Optimal Inventory Control in Cardboard Box Producing Factories: A Case Study

Catherine D. Black

Thesis presented in partial fulfilment of the requirements for the degree Master of Science in Engineering Sciences at the Department of Applied Mathematics of the University of Stellenbosch, South Africa

Supervisor: Prof J.H. van Vuuren December 2004

Optimal Inventory Control in Cardboard Box Producing Factories: A Case Study

Catherine D. Black

Thesis presented in partial fulfilment of the requirements for the degree Master of Science in Engineering Sciences at the Department of Applied Mathematics of the University of Stellenbosch, South Africa

Supervisor: Prof J.H. van Vuuren July 2004

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and that I have not previously in its entirety or in part submitted it at any university for a degree.

Signature: ____________________ Date: ____________________

Abstract

This thesis is a case study in optimal inventory control, applied to *Clickabox* factory, a South African cardboard box producer from whom cardboard boxes may be ordered at short notice via the internet.

The problem of developing a decision–support system for optimal stockholding at the factory, in order to minimize cardboard off–cut wastage subject to required service levels, is addressed in this thesis. Previously a simple replenishment policy, based largely on experience, was implemented at the factory. The inventory model developed for and applied to *Clickabox* in this thesis takes account of a raw materials substitution cascade, as well as the stochasticity of demand, and other factors such as cost, service level and spatial requirements for the storage of stock. This combination of stochastic demand and product substitution has not, to the author's knowledge, previously been dealt with in the literature.

There are two primary deliverables of this study. The first is a suggestion as to the suitable stock composition (cardboard types from which boxes may be manufactured) to be kept in inventory at the factory. The second deliverable is a computerised decision–support system, based on the inventory model developed, to aid in future inventory replenishment decisions at *Clickabox*.

Some of the results of this thesis have, at the time of writing, already been implemented with success at the factory. These include the suggestions given to the management of *Clickabox* as to the suitable stock types to be held in inventory, which have been implemented in stages since March 2003. The suggested stock composition has proven to be superior to the previous stock types held, in terms of a reduction in off–cut wastage and increased availability of suitable boards.

Opsomming

Hierdie tesis is 'n gevallestudie in optimale voorraadbeheer, toegepas op *Clickabox* fabriek, 'n Suid–Afrikaanse kartondoosprodusent by wie kartondose op kort kennisgewing via die internet bestel kan word.

In hierdie tesis word 'n besluitnemingsteunstelsel ontwikkel vir optimale bestuur van voorraad by die fabriek, wat karton afknipselvermorsing onderhewig aan vereiste diensvlakke minimeer. Vantevore is 'n eenvoudige voorraad aanvullingstrategie, wat hoofsaaklik op ondervinding gebaseer was, by die fabriek toegepas. 'n Wetenskaplike gefundeerde voorraadmodel word vir *Clickabox* ontwikkel en toegepas, waarin 'n rou–voorraad kaskade–substitusie proses in aanmerking geneem word, asook die stogastiese vraag na kartondose en faktore soos prys, diensvlakke en benodigde stoorruimte. Hierdie kombinasie van stogastiese vraag en rou–voorraad kaskade–substitusie is, tot die skrywer se kennis, nog nie in die literatuur behandel nie.

Die studie het twee hoof–uitkomste ten doel. Die eerste is 'n aanbeveling ten opsigte van 'n geskikte rou–voorraad samestelling (kartontipes waaruit kartondose geproduseer kan word) wat by die fabriek in voorraad gehou moet word. Die tweede is 'n rekenaarmatige besluitnemingsteunstelsel, wat op die ontwikkelde voorraadbeheermodel gegrond is, en wat vir toekomstige besluite in verband met voorraadaanvulling by *Clickabox* bedoel is.

Van die resultate wat in hierdie tesis vervat is, is reeds ten tyde van die opskryf daarvan doeltreffend by die fabriek geïmplementeer. Ondermeer is die aanbeveling in verband met die geskikte voorraadsamestelling, geleidelik vanaf Maart 2003 by die fabriek ingefaseer. Dit het duidelik geword dat hierdie samestelling beter as die vorige voorraadprofiel funksioneer, in terme van 'n verlaging in afknipselvermorsing en 'n verhoging in die beskikbaarheid van geskikte kartonne.

Acknowledgements

The author hereby wishes to express her gratitude towards those without whose support the completion of this thesis would have been impossible:

- Prof Jan van Vuuren for his guidance, insight, patience and dedication, particularly given the difficulties of distance.
- The staff of *Clickabox*, and in particular Mr Piet Taljaard, for their enthusiastic support of the project, willingness to provide all the information required, and tolerance of my frequent presence at the factory.
- Mr Werner Grundlingh for his assistance with technical typesetting issues.
- *T-Systems SA* for their sponsorship of this research project.
- The Department of Applied Mathematics of the University of Stellenbosch, for the use of their computing facilities.
- My husband, who bore the brunt of my frustrations along the way.

Terms of Reference

This thesis is a case study in optimal inventory control at *Clickabox*, a cardboard box manufacturing company in the South African Western Cape Industria. The company's inventory control practices take into account a number of factors, most notably non-stationary, partially observed stochastic demand and cascading product substitution. To the author's knowledge this combination of factors has not previously been dealt with in the literature.

The company was founded by Mr Bob Fuller in May 1988 as *Corrucape Packaging CC*, the first private carton manufacturing plant in the greater Cape Town area. Its turnover grew steadily and reached a high point in 1996, when it achieved an annual turnover of R4.3 million. It was converted to a PTY LTD company in 1997. Stiff competition and Mr Fuller's approach with respect to remaining a one–man business led to a decline in turnover in 1997 and 1998, and as a result he decided to sell the company. In October 1999 the company was bought by Mr Piet Taljaard. The new directors, the Taljaard Family Trust, changed the company name to *Clickabox Pty Ltd* in November 2000, reflecting the company's new focus on e–commerce.

The need for a scientific inventory control process was identified during the development of a web–based interface for *Clickabox* that allows customers to receive quotes and place orders online. This website interfaces with the *Pastel* [58] accounting system, used by *Clickabox*, that maintains information on stock levels, sales, *etc.* It was developed during the period October 2000 – August 2001 by *Netcommerce Consulting* [69], a company owned and managed by Mr Leon Swanepoel.

Mr Swanepoel introduced Mr Taljaard, the director of *Clickabox* factory, to Prof Jan van Vuuren from the Department of Applied Mathematics at the University of Stellenbosch, in March 2001. At this meeting Mr Taljaard expressed his interest in a solution to his inventory control problem. Since then, the factory has twice been used as a case study for projects forming part of a project driven postgraduate course at the Department of Applied Mathematics, called *Methods of Operations Research* [25]. This thesis is the first study beyond Honours level to be conducted at *Clickabox*.

Prof van Vuuren was the supervisor for this thesis. The first meeting between the author and the director of *Clickabox* took place on 8 May 2001, and after a number of subsequent visits a research proposal was drawn up and presented to the director. The factory was visited by the author on a weekly basis during the two months July to August 2001, for the purposes of observing the quoting, ordering, manufacturing, and other administrative processes. Subsequent to that period, the factory was visited by the author whenever required, but in any case at least on a bi–monthly basis. The computing facilities of the Department of Applied Mathematics, as well as private facilities, were used during numerical simulations so as to obtain model solutions. Much of the data required were collected by the author on site at the factory, or provided by correspondence with Mr Taljaard. Mr Jannie Brandt, a programmer from *Netcommerce Consulting*, assisted with extraction of data from the Pastel database. Work on this thesis was completed in July 2004, and work emanating from this study was presented twice at annual conferences of the Operations Research Society of South Africa (ORSSA).

Definition of Symbols

A number of symbols will conform to the following convention:
$\mathcal{A}$ Symbol denoting a **set**. (Caligraphy capitals)
$\mathbf{A}$ Symbol denoting a **matrix**. (Boldfaced capitals)
$\underline{a}$ Symbol denoting a **vector**. (Underlined letters)

$\alpha^{(\beta)}$	Service level of board β
A_t	Total floor area at *Clickabox* factory [m^2]
A_s	Floor area of the storage space at *Clickabox* factory [m^2]
$A^{(\beta)}$	Area of board type β [m^2]
β	Fraction of demand that is backordered during a stockout period
$\mathcal{B}$	Set of indices of board types kept in inventory, $\lvert\mathcal{B}\rvert = b$ and $\mathcal{B} = \mathcal{B}_{AC} \cup \mathcal{B}_{DWB}$
$\mathcal{B}_{AC}$	Set of indices of board types of cardboard type AC kept in inventory, $\lvert\mathcal{B}_{AC}\rvert = 28$
$\mathcal{B}_{DWB}$	Set of indices of board types of cardboard type DWB kept in inventory, $\lvert\mathcal{B}_{DWB}\rvert = 18$
χ_j^n	The sum of n independent, identically distributed random variables with distribution $r_{j,z}^{\underline{v}_i}$ in class $j \in \mathcal{K}$
$\overline{c}$	Cost of capital rate [Rands per week]
$D^{(\beta)}$	Average annual demand for board type β [Sheets per annum]
$d_t^{\underline{v}_i}$	Demand class of board preference vector $\underline{v}_i$ in week t
$\Phi(\xi)$	The probability density function of random demand ξ
$\phi_{i,f}$	The wastage cost incurred when the f–th board in board preference vector $\underline{v}_i$ is used instead of the optimal board [Rands per board]
$f_n(x)$	Discounted expected cost for an n–period model under an optimal control policy with an on hand inventory level of x [Rands per week]
$F_t(f_t)(j,x)$	Expected holding cost in time period t, given demand state j and inventory level x [Rands per time period]
$\mathcal{G}$	Set of grid points (potential stock boards)
$g_{i,\beta}$	The wastage per board when one sheet of type $i \in S$ is cut out of a board of type $\beta \in \mathcal{B}$ [m^2]
${g'}_{i,\beta}$	The percentage wastage when one sheet of type $i \in S$ is cut out of a board of type $\beta \in \mathcal{B}$
$G_t^{(\beta)}(u_t^{(\beta)})$	The single week expected cost for board type β and inventory position u_t [Rands per week]

$\eta^{\varpi}_{(z)}$	Spatial constraint for cardboard type z of rank ϖ [Number of boards]
h'	The height to which boards may be stacked [m]
$h^{(\beta)}$	Holding cost per week of board type β [Rands per week]
h^T	Total holding cost per week over all board types [Rands per week]
$H_t^{(\beta)}(u_t^{(\beta)})$	The total single week cost for board type β and inventory position u_t, comprising the purchasing cost and the inventory cost $G_t(u_t \vert \pi_t, l)$ [Rands per week]
$\underline{I}_t^{\underline{v}_i}$	Vector of information available at the start of week t for board preference vector $\underline{v}_i$
$\kappa^{(\beta)}$	Number of order cycles in a year for board type β
K_t^j	The fixed order cost in week t and demand state j [Rands per order]
$\overline{K}_t^j$	The expected fixed order cost in week $t+1$ and demand state j [Rands per order]
$\mathcal{K}$	Set of indices of demand classes, $\vert\mathcal{K}\vert = 7$
$L(\overline{S}, x)$	Expected period cost for on hand inventory level x and order–up–to level $\overline{S}$
$L_{\mathcal{B}_f}$	The length of entry f in the set $\mathcal{B}$ [m]
$L_{\mathcal{G}_f}$	The length of entry f in the set $\mathcal{G}$ [m]
$L_{\mathcal{O}_f}$	The length of entry f in the set $\mathcal{O}$ [m]
l	Lead–time for delivery of raw materials [Number of weeks]
$m_t^{(i,\beta)}$	The maximum number of sheets, optimally produced by board preference vector $\underline{v}_i, i \in \mathcal{V}$, that can be produced from board type $\beta \in \mathcal{B}$ in week t [Sheets per board]
$\overline{m}^{(\beta,f)}$	The maximum number of sheets of type $f \in \mathcal{S}$ that can be produced from board type $\beta \in \mathcal{B}$ [Sheets per board]
$n_j(\underline{\sigma}^{\underline{v}_i})$	The number of times that state j occurs in the sequence $\underline{\sigma}^{\underline{v}_i}$
$\mathcal{O}^i$	Modified set of past orders (excluding those made by β_1 to β_i)
$\mathcal{O}$	Set indices of past orders
$\Pi_{j,t}^{\underline{v}_i}$	The probability of board preference vector $\underline{v}_i$ being in demand state j, given the information available up to week t
$\mathbf{P}^{\underline{v}_i}$	The transition probability matrix for board preference vector $\underline{v}_i$
$P_{j,k}^{\underline{v}_i}$	The probability of the demand state of board preference vector $\underline{v}_i$ changing from state j to state k
$p^{(\beta)}$	Purchasing cost per unit of board type β [Rands per board]
$q_t^{(\beta)}$	The quantity of board type β ordered in week t [Boards per week]
$\overline{q}^*$	Economic order quantity [Number of boards]
R_c	Rental cost per volume of stock per week [Rands per m^3 per week]
R_p	Total annual rent paid [Rands per annum]
R_s	Proportion of annual rent paid attributed to storage space [Rands per annum]
R_T	Total rent per period [Rands per week]
$\overline{r}$	Re–order level [Number of boards]
$\mathbf{r}^{\underline{v}_i}$	Probability distribution of demand for board preference vector $\underline{v}_i$
$r_{j,k}^{\underline{v}_i}$	Probability of a demand realisation in demand class k, given a current demand state of j distribution of demand for board preference vector $\underline{v}_i$
$\hat{r}_{j,k}^{\underline{v}_i}$	The probability distribution of the lead time demand of board preference vector $\underline{v}_i$

$\mathcal{S}$	Set of indices of possible sheet types for which orders may be received, $\lvert\mathcal{S}\rvert = s$
$\overline{S}$	Order–up–to or base–stock level [Number of boards]
$S^{*(\beta)}$	Optimal number of stockouts of board type β in a year [Stockouts per annum]
$\overline{S}^{(\beta)}$	Probability of a stockout of board type β in each order cycle
$\overline{s}$	Re–order level [Number of boards]
$s^{(\beta)}$	Cascading shortage cost per unit of board type β [Rands per board]
T	Point of time at which the myopic behaviour of the demand is terminated
$\mathcal{T}$	Set of indices of one–week time periods, $\lvert\mathcal{T}\rvert = 52$
τ	Expediting factor
θ	Discount factor
$u_t^{(\beta)}$	Inventory position of board type β at the start of week t [Number of boards]
$\upsilon_j^{\underline{v}_i}$	Mean of distribution j
$\upsilon_j^{(2)\underline{v}_i}$	Second moment about the mean of distribution j
$v^{(\beta)}$	Volume of board β [m^3]
$\mathcal{V}$	Set of indices of board preference vectors, $\lvert\mathcal{V}\rvert = \mu$
$\underline{v}_i$	Board preference vector i
$W_{\mathcal{B}_f}$	The width of entry f in the set $\mathcal{B}$ [m]
$W_{\mathcal{G}_f}$	The width of entry f in the set $\mathcal{G}$ [m]
$W_{\mathcal{O}_f}$	The width of entry f in the set $\mathcal{O}$ [m]
$w_t^{\underline{v}_i}$	Realised demand for board preference vector $\underline{v}_i$ in week t [Number of boards]
$\underline{W}_t^{\beta}$	Realised demand for board type β in week t [Number of boards]
$W_{t,i}^{\beta}$	The i–th level demand for board type β in week t [Number of boards]
$\hat{W}_t^{\beta}$	Realised demand for board type β during the lead time from week t [Number of boards]
$\Psi^{(\beta)}$	Shortage cost of board type β [Rands per week]
$\mathcal{X}$	Set of indices of boards in the board preference vector, $\lvert\mathcal{X}\rvert = 3$
$x_t^{(\beta)}$	Inventory level of board type β in week t [Number of boards]
$\zeta_t^{(i,k)}$	The set of all possible demand state sequences $\underline{\sigma}_t^{\underline{v}_i} = (d_t^{\underline{v}_i}, d_{t+1}^{\underline{v}_i}, \ldots d_{t+l}^{\underline{v}_i})$ such that $d_t^{\underline{v}_i} = k$

Glossary

Adaptive policy. A policy in which the information gained during each time period is used to update the estimates of unknown parameters, for use during the subsequent periods.

Backorder. A customer demand that has not been met due to a *stockout* situation, where the customer is prepared to wait for the *raw materials* to arrive in stock.

Base–stock level. The *inventory level* to which an inventory replenishment order should bring the stock on hand.

Base–stock Policy A *continuous review* inventory replenishment policy which comprises a single parameter, namely an *order–up–to* or *base–stock level*, $\overline{S}$.

Bill of Materials. A listing of *raw materials* required by a *manufacturer* to complete or produce a specified product.

Board. A piece of cardboard received as is from a *supplier*, not yet cut to the correct dimensions for the manufacturing of a cardboard box.

Board Preference Vector. A set of three *boards* from which a *sheet* order may be produced, listed in order of increasing *offcut wastage* incurred.

Certainty equivalent control. Inventory replenishment policies under which some data are observed, the unknown parameters are estimated by maximum likelihood methods, and inventory policies are chosen, assuming that the demand distribution parameters equal the estimated values.

Cost of capital rate. The cost of financing an investment, such as the interest paid on a loan.

Continuous review. An inventory control policy under which replenishment orders may be placed at any time.

Decision Support System. A computerised system designed to assist managers in selecting and evaluating courses of action, by providing a logical analysis of the relevant factors influencing decisions.

Economic Order Quantity. The optimal replenishment order quantity that minimizes the *holding* and *order costs* of *on hand inventory.*

Fill Rate. The percentage of inventory items demanded during a fixed time period that will be in stock when needed.

Holding cost. The cost per unit of holding stock in inventory. This comprises *rental cost*, insurance, and the *opportunity cost* of tied–up capital investment.

Inventory level. *On hand inventory* less *backorders.*

Inventory position. *On hand inventory* together with stock on order from *raw materials suppliers*, less *backorders.*

ISO9000. Certification Standards created by the International Organization for Standardizations in 1987 that now play a major role in setting process documentation standards for global manufacturers. These standards are recognized in over 100 countries. ISO9000 provides general requirements for various aspects of a firm's operations, including Purchasing, Design Controls, Contracts, Inspection, Calibration, *etc.* [2].

K–convexity. A condition used to prove the optimality of the $(\overline{s}, \overline{S})$ policy in the case of both fixed and variable *order costs.* A function is said to be K–convex if the secant line connecting any two points on the graph of the function, when extended to the right, is never more than K units above the function.

Lead Time. The time span between the placing of a replenishment order and its subsequent arrival into inventory.

Lost Sales. Potential sales that are lost, because there is no *stock* available in inventory (if the waiting time for delivery of an order is too long).

Manufacturer. A company involved in a series of interrelated activities and operations involving the design, material selection, planning, production, quality assurance and marketing of commercial goods.

Moving Average. The average over the last N points in a set of data is said to be an "N–period moving average." This is sometimes used to make forecasts, based on the most recent data.

Myopic Policy. A policy which does not take future costs as a result of current decisions into account.

Non–stationary Demand. Demand having a probability distribution that changes over time.

Obsolescence. *Stock* that is no longer usable for its intended purpose through expiration, contamination, damage, or change of need.

Offcut Wastage. The wastage incurred when *boards* are cut into *sheets* of the required dimension in order to produce cardboard boxes, resulting in pieces of cardboard too small for re–use.

On hand inventory. The *stock* immediately available in inventory.

Opportunity cost. Potential earnings from an alternative investment, such as the interest which would be earned on capital were it to be placed in a bank account, instead of being invested in inventory.

Order cost. The costs involved in placing an inventory replenishment order at a *supplier*.

Order cycle. The length of time between the placement of successive orders to replenish an inventory.

Order quantity. The number of *stock* items to be ordered when an inventory replenishment order is placed.

Order–up–to level The *on hand inventory* level up to which a replenishment order should bring the stock on hand in an $(\overline{s}, \overline{S})$*–model* or under a *base–stock policy.*

Pallet. A rectangular wooden based support for unitized lots of cardboard, subject to standards of length and width for storage in pre–determined places. Construction of the wooden base is such that there is air space between the bottom of the pallet and the load bearing surface of the pallet sufficient to allow the insertion of lifting forks so as to transport the pallet by forklift.

Partially Observed Demand. Demand for which the underlying distribution is not *completely observed* — it is only partially observed through the demand that has actually realised.

Periodic Review System. An inventory replenishment or control policy in which the order cycle is a fixed period of time.

Product Substitution. The use of a non–primary or sub–optimal product or component (at a cost), normally when the primary or optimal item is not available.

Purchasing cost. The cost per unit of *raw materials* purchased from a *supplier.*

$(Q, \overline{r})$ **Model.** An inventory replenishment or control policy in which an order of magnitude Q inventory units is placed if the *inventory level* reaches the *re–order level* $\overline{r}$.

Raw materials. Materials purchased by a *manufacturer* to be used in the manufacturing of products.

Recyclable Materials. Goods that may be collected for re–use as *raw materials* to manufacture new products.

Rental cost. The cost of hiring space used for warehouse and inventory control functions, including office space. For owned buildings a fair market rental value or depreciation is used instead.

Re–order level. The *inventory level* at or below which a purchase requisition is initiated. It is a combination of expected usage during the *lead time* period and a *safety stock* buffer.

Re–order Quantity. The number of *stock* items to be ordered when a *re–order level* is reached.

Safety stock. Inventory which serves to promote continuous supply when unpredictable demands exceed forecasts, or the delivery of *raw materials* from a *supplier* is delayed.

Scrap Material. Material that is deemed worthless to a production facility and is only valuable to the extent to which it can be recycled.

Service Level. The probability of not running short of *stock* before an inventory replenishment order arrives (*i.e.*, the percentage of *order cycles* during the year in which there were no *stockouts*).

Set–up Cost. The marginal cost of a machine or workstation setup. This generally includes the labour and the materials cost associated with the *scrap material* generated by the setup.

Sheet. A piece of *cardboard* that has been cut to the exact dimensions required for the manufacturing of a cardboard box order.

Shortage cost. The cost incurred when a sub–optimal inventory item must be used to produce an order, due to a *stockout* of the optimal inventory item.

Simulation. A representation of reality, often used for experimentation. In operations management, computer simulations of complex systems, such as factories or service processes, are often utilized in to order experiment with changes in management decisions. Experimenting with a simulation model may help to identify problems and opportunities for improvement, without having to actually build (or change) the physical system in order to evaluate the repercussions of such changes. Computer simulations of this type are generally discrete event simulation models, as opposed to continuous simulation models in which quantities have continuous measure.

$(\overline{s}, \overline{S})$ **Model.** An inventory replenishment or control policy in which an order is placed if the *inventory level* is less than or equal to some specified value, $\overline{s}$. The size of the order placed is sufficient to raise the *inventory level* to the *order–to level* $\overline{S}$.

Stationary Demand. Demand having a single probability distribution that does not change over time.

Stock. The commodity or commodities on hand in a storeroom or warehouse to support operations.

Stockout. The condition existing when a supply requisition cannot be filled from *stock*.

Supplier. A provider of *raw materials*.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Inventory is held by manufacturing companies for a number of reasons, such as to allow for flexible production schedules and to take advantage of economies of scale when ordering stock [55]. Most importantly, an inventory acts as a buffer between supply and demand, compensating for variations in demand and safeguarding against variations in delivery lead time of raw materials. Failure to meet demand compromises customer satisfaction, and may lead to high costs of emergency production [79]. The efficient management of inventory systems is therefore a crucial element in the operation of any production company [19]. Benefits from efficient inventory management to the customers include increased "off the shelf" availability of products, whilst benefits to the management include reduced tied–up investment capital in the inventory, reduced operating costs associated with warehousing functions, and a reduction in obsolescence accrual [35].

There are a number of factors that should be taken into account when developing an inventory model, such as the nature of the demand for the product, customer requirements and costs involved. In this thesis, an inventory control model will be developed for a manufacturing company where the production process is characterised by non–stationary, partially observed stochastic demand and a cascade of stock substitution during production.

The remainder of this chapter is structured as follows: *Clickabox*, the factory on which the case study for this thesis is based, is introduced in §1.1, and a brief and informal description of the research problem to be considered in this thesis is given in §1.2. Finally, an overview of the structure of the remaining chapters in the thesis is given in §1.3.

1.1 Introduction to *Clickabox* Factory

Clickabox is a cardboard box manufacturer in the South African Western Cape. It is a privately owned factory which caters for a niche in the local cardboard packaging industry characterised by short delivery times, and therefore availability of stock is a high priority for the company.

1.1.1 Products and Services

Clickabox has become an established manufacturer and supplier of cardboard boxes to industries in the Western Cape. Its products include cartons (printed with either one or two colours), creased boards, pads and sheets – all manufactured from corrugated cardboard bought in from a number of large manufacturers, such as *Mondi* [52], *Nampak* [56], and *Atlantic packaging* [5], who typically have long delivery response times for orders.

The company's focus is on the manufacturing of smaller order quantities and delivering with a shorter response time than its competitors. This is the perceived gap left in the market by the larger players mentioned above, and it is within this niche that *Clickabox* competes.

The activities of *Clickabox* are divided into two kinds of business: *long orders* and *quick orders*. Customers who are prepared to wait for up to ten working days for their orders to be met may place a long order. Boxes ordered as such are less expensive, as partially processed boards are bought in from the supplier (whose lead time is ten days) at no extra cost to *Clickabox* and cut to size, so that offcut wastage is minimal. The company's focus is, however, on the market for quick orders. For these orders, produced from the unprocessed boards kept in stock at the factory, *Clickabox* guarantees product delivery within two working days. This is significantly faster than deliveries made by larger companies, which may take up to three weeks to deliver an order to individuals. The closest competitor to *Clickabox*, in terms of delivery time, is *Cape Town Box* [14], which guarantees a four day lead time [71].

Boxes are made to order; even very small orders are accepted by *Clickabox*, and free delivery is offered on large orders. A major part of the service delivery of *Clickabox* is its ability to supply quick quotes to customers, and its ability to perform all the transactions accurately and quickly, in electronic fashion. The company website [57], from which it derives its name, forms a significant part of its service offering in this regard. Customers may register, request quotes, and view inventory and product information online. The website gives the company a competitive advantage in that it makes its products accessible to inexperienced clients.

1.1.2 The Industry as a Whole

The South African corrugated packaging industy is very competitive and has undergone a rationalisation phase during the period 1999–2001 [72]. Major players in the industry, like *Mondi* [52] and *Kohler* [41], have rationalised their capacity and have closed some of their production plants. Other companies, like *Corruboard*, *Smart Packaging* and *Naledi*, have closed down completely. With steeply rising input costs (due to price increases in the paper industry) and a slump in the fruit exports from the South African Western Cape during the period 2000–2001 [46], the capacity still exceeds the demand and there are very few new entrants in the market. This abundance of competition heightens the need for *Clickabox* to remain competitive in its pricing. Direct competitors include *Cape Town Box* [14], *Boxes for Africa* [13], and a host of smaller manufacturers. This cut–throat competitiveness necessitates streamlining of all processes if companies wish to survive.

1.2 Informal Problem Description

The goal of this study is to establish a good inventory management policy for *Clickabox* factory.

The service offering of *Clickabox*, namely delivery of quick orders within two working days, is dependent on the availability of suitable cardboard in stock. Suitability of a stock board is determined by the amount of off–cut wastage that results when it is used to meet an order, which, in turn, depends on the dimensions of orders for boxes. The objective of the model in this thesis will be to minimize stockholding costs, subject to the following constraints:

I Raw material off–cut wastage is to be limited to at most 15% of the surface area of a stock board used.

II A service level of 95% is to be achieved for all orders being met with stock boards in inventory.

The first deliverable of the study is a recommendation as to a suitable set of boards which should be kept in stock, in order to meet the above mentioned objective, subject to the constraints. The second deliverable is a computerised decision support system, which, given certain inputs (such as inventory composition and current inventory levels), will provide re–order and order–to levels for each of the boards kept in stock.

1.3 Thesis Overview

Apart from this introductory chapter, this thesis comprises a further six chapters. Chapter 2 opens with a very brief survey of the large body of inventory theory literature, and then examines in more detail the literature related to specific aspects of the inventory problem at *Clickabox*. This is followed, in Chapter 3, by an in–depth description of *Clickabox* factory, its production and ordering processes and its business objectives. Chapter 4 is devoted to an analysis of the demand for cardboard boxes, in order to determine which board sizes are ideal to keep in stock, and to derive demand distributions for these optimal board sizes. A theoretical inventory model is formulated in Chapter 5 and then a sub–optimal control policy is developed, which is more practical with regards to computational requirements. The verification and results of the computerised simulation model, built on the inventory model of Chapter 5, are discussed in Chapter 6. The computerised decision support system developed for use at *Clickabox* is based on this simulation model.

Chapter 2

Literature Review

There is a vast body of literature concerning inventory theory. This chapter is aimed at briefly tracing the development of inventory theory, in order to place the topic of this thesis in context, and then exploring, in some detail, specific concepts in inventory modelling. The relevance of each of these concepts to the situation at *Clickabox* factory is explained and utilised in the approach taken in the remainder of this study.

2.1 Brief Overview of Inventory Theory

The goal of inventory modelling is typically to find a policy (usually comprising elements such as re–order levels and re–order quantities) which minimises total inventory cost, subject to a given service level. Total inventory cost normally comprises ordering and setup costs, unit purchase costs, holding costs and shortage costs.

An early Operations Management philosophy, namely the theory of an *economic order quantity* (EOQ), where inventory costs are minimised for independent demand, was developed as early as 1913 [28]. Early works include those of Harris (1913, [30]), and Wilson (1934, [78]) on the classic economic lot size model, which recommends an optimal production batch size by a trade–off of the inventory holding cost against production change–over costs. This formed the basis of the EOQ model, still used widely today. The objective of the model is to minimize total cost, assuming continuous review, a known, constant demand, and a known, constant lead time. As shown in Figure 2.1, the minimum cost is incurred when the cost of holding stock is balanced with the cost of ordering stock. The EOQ is given by $\overline{q}^* = \sqrt{2KD/h}$, where K is the fixed ordering cost, D is the average annual demand, and h is the holding cost per unit per year. The re–order point is given by the demand per period multiplied by the lead time (in number of periods). The reader is referred to Hadley and Whitin (1963, [29]), Johnson and Montgomery (1974, [34]), and Winston (1994, [80]) for discussions on the EOQ model and its applications.

The largest body of literature, however, stems from the post World War II period. A classic work is that of Arrow, *et al.* (1951, [3]). They derived an optimal inventory policy for problems in which demand is known and constant, and then for single period problems in which demand is random, with a known probability distribution. They also analysed

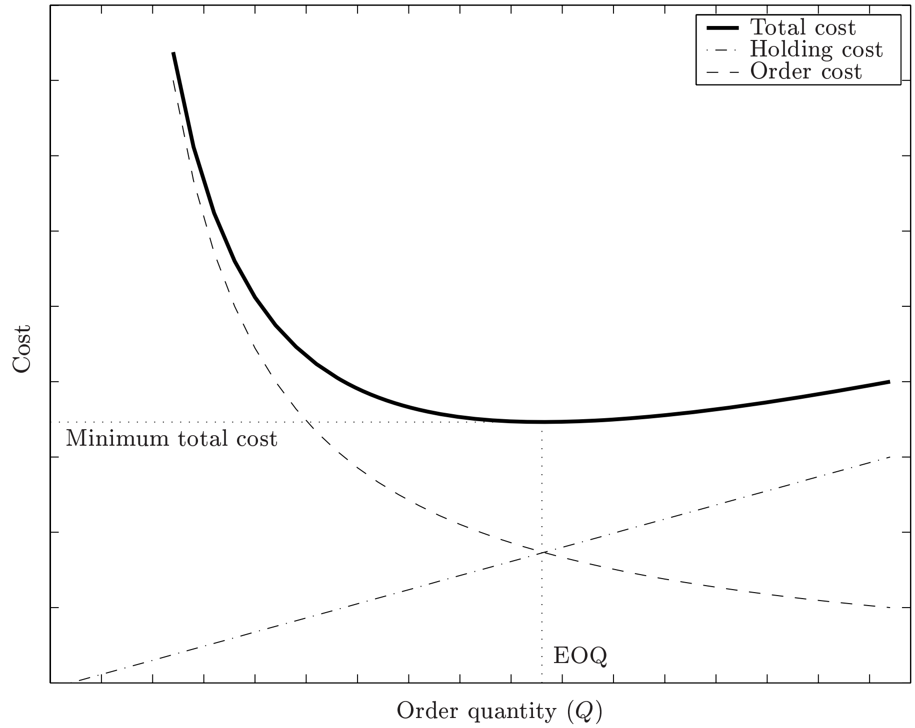


Figure 2.1: *A graphical depiction of the concept of an economic order quantity, which is the order quantity at which the holding and ordering costs are balanced, in order to minimise total cost.*

the general dynamic problem, under the assumption of a fixed setup cost and a unit order cost, proportional to the order size. Under these assumptions the optimal inventory policy was suspected to be an $(\overline{s}, \overline{S})$ policy, in which an order is placed if the inventory level is less than or equal to some specified value, $\overline{s}$, at the beginning of a period. The size of the order placed is sufficient to raise the inventory level to $\overline{S}$. The optimality of this policy was proven in later years for various cases (see Scarf (1960, [64]) and Bensousson, *et al.* (1983, [11])). The $(\overline{s}, \overline{S})$ inventory policy is illustrated schematically in Figure 2.2.

A number of important advances were contained in the monograph of Arrow, *et al.* (1958, [4]), which provided a foundation for future work in inventory theory. A paper by Karlin and Scarf (1958, [39]), appearing in this monograph, investigated delivery lags, *i.e.* cases in which there is a positive lead time from the supplier, with two major results. The first result was for the case of *backordered* sales, where in a stockout situation the customer is prepared to wait for his order to be delivered. The optimal re–ordering policy was shown to be a function of the inventory position (the sum of the stock on hand and the stock on order less backordered stock). The second result was for the case of *lost* sales, where in a stockout situation the customer is not prepared to wait and the sale is lost. It was shown that the simple policy that is optimal for the case of backordering is not optimal for the lost sales case. A detailed study of optimal policies was presented for the case of lost sales, delivery lags and a linear purchasing cost. The analysis was conducted by means of the standard dynamic programming formulation of the inventory problem. For an on hand inventory of x units, an initial order–up–to quantity of $\overline{S}$ units, a linear unit

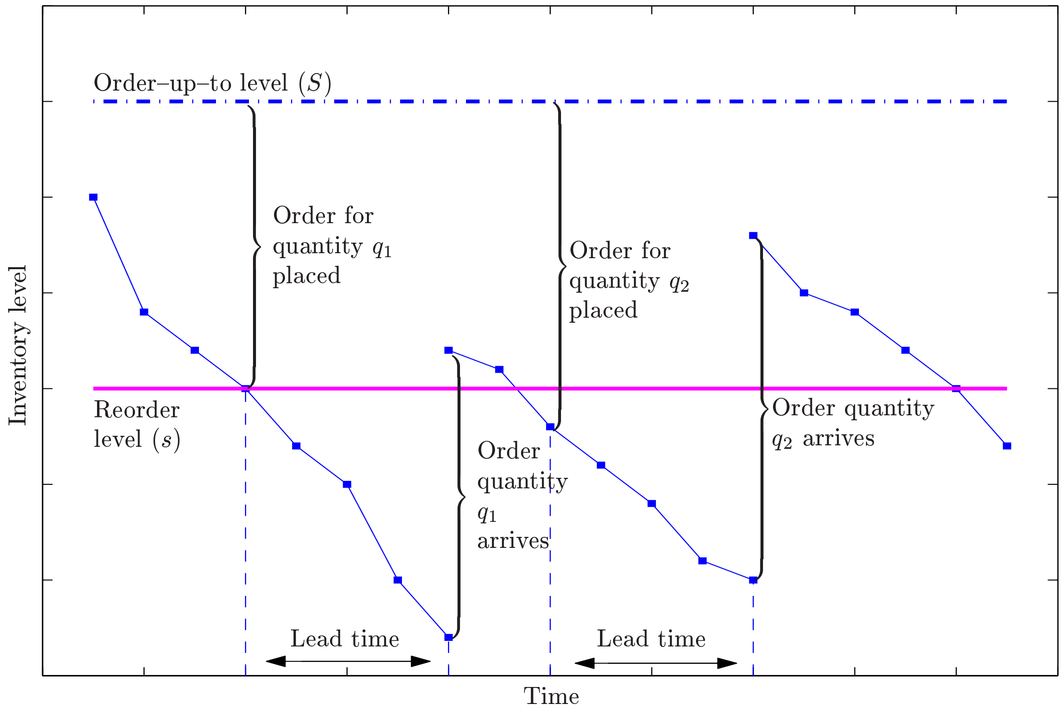


Figure 2.2: *A schematic illustration of stock movement over time under the* $(\overline{s}, \overline{S})$ *inventory policy. An order is placed, when the inventory level drops below* $\overline{s}$, *for a quantity sufficient to raise the inventory level to* $\overline{S}$. *The inventory level continues to decrease during the lead time period, until the arrival of the order. The cycle then repeats itself.*

order cost of p monetary units, a single period delivery lag and a random demand of ξ units per period governed by a probability distribution with density $\Phi(\xi)$, the problem was formulated as follows: Let $L(\overline{S}, x)$ be the expected period cost (holding and shortage costs) and ϑ the discount factor. For the case of lost sales, the stock level will not become negative, so the new inventory level, at the end of the period, is given by $\max\left\{\overline{S} - \xi, 0\right\}$, *i.e.* $x = \overline{S} - \xi$ if $\xi \in \left\{0, \ldots, \overline{S}\right\}$, and $x = 0$ if $\xi \geq \overline{S}$. Then $f(x)$, the discounted expected costs associated with an optimal decision, satisfies the dynamic programming recursion

$$f(x) = \min_{\overline{S} \geq x} \left\{ p(\overline{S} - x) + L(\overline{S}, x) + \vartheta \left[\int_0^{\overline{S}} f(\overline{S} - \xi)\Phi(\xi)\ \mathrm{d}\xi + f(0) \int_{\overline{S}}^{\infty} \Phi(\xi)\ \mathrm{d}\xi \right] \right\}. \tag{2.1}$$

If delivery is instantaneous, the expected cost becomes $L(\overline{S})$, a function of the immediately available inventory only. For the case of backlogging, the stock level can become negative, and so the new stock level is given simply by $x = \overline{S} - \xi$. Equation (2.1) is then reduced to

$$f(x) = \min_{\overline{S} \geq x} \left\{ p(\overline{S} - x) + L(\overline{S}) + \vartheta \int_0^{\infty} f(\overline{S} - \xi)\Phi(\xi)\ \mathrm{d}\xi \right\}. \tag{2.2}$$

An important result in inventory theory has been the establishment of the optimality of $(\overline{s}, \overline{S})$ policies under various conditions. Scarf (1960, [64]) introduced the notion of

K–convexity, a condition he used to prove the optimality of an $(\overline{s}, \overline{S})$ policy in the case of both fixed and variable ordering costs K. A function $f(x)$ is said to be K–convex if the secant line connecting any two points on the graph of the function, when extended to the right, is never more than K units above the function, or in algebraic terms, if

$$f(x) + a\left[\frac{f(x) - f(x-b)}{b}\right] \leq f(x+a) + K, \tag{2.3}$$

for $a, b > 0$ and all x.

Scarf [64] defined $f_n(x)$ to be the cost function associated with optimal decisions for an n–period inventory problem. He showed that the function $ps + L(s) + \vartheta \int_0^\infty f_{n-1}(s-\xi)\Phi(\xi)\mathrm{d}\xi$ is K–convex, and proved, by induction, the optimality of the $(\overline{s}, \overline{S})$ policy in the case where backlogging is allowed. The only constraint on the parameters of the problem is that if the setup costs vary over time, they must decrease with increasing time.

Iglehart (1963, [31]) investigated the limiting behaviour of the value function,

$$f_n(x) = \min_{s \geq x}\left\{p(s-x) + L(s) + \vartheta \int_0^\infty f_{n-1}(s-\xi)\Phi(\xi)\ \mathrm{d}\xi\right\}, \tag{2.4}$$

and the optimal policies, (s_t, S_t), as $t \to \infty$, when the discount factor is $\vartheta = 1$.

The nature of the demand process is an important factor that affects the type of optimal policy in a stochastic inventory model. The inventory control literature that followed after 1960 is categorized according to the type of demand studied, *i.e. stationary* or *non–stationary*, and whether the demand is *fully* or *partially observed*. In a stationary demand process, the demand follows a single probability distribution, whilst in a non–stationary process the probability distribution of the demand varies with time. A fully observed process is one for which all parameters are known with certainty. A problem with partial information is one in which the demand distribution possesses one or more unknown parameters that may be either discrete or continuous. In the case of partial information, the estimate of the unknown parameters is usually updated as the actual demand is observed over time.

2.1.1 Stationary Demand, Fully Observed

The inventory problem with fully observed and stationary demand forms the body of most of the classical theory of inventory control. As discussed above, Scarf (1960, [64]) presented the result that if the demands during successive periods are independent and identically distributed random variables, and the demand is fully observed, an (s, S) policy is optimal.

2.1.2 Non–stationary Demand, Fully Observed

Inventory problems with fully observed, non–stationary demand, although more complex, have also been studied extensively. The dynamic inventory model of Arrow, *et al.* (1951,

[3]) mentioned above was extended by Karlin (1960, [38]), who presented an infinite horizon, multi–period inventory model with stochastic non–stationary demands. He assumed that production decisions are made during each period and, since the time horizon is infinite, disposal is not an issue. He established the optimality of the *base–stock* policy, and showed that if demand increases stochastically over time, the optimal base–stock values also increase. The base–stock policy is a continuous review, one–for–one replenishment policy which comprises a single parameter, namely an order–up–to level $\overline{S}$. An order for $q_t = \overline{S} - u_t$ units is placed at the start of period t to arrive for use at the start of period $t + l$, assuming an inventory position u_t at the start of period t, a delivery lag l and a base–stock level $\overline{S}$. A typical change in stock levels over time under a base–stock policy is shown in Figure 2.3. A base–stock policy is suitable when the cost of ordering is negligible and there is no penalty for small orders, and is inexpensive to implement in cases when storage allocations cannot be changed without incurring significant costs [21].

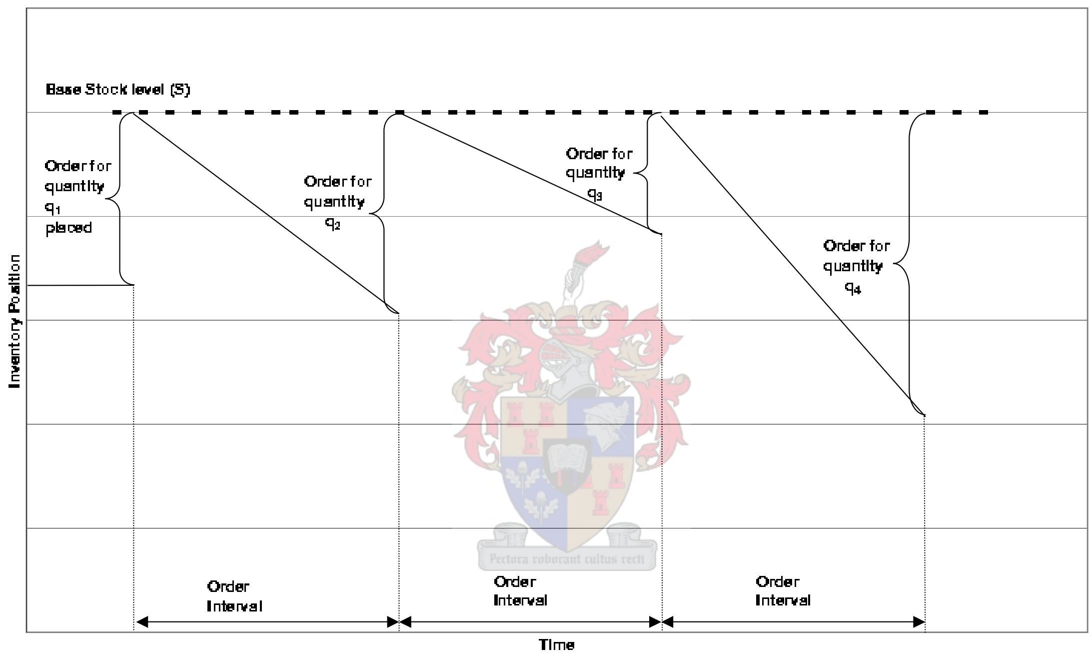


Figure 2.3: *A schematic illustration of stock movement over time under the base–stock inventory policy, showing the placement of an order during every period that is sufficient to bring the inventory position up to the base–stock level $\overline{S}$. This order arrives the lead time number of periods later.*

Veinott (1963, [76]) extended the work of Karlin [38] by generalizing his results on the ordering of base–stock levels. Bensousson, *et al.* (1983, [11]) formulated the problem of non–stationary, stochastically independent demands and proved the optimality of an $(\overline{s}, \overline{S})$ policy for both finite and infinite horizons. Song and Zipkin (1993, [66]) presented another single–item, continuous review model with non–stationary demand. They assumed that the demand follows a doubly stochastic Poisson process, with the rate governed by a Markov process. The demand state of the Markov process represents randomly changing environmental factors, such as fluctuating economic conditions and seasonal variability. Demand for each period is a random variable whose distribution function is dependent on the demand state (generated by the Markov process) in that period. They formulated a dynamic program to compute an optimal policy, using *modified value iteration*. Modified value iteration is an algorithm for finding optimal policies for partially observed Markov decision processes, aimed at accelerating the iteration process by reducing the number of dynamic programming updates to its convergence. Value iteration starts with an initial value function, which represents the expected total discounted reward for following a certain policy, and iteratively performs dynamic programming updates to generate a sequence of value functions, which converges to the optimal value function [83]. This typically requires a large number of dynamic programming updates, so the strategy followed in modified value iteration is to improve the value functions by means of certain additional steps. After a value function is improved by a dynamic programming update, it is fed to the additional steps for improvements.

Sethi and Cheng (1997, [65]) presented a more general model than that of Song and Zipkin [66], including the case of seasonal demand and incorporating service level and storage constraints. They used the concept of K–convexity, first utilized by Scarf (1960, [64]), to establish the optimality of the state–dependent $(\overline{s}, \overline{S})$ policy in the case of Markovian demand and full backlogging. The first assumption they made, under which the optimal policy is an $(\overline{s}, \overline{S})$ policy, was

$$K_t^j \geq \overline{K}_{t+1}^j \equiv \sum_{j=1}^{n} P_{jk} K_{t+1}^j \geq 0, \quad t \in \mathcal{T},\ j,k \in \mathcal{K}, \tag{2.5}$$

where K_t^j is the fixed ordering cost for period t and demand state j, $\overline{K}_{t+1}^j$ is the expected fixed ordering cost for period $t+1$ and demand state j, $\mathcal{K} = \{1, 2, \ldots, n\}$ is the finite set of possible demand states, $\mathcal{T} = \left\{0, 1, \ldots, \hat{t}\right\}$ is the planning horizon of the problem, and P_{jk} are the transition probabilities of the Markov process, in other words the probability of the demand state changing from state j to state k, for all $j, k \in \mathcal{K}$. This assumption states that the fixed cost of ordering during a given period with demand state j should be no less than the expected fixed cost of ordering during the next period. The second assumption made was that

$$p_t^j x + F_{t+1}(f_{t+1})(j, x) \to \infty \quad \text{as } x \to \infty, \quad t \in \mathcal{T}, \quad j \in \mathcal{K}, \tag{2.6}$$

where p_t^j is the purchasing cost for period t and demand state j, x is the surplus level (of inventory or backlog) at the beginning of a period and $F_{t+1}(f_{t+1})(j, x)$ is the expected holding cost. This assumption states that either $p_t^j > 0$ or $F_{t+1}(f_{t+1})(j, x) \to \infty$ as $|x| \to \infty$, or both. This generalises the usual assumption that the inventory carrying cost, h, is positive. Under these and other standard assumptions, they proved that there exist sequences of numbers s_t^j, S_t^j for all $t \in \mathcal{T},\ j \in \mathcal{K}$, with $s_t^j \leq S_t^j$, such that the order quantity under an optimal policy is given by

$$q_t(j, x) = (S_t^j - x)\Gamma(s_t^j - x), \tag{2.7}$$

where the step function $\Gamma(\bullet)$ is defined as $\Gamma(z) = 0$ when $z \leq 0$, and $\Gamma(z) = 1$ when $z > 0$.

Graves (1999, [27]) presented a model for a single–item inventory system with a deterministic lead–time, but subject to a stochastic, non–stationary demand process, in which the demand process behaves like a random walk. The demand process is an integrated moving average process, for which an exponential–weighted moving average provides the optimal forecast. An *adaptive* base–stock policy, in which the information gained during each period is used to update the estimates of an unknown parameter (the optimal order–up–to quantity) was proposed for inventory replenishment. It was observed that the safety stock required for the case of non–stationary demand is much greater than for stationary demand; furthermore, the relationship between safety stock and the replenishment lead–time becomes convex when the demand process is non–stationary, quite unlike the case of stationary demand.

Kambhamettu (2000, [36]) investigated the problem of parameter estimation when the observed demand is generated by discrete parametric probability distributions. Demand was modelled as a partially observed Markov decision process, and it was assumed that the demand states are characterised by either the Poisson or the Negative Binomial distribution. The *estimation maximization algorithm*, developed by Baum, *et al.* (1970, [9]), was applied to estimate the parameters, and the model with the best estimates was selected.

2.1.3 Stationary Demand, Partially Observed

Stationary, partial information problems are more difficult to solve than cases where demand is fully observed. Scarf (1959 [63], 1960 [64]) was a pioneer in the use of Bayesian techniques for inventory control. Bayesian analysis is a statistical procedure which endeavours to estimate the parameters of an underlying distribution, based on the observed data. It begins with a *prior distribution*, which may be based on anything, including an assessment of the relative likelihoods of parameters or the results of non–Bayesian observations. In practice, it is common to assume a uniform distribution over an appropriate range of values for the prior distribution. The Bayesian method allows for updating of the demand distribution as new data become available, while avoiding storage of all historical data. It provides a rigorous framework for dynamic demand updating when the demand distribution is not known with certainty.

Scarf (1959, [63]) studied a conventional dynamic inventory problem in which the purchase cost is strictly proportional to the quantity purchased, so that the optimal policy is defined in period t by a single critical number $\overline{S}$, the order–up–to level. The innovation in the paper was to allow the density of demand $\Phi(\xi, \omega)$, where ξ represents the demand, to depend on an unknown statistical parameter, ω, which describes the demand density. As time evolves, the sequence of realized demands generates holding, shortage, and purchase costs, but in addition, more is learnt about the true value of the underlying parameter. For the analysis to be manageable, the demand distribution is assumed to take the form $\Phi(\xi, \omega) = \beta(\omega) e^{-\xi\omega} \overline{r}(\xi)$, where β and $\overline{r}$ are functions of the unknown parameter and the demand respectively. With this specification, if the t–th period is entered with a knowledge of the current stock level, x, and a history of past demands, $\xi_1, \ldots, \xi_{t-1}$, the entire history may be summarized in the sufficient statistic,

$$\nu = \frac{\sum_{i=1}^{t-1} \xi_i}{t-1}, \tag{2.8}$$

so that the dynamic programming formulation depends only on the variables x and ν. The monotonicity of $S_t(\nu)$ was demonstrated in (1959, [63]), and the asymptotic behaviour of $S_t(\nu)$ and ν, as $t \to \infty$, was determined.

Karlin (1960, [38]), Scarf (1960, [64]) and Iglehart (1964, [32]) studied dynamic inventory policy updating when the demand density has unknown parameters and is a member of the exponential or range families. They showed that an adaptive critical value (or order–up–to) policy is optimal, where the critical value is determined dynamically.

Conrad (1976, [18]) examined the effect of *demand censoring* by the inventory level on Poisson demand estimation. Demand censoring occurs when there are lost sales and no backordering, resulting in partially observed demand. He proposed an unbiased maximum likelihood estimate of the Poisson parameter.

Azoury (1985, [7]) and Miller (1986, [51]) generalised and extended the results of Karlin [38], Scarf [64] and Iglehart [32] to other classes of demand distributions. Azoury (1984, [6]) also investigated the effect of dynamic Bayesian demand updating on optimal order quantities. She concluded that Bayesian demand updating, compared to the non–Bayesian method, yields a more flexible optimal policy by allowing updates of the order quantities in future periods. The Bayesian approach is generally difficult to implement, because of extensive computational demands. Lovejoy (1990, [47]) showed that a simple inventory policy based on a critical value (using a *myopic* parameter adaptive technique) may be optimal or near–optimal in some inventory models. A myopic policy is a replenishment policy that minimizes the average total cost per product until the inventory is depleted, ignoring the influence of future costs on the current decision. He also gave two numerical examples to illustrate the performance of simple myopic policies. Lovejoy (1992, [48]) further extended this analysis of myopic policies by considering policies that terminate the myopic behaviour at some point in time, which may be fixed or may be random.

Lariviere and Porteus (1995, [43]) considered Bayesian techniques for a lost sales problem in which sales, not true demand, are observed. Because of lost sales, the retailer may learn more about the true demand process by holding inventory at a higher level initially to establish quickly a better estimate of the true demand. Gallego, *et al.* (1996, [26]) demonstrated a so–called *Min–Max* technique for the analysis of various distribution–free finite horizon models for which the distribution is specified by a limited number of parameters, such as the mean and variance, or a set of percentiles of demand. This Min–Max technique is a linear programming approach, where the objective is to minimize the maximum expected cost over all demand distributions, satisfying a set of linear constraints.

Ding and Puterman (1998, [20]) investigated the effect of demand censoring on the optimal policy in newsvendor inventory models with general parametric demand distributions and unknown parameter values. The main result of the paper is that the combined effect of an unknown demand distribution and unobservable lost sales results in higher optimal order quantities than in the fully observable demand case. This illustrates the trade–off between information and optimality, in the sense that it is optimal to set the inventory level higher during earlier periods in order to obtain additional information about the demand distribution, so as to allow for better decisions during later periods.

2.1.4 Non–stationary Demand, Partially Observed

Considerably less work has been done on the more complex inventory problems with non–stationary demand and partial information. One paper that considers a problem in this class is by Kurawarwala and Matsuo (1996, [42]). They presented a growth model to estimate the parameters of a demand process over its entire life cycle. In their base case, production decisions are made at the beginning of the problem for the entire life cycle. They presented a technique with which the initial estimation of parameters of their forecasting model is made. However, they do not thoroughly address the issue of revising these estimates, using new observations.

