Master's Thesis

Evaluation of emergency ordering policies at Syncron

Niklas Jonsson Åberg and Staffan Theander

2016-01-17

Institution: Production Management, Faculty of Engineering, Lund University
Supervisor at Lund University: Johan Marklund
Supervisor at Syncron: Mikael Blom
Examiner at Lund University: Peter Berling

Preface

This master's thesis was conducted at Production Management, Lund University, in cooperation with Syncron in Stockholm during the Autumn 2015. It constitutes the final part of our five-year engineering studies at Lund university.

Many people have helped us throughout this project. First of all, we want to thank Syncron for facilitating this study. We have had a great time in the Stockholm Office and got to know many friendly and interesting people. Through a multitude of conversations with employees in different positions and countries around the world we have learned a lot about the organization, the industry and about inventory management. Especially we want to thank Mikael Blom for introducing us to the problem and for his supervising. We have had many inspiring and valuable discussions.

We also want to thank our supervisors at Lund University, Peter Berling and Johan Marklund, for many helpful comments throughout the study. The feedback has been very constructive and we did always leave our meetings with new insights and a lot of optimism.

Lund, January 2016

Niklas Jonsson Åberg | Staffan Theander

Abstract

The aim of our research has been to identify what prevents Syncron from implementing a single item emergency ordering policy in their inventory management system, GIM. Emergency ordering is defined here as the ability to use two supply sources, a normal one and a faster more expensive emergency source. We further wanted to investigate under what circumstances an emergency ordering policy could be used and what the benefits of such a policy would be in comparison to Syncron´s current solutions.

To answer these questions, we first analyzed the company´s single item and multi-item inventory control mechanisms. Second, we searched the academic literature for emergency ordering polices that would be compatible with the existing system, and that would have the potential to reduce the costs while retaining service levels. Third, the chosen models were tested against Syncron´s current single and dual supplier ordering policies through a simulation study.

The primary obstacles preventing Syncron from implementing an emergency ordering policy is their multi-item optimization algorithm. The target of this algorithm is to obtain a certain overall service level while minimizing the total stock value. The problems of this, in the context of introducing an emergency ordering policy from academic literature, are primary twofold. First, making a total cost optimization, where the extra cost for the emergency replenishment option is weighted against the reduction in inventory, is not supported. Second, most emergency ordering policies in literature utilize a backorder cost instead of a service level requirement and the translation between these concepts is often unsatisfactory.

Given these limitations there were few feasible models to evaluate. A compromise was made to select two cost optimizing policies that could cut costs significantly if Syncron would make larger changes, and two heuristics which were tailor-made to fit the the current situation, but which lacked the cost optimizing feature.

The chosen policies were

- The model in Song and Zipkin (2009) for items with Poisson demand,
- an adaption of Roslings (2002) Lost Sales model for fast and erratic items that is referred to in Axsäter (2006),
- a short horizon emergency heuristic with a Cost/Service level ratio limit for items with fast and erratic demand and,
- a long horizon emergency heuristic with a Cost/Service level ratio limit for items with fast and erratic demand.

The most important result of our simulation study is that a simple single supplier (R, Q) policy outperformed all the non-optimizing policies, including Syncron´s rush ordering heuristic, under the real conditions tested. Our simulation study further showed that both the model by Song and Zipkin (2009) and the lost sales model performed better than both Syncron´s current solutions, given that all requirement for these policies were met.

Our tailor-made solutions performed only slightly better than Syncron´s rush ordering heuristics under the real conditions. Our sensitivity analysis

showed that, under conditions that are attractive for emergency ordering in general, the short horizon emergency heuristic performed better than both Syncron´s policies.

Our recommendation to Syncron is to use more caution before enabling the existing rush ordering heuristic for a customer. In most cases this heuristic will do more harm than good. We further encourage Syncron to investigate whether the tailor-made polices can have more apparent benefits in a multi-item setting, where a Cost/Service level ratio limit can be set for a group of items. This would enable the filtering out of items not suited for emergency ordering.

Sammanfattning

Målet med vår studie var att identifiera vad som hindrar företaget Syncron från att implementera en litteraturbaserad nödorderpolicy i deras lagerstyrningssystem, GIM. En nödorderpolicy definieras här som en beslutsregel som väger ett normalt leveransalternativ mot ett snabbare och dyrare leveransalternativ. Vidare ville vi undersöka under vilka omständigheter en nödorderpolicy kan och bör användas och vilka fördelar en sådan beslutsregel skulle ha jämtemot företagets nuvarande lösningar.

För att svara på dessa frågor började vi med att analysera och formulera företagets lagerstyrningsmekanismer matematiskt för en och flera artiklar. Sedan kartlades de nödordermodeller som finns i den akademiska litteraturen för att hitta modeller som var kompatibla med Syncrons system och som hade potentialen att sänka kostnader med bibehållen servicenivå. Avslutningsvis genomförde vi en simuleringsstudie där valda nödordermodeller testades mot Syncrons nuvarande lösningar.

Det som hindrar Syncron från att införa en litteraturbaserad nödorderpolicy är deras optimeringsalgoritm för flera artiklar. Målet med denna algoritm är att minimera bundet kapital i lagret under bivillkoret att en given total servicenivå uppnås. Detta skapar framförallt två problem. För det första negligeras kostnadsskillnaden mellan de två leveransalternativen och för det andra så bygger de flesta nödorderpolicyerna i litteraturen på en bristkostnad istället för på ett servicenivåbivillkor och översättningen mellan dessa koncept är ofta ej tillfredställande.

På grund av dessa hinder fanns det få möjliga modeller att välja mellan. Vi fick därför göra en kompromiss. Två modeller valdes som krävde stora förändringar hos Syncron, men som samtidigt gjorde fullständiga kostnadsoptimeringar. Utöver dessa byggde vi två skräddarsydda heuristiker som passade Syncrons nuvarande situation.

De valda modellerna var:

- Modellen i Song och Zipkin (2009) för artiklar med Poisson efterfrågan,
- en anpassning av Roslings (2002) ”Lost Sales” modell som refereras i Axsäter (2006), för artiklar med snabb och ojämn efterfrågan,
- en heuristik med kort tidshorisont och med en kostnad/service-kvotsbegränsning för artiklar med snabb och ojämn efterfrågan, samt

- en heuristik med lång tidshorisont och med en kostnad/service-kvotsbegränsning för artiklar med snabb och ojämn efterfrågan.

Det mest intressanta resultatet från vår simuleringsstudie var att en enkel (R, Q)-policy med ett leveransalternativ presterade bättre än alla icke kostnadsoptimerande beslutsregler, inklusive Syncrons nuvarande, under de verkliga förhållanden som testades.

Vidare visade studien att både modellen av Song och Zipkin och den anpassade "Lost Sales" modellen presterade bättre än Syncrons modeller för en och flera leveransalternativ.

Våra skräddarsydda modeller presterade bara marginellt bättre än Syncrons nuvarande nödorderpolicy under de verkliga förhållandena. En känslighetsanalys visade dock att heuristiken med kort tidshorisont presterade bättre än Syncrons nuvarande modell under förhållanden som över lag är attraktiva för en nödorderpolicy.

Vår rekommendation till Syncron är att göra en analys innan de låter en kund använda den nuvarande nödorderpolicyn. I de flesta fallen kommer heuristiken göra med skada än nytta. Vidare rekommenderar vi Syncron att undersöka om våra skräddarsydda beslutsregler kan göra mer nytta i ett flerartikelsystem, eftersom en kostnad/service-kvotsbegränsning kan sättas gemensamt för en grupp av artiklar. Detta skulle medföra att artiklar som inte lämpar sig för nödorder, undviker att använda dem.

