift*Regular{i,COL.ExtNetPrice};

    %check if it is a product with demand shift
    if ~isempty(shifted_prod)
        for l=1:size(shifted_prod,2)
            if (isequal(Regular{i,COL.ItemNumber},shifted_prod{l}))

Regular{i,COL.QtyShipped}=shift_amount{l}*Regular{i,COL.QtyShipped};

Regular{i,COL.OrderQty}=shift_amount{l}*Regular{i,COL.OrderQty};
```

```
            end
        end
    end

    % check if it is a product with substitution
    if ~isempty(old_prod)
        for l=1:size(old_prod,2)
            if (isequal(Regular{i,COL.ItemNumber},old_prod{l}))
                % the grip is one of the obsoleted ones
                if rand()<=prob(l)
                    Regular{i,COL.ItemNumber}=new_prod1{l};
                    Regular{i,COL.StdCost}=new_cost1{l};
                    Regular{i,COL.ExtNetPrice}=new_price1{l};
                else Regular{i,COL.ItemNumber}=new_prod2{l};
                    Regular{i,COL.StdCost}=new_cost2{l};
                    Regular{i,COL.ExtNetPrice}=new_price2{l};
                end
            end
        end
    end
end
%%
% identify starting and ending quarters (they must conincide for Systems and
OTC
% files)

date=zeros(R,1);
% create array with dates in numbers
for i=1:length(Regular(:,13))
    date(i)=Regular{i,13};
end
% find initial and final month
m=min(date);
M=max(date);

clear date

% find last day of last month in the orders
D=datevec(M);
D(3)=1;
N1=addtodate(M,1,'month');
D1=datevec(N1);
D1(3)=1;
daysinlastmonth=datenum(D1)-datenum(D);
D(3)=daysinlastmonth;
%find first day of first month in the orders
d=datevec(m);
d(3)=1;

% count how many quarters are taken into consideration
years=D(1)-d(1)+1; %how many years

%find first quarter
```

```
firstQuarter=ceil(d(2)/3);
if mod((d(2)+2),3)
    fprintf('*** Warning: the dates must start from the first month of a
quarter ***\n');
end

%find last quarter
lastQuarter=ceil(D(2)/3);
if mod(D(2),3)
    fprintf('*** Warning: the dates must end in the last month of a quarter
***\n');
end

cc=1; %counter for the quarters identification
for y=1:years
    for q=1:4
        if (y>1 && y<years) || (y==1 && y~=years && q>=firstQuarter) || (y~=1
&& y==years && q<=lastQuarter) || (y==1 && y==years && q>=firstQuarter &&
q<=lastQuarter)
            quarters(cc,1)=q;
            quarters(cc,2)=d(1)+y-1;
            cc=cc+1;
        end
    end
end

%Create an index for all the item numbers
k=0;
Ind=length(index);

if isempty(index)
    k=1;
    index=Regular(1,COL.ItemNumber);
    costs=0;
    profits=0;
    count=0;
end

for i=1:R
    for j=1:Ind+k
        flag=0;
        if (isequal(index(j,1),Regular(i,COL.ItemNumber)))
            flag=1;
            break
        end
    end;
    if (flag==0)
        k=k+1;
        index(Ind+k,1)=Regular(i,COL.ItemNumber);
        costs(Ind+k,1)=0;
        profits(Ind+k,1)=0;
        count(Ind+k,1)=0;
    end
```

```
end

%Create an order index
%Note that a "unique orders" index is created for Systems Order
indexP=Regular(:,COL.OrderNumber); %take Data column
k=1;
indexo(k,1)=indexP(1);
for i=1:length(indexP)
    for j=1:k
        flag=0;
        if (isequal(indexo(j,1),indexP(i)))
            flag=1;
            break
        end
    end;
    if (flag==0)
        k=k+1;
        indexo(k,1)=indexP(i);
    end;
end;
clear indexP

%%build W functions, best thing is add two columns with gamma and eta to
%%every item
%import file
%define Customer expectations curve name
WfileName = 'survey.xls';