Treharne and Sox (2002, [73]) examined several different policies for an inventory control problem in which the demand process is non–stationary and partially observed. The probability distribution for the demand during each period is determined by the state of a Markov chain; this underlying distribution is called the core process. However, the state of this core process is not directly observed; only the actual demand (defined as w_t) is observed by the decision maker. The inventory control problem is a composite–state, partially observed Markov decision process. For an inventory position u_t, the single–period expected inventory cost function is defined as

$$G_t(u_t|\pi_t, l) = \mathrm{E}_{\hat{w}_{t,l}|\pi_t}\left[\Psi \max\left\{0, \hat{w}_{t,l} - u_t\right\} + h \max\left\{0, u_t - \hat{w}_{t,l}\right\}\right], \tag{2.9}$$

where $\hat{w}_{t,l} = \sum_{n=0}^{l} w_{t+n,l}$ represents the observed demand during the lead time l, π_t is a matrix characterizing the current belief of the demand distribution, Ψ represents the unit shortage cost, h represents the unit holding cost, p represents the unit purchase cost, and $\mathrm{E}_{\hat{w}_{t,l}|\pi_t}[\bullet]$ denotes the expected value operator, which gives the expected value of the cost function based on the calculation of lead time demand, which is conditional on the current belief of the demand distribution. The transition equation, by which the most recent observation, w_t, is used to update π_{t+1}, is

$$\pi_{k,t+1} = T_k(\pi_t|w_t = z) = \frac{\sum_{j=1}^{N} \pi_{j,t} r_{j,z} P_{j,k}}{\sum_{j=1}^{N} \pi_{j,t} r_{j,z}}, \tag{2.10}$$

where $r_{j,z}$ is the probability that the observed demand is z, given a demand state j, and $P_{j,k}$ is the transition probability of the Markov decision process between states j and k.

The order quantity in period t was defined as q_t, and the dynamic programming recursion

$$J_t(u_t, \pi_t) = -pu_t + \min_{y_t \geq u_t}\left\{py_t + G_t(y_t|\pi_t, l) + \mathrm{E}_{w_t|\pi_t}\left[J_{t+1}((y_t - w_t), T(\pi_t|w_t))\right]\right\}, \tag{2.11}$$

where $y_t = u_t + q_t$, was then shown to be convex for all u_t. This demonstrates that, in the absence of fixed ordering costs, a state–dependent base–stock policy is optimal for the problem. In the case of a positive fixed order cost, Treharne and Sox [74] proved that the optimal policy is a state–dependent $(\overline{s}, \overline{S})$ policy.

Composite–state, partially observed Markov decision process problems are in practice often solved by means of *certainty equivalent control* (CEC) policies. Under these policies some data are observed, the unknown parameters are estimated by maximum likelihood methods, and inventory policies are chosen, assuming that the demand distribution parameters equal the estimated values. However, Treharne and Sox presented results that demonstrate that there are other practical control policies that almost always provide much better solutions to this problem than the CEC policies commonly used [73]. The policies they compared were the myopic, limited look–ahead (LLA), open–loop feedback control (OLFC), and the CEC policies. Under the OLFC policy it is assumed that feedback will not be used in future periods, in other words the policy does not anticipate the use of future information about the prior distribution. It is implemented on a *rolling–horizon* basis, where π_t is updated using the prior observations. The LLA policy optimises the dynamic problem for only a limited number of periods into the future. For the myopic LLA policy, only the current period costs are minimised, in other words (2.11) is reduced to

$$J_t(u_t, \pi_t) = -pu_t + \min_{y_t \geq u_t} \{py_t + G_t(y_t|\pi_t, l)\}. \tag{2.12}$$

The computational results in [73] also indicate how specific problem characteristics influence the performance of each of the alternative policies.

2.2 Important Concepts in Inventory Modelling

In this section, an outline is given of literature related to various concepts in inventory modelling which are relevant to the case study presented in this thesis. One such concept, the *nature of the demand process*, has already been dealt with in some detail in the previous section. The nature of the demand at *Clickabox* is non–stationary, partially observed demand. This places it in the class of problems dealt with in §2.1.4. The transition between demand states at *Clickabox* will be modelled as a Markov decision process. A number of papers concerned with Markovian–modulated demand have been mentioned in §2.1; these and other related papers will be discussed in some detail. Another distinctive aspect of the inventory at *Clickabox* is its cascading *product substitution*. Other concepts to be discussed in this section are *lead time*, *stockout situations*, and *advance demand information*.

2.2.1 Modelling of a Markovian Decision Process

The modelling of Markovian–modulated demand was discussed in §2.1.2, in particular with reference to the study by Sethi and Cheng (1997, [65]). The use of a Markov decision process in the case of non–stationary, partially observed demand was introduced in §2.1.4. The methodology behind the adaptive inventory control policy proposed by Treharne and Sox (2002, [73]) and summarised in §2.1.4 will be followed in this thesis, with a number of adaptions, for the development of an inventory model for *Clickabox*. However, a significant adaption involves the definition of demand states. The approach taken by Treharne and Sox [73] is that the demand states, determined by the state of a Markov chain, represent the underlying demand distribution during that time period. Demand classes are defined, to represent ranges of values. The demand distributions then represent the probability of a time period demand realisation in each of the demand classes. In other words, if the underlying demand distribution is weighted toward large demand realisations (a distribution which would, for example, be in effect during peak seasons), it is probable that the time period demand realisation will be for one of the higher demand classes. It would, however, still be possible that the demand realisation falls into one of the lower demand classes. This approach was attempted for the *Clickabox* application. However, as will be discussed in Chapter 4, the nature of the demand suggested a modification. The demand states and demand classes are merged in this thesis, to form just one set of demand classes, each representing a range of potential demand realisations. The transition between demand classes is determined by the state of the Markov chain, and the actual demand realisation within the class is modelled by a set of probability distributions, one for each demand class.

2.2.2 Multiple Products and Stock Substitution

Veinott (1965, [77]) presented the earliest work on an optimal policy for a multi–product inventory model. This study was generalised by Ignall and Veinott (1969, [33]). They gave conditions under which the optimal order quantity is a monotone function of the initial inventory. Bassok, *et al.* (1999, [8]) followed the approach of Veinott [77] and Ignall and Veinott [33], and developed a single period, periodic review model with stochastic demand and downward substitution. They considered N products and N demand classes with full downward substitution, where excess demand for class i can be substituted, using product j, for $i \geq j$. They showed that a greedy allocation policy, for the allocation of products to demand classes, is optimal. The algorithm works sequentially from product 1 to product N. Demand for class i is satisfied first with stock of product i, and then if necessary, leftover stock of product $i-1, i-2, \ldots, 1$ is used to satisfy remaining unmet demand of class i. A similar approach is taken in the inventory model for *Clickabox*. However, the situation at *Clickabox* is more complex in the sense that strict downward substitution is not appropriate. This is a result of the directional property of cardboard used in the manufacturing of boxes — each board has two properties, namely length and width, both of which affect the board's structural suitability as a substitute, as a substitute board must have both dimensions at least as large as the board for which it is a substitute. The length and width of the board are independent of each other, in other words if board A has a length greater than that of board B, it will not necessarily have a width greater than that of board B. Boards cannot, therefore, be arranged in a simple list where product i is smaller than, and therefore can be substituted by, product $i+1$.

Drezner, *et al.* (1995, [23]) considered the substitution problem for a two–product inventory, and established optimal order and substitution quantities, using the standard EOQ modelling approach. Drezner, *et al.* (2000, [22]) formulated the problem for an n–product inventory, where product j can substitute products $j+1, \ldots, n$, at certain costs. They used the same approach as Drezner, *et al.* [23], but found that the total cost function may not be convex if the number of products exceeds two. They then reformulated the problem to determine analytically the optimal run–out time for the n products. They formed a new cost function, which was shown to be convex, and found the optimal decision parameters (order and substitution quantities) by backward substitution. However, in both [23] and [22], demand was assumed to be known and deterministic.

Downs, *et al.* (2001, [21]) developed a base–stock inventory model which incorporates multiple products, multiple resource constraints, lost sales and delivery lags. Nonparametric estimates of the expected holding and shortage costs were developed, and a mathematical programming approach was taken in determining a near–optimal policy, which compared favourably with the optimal solution in some numerical experiments. However, a critical assumption of the model developed is that the cost of placing orders is negligible, which is not appropriate in the case of *Clickabox*, as there is a minimum order quantity which results in a high penalty cost if orders for small quantities are placed frequently.

2.2.3 Stockout Situations

There are two general situations for the demand process in a stockout period — either the demand is backordered, in other words customers are prepared to wait until their demand is satisfied, or sales are lost, in other words customers are not prepared to wait, and consequently go elsewhere to satisfy their needs. The former is more widely covered in the literature, partly because inventory studies have historic roots in military applications, where the assumption of backlogging is realistic [15]. Montgomery, *et al.* (1973, [53]) investigated a more realistic case — a mixture of lost sales and backorders. They considered first a deterministic demand model, and then a model with stochastic demand for a particular case with a time–independent backorder cost and neglecting the stockout period in computing the expected order cycle length. Kim and Park (1985, [40]) extended their work to allow for time–weighted backorders. This is implemented in a $(Q, \overline{r})$ inventory model, where a fraction β of demand is backordered during a stockout period, and the remaining fraction of $1 - \beta$ is lost. The $(Q, \overline{r})$ policy states that an order of a fixed quantity Q should be placed when the inventory level drops to the re–order point $\overline{r}$. This policy is illustrated in Figure 2.4.

The $(Q, \overline{r})$ policy is applicable to situations in which demand can only be for a single unit at a time, as the danger exists that a demand for more than one unit could reduce the inventory level to below the re–order point $\overline{r}$, in which case the computations that led to the calculation of the optimal re–order quantity Q are invalidated. On the other hand, the $(\overline{s}, \overline{S})$ inventory model is typically used for situations in which demand for more than one unit at a time occurs. The model in [40] was developed, based on a heuristic treatment of a lot–size re–order–point policy. The situation at *Clickabox* is in essence different to the two general situations of backordering or lost sales, and to those in [40] and [53], in the sense that product substitution takes place. If the optimal board to satisfy an order is not available, the next best board is substituted, at a higher cost to the customer. The 'lost sale' of the optimal board is penalised in the wastage cost incurred when board substitution takes place. In the case of no board being available to satisfy an order, it will be assumed that demand is backordered.

2.2.4 Variability of Lead Time

The majority of inventory literature assumes lead time to be a constant. Bookbinder and Çakanyildirim (1999, [12]) considered a $(Q, \overline{r})$ model with constant demand, random

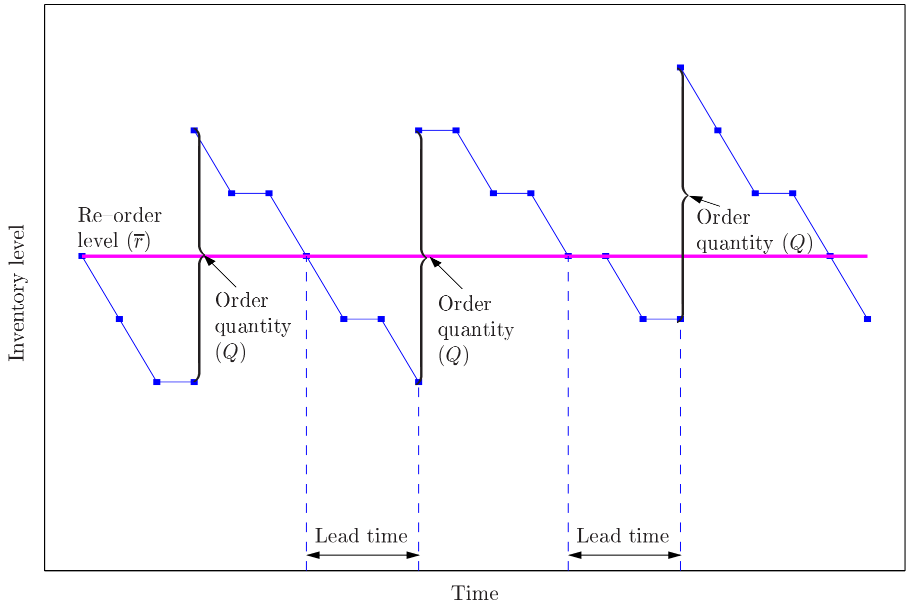


Figure 2.4: *Schematic illustration of the stock movement over time under the* $(Q, \overline{r})$ *inventory policy. An order for* Q *units is placed, to arrive the lead time number of periods later, each time the inventory level reaches a level of* $\overline{r}$.

lead time, and expedited orders. For expedited orders, a shorter–than–average lead time may be obtained at a cost. The expediting factor τ is the constant of proportionality between the expedited lead time and the ordinary lead time. The expected cost function was shown to be jointly convex in the decision variables $Q, \overline{r}$ and τ. The dynamics of lead time variability could be incorporated into an inventory model for *Clickabox*. However, due to the already complex nature of the problem considered in this thesis, it will be assumed that lead time is constant, as is standard practice.

2.2.5 Advance Demand Information

Karaesmen, *et al.* (2002, [37]) investigated the value of advance demand information in improving the performance of inventory systems. They discussed two cases with differing delivery requirements. Firstly, a model was considered in which each customer announces a future date for delivery and requires a timely delivery (*i.e.*, the delivery may not be early or late). The second model assumes that the customer submits requirements in advance, and accepts early deliveries. Karaesmen, *et al.* [37] identified conditions in each case under which advance demand information may bring significant benefits. One important factor was found to be the average system load — the relative benefits of advance demand information disappear in extremely high system loads. Another factor is the nature of the delivery requirements — if customers order in advance and accept early deliveries, there is a significant benefit to the manufacturer. However, if early deliveries are not accepted, the value of advance demand information is not as significant. This is a topic for potential further study at *Clickabox*, in the sense that if the value of advance demand information can be determined, rewards may be offered to regular customers for submitting orders in advance, or in a more regular fashion. This topic, however, falls outside the scope of this thesis.

2.3 Chapter Summary

A brief overview of literature concerning inventory theory was given in this chapter. The inventory control literature considered was broadly divided into categories according to the nature of the demand process, and work done in each of these categories was discussed. A number of concepts in inventory theory considered especially relevant to this study were then explored, and their applications in the inventory model to be developed in this thesis were discussed. The non–stationary nature of the demand process at *Clickabox* suggests the use of a Markovian decision process for the modelling of the demand, following the approach in a number of works cited in §2.2.1. A core characteristic of the situation at *Clickabox* is the existence of multiple products and cascading stock substitution — various approaches to similar situations were discussed. The issue of the reaction to stockout situations was raised, as well as that of the variability of lead time. Finally, the value of advance demand information was discussed, and highlighted as an area of potential further study at *Clickabox*.

Chapter 3

Clickabox Factory

In order to develop an appropriate inventory model for *Clickabox* factory, it is necessary to have a clear understanding of *Clickabox*'s products, business objectives, as well as administrative and production processes. The objective of this chapter is to provide the reader with the information necessary to gain this understanding. It is against the background provided in this chapter that the assumptions made for the development of the inventory model in Chapter 5 should be understood.

Figure 3.1: *View of Clickabox from Parin Street. The office block and parking lot are visible. To the left of the (white) office block the receiving goods entrance is visible, whilst the finished goods exit is just off the photograph, to the right.*

3.1 *Clickabox*: The Business

In this section, the business objectives and business strategy of *Clickabox* will be discussed. These are drivers of the company's operational strategy, of which the management of its inventory forms a major part. The financial situation of the company will also be discussed briefly, in order to give an indication of the growth in business that is taking place.

3.1.1 Business Objectives

There were two primary business objectives set out by the management of *Clickabox* for the 2003 financial year. The first was to achieve a growth in real (after–tax) income of at least 10% per annum. There were various deliverables associated with this objective, which included:

1. Achieving ISO9000 accreditation by June 2003, by measuring waste more accurately, reducing waste by 25%, and improving customer loyalty. The ISO9000 accreditation was achieved in May 2003.

2. Achieving a growth of at least 10% in annual turnover in real terms. A growth of 32% was achieved over the 2003 financial year.

3. Optimising the stockholding process, by developing an inventory model, which is the topic of this thesis.

4. The addition of die–cut capability by June 2003. This would have involved the purchasing of a new machine, which allows boxes to be made to interlock, instead of using the current methods of glueing or stitching. There were three options available: a roller bed die–cutter which costs approximately R100 000, a platten–type die–cutter which ranges in price from R200 000 to R1 000 000, or a rotary die–cutter, the high speed machine used by large scale manufacturers such as *Nampak* [56], which costs approximately R3 000 000. The decision as to which of these machines to purchase has been put on hold temporarily, due to uncertain market conditions and the resultant risk of over–capitalisation.

The second primary objective was to be a recognised leader in e–commerce in the packaging industry in South Africa by December 2004. It was envisaged that this objective would be met by achieving the following goals:

1. The development of the online quote program by June 2003 to the point where it is a saleable product on the international market. The internet quote program expansion was still in progress at the time of writing of this thesis.

2. Selling at least one copy of the online quote program in the financial year ending February 2004.

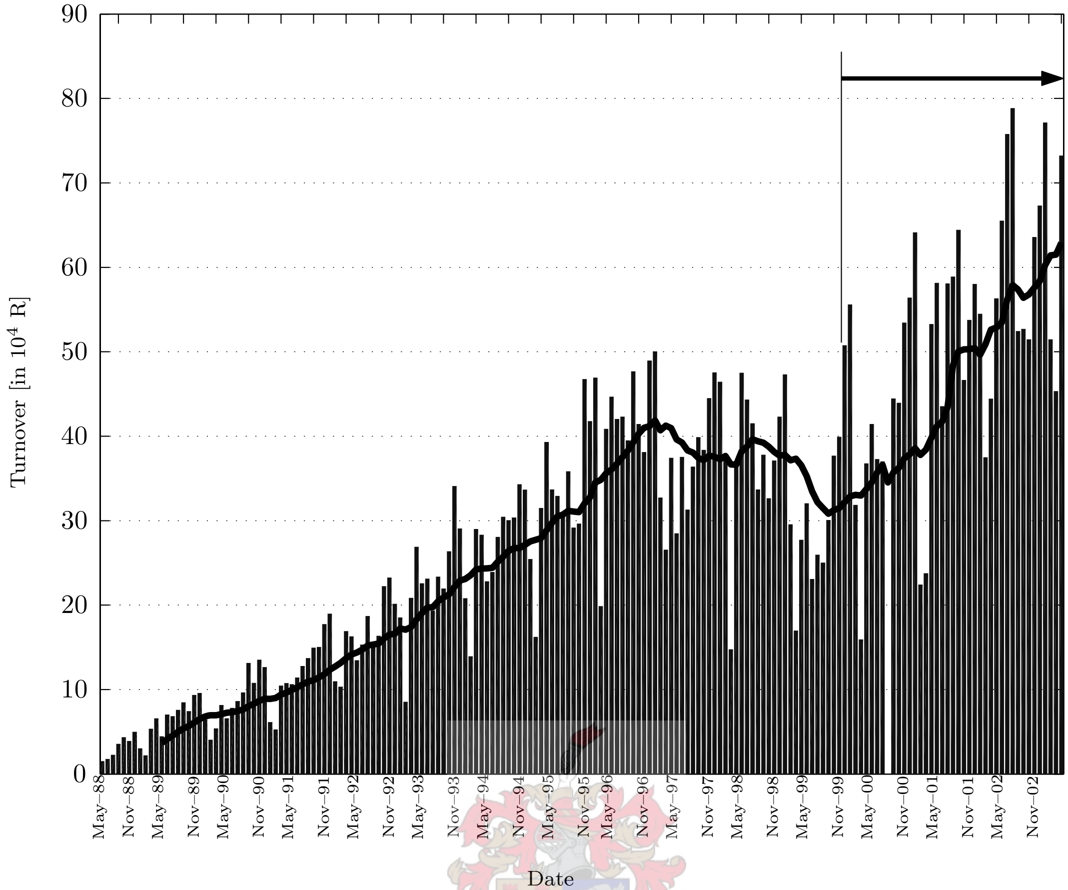


Figure 3.2: *The monthly turnover of Clickabox since its foundation in May 1988, with a moving average line to show the growth in turnover over the 14 year period. The period during which the company has been under the current ownership is indicated by an arrow on the graph.*

The director of *Clickabox* is in the process of drawing up a business plan for this project in conjunction with Mr Leon Swanepoel (a software developer) [69]. The plan involves the extension of the program to apply to not only the cardboard box industry, but also to other industries where two dimensional stock selection is required.

3.1.2 Business Strategy

The business strategy of *Clickabox* involves improving productivity through improved management and control systems. Consultants were employed to assist in the implementation of ISO9000 standards during the period May 2001 to May 2003. Additional future marketing and advertising channels include the company website, commission–based agents, advertising material, such as calendars and flyers, and targeted fax or email advertisements to companies that are similar to existing customers. This thesis forms a part of the stockholding optimization project which is an integral part of the business strategy of the company. Another element to the strategy is recapitalisation of the business, involving replacement of some of the older machinery.

3.1.3 Financial Situation

In the first full financial year of operation as *Clickabox Pty Ltd*, under the current ownership (namely the year ending February 2001), the turnover was R4 516 944 and the operating profit before other income was R238 075. This represents a 27% growth in turnover from the previous financial year, the year during which the change in ownership took place. This growth in turnover continued during the following two years, with a growth of 23% in the year ending February 2002, and 32% in the year ending February 2003.

The monthly turnover of the company since it was founded in May 1988 is shown graphically in Figure 3.2. The arrow indicates the period during which the company has been under the current ownership. The continuous line is a moving average, which shows the steady growth in turnover since the change in ownership in October 1999.

A real growth of 10% per year is targeted for operating profit. The value of stock held in inventory was approximately R450 000 at the time of writing.

3.2 *Clickabox*: The Factory

Some important operational aspects of the factory, such as its location, the factory layout, machinery used and staff employed by *Clickabox* are introduced in this section.

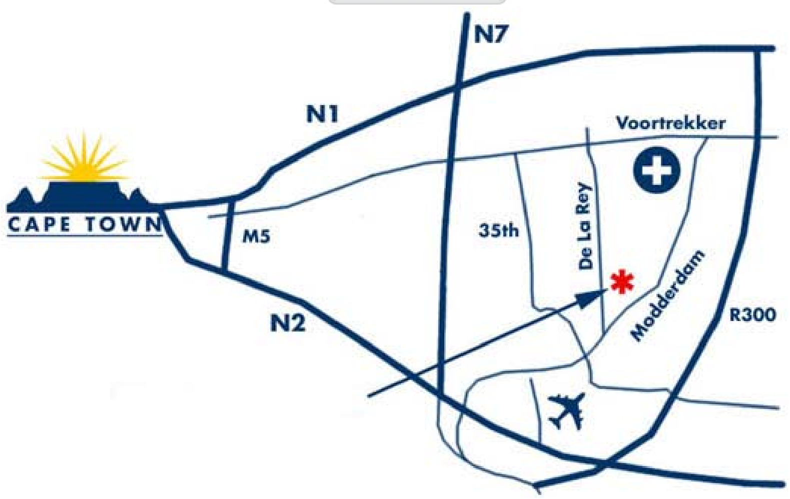


Figure 3.3: *The situation of Clickabox (indicated by the arrow) relative to the City of Cape Town and some major roads, landmarks and Cape Town International Airport.*

3.2.1 Factory Location and Layout

The factory is located in Parow Industrial Area (12 Parin Street), about 50km from the city of Cape Town and 15km from Cape Town International Airport (as shown in Figure 3.3).

The factory is divided into five functional areas. A floor plan of the factory is shown in Figure 3.4. The raw materials store is a 59.3m by 15m area, with access to Parin Street for the receiving of raw material deliveries from suppliers. Raw materials are stored in demarcated bays, which run down the length of the raw materials store, on either side of a walkway down the centre, as shown in Figure 3.5(a). Boards may be stacked to a height of 3.2m in these demarcated bays.

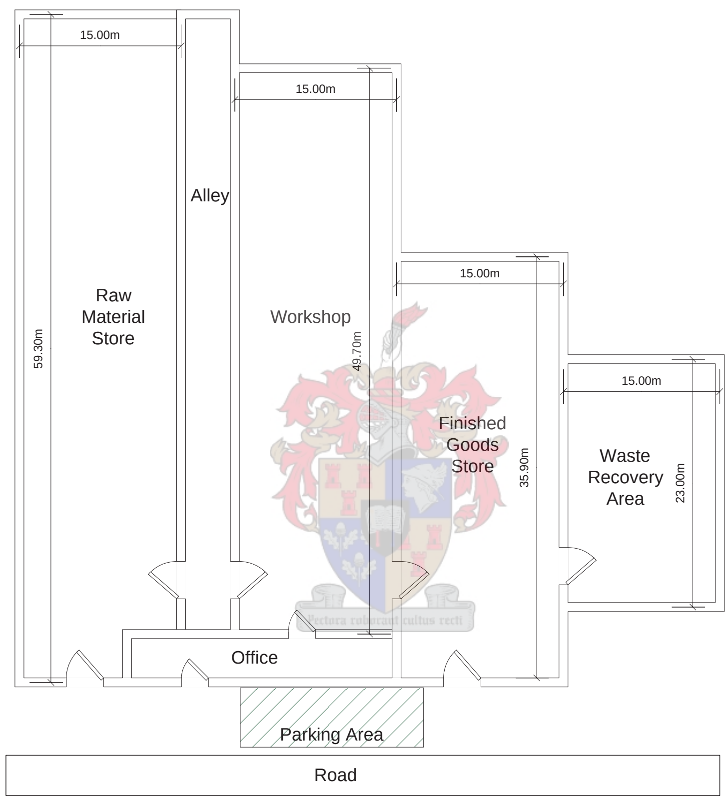


Figure 3.4: *A floor plan of the factory, showing its division into five functional areas.*

Adjacent to the raw materials store is the workshop, a 49.7m by 15m area where manufacturing of the boxes takes place, as shown in Figure 3.6. The equipment used in the production of the boxes is arranged in two production lines in the workshop, one line suitable for large batches and the other for small batches, as the setup time and unit production times differ for the various machines on the two lines.

The finished goods store, shown in Figure 3.5(b), is a 35.9m by 15m area, adjacent to the workshop and the waste recovery area. Completed boxes are stored on wooden pallets in this area, awaiting dispatch, as shown in Figure 3.5(b). It holds three categories of finished goods. The first category consists of boards that have been processed (cut down

(a) Raw Materials Store

(b) Finished Goods Store

(c) Waste Recovery Area

(d) Foyer to Office Area

Figure 3.5: *The storerooms: The raw materials store, where the stock boards are kept in bays, the finished goods store, where goods ready for delivery are kept, and the waste recovery area, where offcuts are kept for re–use or recycling. Also shown is the foyer of the office block, where administration, receiving of quotes,* etc. *takes place.*

to creased sheets of specified dimensions, which may be folded into boxes) and packed to fill an order. These boards are generally dispatched as soon as the manufacturing of the entire order is complete. The second category consists of boards that do not require any processing which are sold as is. The third category are boards that are bought specifically for manufacturing of a particular, popular box, for customers who regularly order boxes of the same type and dimensions. These boards are kept in the receiving area until processing. They are processed as soon as there is spare capacity, and then transferred immediately to the finished goods store. The finished goods store has access to Parin Street for the dispatching of orders.

The waste recovery area, shown in Figure 3.5(c), is a 23m by 15m area which provides storage space for large offcuts, which are stored for later use, and unusable offcuts, which are recycled. Offcuts of more than 50cm in each direction are classified as re–usable. These are entered into the database as raw materials (available for use in production)

Figure 3.6: *The production area, where the manufacturing of boxes takes place. The machinery is organised in two production lines, on either side of the central aisle. The production line on the left of the photo is used for small order batches and the line on the right is used for large order batches.*

at half their original cost price. This ensures that they will be the first choice for the production of boxes when they are of a suitable size. Offcuts of more than 50cm in just one direction are entered into the database as raw materials, but at a price of zero, and so will also be used as soon as possible. Other offcuts are put into a bin for recycling. These scraps are collected weekly by *Nampak* [56], who pays *Clickabox* R0.38/kg for the scraps. This results in an income of approximately R750 per month. The entire waste recovery operation, including both the recycling and re–using of offcuts, currently generates an income of approximately R1 875 per month [75].

Finally, there is an office area at the front of the warehouse, approximately 5m by 15m in dimension, where the administration, receiving of quotes, *etc.* takes place. This comprises a separate office for the director, a meeting room, and adjacent offices for the other administrative staff. This gives a total floor area of approximately 2518.5m^2. The foyer at the entrance to the office area is shown in Figure 3.5(d).

(a) Slitter/Creaser (b) Beamslotter

(c) Stapler (d) Strapper

Figure 3.7: *Machinery used for small order batches, and the strapper. The Slitter/Creaser cuts the cardboards into sheets, and makes the required creases. The Beamslotter cuts slots in the boards for folding. The Stapler uses metal staples to join box flaps, and the strapper bundles the completed boxes together.*

3.2.2 Workshop Machinery and Manufacturing Process

As mentioned in §3.2.1, the equipment used for the production of cardboard boxes is arranged in two production lines in the workshop, one line suitable for large order batches and the other for small order batches. Machines are set up manually according to specifications on a so–called works ticket, which is printed for each order. All machines require the operators to feed through the cardboard manually. Boxes are manufactured according to a cutting pattern, which specifies the required creases and slots that need to be cut into the cardboard.

Small order batches are processed by three machines in succession. These machines are shown in Figure 3.7(a)–(c). The first machine in this sequence is the slitter/creaser (Figure 3.7(a)), which cuts the cardboards into sheets of the required dimensions, and makes creases in the cardboard as required for folding. This machine has an average setup

(a) Printer/Slitter/Creaser (b) Gluer

Figure 3.8: *Machinery used for large order batches. The Printer/Slitter/Creaser cuts boards into sheets, makes the required creases, and prints in one or two colours. The Gluer joins the folds of boxes.*

time of 3 to 10 minutes, depending on the number of creases required, and a production run time of approximately 400 sheets an hour, for average sheet sizes [24].

The second machine is the beamslotter (Figure 3.7(b)), which cuts slots into the cardboard, as specified on the works ticket. The beamslotter has a setup time of 5 minutes, and a production run time of approximately 205 sheets an hour [24].

Finally, there are four staplers, which are used to join the box flaps (one is shown in Figure 3.7(c)). The customer may choose whether to have the boxes stapled or glued, but in general, when it is not specified, the stapler will be used for the small order batches.

There are also two strapping machines (one is shown in Figure 3.7(d)), which are used to bundle the completed boxes together, so that bundles may be transported together conveniently and tidily.

Larger order batches are processed by the printer/creaser/slotter (shown in Figure 3.8(a)) — a single machine which performs all the functions of the slitter/creaser and the beamslotter, as well as any printing on boxes, if required by the customer. This machine takes longer to set up than the other machines, 35 to 200 minutes, depending on the number of colours to be printed, but it is more efficient for large batch processing, as it has a shorter unit processing time (approximately 800 sheets per hour) [24].

The boxes in large order batches are then joined by the carton glueing machine, shown in Figure 3.8(b). The gluer has a 15 minute setup time and a processing time of approximately 800 sheets per hour [24].

There are two forklifts used to transport the boards around the factory (see Figure 3.9(a)). Completed boxes are delivered to the customer using the delivery vehicle shown in Figure 3.9(b).

(a) Forklift (b) Delivery Vehicle

Figure 3.9: *A forklift is used for moving piles of boards or boxes around the factory, and a delivery vehicle is used for transportation of the finished goods to the customer.*

The stages of the manufacturing process described here are illustrated schematically in Figure 3.10.

3.2.3 Factory Staff

Piet and Miemie Taljaard are the owners and directors of the company, and it is managed by Piet Taljaard. There are eight other permanent staff members: a supervisor of quotes and technical issues, an accountant, and six factory staff members, who perform production tasks in the workshop. When necessary, in periods of high demand, additional workers are employed (generally up to three) on an hourly basis [70].

3.3 Raw Materials

Most raw materials are ordered from *Mondi* [52]. Other suppliers, such as *Nampak* [56], are used when *Mondi* cannot meet the demand. Figure 3.11 shows a raw materials delivery being received at *Clickabox*.

Cardboard is classified by its type, flute and category. There is a range of cardboard types available from *Mondi*. Of these, only two types are kept in stock at *Clickabox*. These are the *Mondi Liner* (single corrugated board) and *Double Wall Board* (double corrugated board). The reason for this is that the other types are more expensive and therefore less popular. If required, these boards may be bought in for special orders. The

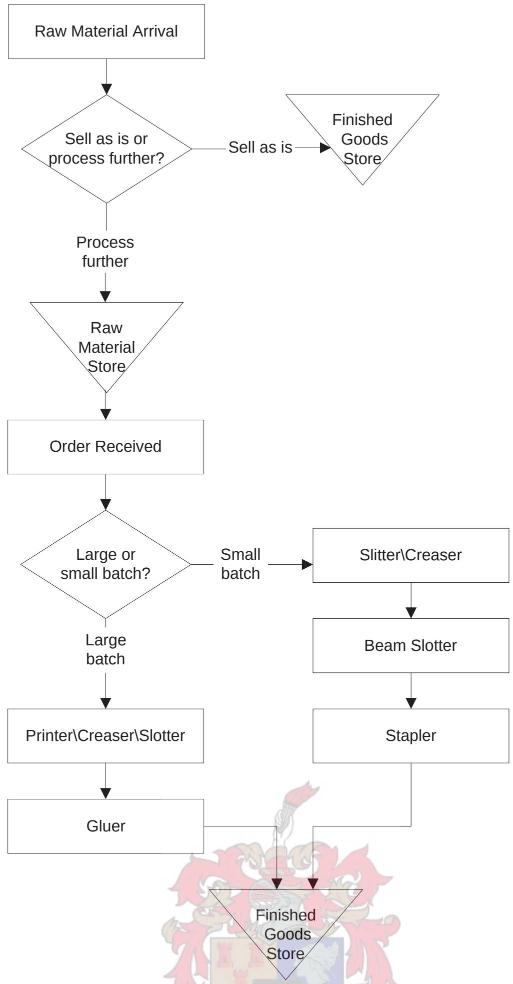


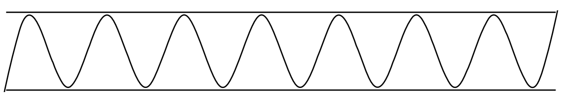
Figure 3.10: *Manufacturing Process Flow Diagram*

single corrugated board is the most common material used. Double corrugated board is used when stronger boxes and extra padding are required. Cross sections of these two cardboard types are shown schematically in Figure 3.12.

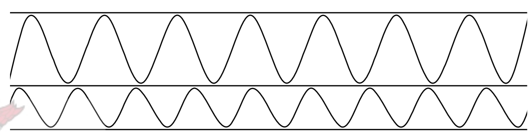
The flute type describes the structure of the wave shaped cardboard material that makes up a board's corrugation. There are a number of flute types, each with a different thickness and therefore different usages. The flute types used by customers of *Clickabox* are described in Table 3.1. Double corrugated board is composed of a combination of a B and a C flute. The fluting gives the board a directional property, the width of the board is measured in the flute direction. The directional property is important in the production process as it determines to a large degree the structural properties and strength of a box.

Cardboards are divided into categories, dependent on their quality and thickness. In the industry, these are referred to as classes, however the term classes will be used in a different sense in this thesis. Hence classes of cardboard will be referred to as categories. The categories allow for a balance between cost and performance for various applications. Thickness of linerboard increases from Category A (typically 140 or 150g/m) to Category

Figure 3.11: *A truck bringing a delivery of raw materials from a supplier to the delivery entrance of Clickabox, in Parin Street.*



(a) Profile of a Single Corrugated Board



(b) Profile of a Double Corrugated Board

Figure 3.12: *Profiles of the two cardboard types kept in inventory at* Clickabox.

D (typically 250g/m). The fluting is likely to deform the liner of Category A board, making the back of the board unsuitable for printing or lamination. However the thicker Category C and D boards may be laminated on both sides.

3.4 Order Classification

There are three classes of orders received by *Clickabox*, namely:

Standard Orders: These comprise orders for box types, such as "A4", which can only be manufactured in one way, or recurring orders for an established customer, such as *Continental China* [17], which has a few set box orders. The raw materials used for these standard orders may therefore be processed as soon as they arrive and there is spare time in between the production of other orders, as mentioned in §3.2.1, and need not be kept (unprocessed) in the warehouse. The processed material is then stored as "Finished Goods Stock". This produce will always be either on hand or on order, as it is re–ordered as soon as levels are low. Some raw materials are sold

FLUTE	THICKNESS	COMMENTS
C	4.4 mm	good stacking strength
		good crushing resistance
		very common material
		typical packaging for glass, furniture, dairy products
B	3.2 mm	good puncture resistance
		less space consumed in warehouse
		typical packaging for canned goods, displays
E	1.6 mm	light weight
		strong alternative to paper board
		superior printing surface
		excellent for custom die cut boxes
		typical packaging for displays, point of purchase boxes

Table 3.1: *The properties and usages of the different flute types available.*

as is. Boards currently used for standard orders, at the time of writing, are shown in Table 3.2.

Reference	Type	Category	Flute	Size (in mm)
Stock 600	Mondi Liner	A	C	600×400×400
Stock 505	Mondi Liner	A	C	505×350×355
Ref A3	Mondi Liner	B	C	430×305×290
Ref A4	Mondi Liner	B	C	305×215×290
P–AD4	Mondi Liner	B	C	440×360×408
P–EXP	Double Wall Board	A	B and C	526×418×318
P–ONT	Double Wall Board	A	B and C	458×458×254
P–W2	Mondi Liner	B	C	500×250×250
P 1	Mondi Liner	B	C	560×334×500
P 3	Mondi Liner	B	C	560×334×286
P 4	Mondi Liner	B	C	750×463×200
PTD	Mondi Liner	B	C	750×463×140
PTS	Mondi Liner	B	C	750×463×70

Table 3.2: *The properties of the standard order raw materials boards (in other words, boards used for boxes commonly ordered from Clickabox), which may be manufactured as soon as the raw materials arrive, when there are spare time lulls in between the production of other orders.*

Special Orders: These are orders for boxes not commonly ordered, such as for boxes of a flute type not stocked by *Clickabox*. The cardboard required to produce these orders is therefore brought in on request and not kept in inventory.

Standard Stock Orders: These are orders that utilise the board sizes most commonly ordered, as determined from experience. These are the board sizes kept in inventory, and used by *Clickabox*'s quote program in the preparation of a quote and order form. A section of this thesis will be devoted to determining the ideal dimensions of standard stock board to be kept in inventory, in order to minimize expected average wastage. Prior to this study the standard stock profile consisted of 28 different sizes of Category A, Flute C board (termed AC board) and 18 sizes of Double Wall Board (termed DWB board). The make–up of the previous stock profile is given in Table 3.3.

Category	Flute	Length (in mm)	Width (in mm)
A	C	1 000	1 400
A	C	1 220	21 00
A	C	1 250	1 300
A	C	1 360	2 100
A	C	1 400	1 300
A	C	1 425	1 125
A	C	1 500	1 500
A	C	1 530	1 300
A	C	1 540	910
A	C	1 550	1 400
A	C	1 555	1 540
A	C	1 600	1 300
A	C	1 650	1 500
A	C	1 700	1 000
A	C	1 700	1 200
A	C	1 710	1 500
A	C	1 850	1 510
A	C	1 860	1 200
A	C	1 870	1 055
A	C	1 920	910
A	C	2 000	1 200
A	C	2 000	1 500
A	C	2 100	1 000
A	C	2 200	1 200
A	C	2 200	1 500
A	C	2 300	1 000
A	C	2 300	1 200
A	C	2 358	1 062

(a) Dimensions of AC Stock Board

Category	Flute	Length (in mm)	Width (in mm)
DWB	DWB	1 200	2 100
DWB	DWB	1 350	2 100
DWB	DWB	1 350	2 270
DWB	DWB	1 440	1 020
DWB	DWB	1 500	1 050
DWB	DWB	1 500	1 300
DWB	DWB	1 500	1 400
DWB	DWB	1 600	1 000
DWB	DWB	1 600	1 800
DWB	DWB	1 700	1 200
DWB	DWB	1 840	1 200
DWB	DWB	1 850	1 500
DWB	DWB	1 920	1 800
DWB	DWB	2 000	1 200
DWB	DWB	2 120	1 510
DWB	DWB	2 200	1 350
DWB	DWB	2 300	1 600
DWB	DWB	2 380	1 200

(b) Dimensions of DWB Stock Board

Table 3.3: *Standard stock, the board sizes kept in inventory at* Clickabox *prior to this case study, from which boxes ordered as 'quick orders' were manufactured.*

3.5 Processes at *Clickabox*

The current inventory control processes of *Clickabox* are described in this section. This will allow for a meaningful evaluation of the results obtained by the inventory model developed in Chapter 5, in terms of feasibility of the model, the costs of implementing the model, and potential savings as a result of the model solution (in Chapter 6). The processes followed by *Clickabox*, from providing quotations to the delivery of finished boxes, are then described.

3.5.1 Ordering of Stock from Raw Materials Suppliers

The minimum order quantity per raw material board size that *Clickabox* may order from its suppliers for each order is 1 000 units. However, orders of 500 boards are taken subject to a surplus cost of R0.37 per board. Furthermore, *Mondi* [52] specifies that, where possible, orders should be deckled to one of two widths in order to fit the width of their machinery. Orders are processed at the supplier as soon as the deckle width is full. For punctual delivery, it is optimal to place an order which fills the deckle width. So if, for example, one orders a type B flute, which is uncommon, one must either wait for a very long time or pay for the full width of the cardboard from which boards are cut. The deckle widths are 2 415mm and 2 275mm.

The minimum order quantity policy was, however, under review at the time of writing, and furthermore, it is possible for *Clickabox* to use other suppliers who do not impose minimum order quantities. These restrictions will therefore not be included in the model when computing the suggested raw material re–order quantities. The concept of minimum order quantities is, instead, cited as a potential area of future study at *Clickabox*.

Currently, a continuous review ($\overline{s}$,$\overline{S}$) inventory policy is followed at *Clickabox*. Re–ordering is done daily, when necessary. Two values are defined for this policy:

Re–order level, $\overline{s}$: The stock level at which re–ordering of stock should take place.

Order–up–to level, $\overline{S}$: The stock level up to which the replenishment order brings the stock on hand.

Each morning the stock level of each board type used the previous day is taken. If the level has fallen below the re–order level, an order is placed for the re–order quantity. Currently the re–order level is 300 for all board types. The order–up–to–level is 800 for DWB boards and 1300 for AC boards. The lead time for orders to arrive is approximately two weeks.

3.5.2 Ordering of Boxes from *Clickabox* by clients

The processes followed by *Clickabox* when receiving orders from customers during the generation of a quotation, and during the manufacturing of boxes, are described in this section.

Box dimensions and types

In the packaging industry, the size of corrugated boxes is expressed by three dimensions: Length (L), Width (W) and Depth (D), all in millimeters (mm). These measurements refer to the inside dimensions of the box. The length, L, is the larger dimension of the opening, the width, W, is the smaller dimension of the opening and the depth, D, is the dimension perpendicular to the opening. This differs from the terminology used for

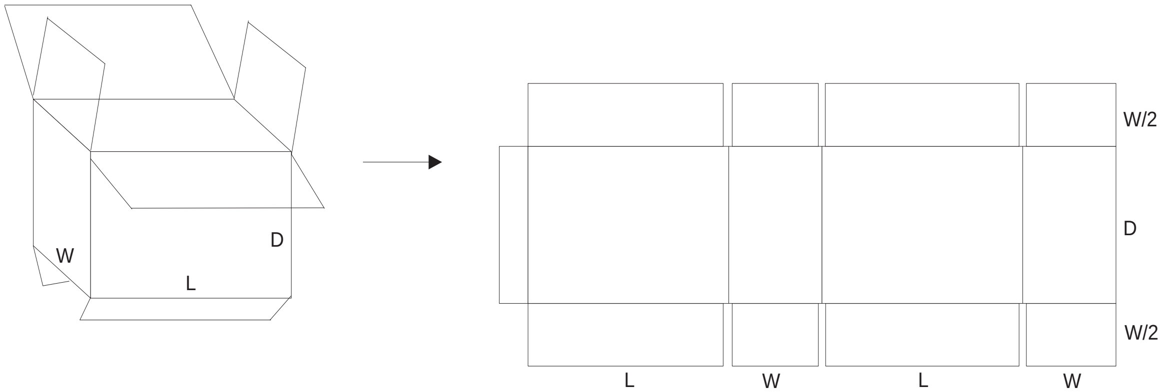


Figure 3.13: *The 'Regular Slotted Carton', and the cutting pattern according to which it is manufactured.*

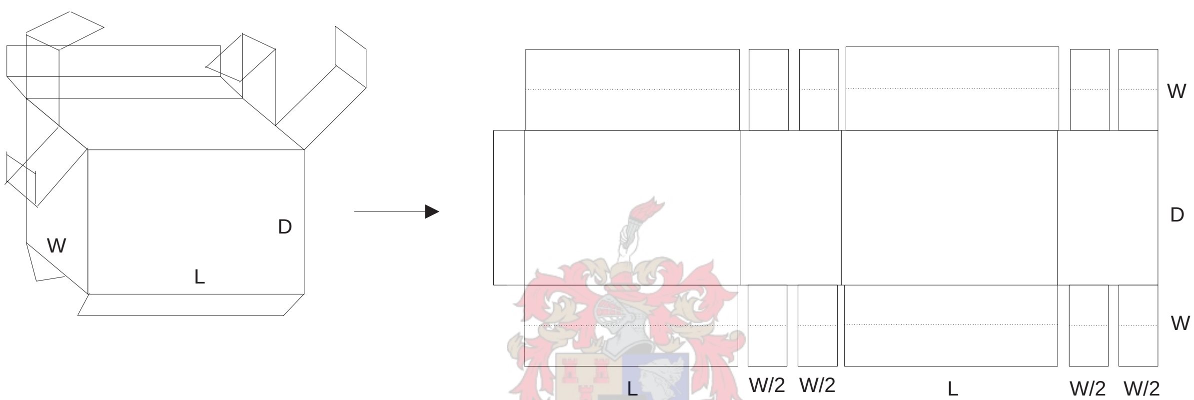


Figure 3.14: *The 'Tuck–in–Flap' box, and the cutting pattern according to which it is manufactured.*

cardboard sheets, where the length and width are determined by the directional property of the fluting, as discussed previously.

There are a number of different box styles that may be ordered by clients, each with different creasing and cutting requirements. The most commonly ordered box style is the 'Regular Slotted Carton'. This box is produced from one sheet of cardboard, with very little manufacturing wastage and is therefore the most economical. The Regular Slotted Carton is shown schematically, together with its cutting pattern, in Figure 3.13.

Another box style, the 'Tuck–in–flap', is shown together with its cutting pattern in Figure 3.14.

Quote Generation

The *Clickabox* website [57], mentioned in §1.1.2 and from which *Clickabox* derives its name, provides customers with access to a computer program which allows the user to request an online quotation for a specific box, and to place an order. Before this program was implemented, pricing was done by means of an excel sheet, manual look–up of stock on hand in inventory, and then a manual stock reserving process. This was a time–consuming process, and led to a number of problems. One such problem arose when the manual stock reserving process was delayed, and completed after the material on which the quote was based had already been used for another order. The following computerised process has therefore replaced the manual method.

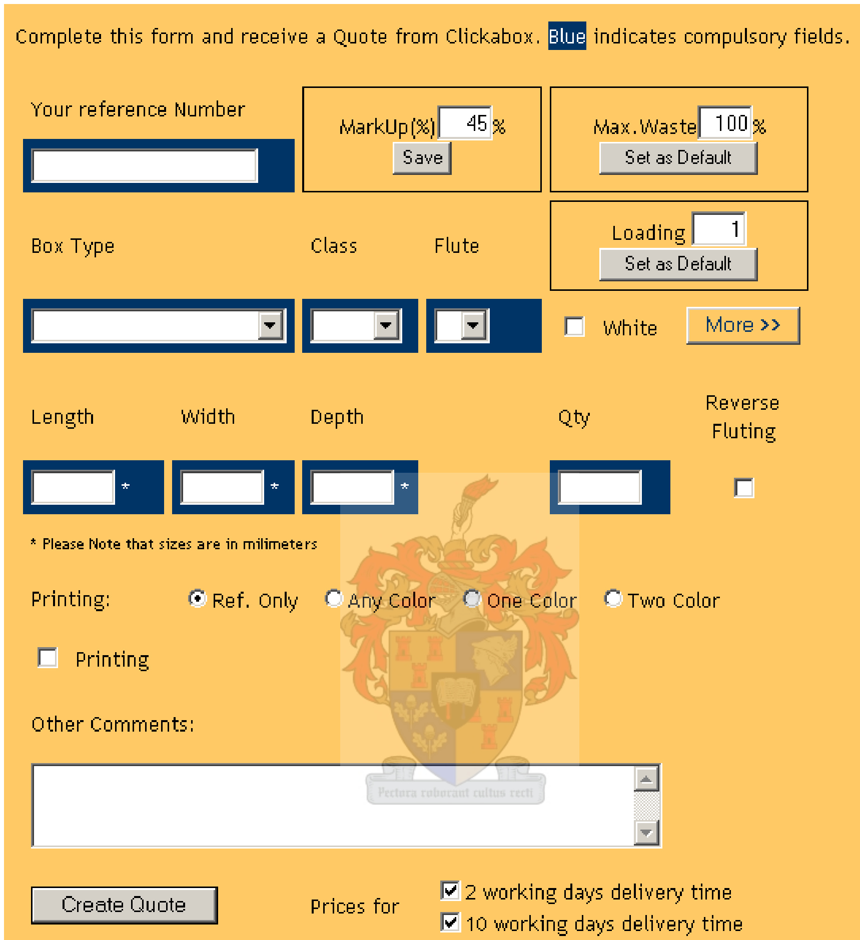


Figure 3.15: *The order screen from the Clickabox Website* [57], *on which the details of the box required are entered by the customer.*

1. **A customer requests a quotation for a certain number of boxes of specific dimensions.** This request may either be entered by a client on the website or may be received telephonically and then processed on the website by an employee at the *Clickabox* office block. The *Clickabox* website requests input from the client as to the box style, board specifications (category and flute), box size (length, width and depth), order quantity required, and whether or not printing is required. The basic order screen from the website, on which this information is entered, is shown in Figure 3.15.

2. **The required sheet (cutting pattern) dimensions are calculated.** The sheet dimensions are determined by the box dimensions and style. These dimensions include an allowance for creasing, which is dependent on the flute and cardboard type. The creasing allowances are given in Table 3.4.

Type	Flute	Creases against flute direction (a)	Wall thickness in flute direction (b)	Creases in flute direction (c)	Wall thickness against flute direction (d)
SC	B	3	2	9	1
SC	C	5	3	10	3
SC	E	2	2	5	1
DWB	B and C	9	6	18	7

Table 3.4: *Creasing allowances (in mm), dependent on flute and cardboard type. These allowances are included in the calculation of the dimensions of the cardboard sheet required to manufacture the board ordered.*

For example, the dimensions of the sheet required to manufacture a Regular Slotted Carton of length L, width W, and depth D are then $(L+a)+(W+a)+(L+a)+(W+d)$ by $(W/2+b)+(D+c)+(W/2+b)$, as illustrated in Figure 3.16, where the symbols a, b, c and d are as defined in Table 3.4.

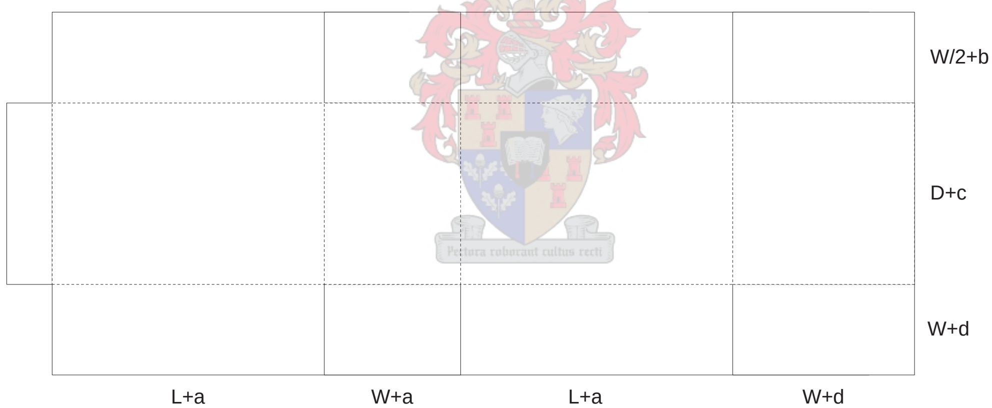


Figure 3.16: *The dimensions of the sheet required to manufacture a regular slotted carton of length L, width W and depth D.*

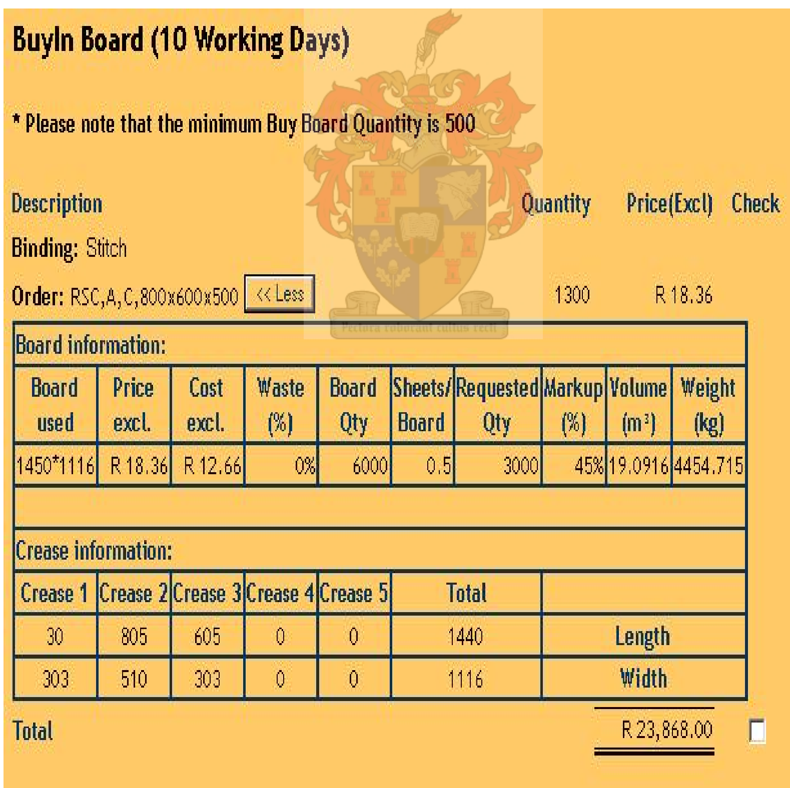
3. **The delivery option is selected.** The customer must also decide whether to order from stock, in which case lead time for the order is two working days, or to take the "buy–in" option. This is the cheaper option for the client: exactly the right cardboard size is ordered from the suppliers for manufacturing and therefore there is no wastage. The disadvantage is that the order may take more than seven working days if this option is executed.