Abbreviations

BS	Buffer Stock
EOQ	Economic Order Quantity
FOC	Fixed Order Cycle
FOQ	Fixed Order Quantity
G	Loss function
GIM	Syncron's Global Inventory Management System
h	Holding cost per unit and time unit
IL	Inventory Level
IP	Inventory Position
ORS	Operation Research Study
Q	Order quantity
R	Reorder point
(R, Q)	Single supplier ordering policy with reorder point R and order quantity Q
(R, Q)-LH	Emergency ordering policy based on an (R, Q) policy for a single supplier and a long horizon heuristic for emergency orders
(R, Q)-LS	Emergency ordering policy based on an (R, Q) policy where the R is determined according to a lost sales model. Lost sales are satisfied through emergency orders
(R, Q)-RH	Emergency ordering policy based on an (R, Q) policy for a single supplier and Syncron's rush ordering heuristic
(R, Q)-SH	Emergency ordering policy based on an (R, Q) policy for a single supplier and a short horizon heuristic for emergency orders

ROL	Run Out Level. A parameter used in Syncron's Rush order heuristic
S&Z	Emergency ordering policy by Song and Zipkin (2009)
s_1	Order up to level one in the emergency ordering policy by Song and Zipkin (2009)
s_2	Order up to level two in the emergency ordering policy by Song and Zipkin (2009)
S1	S1-service level or α service level
S2	S2-service level, fill rate or β service level
S3	S3-service level or ready rate
SL	Service Level
TSL	Target Service Level
TSV	Target Stock Value
VAU	Value of Annual Usage

Table of contents

1 Introduction

This chapter will give the reader an introduction to the studied phenomenon, describe the goal and boundaries of the research as well as introduce the company hosting the master's thesis.

1.1 Background

Syncron is a global leader in cloud based after market service optimization with numerous software applications such as pricing, inventory management and order handling as part of their product offering. Today Syncron's application for inventory control (hence referred to as GIM) includes a simple heuristic for emergency ordering based on stock-out probability, which does not take cost or achieved service level into account and is therefore hard to evaluate. Last spring Schmidt (2015) wrote a master´s thesis for the company, mapping publications of emergency ordering models. His paper summarized findings within the area and identified problems as well as benefits of implementing more advanced emergency ordering in Syncron's GIM software.

1.2 Problem formulation

The problem of finding optimal inventory policies for a single stock point with two supply options in inventory control is an old and well researched one. A multitude of research papers has been published in the area since Whittemore and Saunders (1977) and many different models have been presented. In practice, little of this research has trickled down and been implemented at firms making use of inventory control methods. The reasons behind this may range from the multitude of conditions attached to some models to the computational complexity of others.

We define emergency ordering as the ability to use two supply sources, a normal one and a faster more expensive emergency source. Based on Schmidt's (2015) research at Syncron and previous publications this thesis aims to answer:

- What are the theoretical and practical problems that prevent Syncron from implementing an emergency ordering policy in GIM based on existing research within the field?
- If an emergency ordering policy could be implemented in GIM under what circumstances should the policy be used?
- How would stated ordering policy perform in comparison to their current heuristic? Against a policy without emergency ordering?

1.3 Purpose

The purpose of this thesis is to investigate the challenges of introducing emergency ordering policies in GIM and try to identify and evaluate such policies against their current heuristic.

1.4 Delimitations

What to study

The research we aim to carry out should review the conditions at Syncron under which an emergency ordering policy must operate to be implementable, identify an ordering policy that can be used under such conditions, and evaluate under which circumstances in terms of policy parameters, demand patterns and service requirements the policy should be used.

A case study should be conducted based on customer demand data from Syncron to evaluate the chosen ordering policy. The data should come from a retailer facing direct customer demand and be narrowed down to a representative sample of items.

What not to study

Focus will be on reviewing existing ordering policies through a literature review. An identified policy should be based on previous research papers and may be adapted to fit with requirements. We will not try to create new concepts but rather pick and adapt existing ones from literature and Syncron's inventory management system.

2 Methodology of this thesis

This section describes the methodology used in this master's thesis.

The research methodology used in this thesis is similar to the Balanced Approach described by Golicic et al. (2005) in terms of including both qualitative and quantitative loops. The research centers around a problem or phenomenon to which contributions are made iteratively. The early parts of the research project were dominated by qualitative research, providing substantive theory necessary to identify variables for quantitative research. After identifying ordering policies to solve the subparts of the studied problem, the ordering policies were quantitatively tested through simulation.

Previous research has shown that the solution to the general problem of emergency ordering with two supply sources is highly complex, see Whittemore and Saunders (1977). To make the problem manageable our problem statement was narrowed down to applications that can be used at Syncron. Briefly, it was concerned with the compatibility with Syncron´s inventory management system GIM and the industrial environment that GIM operates in, especially minimizing costs under a service level constraint. Our work therefore has similarities to a case study in that we closely examine conditions in one organization. But the goal of our research was not the same as that of a classical case study and therefore we did not employ it as our primary method of study.

The subject area of this thesis, Inventory Control, is part of the field Operations Research (OR). In addition to this our research was carried out at a specific company using input data from their database. Our aim was therefore to choose a research approach that fitted this situation. With this motivation we believe that the best approach to take was that of an Operations Research Study (hereafter referred to as ORS), as defined by Hillier and Lieberman (2005, pp. 8-24). The approach can be viewed through a series of overlapping steps.

1. Define the problem of interest and gather relevant data
2. Formulate a mathematical model to represent the problem
3. Develop computer-based procedure for deriving solutions to the problem from the model
4. Test the model and refine it as needed
5. Prepare for the ongoing application of the model as prescribed by management (not applied in this thesis)
6. Implement (not applied in this thesis)

The fifth and the sixth step of the approach were not included in this master's thesis. Possibility to implement the policy was an important aspect throughout this report but the implementation decision was left to the company. The four first steps of our adapted approach of an ORS are described next.

2.1 Define the problem of interest and gather relevant data

The first step of an ORS concerns a big part of our research project, including problem identification, data gathering, literature review and qualitative description of the problem.

2.1.1 Identify the problem

The problem was initially described by Syncron and then refined to the problem formulation and purpose, stated in the chapter 1, through multiple iterations with the supervisor and the examiner.

2.1.2 Data collection

In order to completely understand the problem, data was collected through interviews at Syncron as well as studying internal documents describing GIM. This ensured that we and Syncron had a common understanding of the problem.

Informal interviews with employees at Syncron were performed during the whole course of the project. Initially interviews and company presentations aimed at providing us with an understanding of GIM and the rush ordering heuristic as well as to the industrial environment that GIM is operating in. Later in the project the focus of our meetings was shifted towards validating data and assumptions as well as ensuring that our research remained relevant for the company.

2.1.3 Literature review

To supplement the information gathered at Syncron, a literature review was performed that mapped publications on emergency ordering models. The literature review performed in this project is similar to Mayring's (2003) process for content analysis, discussed in *Research Methodologies in Supply Chain Management* (Kotzab et al. 2005, pp. 94-95). Our adaption of this approach follows below.

Scope

The initial scope of the literature review was broad including all types of models within the field of emergency ordering. The scope was narrowed down through numerous iterations, ultimately ending up in a few relevant articles.

Material collection

Material was collected with four different methods, outlined below.

- Schmidt (2015) performed a literature study in his master's thesis. The articles identified as relevant there were prequalified.
- The expertise and experience of Johan Marklund and Peter Berling within this field were utilized to gain knowledge of interesting articles on the topic.
- Reference lists from already chosen articles (snowball technique) were examined for other interesting publications.

- Articles with interesting models were searched for citations.

Criteria for selection

To compile the first list of material, the titles and the abstracts of the articles were read. The articles were selected based on subjective assessment of the relevance. The first list consisted of total of 49 articles, which are presented in Appendix I.

Classification of models

The dimensions examined as a basis for our classification were initially derived from our problem specification and through supplementing discussions with our supervisors. These dimensions were revised to achieve a better match with the existing literature. The final dimensions are presented below.

- Periodic vs. continuous review
- Demand distribution
- Deterministic vs. stochastic lead times
- Handling of unsatisfied demand
- Policy/decision mechanism
- Information required to implement the model
- Constraints under which the model is valid.

The 49 articles were then examined in terms of these dimensions and 10 articles were chosen for a detailed review based on deemed relevance for Syncron. Having reviewed the articles and discussed problems and opportunities with the different models with our supervisors, it was decided to classify the policies. The first differentiator was whether the polices take the timing of outstanding orders into account or not. Those that do were further classified based on how many indexes (i.e. inventory positions or inventory levels) they track and if the review is continuous or periodic. The policies that do not take the timing of outstanding orders into account, were all single index policies. These were separated based on the number of reorder points that they use. The classification is described in detail in section 5.2.