%Read Spreadsheet into Matlab matrix
curves = xlsread(WfileName, 'FINAL DATA');

%creates OmegaInt containing which interesting items are in every order
%also creates NWInt which contain the quantities and the W relative to the
%items stored in OmegaInt
currentorder={'null'};
j=0;
OmegaInt=cell(length(indexo),1);
NWInt=cell(length(indexo),1);
PrOrd=zeros(length(indexo),1);
for i=1:size(Regular,1)
    %select order
    if (~strcmpi(currentorder,Regular(i,COL.OrderNumber)));
        j=j+1;
        currentorder=Regular(i,COL.OrderNumber);
    end
    PrOrd(j,1)=PrOrd(j,1)+Regular{i,COL.ExtNetPrice}-
Regular{i,COL.StdCost}*...
        (Regular{i,COL.QtyShipped}+Regular{i,COL.OrderQty});
    %find item in the indexInt
    for l=1:length(indexInt)
        %if the item is found in indexInt
        if (strcmp(indexInt(l),Regular(i,COL.ItemNumber)))
            %pick category
            category={'null'};
```

```
            %If it is a system order
            if (strcmp('EM', Regular(i,COL.ALevel)))
                %check the system type
                %if it is a real system (the Order Class is a number)
                if ~iscellstr(Regular(i,COL.OrderClass))
                    %convert the Order Class into a string and store it
                    category=num2str(Regular{i,COL.OrderClass});
                    %else it is a special EM order without a frame; in
this...
                    %case the value 'EM' is used to tell to the customer...
                    %expectations function to treat it like a general EM
order
                else
                    category=Regular{i,COL.ALevel};
                end
                %If it is an OTC order
            else if (strcmp('OTC', Regular(i,COL.OrderClass)))
                    category=Category{1,1};
                else
                    fprintf('Error in evaluating order type \n');
                end
            end
            %write in OmegaInt that item l is part of order j
            %write in NWInt the quantity with which item l is sold in order
            %j and the W which should be used
            if isempty (OmegaInt{j,1})
                OmegaInt(j,1)={l};

NWInt(j,1)={Regular{i,COL.OrderQty}+Regular{i,COL.QtyShipped}};
                NWInt{j,2}=customersexp(curves, category, L(l,1));
            else
                OmegaInt{j,1}=vertcat(OmegaInt{j,1},l);

NWInt{j,1}=vertcat(NWInt{j,1},Regular{i,COL.OrderQty}+Regular{i,COL.QtyShippe
d});
                NWInt{j,2}=vertcat(NWInt{j,2}, customersexp(curves, category,
L(l,1)));
            end
        end
    end
end;


%%% store the daily demand, the costs and the profits for every item and
%count how many times they appear in the spreadsheet

demand=zeros(length(index),size(quarters,1));

for j=1:R

    for i=1:length(index)

        %this is for robustness to strange data
```

```
        if iscell(Regular{i,COL.QtyShipped})
            Regular{i,COL.QtyShipped}=Regular{i,COL.QtyShipped}{1};
        end

        if strcmp(index(i),Regular(j,COL.ItemNumber))
            costs(i)=costs(i)+Regular{j,COL.StdCost};

profits(i)=profits(i)+(Regular{j,COL.ExtNetPrice}/(Regular{j,COL.QtyShipped}+
Regular{j,COL.OrderQty})-Regular{j,COL.StdCost});
            count(i)=count(i)+1;
            %identify quarter
            RDate=datevec(Regular{j,COL.OrderEntry});
            Quart=ceil(RDate(2)/3);
            for l=1:size(quarters,1)
                if sum(quarters(l,:)==[Quart, RDate(1)])==2

demand(i,l)=demand(i,l)+(Regular{j,COL.QtyShipped}+Regular{j,COL.OrderQty});
                end
            end
        end

    end

end

% Calculate the Virtual Profit

VirtualProfits=zeros(length(index),1);

j=[];
NetProfit=0;
currentorder={'null'};

for i=1:R
    if (strcmpi(currentorder,'null'));
        currentorder=Regular(i,1);
    end;
    if (~strcmpi(Regular(i,1),currentorder))
        VirtualProfits(j)=VirtualProfits(j)+NetProfit;
        j=[];
        NetProfit=0;
        currentorder=Regular(i,1);
    end;