4. **A list of boards that may be used in the manufacturing of the required boxes is compiled.** If the "buy–in" option is chosen, a board of the optimal size

is ordered from the supplier, and it is priced accordingly. If not, the program then searches through all the boards in the Pastel database (where on hand inventory data is stored) to obtain a list of viable boards from which the required sheet may be cut. This list is sorted in order of increasing wastage, and then available quantities are checked. The result is an array of boards, the on hand inventory of that board, and the cost of using that board.

5. **A quote is compiled.** Activity based costing is used. A fixed setup cost is charged, and divided by the number of boards ordered to give a unit cost per board. This is the only form of quantity discount given, in that the greater the quantity of boards ordered, the lower the unit allocation of the fixed cost. This is added to the variable cost (dependent on the production time for that board type) and raw materials cost per board, and the full amount is multiplied by a markup. The average markup is 1.45; discounts are given to larger customers.

   This costing information is presented to the user in the form of an invoice. A quotation generated for the "buy–in" option is shown in Figure 3.17, and a quotation generated for the same order using the stock board option is shown in Figure 3.18. The price quoted is calculated on the total area used, so the use of sub–optimal boards is more expensive.



BuyIn Board (10 Working Days)

* Please note that the minimum Buy Board Quantity is 500

Description	Quantity	Price(Excl)	Check
Binding: Stitch			
Order: RSC,A,C,800x600x500 << Less	1300	R 18.36	

Board information:

Board used	Price excl.	Cost excl.	Waste (%)	Board Qty	Sheets/ Board	Requested Qty	Markup (%)	Volume (m³)	Weight (kg)
1450*1116	R 18.36	R 12.66	0%	6000	0.5	3000	45%	19.0916	4454.715

Crease information:

Crease 1	Crease 2	Crease 3	Crease 4	Crease 5	Total	
30	805	605	0	0	1440	Length
303	510	303	0	0	1116	Width

Total R 23,868.00

Figure 3.17: *The quotation screen from the Clickabox Website* [57], *giving a quotation for a buy–in board order — one of the two options available to the customer, including the price and creasing information.*

Stock Board (2 Working Days)

Description	Quantity	Price(Excl)	Check
Binding: Stitch			
Order: RSC,A,C,800x600x500 << Less	1284	R 18.73	

Board information:									
Board used	Price excl.	Cost excl.	Waste (%)	Board Qty	Sheets/ Board	Requested Qty	Markup (%)	Volume (m³)	Weight (kg)
AC1460*2270	R 18.73	R 12.92	2.4%	1284	1	3000	45%	19.0916	2227.3575

Crease information:						
Crease 1	Crease 2	Crease 3	Crease 4	Crease 5	Total	
30	805	605	0	0	1440	Length
303	510	303	0	0	1116	Width

	Quantity	Price(Excl)	Check
Order: RSC,A,C,800x600x500 << Less	16	R 27.50	

Board information:									
Board used	Price excl.	Cost excl.	Waste (%)	Board Qty	Sheets/ Board	Requested Qty	Markup (%)	Volume (m³)	Weight (kg)
AC2280*2270	R 27.50	R 18.97	59.92%	61	1	3000	45%	19.0916	2227.3575

Crease information:						
Crease 1	Crease 2	Crease 3	Crease 4	Crease 5	Total	
30	805	605	0	0	1440	Length
303	510	303	0	0	1116	Width

	Quantity	Price(Excl)	Check
Total	1300	R 24,489.32	☐

Figure 3.18: *The quotation screen from the Clickabox Website* [57], *giving a quotation for a stock board order, including the price and creasing information for two different stock board types which would be used to satisfy the order.*

Processing of a sales order

If the quote is accepted by the client, a sales order is processed, and a quote number is generated. This reserves the required raw materials in inventory. If the raw materials are to be bought in, a board purchasing order is filled out. Once the quote has been issued, either manually or via the quote program, and accepted, a works ticket is generated, which details the dimensions of the box, creases required, and any special directives. This is the document with the specifications used by the machine operators to set up the machinery during the production process. The number of boxes specified on the works ticket allows for a percentage wastage, dependent on which machinery is to be used. For example, the printer has a waste percentage of 2.5%. Wastage from the printer could, for example, be due to smearing of ink.

3.5.3 Manufacturing Process Protocol

The works ticket issued by the internet quote program, is pinned on the scheduling board in the workshop. From the specifications on the works ticket, such as batch size and whether or not printing is required, it is decided on which production line the order is to be processed — that for small order batches or that for large order batches, as mentioned in §3.2.2. This decision, made by the foreman on the floor, may also be influenced by the current workloads on either production line.

The boards are then cut to size, and the printing, slotting, creasing and joining are performed, as described in §3.2.2. Once the boxes have been manufactured, it is confirmed exactly how many boards were used for the order. There are a number of possible causes for discrepancies in this figure. The works ticket contains an allowance for wastage — if, for example, 700 boxes are ordered, it may be calculated that 714 need to be manufactured, taking into account the waste allowance. It is possible, however, that there may only be raw materials to produce 710 boxes in the warehouse, as a delivery of raw materials could have been 10 boards short. Then only 710 will be made, so this needs to be entered on the works ticket and the inventory updated accordingly. There are daily stock–takes of the raw materials in order to keep inventory information as accurate as possible.

The completed boxes are finally moved to the finished goods store, where they are stacked and await dispatching. Offcuts are moved to the Wastage recovery area for re–use or recycling.

3.5.4 Administrative Manufacturing Process

The accounting package used is *Pastel Manufacturing* [58]. All information about stock — raw material and finished goods, clients *etc.* — is stored in this system. Each raw materials type is assigned a code, and each box order is associated in Pastel with a set of raw material codes. This is known as the *Bill of Materials*. An example of a Bill of Materials from Pastel is shown in Figure 3.19.

CLICKABOX (PTY) LTD — 27/06/2001 14:22 Page: 1
Prepared by: CORRUCAPE (PTY) LTD

Bill of Materials

Code :	505	AC/RSC/505*350*355			
Manufactured Item :	505	505 × 350 × 355			
Item	Description	Unit	Quantity	Average Cost	Last Cost
505–AC	1 748 × 717 Cr 176–365–176		1.000	3.277	3.170
Total for Components				3.277	3.170
Cost 1				0.000	0.000
Cost 2				0.000	0.000
Cost 3				0.000	0.000
Total Cost for Manufactured Item				3.277	3.170
Update Manuf. Selling Prices : No					
No. of Notes : 0					

Figure 3.19*: Bill of Materials*

When a box is manufactured, it is necessary to perform a process, known as the administrative manufacturing process, in order to update the inventory. A finished goods code is generated, representing the type and dimensions of the finished box. The finished goods inventory is credited with the number of these boxes manufactured. The raw materials that made up the boxes, as specified in the Bill of Materials, are subtracted from the raw materials inventory.

3.5.5 Post Manufacturing Process

Once completed, the boxes are bundled and strapped together so that they are easy to handle in batches. A delivery note is completed with the totals of boxes to be delivered and is attached to order bundles; this is signed on delivery. An invoice is sent out to clients after delivery, and statements are sent at the end of each month.

3.6 Chapter Summary

The various processes of *Clickabox* factory were described in some detail in this chapter, in order to provide the reader with an understanding of the environment in which this study was conducted. The business strategy and objectives of the factory were discussed in §3.1, as well as its financial situation, highlighting the importance to *Clickabox* of an efficient inventory management system. The layout, products and processes of the factory were described in §3.2–3.5, focussing, in particular, on previous inventory control practice.

Chapter 4

Analysis of Board Demand

There are a number of factors influencing the demand for cardboard boxes. The corrugated carton industry in the Western Cape is highly dependent on the fruit industry, and so demand increases notably during the fruit–picking season, May and October to January. Also, being a small but growing manufacturer, the averages of board size and quantity ordered from *Clickabox* may be affected significantly with the acquisition of even one new, large client [70]. Thus any calculations of optimal board or re–order strategy must be dynamic. The orders placed over a two year period for boxes from *Clickabox* factory[1] were analysed in order to answer the first research question formulated in §1.2 (which boards to keep in stock) and to begin the process of answering the second research question (what inventory policy to follow) by examining the nature of the demand process. The demand data available are described in §4.1, and then the various steps taken towards determining an optimal set of stock boards to be held in inventory, based on these historic data, are described in §4.2. Finally, in §4.3, the nature of the demand process is discussed, and the way in which this demand will be modelled in Chapter 5 is detailed. The notion of *board preference vectors* is introduced, and a set of demand distributions for each of these board preference vectors is derived.

4.1 Description of Demand Data

Data from the period 1 February 2001 — 31 January 2003 were used to obtain an impression of demand at *Clickabox*. This set of data was extracted from works tickets which are stored by *Clickabox* in excel worksheets for each board manufactured. It was analyzed by means of a program written by the author in Visual Basic [50][2]. Data were extracted from a total of 3 965 works tickets, representing all orders processed during the period mentioned above. February 2001 is the earliest period for which data were available, as it is only since then that works tickets have been created electronically and stored by *Clickabox*.

[1] A snapshot of this data is given in Appendix D. The full list of works tickets is included on a CD inside the back cover of this thesis.

[2] The relevant Visual Basic source code, as well as the structure of the Microsoft Access database tables, are given in Appendix C.

The following details of each order were observed:

- **Order Date**. The date on which the order was placed.

- **Board Style**. The type, category and flute of cardboard required to process the order.

- **Box Design**. The design of the box required, for example a *regular slotted carton*. This design influences the dimensions of the board required to manufacture the box.

- **Box Dimensions**. The length, width and depth of the required box.

- **Quantity**. The number of boxes ordered.

- **Sheet dimensions required**. The dimensions of the sheet (length and width) required to produce the box ordered, calculated from the box dimensions, style and design, with standard provisions for wastage, *etc*. For the purposes of establishing demand distributions in this chapter, and for the inventory model which will be developed in Chapter 5, customer orders are considered to be sheet orders, *i.e.* the box design and dimensions are no longer relevant, only the dimensions and cardboard type of the sheet are required.

- **Board used**. A list of the dimensions of the boards actually used to process each order, is stored. Each sheet may be produced by a number of boards, with varying amounts of wastage associated with each board type chosen to fulfil an order for a specific sheet type. Multiple board types may be used to process an order, if there is not sufficient stock to process the entire order with the optimal board. The works ticket lists all board types, and the quantity of each board type that was used, in order of preference (in terms of wastage incurred).

CLICKABOX WORKS TICKET						Date: 27/02/2002
Design:	RSC	Board Type: DWB			R/m^2: R4.46	
Size:	550 × 330 × 430				Printing: Ref only	
Quantity: 50		Delivery: 05/03/2002			Joint: Stitch	
Delivery address: Reactor Road, Stikland Triangle Farm						
Creasings:	35	559	339	559	337	Sheet Length: 1829
		171	448	171		Sheet Width: 790
Stock Board		Quantity		Sheets/board	% Waste	Board Area
DWB 1920×0900		10		1L×1W=1	18.94%	1 728 000
DWB 1920×1800		15		1L×2W=2	18.94%	3 456 000
DWB 2100×1700		25		1L×2W=2	22.87%	3 570 000

Figure 4.1: *Works ticket produced upon a client's acceptance of a quotation, indicating the board style, box design and dimensions, delivery details, and the quantity required, as well as creasing and printing instructions. Also indicated are the stock boards from which this order will be produced, and the quantities of each board that will be used.*

An example of a works ticket is given in Figure 4.1.

4.2 Determining Suggested Stock Board Profile

The steps taken towards determining an optimal set of stock boards to be kept in inventory, using the historical data described in the previous section, are now described. An ABC analysis was conducted on the various cardboard types available, and the cardboard types responsible for the largest percentage of annual profit were determined. The results of this analysis are given in §4.2.1. The demand data for these cardboard types were then analysed, and a heuristic was used to determine a set of suitable stock boards (as detailed in §4.2.2–4.2.4) to be kept in stock. The resulting set of boards recommended to be kept in inventory is given in §4.2.5, and compared to the previous stock profile.

4.2.1 ABC Analysis of Cardboard Types

Pareto's law is a concept developed by Vilfredo Pareto, a 19th century Italian economist, who noticed a common phenomenon that a small percentage of a group of individuals typically accounts for the largest fraction of the group impact [55]. ABC Analysis is a form of Pareto analysis, developed by *General Electric* during the 1950's, and applied to a set of products kept in inventory in order to classify them according to financial impact [81]. The ABC Classification places items into three basic categories, which may be treated differently in the processes of stock planning and control. These categories are:

Class A Items: Items that, according to an ABC classification, belong to a small set of products that represents approximately 75–80%[3] of the annual demand, usage or production volume, in monetary terms, but only some 15–20% of the inventory items. For the purpose of stock control and planning, the greatest attention is paid to this category of products. Class A items typically are of strategic importance to the business concerned.

Class B Items: An intermediate set of products, representing approximately only 15–20% of the annual demand, usage or production value, but some 20–25% of the total number of inventory items. Less management attention is paid to this class than to the Class A items.

Class C Items: Products which, according to an ABC classification, belong to the 60–65% of inventory that typically represent only approximately 5–10% of the annual demand, usage or production value. Least attention is paid to this category for the purpose of stock control and planning.

The first step in an ABC Classification at *Clickabox* was to determine a ranking, according to the total profit yielded by each cardboard type per year, termed the annual usage value[4]. Profit per board was calculated as the difference between raw material cost and

[3]These percentages represent typical values, which will vary for different sets of data. In the classification process, only the percentage of items is set for each class, the usuage value represented by each class is then derived [55].

[4]The profit per board multiplied by the average number of boards of that type sold in a year.

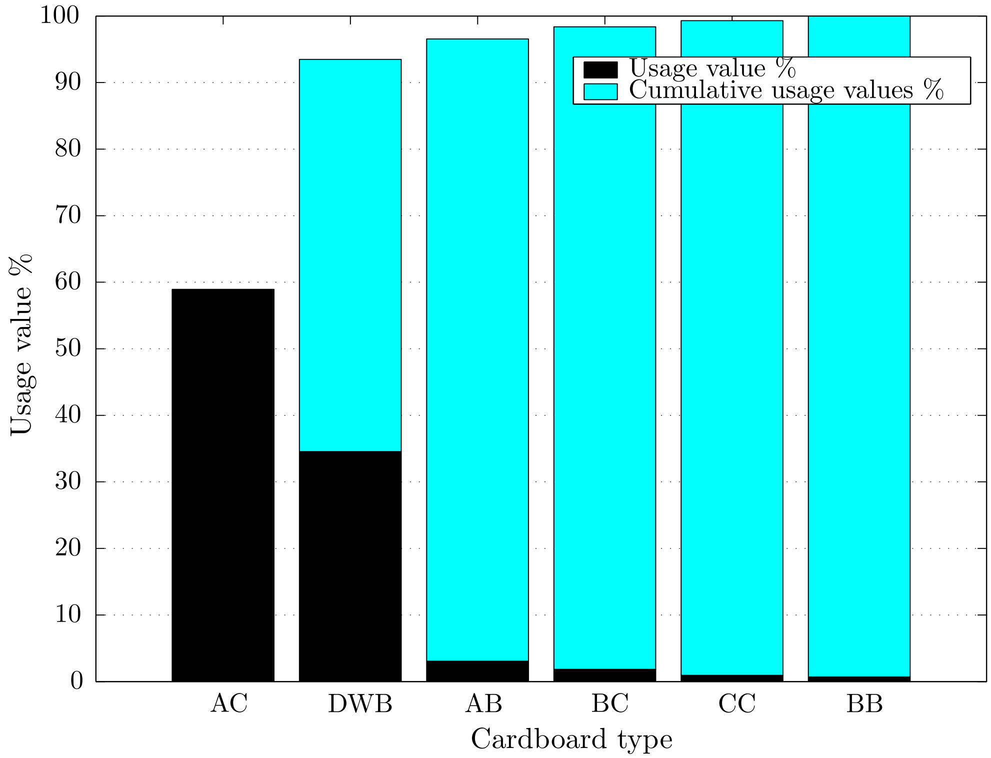


Figure 4.2: *The individual and cumulative usage values of the six cardboard types available, used for the ABC classification of the cardboard types at Clickabox.*

board selling price (per square metre) multiplied by the average board size for that type. Next, products were ranked in order of decreasing usage value. The cumulative usage value was then calculated progressively, and expressed as a percentage of the total usage value. These cumulative usage values are shown in Figure 4.2.

Finally, the cumulative percentage of items was derived, by expressing the rank number as a percentage of the total number of items (here, cardboard types). In other words each cardboard type comprises 16.6% of the number of different cardboard types that may be kept in inventory. The class break points were then set, as shown in Figure 4.3 and Table 4.1, with the top 16.6% of the items (cardboard type AC) being classified as Class A items, the next 16.6% of the items (cardboard type DWB) being classified as Class B items, and the remaining 66.6% of the items (cardboard types AB, BC, CC and BB) being classified as Class C items.

The results of this analysis showed that the cardboard type "AC" is a Class A Item, accounting for approximately 59% of the profit, the "DWB" boards are Class B Items, accounting for approximately 35% of the profit, and the other cardboard types are the Class C Items, collectively accounting for approximately only 6% of the profit. It was consequently decided that the analysis of *Clickabox's* inventory policy would focus on AC and DWB boards only.

Rank	Cumulative % of Items	Board Type	% Usage Value	Cumulative % Usage Value	Class
1	16.6	AC	58.92	58.92	A
2	33.3	DWB	34.56	93.48	B
3	50.0	AB	3.07	96.55	C
4	66.6	BC	1.83	98.38	C
5	83.3	CC	0.93	99.31	C
6	100	BB	0.69	100.00	C

Table 4.1: *Determination of the class break points, in terms of the usage value, for the different classes of items (cardboard types) in the ABC classification at Clickabox.*

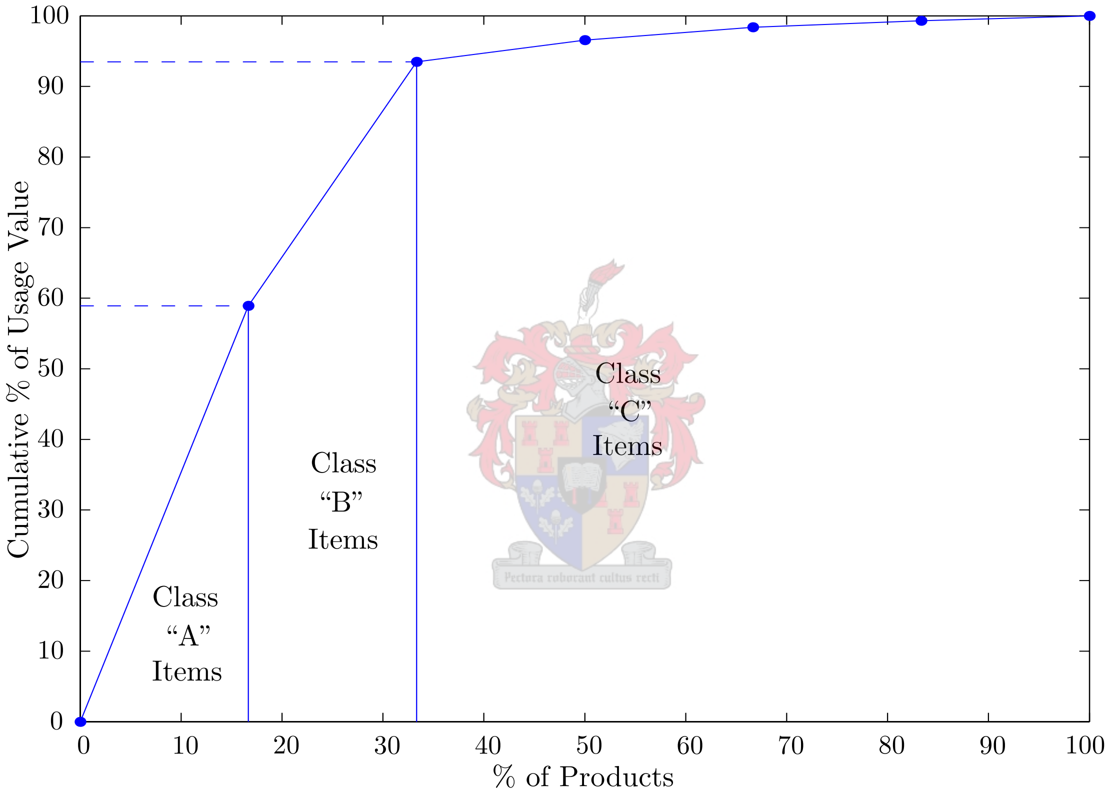
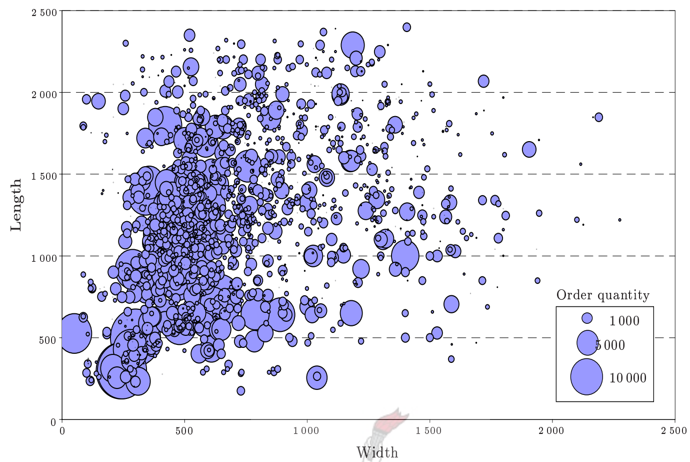


Figure 4.3: *Graphical representation of the class breakpoints in the ABC Classification, showing the cumulative percentage of the total usuage value represented by each class of products.*

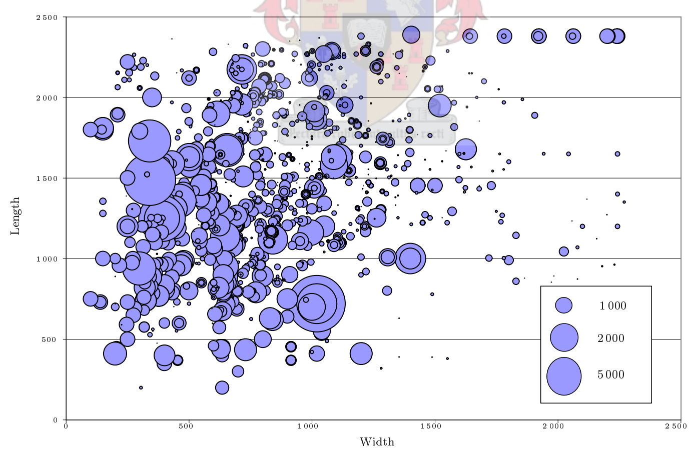
4.2.2 Graphical Representation of Data

Given the results of the ABC analysis, it was decided that the demand data will be analysed as two separate sets, namely Category A — Flute C board (AC cardboard types) and Double Wall Board (DWB cardboard types). Other grades of board may be bought in for special orders and should not be kept in stock, as cumulatively they account for less than 7% of the annual usage value.

The scatter charts in Figure 4.4 were plotted to give a graphical indication of the distribution of order sizes for sheets of various dimensions of both AC and DWB cardboard types. They show the considerable range of different sheet dimensions ordered, one of the properties of the demand profile of *Clickabox* that makes both the choice of stock boards and the choice of inventory control policy difficult.



(a) AC Board



(b) DWB Board

Figure 4.4: *Scatter charts representing all orders placed over the two year period 1 February 2001 — 31 January 2003, for each of the class A and B cardboard types, giving an indication of the range of sheet dimensions ordered from* Clickabox, *and the order quantities involved.*

4.2.3 Restrictions on Stock Boards

Two restrictions are placed on boards under consideration to be made stock boards, for the purposes of the heuristic described below. The first restriction is on the dimensions of the board. The maximum length of board that may be kept in stock at *Clickabox* is 2 490mm, and the maximum width is 2 370mm, due to the capacity of equipment used during the production process. Boards may be ordered by *Clickabox* in dimensions of any integer millimetre measurement from the supplier, but the process described below will use steps of 10mm for practical reasons.

The second restriction is on the minimum percentage of orders that must be met by the set of stock boards, in order to satisfy the service level of 95% of orders being met by boards in inventory (as dictated by the management of *Clickabox* in thesis objective II, see §1.2). This is implemented in the heuristic by setting a minimum number of historical orders which should be met by each stock board[5]. This criterion is reset at the beginning of each iteration of the heuristic, dependent on the number of historical orders that have not yet been filled by boards already selected for stock, and on the number of stock boards still to be selected. So, for example, if there are five boards to be kept in stock, each board is expected to fill at least 20% of the historical orders. If then the first board found by the heuristic fills more than this minumum percentage, say 50% of the historical orders, the second board will only be required to fill a quarter of the remaining historical orders, in other words an eighth of the original number of historical orders (in this case this percentage would be 12.5%). If the second board fills 50% of the remaining historical orders, the third board will only be required to fill a third of the remaining historical orders, in other words a twelfth of the original number of historical orders, and so forth.

4.2.4 Heuristic for Determining a Suggested Stock Profile

Demand data were imported into a database, and an application was written by the author to analyze the data[6]. The demand data are represented by a set $\mathcal{O}$ of ordered pairs $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$, where $L_{\mathcal{O}_f}$ and $W_{\mathcal{O}_f}$ represent the length and width respectively of the sheet required to produce entry f in the set of all boxes ordered from *Clickabox*.

It is assumed, for the purposes of this study, that the number of boards of each cardboard type will be chosen as the previous values, that is, 28 AC board types and 18 DWB board types. This is done primarily to allow comparisons between the previous and suggested stock board profiles. Other assumptions made in the process followed to determine a suggested stock profile are that the boards are always available, and that the reuse of off–cuts is ignored.

The heuristic process followed to determine a suggested stock profile, is outlined below.

[5]This is calculated by dividing the number of historical order dimensions for each cardboard type by the number of boards of that cardboard type to be kept in stock.

[6]This was implemented by means of programs written by the author in Microsoft Access and Visual Basic, and the relevant source code is given in §C.1 of Appendix C.

1. A two–dimensional grid was created, with the horizontal axis representing the length of potential stock boards and the vertical axis representing width. The range of the horizontal axis was defined as [0, 2 490]mm, with values increasing in steps of 10mm, as discussed previously. Values on the vertical axis increase from 0mm to 2 370mm, also in steps of 10mm. This grid is represented by the set $\mathcal{G}$ of grid points $(L_{\mathcal{G}_j}, W_{\mathcal{G}_j})$, where $L_{\mathcal{G}_j}$ and $W_{\mathcal{G}_j}$ represent the length and width respectively of the j–th potential stock boards. Note that the directional properties of the cardboard do not allow for board orientation to be changed.

2. The constraint on the minimum number of historical orders to be filled by each stock board is calculated as described above in §4.2.3.

3. The following procedure was then followed for each data point $(L_{\mathcal{G}_j}, W_{\mathcal{G}_j})$ on the grid, and for each point $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$ in the set of demand points:

   (a) The grid point $(L_{\mathcal{G}_j}, W_{\mathcal{G}_j})$ and order $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$ were compared for each combination of $j \in \{1, \ldots, 46\}$ and $f \in \{1, \ldots, 3\,965\}$. If $L_{\mathcal{O}_f} > L_{\mathcal{G}_j}$ or $W_{\mathcal{O}_f} > W_{\mathcal{G}_j}$ then the box ordered could not be produced from the board investigated at the gridpoint, and a variable is incremented to store this information.

   (b) If $L_{\mathcal{O}_f} \leq L_{\mathcal{G}_j}$ and $W_{\mathcal{O}_f} \leq W_{\mathcal{G}_j}$, the following calculations were made: the number of sheets that could be produced lengthwise out of the board size being investigated was calculated as $\lfloor L_{\mathcal{G}_j}/L_{\mathcal{O}_f} \rfloor$, and similarly the number of sheets that could be made widthwise out of the board size being investigated was calculated as $\lfloor W_{\mathcal{G}_j}/W_{\mathcal{O}_f} \rfloor$. The factor $\overline{m}^{(j,f)}$, the total number of sheets that could be produced from a board of this size, was determined as the product of these factors, $\overline{m}^{(j,f)} = \lfloor L_{\mathcal{G}_j}/L_{\mathcal{O}_f} \rfloor \times \lfloor W_{\mathcal{G}_j}/W_{\mathcal{O}_f} \rfloor$.

   (c) Now the wastage $g_{f,j}$, when $\overline{m}^{(j,f)}$ sheets of dimensions $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$ were to be produced from boards of dimension $(L_{\mathcal{G}_j}, W_{\mathcal{G}_j})$, was calculated as

   $$g_{f,j} = L_{\mathcal{G}_j} \times W_{\mathcal{G}_j} - (\overline{m}^{(j,f)} \times L_{\mathcal{O}_f} \times W_{\mathcal{O}_f}).$$

   Total wastage is then given by wastage per board multiplied by the quantity ordered, q, *i.e.* $g_{f,j} \times q$.

Steps (1) and (3) are illustrated graphically in Figure 4.5.

In the figure the point $(L_{\mathcal{G}_j}, W_{\mathcal{G}_j})$ is the point on the grid under investigation, and point $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$ represents one of the boards in the set of orders, $\mathcal{O}$. Note that $L_{\mathcal{O}_f}$ and $W_{\mathcal{O}_f}$ may take on any integer value, whilst $L_{\mathcal{G}_j}$ and $W_{\mathcal{G}_j}$ are multiples of 10, and thus the point $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$ will not necessarily lie on the grid.

As illustrated by the dotted lines, the factors for this point are as follows: $\lfloor L_{\mathcal{G}_j}/L_{\mathcal{O}_f} \rfloor$ = 2, $\lfloor W_{\mathcal{G}_j}/W_{\mathcal{O}_f} \rfloor = 2$, and $\overline{m}^{(j,f)} = 2 \times 2 = 4$. In other words, two sheets may be cut out of the board lengthwise, and two sheets widthwise; in total four sheets of size $(L_{\mathcal{O}_f}, W_{\mathcal{O}_f})$ may therefore be produced from board $(L_{\mathcal{G}_j}, W_{\mathcal{G}_j})$. The shaded area represents the wastage per board, $\left(L_{\mathcal{G}_j} \times W_{\mathcal{G}_j} - \left(\overline{m}^{(j,f)} \times L_{\mathcal{O}_f} \times W_{\mathcal{O}_f}\right)\right)$, which is then multiplied by the number of boards required to obtain the total area of wastage. The number of boards required to fill an order is obtained by dividing the number of sheets ordered by the board factor $\overline{m}^{(j,f)}$.

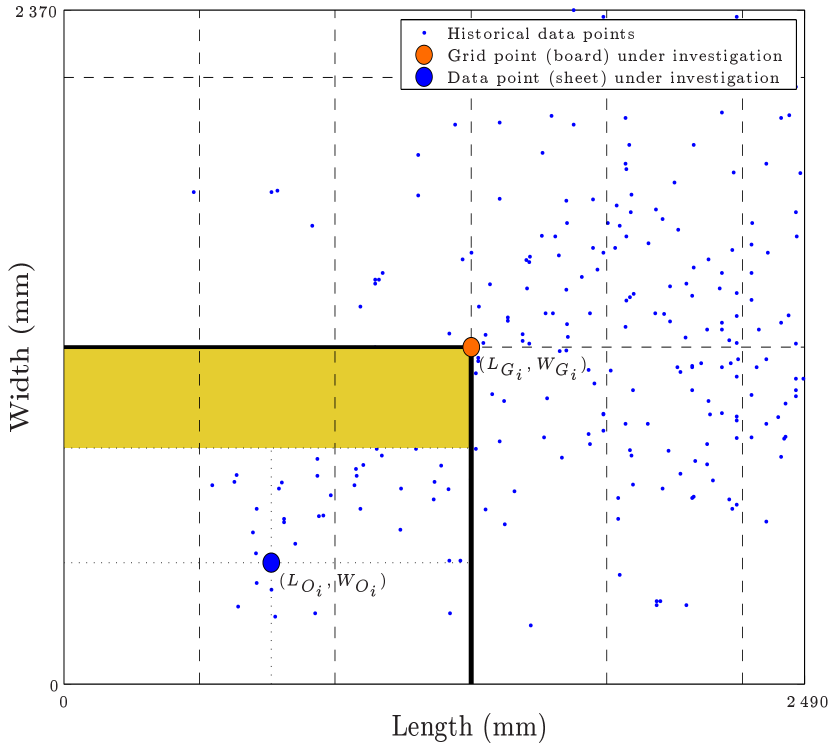


Figure 4.5: *Illustration of the process followed to find the suggested stock boards, where each point on a grid is analysed in terms of the wastage incurred in producing each order within its range.*

4. The total wastage for each grid point was then summed over all demand points, and this value was stored in the database.

5. The board size with the lowest total wastage, which satisfied the constraint (from step (2)) of the minimum number of orders to be met, was then selected.

6. A subset, $\mathcal{O}^1$, of the demand data set $\mathcal{O}$ was then created as follows:

   (a) For each demand data point, if the sheet ordered could not be produced from the "optimal board" selected, (the check for validity was conducted as in Step 3(a) above), the length, width and quantity demanded were inserted into the new demand set $\mathcal{O}^1$ (as this order had to be satisfied by another stock board.)

   (b) Wastage was calculated for each demand data point, but unlike in Step 3(c), where total wastage was calculated, here the percentage waste,

   $$g'_{f,j} = 100 \times (L_{\mathcal{G}_j} \times W_{\mathcal{G}_j} - (\overline{m}^{(j,f)} \times L_{\mathcal{O}_f} \times W_{\mathcal{O}_f}))/(\overline{m}^{(j,f)} \times L_{\mathcal{O}_f} \times W_{\mathcal{O}_f}),$$

   was considered. If this wastage was more than the threshold value of 15% (as dictated by the management of *Clickabox* in thesis objective I), the order was also inserted into the new demand set $\mathcal{O}^1$.

7. Steps (2) to (6) above were then re–executed on the new set $\mathcal{O}^1$, to find the second board to be kept in stock, and in the process creating a new demand set $\mathcal{O}^2$. This

process was repeated until the specified number of stock boards were found, or all the orders in the data set were satisfied.

4.2.5 Results

The process detailed in §4.2.4 was conducted on the data available, from the period 1 February 2001 — 31 January 2003. The analysis described in §4.2.4 was first conducted to find the 28 suggested AC boards to keep in inventory. The resultant set of AC boards, as well as the previous stock boards and all orders placed at *Clickabox* during the two year period for which the analysis was conducted, are shown graphically in Figure 4.6.

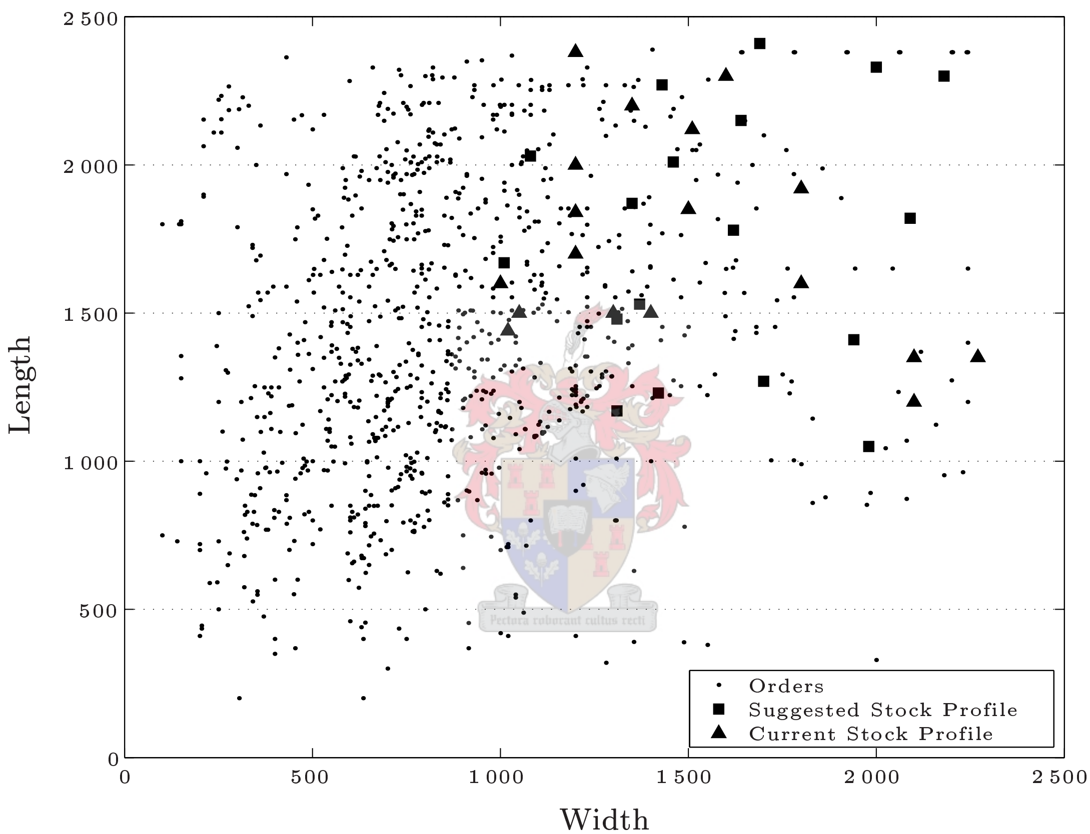


Figure 4.6: *Depiction of previous and proposed AC stock boards profile against sheet orders received.*

The dimensions of the previous and suggested stock profiles are given in Table 4.2, along with the average percentage of (retrospective) wastage per board, and the percentage of total orders during the two year period met optimally by each board.

The performance of the suggested stock profile was compared to that of the previous stock profile kept in inventory, with respect to the historical data available. This was achieved by means of a computerised procedure which calculated the optimal board to be used for each sheet order during the period 31 January 2001 — 1 February 2003, and then stored data as to the percentage of wastage incurred for that order[7]. In other words, assuming optimal stock boards were always available, various statistics, such as average and total wastage, were calculated for both the previous and the suggested stock. It is important to note that the figures do not give the actual performance of the previous stock boards, but rather an optimistic measure of performance, as in reality there was not always sufficient optimal stock boards available, and sub–optimal boards had to be used frequently.

Length (mm)	Width (mm)	Avg. % Waste	% Orders Met
1 000	1 400	4.06%	0.66%
1 220	21 00	25.92%	5.01%
1 250	1 300	10.86%	2.95%
1 360	2 100	11.86%	5.94%
1 400	1 300	10.85%	3.30%
1 425	1 125	11.87%	3.38%
1 500	1 500	18.19%	6.87%
1 530	1 300	14.01%	5.48%
1 540	910	17.50%	6.72%
1 550	1 400	10.20%	2.72%
1 555	1 540	13.37%	2.52%
1 600	1 300	5.77%	0.50%
1 650	1 500	12.49%	2.29%
1 700	1 000	16.64%	2.8%
1 700	1 200	8.37%	3.18%
1 710	1 500	9.87%	2.33%
1 850	1 510	10.56%	2.72%
1 860	1 200	10.44%	2.56%
1 870	1 055	12.58%	3.11%
1 920	910	15.39%	5.01%
2 000	1 200	12.98%	3.42%
2 000	1 500	8.88%	1.59%
2 100	1 000	18.36%	4.82%
2 200	1 200	10.47%	2.14%
2 200	1 500	11.09%	3.07%
2 300	1 000	17.20%	5.01%
2 300	1 200	5.93%	1.17%
2 358	1 062	12.33%	5.59%

(a) Current AC Stock Boards

Index	Length (mm)	Width (mm)	Rank	Avg. % Waste	% Orders Met
1	1 030	2 370	B	12.73%	1.59%
2	1 260	2 300	A	15.30%	5.79%
3	1 280	1 300	B	9.57%	2.45%
4	1 330	2 370	B	5.10%	0.78%
5	1 360	2 300	A	12.93%	5.32%
6	1 380	1 310	B	7.44%	1.40%
7	1 460	2 370	A	13.71%	3.65%
8	1 470	1 480	A	12.27%	2.76%
9	1 500	1 540	B	14.39%	1.20%
10	1 510	1 810	A	18.58%	9.32%
11	1 530	1 380	A	12.67%	5.36%
12	1 550	1 020	B	10.02%	1.55%
13	1 680	1 080	A	12.28%	3.69%
14	1 720	1 210	A	12.78%	4.89%
15	1 800	1 200	B	10.45%	1.55%
16	1 860	1 000	B	8.54%	1.09%
17	1 860	1 490	A	15.96%	8.08%
18	1 910	1 880	A	14.92%	5.09%
19	2 000	1 400	B	6.21%	1.48%
20	2 030	1 240	A	12.47%	3.15%
21	2 110	1 010	A	11.85%	3.30%
22	2 110	1 680	A	17.04%	7.15%
23	2 200	1 200	B	8.79%	1.28%
24	2 260	1 520	A	16.48%	4.50%
25	2 260	2 160	A	13.98%	6.99%
26	2 300	1 220	B	11.30%	2.17%
27	2 300	1 710	A	10.13%	3.42%
28	2 370	1 250	B	10.38%	0.82%

(b) Suggested AC Stock Boards

Table 4.2: *The dimensions of the previous and the proposed AC board stock profiles, indicating the average wastage per board and percentage of orders optimally met by each board. Also given is the rank of each stock board, an indication of the predicted frequency of use of each of the stock boards (corresponding to the percentage of historical orders met optimally with each board).*

The average wastage for each order (over all AC stock boards) was 12.43% for the previous stock profile, and 12.09% for the suggested stock profile. The suggested stock profile also performs slightly better in terms of total wastage for each order, where the wastage per board is weighted by the quantity of boards in the order, with an average of 11.98% wastage compared to the 12.18% wastage using the previous stock profile. The suggested

[7]This procedure was implemented by means of programs written by the author in Microsoft Access and Visual Basic, and the relevant source code is given in §C.3 of Appendix C.

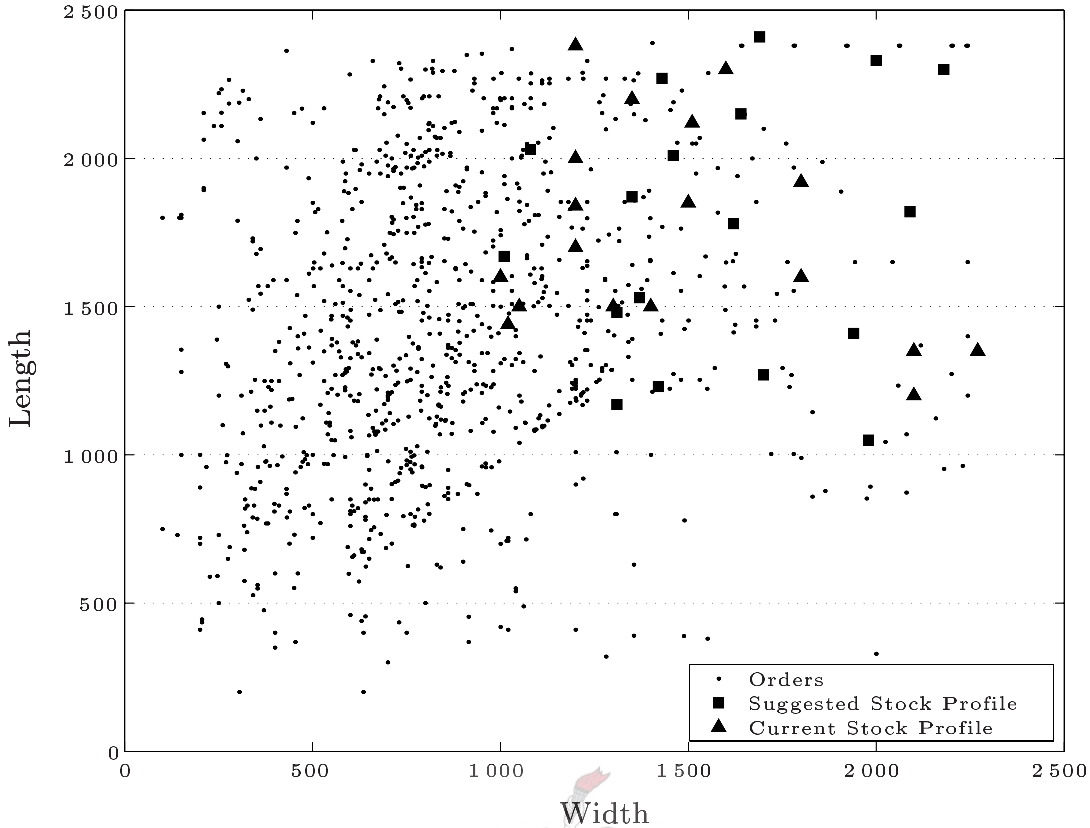


Figure 4.7: *Depiction of previous and proposed DWB board stock profile against sheet orders received.*

stock profile affects a reduction in total retrospective waste over the previous stock profile, during the two year period 31 January 2001 — 1 February 2003, of 9.37%.

The suggested stock boards satisfied 99.81% of the historical demand, compared to the 96.85% attained by the previous stock profile. In terms of the percentage of total orders met with less than the threshold wastage (15%, as defined in thesis objective I), the suggested stock profile again performed better, with 65.67% compared to the 59.52% of the previous stock profile.

Also given in Table 4.2 is the ranking of each proposed stock board, which will be used in Chapter 5 in the allocation of storage bays. These rankings are determined by the percentage of historical orders met by each stock board. All stock boards which met more than a threshold percentage of historical orders optimally were classified as rank A boards, and the remainder as rank B boards. This threshold percentage was set to 2.5% for the AC boards, and 5% for the DWB boards, for reasons that will be explained later. The rankings estimate, given the previous sheet order data, which stock boards are likely to be used most frequently for sheet orders in the future, if the past data is any indication of future demand.

A similar analysis was then conducted to find the 18 optimal DWB boards to keep in inventory. The set of suggested DWB stock boards emanating from the analysis, as well as the previous stock boards and all orders placed on *Clickabox* during the two year period for which the analysis was conducted, are shown graphically in Figure 4.7.

The dimensions of the previous and suggested stock profiles for the DWB cardboard type are given in Table 4.3, along with the average percentage of (retrospective) wastage per board, and the percentage of total orders in the two year period met optimally by each board.

Results in terms of average wastage per board were again similar. The average wastage for each order (over all DWB stock boards) was 14.59% for the previous stock profile, and 14.01% for the suggested stock profile. The average wastage weighted by order quantity was 13.62% for the previous stock profile and only 12.65% for the suggested stock profile. The suggested stock profile affects a reduction in total retrospective waste over the previous stock profile, during the two year period 31 January 2001 — 1 February 2003, of 16.17%.

The suggested DWB stock boards satisfied 98.71% of the historical demand, compared to the 97.95% attained by the previous stock profile. The suggested stock profile also performed far better in terms of the percentage of total orders met with less than the threshold wastage, with 57.31% compared to the 47.13% of the previous stock profile.

Length	Width	Avg. % Waste	% Orders Met
1 200	2 100	19.24%	5.43%
1 350	2 100	22.09%	5.31%
1 350	2 270	10.85%	1.55%
1 440	1 020	5.84%	2.63%
1 500	1 050	9.24%	1.25%
1 500	1 300	16.01%	10.69%
1 500	1 400	13.57%	2.81%
1 600	1 000	7.09%	0.66%
1 600	1 800	14.93%	4.54%
1 700	1 200	14.68%	6.57%
1 840	1 200	12.98%	3.40%
1 850	1 500	21.50%	13.31%
1 920	1 800	16.51%	6.15%
2 000	1 200	15.52%	6.63%
2 120	1 510	11.38%	4.12%
2 200	1 350	10.88%	3.58%
2 300	1 600	15.86%	8.66%
2 380	1 200	24.52%	12.72%

(a) Current DWB Stock Boards

Index	Length	Width	Rank	Avg. % Waste	% Orders Met
29	1 480	1 310	A	14.4%	9.42%
30	1 780	1 620	A	16.23%	7.31%
31	1 820	2 090	A	16.75%	5.67%
32	1 050	1 980	B	12.50%	1.52%
33	1 230	1 420	B	12.72%	4.15%
34	1 270	1 700	B	11.83%	2.11%
35	1 170	1 310	B	13.06%	3.10%
36	1 410	1 940	A	18.42%	6.37%
37	1 530	1 370	B	12.93%	2.63%
38	1 670	1 010	A	15.48%	5.67%
39	2 010	1 460	A	14.33%	5.26%
40	2 150	1 640	A	12.97%	7.66%
41	2 030	1 080	B	12.86%	5.03%
42	2 330	2 000	B	12.5%	4.56%
43	1 870	1 350	A	18.49%	11.70%
44	2 270	1 430	A	12.71%	5.03%
45	2 410	1 690	A	15.22%	6.78%
46	2 300	2 180	B	8.77%	4.74%

(b) Suggested DWB Stock Boards

Table 4.3*: The dimensions of the previous and the proposed DWB board stock profiles, indicating the average wastage per board and percentage of orders optimally met by each board. Also given is the rank of each stock board, an indication of the predicted frequency of use of each of the stock boards (corresponding to the percentage of historical orders met optimally with each board).*

4.2.6 Evaluation of Results

The results in §4.2.5 were presented to the director of *Clickabox* during a meeting in February 2003 [70]. The proposed changes in stock profile were accepted by the director, and have been implemented since March 2003. It was noted by the director that many of the changes proposed involved increasing the width of boards kept. This seemed reasonable as previously strict constraints on the width of boards kept in inventory, imposed by the machinery available at the time, have been relaxed with the procurement of new equipment. However, since these new acquisitions the stock profile had not been revised.

The suggested stock boards were entered into the Pastel system, with zero inventory levels. It was noted over the next month (February 2003) that for most orders placed the new stock boards were chosen by the quote program, above the previous stock, as the optimal boards with which to produce the boxes ordered. The new boards in Tables 4.2(b) and 4.3(b) were then phased into stock during March 2003.

4.3 Demand Distributions

In this section, the concept of a *board preference vector* is introduced, in order to incorporate the cascading product substitution that occurs at *Clickabox* into the modelling of the demand process. This approach is discussed, and the board preference vectors are determined in §4.3.1. An analysis is then conducted of the demand for each of these board preference vectors in §4.3.2. The method of Treharne and Sox [73], outlined in Chapter 2, is further explained, and the modifications made to their approach in this thesis are discussed. The demand classes are then established, and the probability distributions of the demand realisation processes are derived in §4.3.2.1 to §4.3.2.3. The Markovian transition probabilities that govern the transition between demand states are calculated, in §4.3.2.4, for each board preference vector. Finally, in §4.3.2.5, the sheet–to–board conversion factors are discussed.

4.3.1 Board Preference Vectors

Out of the 4 285 works tickets containing data of orders placed for AC or DWB boards during the period 1 February 2001 to 31 January 2003, there were 2 461 different sheet sizes ordered. This indicates that, for many of the sheet sizes ordered, there were only one or two orders placed during the period in question. Demand cannot, therefore, be sensibly modelled by considering the probability distributions of specific *sheet* sizes ordered. However, it is also not viable to simply derive a probability distribution of demand for the stock boards from the historical data of *boards* actually used, as the ideal board would not always have been available.

The approach taken in this thesis was therefore the following. For each sheet order, a vector of the stock boards in Tables 4.2(b) and 4.3(b) that may be used to produce that sheet was constructed, and sorted in order of decreasing wastage[8]. Wastage was calculated as the difference between the area of the board used and the total board area, as a fraction of the total board area. This approach resulted in 526 distinct vectors when applied to the demand data during the period 1 February 2001 to 31 January 2003. The average wastage was calculated for all orders during the two year period for which data were available, under the assumption that enough of the board associated with the least wastage was, in fact, always available. This average wastage was 13.80%. The average wastage, always using the second best board, was 19.84%, whilst that, always using the third best board, was 25.40%. Finally, the average wastage resulting from always using the fourth best board was 28.81%.

Very few sheets (only 1.44%) could be made out of the fourth best board with a wastage of less than 15%, which is the maximum acceptable percentage of wastage set by the management of *Clickabox* (see thesis objective I in §1.2). It was therefore decided that each sheet order would be associated with a vector of only three stock boards, arranged in order of decreasing wastage. This vector is termed a *board preference vector* for the sheet in question. If there is sufficient stock of the board associated with the lowest wastage (the first entry in the board preference vector), this board is to be used to produce sheets ordered. If not, the second, and then third best board is to be used. Should there be insufficent stock of any of the three board options available to produce the order, a high cascading shortage cost is incurred. The restriction on the number of boards in each board preference vector further reduced the number of distributions that had to be computed from 526 to 386. Two examples of board preference vectors are given in Table 4.4. A complete list of board preference vectors is given in Appendix E.

Sheet Index	Board Type	Board 1	Board 2	Board 3
1	AC	$1\,030 \times 2\,370$	$2\,030 \times 1\,240$	$2\,200 \times 1\,200$
29	AC	$1\,460 \times 2\,370$	$1\,380 \times 1\,310$	$1\,720 \times 1\,210$

Table 4.4: *The board preference vectors of two of the popular sheet dimensions.*

The notation used in conjunction with the board preference vectors in the modelling process is now formalised. Suppose that there are s possible sheet types that may be ordered, and let $\mathcal{S} = \{1, \ldots, s\}$ index these sheet types. Suppose further that there are b different board types that are kept in inventory, and let $\mathcal{B} = \{1, \ldots, b\}$ index these board types. Therefore $b = 46$, according to Tables 4.2(b) and 4.3(b). Each sheet may be produced from a number of boards, with varying amounts of wastage associated with each board type chosen to fulfil an order for a specific sheet type. Suppose that a maximum of $m^{(i,\beta)}$ sheets of type $i \in \mathcal{S}$ may be produced from board type $\beta \in \mathcal{B}$.

Now for a sheet $i \in \mathcal{S}$ ordered, a board preference vector, denoted $\underline{v}_i$, is defined, containing the indices of those board types from which that sheet may be produced. As discussed, these board preference vectors are limited to containing three board types. For example, $v_{i,k} = \beta$ indicates that, for the manufacturing of sheet i, the k–th best board to use is board β, where $i \in \mathcal{S}, \beta \in \mathcal{B}$ and $k \in \mathcal{X} := \{1, 2, 3\}$. The board preference vectors

[8]This was implemented by means of programs written by the author in Microsoft Access and Visual Basic, and the relevant source code is given in §C.3 of Appendix C.