2.1.4 Qualitatively describe the problem

After collecting information about GIM and the literature review, the first research question - *What are the theoretical and practical problems that prevent Syncron from implementing an emergency ordering policy in GIM based on existing research within the field?* - was answered. This was done using qualitative information, describing our perception of the phenomenon based on the interviews and the literature review, in an attempt to explain the problems Syncron is facing. The ultimate problem definition was finally derived from both case specific aspects and limitations of academic research.

2.2 Formulate mathematical models to represent the problem

In the second phase of the ORS our qualitative description of the problem was translated into several sub problems which could be solved with different mathematical models. Representative models were found in literature, but models were also tailor-made to better suite the constraints that GIM imposed on the general problem. The final model choices were based on implementation possibilities and gauged overall value for the company. These choices were made in consensus with Syncron and the supervisors of the master's thesis.

We refer to chapters 3 and 4 for the underlying mathematical framework and descriptions of the model environment. The different mathematical models aspiring to represent and solve the problem are described in chapter 5 and 6.

2.3 Develop a computer-based procedure for deriving solutions to the problem from the model

Having decided on representative mathematical models that could solve the problem (see chapter 6), computer based procedures were derived. Excel and Visual Basic were used to implement the algorithms of analytically determining ordering policy parameters. The analytical models are described in section 8.1.

2.4 Test the model and refine it as needed

The fourth phase of the ORS concerns the testing of the policy. In this research project, discrete event simulation based on real data was used to compare the performance of the proposed emergency ordering policies to the polices with and without rush ordering heuristic currently used by Syncron.

2.4.1 Simulation

When evaluating the performance of e.g. an ordering policy, simulation can be a valuable tool since "... simulations try to combine the clarity and generality of mathematical modeling with the practical relevance and external validity of empirical research." (Kotzab et al. 2005, p. 445) In order to ensure reliable test results, five important concepts mentioned by Laguna and Marklund (2005, p. 267) were carefully dealt with:

- Selection of time units
- Length of a run
- Number of runs
- Model verification
- Model validation

Choosing length and number of runs was a trade off between simulation time and accuracy of results. Longer simulation runs lead to a lower standard deviation of estimated averages. Ideally this should be below a threshold value but this was found to be impossible for all our items. When increases in length gave marginal decreases in standard deviation we therefore accepted that value.

The models were verified in different ways. First we made sure that we got the expected simulation results for configurations that could be modeled analytically as well. Secondly the process flows and decision procedures were scrutinized and the handling of unusual situations checked.

In order to get more reliable results sensitivity analyses were performed, simulating the ordering policies for several different parameter values. The results of the simulations were evaluated to gauge whether the proposed order policies outperformed Syncron's rush ordering policy and whether it is sensible to use emergency ordering at all.

2.4.2 Gathering input data

Data gathered from the case company was used as input to the model. The sampling technique used to select test items for the simulation was a mixture between simple random, judgmental and convenience sampling, as defined by Doyle (2011).

Convenience sampling was applied in the sense that we were using data from the first Syncron customer that was willing to share data with us for this project. The retailers selected were also chosen out of convenience, with the only constraints that they should have items with multiple replenishment options, sufficient transaction data and high enough demand.

A list of the items present at the chosen retailers was handed to us, and items satisfying our requirements were sorted in different groups based on demand type. From every group a few items were picked. Some were randomly chosen but we also intentionally selected some items that depicted interesting characteristics, and that would thus be valuable for our research. All in all, we wanted to perform a broad study that covered many combinations of cost and demand characteristics.

Goodness-of-fit tests were also performed on the demand data. This is covered in detail in section 7.3.

2.5 Method for validating models and results

The credibility of results is often expressed in words of validity, reliability and objectivity. Validity concerns whether we measure what we intend to measure. Reliability concerns the repeatability of the results, assuring that the results are not coincidental. Finally, objectivity refers to if the results are influenced by who performs the study.

In chapter 6 all factors influencing model choices are presented. In short the aim was to cover a large part of the problem area, making this research relevant to Syncron. The process for validating and verifying our simulation models was explained in section 2.4.1. A similar procedure was used to validate the analytical models built in excel, see chapter 8.3. This verified our computations and measurements, i.e. that our models were internally valid. To ensure reliability long statistical runs where used in the simulations. A trade of was made between this length and the number of simulations. Extensive documentation of our test approach as well as descriptions of the simulation models were added to the report and appendix to ensure replicability. Throughout the whole study both authors have strived to remain objective.

3 Theoretical framework

This chapter will explain all theory necessary to follow the remaining parts of this report. If you are already familiar with inventory control theory you can just use this chapter as a dictionary. Much of the theory is taken directly from the book Inventory Control by Sven Axsäter (2006) and we refer to this book for more detailed information.

3.1 Basic definitions

3.1.1 Single echelon single item inventory system

A single echelon inventory system is an inventory system where only one location is considered at a time. A multi echelon inventory system also considers other installations, e.g. a central warehouse and a number of distributions centers. In this thesis we only focus on one single installation at a time.

A single item inventory system assumes that an item is not affected by other items and that optimizations are performed considering only one item at a time. This means that we disregard all kinds of impacts from other items, for example requirements of consolidating orders of different items to reach a certain order volume.

3.1.2 Inventory level and inventory position

In inventory control we are usually interested in controlling both the inventory level (IL) and the inventory position (IP). Axsäter (2006, p. 46) defines these as

Inventory level	=	*stock on hand – backorders*
Inventory position	=	*inventory level + outstanding orders*

The stock on hand is the physical stock in a warehouse, backorders are sales orders that have not left the warehouse yet, including reservations for a future date, and outstanding orders are purchase orders that are yet to arrive at the warehouse. Syncron is monitoring the same parameters but call them effective stock (=IP) and stock balance (=IL) instead.

3.1.3 Continuous and periodic review

In a continuous review system the inventory position and the inventory level are reviewed in real time. This means that as soon as an event happens, e.g. a customer places an order, the system is updated and a decision has to be made whether to place an order or not, and in case of placing an order how many pieces to buy. In a periodic review system the inventory level and the inventory position are only monitored at discrete time points, one review period T apart. Naturally as the review period T goes to 0, the periodic review will be the same as continuous review. (Axsäter 2006, p. 47) In the next paragraph we explain a few ordering policies suitable for continuous and periodic review systems.

3.1.4 Basic ordering policies

(R, Q) policy

An (R, Q) policy is a common ordering policy within inventory control. R is the so-called reorder point, telling us when to place an order. If the inventory position falls to or below R, an order is placed. The quantity is determined by the second parameter Q. In case of periodic review another parameter T has to be included, telling us when to compare the inventory position with R. (Axsäter 2006, p 48)

(s, S) policy

An (s, S) policy has a reorder point s and an order up to level S. When the inventory position falls to or below s, we place an order that brings the inventory position back to S. The quantity will thus be $S - IP$. The (s, S) policy is under general conditions the optimal policy for a single echelon inventory system. (Axsäter 2006, p. 49)

(S, T) policy

The (S, T) policy is an order policy used in periodic review systems. The inventory position is reviewed every T time unit and an order is placed to bring the inventory position back to S. The (S, T) policy also goes under the names S policy, order-up-to-S policy and base stock policy. The order quantity will be $S - IP$. (Axsäter 2006, p. 49)

(S-1, S) policy

The (S, T) policy is only used in a periodic review setting. However, as T goes to zero it will be equivalent to the (S-1, S) policy, i.e. an (R, Q) policy with $Q = 1$ and $R = S - 1$. Every time the inventory position is decreased an order is placed to bring the inventory position back to S. Note that the inventory position will always be S and the order quantity will be the same quantity that was ordered by the customer. (Axsäter 2006, p. 49)

3.1.5 Service level definitions

There are numerous definitions of service level, we will present three common measures here.