    %update net profit for order i

    %this is for robustness to strange data
    if iscell(Regular{i,COL.QtyShipped})
        Regular{i,COL.QtyShipped}=Regular{i,COL.QtyShipped}{1};
    end

    NetProfit=NetProfit+(Regular{i,COL.ExtNetPrice}-
```

```
Regular{i,COL.StdCost}*(Regular{i,COL.QtyShipped}+Regular{i,COL.OrderQty}));

    %find item in the index
    for l=1:length(index)
        if (strcmp(index(l),Regular(i,COL.ItemNumber)))
            break
        end
    end
    j=vertcat(j,l);
end

VirtualProfits(j)=VirtualProfits(j)+NetProfit;

clear Regular
```

Table A.3 – Function OtcAnalysis.m

```
% Otc analysis
function [TotalProfits, index, costs, profits, count, demand, quarters,
OmegaInt, NWInt,PrOrd]=OtcAnalysis(fileName, index, costs, profits, count,
COL, raw_shift, raw_subst, big_shift, indexInt, Category, L)

%Import skim data from spreadsheets
[Regular, Otc] = ImportData2('OTC',[],fileName);

O=size(Otc,1);

%%Orders substitutions
% Demand shifts
shifted_prod={};
for i=2:size(raw_shift,1)
    if isequalwithequalnans(raw_shift(i,1),{NaN})
        break
    end
    shifted_prod(i-1)=raw_shift(i,1);
    shift_amount(i-1)=raw_shift(i,2);
end

% Substitutions
old_prod={};
for i=2:size(raw_subst,1)
    if isequalwithequalnans(raw_subst(i,1),{NaN})
        break
    end
    old_prod(i-1)=raw_subst(i,1);
    prob(i-1)=raw_subst{i,2};
    new_prod1(i-1)=raw_subst(i,3);
    new_price1(i-1)=raw_subst(i,4);
    new_cost1(i-1)=raw_subst(i,5);
    new_prod2(i-1)=raw_subst(i,6);
    new_price2(i-1)=raw_subst(i,7);
    new_cost2(i-1)=raw_subst(i,8);
end

for i=1:size(Otc,1)
    %correct qty shipped considering total mean shift
    Otc{i,COL.QtyShipped}=big_shift*Otc{i,COL.QtyShipped};
    Otc{i,COL.OrderQty}=big_shift*Otc{i,COL.OrderQty};
    Otc{i,COL.ExtNetPrice}=big_shift*Otc{i,COL.ExtNetPrice};

    %check if it is a product with demand shift
    if ~isempty(shifted_prod)
        for l=1:size(shifted_prod,2)
            if (isequal(Otc{i,COL.ItemNumber},shifted_prod{l}))
                Otc{i,COL.QtyShipped}=shift_amount{l}*Otc{i,COL.QtyShipped};
                Otc{i,COL.OrderQty}=shift_amount{l}*Otc{i,COL.OrderQty};
            end
        end
```

```
    end

    % check if it is a product with substitution
    if ~isempty(old_prod)
        for l=1:size(old_prod,2)
            if (isequal(Otc{i,COL.ItemNumber},old_prod{l}))
                % the grip is one of the obsoleted ones
                if rand()<=prob(l)
                    Otc{i,COL.ItemNumber}=new_prod1{l};
                    Otc{i,COL.StdCost}=new_cost1{l};
                    Otc{i,COL.ExtNetPrice}=new_price1{l};
                else Otc{i,COL.ItemNumber}=new_prod2{l};
                    Otc{i,COL.StdCost}=new_cost2{l};
                    Otc{i,COL.ExtNetPrice}=new_price2{l};
                end
            end
        end
    end
end
%%
% identify starting and ending quarters (they must conincide for Systems and
OTC
% files)