$\underline{v}_1, \ldots, \underline{v}_s$ are now relabelled as $\underline{v}_1, \ldots, \underline{v}_\mu$, where $\mu \leq s$, by eliminating duplicates of the same set of board preference vector entries (*i.e.* by avoiding vector multiplicities). Let the set $\mathcal{V} = \{1, \ldots, \mu\}$ index this new numbering of the board preference vectors.

4.3.2 Board Demands

As discussed in §2.2.1, the Markovian demand modelling approach to be followed in this thesis is based on that of Treharne and Sox [73]. In their approach to adaptive inventory control techniques, they defined two core processes. The first was the *demand realisation process*, governed by a set of conditional probabilities that give the probability of a demand occuring within a certain *demand class* during the current time period, given that the system is currently in a certain *demand state*. The demand classes represent ranges of quantities in which demand may occur, and the demand states represent the various distributions which may be assumed by the demand process. The second was the *Markov decision process*, governed by a set of transition probabilities that determine the transition between demand states.

The method followed in [73] was modified as follows with respect to the case study at *Clickabox*. The notions of demand classes and demand states were merged, and a number of classes were formulated. The Markov process then determines the transition between these classes, which are ranges of order quantities which may materialise during a time period. The demand realisation process determines, given the current demand class, which quantity within that class will occur.

This modification is motivated by an analysis of the quantities of each board preference vector ordered during each week in §4.3.2.1. In this subsection the clumpy nature of the data is shown to prohibit direct use of any standard theoretical distributions to model the demand. As a result, empirical distributions had to be derived from the historical data. The clumpy nature of the data also suggest a grouping of potential demand realisations around certain common order quantities. The demand classes are derived in §4.3.2.2, and then the distribution to model demand realisation within a class is derived in §4.3.2.3. The Markov transition probabilities are determined in §4.3.2.4, and finally the modelling of the sheet–to–board conversion factors is discussed in §4.3.2.5.

4.3.2.1 Probability Distributions

The demand data of the board preference vectors were analysed in *Arena Input Analyzer* [60]. The *Arena Input Analyzer* tests the data against the following theoretical distributions: Beta, Erlang, Exponential, Gamma, Johnson, Lognormal, Normal, Poisson, Triangular, Uniform and Weibull. It establishes the minimum square error distribution, in other words the distribution that has the smallest sum of squared discrepancies between the histogram frequency and the frequency of the fitted data.

The goodness–of–fit of the minimum square error distribution is then assessed by means of hypothesis testing. The null hypothesis is that the fitted distribution adequately represents the data. The results are reported in the form of a so–called p–value, which is the smallest level of significance at which the null hypothesis is rejected. A larger p–value represents a better fit. Two goodness–of–fit tests are conducted, namely the Chi–square test and the Kolmogorov–Smirnov test. The minimum square error distributions, as well as the results of both goodness–of–fit tests, for three of the board preference vectors are given, as examples, in Table 4.5. It was found, for all board preference vectors tested, that no standard theoretical demand distribution fitted the data at a satisfactory level of significance, and that the test results became worse as data became more sparse.

	Board Pref. Vector 1	Board Pref. Vector 2	Board Pref. Vector 3
Best Distribution	Exponential	Weibull	Weibull
Expression	EXPO(136)	WEIB(0.0373, 0.176)	WEIB(11.9, 0.187)
Square Error	0.006 904	0.005 367	0.015 869
Chi Square Test			
Number of intervals	1	1	2
Degrees of freedom	-1	-2	-1
Test Statistic	0.558	0.304	37.3
Corresponding p-value	< 0.005	< 0.005	< 0.005
Kolmogorov-Smirnov Test			
Test Statistic	0.581	0.411	0.159
Corresponding p-value	< 0.01	< 0.01	< 0.01

Table 4.5: *The results of the distribution analysis, showing the minimum square error distributions, as well as the outcome of the goodness–of–fit tests for three board preference vectors.*

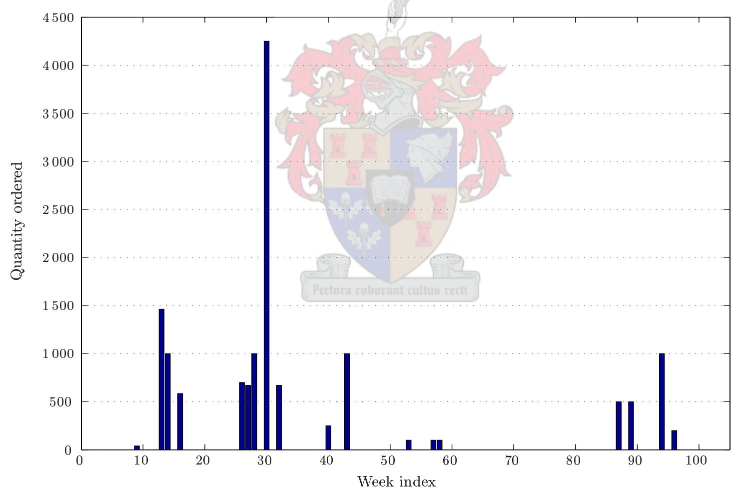


Figure 4.8: *The quantities ordered over a two year period of board preference vector 1, showing how demand quantities tend to occur most frequently for certain values such as 100, 500 and 1 000. Here week 0 corresponds to the period 1 — 7 February 2001, week 1 to the period 8 — 14 February, and so on, until week 103, which corresponds to the period 25 — 31 January 2003.*

The unsuitability of any of the above–mentioned standard theoretical distributions to fit the data is attributed to the clumpy nature of the quantities ordered. The quantities ordered during the two year period 31 January 2001 — 1 February 2003 of the two board preference vectors given in Table 4.4 are shown in Figure 4.8 and Figure 4.9 as an illustration of this phenomenon. It was therefore necessary to model the demand by means

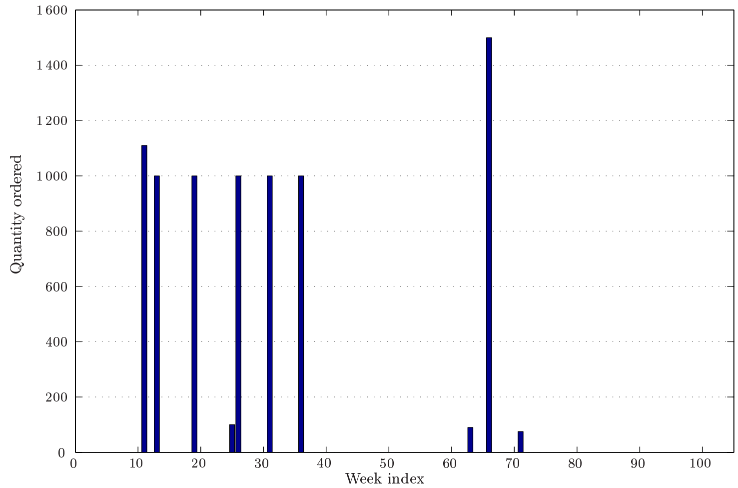


Figure 4.9: *The quantities ordered over a two year period of board preference vector 29, showing how demand quantities tend to occur most frequently for certain values such as 100 and 1 000. Here week 0 corresponds to the period 1 — 7 February 2001, week 1 to the period 8 — 14 February, and so on, until week 103, which corresponds to the period 25 — 31 January 2003.*

of empirical distributions. The first step in the formulation of empirical distributions for week demand is to establish the demand classes on which the distributions will be based.

4.3.2.2 Derivation of Demand Classes

Time period demand for board preference vectors is divided into a number of classes, indexed by the set $\mathcal{K} = \{0, \ldots, k\}$. For example, $k = 0$ may represent a weekly demand of orders of between 0 and 100 sheets, which may be produced by boards from a given board preference vector.

The demand classes in $\mathcal{K}$ were established as follows. The frequency of occurrence of each potential quantity of weekly demand was established from the historical data. The k most frequently ordered values were taken to each represent a demand class, and the demand classes were chosen so that each value was the mode (the most frequently occurring value) of its demand class. The number of demand classes (k) was taken to be the minimum number of classes for which there is a probability of at least 95% that the realised demand quantity will be one of these values, if past data is an indication of future demand.

As shown in Table 4.6, the quantities in the set $\{0, 50, 100, 200, 250, 500, 1\,000\}$ occur in more than 95% of the instances for AC board types, so the value of k for AC board types is taken as seven. The DWB board types have a slightly wider spread of quantities, the set of quantities required to meet the above-mentioned specification contains the seven quantities given for AC boards, as well as the quantities $\{1, 10, 20, 30\}$. However, an increase from seven to eleven demand classes would translate to an increase from 7 497 to 18 513 entries in the transition probability matrix of the resulting Markov process. So in order to ensure manageability and efficiency of the model, it was decided not to include

Quantity	Frequency	Cumulative Probability of Occurrence
0	22 098	91.59%
500	199	92.41%
100	148	93.02%
1 000	147	93.63%
50	136	94.20%
200	106	94.64%
250	88	95.01%
1	87	95.36%
20	57	95.60%
30	48	95.80%
150	44	95.98%
60	44	96.16%
⋮	⋮	⋮

(a) AC Boards

Quantity	Frequency	Cumulative Probability of Occurrence
0	14282	89.76%
50	182	90.90%
1	128	91.70%
100	113	92.41%
500	94	93.01%
10	73	93.46%
20	57	93.82%
1000	55	94.17%
250	52	94.49%
200	51	94.82%
30	47	95.11%
40	37	95.34%
⋮	⋮	⋮

(b) DWB Boards

Table 4.6: *The frequency of occurrence of order quantities (number of sheets ordered per week) for each optimal board type, showing the cumulative probability of occurrence, calculated using the ratio of the frequency of occurrence to the total number of occurrences. The table is truncated at the twelfth demand quantity.*

these additional categories for the DWB board types separately, but instead to use the seven categories determined for the AC board types, into which these four quantities are incorporated in any case. This decision is justified by the results of the ABC classification in §4.2.1, which showed the AC board types to be of far more importance in decision making with regards to inventory control than DWB boards. Furthermore, as will be described later, the demand realisation within a class is represented by a probability distribution over the values in the class, so the information pertaining to the frequency of occurence of the quantities $\{1, 10, 20, 30\}$ is not lost as a result of their ommission as individual classes.

The demand class divisions had to be formulated next, in such a way that the values in the set each form the mode of a class. As the demand classes are required to span all integer order quantities, the classes are formed as being intervals between the midpoints from one mode to the next, with the exception of zero, which forms a class of its own. The classes thus formed are given in Table 4.7, along with the mode, median (value at the centre of the set) and mean (average) of each class. Figure 4.10 gives a graphical depiction of these summary statistics for both board types.

Note that there are only two significant discrepancies amongst the values of the summary statistics for each demand class in Table 4.7. The first is for the class $1-75$, where the mean and median are approximately half of the mode for both cardboard types. The reader is referred to Table 4.6 for an explanation of this observation. Table 4.6 shows that the next three most frequently ordered quantities for the AC board types are in fact $\{1, 20, 30\}$, which is the reason that the median (and mean) of the class $1-75$ are weighted away from its mode of 50, towards the lower range of the class. Similarly for the DWB board types, the values $\{1, 10, 20, 30\}$ were shown to occur almost as frequently as

Index	Class	Mode	Mean	Median
1	0	0	0	0
2	1–75	50	28.66	25
3	76–150	100	110.38	100
4	151–225	200	195.49	200
5	226–375	250	274.92	250
6	376–750	500	525.70	500
7	751+	1 000	1 826.50	1 050

(a) AC Board Types

Index	Class	Mode	Mean	Median
1	0	0	0	0
2	1–75	50	26.81	20
3	76–150	100	108.93	100
4	151–225	200	193.04	200
5	226–375	250	276.31	260
6	376–750	500	510.79	500
7	751+	1 000	1 410.91	1 000

(b) DWB Board Types

Table 4.7: *Summary statistics of the demand classes (for each cardboard type), giving the range, mean, mode and median of each class.*

the value 50, and as mentioned above, this is accounted for in the values of the mean and median.

The second discrepancy is for the class 751+, where the mean is significantly higher than the median or mode for both cardboard types. This is a result of the few very high quantities that occur, albeit infrequently, which weigh the value of the mean away from the median and mode.

4.3.2.3 Derivation of Demand Realisation Distributions

The weekly order quantities in each demand class within the set $\mathcal{K} = \{1, \ldots, 7\}$, for each board preference vector, were investigated next, in order to establish a distribution for this demand. Again, a sample of the board preference vectors were analysed. The normal distribution was found to fit in the majority of the cases tested, which was to be expected, considering the closeness of the mean, median and mode, and the method by which the classes were establised. The exceptions to this observation were cases in which data was too sparse for the fitting of a distribution. However, in these cases the only quantity that occured was, in fact, the mean of the class. The demand realisations in each class are therefore characterised by normal distributions about the mean of the class.

The width of the normal distribution for each class was chosen so that 95% of the values represented by the normal distribution fall within the range of that class. The value of the standard deviation, σ, is therefore chosen so that the range spanned by the class is 4σ [54]. The resulting distributions for classes 2 to 6 are shown schematically in Figure 4.11. The distribution for class 7 is shown separately, in Figure 4.12, as it has a much higher standard deviation, and hence does not display well on the same scale as the other distributions. Note that class 1 is not represented by a distribution, because demand realisations in this class always assume a value of zero.

4.3.2.4 Derivation of Transition Probabilities

Based on the demand classes established in §4.3.2.2, the following analysis was conducted for each board preference vector, based on the historical demand data available[9]. The number of transitions from each demand class $j \in \mathcal{K}$ to each other demand class $m \in \mathcal{K}$ was calculated. This was expressed as a fraction of the total number of incidents of the demand in class j. These are conditional probabilities (with the condition being that demand class m is the current one realised) that quantify the likelihood that demand class j will be the next one realised. The conditional probabilities therefore determine the transition between states in the Markov demand process. These computations were conducted for each initial demand class j of each of the board preference vectors. The transition probabilities for one of the board preference vectors is given, as an example, in Table 4.8.

Demand State		To						
		$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$
	$m=1$	0.49	0.11	0.04	0.09	0.04	0.09	0.13
F	$m=2$	0.38	0.63	0.13	0.13	0.00	0.63	0.25
r	$m=3$	0.50	0.00	0.00	0.00	0.00	0.00	0.00
o	$m=4$	0.25	0.25	0.08	0.42	0.00	0.00	0.00
m	$m=5$	0.5	0.5	0.00	0.00	0.00	0.00	0.00
	$m=6$	0.29	0.29	0.14	0.14	0.00	0.00	0.14
	$m=7$	0.64	0.07	0.00	0.00	0.00	0.14	0.14

Table 4.8: *The transition probabilities for board preference vector 6, indicating the probability of a transition from each state to each other state.*

A complete list of the resulting transition matrices is included in a CD inside the back cover of this thesis.

4.3.2.5 Derivation of Conversion Factor Probabilities

As discussed, each sheet order is converted to an order for a board preference vector. However, in this process the information as to how many sheets can be produced from each board is lost. This number may be different for each sheet that is associated with a specific board preference vector. Hence, each order placed for a certain board preference vector may be associated with different sheet–to–board conversion factors.

One intuitive way of countering this information loss would be to expand the board preference vectors, in order to retain this information. This could be achieved by including a conversion factor for each sheet associated with the board preference vector, for each board in the vector, so that orders for the same combination of boards, but different conversion factors, are distinct entries. However, this would increase the number of board preference vectors from 386 to 781. Such an increase is not desirable, as it greatly reduces the number of historical demand entries associated with each board preference vector, thereby decreasing the accuracy of the probabilities derived, based on the historical data.

[9]This was implemented by means of programs written by the author in Microsoft Access and Visual Basic, and the relevant source code is given in §C.4 of Appendix C.

The eventual approach taken was therefore not to include the sheet–to–board conversion factors in the board preference vectors, but rather to derive a probability distribution for the sheet–to–board conversion factors. In this way, when a demand for a board preference vector is realised, the corresponding sheet–to–board conversion factor for each board in the board preference vector is derived from the probability distribution.

This was achieved, again by analysis of the historical data[10], as follows. The average sheet–to–board conversion factor per week, for each board in each board preference vector, was calculated. The averages were weighted to take into account the quantity of each order. The number of occurrences of each of the sheet–to–board conversion factor averages, for each board in each board preference vector, was then divided by the total number of demand occurrences for that board preference vector. The probability of occurrence of the first few sheet–to–board conversion factors for one of the board preference vectors is given, as an example, in Table 4.9. A complete list of the sheet–to–board conversion factor probabilities for all board preference vectors is included on a CD inside the back cover of this thesis.

The notation of the sheet–to–board conversion factor $m^{(i,\beta)}$ is amended to $m_t^{(i,\beta)}$, as the board factor now represents the probable number of sheets, optimally manufactured by board preference vector v_i, that may be produced from board β in week t.

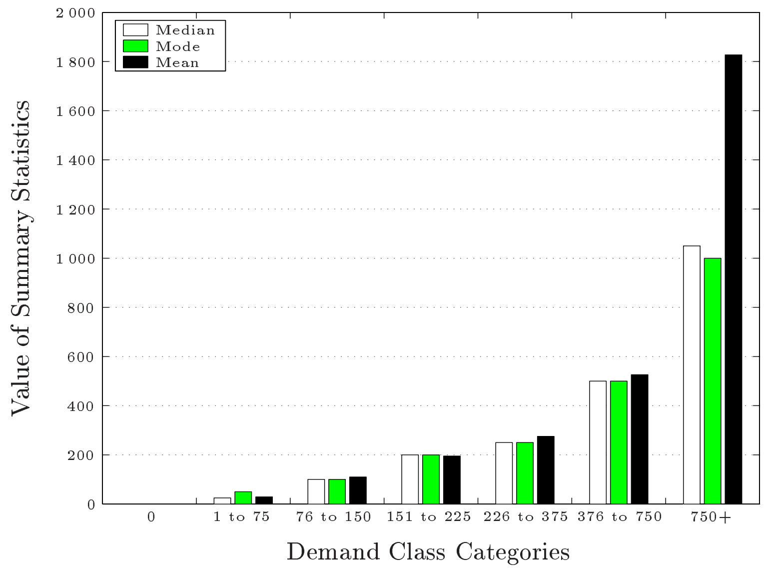
Board Index	Factor	Probability
1	4.00	0.31
1	5.33	0.03
1	6.00	0.03
1	6.26	0.03
1	6.58	0.03
1	6.67	0.03
1	7.45	0.06
1	7.57	0.03
1	8.00	0.40
2	2.00	0.96
2	4.00	0.04
3	2.00	0.03
3	4.00	0.31
3	5.33	0.03
3	6.00	0.03
3	6.26	0.03
3	6.58	0.03
3	6.67	0.03
3	7.45	0.06
3	7.57	0.03
3	8.00	0.40

Table 4.9: *The entries in the board factor matrix for board preference vector 1, giving the probability of occurrence of some potential factors for the board in each of the three positions in the board preference vector.*

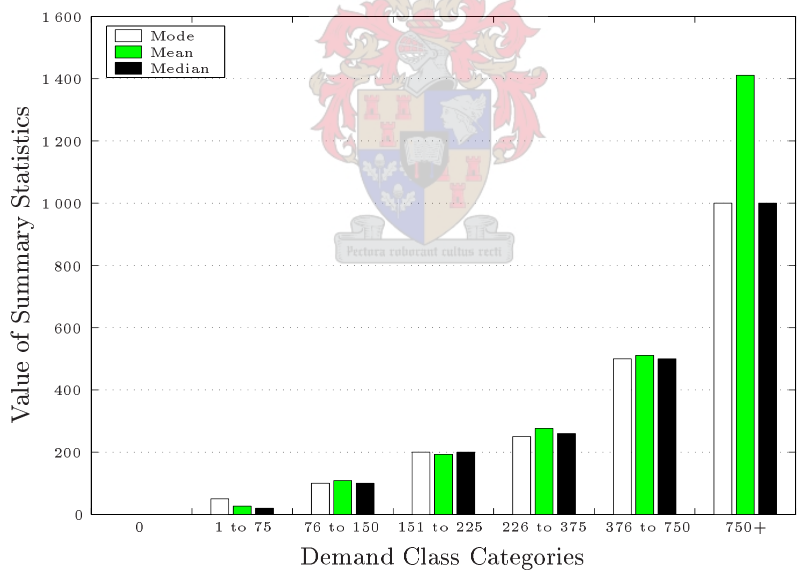
[10]This was implemented by means of programs written by the author in Microsoft Access and Visual Basic, and the relevant source code is given in §C.5 of Appendix C.

4.4 Chapter Summary

The historical data available from *Clickabox* was analysed in this chapter, in order to determine an optimal set of stock boards to be kept in inventory, and to characterise the nature of the demand process for the purposes of inventory modelling. An ABC classification was conducted in §4.2.1, the results of which showed board types AC and DWB to be the most important board types to be kept in stock, because of their high annual usage values. The heuristic used to determine a suitable set of board dimensions to be kept in inventory was described in §4.2.4, and the results were given and discussed in §4.2.5. The resulting suggested stock board profile has been accepted and implemented by the management of *Clickabox* as of March 2003. The new stock profile has proved to perform well compared to the old stock profile in terms of a reduction in wastage, as, when both stock profiles were loaded in the system, the newly suggested stock board types were consistently chosen over the old stock boards to be the optimal boards in inventory to satisfy customer orders. The concept of a board preference vector, used to incorporate the cascading product substitution that takes place at *Clickabox*, was then introduced in §4.3.1. In §4.3.2, demand classes for board preference vectors were defined, and then two core inventory modelling processes were introduced, namely the demand realisation process and the Markov decision process. Based on the historical data available, distributions were derived to model these processes. It is upon these distributions that the inventory model in the following chapter is built.



(a) AC board types



(b) DWB board types

Figure 4.10: *Graphical depiction of summary statistics for each demand class, showing the correlation between the mean, mode and median of each class.*

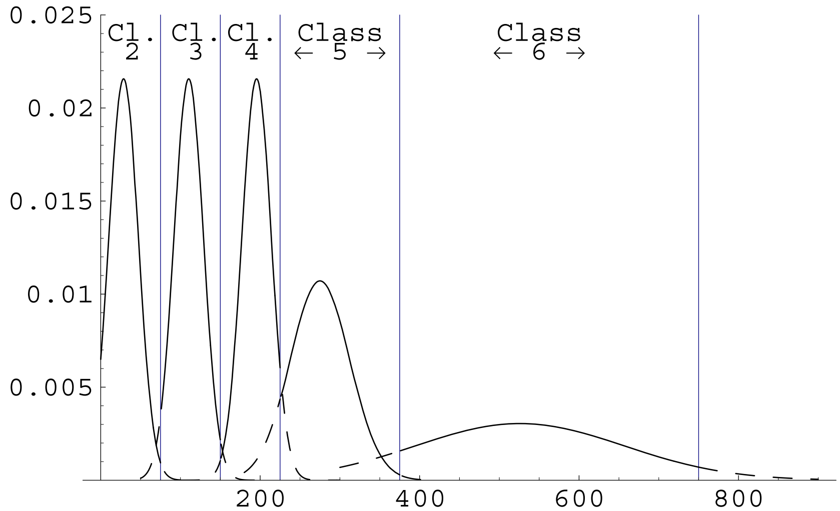


Figure 4.11: *The normal distributions of demand classes 2 to 6, indicating the range of each class. Demand realisations in class 1 assume the value of zero. The demand distribution for class 7 has a much higher standard deviation and so is illustrated separately.*

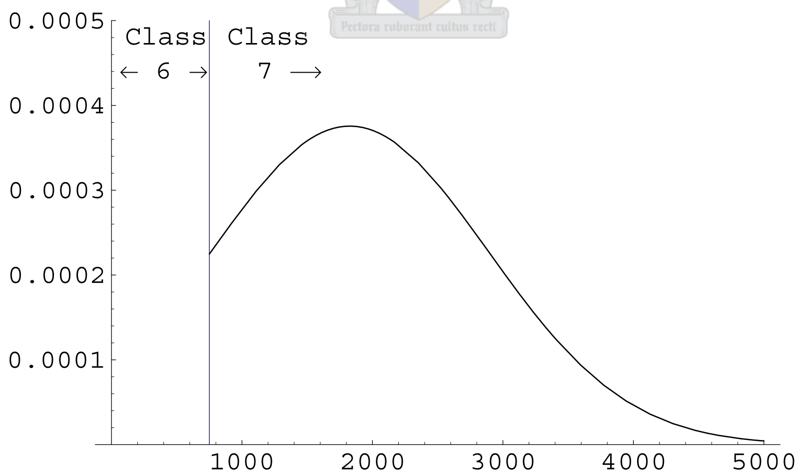


Figure 4.12: *The normal distribution of demand class 7, indicating the range of the class.*

Chapter 5

Inventory Model

The objective in this chapter is to design a strategic mathematical model in order to aid the inventory manager when replenishing stock at *Clickabox*. A *theoretical* optimal control policy is defined for the case of non–stationary, partially observed demand, modelled as a finite state Markov Chain. A sub–optimal control policy, which is more *practical* with respect to computational requirements, is then developed. The objective of this policy is the minimization of expected *tied–up inventory capital* subject to an acceptable level of *offcut wastage costs*, whilst satisfying the given service level requirements.

5.1 General Modelling Assumptions

The following modelling assumptions will be assumed to hold throughout this chapter.

1. The financial year is divided into 52 one–week time periods. These week periods are indexed by the set $\mathcal{T} = \{0, \ldots, 51\}$, with period 0 being the week 1 — 7 February, period 1 being the week 8 — 14 February and so on, until period 51, which is the week 25 — 31 January. The planning horizon for the model is one year.

2. Boards are ordered at the start of a one–week time period, and are delivered at the start of the week, the lead time number of weeks later. Lead time for delivery of boards from suppliers is constant ($l = 2$ weeks). The lead time in reality is variable, but not highly variable (typically between seven and ten working days). This assumption simplifies the modelling process.

3. Unsold boards at the end of the one–year planning horizon maintain their value for future periods.

4. Demand for any box type during any given week arises from one of a number of probability distributions. The distribution changes randomly from one week to the next. At the beginning of any given week it is not certain from which distribution the demand will be generated for that week. However, the transition probabilities for demand from one week to the next are known and fixed.

5. If the optimal board is not available to satisfy an order, the next best board is used. This is termed *stock cascading*, as discussed before. The cost of customer dissatisfaction is represented by the increased wastage cost sustained when using a sub–optimal board. The wastage cost is in reality sustained by the customer and not by *Clickabox*, but for modelling purposes will represent the cost of shortage.

6. In the case of no suitable board being available to produce an order, *i.e.* not even a substitute board is available, demand is completely backordered. As discussed in Chapter 4, this will result in a high shortage cost being incurred.

7. Raw materials are available for production as soon as they arrive at the warehouse. There are at most two raw materials deliveries to the factory per day [70]. The deliveries are offloaded directly into the warehouse and the boards are marked *available for use* as soon as the delivery is recorded into the system. Under normal circumstances this will happen immediately.

8. Income that is redeemed through the processing or recycling of offcuts is neglected. This is justified as the income from recycling is approximately R1 845 per month, compared to a turnover of R450 000 — R600 000 per month in 2002.

9. Orders are assumed to be fulfilled as soon as production is complete, *i.e.* delivery is assumed to take place on the same day that an order is ready. This is normally the case [70], the exceptions are a result of poor production planning and do not fall within the scope of this thesis.

10. No raw materials are damaged in storage. Storage damage occurs when piles are stacked too high and fall over, usually as a result of the warehouse being too full. It is assumed that, given the cost associated with holding stock included in the inventory model, the occurrence of this will be minimal. It is, in fact, estimated by management that less than 0.1% of the stock in inventory at *Clickabox* is lost due to damage in storage [70].

11. Fulfillment of orders is dependent only on availability of stock, *i.e.* machine breakdowns and labour shortages do not influence the production schedule. As in (8) above, exceptions to this assumption fall beyond the scope of this thesis.

5.2 Spatial Constraints

Spatial constraints for the inventory model will be establised as firstly the volume of storage space available, and then as an approximate number of boards that can be stored in the available space.

Available space is calculated in cubic meters. There is a restriction on the height to which boards may be stacked in the warehouse, for convenience purposes and to prevent the boards from being damaged due to pressure in the stack. The maximum height to which boards should be stacked is 3.2 metres. Allowance must also be made for loading and manoeuvering of forklifts, so 45 metres of the length of the storage area and 11 metres of its width are used for storage. Available space in the raw materials store is therefore 1 584m^3.

The approximate number of boards that can be stored in this space is now calculated. The maximum length of boards is 2.49m and the maximum width is 2.38m, due to constraints on the machinery used to process orders. Hence the available space allows for 72 bays in which to store different board dimensions, 18 lengthwise and 4 widthwise in the storeroom. The bays are divided proportionately between the AC and the DWB boards, according to the number of sizes of each cardboard type kept in stock. This results in 44 bays for the AC boards, and 28 bays for the DWB boards. Each bay holds only one type of board; the more popular board dimensions are allocated more than one bay. As discussed in Chapter 4, the stock boards are ranked according to annual usuage value. It is assumed that the 16 most popular AC board dimensions (termed the *Rank A* board types in Table 4.2) are allocated two bays each, and the remaining 12 AC board types (termed the *Rank B* board types in Table 4.2) are allocated one bay each, to make up the total of 44 bays allocated for storage of the AC board types. Similarly, the ten DWB Rank A board types are allocated two bays each, and the remaining eight DWB Rank B board types are allocated one bay each, to make up the total of 28 bays allocated for storage of the DWB board types.

AC boards are 4.4mm thick, while DWB boards are 7.6mm thick. The number of AC boards that can be stored in each bay is therefore $3.2\text{m} \div 0.004\,4\text{m} = 727$ boards, and the number of DWB boards that can be stored is $3.2\text{m} \div 0.007\,6\text{m} = 421$ boards. The constraint on the number of boards of each type that may be stored in the available space, dependent on board type and rank, is defined as $\eta^{\varpi}_{(z)}$, where ϖ is the rank (either A or B), and z is the board type (either AC or DWB). The current constraints on the number of each board type and rank that may be held in inventory are given in Table 5.1(a).

If the threshold values given in Table 5.1(a) are exceeded, the viability of renting additional storage space must be investigated.

$\eta^{A}_{(\text{AC})}$	1 454
$\eta^{B}_{(\text{AC})}$	727
$\eta^{A}_{(\text{DWB})}$	842
$\eta^{B}_{(\text{DWB})}$	421

(a) Current Position

$\eta^{A}_{(\text{AC})}$	2 182
$\eta^{B}_{(\text{AC})}$	727
$\eta^{A}_{(\text{DWB})}$	1 263
$\eta^{B}_{(\text{DWB})}$	421

(b) Upper Limit

Table 5.1: *Spatial constraint for each board type and rank, given as the number of boards, that can be stored in inventory. Table (a) gives the constraints based on the current storage position, and Table (b) gives the so–called upper limit, which represents the approximate number of boards which could be stored in inventory were all viable available space to be used for storage.*

The current wastage recovery area has a usable area of 11 metres by 23 metres, which potentially allows for an additional 20 bays for AC boards and an additional 12 bays for

DWB boards, in other words an additional 14 540 AC and 5 052 DWB boards may be stored in this area, if required, at an additional cost. It is assumed that any additional storage space will be occupied proportionally between AC and DWB boards. The approximate number of boards which could then be kept in storage, based on one bay for the rank A boards of both board types, are given in Table 5.1(b). These are the values that will be used in the optimisation model as the upper limit of the order–to level.

5.3 Board Holding costs

Three categories of holding costs (dependent on the quantity of boards kept in the inventory) have been identified. These are the *rental cost* of floorspace in the factory, the *opportunity cost* of tied–up capital, and the *cost of insurance*.

5.3.1 Rental Cost

The building in which the *Clickabox* factory is housed is owned by the director of the factory, and *Clickabox* pays a monthly rental to him at a market–related rate. This is a fixed rental charge, independent of inventory levels. However, an increase in the amount of space utilised results in decreased manoeuvrability of the forklifts in the warehouse, and therefore an increase in the time required to fetch raw materials for production. The modelling of rental costs therefore takes into account the positive correlation between the amount of stock held and the cost of holding the stock. This is done as follows.

The proportion of total rental cost attributed to the storage area is given by

$$\begin{aligned} R_s &= \frac{A_s}{A_t} R_p \\ &= \frac{1\,584\text{m}^2}{2\,518.5\text{m}^2}\text{R415 400} \\ &= \text{R261}\,264.09, \end{aligned}$$

where A_t represents the total floor area of the factory, A_s represents the floor area of the storage space (as calculated in §3.2), and R_p the total rent paid per annum.

Rental cost for the storage area per volume of stock per week is now calculated as

$$\begin{aligned} R_c &= \frac{R_s}{52 A_s h'} \\ &= \frac{\text{R261}\,264.09}{52 \text{ weeks} \times 1\,584\text{m}^2 \times 3.2\text{m}} \\ &= \text{R0.99 per m}^3 \text{ per week,} \end{aligned} \tag{5.1}$$

where h' represents the height to which boards may be stacked.

This rate applies to the available space in the raw materials store, which can store approximately 31 988 AC boards and 11 788 DWB boards (as calculated in §5.2). If inventory levels exceed these thresholds, an additional cost will be incurred, as it will be necessary to utilize the current waste recovery area for the storage of stock. The waste recovery area currently generates an income of approximately R1 845 per month, or R461.25 per week [75]. The additional cost incurred is the opportunity cost of converting the space utilized for this recovery operation to storage space for inventory. Space available in this area is sufficient for the storage of a further 14 540 AC and 5 052 DWB boards.

Boards vary in dimension, so rental cost per board is different for each board kept in stock, and is given by the cost per cubic metre, R_c, multiplied by the volume of the board. There are 28 AC board types and 18 DWB board types kept in inventory from which sheets may be produced; let the sets $\mathcal{B}_{AC} = \{1, \ldots, 28\}$ and $\mathcal{B}_{DWB} = \{29, \ldots, 46\}$ index these board types respectively, and let $\mathcal{B} = \mathcal{B}_{AC} \cup \mathcal{B}_{DWB}$.

The rental cost function for board $\beta \in \mathcal{B}$ during week t is given by $x_t^{(\beta)} R_c v^{(\beta)}$, where $x_t^{(\beta)}$ is the number of boards of type β in inventory during week t, where $v^{(\beta)}$ is the volume per board of type β and R_c is the rental cost per volume of stock, as calculated in (5.1). The total rental cost is therefore given by

$$R_T = \begin{cases} \sum_{\beta=1}^{46} x_t^{(\beta)} R_c v^{(\beta)}, & \text{if } x_t^{(\beta)} \leq \eta_{(z_\beta)}^{\varpi_\beta} \text{ for all } \beta \in \mathcal{B} \\ \sum_{\beta=1}^{46} x_t^{(\beta)} R_c v^{(\beta)} + \text{R461.25}, & \text{otherwise,} \end{cases} \tag{5.2}$$

where $\varpi_\beta \in \{\text{A}, \text{B}\}$ denotes the rank and $z_\beta \in \{\text{AC}, \text{DWB}\}$ denotes the type of the board indexed by $\beta \in \mathcal{B}$. In other words, the additional rental cost for the waste recovery area is incurred if the inventory level of any of the stock board types exceeds the limit $\eta_{(z_\beta)}^{\varpi_\beta}$ for any board $\beta \in \mathcal{B}$, as shown in Table 5.1.

The values of $v^{(\beta)}$, the volume of each board β, are listed in Tables A.1 and A.2 of Appendix A.

5.3.2 Opportunity Cost

The loss due to capital tied up in stock is an opportunity cost, calculated in terms of the interest that would have been earned, were that money invested in the bank. Tied–up capital is therefore directly proportional to the cost of the raw materials. For a raw material purchase price of $p^{(\beta)}$ [Rands per board of type β] and a cost–of–capital–per–week rate $\overline{c}$, the opportunity cost of keeping one unit of board β in stock for one week is given by $\overline{c}p^{(\beta)}$. The purchase cost $p^{(\beta)}$ is given by

$$p^{(\beta)} = \begin{cases} A^{(\beta)} \times 2.54, & \text{if } z_\beta = \text{AC} \\ A^{(\beta)} \times 4.46, & \text{if } z_\beta = \text{DWB}, \end{cases} \tag{5.3}$$

where $A^{(\beta)}$ denotes the surface area of board β.

The total tied–up capital during week t is therefore

$$\begin{aligned} P &= \sum_{\beta=1}^{46} \overline{c} p^{(\beta)} x_t^{(\beta)} \\ &= 2.54\overline{c} \sum_{\beta=1}^{28} A^{(\beta)} x_t^{(\beta)} + 4.46\overline{c} \sum_{\beta=29}^{46} A^{(\beta)} x_t^{(\beta)}. \end{aligned} \tag{5.4}$$

5.3.3 Insurance Cost

Insurance costs are calculated on the total value of stock and machinery housed in the factory. This amount is reviewed annually, and the value of stock insured is calculated based on the estimated average stockholding for the year to come.

However, it is practically very difficult to link a variation in the number of boards in stock month by month to monthly insurance costs. The monthly insurance premium is, in fact, predetermined, and the number of boards insured at any stage depends on the composition of the inventory (in terms of board sizes, types and numbers) during the same month of the previous financial year, and the cost price of each board in stock.

This cost variation is negligible compared to the opportunity cost of the capital investment in the inventory. Insurance cost will therefore be considered a fixed cost for modelling purposes.

Current insurance rates are approximately R2 280 per month, *i.e.* R570 per week, for insurance on a replacement value of R500 000[1] of stock in inventory.

5.3.4 Total Holding Costs

The unit holding cost for board β is therefore

$$h^{(\beta)} = x_t^{(\beta)} \left(R_c v^{(\beta)} + \overline{c} p^{(\beta)} \right) + \frac{\text{R570.00}}{46}, \tag{5.5}$$

and the total holding cost is given by

$$h^T = P + R_T + \text{R570.00}, \tag{5.6}$$

where P and R_T are given in (5.4) and (5.2) respectively.

5.4 Cascading Shortage Costs

Shortage costs are difficult to estimate, due to the complicated nature of customer sentiment. This cost should incorporate bad sentiment if customers' orders are not met within the promised lead time, and also if the orders are met, but at a high cost due to the use of sub–optimal boards, with much wastage. In the case of *Clickabox*, the cascading shortage cost $s^{(\beta)}$ is formulated as the expense incurred due to this wastage (although it is, in fact, incurred by the customer and not the factory itself).

This cost may be estimated by

$$s^{(\beta)} = \frac{A^{(k)} - A^{(\beta)}}{A^{(\beta)}} p^{(\beta)}, \tag{5.7}$$

[1]This replacement value was estimated by the management of *Clickabox*, after inspection of the average value of stock in inventory during 2002, for the purposes of determining the insurance premiums for 2003 [70].

if board type k is the board type used instead of the optimal board type β, where $p^{(\beta)}$ is the purchase cost of board type β, and $A^{(\beta)}$ is the area of board type β as before.

However, the actual shortage cost depends on the composition of the board preference vectors and on the inventory levels, and calculation of the actual shortage cost therefore requires a knowledge of a number of issues not yet addressed. For the purposes of calculating service levels, however, the simplistic cascading shortage cost estimate in (5.7) is used. This concept will be extended later to comprise a wastage cost defined for each board in each board preference vector, and a shortage cost dependent on a combination of these wastage costs and the inventory levels of each board in the board preference vector.

In the case of no boards being available to satisfy an order, the order is backlogged, and a fixed stockout penalty cost is incurred.

5.5 Service Level Measures

In this section the choice of service level will be discussed, beginning with a clarification of the difference between *service level* and *fill rate*. A *service level* is the required percentage of order cycles during the year, during which there will be no shortages. A *fill rate* is the required percentage of units demanded that will be in stock when needed. The fill rate is generally significantly higher than the service level.

An algebraic computation can be performed to determine an appropriate service level $\alpha^{(\beta)}$ for each board type $\beta \in \mathcal{B}$, taking into account the trade–off between holding cost $h^{(\beta)}$ and shortage costs $s^{(\beta)}$, and the frequency at which the factory is exposed to the possibility of running out of stock [16]. This is calculated individually for each board type in the four steps outlined below.

Step 1: *Computing the optimal number of stockouts each year, $S^{*(\beta)}$.* This is achieved by taking the ratio of the annual holding cost $h^{(\beta)}$ of board type β to the shortage cost $s^{(\beta)}$ for board type β, because if the holding cost is relatively high compared to the shortage cost, the factory should plan on stocking out relatively often, since the cost of carrying high safety stocks would be prohibitive. Conversely, with relatively high shortage costs, the factory may plan for a low probability of stocking out by carrying high safety stock. Therefore the planned number of stockouts per year is given by

$$S^{*(\beta)} = \frac{h^{(\beta)}}{s^{(\beta)}}, \quad \beta \in \mathcal{B}. \tag{5.8}$$

Step 2: *Computing the number, $\kappa^{(\beta)}$, of order cycles each year.* This is given by the average annual demand $D^{(\beta)}$ for boards of type β divided by the quantity ordered during each cycle, that is

$$\kappa^{(\beta)} = \frac{D^{(\beta)}}{q^{(\beta)}}, \quad \beta \in \mathcal{B}. \tag{5.9}$$

In the case of a fixed review problem, $\kappa^{(\beta)}$ is the number of review periods in a year. This indicates the potential number of exposures to stockout the factory will experience for board β each year.

Step 3: *Computing the probability* $\overline{S}^{(\beta)}$ *of a stockout during each order cycle* given the optimal number of stockouts computed in (5.8) and the number of exposures to stockouts given by (5.9). Suppose, as an example, the optimal number of stockouts is one every two years, and there are five order cycles a year. Then there are ten order cycles in the two year period. If there is one stock out during these ten order cycles, there is a one in ten, or 10%, chance of stockout during the order cycle. In general this probability is

$$\overline{S}^{(\beta)} = \frac{S^{*(\beta)}}{\kappa^{(\beta)}}, \quad \beta \in \mathcal{B}. \tag{5.10}$$

Step 4: *Deriving the service level,* $\alpha^{(\beta)}$*,* as the probability of not stocking out of board β during each order cycle, *i.e.*

$$\begin{aligned} \alpha^{(\beta)} &= 1 - \overline{S}^{(\beta)} \\ &= 1 - \frac{S^{*(\beta)}}{\kappa^{(\beta)}} \\ &= 1 - \frac{h^{(\beta)} q^{(\beta)}}{s^{(\beta)} D^{(\beta)}}, \quad \beta \in \mathcal{B}, \end{aligned} \tag{5.11}$$

by utilisation of (5.8)–(5.10).

A major assumption of this computation is that holding and shortage costs are linear, *i.e.* carrying 100 boards costs twice as much as carrying 50 boards, and a stockout of five boards costs five times as much as a stockout of one board.

Service levels calculated as above will almost always be lower than intended, as the calculation does not take into account all the issues that affect customer perceptions, such as the length of stockout periods. Plossl and Wight (1967, [59]) suggest that the service level should be inflated to compensate for this phenomenom. However, they do not suggest by how much.

In view of the above discussion, the specification by the management of *Clickabox* that "95% of orders should be met within the specified time" is, in fact, a fill rate.

The above method was used to calculate a theoretically appropriate service level for each stock board at *Clickabox*.

A number of additional assumptions are made for the purposes of this calculation:

1. Rental cost and the cost of insurance are assumed to be fixed costs; holding cost is therefore only the opportunity cost defined in (5.4). Thus the assumption required for the service level computation, *i.e.* that the holding cost is linear, is assumed to hold.

2. The cost–of–capital–per–week rate is derived from the prime interest rate in South Africa at time of writing, namely 17% per annum [62].

3. The shortage cost, defined in (5.7) are also assumed to be linear for the purpose of the service level computation.

4. It is assumed that the sub–optimal board used will be the board type, amongst those having each dimension greater than or equal to those of the optimal board type, that is closest in area to the optimal board type. This is a 'best case' scenario, as this 'second best' board will not in reality always be available, and a different board, associated with a higher wastage cost, would have to be used. The reader should note that this is a simplifying assumption made in order to define a simple linear shortage cost per board for the purposes of the service level calculations only. As mentioned in §5.4, the modelling of the total shortage cost in a realistic manner is more complex and will be discussed in detail later.

The 'sub–optimal' board types were determined by means of a program written by the author in Visual Basic [50] and Microsoft Access [49][2]. The program also takes into account the possibility of using more than one board type, for example board type AC 2 300 × 1 200 cannot be substituted by a single board type, as there is no other stock board type of length 2 300 or greater. This board is most efficiently substituted by two boards of type AC1 250 × 1 300, joined lengthwise. The set of board types and their sub–optimal board types are given in Appendix A.

The optimal service level is now calculated for board type AC 1 260×2 300, as an example.

Step 1: The optimal number of stockouts each year is given by

$$\begin{aligned} S^{*(\beta)} &= \frac{h^{(\beta)}}{s^{(\beta)}} \\ &= \frac{p^{(\beta)}\overline{c}}{s^{(\beta)}} && \text{(Assumption (1), Equation (5.4))} \\ &= p^{(\beta)}\overline{c}\frac{A^{(\beta)}}{(A^{(k)} - A^{(\beta)})p^{(\beta)}} && \text{(Assumption (3), Equation (5.7))} \\ &= 0.17 \times \frac{2.898\text{m}^2}{0.23\text{m}^2}, && \text{(Assumption (3))} \\ &= 2.14. \end{aligned}$$

Thus the optimal number of stockouts is approximately two each year.

Step 2: The order quantity during each cycle is 1 000, as discussed in §3.5.1. The number of order cycles each year is then

$$\begin{aligned} \kappa^{(\beta)} &= \frac{D^{(\beta)}}{q^{(\beta)}} \\ &= \frac{38\,315}{1\,000} \\ &= 38.32. \end{aligned}$$

This gives an indication of the number of exposures to stockouts each year.

Step 3: The probability during each order cycle of a stockout is

$$\begin{aligned} \overline{S}^{(\beta)} &= \frac{S^{*(\beta)}}{\kappa^{(\beta)}} \\ &= 0.055\,8. \end{aligned}$$

[2]The relevant code is given in §C.2 of Appendix C

Step 4: The probability of not stocking out is therefore

$$\begin{aligned} \alpha^{(\beta)} &= 1 - \overline{S}^{(\beta)} \\ &= 0.944\,2. \end{aligned}$$

Index	Board Type	Stockouts per year	Order cycles per year	Stockout probability per order cycle	Theoretical Service Level
1	AC 1 030 × 2 370	0.58	12.21	0.05	95.22%
2	AC 1 260 × 2 300	2.14	38.32	0.06	94.41%
3	AC 1 280 × 1 300	1.97	11.27	0.17	82.54%
4	AC 1 330 × 2 370	1.74	5.93	0.29	70.67%
5	AC 1 360 × 2 300	1.60	31.19	0.05	94.87%
6	AC 1 380 × 1 310	1.01	6.91	0.15	85.36%
7	AC 1 460 × 2 370	0.84	24.55	0.03	96.59%
8	AC 1 470 × 1 480	2.75	9.57	0.29	71.23%
9	AC 1 500 × 1 540	0.93	5.04	0.18	81.60%
10	AC 1 510 × 1 810	0.54	46.06	0.01	98.82%
11	AC 1 530 × 1 380	0.54	34.74	0.02	98.43%
12	AC 1 550 × 1 020	1.15	5.93	0.19	80.59%
13	AC 1 680 × 1 080	1.16	17.08	0.07	93.23%
14	AC 1 720 × 1 210	0.81	33.55	0.02	97.58%
15	AC 1 800 × 1 200	1.03	5.31	0.19	80.64%
16	AC 1 860 × 1 000	1.17	10.48	0.11	88.87%
17	AC 1 860 × 1 490	0.71	52.06	0.01	98.64%
18	AC 1 910 × 1 880	0.47	29.15	0.02	98.38%
19	AC 2 000 × 1 400	0.75	6.48	0.12	88.43%
20	AC 2 030 × 1 240	0.96	25.41	0.04	96.22%
21	AC 2 110 × 1 010	0.71	24.23	0.03	97.06%
22	AC 2 110 × 1 680	1.55	29.78	0.05	94.79%
23	AC 2 200 × 1 200	2.70	4.02	0.67	32.79%
24	AC 2 260 × 1 520	1.17	18.21	0.06	93.56%
25	AC 2 260 × 2 160	0.28	27.65	0.01	98.99%
26	AC 2 300 × 1 220	3.05	10.95	0.28	72.16%
27	AC 2 300 × 1 710	0.44	21.89	0.02	98.01%
28	AC 2 370 × 1 250	1.38	5.17	0.27	73.34%

Table 5.2: *The optimal number of stockouts each year ($S^{*(\beta)}$), the number of order cycles a year (κ), the probability during each order cycle of a stockout ($\overline{S}^{(\beta)}$), and the theoretical service levels for each of the suggested AC Stock Boards to be kept in inventory.*

The theoretical service level for board type AC 1 260 × 2 300 is therefore 94.42%. The theoretical service levels for the other AC boards, calculated by the process detailed above, are given in Table 5.2.

Note that for the board AC 2 200 × 1 200 a very close substitute, namely AC2 300 × 1 220, is available with only 5.92% wastage. The cost of shortage is therefore quite low, and so the optimal number of stockouts a year is high. This, along with a low demand and therefore few order cycles a year, result in a service level of only 32.79%. In contrast, board AC1 910 × 1 880 has a 98.38% service level as the closest substitute is board type AC2 260 × 2 160, with a large wastage of 26.44%.

The average theoretical service level for AC boards is 87.61%, and the average theoretical service level for DWB boards is 90.68%. The theoretical service levels for the DWB boards are given in Table 5.3.

The fill rate used in this study is as specified by the factory director. There is currently no record kept of the factory's performance in terms of wastage loss and orders being

Index	Board Type	Stockouts per year	Order cycles per year	Stockout probability per order cycle	Theoretical Service Level
29	DWB 1 050 × 1 980	0.20	5.67	0.04	96.39%
30	DWB 1 170 × 1 310	1.22	5.57	0.22	78.14%
31	DWB 1 230 × 1 420	0.72	5.28	0.14	86.37%
32	DWB 1 270 × 1 700	0.64	9.07	0.07	92.98%
33	DWB 1 410 × 1 940	0.44	14.62	0.03	97.02%
34	DWB 1 480 × 1 310	2.10	14.25	0.15	85.30%
35	DWB 1 530 × 1 370	0.45	6.04	0.07	92.51%
36	DWB 1 670 × 1 010	0.57	10.83	0.05	94.77%
37	DWB 1 780 × 1 620	0.76	12.26	0.06	93.77%
38	DWB 1 820 × 2 090	0.53	17.88	0.03	97.01%
39	DWB 1 870 × 1 350	1.05	21.80	0.05	95.20%
40	DWB 2 010 × 1 460	0.84	9.97	0.08	91.54%
41	DWB 2 030 × 1 080	0.35	6.29	0.06	94.37%
42	DWB 2 150 × 1 640	1.10	9.21	0.12	88.09%
43	DWB 2 270 × 1 430	0.67	7.13	0.09	90.64%
44	DWB 2 300 × 2 180	0.27	6.10	0.04	95.54%
45	DWB 2 330 × 2 000	0.27	4.94	0.05	94.56%
46	DWB 2 410 × 1 690	2.82	8.84	0.32	68.06%

Table 5.3: *The optimal number of stockouts each year ($S^{*(\beta)}$), the number of order cycles a year (κ), the probability during each order cycle of a stockout ($\overline{S}^{(\beta)}$), and the theoretical service levels for each of the suggested DWB Stock Boards to be kept in inventory.*

met; these figures were estimated by two methods. The first was by consultation with the director of the factory, who is involved with all the operational aspects of the factory and is perceived to have a good idea of the situation. The estimates provided by the factory director were averages for all board types. The second method was by analysis of data of orders placed and met during the time period 1 February 2001 — 31 January 2003. This was done for AC and DWB board types separately. These estimates are given in Table 5.4.

	Opinion of director	Observed values	Observed values	Required
	All	AC	DWB	All
Orders met on time	90%			≥95%
Orders made with ideal board	80%	72.13%	74.27%	n\a
Off–cut wastage	30%	29.76%	29.68%	≤15%

Table 5.4: *Performance Measures*

Note that, due to the nature of the data available, the number of orders processed with ideal board is underestimated, as it is obtained by a count of only those orders made with more than one board type, and does therefore not include those orders filled completely with a sub–optimal board.

5.6 Optimal Control Policy

In this section an inventory model for *Clickabox* is derived for the case of non–stationary, partially observed demand with a positive lead time, l. The demand during each week may be generated by any one of a number of probability distributions. The probability distribution for the demand is determined by a Markov decision process. An expression for the expected inventory cost for each week is derived, and then an optimal control policy, which minimises this cost subject to the service levels derived in §5.5, is found.

5.6.1 Board Preference Vector Demand

The *demand classes* were defined in Chapter 4 to be seven ranges of values which may be assumed by the demand process (see Figure 4.7). For the remainder of this chapter the terms demand state and demand class will be used interchangeably, as each demand class represents a state in the Markovian process. The demand state process for each board preference vector, in other words the sequence of classes that may be taken on by the demand distribution for each board preference vector $\underline{v}_i$, is not known with certainty. It is modelled as a finite state Markov Chain, which may take on any one of a number of states during each week t, indexed by the set $\mathcal{K} = \{1, 2, \ldots, 7\}$.

The demand class of board preference vector $\underline{v}_i$ during week t is denoted by

$$d_t^{\underline{v}_i}, \quad i \in \mathcal{V},\ t \in \mathcal{T}. \tag{5.12}$$

The Markov process is governed by a transition probability matrix, $\mathbf{P}^{\underline{v}_i}$. Each time the system is in class k (that is, if $d_t^{\underline{v}_i} = k$ for some time $t \in \mathcal{T}$), there is a fixed probability $P_{k,j}^{\underline{v}_i}$ (the entry in row k and column j of a matrix of probabilities, $\mathbf{P}^{\underline{v}_i}$) that it will next be in class j (that is, $d_{t+1}^{\underline{v}_i} = j$, $t \in \mathcal{T} \setminus \{51\}$). These values, $P_{k,j}^{\underline{v}_i}$, are known as the transition probabilities of the Markov chain,

$$P_{k,j}^{\underline{v}_i} = \Pr\left[(d_{t+1}^{\underline{v}_i} = j)|(d_t^{\underline{v}_i} = k)\right]. \tag{5.13}$$

The transition probabilities satisfy the conditions

$$P_{k,j}^{\underline{v}_i} \geq 0 \quad \text{and} \quad \sum_{j=1}^{7} P_{k,j}^{\underline{v}_i} = 1, \quad k, j \in \mathcal{K},\ i \in \mathcal{V}. \tag{5.14}$$

The transition probability matrices $\mathbf{P}^{\underline{v}_1}, \ldots, \mathbf{P}^{\underline{v}_\mu}$ are themselves assumed to be stationary. Stationary demand for the board preference vector $\underline{v}_i$ may be modelled by taking $\mathbf{P}^{\underline{v}_i} = \mathbf{I}$, the identity matrix, whilst trends in demand for board preference vector $\underline{v}_i$ may be modelled by taking $\mathbf{P}^{\underline{v}_i}$ as an upper or lower triangular matrix.