S1

S1, often referred to as α service level, is the probability of no stock out during an order cycle. S1 is simple to use and to understand but does not take the order quantity into consideration. Thus it should only be used with care in practice. For a simple (R, Q) policy we have

$$S1 = P(D(L) \leq R), \tag{3.1}$$

where $D(L)$ is the demand during the lead time. (Axsäter 2006, pp. 94-97)

S2

S2, also referred to as fill rate or β service level, is the fraction of demand satisfied directly from stock on hand. In contrary to S1 the fill rate takes the batch quantity into account. If demand is continuous or pure Poisson, that is a customer cannot order more than one unit at a time, we have

$$S2 = P(IL > 0) = 1 - P(IL \leq 0). \tag{3.2}$$

In other situations, for example if demand is compound Poisson distributed, we calculate the fill rate as expected quantity satisfied from stock on hand divided by the expected total quantity demanded. (Axsäter 2006, pp. 97-99)

S3

S3, or ready rate, is the fraction of time with positive inventory level. It can be defined as

$$P(IL > 0) = 1 - P(IL \leq 0). \tag{3.3}$$

Note that for continuous or pure Poisson demand, fill rate and ready rate are equivalent. (Axsäter 2006, pp. 97-99)

3.2 Deterministic model for determination of order quantities

For successful inventory control, using any of the above policies, values of the parameters need to be determined. There are approaches to joint optimization of the parameters but often, computational complexity prevents simultaneous optimization of all parameters in large systems. Therefore, it is common to optimize one parameter before the other. In this chapter we explain the most common way of determining the order quantity.

3.2.1 Economic order quantity

One of the most well known models within inventory control is the economic order quantity (EOQ), also referred to as the Wilson formula. The formula was first derived by Ford W. Harris in 1913 and balances holding costs and ordering costs in an optimal way, given constant deterministic demand and lead times and no backorders allowed. We introduce the following notation:

d	demand per time unit
A	ordering cost
h	holding cost per unit and time unit
Q	quantity ordered
C	total cost

The average inventory level will be $Q/2$ and we will order every d/Q time unit. See Figure 1. The total costs can thus be expressed as

$$C = \frac{Q}{2} * h + \frac{d}{Q} * A. \tag{3.4}$$

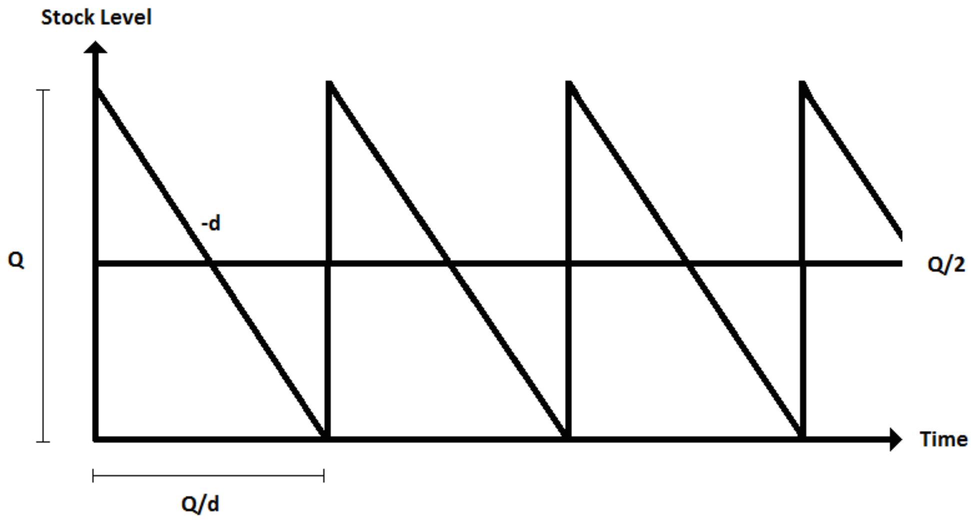


Figure 1: Constant demand model

Since C(Q) is convex the first order condition will give us the Q that gives the lowest costs. The EOQ formula is finally given by

$$Q = \sqrt{\frac{2Ad}{h}}. \tag{3.5}$$

If we allow backorders, see Figure 2, the expression becomes slightly more complicated but the same procedure can be used to derive it. We define b as the backorder cost per unit and time unit and α the fraction of the order quantity used to satisfy backorders. The cost function can now be expressed as:

$$C = (1-\alpha)^2 \frac{Q}{2} h + \alpha^2 \frac{Q}{n} b + \frac{d}{Q} A. \tag{3.6}$$

Since the cost function is convex in α we can set the derivative equal to zero, and obtain the optimal fraction as:

$$\alpha = \frac{h}{h+b}. \tag{3.7}$$

Inserting α in the cost function and setting the derivative with respect to Q equal to zero, we get the optimal order quantity as:

$$Q = \sqrt{\frac{2Ad(h+b)}{hb}}. \tag{3.8}$$

See Axsäter (2006, pp. 60-61) for more details.

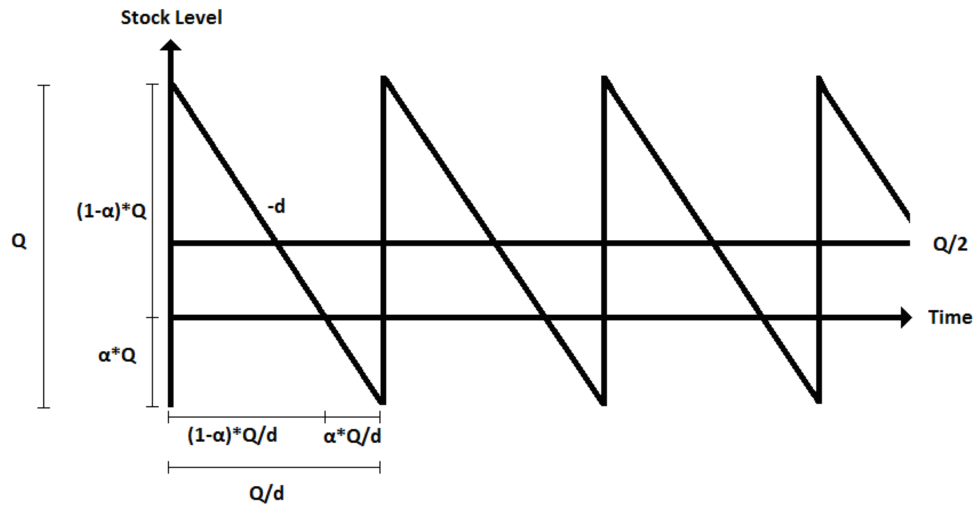


Figure 2: Constant demand model with backorders

In case demand is assumed to be constant we would have to order one lead time before we need the units. Thus R should equal the demand during the lead time.

3.3 Stochastic demand models

When modeling inventory systems an assumption must be made regarding how to represent customer demand. When demand is stochastic this is most commonly done by fitting a probability distribution to customer demand data. In the next section we will briefly explain the most common distributions used for this purpose. For more details, see Axsäter (2006, pp. 77-86).

3.3.1 Poisson demand

Customer arrival rates can often accurately be modeled as a Poisson process with some intensity λ. This can be interpreted as if customers arrive at random with the time between arrivals exponentially distributed and the total number of customers arriving during some time interval t Poisson distributed. In the case when each customer only orders one item this means that demand also is Poisson distributed, and we have probability of total demand k during time t

$$P(k) = \frac{(\lambda t)^k}{k!} e^{-\lambda t}, \qquad k = 0,1,2 \ldots \tag{3.9}$$

(Axsäter 2006, pp. 77-78).

When demand is low and independent from other items this is often a good assumption. In the spare parts market, where demand often derives from the stochastic break down of some machine, it is a common assumption.