date=zeros(0,1);
% create array with dates in numbers
for i=1:length(Otc(:,13))
    date(i)=Otc{i,13};
end
% find initial and final month
m=min(date);
M=max(date);

clear date

% find last day of last month in the orders
D=datevec(M);
D(3)=1;
N1=addtodate(M,1,'month');
D1=datevec(N1);
D1(3)=1;
daysinlastmonth=datenum(D1)-datenum(D);
D(3)=daysinlastmonth;
%find first day of first month in the orders
d=datevec(m);
d(3)=1;

% count how many quarters are taken into consideration
years=D(1)-d(1)+1; %how many years

%find first quarter
firstQuarter=ceil(d(2)/3);
if mod((d(2)+2),3)
```

```
    fprintf('*** Warning: the dates must start from the first month of a quarter ***\n');
end

%find last quarter
lastQuarter=ceil(D(2)/3);
if mod(D(2),3)
    fprintf('*** Warning: the dates must end in the last month of a quarter ***\n');
end

cc=1; %counter for the quarters identification
for y=1:years
    for q=1:4
        if (y>1 && y<years) || (y==1 && y~=years && q>=firstQuarter) || (y~=1 && y==years && q<=lastQuarter) || (y==1 && y==years && q>=firstQuarter && q<=lastQuarter)
            quarters(cc,1)=q;
            quarters(cc,2)=d(1)+y-1;
            cc=cc+1;
        end
    end
end

%Create an index for all the item numbers
k=0;
Ind=length(index);

if isempty(index)
    k=1;
    index=Otc(1,COL.ItemNumber);
    costs=0;
    profits=0;
    count=0;
end

for i=1:O
    for j=1:Ind+k
        flag=0;
        if (isequal(index(j,1),Otc(i,COL.ItemNumber)))
            flag=1;
            break
        end
    end;
    if (flag==0)
        k=k+1;
        index(Ind+k,1)=Otc(i,COL.ItemNumber);
        costs(Ind+k,1)=0;
        profits(Ind+k,1)=0;
        count(Ind+k,1)=0;
    end
end

```

```
%The Otc orders are added as "every line is a different order". This
%is necessary to consider the fact that items are assumed to be shippable
%separately if necessary in otc orders
%A fictious incremental number is given to the Otc orders so that every
%line of the Xls has now a unique order number
for i=1:size(Otc,1)
    Otc{i,COL.OrderNumber}=horzcat(Otc{i,COL.OrderNumber}, 'OTC');
    Otc{i,COL.OrderNumber}=horzcat(Otc{i,COL.OrderNumber}, num2str(i));
end

%%build W functions, best thing is add two columns with gamma and eta to
%%every item
%import file
%define Customer expectations curve name
WfileName = 'survey.xls';

%Read Spreadsheet into Matlab matrix
curves = xlsread(WfileName, 'FINAL DATA');

%Creates OmegaInt containing which interesting items are in every order
%also creates NWInt which contain the quantities and the W relative to the
%items stored in OmegaInt
A=size(Otc,1);
currentorder={'null'};
j=0;
OmegaInt=cell(A,1);
NWInt=cell(A,1);
PrOrd=zeros(0,1);
for i=1:size(Otc,1)
    %select order
    if (~strcmpi(currentorder,Otc(i,COL.OrderNumber)));
        j=j+1;
        currentorder=Otc(i,COL.OrderNumber);
    end
    PrOrd(j,1)=PrOrd(j,1)+Otc{i,COL.ExtNetPrice}-Otc{i,COL.StdCost}*...
        (Otc{i,COL.QtyShipped}+Otc{i,COL.OrderQty});
    %find item in the indexInt
    for l=1:length(indexInt)
        %if the item is found in indexInt
        if (strcmp(indexInt(l),Otc(i,COL.ItemNumber)))
            %pick category
            category={'null'};
            %If it is a system order
            if (strcmp('EM', Otc(i,COL.ALevel)))
                %check the system type
                %if it is a real system (the Order Class is a number)
                if ~iscellstr(Otc(i,COL.OrderClass))
                    %convert the Order Class into a string and store it
                    category=num2str(Otc{i,COL.OrderClass});
                    %else it is a special EM order without a frame; in
this...
                    %case the value 'EM' is used to tell to the customer...
                    %expectations function to treat it like a general EM
order
```