The demand state process is partially observed and the observed demand for board preference vector $\underline{v}_i$ up to the start of week $t \in \mathcal{T}$ is defined as the measurement process $\{w_0^{\underline{v}_i}, \ldots, w_{t-1}^{\underline{v}_i}\}$. The observed demand $w_t^{\underline{v}_i}$ for board preference vector $\underline{v}_i$ during week t is the sum of all order quantities of orders placed during week t for sheets in $\mathcal{S}$ that may be produced by boards indexed in $\underline{v}_i$. It is assumed, for expected optimal replenishment purposes, that the best board in each board preference vector, that is the board $v_{i,1}$, will be used to fulfil an order. However, the inventory cost function will take into account the shortage cost incurred when the demand must, to some extent, be met by boards $v_{i,2}$ and $v_{i,3}$, due to stockout.

As discussed in Chapter 4, the demand realisation process is characterised by a normal distribution about the mean of each class. The demand realisation during a period is conditional on the current demand class. If $d_t = k$, the probability distribution of demand for board preference vector $\underline{v}_i$ during week t is given by

$$\Pr[(w_t^{\underline{v}_i} = z)|(d_t = k)] = r_{k,z}^{\underline{v}_i}, \qquad k \in \mathcal{K},\ t \in \mathcal{T},\ z \in \{0\} \cup \mathbb{R}^+,\ i \in \mathcal{V}. \tag{5.15}$$

5.6.2 Inventory Position

The quantity of the order placed by *Clickabox* at its suppliers during week t for board β is denoted by $q_t^{(\beta)}$. The inventory position for board β at the start of period t is defined as the net inventory level $x_t^{(\beta)}$ (on hand inventory less backorders) added to the stock on order at the beginning of week t, that is

$$u_t^{(\beta)} = x_t^{(\beta)} + \sum_{f=1}^{l} q_{t-f}^{(\beta)}, \qquad t \in \mathcal{T},\ \beta \in \mathcal{B}. \tag{5.16}$$

Here $q_{-1}^{(\beta)}, q_{-2}^{(\beta)}, \ldots, q_{-l}^{(\beta)}$ are assumed to represent orders placed during the l weeks immediately prior to the one year time window for which an optimal inventory policy is sought. All orders placed within the lead time number of weeks *before* any week t are included in the inventory position, as the demand considered in the inventory model is the lead time demand. Under the assumption of timely delivery of raw materials orders, all these orders will arrive within that lead time number of weeks, as measured from week t onwards.

The prior distribution, $\mathbf{\Pi}_t^{\underline{v}_i}$, is a rectangular matrix characterising the current belief of the distribution of the demand state $d_t^{\underline{v}_i}$, given the information available up to the start of week t, that is

$$\Pi_{k,t}^{\underline{v}_i} = \Pr[(d_t^{\underline{v}_i} = k)|\underline{I}_t^{\underline{v}_i}], \qquad t \in \mathcal{T},\ k \in \mathcal{K},\ i \in \mathcal{V}, \tag{5.17}$$

where $\underline{I}_t^{\underline{v}_i}$ is the vector of information available at the start of week t for board preference vector $\underline{v}_i$. The initial prior distribution, $\Pi_{k,0}^{\underline{v}_i}$, is assumed to be externally specified, based on market analysis and industry knowledge. The information vector $\underline{I}_t^{\underline{v}_i}$ is defined as

$$\underline{I}_0^{\underline{v}_i} = (\Pi_{1,0}^{\underline{v}_i}, \Pi_{2,0}^{\underline{v}_i}, \ldots, \Pi_{7,0}^{\underline{v}_i}), \tag{5.18}$$

together with

$$\underline{I}_t^{\underline{v}_i} = \begin{bmatrix} u_0^{v_{i,1}} & \ldots & u_{t-1}^{v_{i,1}} \\ u_0^{v_{i,2}} & \ldots & u_{t-1}^{v_{i,2}} \\ u_0^{v_{i,3}} & \ldots & u_{t-1}^{v_{i,3}} \\ w_0^{\underline{v}_i} & \ldots & w_{t-1}^{\underline{v}_i}, \end{bmatrix}, \qquad i \in \mathcal{V},\ t \in \mathcal{T} \setminus \{0\}\,. \tag{5.19}$$

At the start of any week t, the previous values of the order quantities placed, $(q_{t-1}^{(\beta)}, q_{t-2}^{(\beta)}, \ldots, q_{t-l}^{(\beta)})$, the current inventory level $(x_t^{(\beta)})$ and the current inventory position $(u_t^{(\beta)})$ are assumed to be known for each board $\beta \in \mathcal{B}$. Also known are the values of the prior distribution $\Pi_{j,t}^{\underline{v}_i}$ and the previous demand observations in the information matrix $\underline{I}_t^{\underline{v}_i}$.

As discussed, optimal inventory management should allow for the best board associated with a sheet to be available, on expectation. However, as a result of fluctuations in the actual demand this will not always be the case in reality. Observed demand for a board during a week is the sum of the observed demand for each board preference vector in which that board is the first choice, referred to as first level demand, added to the sum of the vector demands where that board was a second choice (which could not be satisfied by the first choice board), referred to as second level demand, and similarly where it was the third choice, called the third level demand.

In order to determine the realised demand for a board, it must be decided how available stock of a board will be allocated to board preference vector demand. In other words, if there is insufficient stock of a board to fill all the demand for which it is the optimal board, it must be decided which of the board preference vector demands will be supplied by the first choice board. The remainder of the demand will be filled by the second or third choice board. It is necessary to allocate available stock of a board to a specific board preference vector demand as the sheet–to–board conversion factors differ for different boards in a board preference vectors. As a result, different quantities of each board in the board preference vector would be required to fill a demand.

The allocation is based on the wastage cost, associated with each board in each board preference vector, incurred when a sub–optimal board is used. For the purpose of this thesis, a greedy algorithm was adopted, which attempts to minimise the wastage cost incurred by different allocations. Hence board allocation is conducted such that available stock of the first choice board is first allocated to the board preference vector for which the wastage cost of using the second choice board is the highest. Similarly, available stock of the second choice board is first allocated to the board preference vector for which the wastage cost of using the third choice board is the highest.

Define $\phi_{i,f}$ as the wastage cost incurred, per board, when the f–th best board in board preference vector $\underline{v}_i$ is used instead of the optimal board to meet a demand for board preference vector $\underline{v}_i$, where $f \in \{2,3\}$. This wastage cost is given by

$$\phi_{i,f} = (A^{v_{i,f}} - A^{v_{i,1}}) \div A^{v_{i,1}}. \tag{5.20}$$

Now the realised demand for board β during week t is given by

$$\underline{W}_t^{(\beta)} = \left[W_{t,1}^{(\beta)}; W_{t,2}^{(\beta)}; W_{t,3}^{(\beta)}\right], \tag{5.21}$$

where

$$W_{t,1}^{(\beta)} = \sum_{v_{i,1}=\beta} \frac{w_t^{\underline{v}_i}}{m_t^{(i,\beta)}} \tag{5.22}$$

$$W_{t,2}^{(\beta)} = \sum_{k=1}^{b}\left[\sum_{\substack{v_{i,1}=k\\ v_{i,2}=\beta}}\left(\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}} - \min\left\{\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}}, \max\left\{x_t^{(k)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2}\\ v_{j,1}=k; v_{j,2}=\beta}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,k)}}, 0\right\}\right\}\right)\frac{m_t^{(i,k)}}{m_t^{(i,\beta)}}\right] \tag{5.23}$$

$$
\begin{aligned}
W_{t,3}^{(\beta)} \;=\; & \sum_{k=1}^{b} \Bigg[\sum_{\substack{v_{i,1}=k \\ v_{i,3}=\beta}} \Bigg(\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}} - \min \Bigg\{ \frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}}, \max \Bigg[x_t^{(k)} + \max \Bigg(x_t^{v_{i,2}} - \sum_{v_{m,1}=v_{i,2}} \frac{w_t^{\underline{v}_m}}{m_t^{(m,v_{i,2})}}, 0 \Bigg) \\
& - \sum_{\substack{\phi_{j,3}>\phi_{i,3} \\ v_{j,1}=k; v_{j,3}=\beta}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,k)}}, 0 \Bigg] \Bigg\} \Bigg) \times \frac{m_t^{(i,k)}}{m_t^{(i,\beta)}} \Bigg]
\end{aligned}
\tag{5.24}
$$

for all $t \in \mathcal{T},\ k, \beta \in \mathcal{B}$, and $i \in \mathcal{V}$, where $m_t^{(i,\beta)}$ represents the number of sheets, optimally manufactured by board preference vector v_i, that may be produced from board β in week t.

The rationale behind these expressions is perhaps best explained by means of an example.

Consider the board preference vectors,

$$
\underline{v}_1 = \begin{bmatrix} b_1 \\ b_2 \\ b_4 \end{bmatrix}, \underline{v}_2 = \begin{bmatrix} b_3 \\ b_1 \\ b_4 \end{bmatrix}, \underline{v}_3 = \begin{bmatrix} b_3 \\ b_1 \\ b_2 \end{bmatrix}, \underline{v}_4 = \begin{bmatrix} b_2 \\ b_1 \\ b_4 \end{bmatrix}, \underline{v}_5 = \begin{bmatrix} b_4 \\ b_2 \\ b_1 \end{bmatrix}, \underline{v}_6 = \begin{bmatrix} b_4 \\ b_3 \\ b_1 \end{bmatrix},
$$

with demands during week t of $w_t^{\underline{v}_1}$, $w_t^{\underline{v}_2}$, $w_t^{\underline{v}_3}$, $w_t^{\underline{v}_4}$, $w_t^{\underline{v}_5}$ and $w_t^{\underline{v}_6}$ respectively. Let the inventory levels of the boards b_1, b_2, b_3 and b_4 at the start of week t be $x_t^{(b_1)}$, $x_t^{(b_2)}$, $x_t^{(b_3)}$, and $x_t^{(b_4)}$ respectively. Now the first level demand for board b_1, the sum of all the board preference vector demands where $v_{i,1} = b_1$, given by (5.22), is

$$
W_{t,1}^{(b_1)} = \sum_{v_{i,1}=b_1} \frac{w_t^{\underline{v}_i}}{m_t^{(i,b_1)}} = \frac{w_t^{\underline{v}_1}}{m_t^{(1,b_1)}}. \tag{5.25}
$$

The second level demand for board b_1 is the sum of all the board preference vector demands where $v_{i,2} = b_1$ and where the demand is not met by the best board in that vector. This demand is calculated for each board type k, over all board preference vectors for which $v_{i,1} = k$ and $v_{i,2} = b_1$, as the 'left over' demand to be met by board type b_1 is dependent on the inventory level of each board k. The sum of all demands for board preference vector $\underline{v}_j$ where $v_{j,1} = k$, $v_{j,2} = b_1$, and the shortage cost incurred by the second best board in $\underline{v}_j$ is higher than that incurred by the second best board in $\underline{v}_i$, is subtracted from the inventory position of board k. The second level demand for board b_1, given by (5.23), is

$$
W_{t,2}^{(b_1)} = \sum_{k=b_1}^{b_4} \Bigg[\sum_{\substack{v_{i,1}=k \\ v_{i,2}=b_1}} \Bigg(\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}} - \min \Bigg\{ \frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}}, \max \Bigg\{ x_t^{(k)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2} \\ v_{j,1}=k \\ v_{j,2}=b_1}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,k)}}, 0 \Bigg\} \Bigg\} \Bigg) \times \frac{m_t^{(i,k)}}{m_t^{(i,b_1)}} \Bigg]
$$

$$
\begin{aligned}
&= \sum_{\substack{v_{i,1}=b_2\\ v_{i,2}=b_1}} \left(\frac{w_t^{\underline{v}_i}}{m_t^{(i,b_2)}} - \min\left\{ \frac{w_t^{\underline{v}_i}}{m_t^{(i,b_2)}}, \max\left\{ x_t^{(b_2)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2}\\ v_{j,1}=b_2\\ v_{j,2}=b_1}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,b_2)}}, 0 \right\} \right\} \right) \times \frac{m_t^{(i,b_2)}}{m_t^{(i,b_1)}} \\
&\quad + \sum_{\substack{v_{i,1}=b_3\\ v_{i,2}=b_1}} \left(\frac{w_t^{\underline{v}_i}}{m_t^{(i,b_3)}} - \min\left\{ \frac{w_t^{\underline{v}_i}}{m_t^{(i,b_3)}}, \max\left\{ x_t^{(b_3)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2}\\ v_{j,1}=b_3\\ v_{j,2}=b_1}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,b_3)}}, 0 \right\} \right\} \right) \times \frac{m_t^{(i,b_3)}}{m_t^{(i,b_1)}} \\
&= \left(\frac{w_t^{\underline{v}_4}}{m_t^{(4,b_2)}} - \min\left\{ \frac{w_t^{\underline{v}_4}}{m_t^{(4,b_2)}}, \max\left\{ x_t^{(b_2)} - 0, 0 \right\} \right\} \right) \times \frac{m_t^{(4,b_2)}}{m_t^{(4,b_1)}} \\
&\quad + \left(\frac{w_t^{\underline{v}_2}}{m_t^{(2,b_3)}} - \min\left\{ \frac{w_t^{\underline{v}_2}}{m_t^{(2,b_3)}}, \max\left\{ x_t^{(b_3)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2}\\ v_{j,1}=b_3\\ v_{j,2}=b_1}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,b_3)}}, 0 \right\} \right\} \right) \times \frac{m_t^{(2,b_3)}}{m_t^{(2,b_1)}} \\
&\quad + \left(\frac{w_t^{\underline{v}_3}}{m_t^{(3,b_3)}} - \min\left\{ \frac{w_t^{\underline{v}_3}}{m_t^{(3,b_3)}}, \max\left\{ x_t^{(b_3)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2}\\ v_{j,1}=b_3\\ v_{j,2}=b_1}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,b_3)}}, 0 \right\} \right\} \right) \times \frac{m_t^{(3,b_3)}}{m_t^{(3,b_1)}}
\end{aligned}
\tag{5.26}
$$

Note that the remaining demand for board k is converted from the number of boards of board type k required to the number of board preference vectors $\underline{v}_i$ required, by multiplying by the factor $m_t^{(i,k)}$, and then to the number of boards of board type b_1 required, by dividing by the factor $m_t^{(i,b_1)}$.

The third level demand for board b_1 is the sum of all the board preference vector demands where $v_{i,3} = b_1$ and the demand is not met by the best or second best board in the board preference vector. A summation is performed for all boards k, over all board preference vectors having $v_{i,1} = k$ and $v_{i,3} = b_1$. In this example, this applies to board b_4, in vectors $\underline{v}_5$ and $\underline{v}_6$. Available stock of the first choice board $(x_t^{(k)})$ is added to the available stock of the second choice board (given by the difference of the inventory position of that board and the sum of all first level demand for that board). The sum of all board preference vector demands incurring a higher shortage cost than that incurred by the board preference vector under consideration is then subtracted, as this demand would be given preference in being supplied. Finally, this is subtracted from the total demand for that board preference vector, and converted to demand for board b_1. Third level demand is therefore given by

$$
\begin{aligned}
W_{t,3}^{(b_1)} &= \left(\frac{w_t^{v_5}}{m_t^{(5,b_4)}} - \min\left\{ \frac{w_t^{v_5}}{m_t^{(5,b_4)}}, \max\left[x_t^{(b_4)} + \max\left(x_t^{v_{5,2}} - \sum_{v_{m,1}=v_{5,2}} \frac{w_t^{v_m}}{m_t^{(m,v_{5,2})}}, 0 \right) \right.\right.\right. \\
&\qquad \left.\left.\left. - \sum_{\substack{\phi_{j,3}>\phi 5,3 \\ v_{j,1}=b_4 \\ v_{j,3}=b_1}} \frac{w_t^{v_j}}{m_t^{(j,b_4)}}, 0 \right] \right\} \right) \times \frac{m_t^{(5,b_4)}}{m_t^{(5,b_1)}} \\
&+ \left(\frac{w_t^{v_6}}{m_t^{(6,b_4)}} - \min\left\{ \frac{w_t^{v_6}}{m_t^{(6,b_4)}}, \max\left[x_t^{(b_4)} + \max\left(x_t^{v_{6,2}} - \sum_{v_{m,1}=v_{6,2}} \frac{w_t^{v_m}}{m_t^{(m,v_{6,2})}}, 0 \right) \right.\right.\right. \\
&- \left.\left.\left. \sum_{\substack{\phi_{j,3}>\phi 6,3 \\ v_{j,1}=b_4 \\ v_{j,3}=b_1}} \frac{w_t^{v_j}}{m_t^{(j,b_4)}}, 0 \right] \right\} \right) \times \frac{m_t^{(6,b_4)}}{m_t^{(6,\beta)}} \\
&= \frac{w_t^{v_5}}{m_t^{(5,b_1)}} - \frac{m_t^{(5,b_4)}}{m_t^{(5,b_1)}} \times \min\left\{ \frac{w_t^{v_5}}{m_t^{(5,b_4)}}, \max\left[x_t^{(b_4)} + \max\left(x_t^{b_2} - \frac{w_t^{v_4}}{m_t^{(4,b_2)}}, 0 \right) \right.\right. \\
&- \left.\left. \sum_{\substack{\phi_{j,3}>\phi_{5,3} \\ v_{j,1}=b_4; v_{j,3}=b_1}} \frac{w_t^{v_j}}{m_t^{(j,b_4)}}, 0 \right] \right\} + \frac{w_t^{v_6}}{m_t^{(6,b_1)}} - \frac{m_t^{(6,b_4)}}{m_t^{(6,b_1)}} \times \min\left\{ \frac{w_t^{v_6}}{m_t^{(6,b_4)}}, \max\left[x_t^{(b_4)} \right.\right. \\
&+ \left.\left. \max\left(x_t^{b_3} - \left\{ \frac{w_t^{v_2}}{m_t^{(2,b_3)}} + \frac{w_t^{v_3}}{m_t^{(3,b_3)}} \right\}, 0 \right) - \sum_{\substack{\phi_{j,3}>\phi 6,3 \\ v_{j,1}=b_4; v_{j,3}=b_1}} \frac{w_t^{v_j}}{m_t^{(j,b_4)}}, 0 \right] \right\}.
\end{aligned} \tag{5.27}
$$

Now suppose the inventory positions of the boards b_1, b_2, b_3 and b_4 at the start of week t are as follows: $x_t^{(b_1)} = 10$, $x_t^{(b_2)} = 12$, $x_t^{(b_3)} = 8$, and $x_t^{(b_4)} = 10$, and the board factor and demand realisation data for week t are as given in Table 5.5.

$\underline{v}_1$	$m_t^{(1,\beta)}$	$\underline{v}_2$	$m_t^{(2,\beta)}$	$\underline{v}_3$	$m_t^{(3,\beta)}$	$\underline{v}_4$	$m_t^{(4,\beta)}$	$\underline{v}_5$	$m_t^{(5,\beta)}$	$\underline{v}_6$	$m_t^{(6,\beta)}$
b_1	1	b_3	1	b_3	2	b_2	1	b_4	1	b_4	1
b_2	2	b_1	1	b_1	2	b_1	1	b_2	4	b_3	1
b_4	1	b_4	1	b_2	1	b_4	3	b_1	3	b_1	1
$w_t^{\underline{v}_1} = 8$		$w_t^{\underline{v}_2} = 14$		$w_t^{\underline{v}_3} = 21$		$w_t^{\underline{v}_4} = 10$		$w_t^{\underline{v}_5} = 2$		$w_t^{\underline{v}_6} = 12$	

Table 5.5: *Sheet–to–board conversion factors and board preference vector demand data for an example of the calculation of realised demand for a board preference vector.*

Furthermore, let the shortage costs $\phi_{i,k}$ for each of the second and third choice boards in each board preference vector be as given in Table 5.6.

$i \rightarrow$	1	2	3	4	5	6
$k = 2$	5	2	6	1	2	6
$k = 3$	8	3	8	5	4	10

Table 5.6: *Shortage costs $\phi_{i,k}$, representing the cost incurred when the k–th best board in board preference vector i is used instead of the best board, for use in the example of the calculation of realised demand.*

Now the first level demand for board b_1, given by (5.25), is

$$W_{t,1}^{(b_1)} = \frac{w_t^{\underline{v}_1}}{m_t^{(1,b_1)}} = 8.$$

The second level demand, given by (5.26), is

$$\begin{aligned}
W_{t,2}^{(b_1)} &= \left(\frac{w_t^{\underline{v}_4}}{m_t^{(4,b_2)}} - \min\left\{\frac{w_t^{\underline{v}_4}}{m_t^{(4,b_2)}}, x_t^{(b_2)}\right\}\right) \times \frac{m_t^{(4,b_2)}}{m_t^{(4,b_1)}} \\
&\quad + \left(\frac{w_t^{\underline{v}_2}}{m_t^{(2,b_3)}} - \min\left\{\frac{w_t^{\underline{v}_2}}{m_t^{(2,b_3)}}, \max\left\{x_t^{(b_3)} - \frac{w_t^{\underline{v}_3}}{m_t^{(3,b_3)}}, 0\right\}\right\}\right) \times \frac{m_t^{(2,b_3)}}{m_t^{(2,b_1)}} \\
&\quad + \left(\frac{w_t^{\underline{v}_3}}{m_t^{(3,b_3)}} - \min\left\{\frac{w_t^{\underline{v}_3}}{m_t^{(3,b_3)}}, \max\left\{x_t^{(b_3)} - 0, 0\right\}\right\}\right) \times \frac{m_t^{(3,b_3)}}{m_t^{(3,b_1)}} \\
&= \left(\frac{10}{1} - \min\left\{\frac{10}{1}, 12\right\}\right) \times \frac{1}{1} + \left(\frac{14}{1} - \min\left\{\frac{14}{1}, \max\left\{8 - \frac{21}{2}, 0\right\}\right\}\right) \times \frac{1}{1} \\
&\quad + \left(\frac{21}{2} - \min\left\{\frac{21}{2}, 8\right\}\right) \times \frac{2}{2} \\
&= 16.5.
\end{aligned}$$

Note that there was sufficient stock of board b_2 to fill the demand for board preference vector $\underline{v}_4$, so there is no second level demand for board b_2 for $\underline{v}_4$. However, there was insufficient stock of board b_3 to meet the combined demand for board preference vectors $\underline{v}_2$ and $\underline{v}_3$, so the outstanding quantity must be met by board b_1. The demand for board preference vector $\underline{v}_3$ is met first, as it incurrs a higher shortage cost than board preference vector $\underline{v}_2$. The sheet–to–board conversion factor indicates that two sheets of $\underline{v}_3$ may be produced from each board b_3, so 10.5 boards are required. There is an on hand inventory of 8, leaving a shortage of 2.5 boards to meet demand for board preference vector $\underline{v}_3$. There is no remaining stock of board b_3 to fill any of the demand for board preference vectors $\underline{v}_2$, so the entire quantity of 14 becomes second level demand for board b_1. Note also that non–integer demand quantities are allowed, as values such as the sheet–to–board conversion factors are averages over a week.

The third level demand, given by (5.27), is

$$
\begin{aligned}
W_{t,3}^{(b_1)} &= \frac{w_t^{\underline{v}_5}}{m_t^{(5,b_1)}} - \frac{m_t^{(5,b_4)}}{m_t^{(5,b_1)}} \times \min\left\{\frac{w_t^{\underline{v}_5}}{m_t^{(5,b_4)}}, \max\left[x_t^{(b_4)} + \max\left(x_t^{b_2} - \frac{w_t^{\underline{v}_4}}{m_t^{(4,b_2)}}, 0\right)\right.\right.\\
&- \left.\left.\frac{w_t^{\underline{v}_6}}{m_t^{(6,b_4)}}, 0\right]\right\} + \frac{w_t^{\underline{v}_6}}{m_t^{(6,b_1)}} - \frac{m_t^{(6,b_4)}}{m_t^{(6,b_1)}} \times \min\left\{\frac{w_t^{\underline{v}_6}}{m_t^{(6,b_4)}}, \max\left[x_t^{(b_4)} + \max\left(x_t^{b_3}\right.\right.\right.\\
&- \left.\left.\left.\left\{\frac{w_t^{\underline{v}_2}}{m_t^{(2,b_3)}} + \frac{w_t^{\underline{v}_3}}{m_t^{(3,b_3)}}\right\}, 0\right), 0\right]\right\}\\
&= \frac{2}{3} - \frac{1}{3}\min\left(\frac{2}{1}, \max\left[10 + \max\left\{12 - \frac{10}{1}, 0\right\} - \frac{12}{1}, 0\right]\right) + \frac{12}{1} - \frac{1}{1}\min\left(\frac{12}{1},\right.\\
&\left.\max\left[10 + \max\left\{8 - \frac{14}{1} - \frac{21}{2}, 0\right\}, 0\right]\right).
\end{aligned}
$$

Note that of the demand of 14 for the two board preference vectors with b_1 in third position, a quantity of 10 was met by the optimal board b_4, 2 by one of the second best boards, b_2, and none by the other second best board (b_3). The remaining demand quantity of 2 is the third level demand (in number of sheets) for board b_1.

5.6.3 Shortage Cost Revisited

The total shortage cost $\Psi^{(\beta)}$ is given by the quantity of sheets that have to be manufactured by the second best board, if there is not a sufficient quantity of the best board to fulfil an order, multiplied by the corresponding wastage, added to the quantity of sheets that must be manufactured by the third best board, if there is not a sufficient quantity of the second best board to fulfil the shortage, multiplied by the corresponding wastage, added to the quantity of sheets that is backordered, multiplied by the cost of backordering. The cost of backordering is represented by the shortage cost $\phi_{j,4}$. It is assumed that boards are first used to supply demand for the orders where they are the first choice, then, if there are any remaining boards of this type in stock, they are used to supply demand for orders for which they are the second choice, and finally for orders for which they are the third choice. Priority is given to the supply of the first choice boards for the board preference vectors which incur the highest wastage penalty when using the second best board. Similarly, the demand for second best boards is prioritised in terms of descending wastage costs, where the wastage cost refers to the cost of using the third best board instead for that order.

The total shortage cost for board β is calculated, in view of equations (5.23) and (5.24), as

$$
\begin{aligned}
\Psi^{(\beta)} = & \sum_{k=1}^{b} \Bigg[\sum_{\substack{v_{i,1}=k \\ v_{i,2}=\beta}} \phi_{i,2} \frac{m_t^{(i,k)}}{m_t^{(i,\beta)}} \Bigg(\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}} - \min\Bigg\{ \frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}}, \max\Bigg\{ u_t^{(k)} - \sum_{\substack{\phi_{j,2}>\phi_{i,2} \\ v_{j,1}=k; v_{j,2}=\beta}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,k)}}, 0 \Bigg\} \Bigg\} \Bigg) \\
& + \sum_{\substack{v_{i,1}=k \\ v_{i,3}=\beta}} \phi_{i,3} \frac{m_t^{(i,k)}}{m_t^{(i,\beta)}} \Bigg(\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}} - \min\Bigg\{ \frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}}, \max\Bigg[u_t^{(k)} + \max\Bigg(u_t^{v_{i,2}} - \sum_{v_{m,1}=v_{i,2}} \frac{w_t^{\underline{v}_m}}{m_t^{(m,v_{i,2})}} \\
& , 0 \Bigg) - \sum_{\substack{\phi_{j,3}>\phi_{i,3} \\ v_{j,1}=k; v_{j,3}=\beta}} \frac{w_t^{\underline{v}_j}}{m_t^{(j,k)}}, 0 \Bigg] \Bigg\} \Bigg) + \phi_{j,4} \frac{m_t^{(i,k)}}{m_t^{(i,\beta)}} \Bigg(\max\Bigg\{ \max\Bigg\{ \sum_{\substack{v_{i,1}=k \\ v_{i,3}=\beta}} \Bigg[\frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}} - \\
& \max\Bigg(u_t^{v_{i,2}} - \sum_{v_{m,1}=v_{i,2}} \frac{w_t^{\underline{v}_m}}{m_t^{(m,v_{i,2})}}, 0 \Bigg) \Bigg], 0 \Bigg\} - \max\Bigg\{ u_t^{k} - \sum_{\substack{v_{i,1}=k \\ v_{i,2}=\beta}} \frac{w_t^{\underline{v}_i}}{m_t^{(i,k)}}, 0 \Bigg\}, 0 \Bigg\} \Bigg) \Bigg] \\
& \text{for all } t \in \mathcal{T},\ i \in \mathcal{V},\ \beta \in \mathcal{B}. \qquad (5.28)
\end{aligned}
$$

The calculation of realised demand and total shortage costs was implemented by means of a greedy algorithm, which is described in detail in Appendix B.

5.6.4 Optimal Inventory Policy

The inventory control process begins with the information vector $\underline{I}_t^{\underline{v}_i}$, and proceeds as follows. The order placed the lead time l periods previously, if an order was placed, is received. The demand $w_t^{\underline{v}_i}$ occurs and inventory costs are incurred. The time index advances to $t+1$, and the demand state process advances from $d_t^{\underline{v}_i}$ to $d_{t+1}^{\underline{v}_i}$, according to the transition matrix $\mathbf{P}^{\underline{v}_i}$. The *posterior distribution*, $\Pi_{k,t+1}^{\underline{v}_i}$, which will become the *prior distribution* for week $t+1$, is now computed. The likelihood of occurence of the observed distribution is calculated as $\Pr[(w_t^{\underline{v}_i} = z)|(d_t^{\underline{v}_i} = k)]\Pr[(d_{t+1}^{\underline{v}_i} = m)|(d_t^{\underline{v}_i} = k)]$. The likelihood function is multiplied by the prior distribution, and then the function is normalized to obtain a unit probability over all possible demand states. This results in a so–called transition function which is applied to determine the transition from the prior distribution to the posterior distribution. The transition function T_m is given by

$$
\begin{aligned}
\Pi_{m,t+1}^{\underline{v}_i} &= T_m(\Pi_{k,t}^{\underline{v}_i} | I_t^{\underline{v}_i}) \\
&= \frac{\sum_{k=1}^{N} \Pi_{k,t}^{\underline{v}_i} \Pr[(w_t^{\underline{v}_i} = z)|(d_t^{\underline{v}_i} = k)]\Pr[(d_{t+1}^{\underline{v}_i} = m)|(d_t^{\underline{v}_i} = k)]}{\sum_{k=1}^{N} \Pi_{k,t}^{\underline{v}_i} \Pr[(w_t^{\underline{v}_i} = z)|(d_t^{\underline{v}_i} = k)]},
\end{aligned}
\qquad (5.29)
$$

for $t \in \mathcal{T} \setminus \{51\}$, $z \in \mathcal{N}$, $k, m \in \mathcal{K}$, $i \in \mathcal{V}$. This is written, using the notation in (5.14) and (5.16), as

$$\Pi^{\underline{v}_i}_{m,t+1} = \frac{\sum_{k=1}^{N} \Pi^{\underline{v}_i}_{k,t} r^{\underline{v}_i}_{k,z} P^{\underline{v}_i}_{k,m}}{\sum_{k=1}^{N} \Pi^{\underline{v}_i}_{k,t} r^{\underline{v}_i}_{k,z}}, \qquad t \in \mathcal{T} \setminus \{51\}, \ z \in \mathcal{N}, \ k, m \in \mathcal{K}, \ i \in \mathcal{V}. \tag{5.30}$$

The inventory position, defined in (5.16), follows the transition equation

$$u^{(\beta)}_{t+1} = u^{(\beta)}_t + q^{(\beta)}_t - W^{(\beta)}_{t,1}, \qquad t \in \mathcal{T} \setminus \{51\}, \ \beta \in \mathcal{B}, \ i \in \mathcal{V}, \tag{5.31}$$

where $W^{(\beta)}_{t,1}$ is the first level demand for board β in week t, as defined in (5.22). For an inventory position $u^{(\beta)}_t$, a shortage cost $\Psi^{(\beta)}$ and a holding cost $h^{(\beta)}$, the single period expected inventory cost function for board β is defined in [73] as

$$G^{(\beta)}_t(u^{(\beta)}_t) = \mathrm{E}\left[\Psi^{(\beta)} + h^{(\beta)} \max\left\{0, u^{(\beta)}_t - \hat{W}^{(\beta)}_{t,1}\right\}\right], \qquad t \in \mathcal{T} \setminus \{51\}, \ \beta \in \mathcal{B}, \tag{5.32}$$

where the first level demand for the lead time from week t onwards is given by $\hat{W}^{(\beta)}_{t,1} = \sum_{n=0}^{l} W^{(\beta)}_{t+n,1}$, $t \leq 51 - l$, and $\mathrm{E}[\bullet]$ denotes the expected value operator. Only the first level demand is considered as the model allows for the best board associated with a sheet to be available, on expectation.

The total single period inventory cost function is defined as

$$H^{(\beta)}_t(u^{(\beta)}_t) = G^{(\beta)}_t(u^{(\beta)}_t) + p^{(\beta)} q^{(\beta)}. \tag{5.33}$$

Let ζ_t be the set of all possible lead time demand state sequences $\underline{\sigma}^{\underline{v}_i}_t = (d^{\underline{v}_i}_t, d^{\underline{v}_i}_{t+1}, \dots d^{\underline{v}_i}_{t+l})$, and define $\zeta^{(i,k)}_t = \left\{\underline{\sigma}^{\underline{v}_i}_t \in \zeta : d^{\underline{v}_i}_t = k, \text{ for all } k \in \mathcal{K}, \ t \in \mathcal{T}, \ i \in \mathcal{V}\right\}$. The probability distribution for weekly demand for board preference vector $\underline{v}_i$ was defined in (5.15) as $r^{\underline{v}_i}_{k,z} = \Pr[(w^{\underline{v}_i}_t = z)|(d^{\underline{v}_i}_t = k)]$. The probability distribution for the lead time demand for board preference vector $\underline{v}_i$ is calculated over all possible demand state sequences, as the probability of that sequence occuring, multiplied by the probability of an observed lead time demand of z, given that sequence, *i.e.*

$$\hat{r}^{\underline{v}_i}_{k,z} = \sum_{\underline{\sigma}^{\underline{v}_i}_t \in \zeta^{(i,k)}_t} \Pr[\underline{\sigma}^{\underline{v}_i}] \Pr[(\hat{w}^{\underline{v}_i}_t = z)|\underline{\sigma}^{\underline{v}_i}_t] \qquad \text{for all } t \in \mathcal{T} \setminus \{51\}, \ i \in \mathcal{V}, \ k \in \mathcal{K}, \tag{5.34}$$

where $\hat{w}^{\underline{v}_i}_t$ is the lead time demand observations of board preference vector $\underline{v}_i$, *i.e.* $\hat{w}^{\underline{v}_i}_t = \sum_{f=0}^{l} w^{\underline{v}_i}_{t+f}$ for all $t \leq 51 - l$, and $\Pr[\underline{\sigma}^{\underline{v}_i}] = \mathbf{P}_{d^{\underline{v}_i}_t, d^{\underline{v}_i}_{t+1}} \mathbf{P}_{d^{\underline{v}_i}_{t+1}, d^{\underline{v}_i}_{t+2}} \dots \mathbf{P}_{d^{\underline{v}_i}_{t+l-1}, d^{\underline{v}_i}_{t+l}}$ is the probability of demand state sequence $\underline{\sigma}^{\underline{v}_i}$ occurring.

Now let $n_k(\underline{\sigma}^{\underline{v}_i}_t)$ be the number of times that state k occurs in the sequence $\underline{\sigma}^{\underline{v}_i}_t$, and let χ^n_k denote the sum of n independent, identically distributed random variables with distribution $r^{\underline{v}_i}_{k,z}$ in state $k \in \mathcal{K}$.

From these definitions, the lead time demand distribution, given the state sequence $\underline{\sigma}^{\underline{v}_i}_t$, is given by

$$\Pr[(\hat{w}^{\underline{v}_i}_t = z)|\underline{\sigma}^{\underline{v}_i}] = \Pr\left[\sum_{k=1}^{k'} \chi_k^{n_k(\underline{\sigma}^{\underline{v}_i})} = z\right] \qquad \text{for all } t \in \mathcal{T}, \ i \in \mathcal{V}, \ k \in \mathcal{K}, \tag{5.35}$$

which is calculated as a simple convolution of $l+1$ independent random variables.

For example, given a lead time of 2 weeks, and two possible demand states, say k_1 and k_2, the set of possible demand state sequences for $k = k_1$ is $\zeta_t^{(i,k_1)} = \{(k_1, k_1), (k_1, k_2)\}$ and for $k = k_2$ is $\zeta_t^{(i,k_2)} = \{(k_2, k_1), (k_2, k_2)\}$. Then

$$\begin{aligned} \hat{r}_{k,z}^{\underline{v}_i} &= \Pr[(k_1, k_1)]\Pr[(\hat{w}_t^{\underline{v}_i} = z)|(k_1, k_1)] + \Pr[(k_1, k_2)]\Pr[(\hat{w}_t^{\underline{v}_i} = z)|(k_1, k_2)] \\ &= P_{k_1,k_1}\Pr[(\hat{w}_t^{\underline{v}_i} = z)|(k_1, k_1)] + P_{k_1,k_2}\Pr[(\hat{w}_t^{\underline{v}_i} = z)|(k_1, k_2)], \end{aligned}$$

for all $t \in \mathcal{T} \setminus \{51\}$, $i \in \mathcal{V}$, $k \in \mathcal{K}$. The demand realisation probabilities are given by $\Pr[(\hat{w}_t^{\underline{v}_i} = z)|(k_1, k_1)] = \Pr[\chi_{k_1}^2 = z]$ and $\Pr[(\hat{w}_t^{\underline{v}_i} = z)|(k_1, k_2)] = \Pr[\chi_{k_1}^1 + \chi_{k_2}^1 = z]$.

If $p^{(\beta)}$ is the linear procurement cost per board of type β, as before, and $y_t^{(\beta)} = u_t^{(\beta)} + q_t^{(\beta)}$, then the total period cost to be minimised is the sum of the purchase cost $p^{(\beta)}q^{(\beta)}$ and the single period cost $G_t^{(\beta)}(y_t^{(\beta)})$ defined in (5.32).

The dynamic programming recursion for the optimal inventory control policy is therefore

$$J_t^{(\beta)}(u_t^{(\beta)}, \mathbf{\Pi}_t) = -p^{(\beta)}u_t^{(\beta)} + \min_{y_t^{(\beta)} \geq u_t^{(\beta)}} \left\{ p^{(\beta)}y_t^{(\beta)} + G_t^{(\beta)}(y_t^{(\beta)}) + \mathrm{E}[J_{t+1}^{(\beta)}(y_t^{(\beta)} - w_t^{(\beta)}, \mathbf{\Pi}_t)] \right\} \tag{5.36}$$

for $t = 0, \ldots, T$, where $t = 0$ indexes the first period and the sequence of calculations occurs over $T, T-1, \ldots, 1, 0$ as noted in (2.11). The terminal cost function is defined as $J_{T+1}^{(\beta)}(u_{T+1}^{(\beta)}, \mathbf{\Pi}_{T+1}) = -p^{(\beta)}u_{T+1}^{(\beta)}$.

Treharne and Sox [73] proved that the function $J_t(u_t, \mathbf{\Pi}_t)$ is nonincreasing and convex for $u_t < S_t(\mathbf{\Pi}_t)$ and nondecreasing and convex for $u_t \geq S_t(\mathbf{\Pi}_t)$, and therefore is convex for all u_t. This demonstrates the optimality of a state–dependant $(\overline{s}, \overline{S})$ policy in the case of a positive ordering cost, and a base–stock policy in the absence of fixed ordering costs.

Hence there exists a pair of values $\left(s_t^{(\beta)}(\mathbf{\Pi}_t), S_t^{(\beta)}(\mathbf{\Pi}_t)\right)$ that minimise the cost function such that the optimal policy is

$$q_t^{(\beta)} = \begin{cases} S_t^{(\beta)}(\mathbf{\Pi}_t) - u_t^{(\beta)}, & \text{if } u_t^{(\beta)} < s_t^{(\beta)}(\mathbf{\Pi}_t) \\ 0, & \text{otherwise.} \end{cases} \tag{5.37}$$

5.7 Sub–optimal Control Policy

Due to the intense computational requirements for determining an optimal solution to the model developed in §5.6, it is necessary to consider sub–optimal control policies, as alternatives. As discussed in §2.1, Treharne and Sox [73] compared a number of sub–optimal policies, applicable to non–stationary, partially observed demand situations, that may be computed more easily than the optimal policy, and found them to perform relatively well. They made recommendations as to which sub–optimal policy should be utilised, based on the trend and correlation of the demand data.

The trend of the demand data is calculated as

$$\mathrm{Trend}(\mathbf{P}^{\underline{v}_i}, \mathbf{\Pi}^{\underline{v}_i}) = \sum_{k=1}^{N}\sum_{j=1}^{N} \Pi_{k,0}^{\underline{v}_i} P_{kj}^{(4)\underline{v}_i} \upsilon_j^{\underline{v}_i} - \sum_{k=1}^{N} \Pi_{k,0}^{\underline{v}_i} \upsilon_j^{\underline{v}_i}, \quad k, j \in \mathcal{K},\ i \in \mathcal{V}, \tag{5.38}$$

where $\upsilon_j^{\underline{v}_i} = \sum_k k r_{j,k}^{\underline{v}_i}$ is the mean of distribution j and $P_{ij}^{(4)\underline{v}_i}$ is the four–step transition probability from state i to state j, derived from the transition matrix defined in (5.12), given by

$$P_{ij}^{(4)\underline{v}_i} = \left(P^{\underline{v}_i} \times P^{\underline{v}_i} \times P^{\underline{v}_i} \times P_{ij}^{\underline{v}_i}\right). \tag{5.39}$$

The correlation of the demand data is given by

$$\mathrm{Corr}(\mathbf{P}^{\underline{v}_i}, \mathbf{\Pi}^{\underline{v}_i}) = \frac{\mathrm{Cov}(w_0^{\underline{v}_i}, w_1^{\underline{v}_i}|\mathbf{\Pi}^{\underline{v}_i})}{\sqrt{\mathrm{Var}(w_0^{\underline{v}_i}|\mathbf{\Pi}^{\underline{v}_i})}\sqrt{\mathrm{Var}(w_1^{\underline{v}_i}|\mathbf{\Pi}^{\underline{v}_i})}}. \tag{5.40}$$

This is found by the calculation of the covariance,

$$\mathrm{Cov}(w_0^{\underline{v}_i}, w_1^{\underline{v}_i}|\mathbf{\Pi}^{\underline{v}_i}) = \sum_{k=1}^{N}\sum_{j=1}^{N} \Pi_{k,0}^{\underline{v}_i} P_{kj} \upsilon_k^{\underline{v}_i} \upsilon_j^{\underline{v}_i} - \sum_{k=1}^{N} \Pi_{k,0}^{\underline{v}_i} \upsilon_k^{\underline{v}_i} \sum_{k=1}^{N}\sum_{j=1}^{N} \Pi_{k,0}^{\underline{v}_i} P_{kj}^{(4)} \upsilon_j^{\underline{v}_i}, \tag{5.41}$$

for $k, j \in \mathcal{K},\ i \in \mathcal{V}$, and of the variances,

$$\begin{aligned}
\mathrm{Var}(w_0^{\underline{v}_i}|\mathbf{\Pi}^{\underline{v}_i}) &= \sum_{k=1}^{N} \Pi_{k,0}^{\underline{v}_i} \upsilon_j^{\underline{v}_i(2)} - \left(\sum_{k=1}^{N} \Pi_{k,0}^{\underline{v}_i} \upsilon_j^{\underline{v}_i}\right)^2, \quad \text{and} \\
\mathrm{Var}(w_1^{\underline{v}_i}|\mathbf{\Pi}^{\underline{v}_i}) &= \sum_{k=1}^{N}\sum_{j=1}^{N} \Pi_{k,0}^{\underline{v}_i} P_{kj} \upsilon_j^{\underline{v}_i(2)} - \left(\sum_{k=1}^{N}\sum_{j=1}^{N} \Pi_{k,0}^{\underline{v}_i} P_{kj} \upsilon_j^{\underline{v}_i}\right)^2,
\end{aligned} \tag{5.42}$$

for $k, j \in \mathcal{K}, i \in \mathcal{V}$, where $\upsilon_j^{\underline{v}_i(2)}$ denotes the second moment of distribution j, *i.e.* $\upsilon_j^{\underline{v}_i(2)} = \sum_k k^2 r_{j,k}^{\underline{v}_i}$.

The trend and correlation of the board demand data were calculated for a number of the board preference vectors. The results were varied for the trend, although the trend was predominately positive. The correlation, however, was negative for all board preference vectors tested. The results for five of the cases tested are given in Table 5.7. According to Treharne and Sox's study [73], this suggests the use of a *myopic policy*, as it is faster than the other strategies, and almost always optimal for this class of problems.

Statistic	B.P.V. 2	B.P.V. 6	B.P.V. 10	B.P.V. 11	B.P.V. 12
Trend	−32.4	61.41	−217.39	−25.74	0.49
Covariance	−4 056.41	−32 013.84	−2 853.15	−21 159.39	−0.56
Variance 1	32 576.86	623 853.81	198 827.48	154 340.59	31.16
Variance 2	19 251.29	590 892.87	7 025.28	91 157.74	60.92
Correlation	−0.16	−0.05	−0.08	−0.18	−0.01

Table 5.7: *Partial results of the trend and correlation analysis conducted on a sample of board preference vectors, indicating the suitability of a myopic policy for adaptive inventory control.*

The myopic policy is a special case of the *Limited Look–ahead Strategies*. These control policies find an optimal solution to the problem for a limited number of weeks into the future, say ρ weeks, instead of for the whole future. The myopic policy minimises only the current period costs, *i.e.* $\rho = 0$.

For all limited look ahead policies, the transition equation $\mathbf{\Pi}_t = T(\mathbf{\Pi}_{t-1}|w_{t-1})$ updates the prior distribution at the beginning of week t. For the myopic policy, the objective is defined as

$$\overline{J_t}^{L0}(u_t, \mathbf{\Pi}_t) = \min_{S_t \geq u_t} \left\{ c(S_t - u_t) + G_t(S_t|\mathbf{\Pi}_t, l) \right\}. \tag{5.43}$$

5.8 Chapter Summary

A strategic mathematical model was developed in this chapter, aimed at assisting management at *Clickabox* in inventory replenishment decision making. The assumptions and constraints under which the model was developed were discussed in §5.1 and §5.2 respectively, and the costs incorporated in the model were defined in §5.3 and §5.4. Service levels for the stock board types were derived in §5.5. A theoretical optimal control policy was then derived in §5.6, based on the approach by Treharne and Sox, and finally a more practical, sub–optimal control policy was described in §5.7.

Chapter 6

Model Results

The inventory model described in Chapter 5 was implemented in Visual Basic[1] in the form of a number of programs. These programs form a logical unit which is referred to throughout this chapter generically as the simulation model, for the purposes of validation and generation of results. The practical implementation of the simulation model is a decision support system, in which an interface allows a user to select from a number of possible simulations, based on the output required. The two principal outputs sought from the simulation model were (i) a single set of static replenishment parameters, *i.e.* a re–order and order–to level for each stock board that minimises expected one–period costs and (ii) a dynamic set of re–order and order–to levels for each stock board that minimises total expected cost each week for a given number of weeks.

The structure and basic elements of the simulation model are outlined in §6.1. In §6.2, the model is shown to be reasonable by presentation of the results of testing the model on four key conditions. The existence of a transient phase, before the system reaches a steady state, is discussed in §6.3, and the Welch method is followed in order to determine a truncation point for this transient phase. In §6.4 and §6.5 the results of the simulation model are presented and discussed. Section 6.4 is devoted to the static single period optimisation solution, giving the desired output outlined in (i) above. In §6.5 the dynamic multiple period optimisation results are given, that is the desired output discussed in (ii) above. Finally, in §6.6, the results of the model are compared and evaluated.

6.1 Structure of the Simulation Model

The simulation model developed consists of three basic elements, each of which will be described in some detail in this section. These are an Access database, containing a number of tables which store data and are updated dynamically during a simulation run; a set of programs written in Visual Basic, based on the assumptions and logic established in Chapter 5, which process the data in the database; and an interface to the programs, which allows a user to select the type of simulation required.

[1]The relevant source code is given in Section C.6 of Appendix C.

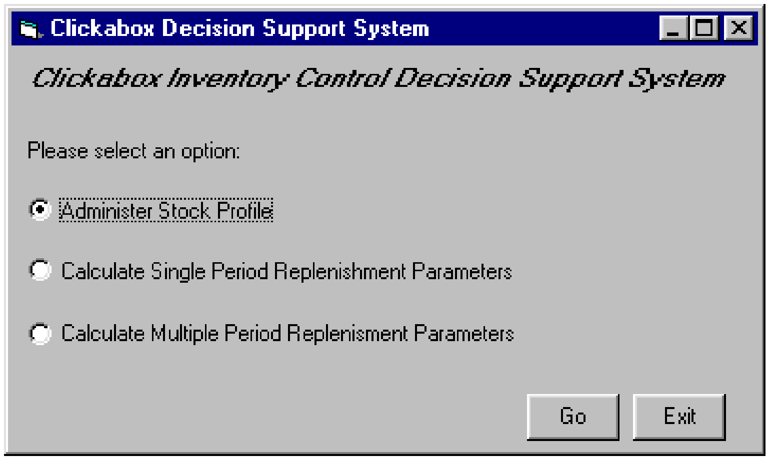


Figure 6.1: *Decision Support System Initial Option Screen*

6.1.1 Decision Support System Interface

The initial option screen of the decision support system developed for interactive use at *Clickabox* is shown in Figure 6.1.

The basic options are to administer the stock profile — this includes adding or removing stock board types and changing values such as the purchase cost of stock board — or to run either a single or multiple period optimisation simulation.

6.1.2 Simulation Database

The data used by the simulation model are the matrices and distributions derived in Chapter 4, that is the demand realisation distribution (given in §4.3.2.2), the transition probabilities and sheet–to–board conversion factors (given in §4.3.2), as well as required information about the stock, such as the board dimensions and unit purchase price. The dimensions and ranks of boards kept in inventory are given in Tables 4.2 and 4.3, and purchase and holding costs for each board were computed in (5.3) and (5.5) respectively. This information is stored in tables in the Access database and referenced by the Visual Basic code. The structure of the main tables in the database is given in Appendix C.

6.1.3 Program Code

The program code is included, with commentary, in Appendix C. The basic stages of the single period optimisation are illustrated in Figure 6.2. Each of these stages represents a program which is executed in sequence, reading and ammending the database.

The simulation model calculates the total cost incurred for each board type for all potential values of the re–order level $\overline{s}$ and the order–to level $\overline{S}$. For practical[2] and com-

[2]It was decided, by consultation with the director of *Clickabox* [70], that orders from the supplier should be placed in multiples of 100, so as not to complicate practical issues such as bundling and stock

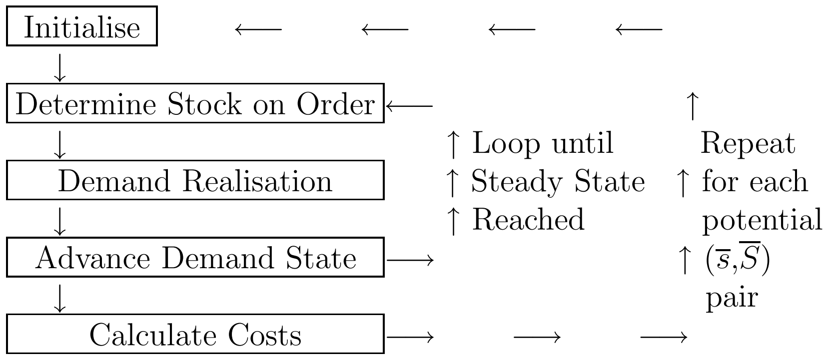


Figure 6.2: *Stages of the Single Period Optimisation Simulation*

putational purposes, these calculations were performed in steps of 100 for both values. Potential values for the order–to level $\overline{S}$ range from the quantity $(\overline{s} + 100)$ to the upper limit of the spatial constraint for the board type and rank in question, as given in Table 5.1(b). The cost incurred by each potential pair of replenishment parameters, for each stock board type, is written to a table in the database, and the least cost solution is extracted by means of a query. Each repetition of the entire process is referred to as an iteration.

The basic stages of the multiple period optimisation are illustrated in Figure 6.3. Note that, due to its different application as a planning device to be used in a specific situation, compared to the single period model which minimises costs on average, the model begins in a specified demand state. It is not necessary to bring the dynamic model into a steady state. The stages illustrated in Figure 6.3 are repeated for the number of weeks for which the projection is required, for each stock board.

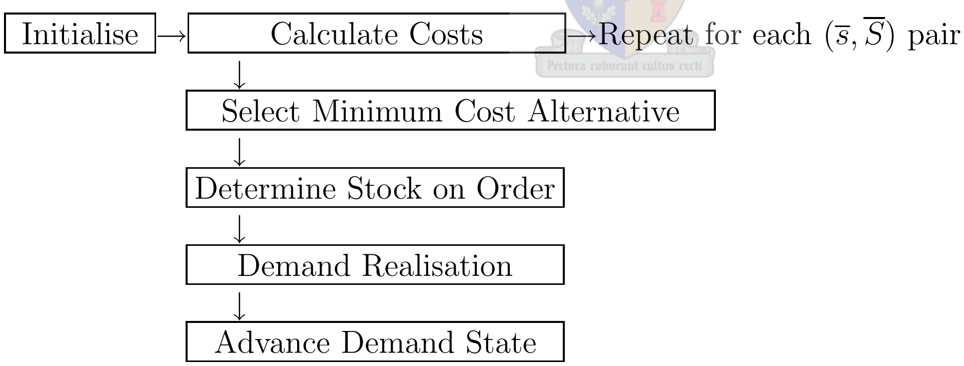


Figure 6.3: *Stages of the Multiple Period Optimisation Simulation*

6.1.4 Decision Support System Output

The output generated by the single period model is a list of suggested replenishment parameters for each of the board types investigated. An example of the output generated

checking.