3.3.2 Compound Poisson demand

If the time between customer arrivals is exponentially distributed and each customer may order several units of an item, demand can be modeled as a compound Poisson distribution. The amount each customer orders is then also a stochastic variable with a known distribution. If this amount is independent from that of other customers and from the arrival distribution the demand follows a compound Poisson process. Axsäter (2006, pp. 77-80) uses the following notation:

f_j	probability of demand size j ($j = 1,2, \ldots$)
f_j^k	probability that k customers give the total demand size j ($j = 1,2, \ldots$)
$D(t)$	stochastic demand in time interval t

The probability of demand j during time interval t then becomes

$$P(D(t) = j) = \sum_{k=0}^{\infty} \frac{(\lambda t)^k}{k!} e^{-\lambda t} f_j^k. \tag{3.10}$$

3.3.3 Logarithmic compounding distribution

It is possible to fit any kind of discrete distribution to the stochastic variable for demand size. For the compounding Poisson distribution this is done by defining the probability vector. Another approach is to use a known distribution. In the case of a logarithmic distribution Axsäter (2006, pp. 80-82) expresses the probability that a customer orders j units as

$$f_j = \frac{\alpha^j}{\ln(1-\alpha)\,j}, \qquad j = 1,2 \ldots \tag{3.11}$$

It it possible to show that when the demand size distribution is logarithmic the demand distribution reduces to a negative binomial distribution:

$$P(D(t) = j) = \frac{r(r+1)\ldots(r+j-1)}{j!}(1-p)^r p^j, \qquad j = 1,2,\ldots \tag{3.12}$$

For discrete demand this is also subject to $r > 0$ and $0 < p < 1$. (Axsäter, pp. 80-82)

The logarithmic compounding distribution is suitable to use for low demand items where the variance of the lead time demand is larger than the mean. (Axsäter, p.85).

3.3.4 Normally distributed demand

When total demand rises to higher levels approximating the demand during the lead time by a continuous distribution becomes a reasonable choice. When demand per time unit is independent and identically distributed the lead time demand during L time units approaches a normal distribution as $L \to \infty$ because of the *central limit theorem.* As a consequence of this and the simplicity of its application, assumptions of normally distributed demand is very common in practice. The normal distribution has two parameters, mean μ and standard deviation σ, and its density function is defined, using a standardized form with $\mu = 0$ and $\sigma = 1$, as

$$\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}, \qquad -\infty < \mathrm{x} < \infty. \tag{3.12}$$

(Axsäter 2006, pp. 85-86)

An important property when using the normal distribution to model demand is that there always exists a probability for negative demand. This makes the assumption less reasonable when real demand is non-negative. Checking that the probability of negative demand is sufficiently small is important when modeling with the normal distribution.

3.4 Calculation of inventory level distribution

When a suitable demand model has been chosen to represent customer demand and the choice of policy has been made it becomes possible to evaluate the performance measures of the model. We are interested in the expected inventory level, the expected number of backorders or, correspondingly, the expected service level. All the measures can be derived once the distribution of the inventory level is known. This section will therefore demonstrate, using a continuous review (R, Q) policy, how the inventory level distribution can be determined under the four demand types listed earlier.

3.4.1 General distribution under (R, Q) policy with continuous review and discrete demand

Lets first define some basic properties of continuous review (R, Q) models. From Axsäter (2007, pp. 88-89) we know that in steady state the inventory position is uniformly distributed on the integers $R+1, \ldots, R+Q$. Under the assumption of constant lead times we can then define an important relationship. First we recall the notations from above.

L	Lead time
IL	Inventory level
IP	Inventory position
$D(t, t+L)$	Stochastic demand in the interval $[t, t + L]$

We can then state the relationship between *IL* and *IP* as

$$IL(t+L) = IP(t) - D(t, t+L). \tag{3.13}$$

Based on this relationship we can then formulate the distribution for the inventory level for discrete demand as

$$P(IL = j) = \frac{1}{Q} \sum_{k=max\{R+1,j\}}^{R+Q} P(D(L) = k - j), \qquad j \leq R + Q. \tag{3.14}$$

See Axsäter (2006, p. 90) for a longer explanation of equations 3.13 and 3.14.

Given that we now know how to calculate the inventory level distribution based on a general discrete demand model lets determine it exactly for the distributions previously listed.

3.4.2 Distribution under Poisson demand

All that now is needed is to define the vector of probabilities for $P(D(L) = k - j)$. In the case of Poisson demand this becomes

$$P(D(L) = k - j) = \frac{(\lambda L)^{k-j}}{(k-j)!} e^{-\lambda L} \tag{3.15}$$

and consequently we have

$$P(IL = j) = \frac{1}{Q} \sum_{k=max\{R+1,j\}}^{R+Q} \frac{(\lambda L)^{k-j}}{(k-j)!} e^{-\lambda L}, \qquad j \leq R + Q. \tag{3.16}$$

3.4.3 Distribution under compound Poisson demand

In analog, for compound Poisson demand we get

$$P(D(L) = k - j) = \sum_{i=0}^{\infty} \frac{(\lambda L)^i}{i!} e^{-\lambda L} f_{k-j}^{i} \tag{3.17}$$

and

$$P(IL = j) = \frac{1}{Q} \sum_{k=max\{R+1,j\}}^{R+Q} \sum_{i=0}^{\infty} \frac{(\lambda L)^i}{i!} e^{-\lambda L} f_{k-j}^{i}, \qquad j \leq R + Q. \tag{3.18}$$

Note that f_{k-j}^{i} is the same expression as f_j^k in section 3.3.2, that is the probability that i customers give the total demand size $k - j$ $(k - j = 1,2, \dots)$.

3.4.4 Distribution under logarithmic compounding demand

Analogously for logarithmic compounding demand the distribution is a negative binomial distribution and we get

$$P(D(L) = k - j) = \frac{r(r+1) \dots (r+k-j-1)}{(k-j)!} (1-p)^r p^{k-j}, \qquad k - j = 1,2, \dots \tag{3.19}$$

$$P(IL = j) = \frac{1}{Q} \sum_{k=max\{R+1,j\}}^{R+Q} \frac{r(r+1) \dots (r+k-j-1)}{(k-j)!} (1-p)^r p^{k-j}, \qquad j \leq R + Q. \tag{3.20}$$

3.4.5 Distribution under normally distributed demand

Under the assumption of normally distributed demand we have to modify our calculations to fit a continuous distribution. By applying the same approach as in case of discrete demand the equation becomes

$$P(IL \leq x) = \frac{1}{Q} \int_{u=R}^{u=R+Q} \left[1 - \Phi\left(\frac{u - x - \mu'}{\sigma'}\right)\right] du = \tag{3.21}$$

$$= \frac{\sigma'}{Q} \left[G\left(\frac{R - x - \mu'}{\sigma'}\right) - G\left(\frac{R + Q - x - \mu'}{\sigma'}\right)\right] \tag{3.22}$$

This is already specified for the normal distribution where G is the loss function

$$G(x) = \int_{x}^{\infty} (v - x)\varphi(v)dv = \varphi(x) - x(1 - \Phi(x)) \tag{3.23}$$

In practice however, the second G term in equation 3.22 is often neglected. Note that in equation 3.21 and 3.23 Φ is the cumulative distribution function of the standard normal distribution.

3.5 Optimization of (R, Q) policy

When an ordering policy and demand pattern has been determined the problem becomes what parameters to use. We will illustrate how this is done with the commonly used (R, Q) policy.

First one must determine whether to optimize the parameters based on a cost expression or to fulfill a target service level (TSL). These two are closely linked and can be thought of as two sides of the same coin. Achieving a TSL will result in a certain probability of stock out during each cycle. Setting the parameters to achieve the lowest possible total cost corresponds to a SL that is not known before the optimization.

Axsäter (2006, pp. 102-103) describes the optimization procedure for the (R, Q) policy. The total cost expression is convex in the reorder point. To find the optimal R with regards to cost we start at the lower bound $R = -Q$ and increase R in unit steps until the total cost is increasing. Recall that Q can be determined with the EOQ-formula presented in section 3.2.1. When optimizing to fulfill a TSL we use the exact same procedure but stop when R gives a $SL > TSL$.

Axsäter (2006, pp. 101-106) shows how to derive the backorder cost, per unit and time unit, that a given service level corresponds to, if all other cost parameters are known. We will just give the result here and refer to Axsäter (2006) for a complete derivation.

For compound Poisson demand we have the following relation between the ready rate and the cost parameters in the optimal solution R^* given a Q.

$$S3(R^*) \leq \frac{b}{h + b} < S3(R^* + 1) \tag{3.24}$$

For pure Poisson demand and continuous review $S_2 = S_3$. This is also true for normally distributed demand and continuous review but we have an equality instead, see Axsäter (2006, pp. 105). The relationship becomes:

$$S3(R^*) = \frac{b}{h + b} = S2(R^*). \tag{3.25}$$

By rewriting equation 3.25 we get an exact expression of the backorder cost.

$$b = \frac{S2(R^*)h}{1 - S2(R^*)} \tag{3.26}$$

The same can be done for equation 3.24, but with the difference that the backorder cost will be somewhere in the interval

$$\frac{S3(R^*)h}{1 - S3(R^*)} \leq b < \frac{S3(R^* + 1)h}{1 - S3(R^* + 1)}. \tag{3.27}$$

3.6 The newsvendor problem and lost sales

In this section we present two famous problems within inventory control. Both models take the cost of lost sales into account but only the second model will be referred to as the *lost sales model* in this thesis.