```
                else
                    category=Otc{i,COL.ALevel};
                end
                %If it is an OTC order
            else if (strcmp('OTC', Otc(i,COL.OrderClass)))
                    category=Category{l,1};
                else
                    fprintf('Error in evaluating order type \n');
                end
            end
            %write in OmegaInt that item l is part of order j
            %write in NWInt the quantity with which item l is sold in order
            %j and the W which should be used
            if isempty (OmegaInt{j,1})
                OmegaInt(j,1)={l};
                NWInt(j,1)={Otc{i,COL.OrderQty}+Otc{i,COL.QtyShipped}};
                NWInt{j,2}=customersexp(curves, category, L(l,1));
            else
                OmegaInt{j,1}=vertcat(OmegaInt{j,1},l);

NWInt{j,1}=vertcat(NWInt{j,1},Otc{i,COL.OrderQty}+Otc{i,COL.QtyShipped});
                NWInt{j,2}=vertcat(NWInt{j,2}, customersexp(curves, category,
L(l,1)));
            end
        end
    end
end;


%store the daily demand, the costs and the profits for every item and
%count how many times they appear in the spreadsheet

demand=zeros(length(index),size(quarters,1));
TotalProfits=zeros(length(index),1);

for j=1:O

    for i=1:length(index)

        %this is for robustness to strange data
        if iscell(Otc{i,COL.QtyShipped})
            Otc{i,COL.QtyShipped}=Otc{i,COL.QtyShipped}{1};
        end

        if strcmp(index(i),Otc(j,COL.ItemNumber))
            costs(i)=costs(i)+Otc{j,COL.StdCost};

profits(i)=profits(i)+(Otc{j,COL.ExtNetPrice}/(Otc{j,COL.QtyShipped}+Otc{j,CO
L.OrderQty})-Otc{j,COL.StdCost});
            count(i)=count(i)+1;
            TotalProfits(i)=TotalProfits(i)+Otc{j,COL.ExtNetPrice}-
Otc{j,COL.StdCost}*(Otc{j,COL.QtyShipped}+Otc{j,COL.OrderQty});
            %identify quarter
```

```
            RDate=datevec(Otc{j,COL.OrderEntry});
            Quart=ceil(RDate(2)/3);
            for l=1:size(quarters,1)
                if sum(quarters(l,:)==[Quart, RDate(1)])==2

demand(i,l)=demand(i,l)+(Otc{j,COL.QtyShipped}+Otc{j,COL.OrderQty});
                end
            end
        end