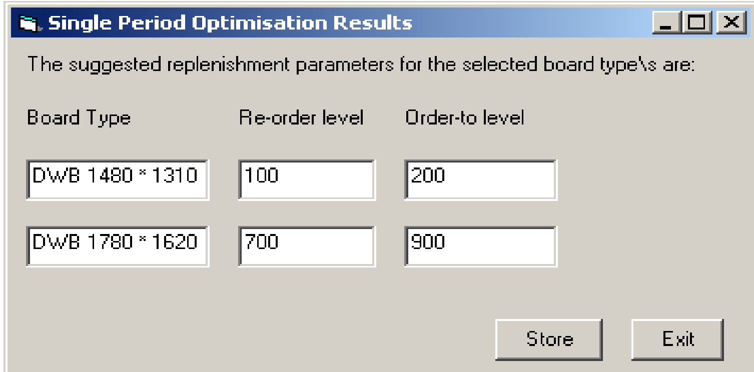


Figure 6.4: *Decision Support System Output Screen: Single Period Optimisation*

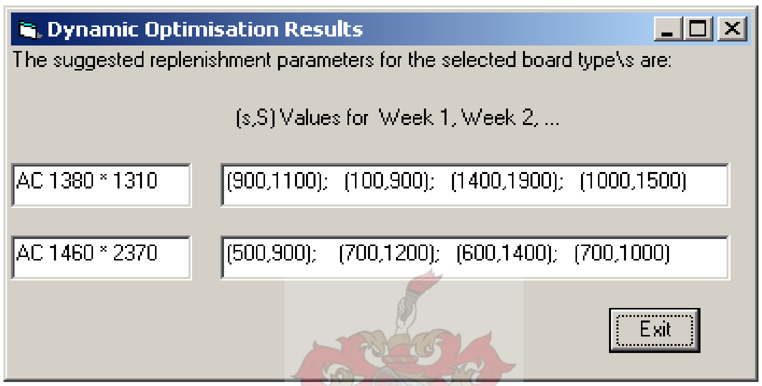


Figure 6.5: *Decision Support System Output Screen: Dynamic Optimisation*

by the decision support system for the single period optimisation is shown in Figure 6.4. An option is available to store the results obtained in the stock table for future use or application in the dynamic model.

The output generated by the dynamic model, namely a list of the suggested $(\overline{s}, \overline{S})$ pairs for each of the specified number of weeks, for each board type selected, is illustrated in Figure 6.5.

6.2 Model Validation

A simulation model may be validated against the following factors [10]:

- Continuity: Small changes made to input parameters should be reflected by correspondingly small changes in the model output.
- Consistency: Model output should not vary significantly if the model is run through a number of iterations under slight changes (such as a change in the random number generator seed).

- Degeneracy: The model should be shown to perform differently with the removal of one or more features of the model, such as resources. If, for example, one of two output–generating resources is removed, the output of the system should be reduced, and overloading of the remaining resource could occur.

- Absurd conditions: Absurd conditions introduced to the model should not necessarily produce absurd results; variables should always remain within their range definition.

A number of tests were conducted to ensure that the model developed conforms to all of the above mentioned conditions. These tests were conducted against the single period model, as it generates results which can more easily be evaluated against these conditions than those generated from the dynamic model. The validation applies, however, to both models, as the components of and logic behind the programs are the same.

6.2.1 Continuity

The continuity of the model was tested by varying the initial stock level used in the simulation, and confirming that the ouput converges to similar results. The initial stock level in the simulation model was assumed to be a half of the order–to value, and the initial stock on order was taken to be zero. This approximation is compensated for in the model by discarding the output from the first few steps in the simulation, until the situation stabilizes, as will be discussed in some detail later. The single period simulation model, which determines the optimal replenishment parameters for each board, was run five times, each time altering the parameters of the model by making a small change to the initial stock level. The initial stock levels used were $0.9I$, $0.95I$, I, $1.05I$ and $1.1I$, where I represents half of the order–to value. The resulting holding and purchase costs for each of 52 successive weeks for each of the five simulations, are shown in Figure 6.6.

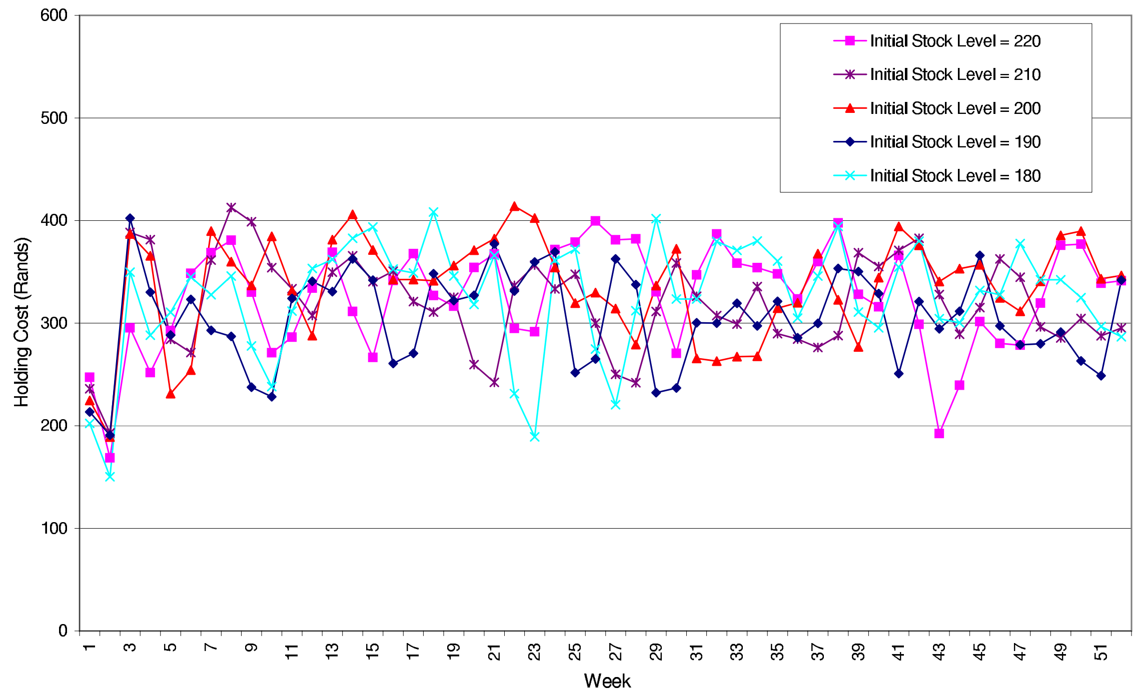
Figure 6.6 illustrates that the holding and purchase cost curves for each simulation run fall within the same range. This establishes continuity in the model.

6.2.2 Consistency

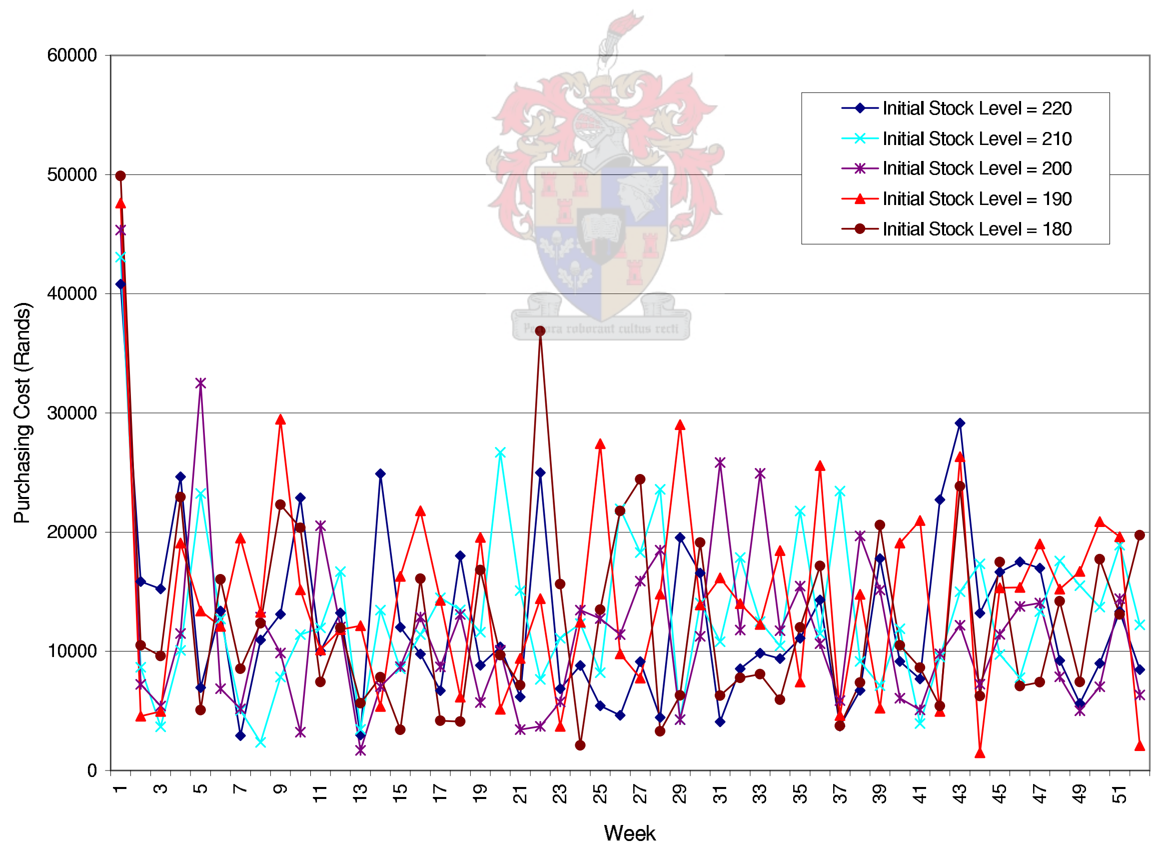
The consistency of the model was verified by an analysis of the output of the single period optimisation, in which the model was run through 1 000 iterations, and the random number generator seed was changed at each iteration. In each iteration, the simulation model was brought to a steady state, and then the minimum total single period cost was calculated. The minimum total costs for the first 100 iterations[3] for two of the board types are shown in Figure 6.7(a). The holding cost over all boards for the first 100 iterations is shown to be consistent in Figure 6.7(b).

The mean of the holding cost over all boards for the 100 iterations shown in Figure 6.3(b) is 20 202.41, and the standard deviation is 352.107. The mean is therefore two orders of

[3]For practical reasons only the first 100 of the 1 000 iterations are shown on the graph.

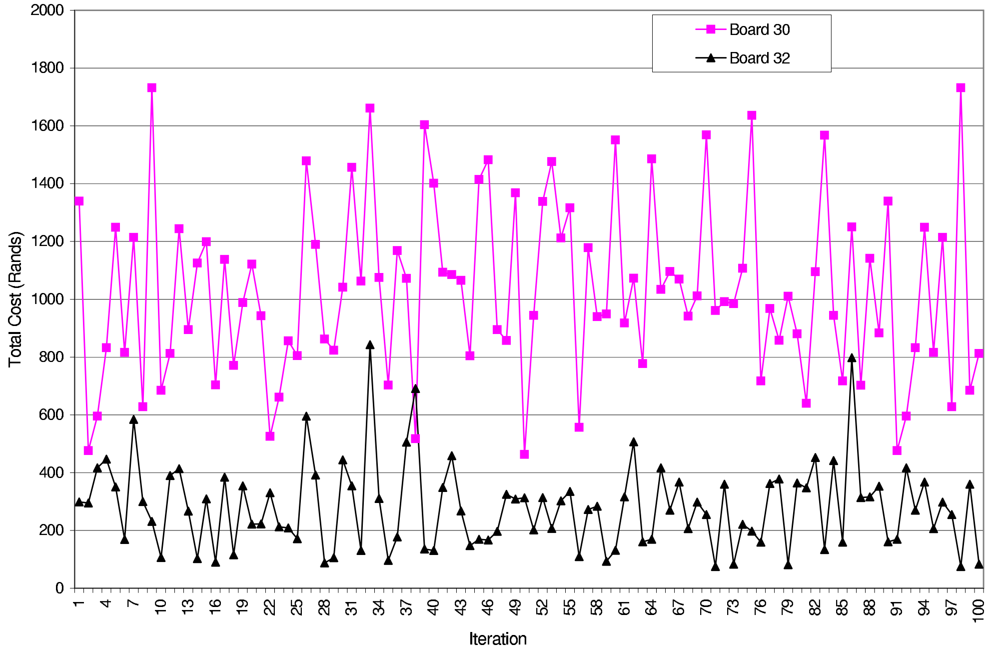


(a) Holding Cost as a function of time over all boards

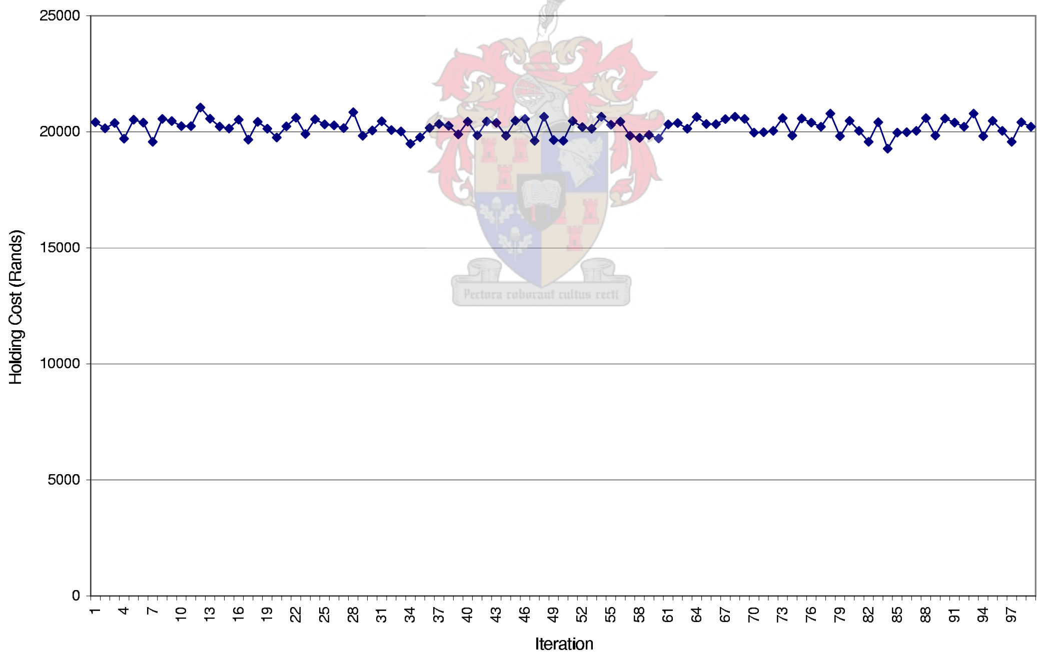


(b) Purchasing Cost as a function of time over all boards

Figure 6.6: *Holding and purchasing costs for five different initial stock levels: 0.9I, 0.95I, I, 1.05I and 1.1I, where I represents half of the order–to value. This demonstrates continuity in the simulation model.*



(a) Minimum total period cost for 100 iterations of the simulation model for boards 30 and 32



(b) Holding Cost over all boards for 100 iterations

Figure 6.7: *The consistency of the model is shown in that the minimum total period costs for boards and the total holding cost fluctuate within a restricted range, over 100 iterations in which only the random generator seed was changed.*

magnitude larger than the standard deviation, which indicates that the expected value is a good measure of expectation, and demonstrates the consistency of the model.

6.2.3 Degeneracy

The non–degeneracy of the model was verified by the observation that a reduction in the number of different board types kept in stock results in an increase in the shortage cost incurred. This was tested by progressively decreasing the number of boards kept in inventory (achieved programatically by setting the stock level of certain stock boards to zero), and running the single period optimisation model to calculate the single period shortage costs incurred each time. The results, illustrating the non–degeneracy of the model, are shown in Figure 6.8.

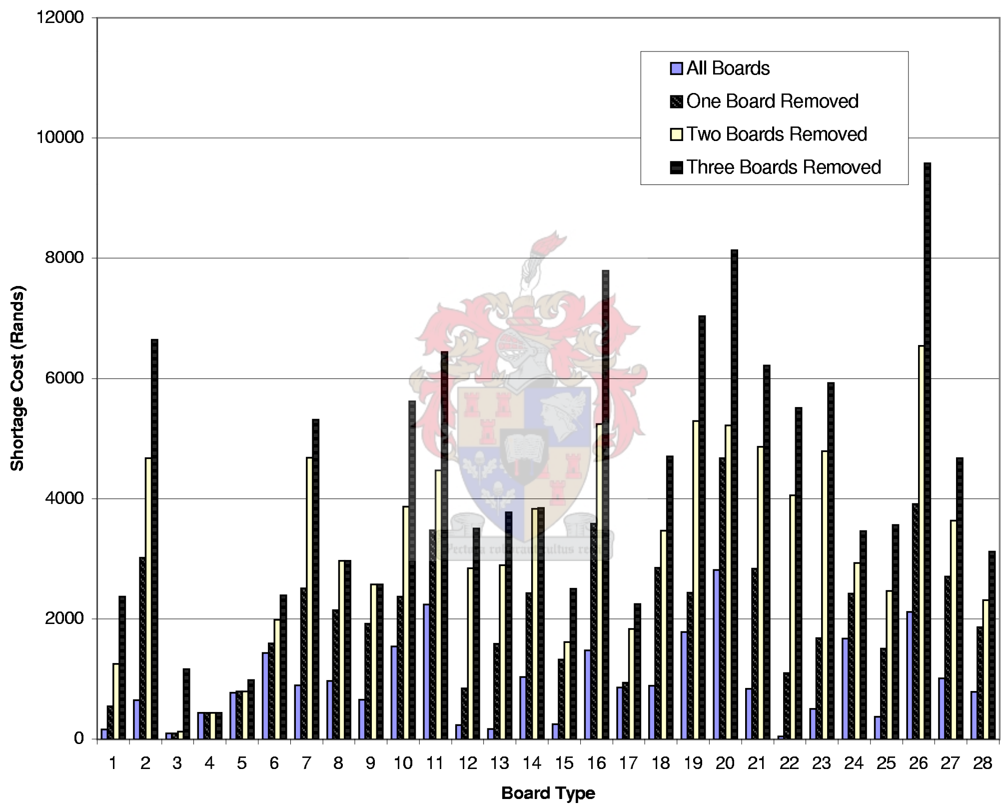


Figure 6.8: *Illustration of the non–degeneracy of the simulation model. Note that the single period shortage cost increases with the reduction of the number of board types available.*

6.2.4 Absurd Conditions

Conditions were set in the programming of the model to ensure that variables were not permitted to vary outside of their defined range. Particularly, the probability realisations are strictly controlled by "if...then" statements[4] so that only the values specified may occur. All variables are initialised to zero, or an appropriate initial value, and terminating conditions are set on all loops. Furthermore, data types are specified in the Access tables, so all data stored or retrieved is done so in a set format (*i.e.* integer, decimal with a set number of decimal places *etc.*).

6.3 Determination of Simulation Truncation Point

An important feature of a non–terminating stochastic model is the presence of a transient phase, which introduces bias into the statistics [10]. This phenomenon is accommodated by determining a truncation point, at which the transient phase ends and the steady state behaviour of the system begins. All data that had been collected up to the truncation point was discarded to eliminate bias. The truncation point was determined by means of the well–known Welch method, which is outlined below.

1. Perform n replications ($n > 5$) of length m of the simulation, where the length represents the number of periods into the future for which costs are calculated. Define X_{ji} as the i–th observation in replication j for all $j = 1, 2, \ldots, n$ and all $i = 1, 2, \ldots, m$.

2. Determine the averages $\overline{X_i} = \sum_{j=1}^{n} \frac{X_{ji}}{n}$ for $i = 1, 2, \ldots, m$. The averaged process reduces the variance to $\frac{1}{n}$ of the original variance.

3. Determine the moving average $\overline{X}_i(w)$, where w is the window length such that $w \leq \frac{m}{2}$, as

$$\overline{X}_i(w) = \begin{cases} \frac{\sum_{s=-w}^{w} \overline{X}_{i+s}}{2w+1} & \text{if} \quad i = w+1, \ldots, m-w \\ \frac{\sum_{s=-(i-1)}^{i-1} \overline{X}_{i+s}}{2i-1} & \text{if} \quad i = 1, \ldots, w. \end{cases} \tag{6.1}$$

4. Plot $\overline{X}_i(w)$ for $i = 1, 2, \ldots, m - w$, and choose the truncation point to be the value of i beyond which $\overline{X}_i(w)$ appears to have converged.

The Welch method was applied to the simulation model, where the values of X_{ji} represent the value of the total cost function (5.33) in each observation i (made in subsequent weeks) of each replication j of the model. Step 4 of the Welch method is illustrated in Figure 6.9, showing the system to enter into a steady state in week four. The simulation model was therefore run for four time periods before the calculation of period costs, for both the single and multiple period solutions.

[4] The program code referenced is contained in Appendix C.6.

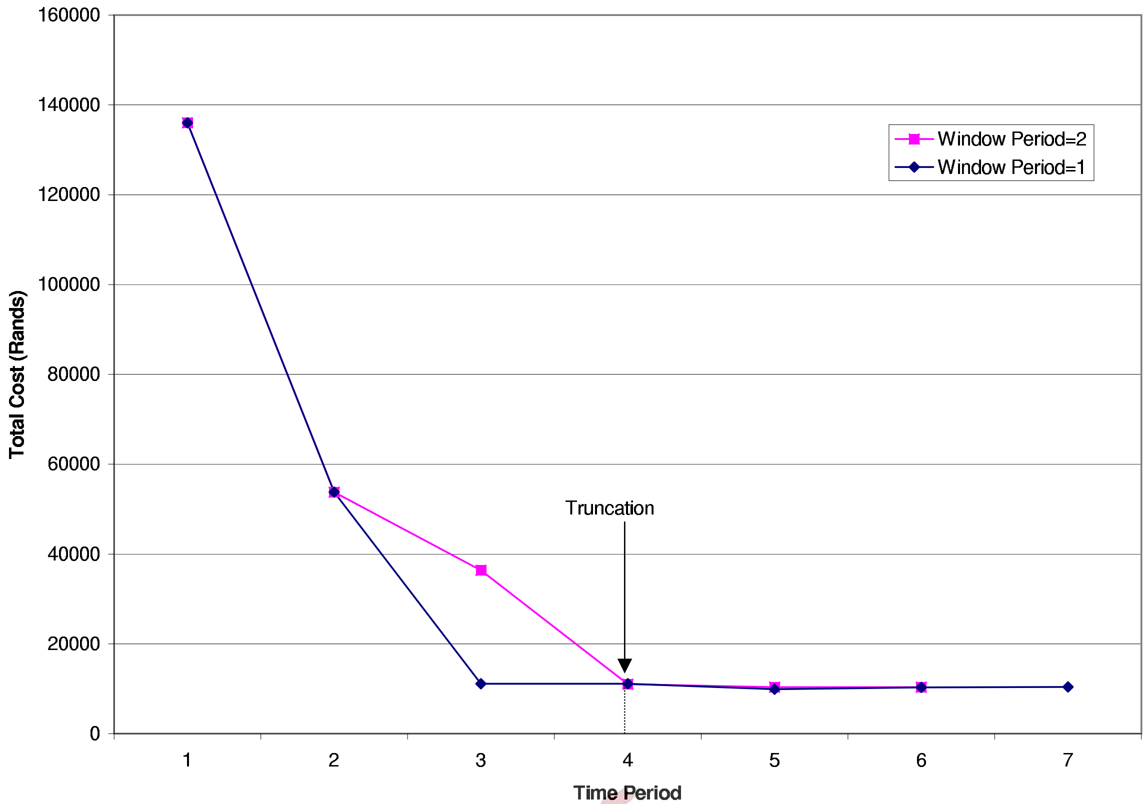


Figure 6.9: *Determination of the truncation point, at which the system enters into a steady state, by means of the Welch method. The total cost function in (5.33) is shown to converge to a steady state, for both a window period of one and two weeks, by week four.*

6.4 Single Period Optimisation Results

The simulation model was run for 1 000 iterations. An iteration is a simulation of demand for a four week time period. In each iteration the total cost incurred in the fourth week of the simulation was calculated (for each stock board), for each set of re–order point and order–to levels. The fourth week was used as it was shown in §6.3 that it is in the fourth period that the system enters into a steady state. The steady state was attained by stepping four times through the inventory control process, updating data pertaining to demand state, inventory levels *etc.*, as described in §5.6.4, by application of equations (5.29), (5.30) and (5.36). The expected period cost for the fourth week was then calculated by equation (5.32). The expected period cost for each set of re–order point and order–to levels was averaged over all iterations, and those parameters that minimise, on average, the expected total cost in the fourth week were selected. The results of this simulation, presented as a suggested re–order point and order–to level pair for each stock board, are shown in Table 6.1. The simulation was run on a Pentium I, with a processing speed of 400MHz and 32MB of RAM, and each iteration took approximately 15 minutes to execute.

Board Type	Re–order Point	Order–to Level
AC 1 030 × 2 370	800	1 200
AC 1 260 × 2 300	900	1 600
AC 1 280 × 1 300	900	1 500
AC 1 330 × 2 370	800	1 100
AC 1 360 × 2 300	1 000	1 600
AC 1 380 × 1 310	1 000	1 700
AC 1 460 × 2 370	800	1 600
AC 1 470 × 1 480	1 000	1 500
AC 1 500 × 1 540	900	1 100
AC 1 510 × 1 810	100	200
AC 1 530 × 1 380	900	1 700
AC 1 550 × 1 020	1 000	1 500
AC 1 680 × 1 080	1 400	1 800
AC 1 720 × 1 210	900	1 200
AC 1 800 × 1 200	1 000	1 200
AC 1 860 × 1 000	900	1 700
AC 1 860 × 1 490	800	1 100
AC 1 910 × 1 880	800	1 300
AC 2 000 × 1 400	800	1 600
AC 2 030 × 1 240	1 000	1 400
AC 2 110 × 1 010	1 100	1 600
AC 2 110 × 1 680	1 200	1 600
AC 2 200 × 1 200	800	1 700
AC 2 260 × 1 520	800	1 500
AC 2 260 × 2 160	1 000	1 400
AC 2 300 × 1 220	1 000	1 400
AC 2 300 × 1 710	800	1 000
AC 2 370 × 1 250	200	900

(a) AC Boards

Board Type	Re–order Point	Order–to Level
DWB 1 480 × 1 530	100	200
DWB 1 780 × 2 150	700	900
DWB 1 820 × 2 300	600	800
DWB 1 050 × 1 820	300	700
DWB 1 230 × 1 270	600	900
DWB 1 270 × 1 410	200	600
DWB 1 170 × 1 230	500	600
DWB 1 410 × 1 820	400	800
DWB 1 530 × 1 780	700	900
DWB 1 670 × 2 030	100	200
DWB 2 010 × 2 150	300	700
DWB 2 150 × 2 410	400	900
DWB 2 030 × 2 270	500	600
DWB 2 330 × 1 820	200	700
DWB 1 870 × 2 010	100	200
DWB 2 270 × 2 410	400	800
DWB 2 410 × 1 270	100	700
DWB 2 300 × 2 410	400	900

(b) DWB Boards

Table 6.1: *Results of the single period optimisation, showing the re–order point and order–to level for each stock board that minimises total cost in (5.32).*

6.5 Multiple Period Optimisation Results

The second objective was to provide a facility by which to obtain a dynamic set of replenishment parameters for each board, for multiple periods into the future. The model takes as input the current inventory position and the number of look–ahead periods required. The results presented in this section were obtained from a simulation run for a four week period. The output of the simulation run is a set of replenishment parameters for each stock board which minimises the one period look ahead cost function in (5.32) for each of four successive weeks. These calculations are based on the assumption that the other boards are replenished according to the stationary replenishment parameters calculated for each board in §6.4. This assumption was necessary as a result of the computational complexity of the problem, as the expected cost of each set of replenishment parameters for each board type is dependant on the replenishment parameters for all other board types. The viable alternatives were to calculate a set of replenishment parameters for each week to be applied uniformly to all board types, or to calculate the optimal dynamic replenishment parameters for each board type individually, whilst assuming some static re–order and order–to values for all other board types. The latter option was selected as being more practical, and the static re–order and order–to values taken to be the parameters calculated in §6.4.

The initial inventory positions and demand states were set, and then the set of replenishment parameters that minimise the one period look ahead cost function was selected. The system then advanced to the next demand state, and inventory levels were adjusted, based on these optimal replenishment parameters. Expected cost for the following period was then calculated for all potential replenishment parameters. The replenishment parameters that minimise expected total cost for the following week were selected, and the cycle was repeated for four weeks. The simulation was again run on a Pentium I, with a processing speed of 400MHz and 32MB of RAM, and each step (generating the results for one board type) took approximately 3.5 hours to execute.

The resulting set of suggested weekly replenishment parameters for one AC stock board type and one DWB stock board type are given in Table 6.2(a) and (b), and illustrated in Figure 6.10(a) and (b). The results for all board types are given in Appendix F.

Week	Re–order Point	Order–to Level
1	500	900
2	700	1 200
3	600	1 400
4	700	1 000

(a) Board Type 11 (AC)

Week	Re–order Point	Order–to Level
1	200	500
2	100	200
3	200	300
4	100	200

(b) Board Type 46 (DWB)

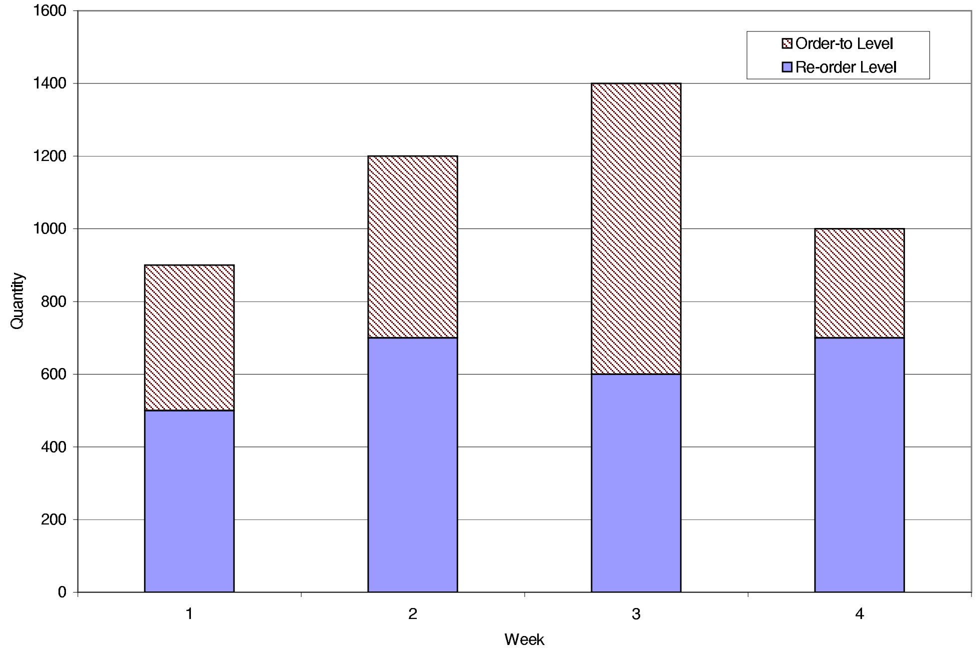
Table 6.2: *Results of the multiple period optimisation, showing the optimal re–order point and order–to level for each of four successive weeks for board types 11 (an AC stock board) and 46 (a DWB stock board).*

This model can be run for any time period from any given week. Data from the previous week's demand — that is inventory levels, stock on order, and demand state — may be provided as input to the model, and then the model may be solved to generate a set of suggested replenishment parameters for the selected number of periods ahead. This is done for each board under the assumption that the other boards are replenished according to the stationary replenishment parameters given in Table 6.1 (a) and (b).

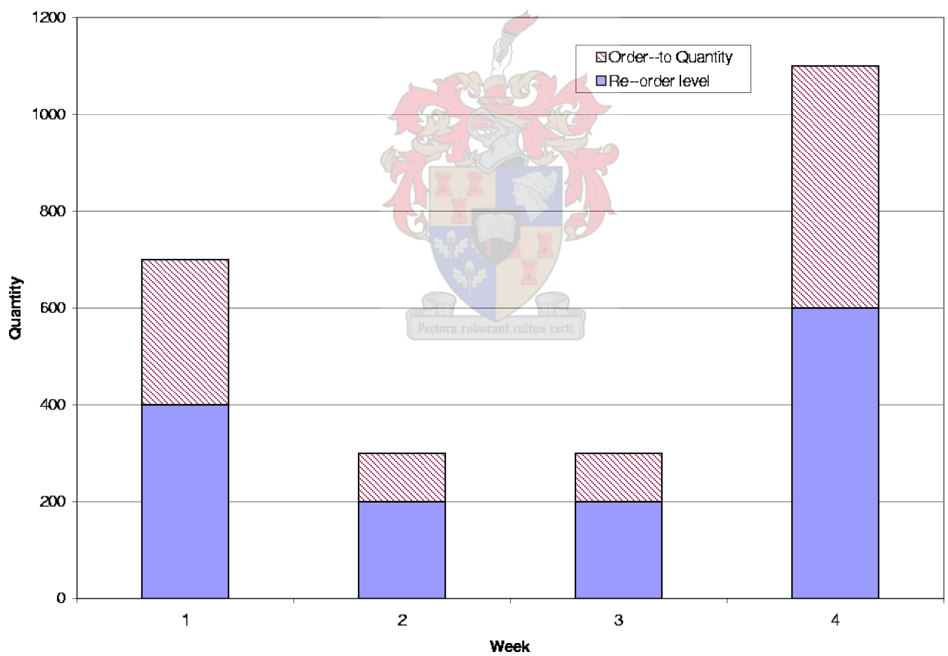
6.6 Evaluation of Results

The average theoretical service levels established in §5.5 were 87.61% for AC boards and 90.68% for DWB boards. The expected service levels, based on the single period optimisation, are 99.67% for AC boards and 91.61% for DWB boards. These service levels represent the percentage of orders expected to be met by stock boards with less than 15% wastage. The expected average service level over all board types is 96.52%. The model results therefore conform to the objectives specified in §1.2, for the set of random orders generated in the simulation.

Two estimates of the expected value of stock in inventory were used to gauge the difference in expected stockholding between the previous and suggested replenishment policies. The



(a) Board Type 7 (AC)



(b) Board Type 29 (DWB)

Figure 6.10: *Suggested dynamic replenishment parameters for one AC and one DWB stock board, giving the re–order and order–to value for each of four successive weeks.*

first was the midpoint between the re–order point and the order–to level, measuring the average expected stock level, and the second was the order–to level, measuring the maximum stock levels. The expected stockholding based on the stationary replenishment values given in Table 6.1 (a) and (b) was then compared to the expected stockholding based on the previous replenishment policy discussed in §3.5.1. The results were a 38% reduction in average stock levels for AC board types and a 22% reduction in the average stock levels for DWB board types.

6.7 Chapter Summary

The implementation of the inventory model derived in Chapter 5 was described in this chapter. The components of the model were discussed in §6.1, and the validation of the model under four key conditions was detailed in §6.2. The determination of the truncation point of the transient phase of the simulation by means of the Welch method was discussed in §6.3. The results of the model, comprising a stationary suggested set of replenishment parameters for each board type, and a dynamic, multiple period, set of replenishment parameters to be used for each board type, were then given in §6.4 and §6.5 respectively. The results of the model were evaluated in §6.6 with respect to previous and current practice at the factory. The model inventory was found to perform at an average expected service level of 96.52% and at a reduction in stockholding of approximately 28%.

Chapter 7

Conclusion

The problem of inventory control in an environment characterised by non–stationary, partially observed demand and cascading product substitution was considered in this thesis. This is a combination of factors which, to the author's knowledge, has not yet been dealt with in the literature. The inventory model developed was implemented at *Clickabox*, a cardboard box manufacturing company in the South African Western Cape. The objective of this model was to minimise stock holding costs, subject to a service level of 95% of orders being met by suitable stock boards in inventory, and to a maximum raw material offcut wastage of 15% of the stock board ordered. The deliverables of the study were a suggested stock profile to be kept in inventory, and a computerised decision support system to aid in future replenishment decisions at *Clickabox*.

7.1 Summary of What Has Been Achieved

Apart from the introductory chapter, in which *Clickabox* factory was introduced informally and an informal problem description given, and this chapter (the conclusion) this thesis comprises a further five chapters.

Chapter 2 contains a brief overview of the vast body of inventory control literature available, paying particular attention to the nature of the demand process in models. The inventory control literature stemming from the foundational work of Arrow, *et al.* (1951, [3]) and Karlin and Scarf (1958, [39]) is categorized according to the type of demand studied, that is stationary or non–stationary, and whether it is fully or partially observed. A number of important concepts in inventory modelling, related to the situation at *Clickabox*, were also discussed. These are the concept of Markovian–modulated demand, cascading product substitution, lead time, the handling of stockout situations, and the value of advance demand information.

Chapter 3 contains a detailed examination of *Clickabox* factory. The layout, products and processes followed at the factory were described, in order to provide the reader with an understanding of the environment in which the study was conducted. The financial situation and business objectives of the company were discussed, highlighting the importance of efficient inventory control practices at the factory.

The focus of Chapter 4 was on the demand process at *Clickabox*. The demand data available were described, and then, based on these historical data, an investigation was conducted into a suitable stock profile to be kept in inventory. This was acheived by conducting an ABC analysis on the different cardboard types, through which the two cardboard types of the highest financial importance were identified. Restrictions on the dimensions of stock board types were established, and a heuristic was developed to obtain a suitable set of stock boards to be kept in inventory, based on historical demand. This suggested stock profile has been implemented with success at *Clickabox*. The concept of a board preference vector was then introduced, in order to incorporate the cascading product substitution that occurs at Clickabox into the modelling of the demand process. The board preference vectors were determined, and an analysis of the demand for each of these board preference vectors was conducted. Based on this analysis, demand was categorised into classes, and the various probability distributions required for the modelling of the demand process were derived.

Chapter 5 opened with the introduction of a number of general modelling assumptions, a description of the spatial constraints at the warehouse, and a discussion of the various inventory costs. A separate section was devoted to an investigation into service level measures, in which theoretically appropriate service levels were calculated for each stock board at *Clickabox*. A theoretical optimal control policy was then derived for the case of non–stationary, partially observed demand, modelled as a finite state Markov Chain. Finally, a sub–optimal control policy, which is more practical with respect to computational requirements, was developed. The objective of this policy is the minimization of expected tied–up inventory capital subject to an acceptable level of offcut wastage costs, whilst satisfying the given service level requirements.

The inventory model derived in Chapter 5 was then implemented in the form of a simulation model. The structure and results of the simulation model were described in Chapter 6. A number of conditions were tested in order to establish the validity of the simulation model. The truncation point, at which the simulation enters into a steady state, was also determined. Two major results obtained by the simulation model were presented.

The first was a stationary set of suggested replenishment parameters for each board type. These suggested replenishment parameters were obtained by repeatedly bringing the system into a steady state and calculating the expected single period cost of each potential set of replenishment parameters, for each board. The result is, for each board, the re–order level and order–to point that obtain, on average, the lowest expected single period total cost. These results are given in Table 6.1.

The second result was a dynamic, multiple period set of replenishment parameters, for each board type. These are the re–order levels and order–to points for each stock board which minimise the one period look–ahead cost function for each of four successive weeks. The model takes as input the current inventory position and the number of look–ahead periods required. The results, as presented in Table 6.2 and Figure 6.5, were obtained from simulation runs for a four week period.

These results allow for two possible implementations. Implementation of the stationary replenishment policy, where the re–order level and order–to point for each board calculated in §6.5 are used during each period, is a simple option, which minimises expected total cost on average. The parameters may be re–calculated, say on a bi–annual basis, to allow for changes in customer requirements. The second option is the dynamic replenishment policy, in which the replenishment parameters are calculated for a specified number of periods in advance, such as for a month in advance. This policy may also be re–calculated as new customer requirements become known.

Finally, the results obtained were evaluated, in terms of the objectives specified in §1.2, and the effect of the suggested replenishment policy on expected inventory levels was discussed. The inventory model was found to perform at an average expected service level of 96.52% of orders met with a raw material offcut wastage of less than 15% of the board ordered, and at a reduction in stockholding of approximately 28%, based on data for the period 1 February 2001 — 31 January 2003.

7.2 Further Work

In view of current levels of inflation, it would be valuable to consider its effect on inventory policies. Wirth (1987, [82]) discussed a model in which the cost of capital is reduced by the rate of inflation to give a more accurate discount rate to be used in the calculation of the optimal re–order quantity.

The approach taken in this thesis was to determine a single, static, stock profile, and then a replenishment policy based on this profile. Due to the cyclical behaviour of certain sources of demand, such as demand for the packaging of agricultural produce, it would be beneficial to consider a policy that allows for dynamic stock board selection. Such a strategy would involve determining an optimal stock board profile at any stage, and then selecting a replenishment policy based on that stock board profile. It would, however, also be necessary to take into account the logistical issues arising from a changing stock profile.

Another area of potential study, an investigation into the expected value of advance demand information, was mentioned in §2.2.5. This is, in fact, an area that has been highlighted by the director of *Clickabox* as a business need [70], as it would enable the company to develop an appropriate reward scheme for customers with regular ordering patterns.

The existence of supplier–imposed minimum order quantity specifications was mentioned in §3.5. These restrictions are specific to the current supplier and subject to change, and so were not included in the model. However, it is noted as an area of potential further study, particularly as a means to compare the benefits of two or more suppliers with different restrictions, in terms of expected penalty costs and stockholding.

Appendix A

Suggested Stock Boards

Stock Board Type	Sub–optimal Board Type	Board factor	Wastage Percentage	Board Volume (m^3)
AC 1 030 × 2 370	AC 1 330 × 2 370	1	22.56%	1.074×10^{-3}
AC 1 260 × 2 300	AC 1 360 × 2 300	1	7.35%	1.275×10^{-3}
AC 1 280 × 1 300	AC 1 380 × 1 310	1	7.95%	7.322×10^{-4}
AC 1 330 × 2 370	AC 1 460 × 2 370	1	8.90%	1.387×10^{-3}
AC 1 360 × 2 300	AC 1 460 × 2 370	1	9.60%	1.376×10^{-3}
AC 1 380 × 1 310	AC 1 530 × 1 380	1	14.38%	7.954×10^{-4}
AC 1 460 × 2 370	AC 1 720 × 1 210	2	16.87%	1.522×10^{-3}
AC 1 470 × 1 480	AC 1 500 × 1 540	1	5.82%	9.573×10^{-4}
AC 1 500 × 1 540	AC 1 510 × 1 810	1	15.48%	1.016×10^{-3}
AC 1 510 × 1 810	AC 1 910 × 1 880	1	23.89%	1.203×10^{-3}
AC 1 530 × 1 380	AC 1 860 × 1 490	1	23.81%	9.290×10^{-4}
AC 1 550 × 1 020	AC 1 680 × 1 080	1	12.86%	6.956×10^{-4}
AC 1 680 × 1 080	AC 1 720 × 1 210	1	12.82%	7.983×10^{-4}
AC 1 720 × 1 210	AC 2 030 × 1 240	1	17.32%	9.157×10^{-4}
AC 1 800 × 1 200	AC 2 030 × 1 240	1	14.19%	9.504×10^{-4}
AC 1 860 × 1 000	AC 2 110 × 1 010	1	12.72%	8.184×10^{-4}
AC 1 860 × 1 490	AC 2 260 × 1 520	1	19.32%	1.219×10^{-3}
AC 1 910 × 1 880	AC 2 260 × 2 160	1	26.44%	1.580×10^{-3}
AC 2 000 × 1 400	AC 2 260 × 1 520	1	18.49%	1.232×10^{-3}
AC 2 030 × 1 240	AC 2 370 × 1 250	1	15.03%	1.108×10^{-3}
AC 2 110 × 1 010	AC 2 200 × 1 200	1	19.28%	9.377×10^{-4}
AC 2 110 × 1 680	AC 2 300 × 1 710	1	9.87%	1.560×10^{-3}
AC 2 200 × 1 200	AC 2 300 × 1 220	1	5.92%	1.162×10^{-3}
AC 2 260 × 1 520	AC 2 300 × 1 710	1	12.66%	1.511×10^{-3}
AC 2 260 × 2 160	AC 2 300 × 1 710	2	37.94%	2.148×10^{-3}
AC 2 300 × 1 220	AC 2 370 × 1 250	1	5.28%	1.235×10^{-3}
AC 2 300 × 1 710	AC 1 510 × 1 810	2	28.05%	1.731×10^{-3}
AC 2 370 × 1 250	AC 1 280 × 1 300	2	10.98%	1.304×10^{-3}

Table A.1: *AC Stock Boards and the corresponding sub–optimal board types, which would be used to meet orders for that board type if there were no stock available of the optimal board type, for the purposes of the calculation of theoretical service levels in §5.5. Also given are the board–to–board conversion factors,* i.e. *the number of the sub–optimal board type that are required to produce a board of the optimal board type dimensions, and the wastage incurred when the sub–optimal board type is used.*

Stock Board Type	Sub–optimal Board Type	Board factor	Wastage Percentage	Board Volume (m^3)
DWB 1 480 × 1 530	DWB 1 530 × 1 370	1	7.50%	1.720×10^{-3}
DWB 1 780 × 2 150	DWB 2 150 × 1 640	1	18.22%	2.908×10^{-3}
DWB 1 820 × 2 300	DWB 2 300 × 2 180	1	24.14%	3.181×10^{-3}
DWB 1 050 × 1 820	DWB 1 820 × 2 090	1	45.34%	1.452×10^{-3}
DWB 1 230 × 1 270	DWB 1 270 × 1 700	1	19.10%	1.187×10^{-3}
DWB 1 270 × 1 410	DWB 1 410 × 1 940	1	21.07%	1.360×10^{-3}
DWB 1 170 × 1 230	DWB 1 230 × 1 420	1	12.25%	1.093×10^{-3}
DWB 1 410 × 1 820	DWB 1 820 × 2 090	1	28.09%	1.950×10^{-3}
DWB 1 530 × 1 780	DWB 1 780 × 1 620	1	27.31%	2.069×10^{-3}
DWB 1 670 × 2 030	DWB 2 030 × 1 080	1	23.07%	2.576×10^{-3}
DWB 2 010 × 2 150	DWB 2 150 × 1 640	1	16.77%	3.284×10^{-3}
DWB 2 150 × 2 410	DWB 2 410 × 1 690	1	13.43%	3.937×10^{-3}
DWB 2 030 × 2 270	DWB 2 270 × 1 430	1	32.46%	3.502×10^{-3}
DWB 2 330 × 1 820	DWB 1 820 × 2 090	2	38.75%	3.222×10^{-3}
DWB 1 870 × 2 010	DWB 2 010 × 1 460	1	13.97%	2.856×10^{-3}
DWB 2 270 × 2 410	DWB 2 410 × 1 690	1	20.30%	4.157×10^{-3}
DWB 2 410 × 1 270	DWB 1 270 × 1 700	2	5.68%	2.326×10^{-3}
DWB 2 300 × 2 410	DWB 2 410 × 1 690	2	38.45%	4.212×10^{-3}

Table A.2: *DWB Stock Boards and the corresponding sub–optimal board types, which would be used to meet orders for that board type if there were no stock available of the optimal board type, for the purposes of the calculation of theoretical service levels in §5.5. Also given are the board–to–board conversion factors,* i.e. *the number of the sub–optimal board type that are required to produce a board of the optimal board type dimensions, and the wastage incurred when the sub–optimal board type is used.*

Appendix B

Realised Demand and Shortage Cost

This appendix contains the algorithm used in the calculation of realised demand and shortage costs, as referenced in §5.6.3. This procedure is followed each week for each board β for which the following condition holds: $\sum_{v_{i,1}=\beta} w_t^{\underline{v}_i}/m_t^{(i,\beta)} < u_t^{(\beta)}$. In other words, the shortage cost is calculated for each board for which there is not sufficient stock on hand in the current period to fulfil all demand for the board preference vectors in which it is the optimal board.

The following procedure is conducted for each board β, during each week t.

Step 1 : *Determine the demand to be met with the best board, that is, the orders that would incur the highest cost should the second best board be used instead of the optimal board.* Select the board preference vector $\underline{v}_i$ for which the following conditions hold:

$$\begin{aligned} v_{i,1} &= \beta, \\ \phi_{i,2} &= \max\left\{\phi_{j,2} : j \in \mathcal{V}\right\}, \\ w_t^{\underline{v}_i} &> 0. \end{aligned}$$

Now set $w^{\underline{v}_i} \leftarrow \max\left\{w_t^{\underline{v}_i} - m_t^{(i,\beta)} u_t^{(\beta)}, 0\right\}$ and $u_t^{(\beta)} \leftarrow \max\left\{u_t^{(\beta)} - w_t^{\underline{v}_i}/m_t^{(i,\beta)}, 0\right\}$. Repeat this sequence until $u_t^{(\beta)} = 0$ or $\sum_{v_{i,1}=\beta} w_t^{\underline{v}_i}/m_t^{(i,\beta)} = 0$.

Step 2 : *Determine the demand to be met with the second best board, that is, the orders that would incur the highest cost should the third best board be used instead of the second best board.* This step is executed for each of the second best boards, in board preference vectors with a positive demand, which have β as the optimal board. In other words, each board j is selected for which the following conditions hold:

$$\left.\begin{aligned} v_{i,1} &= \beta, \\ v_{i,2} &= j, \\ w_t^{\underline{v}_i} &> 0. \end{aligned}\right\} \tag{B.1}$$

Execute the following for each such board j:

a. Set $u_t^{(j)} \leftarrow \max\left\{u_t^{(j)} - \sum_{v_{i,1}=j} w_t^{\underline{v}_i}/m_t^{(i,\beta)}, 0\right\}$, as board j is first used to satisfy its first level demand before satisfying the second level demand of board β.

b. If $\sum_{v_{i,1}=\beta, v_{i,2}=j} w_t^{\underline{v}_i}/m_t^{(i,\beta)} \leq u_t^{(j)}$, all demand can be met with the second best board. Therefore the quantity of unmet demand is reduced, the inventory level of board k is reduced, and the shortage cost is set. Therefore set

$$\begin{aligned}
w_t^{\underline{v}_i} &\leftarrow w^{\underline{v}_i} - \sum_{v_{i,1}=\beta, v_{i,2}=j} w_t^{\underline{v}_i}/m_t^{(i,\beta)}, \\
u_t^{(j)} &\leftarrow u_t^{(j)} - \sum_{v_{i,1}=\beta, v_{i,2}=j} w_t^{\underline{v}_i}/m_t^{(i,\beta)}, \\
\Psi^{(\beta)} &\leftarrow \Psi^{(\beta)} + \phi_{i,2}\left\{\sum_{v_{i,1}=\beta, v_{i,2}=j} w_t^{\underline{v}_i}/m_t^{(i,\beta)}\right\}.
\end{aligned}$$

c. If $\sum_{v_{i,1}=\beta, v_{i,2}=j} w^{\underline{v}_i}/m_t^{(i,\beta)} > u_t^{(j)}$, not all demand can be met with the second best board. It must now be decided which of the demands to fill by using the second best board. This is done, as in step one, by selecting those orders which would incur the highest shortage cost, should it be neccesary to use the third best board. The board preference vector $\underline{v}_i$ for which the conditions in (B.1) hold, and $\phi_{i,3} = \max\{\phi_{i,3} : i \in \mathcal{V}\}$, is selected. The unmet demand and inventory levels are reduced accordingly, and the shortage cost set:

$$\begin{aligned}
w_t^{\underline{v}_i} &\leftarrow \max\left\{w_t^{\underline{v}_i} - m_t^{(i,\beta)} u_t^{(j)}, 0\right\}, \\
u^{(k)} &\leftarrow \max\left\{u^{(j)} - w_t^{\underline{v}_i}/m_t^{(i,\beta)}, 0\right\}, \\
\Psi^{(\beta)} &\leftarrow \Psi^{(\beta)} + \phi_{i,2} \min\left\{w_t^{\underline{v}_i}/m_t^{(i,\beta)}, u_t^{(j)}\right\}.
\end{aligned}$$

Step 2(c) is repeated until $u_t^{(j)} = 0$, at which stage the third best board must then be used to satisfy the remainder of the demand, or

$$\sum_{v_{i,1}=\beta, v_{i,2}=j} w_t^{\underline{v}_i}/m_t^{(i,\beta)} = 0, \tag{B.2}$$

in which case the demand for all board preference vectors where $v_{i,1} = \beta$ and $v_{i,2} = j$ has been filled.

Step 3 : *Determine the wastage from the demand met with the third best board.* This is done following the approach outlined in step 2, but using the third level wastage cost $\phi_{(i,3)}$ instead. In order to avoid repetition the formulaes will not be given here again, but the procedure is similar to that of step 2.

Appendix C

Program Code

This appendix contains the code from the Visual Basic [50] programs written by the author for the analysis of demand data. The program references and ammends data stored in a Microsoft Access [49] database. Each section of this chapter contains a procedure or set of procedures written for a specific purpose, as referenced in the body of the thesis.

The program written to find the optimal set of stock boards is given in §C.1. In this procedure a loop through all possible board dimensions is executed, in order to find a board that could produce the required percentage of historical orders, with the minimum wastage. The board is then added to the table of stock boards, and the loop executed again, until the required number of stock boards has been found. The following section, §C.2, contains the program used to determine the set of sub–optimal stock boards used in the service level calculation in Chapter 5. The code written to calculate the optimal board to use for a sheet order is given in §C.3. This is used in determining retrospective wastage over all the historical data for a given set of boards, in order to compare the performance of different sets of boards in terms of wastage and the number of orders met. It is also used to determine the board preference vector for each sheet order. An iterative process, similar to that in §C.1, is followed, looping through each board and calculating the values of certain decision parameters.

Section C.4 contains the code written to extract the values of the transition probabilities required for the inventory model, as detailed in §4.3.2.4. The code used to determine sheet–to–board conversion factor probabilities, according to the process detailed in §4.3.2.5, is given in §C.5.

The code behind the single and multiple period simulation models is then given in §C.6.

A number of tables in the database are referenced in the code. Two tables store the historical data of orders placed (each cardboard type stored in a separate table), which is updated to indicate the board used to produce each order. A further two tables store the suggested stock boards, and two store the current stock boards, again each board type is stored in a separate table. The structure of the two tables containing orders is identical, they are separated to reduce processing time, and similarly for the four tables containing the suggested stock board dimensions. The tables are therefore referred to in the code as 'Orders' and 'Stock', this was adapted when necessary to refer to the specific tables, *i.e.*

'ACOrders' or 'DWBOrders' and 'SugACStock', 'SugDWBStock', 'CurACStock', and 'CurDWBStock'. A further two tables are the 'ACVectors' and 'DWBVectors tables, containing the information pertaining to the board preference vectors of each board type, referred to in the code as 'Vectors'. Another tables referenced frequently is the transition probability table. The structure of these tables are given in Figure C. Various other tables, in which results are stored, are introduced in the code, and described where necessary.