3.6.1 The newsvendor problem

In the newsvendor problem a newsvendor has to decide how many newspapers to buy for the next day, knowing that unsold papers will be worthless the day after. For every paper the newsvendor can sell he makes a profit c_u and for every unit he has to throw away he makes a loss of c_o. Since demand is seldom deterministic the newsvendor must weigh the cost of lost sales (underage cost) c_u against the throw-away costs (overage cost) c_o. Assuming normally distributed demand we can set up an expression for the expected cost:

$$C(Q) = c_o * \int_{-\infty}^{Q} (Q - x) * f(x)dx + c_u \int_{Q}^{x} (x - Q) * f(x)dx \tag{3.28}$$

where Q is the ordered quantity and $X \in N(\mu, \sigma)$ is the stochastic demand per period with mean μ and standard deviation σ. $f(x)$ can be expressed as

$$\frac{d}{dx}\Phi\left(\frac{x-\mu}{\sigma}\right) = \frac{1}{\sigma}\varphi\left(\frac{x-\mu}{\sigma}\right). \tag{3.29}$$

By inserting 3.29 in 3.28 and simplifying we get

$$C(Q) = c_o(Q - \mu) + (c_o + c_u)\sigma * G\left(\frac{Q-\mu}{\sigma}\right), \tag{3.30}$$

and with the first order condition (C(Q) is convex) we then get

$$\frac{dC(Q)}{dt} = c_o + (c_u + c_o)\left(\Phi\left(\frac{Q-\mu}{\sigma}\right) - 1\right) = 0 \tag{3.31}$$

and ultimately

$$\Phi\left(\frac{Q-\mu}{\sigma}\right) = \frac{c_u}{c_u + c_o}. \tag{3.32}$$

See Axsäter (2006, pp. 114-116) for more details and examples.

3.6.2 Lost sales model

Imagine a situation where a customer will not wait for an item. All demand that occurs during stock outs is then lost and consequently $P(IL < 0) = 0$. This situation is referred to as a lost sales model (Axsäter 2006, pp. 117-119). Let's modify this idea a little and assume that a customer is willing to wait for an emergency order, but not for a normal order. Demand during stock outs trigger emergency orders so that the customer always gets what he/she orders (This makes the name "Lost sales" somewhat misleading but we will keep it for sake of reference). We then set the cost of "lost sales" b to equal the extra cost of ordering through the emergency supplier. If the model operates as a standard (R, Q) policy under continuous review and normally distributed demand we can analytically find the reorder point that optimally balances holding cost and the extra cost of emergency orders.

This model comes with some restrictions. Axsäter (2006, p. 117) makes the assumption that $Q > R$. Our modification also requires that the emergency lead time is sufficiently low for all customers to accept it. Under this assumption there can never be more than one outstanding order at a time, since we cannot hit the reorder point while there is an order outstanding. The analysis focuses on an order cycle defined as the time between two consecutive orders. In order to derive the total cost during an order cycle we need to know the expected quantity of "lost sales" during an order cycle as well as the expected inventory level. We assume a holding cost h per unit and time unit and a shortage cost b per unit unsatisfied demand. Furthermore, we once again assume the demand is normally distributed with mean μ and standard deviation σ. The lost sales quantity will be

$$E[R - D(L)]^- = \int_R^\infty (u - R) * \frac{1}{\sigma'}\varphi\left(\frac{u-\mu'}{\sigma'}\right) du = \sigma' * G\left(\frac{R-\mu'}{\sigma'}\right). \tag{3.33}$$

Further, the expected inventory level just before an order arrives will be

$$E[R - D(L)]^+ = R - \mu' + E[R - D(L)]^- \tag{3.34}$$

and thus the average time the units in a batch of size Q are in stock is

$$E[T_Q] = \frac{Q}{2\mu} + \frac{E[R - D(L)]^+}{\mu}. \tag{3.35}$$

Before we set up the cost expression we also need the average length of an order cycle:

$$T = \frac{Q}{\mu} + \frac{E[R - D(L)]^-}{\mu}. \tag{3.36}$$

Finally we can express the expected total costs per time unit as:

$$C = \frac{h * E[T_Q] * Q + b * E[R - D(L)]^-}{T} = \tag{3.37}$$

$$= \frac{h * \left(\frac{Q}{2\mu} + \frac{R - \mu' + \sigma' * G(\frac{R - \mu'}{\sigma'})}{\mu} \right) * Q + b * \sigma' * G(\frac{R - \mu'}{\sigma'})}{\frac{Q}{\mu} + \frac{\sigma' * G(\frac{R - \mu'}{\sigma'})}{\mu}}. \tag{3.38}$$

Utilizing that $C(R)$ has a single optimum (Axsäter 2006, p. 118) we can find the reorder point that minimizes the costs with a simple linear search over integer reorder points R from $R = 0$, stopping when $C(R) \leq C(R + 1)$.

The fill rate can be expressed as

$$S2 = 1 - \frac{\sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} = \tag{3.39}$$

$$= 1 - \frac{\sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} - \frac{Q}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} + \frac{Q}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} = \tag{3.40}$$

$$= 1 - \frac{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} + \frac{Q}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} = \tag{3.41}$$

$$= 1 - 1 + \frac{Q}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} = \tag{3.42}$$

$$= \frac{Q}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)}. \tag{3.43}$$

4 Company study

In this chapter we provide relevant descriptions of Syncron's inventory management system GIM. The main focus will be on the ordering policies in use. We further state a few assumptions that we have to relate to when searching the academic literature for better alternatives to the rush ordering heuristic used by Syncron.

4.1 Global Inventory Management (GIM)

The target of Syncron's inventory management system is to reach a certain overall service level, defined as number of satisfied picks for all items in a product range divided by the total number of picks for these items, while minimizing the capital tied up in stock. In general, this means that inexpensive items with many picks should be stocked while expensive items that are seldom picked should not. In GIM a suitable service level can be determined both on a group level but also on an item level, depending on the customer's wishes.

First we explain how items are segmented in GIM. Then we describe the two different ordering policies that can be used on an item level. In section 4.1.3 we show how Syncron computes the buffer stock and in 4.1.4 we define the different service level definitions used by Syncron. Finally, we describe how Syncron manages inventory for a group of items, by introducing an optimization problem.

4.1.1 Segmentation of items

Syncron segments items into different classes based on four different parameters. These are picks, value of annual usage, demand and frequency. We will in this thesis only focus on the different demand classes. In total there are nine different classes: fast, positive trend, negative trend, erratic, slow, lumpy, obsolete, non-moving and new. New items do not have enough demand history to be classified into any of the other classes. Obsolete and non-moving items are being phased out and have very low demand. The two trend classes are closely connected to the fast class and the fast class will therefore be used as the representative for those items. The fast and remaining classes will be described in this section.

Fast

Fast items have a high and stable demand that can be assumed to be normally distributed.

Erratic

Erratic items are fast items where the standard deviation of the demand is higher than a given threshold value. These items are also assumed to have a normal demand distribution.

Slow

Items are considered to be of the slow demand type if they have zero demand in more than a given percentage of demand periods. A demand period is usually a month long. These items are assumed to have Poisson demand.

Lumpy

Lumpy items are slow items where the variation is considered high. That is, if the demand variance divided by the estimated demand is higher than a given threshold value greater than one, the item is considered to be lumpy. These items are assumed to have a negative binomial demand distribution.

4.1.2 Ordering policies

In GIM there are two different ordering policies to choose from. Below these will be explained.

FOQ – Fixed order quantity

The fixed order quantity policy (hereafter referred to as FOQ) is what in our theoretical framework, section 3.1.4, is referred to as an (R, Q) policy. The order quantity Q can be determined in numerous ways depending on available cost information and constraints but is by default given as the forecasted demand over the periods to cover. Periods to cover is a parameter that can be automatically set by dividing the EOQ from the Wilson formula, see equation 3.5, with the expected period demand.