    end

end

clear Otc
```

References

[1] Zied Jemai, Fikri Karaesmen, The influence of demand variability on the performance of a make-to-stock queue, European Journal of Operational Research, Volume 164, Issue 1, 1 July 2005, Pages 195-205, ISSN 0377-2217

[2] Rachel Q. Zhang, "Expected Time Delay in Multi-Item Inventory Systems with Correlated Demands, John Wiley & Sons, 1999

[3] Daniel H. Wheeler, "Pulling a Job Shop into Supply Chain Management", 2000

[4] E. A. Silver, D. F. Pyke, and R. Peterson, "Inventory Management and Production Planning and Scheduling", 3rd ed. (1998), John Wiley & Sons, New York, p.33

[5] Garret J. Caterino, "Implementation of Lean Manufacturing in a Low-Volume Production Environment", 2001

[6] Hiep T. Nguyen, "Improving On-Time Delivery Performance Through the Implementation of Lean Supply Chain Management", 2002

[7] D. Smichi-Levi, P. Kaminsky, E. Simchi Levi, 2000, "Designing & Managing the Supply Chain", McGraw Hill

[8] Hadley, G. and T. M. Whitin, 1963 "Analysis of Inventory Systems", Prentice- Hall, Englewood Cliffs.

[9] Zipkin, P., 1986, "Inventory Service-Level Measures: Convexity and Approximations", Management Sci., 32, 975-981.

[10] Federgruen, A. and Y.-S. Zheng, 1992, "An Efficient Algorithm for Computing an Optimal (r, Q) Policy in Cont. Review Stochastic Inventory Systems", Oper. Res., 40, 808-813.

[11] Ming-Zheng Wang; Wen-Li Li , 2007, "Convexity of Service-Level Measures of the Discrete (r,Q) Inventory System", Second International Conference on Innovative Computing, Information and Control, Page(s):417 - 417

[12] S.A. Smith, J.C. Chambers, E. Shlifer, 1980, "Optimal inventories based on job completion rate for repairs requiring multiple items", Management Science 26, 849-853

[13] R. Zheng, 1999, "Expected Time Delay in Multi-Item Inventory Systems with Correlated Demands", Naval Research Logistics 46, 671-688

[14] L. Liu, X. M. Youan, 2000, "Coordinated replenishments in inventory systems with correlated demands", European Journal of Operation Research 123, 490-503

[15] H. Feng, K. Muthuraman, V. Deshpande, 2007, "The Impact of Demand Correlation on Replenishment Policies for Multi-Product Stochastic Inventory Systems with Joint-Replenishment Costs", McCombs Research Paper Series No. IROM-08-08

[16] Chieh-Y. Tsai, Chi-Y Tsai, Po-Wen Huang, 2009, "An association clustering algorithm for can-order policies in the joint replenishment problem", International Journal of Production economics 117, 30-41

[17] W.H. Hausman, H.L. Lee, A.X. Zhang, 1998, "Joint demand fulfillment probability in a multi-item inventory system with independent order-up-to policies", European Journal of Operation Research 109, 646-659

[18] Tai-Yue Wang, Jui-Ming Hu, 2008, "An Inventory control system for products with optional components under service level and budget constraints", European Journal of Operation Research 189, 41-58

[19] S.B. Gershwin, B. Tan, M.H. Veatch, 2009, "Production control with backlog-dependent demand", IIE Transaction 41, Issue 6, 511-523

[20] M. H. Veatch, 2009, "The Impact of Customer Impatience on Production Control", IIE Transactions 41, Number 2, 95-102

[21] C. Z. Mooney, 1999, "Monte Carlo Simulation", SAGE

[22] L.W.G. Strijbosch, J.J.A. Moors, 2006, "Modified normal demand distributions in (R,S)-inventory control", European Journal of Operational Research, vol. 172, pp 201-212

[23] J. Coakley, C. Carpenter, 1983, "Evaluation of a metric inventory system using simulation", 15th Winter Simulation Conference Proceedings, pp. 589-596

[24] J. Y. Jung, G. Blau, J. F. Pekny, G.s V. Reklaitis, D. Eversdyk, 2004, "A simulation based optimization approach to supply chain management under demand uncertainty", Computers & Chemical Engineering, Volume 28, Issue 10, pp. 2087-2106

[25] F. Grange, 1998, "Challenges in modeling demand for inventory optimization of slow-moving items", Proceedings of the 30th Winter Simulation Conference, pp 1211-1218

[26] D. Palano, 2009, "Management of a high mix production system with interdependent demands: finished goods requirements and raw materials control", M.Eng. thesis, Massachusetts Institute of Technology, Cambridge, MA

[27] A. Facelli, 2009, "Management of a high mix production system with interdependent demands: global and decomposed optimization approaches for inventory control", M.Eng. thesis, Massachusetts Institute of Technology, Cambridge, MA

[28] S. Chugh, 2009, "Management of a high mix production system with interdependent demands: simulation of proposed policies", M.Eng. thesis, Massachusetts Institute of Technology, Cambridge, MA