C.1 Determination of an Optimal Stock Profile

This program determines the set of optimal stock boards, for either the 'AC' or 'DWB' board type. It takes as a parameter the number of stock boards to be found.

```
Private Sub FindStock(numstock)

'Define the connection to the Access database.
strcon = "DSN=Boards;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")


'Set the parameters of the search.
'Get the number of boards already found.
strsql = "SELECT count(id) AS cnt FROM SugStock"
rst.Open strsql, conndb
brdsfound = rst("cnt")
rst.Close
'The maximum length and width of a stock board are predetermined constants.
maxl = 2 490
maxw = 2 370
'The search is started at 0 length and width, and increased in steps of 10.
minl = 0
minw = 0
step = 10


'Repeat this loop until the required number of stock boards have been found.
Do While brdsfound < numstock 'Count the number of orders in the set for which no suitable stock
board from which the order can be produced has been found.
     strsql = "SELECT count(*) AS unmade FROM Orders WHERE board1 is null"
     rst.Open strsql, conndb
     unmade = rst("unmade")
     rst.Close
'Calculate the approximate number of orders that must be met by each stock board. From this number,
a threshold is set to represent the maximum number of 'un–makeable' orders for each stock board. An
order is termed 'un–makeable' if either dimension of the order is greater than those of the potential stock
board. Once this threshold is exceeded, the potential stock board is rejected, as it is impossible to produce
the required allocation of board orders.
     NumPerBoard = unmade / (numstock - brdsfound)
'Initialise the wastage measure
     wastemin = -1
'Loop through the grid of co–ordinates representing potential stock boards, incrementing each dimension
progressively from the minimum to the maximum values.
```

```
     Do While lboard ≤ maxl
          wboard = minw
          Do While wboard ≤ maxw
‘‘Initialise the variables representing cumulative wastage and number of un–makeable orders for this potential stock board.
                WasteSum = 0
                unmade = 0
‘Get a list of all the orders in the set for which there are not yet suitable stock boards, from which the orders can be produced.
                strsql = "SELECT l,w,q FROM Orders WHERE board1 is null"
                rst.Open strsql, conndb
‘Initialise the boolean variable used to stop the loop if the maximum number of un–makeable orders has been reached.
                continue = True
‘The performance of this potential stock board is now calculated against the historic data, in terms of total wastage and number of un–makeable orders.
                Do While Not rst.EOF And continue
                     lsheet = rst(‘‘l") ‘The length of the sheet ordered.
                     wsheet = rst(‘‘w") ‘The width of the sheet ordered.
                     qsheet = rst(‘‘q") ‘The quantity of sheets ordered.
                     If lsheet > lboard Or wsheet > wboard Then
‘The order cannot be made from this potential stock board, so increase the number of un–makeable orders
                          invalid = invalid + 1
                          wasteboard = 0
                     Else
‘Call the GetFactor function (detailed below), in order to calculate the number of sheets that can be made out of each board.
                          m^(L) = GetFactor(lsheet, lboard)
                          m^(W) = GetFactor(wsheet, wboard)
                          m = m^(L) * m^(W)
‘Calculate the total wastage incurred when the board under investigation is used to produce this sheet.
                          g = (q / m) * (lboard * wboard - m * lsheet * wsheet)
                     End If
‘Sum the total waste for this board.
                     WasteSum = WasteSum + g
‘Test the conditions that determine whether the loop through all orders should be continued.
                     If invalid > maxinvalid Then
‘The maximum number of un–makeable sheets has been exceeded and the loop is terminated.
                          continue = False
                     ElseIf wastemin > -1 And WasteSum > wastemin Then
                          continue = False


‘It is not the first board being investigated, where wastemin = -1, and the cumulative wastage so far has exceeded a preceeding boards’ wastage, in other words a better board has already been found, so the loop testing this board should be terminated.
                     Else
                          continue = True ‘Continue the loop
                     End If
                     rst.MoveNext ‘Get the details of the next order in the set.
                Loop
‘All entries in the set have been examined, or a terminating condition has been reached, so the set is closed.
                rst.Close
                If (continue = True And (wastemin = -1)) Or ((WasteSum < wastemin) And
                (invalid < maxinvalid)) Then
```

‘*Either it is the first board being investigated or no better board has yet been found. The temporary parameters, representing the optimal board, are updated with the length, width, and cumulative wastage of this board.*

```
                wastemin = WasteSum
                templ = lboard
                tempw = wboard
            End If
```

‘*The width of the board being investigated is increased by 10*

```
            wboard = wboard + step
```

‘*The loop is executed again, until the maximum width has been reached.*

```
        Loop
```

‘*The length of the board being investigated is increased by 10*

```
        lboard = lboard + step
```

‘*The loop is executed again, until the maximum length has been reached.*

```
    Loop
```

‘***Add the best board found into the set of stock boards.***

```
    res = AddStock(templ, tempw)
```

‘*Increase the number of boards found.*

```
    brdsfound = brdsfound + 1
Loop
End Sub
```

This function, called by the above procedure, includes the board with the specified dimensions as a suggested stock board.

```
Private Function AddStock(L, W) As Integer
```

‘***Define the connection to the database***

```
strcon = ‘‘DSN=Boards;"
Set conndb = CreateObject(‘‘ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject(‘‘ADODB.recordset")
Set rst2 = CreateObject(‘‘ADODB.recordset")
```

‘***Add the specified board to the table containing the stock board dimensions.***

```
strQuery = "INSERT into Stock (l,w) values (" L "," W ")"
rst.Open strQuery, conndb
```

‘***Get the id of the newly entered board.***

```
strQuery = ‘‘SELECT id FROM Stock WHERE L= " L ‘‘ and w =" W
rst.Open strQuery, conndb
Id = rst(‘‘id")
rst.Close
```

‘***Update the table containing orders to indicate which orders should be produced by this board.***

```
strsql = "SELECT l,w,q FROM Orders WHERE board1 is null"
rst.Open strsql, conndb
```

‘***Loop through all orders not yet allocated to a board and test whether each order can be***

produced with the new stock board.

```
Do While Not rst.EOF
     lsheet = rst("l")
     wsheet = rst(‘‘w")
     If lsheet > lboard Or wsheet > wboard Then
```

‘*If either dimension of the order is greater than that of the grid point (board) then the order cant be made with this board.*

```
     Else
```

‘*Calculate the wastage.*

```
         boardfactor = GetFactor(lsheet, lboard, wsheet, wboard)
         percentagewaste = 100 * (lboard * wboard - boardfactor * lsheet * wsheet) /
         (lboard* wboard)
         wasteboard = (q / boardfactor) * (lboard * wboard - boardfactor * lsheet *
        wsheet)
```

‘*Check against the threshold percentage waste, set at 30%*

```
         If percentagewaste ≤ 30 Then
             strsql = "UPDATE Orders SET made = " Id , waste = " percentagewaste
             "WHERE l = " lsheet " and w = " wsheet
             rst2.Open strsql, conndb
        End If
     End If
     rst.MoveNext
Loop
rst.Close
AddStock = 1
End Function
```

This function, also called by the FindStock procedure, determines the number of sheets that can be produced using the board investigated.

```
Private Function GetFactor(Lsheet, Lboard, Wsheet, Wboard) As Integer
```

‘***The number of sheets that can be produced lengthwise from each board is calculated, rounding up to the nearest even number from 2 to 10, a process specified by the management at Clickabox.***

```
Lfactor = Lboard / Lsheet
If Lfactor < 2 Then Lfactor = 1
ElseIf Lfactor < 4 Then Lfactor = 2
ElseIf Lfactor < 6 Then Lfactor = 4
ElseIf Lfactor < 8 Then Lfactor = 6
ElseIf Lfactor < 10 Then Lfactor = 8
Else Lfactor = 10 End If
```

‘***Similarly, the number of sheets that can be produced widthwise from each board is calculated.***

```
Wfactor = Wboard / Wsheet
If Wfactor < 2 Then Wfactor = 1
ElseIf Wfactor < 4 Then Wfactor = 2
ElseIf Wfactor < 6 Then Wfactor = 4
ElseIf Wfactor < 8 Then Wfactor = 6
ElseIf Wfactor < 10 Then Wfactor = 8
Else
```

```
     Wfactor = 10
     boardfactor = Lfactor * Wfactor
End If
GetFactor = boardfactor
End Function
```

C.2 Determination of the Sub–optimal Board Types

This program determines the set of sub–optimal stock boards, for either the ‘AC’ or ‘DWB’ board type, to be used in the service level calculation.

```
Private Sub GetSubOptBoard()
```

‘***Define the connection to the database***

```
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
```

‘***Repeat this procedure for each stock board.***

```
strsql = "SELECT * FROM Stock"
rst.Open strsql, conndb
Do While Not rst.EOF
     optboard1 = rst(‘‘id")
     lopt = rst(‘‘l")
     wopt = rst(‘‘w")
     tempwaste = 9999999
     strsql = "SELECT * FROM Stock WHERE id ≠" optboard "and l ≥" lopt "and w ≥" wopt
     rst2.Open strsql, conndb
     Do While not rst2.eof
```

’*Get the waste incurred when this board is used instead of the optimal board.*

```
         waste = (lopt - rst2("l"))(wopt - rst2("w"))
         if waste ≤ tempwaste then
             tempid = rst2(‘‘id")
             tempwaste = waste
         end if
         rst2.movenext
     loop
     rst2.close
    if tempwaste=9999999 then
```

’*No suitable board has been found with both dimensions greater than those of the optimal board.*

```
         strsql = "SELECT * FROM Stock WHERE id ≠" optboard "and (l ≥" lopt "or w ≥
         " wopt )
         rst2.Open strsql, conndb
```

‘***Find sub–optimal board where one join is required.***

```
        Do While not rst2.eof
```

’*Get the waste incurred when this board is used instead of the optimal board.*

```
            if l ≥ lopt then waste = (lopt - rst2(‘‘l"))(2* rst2(‘‘w") - wopt)
```

```
            elseif w ≥ wopt then waste = (2*rst2(‘‘l") - lopt)(wopt - rst2(‘‘w"))
            endif
             if waste ≤ tempwaste then
                 tempid = rst2(‘‘id")
                 tempwaste = waste
             end if
             rst2.movenext
         loop
         rst2.close
         if tempwaste=9999999 then

 'A sub--optimal board must be found that requires a join in more than one direction.
             strsql = "SELECT * FROM Stock WHERE id ≠ " optboard
             rst2.Open strsql, conndb
             Do While not rst2.eof
                 waste = (2* rst2(‘‘l") - lopt)(2* rst2(‘‘w") - wopt)
                 if waste ≤ tempwaste then
                     tempid = rst2(‘‘id")
                     tempwaste = waste
                 end if
                 rst2.movenext
             loop
             rst2.close
         end if
     end if
     strsql = "INSERT into SubOpt values (" optboard1 "," tempid ")"
     rst2.open strsql, conndb
     rst.movenext
loop
rst.close
```

C.3 Calculation of the Optimal Board to use for a Sheet Order

This procedure finds the optimal stock board with which an order should be produced. There are two similar implementations of this procedure. The first, ‘GetBestBoard’, is used to compare the wastage and number of orders met for the current and suggested stock profiles. The second, ‘GetVectors’ is to find the board preference vectors for each order.

```
Private Sub GetBestBoard()
```

‘***Define the connection to the database***

```
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
```

‘***Get the set of sheets for which the optimal board must be found. This statement is adapted as required.***

```
strsql = "SELECT * FROM Orders WHERE board1 = 0 or board1 is null"
rst.Open strsql, conndb
Do While Not rst.EOF
     lsheet = rst("l")
     wsheet = rst("w")
‘Select all valid stock boards (from either the current or suggested stock board table)
     strsql = "SELECT * FROM Stock WHERE L ≥ " lsheet " and W ≥ " wsheet
     rst2.Open strsql, conndb
     continue = True
     First = True
     tempwaste = 0
     made = 0
‘Test each valid stock board.
     Do While continue And Not rst2.EOF
          lboard = rst2("L")
          wboard = rst2("W")
          board1 = rst2("id")
‘Calculate the number of sheets that can be made out of this board
          lFactor = GetFactor(lsheet, lboard)
          wfactor = GetFactor(wsheet, wboard)
          boardfactor = lFactor * wfactor
‘Calculate the percentage wastage incurred when this sheet is made out of this board.
          percentagewaste = 100 * (lboard * wboard - boardfactor * lsheet * wsheet) /
          (lboard * wboard)
          If First Or percentagewaste < tempwaste Then
              made = board1
              tempwaste = percentagewaste
              tempfact = boardfactor
              First = False
          End If
          rst2.MoveNext
     Loop
     rst2.Close
     strsql = "UPDATE Orders SET board1 = " made ", waste1 = " tempwaste " , factor1 = "
     tempfact " WHERE l = " lsheet " and w= " wsheet
     rst2.Open strsql, conndb
     rst.MoveNext
Loop
rst.Close
End Sub
```

This procedure is run to find the entries in the board preference vector for each order. It takes as a parameter the index of the board to be found, i.e. the first, second or third board.

```
Private Sub GetVector(index)


‘Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
```

```
'Get the set of sheets for which the optimal board must be found.
strsql = "SELECT * FROM Orders WHERE board"index "= 0 or board"index "is null"
rst.Open strsql, conndb

Do While Not rst.EOF
     lsheet = rst("l")
     wsheet = rst("w")
     b1 = rst("board1")
     b2 = rst("board2")
     b3 = rst("board3")
'Select all valid stock boards (from either the current or suggested stock board table)
     strsql = "SELECT * FROM Stock WHERE L ≥ " lsheet " and W ≥ " wsheet " and id <>"
     b1 " and id <> " b2 " and id <> " b3
     rst2.Open strsql, conndb
     continue = True
     First = True
     tempwaste = 0
     made = 0

'Test each valid stock board.
     Do While continue And Not rst2.EOF
          lboard = rst2("L")
          wboard = rst2("W")
          board1 = rst2("id")

'Calculate the number of sheets that can be made out of this board
          lFactor = GetFactor(lsheet, lboard)
          wfactor = GetFactor(wsheet, wboard)
          boardfactor = lFactor * wfactor

'Calculate the percentage wastage incurred when this sheet is made out of this board.
          percentagewaste = 100 * (lboard * wboard - boardfactor * lsheet * wsheet) /
          (lboard * wboard)
          If First Or percentagewaste < tempwaste Then
              made = board"index "
              tempwaste = percentagewaste
              tempfact = boardfactor
              First = False
          End If
          rst2.MoveNext
     Loop
     rst2.Close
     strsql = "UPDATE Orders SET board"index " = " made ", waste"index " = " tempwaste ",
     factor"index " = " tempfact " WHERE l = " lsheet " and w= " wsheet
     rst2.Open strsql, conndb
     rst.MoveNext
Loop
rst.Close
End Sub
```

C.4 Calculation of Transition Probabilities

This procedure calculates and stores the probabilities that govern the transition from one demand state to the next, from the historical data.

```
Private Sub TransitionProb()


‘Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")


‘Initialise the table, inserting an entry for each demand state and each board preference
vector.
cntvec = 1
Do While cntvec < NumVectors
     state = 1     Do While state < 8          strsql = "INSERT into TransitionProb values
(" cntvec "," state "," 0,0,0,0,0,0,0)"
         rst.Open strsql, conndb
         state = state + 1
     Loop
     cntvec = cntvec + 1
Loop


‘Find the probability of a demand realisation in each demand class, given the current de-
mand state, for each board preference vector.
cntvec = 1
‘Loop through all board preference vectors.
Do While cntvec < NumVectors
‘Initialise the variables
     count1 = 0
     count2 = 0
     count3 = 0
     count4 = 0
     count5 = 0
     count6 = 0
‘Loop through all Demand Classes (1 to 7).
     strsql = "SELECT * FROM States"
     rst2.open strsql, conndb
     Do While not rst2.eof
‘Loop through all historical orders for this board preference vector in this state.
         strsql = "SELECT * FROM Orders WHERE vector =" cntvec "and weekstate ="
         rst2("weekstate")
         rst.open strsql, conndb
         Do While not rst.eof
‘Determine the demand class this order falls in to, dependant on the order quantity,
and increment the relevant counter.
             if rst("q") < 76 then count1 = count1 +1
             elseif rst("q") > 75 and rst("q") < 151 then count2 = count2 +1
             elseif rst("q") > 150 and rst("q") < 226 then count3 = count3 +1
             elseif rst("q") > 225 and rst("q") <376 then count4 = count4 +1
             elseif rst("q") > 375 and rst("q") < 756 then count5 = count5 +1
```

```
            elseif rst("q") > 750 then count6 = count6 +1
            end if
            rst.movenext
        loop
‘Sum these counters to get the total number of instances of this demand state for this
board preference vector.
         countall = count1 + count2 + count3 + count4 + count5 + count6
‘Update the RealisationProb table with these probabilities.
        strsql = "UPDATE TransitionProb SET k1=" count1/countall ",k2="
        count2/countall ",k3=" count3/countall ",k4=" count4/countall ",k5="
        count5/countall ",k6=" count6/countall "WHERE vector=" cntvec "and weekstate="
rst2("weekstate")
        rst.Open strsql, conndb
       rst2.movenext
     Loop
     cntvec = cntvec + 1
loop
End Sub
```

C.5 Calculation of Board Factor Probabilities

This procedure calculates the probability of the occurrence of each potential board factor, which represents the number of sheets that can be made out of a stock board.

```
Private Sub FactorProb()
```

‘***Define the connection to the database***

```
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
```

‘***Initialise the table, inserting an entry for each board in each board preference vector.***

```
cntvec = 1
Do While cntvec < NumVectors
     cntbrd = 1
     Do While cntbrd < 4
         strsql = "INSERT into factorprob values (" cntvec "," cntbrd ",0,0,0,0,0,0)"
         rst.Open strsql, conndb
         cntbrd = cntbrd + 1
     Loop
     cntvec = cntvec + 1
Loop
```

‘***Find the probability of the occurrence of each board factor, for each board in each board preference vector.***

```
cntvec = 1
```

‘*Loop through all board preference vectors.*

```
Do While cntvec < NumVectors
```

‘*Count the number of instances of each board preference vector.*

strsql = ”SELECT count(weekid) AS weeks FROM finalacorders WHERE vectorid = ” cntvec
rst.Open strsql, conndb

```
        total = rst("weeks")
        rst.Close
        cntbrd = 1
'Loop through all boards in each board preference vector.
        Do While cntbrd < 4
'Select each possible board factor.
            strsql = "SELECT * FROM factors"
            rst.Open strsql, conndb
            Do While Not rst.EOF


'Count the number of occurrences of each board factor.
                strsql = "SELECT count(weekid) AS wks FROM finalacorders WHERE vectorid
                = " cntvec "and board =" cntbrd "and factor" cntbrd "= " rst("factor")
                rst2.Open strsql, conndb
                bfactor = rst2("wks")
                rst2.Close
'Calculate the probability of each factor, and round off to two decimal places.
                bfactor = 100 * (bfactor / total)
                bfactor = Int(bfactor)
                bfactor = bfactor / 100
                strsql = "UPDATE factorprob SET f" rst("factor") "= " bfactor " WHERE
                vector = " cntvec " and board = " cntbrd
                rst2.Open strsql, conndb
                rst.MoveNext
            Loop
            rst.Close
            cntbrd = cntbrd + 1
        Loop
        cntvec = cntvec + 1
Loop
End Sub
```

C.6 Implementation of the Inventory Model

These procedures constitute the simulation model, and the application of the model into a decision support system for implementation at *Clickabox*.

This is the code that executes the single period simulation model.

```
Private Function SinglePeriodOpt()


'Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
Set rst3 = CreateObject("ADODB.recordset")


it = 1
Do While it < 1000
```

```
    Randomize (it)
    reorder = 100
    Do While reorder < OrderMax - 100  'orderto max set by board type and rank
       orderto = reorder + 100
        Do While orderto < OrderMax
            Initialise(reorder, orderto)
'Step through the model four times to get the system into a steady state.
            count = 1
            Do While count < 5
                res = DetermineSOO(reorder, orderto)
                res = CalculateWaste()
                res = AdvanceDemand()
                count = count + 1
            loop
            res = DetermineSOO(reorder, orderto)
            strsql = "UPDATE Stock SET shortcost = 0"
            rst.open strsql, conndb
            res = CalculateWaste()
            brd = 1
            Do While brd < NumBrds
'Calculate service level attained by each board, as the percentage of board preference vectors where that board is the optimal board that have no unfilled demand.
                strsql = "SELECT count(*) AS cnt FROM Vectors WHERE UnfilledDemand
                = 0 and board1 = " brd
                rst.open strsql, conndb
                cnt = rst("cnt")
                rst.Close
                strsql = "SELECT count(*) AS cnt2 FROM Vectors WHERE UnfilledDemand
                <> 0 and board1 = " brd
                rst.open strsql, conndb
                If cnt + rst("cnt2") = 0 Then sl = 100
                Else sl = 100 * cnt / (cnt + rst("cnt2")) End If
                rst.Close
                strsql = "SELECT shortcost, Sorder1*Cost*L*W*10^-6 AS purchase,
                ([hold]*[Stock]![StockLevel]) AS holding FROM Stock WHERE id = "
                brd
                rst.open strsql, conndb
                strsql = "INSERT into Stationary values (" it "," brd ","
                rst("holding") "," rst("shortcost") "," reorder "," orderto "," sl
                "," rst("purchase") ")"
                rst3.open strsql, conndb
                rst.Close
                brd = brd + 1
            Loop
            orderto = orderto + 100
        Loop
        reorder = reorder + 100
    Loop
    it = it + 1
Loop
End Function
```

This procedure initialises the model, by generation random numbers and applying the formulae of Chapter 5 to calculate initial demand states and update the relevant tables in the database.

```
Private Function Initialise(reorder As Integer, orderto As Integer)

'Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
Set rst3 = CreateObject("ADODB.recordset")

'Initialise the random number generator
Randomize (brd)
'Set the initial demand states (see §5.6.2) from table pi.
vec = 1
Do While vec < NumVectors
     strsql = "SELECT * FROM pi WHERE vector = " vec
     rst2.open strsql, conndb
     prob = Rnd
     If prob ≤ rst2("to1") Then nextdem = 1
     ElseIf prob ≤ (rst2("to1") + rst2("to2")) Then nextdem = 2
     ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3")) Then nextdem = 3
     ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4")) Then
         nextdem = 4
     ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4") +
     rst2("to5")) Then
         nextdem = 5
     ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4") +
     rst2("to5") + rst2("to6")) Then
         nextdem = 6
     ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4") +
     rst2("to5") + rst2("to6") + rst2("to7")) Then
         nextdem = 7
     Else nextdem = 1
     End If
     strsql = "UPDATE Vectors SET demandstate = " nextdem " WHERE vectorid = " vec
     rst.open strsql, conndb
     rst2.Close
     vec = vec + 1
Loop

'Set the initial Stock levels
strsql = "UPDATE Stock SET StockLevel = orderto, Sorder1 = 0, Sorder2 = 0"
rst.open strsql, conndb
'Update a field containing unfilled demand for each board preference vector, based on the demand state
strsql = "UPDATE Vectors SET UnfilledDemand = 0"
rst.open strsql, conndb
vec = 1
Do While vec <NumVecs
     strsql = "SELECT DemandState.Median FROM DemandState INNER JOIN Vectors
     ON DemandState.DemandState = Vectors.DemandState WHERE vectorid = " vec
'The DemandState table stores the median of each demand class.
     rst3.open strsql, conndb
```

```
     If Not rst3.EOF Then
         prob = Rnd
         q = NormDist(prob,rst3("Median"))
         rst3.Close
'Update the Vectors table with the outstanding demand on that board preference vector.
         strsql = "UPDATE Vectors SET UnfilledDemand = " q " WHERE
         vectorid = " vec
         rst3.open strsql, conndb
     Else rst3.Close End If
     vec = vec + 1
Loop
End Function

Private Function DetermineSOO(reorder As Integer, orderto As Integer)

'Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")

'Update stock levels
strsql = "SELECT * FROM Stock"
rst.open strsql, conndb
Do While Not rst.EOF
     StockLevel = rst("stocklevel") + rst("sorder1") + rst("sorder2")
'Stock that has been on order for two weeks arrives and is received into stock
     strsql = "UPDATE Stock SET stocklevel = stocklevel + Sorder2 WHERE id = " rst("id")
     rst2.open strsql, conndb
'Stock on order for one week moves to "Sorder2"
     strsql = "UPDATE Stock SET Sorder2 = Sorder1 WHERE id = " rst("id")
     rst2.open strsql, conndb
'Determine the quantity to be ordered this period
     If stocklevel ≤ reorder Then
         q = orderto - rst("stocklevel") - rst("Sorder1") - rst("Sorder2")
         strsql = "UPDATE Stock SET Sorder1 = " q " WHERE id = " rst("id")
         rst2.open strsql, conndb
     Else
         strsql = "UPDATE Stock SET Sorder1 = 0 WHERE id = " rst("id")
         rst2.open strsql, conndb
     End If
     rst.movenext
Loop
rst.Close
End Function

Private Function CalculateWaste() As Double

'Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
```

```
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
Set rst3 = CreateObject("ADODB.recordset")


shortcost = 0
brd = 1
Do While brd < numbrds


‘Determine the stock level of the board being investigated.      Randomize (brd)
     strsql = "SELECT Stock.StockLevel FROM Stock WHERE id = " brd
     rst2.open strsql, conndb
     If Not rst2.EOF Then stock = rst2("stocklevel")
     Else stock = 0 End If
     rst2.Close


‘Get a list of board preference vectors, in order of descending first level wastage, for which
the optimal board is the board being investigated.
     strsql = "SELECT * FROM Vectors WHERE board1 = " brd " And ((UnfilledDemand) > 0)
     ORDER BY wcost1 DESC"
     rst.open strsql, conndb


‘While there is outstanding demand and stock available of the optimal board, allocate the
stock to the board preference vector.
     Do While Not rst.EOF And stock > 0
          maxwaste = rst("wcost1")
          demand = rst("unfilleddemand")
          vec = rst("vectorid")
          prob = Rnd
          cumprob = 0
          fac = 1
‘Generate a factor representing the number of sheets that can be made from each board from the factor
probability table.
          strsql = "SELECT * FROM FactorProb WHERE vector =" vec "and board = 1 "
          rst2.open strsql, conndb
          Do While cumprob < prob And Not rst2.EOF
               cumprob = cumprob + rst2("probability")
               fac = rst2("factor")
               rst2.movenext
          Loop
          rst2.Close
          demand = demand / fac
          If demand ≤ stock Then   'update all, met with 1st level
              strsql = "UPDATE Stock SET Stock.StockLevel = [Stock]![StockLevel] -"
               demand " WHERE id = " brd
               rst3.open strsql, conndb
               strsql = "UPDATE Vectors SET UnfilledDemand = 0 WHERE
               vectorid = " vec
               rst3.open strsql, conndb
          Else
'Fill this demand with optimal board because of the high potential shortage cost
              strsql = "UPDATE Stock SET Stock.StockLevel = 0 WHERE id = " brd
               rst3.open strsql, conndb
               strsql = "UPDATE Vectors SET UnfilledDemand = UnfilledDemand - "
```

```
            stock " WHERE vectorid = " vec
            rst3.open strsql, conndb
        End If
        strsql = "SELECT Stock.StockLevel FROM Stock WHERE id = " brd
        rst2.open strsql, conndb
        stock = rst2("stocklevel")
        rst2.Close
        rst.movenext
    Loop
    brd = brd + 1
    rst.Close
Loop
'repeat until there is no more stock of the optimal board or no more outstanding demand
brd = 1
Do While brd < numbrds
    brd2 = 1
    shortcost = 0
    Do While brd2 < numbrds
'Second level demand: select the boards with the greatest shortage costs.
        strsql = "SELECT * FROM Vectors WHERE (((Board1) = " brd ") And
        ((Board2) = " brd2 ") And((UnfilledDemand) > 0)) OrDER BY wcost2 DESC"
        rst.open strsql, conndb
        If rst.EOF Then
        Else
            vec = rst("vectorid")
            strsql = "SELECT Stock.StockLevel FROM Stock WHERE id = " brd2
            rst2.open strsql, conndb
            stock = rst2("stocklevel")
            rst2.Close
            Do While Not rst.EOF And stock > 0
                waste1 = rst("wcost2")
                demand = rst("unfilleddemand")
                vec = rst("vectorid")
                prob = Rnd
                cumprob = 0
                fac = 1
                strsql = "SELECT * FROM FactorProb WHERE vector =" vec "and board =2"
                rst2.open strsql, conndb
                Do While cumprob < prob And Not rst2.EOF
                    cumprob = cumprob + rst2("probability")
                    fac = rst2("factor")
                    rst2.movenext
                Loop
                rst2.Close
                demand = demand / fac
                If demand ≤ stock Then
'Update tables with all orders met with first level
                    strsql = "UPDATE Stock SET StockLevel = StockLevel - " demand
                    "WHERE id = " brd2
                    rst3.open strsql, conndb
                    strsql = "UPDATE Vectors SET UnfilledDemand = 0 WHERE vectorid = "
                    vec
                    rst3.open strsql, conndb
                    shortcost = shortcost + (waste1 * demand)
                    strsql = "UPDATE Stock SET shortcost = shortcost + " waste1 *
```

```
                     demand " WHERE id = " brd
                     rst3.open strsql, conndb
                 Else
'Fill this demand with optimal board because of the high potential shortage cost
                     strsql = "UPDATE Stock SET Stock.StockLevel = 0 WHERE id = " brd2
                     rst3.open strsql, conndb
                     strsql = "UPDATE Vectors SET UnfilledDemand = UnfilledDemand - "
                     stock " WHERE vectorid = " vec
                     rst3.open strsql, conndb
                     shortcost = shortcost + (waste1 * stock)
                     strsql = "UPDATE Stock SET shortcost = shortcost + " waste1 *
                     stock " WHERE id = " brd
                     rst3.open strsql, conndb
                 End If
                 strsql = "SELECT Stock.StockLevel FROM Stock WHERE id = " brd2
                 rst2.open strsql, conndb
                 stock = rst2("stocklevel")
                 rst2.Close
                 rst.movenext
             Loop
         End If
'Repeat until no more stock of the optimal board or that there is no more unfilled demand
         rst.Close
         brd2 = brd2 + 1
     Loop
     brd = brd + 1
Loop


'Investigate third level demand
strsql = "SELECT * FROM Vectors WHERE UnfilledDemand > 0"
rst.open strsql, conndb
Do While Not rst.EOF
     maxwaste = rst("wcost2")
     demand = rst("unfilleddemand")
     vec = rst("vectorid")
     prob = Rnd
     cumprob = 0
     strsql = "SELECT * FROM FactorProb WHERE vector = " vec " and board = 3 "
     rst2.open strsql, conndb
     Do While cumprob < prob And Not rst2.EOF
         cumprob = cumprob + rst2("probability")
         fac = rst2("factor")
         rst2.movenext
     Loop
     rst2.Close
     demand = demand / fac
     brd = rst("board3")
     strsql = "SELECT Stock.StockLevel FROM Stock WHERE id = " brd
     rst2.open strsql, conndb
     If rst2.EOF Then stock = 0
     Else stock = rst2("stocklevel") End If
     rst2.Close
     If demand ≤ stock Then
         strsql = "UPDATE Stock SET StockLevel = StockLevel - demand " WHERE id = " brd
         rst3.open strsql, conndb
```

```
        strsql = "UPDATE Vectors SET UnfilledDemand = 0 WHERE vectorid = " vec
        rst3.open strsql, conndb
        shortagecost = shortagecost + demand * maxwaste
        strsql = "UPDATE Stock SET shortcost = shortcost + " demand * maxwaste
        "WHERE id = " brd
        rst3.open strsql, conndb
    Else
        strsql = "UPDATE Stock SET Stock.StockLevel = 0 WHERE id = " brd
        rst3.open strsql, conndb
        strsql = "UPDATE Vectors SET UnfilledDemand = UnfilledDemand - " stock
        "WHERE vectorid = " vec
        rst3.open strsql, conndb
        shortagecost = shortagecost + stock * maxwaste
        strsql = "UPDATE Stock SET shortcost = shortcost + " stock * maxwaste " WHERE
        id = " brd
        rst3.open strsql, conndb
    End If
    rst.movenext
Loop
End Function


Private Function AdvanceDemand()


'Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
Set rst3 = CreateObject("ADODB.recordset")
strsql = "SELECT * FROM Vectors"
rst.open strsql, conndb
Do While Not rst.EOF
    vec = rst("vectorid")
    From = rst("demandstate")
    prob = Rnd
    strsql = "SELECT * FROM TransitionProb WHERE vector = " vec " and FROM = " From
    rst2.open strsql, conndb
    If prob ≤ rst2("to1") Then nextdem = 1
    ElseIf prob ≤ (rst2("to1") + rst2("to2")) Then nextdem = 2
    ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3")) Then nextdem = 3
    ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4")) Then
        nextdem = 4
    ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4") + rst2("to5"))
    Then
        nextdem = 5
    ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4") + rst2("to5")
    + rst2("to6")) Then
        nextdem = 6
    ElseIf prob ≤ (rst2("to1") + rst2("to2") + rst2("to3") + rst2("to4") + rst2("to5")
    + rst2("to6") + rst2("to7")) Then
        nextdem = 7
    Else nextdem = 1 End If
    rst2.Close
```

```
     strsql = "UPDATE Vectors SET demandstate = " nextdem " WHERE vectorid = " vec
     rst2.open strsql, conndb
     rst.movenext
Loop
rst.Close
vec = 1
Do While vec < numvectors
     strsql = "SELECT DemandState.Median FROM DemandState INNER JOIN Vectors ON
     DemandState.DemandState = Vectors.DemandState WHERE vectorid = " vec
     rst3.open strsql, conndb
     If Not rst3.EOF Then
         q = rst3("Median")
         rst3.Close
         strsql = "UPDATE Vectors SET UnfilledDemand = " q " WHERE vectorid =" vec
         rst3.open strsql, conndb
     Else rst3.Close End If
     vec = vec + 1
Loop
End Function
```

The following programs form the code that executes the multiple period simulation model.

```
Private Sub DynamicModel()
```

'*Define the connection to the database*

```
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
Set rst3 = CreateObject("ADODB.recordset")
```

'*Loop through all stock board types*

```
boardid = 1
Do While boardid < numbrds
     weekid = 1
```

'*Initialise the data in the Vectors and Stock tables.*

```
     res = Initialise(boardid)
```

'*Repeat for each week* `Do While weekid < NumWeeks`

'*Loop through all potential sets of replenishment parameters*

```
         orderto = MinOrderTo
         Do While orderto < MaxOrderTo
             reorder = MinReorder
             Do While reorder < orderto
```

'*Evaluate the total cost function*

```
                   res = CalculateCosts(reorder, orderto, weekid, boardid)
                   reorder = reorder + 100
                   Loop
                   orderto = orderto + 100
               Loop
```

'*Select the re–order and order–to levels that minimise total expected costs.*

```
         strsql = "SELECT min(shortcost) AS mincost FROM TempResults WHERE week = "
         weekid " and brd = " boardid
         rst.open strsql, conndb
```

```
          strsql = "SELECT * FROM TempResults WHERE week = " weekid " and brd = "
         boardid " and shortcost = " rst("mincost")
          rst2.open strsql, conndb
         rst.Close
         tmpreorder = rst2("reorder")
         tmporderto = rst2("orderto")
         rst2.Close
'Write the re–order and order–to levels incurring the lowest cost into the results table "Dynamic"
         strsql = "INSERT into Dynamic values (" boardid "," weekid "," tmpreorder ","
         tmporderto ")"
            rst.open strsql, conndb
         res = DetermineSOO(tmpreorder, tmporderto, boardid)
         res = CalculateWaste()
'Invoke a procedure to advance the demand state for each board preference vector according to the transition probabilities.
         res = AdvanceDemand()
         weekid = weekid + 1
     Loop
        boardid = boardid + 1
Loop
End Sub


Private Function CalculateCosts(reorder As Integer, orderto As Integer, wk As Integer,
tstboard As Integer)


'Define the connection to the database
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
Set rst3 = CreateObject("ADODB.recordset")


total = 0
loopbrd = 1
Do While loopbrd < numbrds
     k = 1
     Do While k < numbrds
         If k = loopbrd Then
         Else


'Determine the cost of the second level demand
'Get the stock level of the optimal board
             strsql = "SELECT * FROM Stock WHERE id = " k
             rst2.open strsql, conndb
             stock = rst2("stocklevel") + rst2("sorder1")
             rst2.Close
'Get a list of board preference vectors with "loopbrd" as the second best board and "k" as the optimal board
             strsql = "SELECT * FROM Vectors WHERE board1=" k "and board2=" loopbrd
             " and demandstate > 1"
             rst.open strsql, conndb
             Do While Not rst.EOF
```

```
                waste = rst("wcost1")
                demand = rst("unfilleddemand")
```
'*Calculate the total outstanding demand from all of these board preference vectors that have a first level wastage cost greater than that of the current board preference vector*
```
                strsql = "SELECT sum(unfilleddemand) AS sumdem FROM Vectors WHERE
                board1=" k "and board2=" loopbrd "and wcost1 > " waste
                rst2.open strsql, conndb
```
'*Determine a sheet per board conversion factor for the optimal board*
```
                prob = Rnd
                cumprob = 0
                fac = 1
                strsql = "SELECT * FROM Factorprob WHERE vector = " rst("vectorid")
                " and board = 1"
                rst3.open strsql, conndb
                Do While cumprob < prob And Not rst3.EOF
                    cumprob = cumprob + rst3("probability")
                    m = rst3("factor")
                    rst3.movenext
                Loop
                rst3.Close
```
'*Determine a sheet per board conversion factor for the second best board*
```
                prob = Rnd
                cumprob = 0
                fac = 1
                strsql = "SELECT * FROM Factorprob WHERE vector =" rst("vectorid")
                " and board = 2"
                rst3.open strsql, conndb
                Do While cumprob < prob And Not rst3.EOF
                    cumprob = cumprob + rst3("probability")
                    n = rst3("factor")
                    rst3.movenext
                Loop
                rst3.Close
```
'*If there is sufficient stock to fill all outstanding demand with a greater wastage cost than that of the current board preference vector then the remaining stock (A) is set to the difference between the stock level and the total demand, otherwise it is set to zero*
```
                If stock > rst2("sumdem") Then A = stock - rst2("sumdem")
                Else A = 0 End If
```
'*If the remaining stock is sufficient to fill the demand for the current board preference vector than the quantity (B) that must be filled by the second best board is zero, else it is the demand less the remaining stock*
```
                If A > demand / m Then B = 0
                Else B = demand / m - A End If
                rst2.Close
                total = total + waste * m * B / n
                rst.movenext
            Loop
            rst.Close
```

'***Calculate third level demand***
```
strsql = "SELECT * FROM Vectors WHERE board1=" k "and board3=" loopbrd
            rst.open strsql, conndb
            Do While Not rst.EOF
                waste = rst("wcost2")
                demand = rst("unfilleddemand")
```

```
prob = Rnd
cumprob = 0
fac = 1
strsql = "SELECT * FROM Factorprob WHERE vector = " rst("vectorid")
" and board = 3 "
rst3.open strsql, conndb
Do While cumprob < prob And Not rst3.EOF
    cumprob = cumprob + rst3("probability")
    n = rst3("factor")
    rst3.movenext
Loop
rst3.Close


strsql = "SELECT * FROM Vectors WHERE board1=" k "and board3=" loopbrd
"
and wcost2 > " waste
rst2.open strsql, conndb
tempsum = 0
Do While Not rst2.EOF
    strsql = "SELECT unfilleddemand FROM Vectors WHERE vector = "
   rst2("vectorid")
    rst3.open strsql, conndb
    tempdem = rst3("unfilleddemand")
    rst3.Close
    prob = Rnd
    cumprob = 0
    fac = 1
    strsql = "SELECT * FROM Factorprob WHERE vector = " rst2("vectorid")
    " and board = 1"
    rst3.open strsql, conndb
    Do While cumprob < prob And Not rst3.EOF
        cumprob = cumprob + rst3("probability")
        p = rst3("factor")
        rst3.movenext
    Loop
    rst3.Close
    tempsum = tempsum + tempdem / p
    rst2.movenext
Loop


strsql = "SELECT * FROM Stock WHERE id = " rst("board2")
rst3.open strsql, conndb
stock2 = rst3("stocklevel") + rst3("sorder1")
rst3.Close


prob = Rnd
cumprob = 0
fac = 1
strsql = "SELECT * FROM FactorProb WHERE vector = " rst("vectorid")
" and board = 1 "
rst3.open strsql, conndb
    Do While cumprob < prob And Not rst3.EOF
    cumprob = cumprob + rst3("probability")
    q = rst3("factor")
```

```
                    rst3.movenext
                Loop
                rst3.Close


                strsql = "SELECT sum(unfilleddemand) AS demand2 FROM Vectors WHERE
                board1=" rst("board2")
                rst3.open strsql, conndb
                If stock2 > rst3("demand2") Then D = stock2 - rst3("demand2")
                Else D = 0 End If
                C = demand + D
                If C > tempsum Then A = C - tempsum
                Else A = 0 End If
                If A > demand Then B = demand
                Else B = A End If
                rst3.Close
                rst2.Close
                total = total + waste * (demand - B) / n
                rst.movenext
            Loop
            rst.Close
        End If
        k = k + 1
    Loop
    loopbrd = loopbrd + 1
Loop
strsql = "SELECT * FROM Stock WHERE id = " tstboard
rst.open strsql, conndb
If Not rst.EOF Then
    StockLevel = rst("stocklevel") + rst("sorder1") + rst("sorder2")
    If StockLevel ≤ reorder Then q = orderto - StockLevel
    Else q = 0 End If
    purchase = q * Cost * rst("L") * rst("W")*10^-6
    holdcost = rst("hold") * rst("StockLevel")
End If
rst.Close
strsql = "INSERT into TempResults values (" reorder "," orderto "," total "," wk ","
tstboard "," holdcost "," purchase ")"
rst2.open strsql, conndb
End Function
```

This function is an adaptation of the 'DetermineSOO()' function in the single period optimisation model, with the only difference being the use of the variables 'orderto' and 'reorder' for the stock board currently being investigated and the values stored in the Stock table for the other boards, compared to the single period model which uses the variables for all board types.

```
Private Function DetermineSOO(reorder As Integer, orderto As Integer, brd As Integer)
```

'***Define the connection to the database***

```
strcon = "DSN=NewBoard;"
Set conndb = CreateObject("ADODB.Connection")
conndb.Open strcon
Set rst = CreateObject("ADODB.recordset")
Set rst2 = CreateObject("ADODB.recordset")
```

```
'Update stock levels
strsql = "SELECT * FROM Stock"
rst.open strsql, conndb
Do While Not rst.EOF
     StockLevel = rst("stocklevel") + rst("sorder1") + rst("sorder2")
'Stock that has been on order for two weeks arrives and is received into stock
     strsql = "UPDATE Stock SET stocklevel = stocklevel + Sorder2 WHERE id = " rst("id")
     rst2.open strsql, conndb
'Stock on order for one week moves to "Sorder2"
     strsql = "UPDATE Stock SET Sorder2 = Sorder1 WHERE id = " rst("id")
     rst2.open strsql, conndb
'Determine the quantity to be ordered this period
    If rst("id") = brd Then
         If StockLevel ≤ reorder Then
             q = orderto - rst("stocklevel") - rst("Sorder1") - rst("Sorder2")
             strsql = "UPDATE Stock SET Sorder1 = " q " WHERE id = " rst("id")
             rst2.open strsql, conndb
         Else strsql = "UPDATE Stock SET Sorder1 = 0 WHERE id = " rst("id")
             rst2.open strsql, conndb
         End If
         rst.movenext
     Else
'Use the reorder and orderto values resulting from the single period optimisation, as stored in the Stock table.
         If StockLevel ≤ rst("reorder") Then
             q = rst("orderto") - rst("stocklevel") - rst("Sorder1") - rst("Sorder2")
             strsql = "UPDATE Stock SET Sorder1 = " q " WHERE id = " rst("id")
             rst2.open strsql, conndb
         Else
             strsql = "UPDATE Stock SET Sorder1 = 0 WHERE id = " rst("id")
             rst2.open strsql, conndb
         End If
         rst.movenext
     End If
Loop
rst.Close
End Function
```

Field	Description
WeekId	Week id, from the set $\{0, \ldots, 51\}$
WeekState	Week state, from the set $\{L, I, H\}$
L	Sheet Length
W	Sheet Width
Q	Order Quantity
Board1	Id of optimal board
Waste1	Wastage when this order is produced from board 1
Factor1	The number of sheets of this order that can be produced by board 1
Board2	Id of second–best board
Waste2	Wastage when this order is produced from board 2
Factor2	The number of sheets of this order that can be produced by board 2
Board3	Id of third–best board
Waste3	Wastage when this order is produced from board 3
Factor3	The number of sheets of this order that can be produced by board 3
Vector	The id of the board preference vector for this order

(a) Orders

Field	Description
Id	Board Id
L	Board Length
W	Board Width
StockLevel	On Hand Inventory Level

(b) Stock

Field	Description
Vector	Vector Id
DemandState	Initial Demand State
k1	Probability of a transition from the initial demand state to demand state $k = 1$
k2	Probability of a transition from the initial demand state to demand state $k = 2$
k3	Probability of a transition from the initial demand state to demand state $k = 3$
k4	Probability of a transition from the initial demand state to demand state $k = 4$
k5	Probability of a transition from the initial demand state to demand state $k = 5$
k6	Probability of a transition from the initial demand state to demand state $k = 6$
k7	Probability of a transition from the initial demand state to demand state $k = 7$

(c) TransitionProb

Field	Description
Vector	Vector Id
Board1	Board Id of optimal board
Board2	Board Id of second–best board
Board3	Board Id of third–best board
DemandState	Current demand state of board preference vector
UnfilledDemand	Outstanding demand on this board preference vector
wcost1	Wastage cost when the second–best board is used instead of the optimal board
wcost2	Wastage cost when the third–best board is used instead of the second–best board

(d) Vectors

Figure C.1: *Structure of the tables referenced in the code. The* Orders *table contains the demand history, and various calculated values such as the id of the board preference vector used to make each order. The* Stock *table contains the dimensions of stock boards. The* TransitionProb *table contains the demand state transition probabilities and the* Vectors *table contains information pertaining to the board preference vectors.*

Appendix D

Snapshot of Demand Data

Date	Style			Box Size			Qty	Board	Board	Board	Board
	Type	Class	Flute	L	W	D		Used 1	Used 2	Length	Width
06/03/01	RSC	A	C	145	145	205	2 500	2 000 × 1 500		628	365
06/03/01	RSC	A	C	145	145	210	2 500	2 000 × 1 500		628	370
07/03/01	RSC	A	C	415	285	260	1 000	1 530 × 1 300		1 448	560
07/03/01	RSC	A	C	415	285	260	1 000	1 458 × 560		1 448	560
07/03/01	RSC	A	C	510	348	225	25	1 860 × 1 200		1 764	589
12/03/01	FOLF	A	C	995	240	750	250	1 400 × 1 300	1 530 × 1300	1 275	1 240
12/03/01	FOLF	A	C	995	240	750	250	1 285 × 1 240		1 275	1 240
12/03/01	FOLF	A	C	995	240	750	500	1 285 × 1 240		1 275	1 240
12/03/01	FOLF	A	C	995	240	750	1 000	1 285 × 1 240		1 275	1 240
19/03/01	RSC	A	C	1108	460	380	250	1 795 × 0910	1 700 × 1 000	1 608	856
19/03/01	RSC	A	C	1108	460	380	500	1 795 × 0910	1 700 × 1 000	1 608	856
20/03/01	RSC	A	C	564	485	325	250	2 358 × 1 062		2 146	825
20/03/01	RSC	A	C	1108	460	380	250	1 795 × 0910	1 700 × 1 000	1 608	856
20/03/01	RSC	A	C	1108	460	380	250	1 618 × 856		1 608	856
20/03/01	Sleeve	DWB	–	1400	650	850	4	2 380 × 1 200		2 103	850
20/03/01	Lid	DWB	–	1420	670	150	8	1 840 × 1 200		1 758	1 000
20/03/01	RSC	A	C	620	504	454	250	2 358 × 1 062		2 296	974
20/03/01	RSC	A	C	620	252	454	500	1 850 × 1 510		1 792	722
20/03/01	RSC	A	C	454	252	620	500	1 540 × 910	1 795 × 910	1 460	888
20/03/01	RSC	A	C	460	360	210	5000	1 700 × 1 200	1 860 × 1 200	1 688	586
22/03/01	RSC	A	C	620	252	454	500	1 850 × 1 510		1 792	722

Table D.1*: Snapshot of the works tickets stored by Clickabox, containing all information necessary for the production of an order, such as the dimensions, cardboard type, and box design required.*

Appendix E

Board Preference Vectors

Index	Board 1	Board 2	Board 3
1	AC 1 030 × 2 370	AC 2 030 × 1 240	AC 2 200 × 1 200
2	AC 1 030 × 2 370	AC 2 110 × 1 680	AC 1 910 × 1 880
3	AC 1 030 × 2 370	AC 2 200 × 1 200	AC 2 300 × 1 220
4	AC 1 030 × 2 370	AC 2 260 × 1 520	AC 2 110 × 1 680
5	AC 1 030 × 2 370	AC 1 550 × 1 020	AC 2 110 × 1 680
6	AC 1 260 × 2 300	AC 1 360 × 2 300	AC 1 330 × 2 370
7	AC 1 260 × 2 300	AC 1 360 × 2 300	AC 2 110 × 1 680
8	AC 1 260 × 2 300	AC 2 300 × 1 710	AC 1 360 × 2 300
9	AC 1 260 × 2 300	AC 2 370 × 1 250	AC 1 360 × 2 300
10	AC 1 280 × 1 300	AC 1 380 × 1 310	AC 1 260 × 2 300
11	AC 1 280 × 1 300	AC 1 380 × 1 310	AC 1 530 × 1 380
12	AC 1 280 × 1 300	AC 1 380 × 1 310	AC 1 720 × 1 210
13	AC 1 280 × 1 300	AC 2 260 × 1 520	AC 1 380 × 1 310
14	AC 1 330 × 2 370	AC 1 280 × 1 300	AC 1 460 × 2 370
15	AC 1 330 × 2 370	AC 1 460 × 2 370	AC 1 380 × 1 310
16	AC 1 330 × 2 370	AC 1 460 × 2 370	AC 1 510 × 1 810
17	AC 1 360 × 2 300	AC 1 330 × 2 370	AC 1 280 × 1 300
18	AC 1 360 × 2 300	AC 1 330 × 2 370	AC 1 460 × 2 370
19	AC 1 360 × 2 300	AC 1 330 × 2 370	AC 1 470 × 1 480
20	AC 1 360 × 2 300	AC 1 330 × 2 370	AC 1 530 × 1 380
21	AC 1 360 × 2 300	AC 1 330 × 2 370	AC 1 550 × 1 020
22	AC 1 360 × 2 300	AC 1 460 × 2 370	AC 1 380 × 1 310
23	AC 1 360 × 2 300	AC 1 530 × 1 380	AC 1 470 × 1 480
24	AC 1 360 × 2 300	AC 1 550 × 1 020	AC 1 460 × 2 370
25	AC 1 380 × 1 310	AC 1 360 × 2 300	AC 1 330 × 2 370
26	AC 1 380 × 1 310	AC 1 530 × 1 380	AC 1 470 × 1 480
27	AC 1 380 × 1 310	AC 1 720 × 1 210	AC 1 530 × 1 380
28	AC 1 380 × 1 310	AC 2 370 × 1 250	AC 1 530 × 1 380
29	AC 1 460 × 2 370	AC 1 380 × 1 310	AC 1 720 × 1 210
30	AC 1 460 × 2 370	AC 1 500 × 1 540	AC 1 510 × 1 810
31	AC 1 460 × 2 370	AC 1 510 × 1 810	AC 1 720 × 1 210
32	AC 1 460 × 2 370	AC 1 530 × 1 380	AC 1 470 × 1 480
33	AC 1 460 × 2 370	AC 1 680 × 1 080	AC 1 510 × 1 810
34	AC 1 460 × 2 370	AC 1 680 × 1 080	AC 1 720 × 1 210
35	AC 1 460 × 2 370	AC 1 720 × 1 210	AC 1 530 × 1 380
36	AC 1 460 × 2 370	AC 2 300 × 1 710	AC 2 200 × 1 200
37	AC 1 460 × 2 370	AC 1 550 × 1 020	AC 1 380 × 1 310
38	AC 1 460 × 2 370	AC 1 550 × 1 020	AC 1 680 × 1 080
39	AC 1 460 × 2 370	AC 1 550 × 1 020	AC 1 910 × 1 880
40	AC 1 470 × 1 480	AC 1 460 × 2 370	AC 1 500 × 1 540

Table E.1: *A list of the preference vector index and composition of all AC board preference vectors.*