The order level (or reorder point) R depends on the demand type:

Demand type	Order level
Fast	Buffer stock + forecasted demand (lead time)
Erratic	Buffer stock + forecasted demand (lead time)
Slow	Buffer stock
Lumpy	Buffer stock

How to calculate the buffer stocks is shown in the section 4.1.3.

FOC – Fixed order cycle

The fixed order cycle policy (hereafter referred to as FOC) is what in our theoretical framework, section 3.1.4, is referred to as an (S, T) policy. The order quantity for the FOC is the order level R (or order up to level S) minus the effective stock (inventory position) at the time of order placement. The order level R is the same as for the FOQ but with the difference that the review time T is added to the lead time, see below:

Demand type	Order level
Fast	Buffer stock + forecasted demand (lead time + review period)
Erratic	Buffer stock + forecasted demand (lead time + review period)
Slow	Buffer stock
Lumpy	Buffer stock

4.1.3 Buffer stock calculations

The parameters used in the ordering policies depend on the buffer stock (BS), which is Syncron's word for safety stock. The buffer stock depends on the target service level for an item as well as the demand type the item belongs to.

Fast and erratic items

Fast and erratic items are, as already explained, modeled with a normal demand distribution. Syncron uses below relation to calculate the buffer stock:

$$TSL = 1 - \frac{\sigma'}{Q} G(K) \Rightarrow \tag{4.1}$$

$$\Rightarrow G(K) = \frac{Q}{\sigma'}(1 - TSL) \tag{4.2}$$

where Q is the order quantity, TSL is the target service level, σ' is the standard deviation of the lead time demand and K is the safety factor. G is the loss function defined in chapter 3, see equation 3.23, and the translation between K and G can be found in tabulated form in Appendix III. From the definition of the safety factor we can then calculate the buffer stock:

$$K = \frac{BS}{\sigma'} \Rightarrow \tag{4.3}$$

$$\Rightarrow BS = K * \sigma'. \tag{4.4}$$

A natural and common interpretation of the buffer stock is that it consists of K standard deviations.

The calculations are very similar to the ones provided in chapter 3, see equations 3.21 and 3.22. However it seems that Syncron deems the term $G(K + \frac{Q}{\sigma'})$ to be negligible, which is a decent approximation if $\frac{Q}{\sigma'}$ is sufficiently large. However this is not always the case and in our analytical models (described in chapter 8) for computing policy parameters we therefore include the term $G(K + \frac{Q}{\sigma'})$.

For erratic items Syncron uses a maximum standard deviation limit. This is done to avoid high standard deviations causing very high reorder points, but was ignored in our analytical models since we did not have access to the necessary parameters.

Slow and lumpy items

Slow and lumpy items are modeled with a Poisson demand and a negative binomial distribution respectively. The buffer stock is calculated by taking the smallest n such that below inequallity is satisfied.

$$\sum_{k=0}^{n} f_X(k) \geq TSL \tag{4.5}$$

where $f_X(k)$ is the probability mass function of the Poisson and the Negative Binomial distribution respectively and TSL is the target service level. If $k = 0$, but the estimated demand is positive, k is set to 1 in order to avoid zero buffer stock for items that have low demand.

4.1.4 Service level definitions

The users of the GIM can select between three different service level definitions which all are computed for a given time period, usually a week or a month. The first, called "back orders", is the fill rate already seen in the theory chapter,

$$SL = \frac{Total\ demand\ quantity - total\ back\ orders}{Total\ demand\ quantity}.$$

The second and the third are defined the same way,

$$SL = \frac{Total\ number\ of\ customer\ order\ lines\ that\ could\ be\ fully\ delivered}{Total\ number\ of\ customer\ order\ lines}.$$

There is however one difference. When computing the service level with the second option, "at request date", the order lines need to be fulfilled at the requested date. When computing the service level with the third option, "over the counter", the order lines instead need to be fulfilled at the time of order arrival. An order line here is defined as any number of units ordered for a single item and a single customer at a single occasion.

4.1.5 Optimizing policy parameters

In GIM, the primary target is to attain an overall target service level for a group of items while minimizing the capital tied up in stock. In this section we explain how this is done by setting up an optimization problem, and showing how to solve the problem. We start by introducing the following notations for a set of items, $i \in M$, $M = \{0, 1, \ldots, m\}$:

Decision variables

SL_i	service level for item i defined as the fraction of complete picks satisfied directly from stock.

Non-decision variables

$TSV_i(SL_i)$	target stock value for item i given SL_i. Stock value is defined as the average inventory level multiplied by the value of the item.

Constant parameters

P_i	picks per year of item i
TSL	the target service level for group M
a_i	the minimum service level allowed for item i
b_i	the maximum service level allowed for item i
A	the minimum target service level allowed for group M
B	the maximum target service level allowed for group M
C	the maximum target stock value for group M

The optimization problem can then be stated as

Objective function

$$Min \sum_{i \in M} TSV_i(SL_i) \tag{4.5}$$

Constraints

$$\frac{\sum_{i \in M} SL_i * P_i}{\sum_{i \in M} P_i} \geq TSL \tag{4.6}$$

$$\sum_{i \in M} TSV_i(SL_i) < C \tag{4.7}$$

$$0 \leq A \leq TSL \leq B \leq 1 \tag{4.8}$$

$$P_i \geq 0, \qquad \forall i \in M \tag{4.9}$$

$$TSV_i(SL_i) > 0, \qquad \forall i \in M \tag{4.10}$$

$$0 \leq a_i \leq SL_i \leq b_i \leq 1, \qquad \forall i \in M \tag{4.11}$$

Optimization algorithm

In order to solve the optimization problem we follow below algorithm.

1. Set $SL_i = a_i$, for all $i \in M$. This will also give us initial values for the target stock values $TSV_i(SL_i)$ and the achieved service level $\frac{\sum_{i \in M} SL_i * P_i}{\sum_{i \in M} P_i}$ for group M.
2. Increase the service level SL_i, for the item that has the lowest marginal increase in target stock value $TSV_i(SL_i)$ per additional pick covered, just enough to cover another pick. If the new solution violates any of the constraints, we instead look at the next least costly increase in service level.
3. If $\frac{\sum_{i \in M} SL_i * P_i}{\sum_{i \in M} P_i} \geq TSL$ is true we stop. Otherwise we go back to step 2.

4.2 Rush ordering heuristic

Even though the optimization scheme of section 4.1.5 is supposed to achieve an overall target service level, there are users that want to improve the service level for single items by placing emergency orders when the probability of stock out is at a certain level. Syncron has therefore developed a heuristic for placing emergency orders that have a higher unit cost but a shorter lead time. Syncron uses the word rush orders for this kind of orders. They are always added on top of the (R, Q) policy, which means that R and Q are determined before and independent of any rush order decision. We will hereafter refer to this heuristic as (R, Q)-RH.

4.2.1 (R, Q)-RH

We start with introducing a few important concepts that are used in (R, Q)-RH. See Figure 3 for a visualization of how (R, Q)-RH functions.

Rush order horizon	The rush order horizon is defined as [current date + emergency order lead time LT_E, current date + normal lead time LT_N], that is the time period in which we can avoid stock out with an emergency order but not with a normal order.
ROL	Run Out Level. This is a level that can be set by the user, that does not necessarily have to be zero. When the inventory level falls to or below this limit we consider this a stock out.
$\boldsymbol{L}$	Run Out Time Limit. This limit can be set by the user and describes the maximum stock out (as defined with ROL) time accepted without placing an emergency order, during the rush order horizon, given that demand is constant and equal to the mean demand per time unit. By default $L = 0$.

Stock out interval In (R, Q)-RH a stock out interval is in most cases defined as time between the arrival of two succeeding outstanding normal orders. However, there are two exceptions.

- The first stock out interval starts at the beginning of the rush order horizon and ends when the next outstanding order arrives.
- The last stock out interval start when the last outstanding order in the rush order horizon arrives and ends at the end of the rush order horizon.