Index	Board 1	Board 2	Board 3
41	AC 1 470 × 1 480	AC 1 500 × 1 540	AC 1 510 × 1 810
42	AC 1 470 × 1 480	AC 1 860 × 1 490	AC 1 500 × 1 540
43	AC 1 470 × 1 480	AC 2 200 × 1 200	AC 2 260 × 1 520
44	AC 1 500 × 1 540	AC 1 510 × 1 810	AC 1 260 × 2 300
45	AC 1 500 × 1 540	AC 1 510 × 1 810	AC 1 360 × 2 300
46	AC 1 500 × 1 540	AC 1 510 × 1 810	AC 1 860 × 1 490
47	AC 1 500 × 1 540	AC 1 510 × 1 810	AC 1 550 × 1 020
48	AC 1 510 × 1 810	AC 1 260 × 2 300	AC 1 360 × 2 300
49	AC 1 510 × 1 810	AC 1 260 × 2 300	AC 2 370 × 1 250
50	AC 1 510 × 1 810	AC 1 360 × 2 300	AC 1 330 × 2 370
51	AC 1 510 × 1 810	AC 1 360 × 2 300	AC 1 460 × 2 370
52	AC 1 510 × 1 810	AC 1 360 × 2 300	AC 1 550 × 1 020
53	AC 1 510 × 1 810	AC 1 460 × 2 370	AC 1 530 × 1 380
54	AC 1 510 × 1 810	AC 1 460 × 2 370	AC 1 910 × 1 880
55	AC 1 510 × 1 810	AC 1 460 × 2 370	AC 2 110 × 1 680
56	AC 1 510 × 1 810	AC 1 530 × 1 380	AC 1 500 × 1 540
57	AC 1 510 × 1 810	AC 1 720 × 1 210	AC 1 530 × 1 380
58	AC 1 510 × 1 810	AC 1 860 × 1 490	AC 1 550 × 1 020
59	AC 1 510 × 1 810	AC 2 300 × 1 220	AC 1 260 × 2 300
60	AC 1 510 × 1 810	AC 1 550 × 1 020	AC 1 460 × 2 370
61	AC 1 510 × 1 810	AC 1 550 × 1 020	AC 1 910 × 1 880
62	AC 1 510 × 1 810	AC 1 550 × 1 020	AC 2 110 × 1 680
63	AC 1 530 × 1 380	AC 1 030 × 2 370	AC 1 550 × 1 020
64	AC 1 530 × 1 380	AC 1 470 × 1 480	AC 1 460 × 2 370
65	AC 1 530 × 1 380	AC 1 470 × 1 480	AC 1 500 × 1 540
66	AC 1 530 × 1 380	AC 1 470 × 1 480	AC 2 260 × 1 520
67	AC 1 530 × 1 380	AC 1 500 × 1 540	AC 1 510 × 1 810
68	AC 1 530 × 1 380	AC 1 510 × 1 810	AC 1 860 × 1 490
69	AC 1 530 × 1 380	AC 1 860 × 1 490	AC 2 000 × 1 400
70	AC 1 530 × 1 380	AC 2 030 × 1 240	AC 1 860 × 1 490
71	AC 1 530 × 1 380	AC 2 260 × 2 160	AC 1 470 × 1 480
72	AC 1 530 × 1 380	AC 1 550 × 1 020	AC 1 470 × 1 480
73	AC 1 530 × 1 380	AC 1 550 × 1 020	AC 1 510 × 1 810
74	AC 1 530 × 1 380	AC 1 550 × 1 020	AC 2 030 × 1 240
75	AC 1 680 × 1 080	AC 1 510 × 1 810	AC 1 720 × 1 210
76	AC 1 680 × 1 080	AC 1 720 × 1 210	AC 1 530 × 1 380
77	AC 1 680 × 1 080	AC 1 720 × 1 210	AC 1 800 × 1 200
78	AC 1 680 × 1 080	AC 1 720 × 1 210	AC 2 110 × 1 010
79	AC 1 680 × 1 080	AC 1 860 × 1 000	AC 1 720 × 1 210
80	AC 1 680 × 1 080	AC 1 860 × 1 000	AC 2 030 × 1 240
81	AC 1 680 × 1 080	AC 1 860 × 1 490	AC 1 720 × 1 210
82	AC 1 680 × 1 080	AC 2 260 × 2 160	AC 1 030 × 2 370
83	AC 1 680 × 1 080	AC 2 260 × 2 160	AC 2 200 × 1 200
84	AC 1 680 × 1 080	AC 2 300 × 1 220	AC 2 370 × 1 250
85	AC 1 680 × 1 080	AC 1 550 × 1 020	AC 1 720 × 1 210
86	AC 1 680 × 1 080	AC 1 550 × 1 020	AC 1 860 × 1 000
87	AC 1 720 × 1 210	AC 1 530 × 1 380	AC 1 470 × 1 480
88	AC 1 720 × 1 210	AC 1 530 × 1 380	AC 1 800 × 1 200
89	AC 1 720 × 1 210	AC 1 800 × 1 200	AC 1 030 × 2 370
90	AC 1 720 × 1 210	AC 1 800 × 1 200	AC 1 680 × 1 080
91	AC 1 720 × 1 210	AC 1 800 × 1 200	AC 1 910 × 1 880
92	AC 1 720 × 1 210	AC 1 800 × 1 200	AC 2 030 × 1 240
93	AC 1 720 × 1 210	AC 1 800 × 1 200	AC 2 110 × 1 680
94	AC 1 720 × 1 210	AC 1 800 × 1 200	AC 2 260 × 2 160
95	AC 1 720 × 1 210	AC 2 030 × 1 240	AC 1 860 × 1 490
96	AC 1 720 × 1 210	AC 2 110 × 1 010	AC 1 800 × 1 200
97	AC 1 800 × 1 200	AC 1 260 × 2 300	AC 2 370 × 1 250
98	AC 1 800 × 1 200	AC 1 910 × 1 880	AC 2 030 × 1 240
99	AC 1 800 × 1 200	AC 2 030 × 1 240	AC 2 200 × 1 200
100	AC 1 800 × 1 200	AC 2 110 × 1 680	AC 1 910 × 1 880
101	AC 1 800 × 1 200	AC 2 260 × 2 160	AC 1 030 × 2 370
102	AC 1 800 × 1 200	AC 2 260 × 2 160	AC 2 030 × 1 240
103	AC 1 800 × 1 200	AC 2 370 × 1 250	AC 1 330 × 2 370
104	AC 1 800 × 1 200	AC 2 370 × 1 250	AC 1 910 × 1 880
105	AC 1 860 × 1 000	AC 1 720 × 1 210	AC 2 110 × 1 010
106	AC 1 860 × 1 000	AC 1 720 × 1 210	AC 2 300 × 1 220
107	AC 1 860 × 1 000	AC 1 860 × 1 490	AC 2 000 × 1 400
108	AC 1 860 × 1 000	AC 2 110 × 1 010	AC 1 800 × 1 200
109	AC 1 860 × 1 000	AC 2 110 × 1 010	AC 2 260 × 2 160
110	AC 1 860 × 1 490	AC 1 030 × 2 370	AC 2 260 × 1 520

Table E.1 (cntd.): *AC Board Preference Vectors*

Index	Board 1	Board 2	Board 3
111	AC 1 860 × 1 490	AC 1 470 × 1 480	AC 2 260 × 1 520
112	AC 1 860 × 1 490	AC 1 720 × 1 210	AC 2 000 × 1 400
113	AC 1 860 × 1 490	AC 1 860 × 1 000	AC 2 110 × 1 010
114	AC 1 860 × 1 490	AC 1 910 × 1 880	AC 1 800 × 1 200
115	AC 1 860 × 1 490	AC 2 000 × 1 400	AC 1 280 × 1 300
116	AC 1 860 × 1 490	AC 2 000 × 1 400	AC 1 530 × 1 380
117	AC 1 860 × 1 490	AC 2 000 × 1 400	AC 2 260 × 1 520
118	AC 1 860 × 1 490	AC 2 000 × 1 400	AC 2 260 × 2 160
119	AC 1 860 × 1 490	AC 2 000 × 1 400	AC 2 370 × 1 250
120	AC 1 860 × 1 490	AC 2 000 × 1 400	AC 1 550 × 1 020
121	AC 1 860 × 1 490	AC 2 260 × 1 520	AC 2 110 × 1 680
122	AC 1 860 × 1 490	AC 2 260 × 2 160	AC 1 030 × 2 370
123	AC 1 860 × 1 490	AC 1 550 × 1 020	AC 2 260 × 1 520
124	AC 1 910 × 1 880	AC 1 030 × 2 370	AC 2 030 × 1 240
125	AC 1 910 × 1 880	AC 1 680 × 1 080	AC 1 860 × 1 000
126	AC 1 910 × 1 880	AC 1 860 × 1 000	AC 1 510 × 1 810
127	AC 1 910 × 1 880	AC 1 860 × 1 000	AC 1 720 × 1 210
128	AC 1 910 × 1 880	AC 1 860 × 1 000	AC 2 110 × 1 010
129	AC 1 910 × 1 880	AC 1 860 × 1 000	AC 2 300 × 1 710
130	AC 1 910 × 1 880	AC 1 860 × 1 490	AC 1 860 × 1 000
131	AC 1 910 × 1 880	AC 1 860 × 1 490	AC 2 000 × 1 400
132	AC 1 910 × 1 880	AC 2 030 × 1 240	AC 1 860 × 1 490
133	AC 1 910 × 1 880	AC 2 110 × 1 010	AC 2 260 × 2 160
134	AC 1 910 × 1 880	AC 2 260 × 2 160	AC 1 030 × 2 370
135	AC 1 910 × 1 880	AC 2 300 × 1 710	AC 2 260 × 2 160
136	AC 1 910 × 1 880	AC 1 550 × 1 020	AC 1 680 × 1 080
137	AC 2 000 × 1 400	AC 1 280 × 1 300	AC 2 260 × 1 520
138	AC 2 000 × 1 400	AC 2 260 × 1 520	AC 2 110 × 1 680
139	AC 2 000 × 1 400	AC 2 260 × 2 160	AC 2 260 × 1 520
140	AC 2 030 × 1 240	AC 1 720 × 1 210	AC 1 330 × 2 370
141	AC 2 030 × 1 240	AC 1 860 × 1 490	AC 2 000 × 1 400
142	AC 2 030 × 1 240	AC 2 000 × 1 400	AC 2 300 × 1 220
143	AC 2 030 × 1 240	AC 2 000 × 1 400	AC 2 370 × 1 250
144	AC 2 030 × 1 240	AC 2 200 × 1 200	AC 1 860 × 1 490
145	AC 2 030 × 1 240	AC 2 200 × 1 200	AC 2 000 × 1 400
146	AC 2 030 × 1 240	AC 2 200 × 1 200	AC 2 260 × 1 520
147	AC 2 030 × 1 240	AC 2 300 × 1 220	AC 2 370 × 1 250
148	AC 2 110 × 1 010	AC 2 030 × 1 240	AC 2 200 × 1 200
149	AC 2 110 × 1 010	AC 2 200 × 1 200	AC 2 300 × 1 220
150	AC 2 110 × 1 010	AC 2 260 × 1 520	AC 2 110 × 1 680
151	AC 2 110 × 1 010	AC 2 260 × 2 160	AC 1 030 × 2 370
152	AC 2 110 × 1 010	AC 2 260 × 2 160	AC 2 030 × 1 240
153	AC 2 110 × 1 010	AC 2 260 × 2 160	AC 2 200 × 1 200
154	AC 2 110 × 1 680	AC 1 030 × 2 370	AC 2 030 × 1 240
155	AC 2 110 × 1 680	AC 1 680 × 1 080	AC 2 260 × 2 160
156	AC 2 110 × 1 680	AC 1 910 × 1 880	AC 1 680 × 1 080
157	AC 2 110 × 1 680	AC 1 910 × 1 880	AC 1 860 × 1 000
158	AC 2 110 × 1 680	AC 1 910 × 1 880	AC 1 860 × 1 490
159	AC 2 110 × 1 680	AC 1 910 × 1 880	AC 2 260 × 2 160
160	AC 2 110 × 1 680	AC 1 910 × 1 880	AC 2 300 × 1 710
161	AC 2 110 × 1 680	AC 2 260 × 2 160	AC 1 030 × 2 370
162	AC 2 110 × 1 680	AC 2 260 × 2 160	AC 2 030 × 1 240
163	AC 2 110 × 1 680	AC 2 300 × 1 710	AC 1 500 × 1 540
164	AC 2 110 × 1 680	AC 2 300 × 1 710	AC 2 110 × 1 010
165	AC 2 110 × 1 680	AC 2 300 × 1 710	AC 2 260 × 2 160
166	AC 2 110 × 1 680	AC 1 550 × 1 020	AC 1 910 × 1 880
167	AC 2 200 × 1 200	AC 1 680 × 1 080	AC 2 300 × 1 220
168	AC 2 200 × 1 200	AC 2 110 × 1 010	AC 1 380 × 1 310
169	AC 2 200 × 1 200	AC 2 110 × 1 010	AC 2 300 × 1 220
170	AC 2 200 × 1 200	AC 2 110 × 1 680	AC 1 720 × 1 210
171	AC 2 200 × 1 200	AC 2 260 × 1 520	AC 1 720 × 1 210
172	AC 2 200 × 1 200	AC 2 300 × 1 220	AC 1 260 × 2 300
173	AC 2 200 × 1 200	AC 2 300 × 1 220	AC 2 370 × 1 250
174	AC 2 260 × 1 520	AC 1 260 × 2 300	AC 2 300 × 1 710
175	AC 2 260 × 1 520	AC 1 380 × 1 310	AC 2 300 × 1 710
176	AC 2 260 × 1 520	AC 1 500 × 1 540	AC 1 550 × 1 020
177	AC 2 260 × 1 520	AC 1 720 × 1 210	AC 2 300 × 1 220
178	AC 2 260 × 1 520	AC 1 860 × 1 490	AC 2 300 × 1 710
179	AC 2 260 × 1 520	AC 2 110 × 1 680	AC 1 260 × 2 300
180	AC 2 260 × 1 520	AC 2 110 × 1 680	AC 1 910 × 1 880

Table E.1 (cntd.): *AC Board Preference Vectors*

Index	Board 1	Board 2	Board 3
181	AC 2 260 × 1 520	AC 2 110 × 1 680	AC 2 300 × 1 710
182	AC 2 260 × 1 520	AC 2 200 × 1 200	AC 2 110 × 1 680
183	AC 2 260 × 1 520	AC 2 200 × 1 200	AC 2 300 × 1 220
184	AC 2 260 × 1 520	AC 2 300 × 1 710	AC 1 470 × 1 480
185	AC 2 260 × 1 520	AC 2 300 × 1 710	AC 1 530 × 1 380
186	AC 2 260 × 1 520	AC 2 300 × 1 710	AC 2 260 × 2 160
187	AC 2 260 × 1 520	AC 1 550 × 1 020	AC 2 110 × 1 680
188	AC 2 260 × 2 160	AC 1 030 × 2 370	AC 1 260 × 2 300
189	AC 2 260 × 2 160	AC 1 030 × 2 370	AC 1 280 × 1 300
190	AC 2 260 × 2 160	AC 1 030 × 2 370	AC 1 510 × 1 810
191	AC 2 260 × 2 160	AC 1 030 × 2 370	AC 2 030 × 1 240
192	AC 2 260 × 2 160	AC 1 030 × 2 370	AC 2 200 × 1 200
193	AC 2 260 × 2 160	AC 1 030 × 2 370	AC 2 260 × 1 520
194	AC 2 260 × 2 160	AC 1 260 × 2 300	AC 1 360 × 2 300
195	AC 2 260 × 2 160	AC 1 460 × 2 370	AC 2 300 × 1 710
196	AC 2 260 × 2 160	AC 1 470 × 1 480	AC 2 260 × 1 520
197	AC 2 260 × 2 160	AC 1 510 × 1 810	AC 1 260 × 2 300
198	AC 2 260 × 2 160	AC 2 030 × 1 240	AC 2 200 × 1 200
199	AC 2 260 × 2 160	AC 2 200 × 1 200	AC 1 510 × 1 810
200	AC 2 260 × 2 160	AC 2 200 × 1 200	AC 2 300 × 1 220
201	AC 2 260 × 2 160	AC 2 260 × 1 520	AC 1 260 × 2 300
202	AC 2 260 × 2 160	AC 2 260 × 1 520	AC 2 110 × 1 680
203	AC 2 260 × 2 160	AC 2 260 × 1 520	AC 2 300 × 1 710
204	AC 2 260 × 2 160	AC 2 300 × 1 220	AC 1 260 × 2 300
205	AC 2 260 × 2 160	AC 2 300 × 1 220	AC 2 370 × 1 250
206	AC 2 300 × 1 220	AC 1 260 × 2 300	AC 2 370 × 1 250
207	AC 2 300 × 1 220	AC 2 370 × 1 250	AC 1 280 × 1 300
208	AC 2 300 × 1 220	AC 2 370 × 1 250	AC 1 330 × 2 370
209	AC 2 300 × 1 220	AC 2 370 × 1 250	AC 1 910 × 1 880
210	AC 2 300 × 1 220	AC 2 370 × 1 250	AC 2 260 × 1 520
211	AC 2 300 × 1 220	AC 2 370 × 1 250	AC 2 260 × 2 160
212	AC 2 300 × 1 220	AC 2 370 × 1 250	AC 2 300 × 1 710
213	AC 2 300 × 1 710	AC 1 330 × 2 370	AC 1 460 × 2 370
214	AC 2 300 × 1 710	AC 1 470 × 1 480	AC 1 500 × 1 540
215	AC 2 300 × 1 710	AC 1 500 × 1 540	AC 1 510 × 1 810
216	AC 2 300 × 1 710	AC 1 500 × 1 540	AC 2 260 × 2 160
217	AC 2 300 × 1 710	AC 1 510 × 1 810	AC 1 260 × 2 300
218	AC 2 300 × 1 710	AC 1 510 × 1 810	AC 2 260 × 2 160
219	AC 2 300 × 1 710	AC 1 510 × 1 810	AC 2 300 × 1 220
220	AC 2 300 × 1 710	AC 1 530 × 1 380	AC 1 470 × 1 480
221	AC 2 300 × 1 710	AC 2 110 × 1 010	AC 2 260 × 2 160
222	AC 2 300 × 1 710	AC 2 110 × 1 680	AC 1 910 × 1 880
223	AC 2 300 × 1 710	AC 2 260 × 2 160	AC 1 510 × 1 810
224	AC 2 300 × 1 710	AC 2 260 × 2 160	AC 2 200 × 1 200
225	AC 2 300 × 1 710	AC 2 260 × 2 160	AC 2 300 × 1 220
226	AC 2 300 × 1 710	AC 2 300 × 1 220	AC 2 370 × 1 250
227	AC 2 370 × 1 250	AC 1 280 × 1 300	AC 1 380 × 1 310
228	AC 2 370 × 1 250	AC 1 280 × 1 300	AC 2 260 × 1 520
229	AC 2 370 × 1 250	AC 1 330 × 2 370	AC 1 280 × 1 300
230	AC 2 370 × 1 250	AC 1 720 × 1 210	AC 1 800 × 1 200
231	AC 2 370 × 1 250	AC 1 910 × 1 880	AC 2 030 × 1 240
232	AC 2 370 × 1 250	AC 2 260 × 1 520	AC 2 300 × 1 710

Table E.1 (cntd.): *AC Board Preference Vectors*

Index	Board 1	Board 2	Board 3
1	DWB 1 480 × 1 310	DWB 1 410 × 1 940	DWB 1 530 × 1 370
2	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 1 780 × 1 620
3	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 1 270 × 1 700
4	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 1 410 × 1 940
5	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 1 670 × 1 010
6	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 2 030 × 1 080
7	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 1 870 × 1 350
8	DWB 1 480 × 1 310	DWB 1 530 × 1 370	DWB 2 300 × 2 180
9	DWB 1 480 × 1 310	DWB 2 010 × 1 460	DWB 1 530 × 1 370
10	DWB 1 780 × 1 620	DWB 1 820 × 2 090	DWB 1 870 × 1 350
11	DWB 1 780 × 1 620	DWB 1 670 × 1 010	DWB 2 150 × 1 640
12	DWB 1 780 × 1 620	DWB 2 010 × 1 460	DWB 1 670 × 1 010
13	DWB 1 780 × 1 620	DWB 2 010 × 1 460	DWB 2 150 × 1 640
14	DWB 1 780 × 1 620	DWB 2 010 × 1 460	DWB 2 270 × 1 430
15	DWB 1 780 × 1 620	DWB 2 010 × 1 460	DWB 2 300 × 2 180
16	DWB 1 780 × 1 620	DWB 2 150 × 1 640	DWB 1 820 × 2 090
17	DWB 1 820 × 2 090	DWB 1 480 × 1 310	DWB 1 530 × 1 370
18	DWB 1 820 × 2 090	DWB 1 780 × 1 620	DWB 2 010 × 1 460
19	DWB 1 820 × 2 090	DWB 1 050 × 1 980	DWB 1 270 × 1 700
20	DWB 1 820 × 2 090	DWB 1 050 × 1 980	DWB 2 030 × 1 080
21	DWB 1 820 × 2 090	DWB 1 050 × 1 980	DWB 2 330 × 2 000
22	DWB 1 820 × 2 090	DWB 1 530 × 1 370	DWB 2 030 × 1 080
23	DWB 1 820 × 2 090	DWB 2 030 × 1 080	DWB 2 330 × 2 000
24	DWB 1 820 × 2 090	DWB 2 030 × 1 080	DWB 2 300 × 2 180
25	DWB 1 820 × 2 090	DWB 2 330 × 2 000	DWB 2 300 × 2 180
26	DWB 1 820 × 2 090	DWB 1 870 × 1 350	DWB 2 150 × 1 640
27	DWB 1 820 × 2 090	DWB 2 410 × 1 690	DWB 2 330 × 2 000
28	DWB 1 820 × 2 090	DWB 2 300 × 2 180	DWB 2 300 × 2 180
29	DWB 1 050 × 1 980	DWB 2 010 × 1 460	DWB 1 170 × 1 310
30	DWB 1 050 × 1 980	DWB 2 030 × 1 080	DWB 2 330 × 2 000
31	DWB 1 050 × 1 980	DWB 2 330 × 2 000	DWB 2 300 × 2 180
32	DWB 1 230 × 1 420	DWB 1 480 × 1 310	DWB 1 530 × 1 370
33	DWB 1 230 × 1 420	DWB 1 480 × 1 310	DWB 2 410 × 1 690
34	DWB 1 230 × 1 420	DWB 1 270 × 1 700	DWB 1 410 × 1 940
35	DWB 1 230 × 1 420	DWB 1 410 × 1 940	DWB 1 480 × 1 310
36	DWB 1 230 × 1 420	DWB 1 530 × 1 370	DWB 1 270 × 1 700
37	DWB 1 230 × 1 420	DWB 2 010 × 1 460	DWB 1 410 × 1 940
38	DWB 1 230 × 1 420	DWB 2 010 × 1 460	DWB 1 870 × 1 350
39	DWB 1 230 × 1 420	DWB 2 010 × 1 460	DWB 2 410 × 1 690
40	DWB 1 230 × 1 420	DWB 2 410 × 1 690	DWB 1 270 × 1 700
41	DWB 1 230 × 1 420	DWB 2 410 × 1 690	DWB 1 410 × 1 940
42	DWB 1 230 × 1 420	DWB 2 410 × 1 690	DWB 1 530 × 1 370
43	DWB 1 270 × 1 700	DWB 1 820 × 2 090	DWB 1 410 × 1 940
44	DWB 1 270 × 1 700	DWB 1 410 × 1 940	DWB 1 780 × 1 620
45	DWB 1 270 × 1 700	DWB 1 410 × 1 940	DWB 1 670 × 1 010
46	DWB 1 170 × 1 310	DWB 1 230 × 1 420	DWB 1 480 × 1 310
47	DWB 1 170 × 1 310	DWB 1 230 × 1 420	DWB 1 820 × 2 090
48	DWB 1 170 × 1 310	DWB 2 330 × 2 000	DWB 2 270 × 1 430
49	DWB 1 170 × 1 310	DWB 2 330 × 2 000	DWB 2 300 × 2 180
50	DWB 1 170 × 1 310	DWB 2 270 × 1 430	DWB 1 230 × 1 420
51	DWB 1 410 × 1 940	DWB 1 480 × 1 310	DWB 1 530 × 1 370
52	DWB 1 410 × 1 940	DWB 1 780 × 1 620	DWB 1 480 × 1 310
53	DWB 1 410 × 1 940	DWB 1 780 × 1 620	DWB 1 670 × 1 010
54	DWB 1 410 × 1 940	DWB 1 780 × 1 620	DWB 2 010 × 1 460
55	DWB 1 410 × 1 940	DWB 1 780 × 1 620	DWB 2 150 × 1 640
56	DWB 1 410 × 1 940	DWB 1 820 × 2 090	DWB 2 330 × 2 000
57	DWB 1 410 × 1 940	DWB 1 670 × 1 010	DWB 1 820 × 2 090
58	DWB 1 410 × 1 940	DWB 1 670 × 1 010	DWB 1 230 × 1 420
59	DWB 1 410 × 1 940	DWB 1 670 × 1 010	DWB 2 150 × 1 640
60	DWB 1 530 × 1 370	DWB 1 780 × 1 620	DWB 2 010 × 1 460
61	DWB 1 530 × 1 370	DWB 1 820 × 2 090	DWB 1 780 × 1 620
62	DWB 1 530 × 1 370	DWB 1 270 × 1 700	DWB 1 410 × 1 940
63	DWB 1 530 × 1 370	DWB 1 270 × 1 700	DWB 1 870 × 1 350
64	DWB 1 530 × 1 370	DWB 1 410 × 1 940	DWB 1 780 × 1 620
65	DWB 1 530 × 1 370	DWB 2 030 × 1 080	DWB 2 300 × 2 180
66	DWB 1 530 × 1 370	DWB 1 870 × 1 350	DWB 1 780 × 1 620
67	DWB 1 530 × 1 370	DWB 1 870 × 1 350	DWB 1 820 × 2 090
68	DWB 1 530 × 1 370	DWB 1 870 × 1 350	DWB 1 410 × 1 940
69	DWB 1 530 × 1 370	DWB 2 300 × 2 180	DWB 1 870 × 1 350
70	DWB 1 670 × 1 010	DWB 1 820 × 2 090	DWB 1 480 × 1 310

Table E.2: *A list of the preference vector index and composition of all DWB board preference vectors.*

Index	Board 1	Board 2	Board 3
71	DWB 1 670 × 1 010	DWB 1 820 × 2 090	DWB 1 780 × 1 620
72	DWB 1 670 × 1 010	DWB 1 820 × 2 090	DWB 1 050 × 1 980
73	DWB 1 670 × 1 010	DWB 1 820 × 2 090	DWB 1 530 × 1 370
74	DWB 1 670 × 1 010	DWB 1 820 × 2 090	DWB 2 030 × 1 080
75	DWB 1 670 × 1 010	DWB 1 820 × 2 090	DWB 2 410 × 1 690
76	DWB 1 670 × 1 010	DWB 1 230 × 1 420	DWB 1 820 × 2 090
77	DWB 1 670 × 1 010	DWB 1 410 × 1 940	DWB 1 820 × 2 090
78	DWB 1 670 × 1 010	DWB 1 530 × 1 370	DWB 2 030 × 1 080
79	DWB 1 670 × 1 010	DWB 2 150 × 1 640	DWB 1 820 × 2 090
80	DWB 2 010 × 1 460	DWB 1 170 × 1 310	DWB 2 270 × 1 430
81	DWB 2 010 × 1 460	DWB 2 150 × 1 640	DWB 1 820 × 2 090
82	DWB 2 010 × 1 460	DWB 2 150 × 1 640	DWB 2 410 × 1 690
83	DWB 2 010 × 1 460	DWB 1 870 × 1 350	DWB 2 150 × 1 640
84	DWB 2 010 × 1 460	DWB 2 270 × 1 430	DWB 1 230 × 1 420
85	DWB 2 010 × 1 460	DWB 2 270 × 1 430	DWB 2 150 × 1 640
86	DWB 2 010 × 1 460	DWB 2 270 × 1 430	DWB 2 300 × 2 180
87	DWB 2 010 × 1 460	DWB 2 300 × 2 180	DWB 1 170 × 1 310
88	DWB 2 010 × 1 460	DWB 2 300 × 2 180	DWB 2 150 × 1 640
89	DWB 2 010 × 1 460	DWB 2 300 × 2 180	DWB 2 410 × 1 690
90	DWB 2 150 × 1 640	DWB 1 820 × 2 090	DWB 2 410 × 1 690
91	DWB 2 150 × 1 640	DWB 2 030 × 1 080	DWB 2 010 × 1 460
92	DWB 2 150 × 1 640	DWB 2 330 × 2 000	DWB 2 010 × 1 460
93	DWB 2 150 × 1 640	DWB 2 410 × 1 690	DWB 1 050 × 1 980
94	DWB 2 150 × 1 640	DWB 2 410 × 1 690	DWB 1 410 × 1 940
95	DWB 2 150 × 1 640	DWB 2 410 × 1 690	DWB 2 030 × 1 080
96	DWB 2 150 × 1 640	DWB 2 410 × 1 690	DWB 2 330 × 2 000
97	DWB 2 150 × 1 640	DWB 2 300 × 2 180	DWB 2 410 × 1 690
98	DWB 2 030 × 1 080	DWB 2 150 × 1 640	DWB 2 300 × 2 180
99	DWB 2 030 × 1 080	DWB 2 330 × 2 000	DWB 2 150 × 1 640
100	DWB 2 030 × 1 080	DWB 2 330 × 2 000	DWB 2 300 × 2 180
101	DWB 2 030 × 1 080	DWB 1 870 × 1 350	DWB 2 010 × 1 460
102	DWB 2 030 × 1 080	DWB 2 300 × 2 180	DWB 2 010 × 1 460
103	DWB 2 030 × 1 080	DWB 2 300 × 2 180	DWB 1 870 × 1 350
104	DWB 2 030 × 1 080	DWB 2 300 × 2 180	DWB 2 270 × 1 430
105	DWB 2 330 × 2 000	DWB 1 170 × 1 310	DWB 1 670 × 1 010
106	DWB 2 330 × 2 000	DWB 2 270 × 1 430	DWB 2 300 × 2 180
107	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 2 300 × 2 180
108	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 1 820 × 2 090
109	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 1 170 × 1 310
110	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 1 410 × 1 940
111	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 1 670 × 1 010
112	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 2 270 × 1 430
113	DWB 2 330 × 2 000	DWB 2 300 × 2 180	DWB 2 410 × 1 690
114	DWB 1 870 × 1 350	DWB 1 780 × 1 620	DWB 2 010 × 1 460
115	DWB 1 870 × 1 350	DWB 1 820 × 2 090	DWB 1 780 × 1 620
116	DWB 1 870 × 1 350	DWB 1 820 × 2 090	DWB 1 050 × 1 980
117	DWB 1 870 × 1 350	DWB 1 820 × 2 090	DWB 2 010 × 1 460
118	DWB 1 870 × 1 350	DWB 2 010 × 1 460	DWB 1 170 × 1 310
119	DWB 1 870 × 1 350	DWB 2 010 × 1 460	DWB 2 270 × 1 430
120	DWB 1 870 × 1 350	DWB 2 150 × 1 640	DWB 2 030 × 1 080
121	DWB 1 870 × 1 350	DWB 2 410 × 1 690	DWB 1 780 × 1 620
122	DWB 2 270 × 1 430	DWB 1 230 × 1 420	DWB 2 150 × 1 640
123	DWB 2 270 × 1 430	DWB 1 230 × 1 420	DWB 2 410 × 1 690
124	DWB 2 270 × 1 430	DWB 2 150 × 1 640	DWB 2 410 × 1 690
125	DWB 2 270 × 1 430	DWB 2 330 × 2 000	DWB 2 150 × 1 640
126	DWB 2 270 × 1 430	DWB 2 410 × 1 690	DWB 1 270 × 1 700
127	DWB 2 270 × 1 430	DWB 2 410 × 1 690	DWB 2 330 × 2 000
128	DWB 2 270 × 1 430	DWB 2 300 × 2 180	DWB 1 230 × 1 420
129	DWB 2 410 × 1 690	DWB 2 410 × 1 690	DWB 2 410 × 1 690
130	DWB 2 410 × 1 690	DWB 1 780 × 1 620	DWB 1 670 × 1 010
131	DWB 2 410 × 1 690	DWB 1 780 × 1 620	DWB 2 010 × 1 460
132	DWB 2 410 × 1 690	DWB 1 780 × 1 620	DWB 2 150 × 1 640
133	DWB 2 410 × 1 690	DWB 1 780 × 1 620	DWB 2 330 × 2 000
134	DWB 2 410 × 1 690	DWB 1 050 × 1 980	DWB 1 270 × 1 700
135	DWB 2 410 × 1 690	DWB 1 270 × 1 700	DWB 1 410 × 1 940
136	DWB 2 410 × 1 690	DWB 1 270 × 1 700	DWB 1 670 × 1 010
137	DWB 2 410 × 1 690	DWB 1 270 × 1 700	DWB 2 330 × 2 000
138	DWB 2 410 × 1 690	DWB 2 030 × 1 080	DWB 2 330 × 2 000
139	DWB 2 410 × 1 690	DWB 2 330 × 2 000	DWB 2 330 × 2 000
140	DWB 2 410 × 1 690	DWB 2 330 × 2 000	DWB 2 270 × 1 430

Table E.2 (cntd.): *DWB Board Preference Vectors*

Index	Board 1	Board 2	Board 3
141	DWB 2 410 × 1 690	DWB 2 330 × 2 000	DWB 2 300 × 2 180
142	DWB 2 300 × 2 180	DWB 2 300 × 2 180	DWB 2 300 × 2 180
143	DWB 2 300 × 2 180	DWB 1 820 × 2 090	DWB 1 820 × 2 090
144	DWB 2 300 × 2 180	DWB 1 230 × 1 420	DWB 2 410 × 1 690
145	DWB 2 300 × 2 180	DWB 1 170 × 1 310	DWB 1 230 × 1 420
146	DWB 2 300 × 2 180	DWB 1 170 × 1 310	DWB 2 270 × 1 430
147	DWB 2 300 × 2 180	DWB 2 150 × 1 640	DWB 2 410 × 1 690
148	DWB 2 300 × 2 180	DWB 1 870 × 1 350	DWB 1 780 × 1 620
149	DWB 2 300 × 2 180	DWB 2 270 × 1 430	DWB 2 150 × 1 640
150	DWB 2 300 × 2 180	DWB 2 270 × 1 430	DWB 2 410 × 1 690
151	DWB 2 300 × 2 180	DWB 2 410 × 1 690	DWB 1 270 × 1 700
152	DWB 2 300 × 2 180	DWB 2 410 × 1 690	DWB 2 330 × 2 000
153	DWB 2 300 × 2 180	DWB 2 410 × 1 690	DWB 2 270 × 1 430

Table E.2 (cntd.): *DWB Board Preference Vectors*

Appendix F

Multiple Period Optimisation Results

Board Type	Week 1	Week 2	Week 3	Week 4
AC 1 030 × 2 370	(500, 1 300)	(100, 800)	(200, 800)	(600, 1 200)
AC 1 260 × 2 300	(300, 400)	(500, 1 000)	(400, 800)	(400, 900)
AC 1 280 × 1 300	(200, 600)	(100, 200)	(500, 900)	(400, 1 000)
AC 1 330 × 2 370	(600, 900)	(600, 1 000)	(700, 900)	(800, 1 100)
AC 1 360 × 2 300	(500, 1 000)	(100, 400)	(100, 500)	(300, 800)
AC 1 380 × 1 310	(900, 1 100)	(100, 900)	(1 400, 1 900)	(1 000, 1 500)
AC 1 460 × 2 370	(500, 900)	(700, 1 200)	(600, 1 400)	(700, 1 000)
AC 1 470 × 1 480	(1 300, 1 500)	(1 100, 1 900)	(1 000, 1 600)	(1 100, 1 600)
AC 1 500 × 1 540	(500, 700)	(1 000, 1 100)	(800, 900)	(200, 600)
AC 1 510 × 1 810	(1 000, 1 100)	(1 300, 1 500)	(600, 1 900)	(600, 1 500)
AC 1 530 × 1 380	(100, 1 100)	(500, 1 500)	(300, 700)	(300, 1 900)
AC 1 550 × 1 020	(400, 1 300)	(1 500, 1 600)	(400, 800)	(600, 1 200)
AC 1 680 × 1 080	(900, 1 000)	(1 100, 1 600)	(1 200, 1 500)	(1 100, 1 800)
AC 1 720 × 1 210	(500, 1 300)	(1 500, 1 600)	(500, 1 100)	(400, 1 100)
AC 1 800 × 1 200	(1 800, 1 900)	(1 600, 1 700)	(400, 1 000)	(1 100, 1 700)
AC 1 860 × 1 000	(1 000, 1 400)	(500, 1 100)	(400, 1 300)	(900, 1 100)
AC 1 860 × 1 490	(700, 900)	(200, 1 100)	(1 400, 1 900)	(500, 700)
AC 1 910 × 1 880	(100, 300)	(300, 1 800)	(700, 1 500)	(1 400, 1 900)
AC 2 000 × 1 400	(200, 600)	(300, 500)	(100, 1 800)	(100, 300)
AC 2 030 × 1 240	(1 100, 1 400)	(100, 1 400)	(500, 600)	(300,900)
AC 2 110 × 1 010	(1 500, 1 600)	(600, 1 500)	(700, 1 100)	(100, 200)
AC 2 110 × 1 680	(100, 900)	(300, 1 300)	(500, 1 300)	(300, 800)
AC 2 200 × 1 200	(100, 600)	(300, 1 200)	(500, 800)	(600, 900)
AC 2 260 × 1 520	(1 000, 1 300)	(1 100, 1 200)	(100, 200)	(300, 500)
AC 2 260 × 2 160	(1 200, 1 400)	(100, 1 400)	(600, 1 600)	(1 500, 1 800)
AC 2 300 × 1 220	(1 200, 1 800)	(700, 900)	(400, 1 000)	(200, 500)
AC 2 300 × 1 710	(300, 1 000)	(800, 1 200)	(700, 1 000)	(500, 1 000)
AC 2 370 × 1 250	(200, 700)	(100, 700)	(200, 800)	(200, 900)

Table F.1: *Results of a Dynamic Optimisation Simulation for a four week period, for the 28 AC cardboard types. The replenishment paramters for each week are given in the format* $(\overline{s}, \overline{S})$*, where* $\overline{s}$ *represents the re–order level and* $\overline{S}$ *the order–to level.*

Board Type	Week 1	Week 2	Week 3	Week 4
DWB $1\,480 \times 1\,530$	(200, 600)	(200, 600)	(500, 700)	(200, 700)
DWB $1\,780 \times 2\,150$	(300, 500)	(100, 400)	(600, 800)	(700, 900)
DWB $1\,820 \times 2\,300$	(100, 600)	(200, 300)	(500, 900)	(700, 800)
DWB $1\,050 \times 1\,820$	(500, 800)	(100, 600)	(300, 500)	(500, 700)
DWB $1\,230 \times 1\,270$	(300, 800)	(200, 500)	(600, 700)	(200, 700)
DWB $1\,270 \times 1\,410$	(300, 400)	(100, 700)	(800, 900)	(100, 700)
DWB $1\,170 \times 1\,230$	(400, 900)	(300, 700)	(100, 300)	(100, 600)
DWB $1\,410 \times 1\,820$	(100, 900)	(100, 500)	(500, 700)	(800, 900)
DWB $1\,530 \times 1\,780$	(100, 500)	(500, 700)	(700, 800)	(400, 800)
DWB $1\,670 \times 2\,030$	(300, 800)	(400, 900)	(100, 600)	(700, 800)
DWB $2\,010 \times 2\,150$	(400, 700)	(500, 800)	(500, 600)	(500, 600)
DWB $2\,150 \times 2\,410$	(200, 300)	(500, 700)	(200, 900)	(400, 700)
DWB $2\,030 \times 2\,270$	(400, 600)	(100, 500)	(300, 800)	(100, 300)
DWB $2\,330 \times 1\,820$	(400, 700)	(700, 800)	(100, 400)	(500, 800)
DWB $1\,870 \times 2\,010$	(400, 700)	(300, 400)	(200, 700)	(100, 800)
DWB $2\,270 \times 2\,410$	(100, 400)	(100, 700)	(500, 900)	(400, 800)
DWB $2\,410 \times 1\,270$	(400, 500)	(400, 500)	(200, 900)	(200, 400)
DWB $2\,300 \times 2\,410$	(700, 800)	(400, 900)	(100, 800)	(300, 900)

Table F.2: *Results of a Dynamic Optimisation Simulation for a four week period, for the 18 DWB cardboard types. The replenishment paramters for each week are given in the format* $(\overline{s}, \overline{S})$, *where* $\overline{s}$ *represents the re–order level and* $\overline{S}$ *the order–to level.*

Appendix G

Instructions for using Compact Disc

This appendix contains an description of the contents of the compact disc attached to this thesis, and brief instructions for the use of this disc. The disc contains two files, both in Microsoft Excel worksheet format. The first file, named "Data from Clickabox Workstickets" comprises one sheet listing all the worksticket data as stored on a data server by *Clickabox*. The second file, named "Tables", comprises four sheets, namely *TransitionProbAC* (containing the data from the transition probability table for AC board types), *TransitionProbDWB* (containing the data from the transition probability table for DWB board types), *FactorProbAC* (containing the data from the sheet–to–board conversion factor table for AC board types) and *FactorProbDWB* (containing the data from the sheet–to–board conversion factor table for DWB board types).

References

[1] N. Agrawal & S.A. Smith, *Estimating Negative Binomial Demand for Retail Inventory Management with Unobservable Lost Sales*, Naval Research Logistics, 43(6), 839–861, 1996.

[2] J.F. Cox & J.H. Blackstone Jr., *APICS Dictionary*, APICS Educational and Research Foundation, Inc., 10th Edition, Alexandra, VA, p. 58, 2002.

[3] K.J. Arrow, T.E. Harris & J. Marschak, *Optimal Inventory Policy*, Econometrica, 19(3), 250–272, 1951.

[4] K.J. Arrow, S. Karlin & H. E. Scarf, *Studies in the Mathematical Theory of Inventory and Production*, Stanford University Press, Stanford, CA, 1958.

[5] Atlantic Packaging, *Atlantic Packaging, Cape Town, South Africa*, [Online], [cited 2003, December 11], Available from: `http://zapages.com/atlantic--packaging`

[6] K.S. Azoury & B.L. Miller, *Comparison of Optimal Ordering Levels of Bayesian and Non–Bayesian Inventory Models*, Management Science, 30(8), 993–1003, 1984.

[7] K.S. Azoury, *Bayes Solution to Dynamic Inventory Models Under Unknown Demand Distributions*, Management Science, 31(9), 1150–1160, 1985.

[8] Y. Bassok, R. Anupindi & R. Akella, *Single–Period Multiproduct Inventory Models with Substitution*, Operations Research, 47(4), 632–642, 1999.

[9] L.E. Baum, T. Petrie, G. Soules & N. Weiss, *A Maximization Technique Occuring in the Statistical Analysis of Probabilistic Functions of Markov Chains*, The Annals of Mathematical Statistics, 41(1), 164–171, 1970.

[10] J. Bekker, *Simulation of Discrete–Event Stochastic Processes*, Department of Industrial Engineering, University of Stellenbosch, 2002.

[11] A. Bensousson, M. Crouchy & J. Proth, *Mathematical Theory of Production Planning*, North–Holland, Amsterdam, 1983.

[12] J.H. Bookbinder & M. Çakanyildirim, *Random Lead Times and Expedited Orders in (Q, r) Inventory Systems*, European Journal of Operational Research, 115(2), 300–313, 1999.

[13] Boxes for Africa, *Boxes for Africa, Erasmia, South Africa*, [Online], [cited 2004, July 14], Available from: `http://cbn.co.za/buslist/package/Box300.htm`

[14] Cape Town Box Manufacturers, *Cape Town Box Manufacturers, Cape Town, South Africa*, ph.: 021–5058505

[15] F. Cheng & S.P. Sethi, *Optimality of State–dependent $(\overline{s}, \overline{S})$ Policies in Inventory Models with Markov–modulated Demand and Lost Sales*, Production and Operations Management, 8(2), 183–192, 1999.

[16] B.J. Coleman, *Determining the Correct Service Level Target*, Production and Inventory Management Journal, 1, 19–23, 2000.

[17] Continental China, *Continental China, Blackheath, Western Cape, South Africa*, [Online], [cited 2004, July 14], Available from: `http://www.continental.za.com`

[18] S.A. Conrad, *Sales Data and the Estimation of Demand*, Operations Research Quarterly, 27(1), 123–127, 1976.

[19] M.J. Davis, N.J. Aquilano & R.B. Chase, *Fundamentals of Operations Management*, Irwin/McGraw–Hill, New York, NY, Third Ed., pp. 461–464, 1999.

[20] X. Ding & M.L. Puterman, *The Censored Newsvendor and the Optimal Aquisition of Information*, Technical Report, Faculty of Commerce and Business Administration, University of British Columbia, 1998.

[21] B. Downs, R. Metters & J. Semple, *Managing Inventory with Multiple Products, Lags in Delivery, Resource Constraints and Lost Sales: A Mathematical Programming Approach*, Management Science, 47(3), 464–479, 2001.

[22] Z. Drezner & H. Gurnani, *Deterministic Heirarchical Substitution Inventory Models.*, Journal of the Operational Research Society, 51, 1, 129–133, 2000.

[23] Z. Drezner, H. Gurnani & B.A. Pasternack, *An EOQ Model with Substitution Between Products*, Journal of the Operational Research Society, 46, 5, 887–891, 1995.

[24] D. Ford (Foreman at *Clickabox*), *Personal communications*, 27/06/01.

[25] P. du T Fourie, I. Nieuwoudt & J.H. van Vuuren, *Methods of OR: A New Graduate Course*, Submitted to ORiON, 2003.

[26] G. Gallego, J.K. Ryan & D. Simchi–Levi, *Minimax Analysis for Finite Horizon Inventory Models*, Working paper, Columbia University, New York, NY, 1996.

[27] S.C. Graves, *A Single–Item Inventory Model for a Non–Stationary Demand Process*, Manufacturing and Service Operations Management, 1(1), 50–61, 1999.

[28] K.W. Green Jr. & R.A. Inman, *Production and Operations Management Philosophies: Evolution and Synthesis*, Revised as of 22 August 2000, [Online], [cited 2004, July 14], Available from: `http://www.hsu.edu/faculty/greenk/pomphil.htm`

[29] G. Hadley & T. Whitin, *Analysis of Inventory Systems*, Prentice Hall, Englewood Cliffs, NJ, 1963.

[30] F.W. Harris, *How Many Parts to Make at Once*, Factory, The Magazine of Management, 10(2), 135–152, 1913.

[31] D.L. Iglehart, *Dynamic Programming and Stationary Analysis of Inventory Problems*, in: H. Scarf, D. Gilford, M. Shelley, eds., Multistage Inventory Models and Techniques, Stanford University Press, Stanford, CA, 1–31, 1963.

[32] D.L. Iglehart, *The Dynamic Inventory Problem with Unknown Demand Distribution*, Management Science, 10(3), 429–440, 1964.

[33] E. Ignall & A.F. Veinott, Jr., *Optimality of Myopic Invenotry policies for Several Substitute Products*, Management Science, 15(5), 284–304, 1969.

[34] L. Johnson & D. Montgomery, *Operations Research in Production, Scheduling, and Inventory Control*, John Wiley & Sons, Inc, New York, NY, 1974.

[35] W. Johnson, *Distributed National Inventory: Introduction to the project*, [Online], [cited 2003, May 6], Available from: `http://www.ptnintra.mbsa.co.za`

[36] S.S. Kambhamettu, *Estimation for Partially Observable Markov Decision Processes*, MSc. Thesis, Auburn University, Auburn, AL, 2000.

[37] F.K. Karaesmen, G. Liberopoulos & Y. Dallery, *The Value of Advance Demand Information in Production/Inventory systems*, Revised as of September 2002, [Online], [cited 2003, January 17], Available from: `http://www.lgi.ecp.fr/users/fikri/pdfs/advinfo2.pdf`

[38] S. Karlin, *Dynamic inventory policy with varying stochastic demands*, Management Science, 6(3), 231–258, 1960.

[39] S. Karlin & H.E. Scarf, *Inventory Models of the Arrow–Harris–Marschak Type with Time Lag*, in: K.J. Arrow, S. Karlin, H.E. Scarf, eds., Studies in the Mathematical Theory of Inventory and Production, Stanford University Press, Stanford, CA, 1958.

[40] D.H. Kim & K.S. Park, *(Q, r) Inventory Model with a Mixture of Lost Sales and Time–weighted Backorders*, Journal of the Operational Research Society, 36(3), 231–238, 1985.

[41] Kohler Packaging, *Kohler Packaging, 23 Hagart Rd., Pinetown, South Africa*, [Online], [cited 2003, December 11], Available from: `http://www.kohler.co.za/CartonPrint/`

[42] A.A. Kurawarwala & H. Matsuo, *Forecasting and Inventory Management of Short Life–cycle Products*, Operations Research, 44(1), 131–150, 1996.

[43] M.A. Lariviere & E.L. Porteus, *Informational Dynamics and New Product Pricing*, Working paper, Standford University, Stanford, CA., 1995.

[44] H.L. Lee & S. Nahmias, *Single Product, Single–Location Models*, in: S.C. Graves, Rinnooy A.H.G. Kan, P.H. Zipkin, eds., Handbooks in Operations Research and Management Science: Logistics of Production and Inventory, Elsevier Science, Amsterdam, Volume 4, pp. 1–55, 1993.

[45] B.G. Leroux & M.L. Puterman, *Maximum–penalized–likelihood Estimation for Independent and Markov–dependent Mixture Models*, Biometrics, 48, 545–558, 1992.

[46] E. Lombaard (Sunday Times, 27 Oct 2002), *Hunger stalks land of plenty*, [Online], [cited 2004, July 14], Available from: `http://www.sundaytimes.co.za/2002/10/27/news/news10.asp`

[47] W.S. Lovejoy, *Myopic Policies for Some Inventory Models with Uncertain Demand Distributions*, Management Science, 36(6), 724–738, 1990.

[48] W.S. Lovejoy, *Stopped Myopic Policies in Some Inventory Models with Generalized Demand Processes*, Management Science, 38(5), 688–707, 1992.

[49] Microsoft Corporation, *Microsoft Access 2000*, [Online], [cited 2004, July 14], Available from: `http://msdn.microsoft.com/office/previous/access`

[50] Microsoft Corporation, *Microsoft Visual Basic 1987*, [Online], [cited 2004, July 14], Available from: `http://msdn.microsoft.com/vbasic/`

[51] B. Miller, *Scarf's State Reduction Method, Flexibility and a Dependent Demand Inventory Model*, Operations Research, 34(1), 83–90, 1986.

[52] Mondipak Corrugated, *Mondipak Corrugated, Losack Ave., Epping Industrial, South Africa*, [Online], [cited 2004, July 14], Available from: `http://www.mondi.co.za`

[53] D.C. Montgomery, M.S. Bazaraa & A.K. Keswani, *Inventory models with a mixture of backorders and lost sales*, Naval Research Logistics Quarterly, 20, 255–263, 1973.

[54] D.C. Montgomery, & G.C. Runger, *Engineering Statistics*, John Wiley & Sons, Inc, New York, NY, 66, 1998.

[55] S. Nahmias, *Production and Operations Analysis*, Irwin/McGraw–Hill, Burr Ridge, IL, Third Edition, pp. 213–215, 1989.

[56] Nampak Corrugated, *Nampak Corrugated, 27 Boland St., Daljosophat, Paarl, South Africa*, [Online], [cited 2004, July 14], Available from: `http://www.nampak.co.za`

[57] Netcommerce, *Clickabox Website*, [Online], [cited 2004, July 14], Available from: `http://www.clickabox.co.za`

[58] Pastel, *Pastel Accounting Package*, [Online], [cited 2004, July 14], Available from: `http://www.pastel.com`

[59] G.W. Plossl & O.W. Wight, *Production and Inventory Control: Principles and Techniques*, Prentice–Hall, Englewood Cliffs, NJ, 1967.

[60] Rockwell Software Inc, *Arena Arena Input Analyser*, Version 5, [Online], [cited 2004, July 14], Available from: `http://www.arenasimulation.com/`

[61] S. Ross, *A First Course in Probability*, Prentice–Hall, Englewood Cliffs, NJ, 328–331 & 134–139, 2002.

[62] SA Financial Sector Forum, *SA Reserve Bank Website*, [Online], [cited 2003, March 16], Available from: `http://www.resbank.co.za/sarbdata/rates/rates.asp?type=cmr`

[63] H.E. Scarf, *Bayes Solutions of the Statistical Inventory Problem*, Annals of Mathematical Science, 30, 490–508, 1959.

[64] H.E. Scarf, *The Optimality of (s, S) policies in the Dynamic Inventory Problem*, K. Arrow, S. Karlin and P. Suppes (eds.), Mathematical Methods in the Social Sciences, Stanford University Press, Stanford, CA, 1960.

[65] S.P. Sethi & F. Cheng, *Optimality of (s, S) Policies in Inventory Models with Markovian Demand*, Operations Research, 45(6), 931–939, 1997.

[66] J. Song & P. Zipkin, *Inventory Control in a Fluctuating Demand Environment*, Operations Research, 41(2), 351–370, 1993.

[67] J. Song & P. Zipkin, *Managing Inventory with the Prospect of Obsolescence*, Operations Research, 44(1), 215–222, 1996.

[68] C.R. Sox, S.S. Kambhamettu & J. T. Treharne, *Modeling Adaptive Inventory Control Problems*, Working paper, Auburn University, Auburn, AL, 2002.

[69] L. Swanepoel (Owner of *Netcommerce*), E–mail address: *sleon@mweb.co.za*.

[70] P. Taljaard (Director of *Clickabox*), *Personal communications*, 08/05/01, 01/06/01, 13/06/01, 21/06/01, 27/06/01, 11/07/01, 26/07/01, 14/09/01, 28/02/02, 06/06/02, 21/08/02, 15/10/02, 13/12/02, 17/02/03.

[71] P. Taljaard (Director of *Clickabox*), *Correspondence*, 26/07/01, 25/04/02, 05/12/02, 05/02/03, 27/02/03, 16/04/03, 25/04/03, 20/05/03, 21/10/03, 30/10/03.

[72] P. Taljaard, *Clickabox Company Profile*, Unpublished document, p. 2, 2003.

[73] J.T. Treharne & C.R. Sox, *Adaptive Inventory Control for Stationary Demand and Partial Information*, Management Science, 48(5), 607–624, May 2002.

[74] J.T. Treharne & C.R. Sox, *Models and Methods for Adaptive Inventory Control*, Working paper, Auburn University, Auburn, AL, 1999.

[75] N. van Heerden (Accountant for *Clickabox*), *Personal communication, Financials for year ending 28 February 2003*, 05/05/03.

[76] A.F. Veinott, *Optimal Stockage Policies with Non–stationary Stochastic Demands*, in: H. Scarf, D. Gilford, M. Shelly, eds., Multistage Inventory Models and Techniques, Office of Naval Research Monograph Series on Mathematical Methods in Logistics, Stanford University Press, Stanford, CA, Volume 4, pp. 85–115, 1963.

[77] A.F. Veinott, *Optimal Policy for a Multi–Product, Dynamic, Non–stationary Inventory Problem*, Management Science, 12(3), 206–222, 1965.

[78] R.H. Wilson, *A Scientific Routine for Stock Control*, Harvard Business Review, 13, 116–128, 1934.

[79] W.L. Winston, *Operations Research, Applications and Algorithms*, Duxbury Press, Bellmont, CA, Third Edition, p. 870, 1994.

[80] W.L. Winston, *Operations Research, Applications and Algorithms*, Duxbury Press, Bellmont, CA, Third Edition, pp. 869–880, 1994.

[81] W.L. Winston, *Operations Research, Applications and Algorithms*, Duxbury Press, Bellmont, CA, Third Edition, pp. 934–935, 1994.

[82] A. Wirth, *Inventory Control and Inflation: A Review*, International Journal of Operations & Production Management, 9(1), 67–72, 1989.

[83] N.L. Zhang & W. Zhang, *Speeding Up the Convergence of Value Iteration in Partially Observed Markovian Decision Processes*, Journal of Artificial Intelligence Research, 14, 29–51, 2001.