The intervals are indexed from $i = 1$ to $i = I$ where $i = 1$ is the first interval and I the last. Note that we only consider the Rush order horizon and therefore outstanding emergency orders are not considered since they will arrive before the beginning of the emergency order horizon.

Stock out point In (R, Q)-RH there is one stock out point in every stock out interval, whether a stock out is expected to occur or not. The stock out point is the time at which the inventory level is expected to hit ROL, assuming constant demand. Thus it is defined as time

$$T_{intStart} + \frac{IL(T_{intStart}) - ROL}{D},$$

where D is the constant demand and $T_{intStart}$ is the time at which the interval starts. Note that according to this definition, stock out points can occur after the end of the interval.

Decision rule

In order to determine when to place an emergency order and how much to order the heuristic considers all stock out points during the emergency order horizon. The expected difference in time between a stock out and an order arrival is defined as T_i, and the expected difference between the run out level ROL and the expected inventory level just before the outstanding order arrives is defined as q_i, where $i = 1,2, \dots, I$.

$$T_i = \max(0, time\ of\ order\ arrival\ i - \ expected\ time\ of\ stockout\ i) \tag{4.12}$$

$$q_i = \max\,(0, ROL - Inventory\ level\ at\ time\ of\ order\ arrival\ i) \tag{4.13}$$

T_i and q_i will be zero if no stock outs are expected in the stock out interval i.

In order to decide if to place an emergency order we compare the total expected stock out time $T_{Total} = \sum_{i=1}^{I} T_i$ with the Run Out Time Limit (L). If $T_{Total} > L$ we place an emergency order instantaneously. The quantity of the emergency order is determined by taking the maximum of all stock out quantities q_i during the emergency order horizon, $Q_{max} = \max\,(q_1, \dots, q_I)$. This will, assuming constant demand, also cover all other stock outs during the emergency order horizon. To summarize:

If $T_{Total} > L$, *order* Q_{max} *units with emergency order lead time* LT_E.

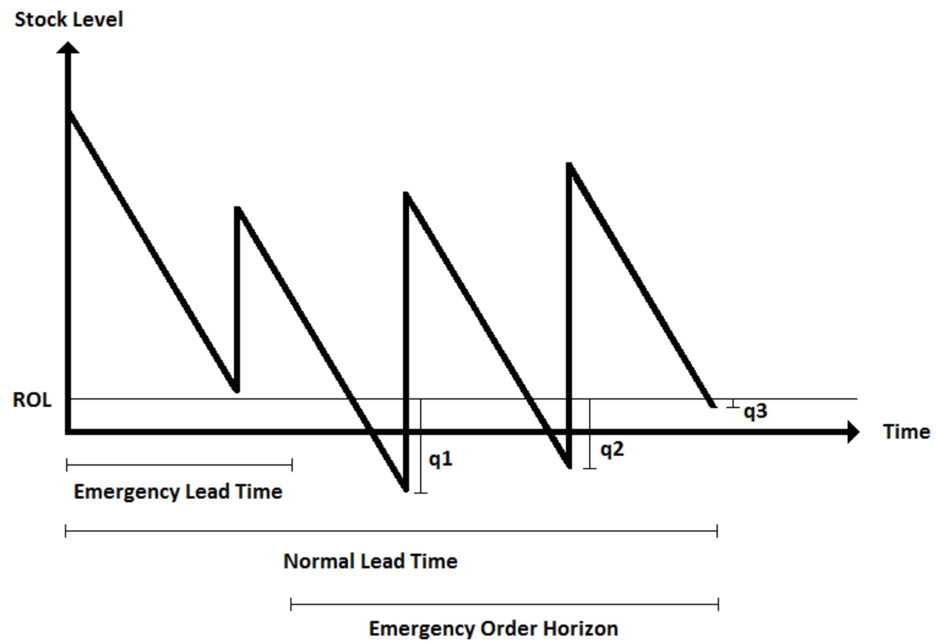


Figure 3: The figure shows the expected demand in the emergency order horizon. Note that we would here, with (R, Q)-RH, order q1 units immediately. This order will also cover the two other expected stock outs during the emergency order horizon.

4.3 Assumptions made about GIM

This section specifies the assumptions and conditions that we have to relate to when choosing and building a suitable emergency ordering policy for Syncron. The assumptions are made to assure compatibility between a new emergency ordering policy and GIM.

Assumption 1	Lead times are constant.
Assumption 2	Emergency lead time and normal lead time can be extracted from the system. The emergency lead time will always be strictly shorter than the normal lead time.
Assumption 3	Holding cost can be extracted from the system.
Assumption 4	Unit cost for normal and emergency orders can be extracted from the system. The unit price for normal orders is always lower than for emergency orders.

Assumption 5	Ordering cost or a given order quantity can be extracted from the system.
Assumption 6	Back order costs can not be directly extracted from the system.
Assumption 7	A target service level will be provided by the system.
Assumption 8	Although Syncron's optimization algorithms assume continuous review, most customers send stock files to GIM with information about inventory levels, inventory positions and outstanding orders on a daily basis. In our simulation study we will assume continuous review.

5 Literature study

In this chapter we present the most relevant articles from our literature review. The complete list of articles reviewed can be found in the Appendix I. For a more complete review of the research in the area of emergency ordering, we refer to literature reviews by Minner (2003) and Schmidt (2015). We have also added an adaption of the lost sales model that was described in section 3.6.2. In section 5.2 the models presented in this chapter are classified according to a number of differentiating aspects.

5.1 List of emergency ordering models

5.1.1 Arts (2009)

Arts (2009) provides a Dual-Index-policy where he uses order up to levels and both normal and emergency inventory positions. The review is periodic and lead times can be both stochastic and deterministic. The deterministic lead times must be integer multiples of the review period and the difference between the lead times must be at least one review period. The costs considered are the holding cost and the extra unit fee for emergency orders. The objective function is the sum of the holding costs and the total extra emergency delivery costs, and is minimized under a service level constraint that is slightly tougher than the normal fill-rate. The backorder costs are thus not explicitly considered and the ordering cost is disregarded. The policy is similar to the policy by Veeraraghavan and Scheller-Wolf (2008) but the method for evaluation of the so called overshoot is an approximation that is 20-30 times faster according to Arts.

One major drawback of Arts' (2009) model is that the optimization of the policy parameters is quite complicated from a mathematical perspective. Arts (2009) also presents a solution to the problem when stochastic instead of deterministic lead times are considered.

Review	Periodic review for normal order Periodic review for emergency orders
Demand	In theory every discrete or discretized continuous demand distribution.
Lead time (L_n/L_e)	Deterministic/Deterministic or Stochastic/Stochastic or a mix
Unsatisfied demand	Backordered
Policy/decision mechanism	Periodic review (S, T) policy called a dual-index policy. Ordering decisions are based on tracking regular and emergency inventory positions and their order up to levels. First an order is placed with the emergency supplier so

	that the emergency inventory position reaches its order up to level. Then a regular order is placed so that the regular inventory position reaches its order up to level.
Information needed	• Variable cost (normal) • Variable cost (emergency) • Holding cost • Lead time (normal) • Lead time (emergency)
Constraints	• Normal lead time minus emergency lead time ≥ 1 review period • Lead times are assumed to be integer multiples of the review period

5.1.2 Axsäter (2006) adapted – A lost sales model

This model is originally presented by Rosling (2002) and referred to in Axsäter (2006). The lost sales model does not explicitly model the second supplier option. Since this has proven to be a highly complex and state dependent problem for longer lead times and large batch quantities, the model works under the assumption that all stock outs will immediately trigger emergency orders. Therefore it is a reactive policy that does not try to forecast stock outs. The main benefit of the model is its simplicity while still comparing the costs of the two supplier options. It can also be modified for other types of demand.

Review	Continuous review for normal orders Continuous review for emergency orders
Demand	Normally distributed
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Triggers emergency orders
Policy/decision mechanism	(R, Q) policy that triggers emergency orders only at stock out with a penalty p that is the extra cost of the emergency shipment. Optimizes R to balance the holding cost against the penalty cost.
Information needed	• Variable cost (normal) • Variable cost (emergency)

	• Holding cost • Lead time (normal) • Lead time (emergency)
Constraints	• Every time stock outs occur, an emergency order should be placed so that they really incur the penalty cost. • Emergency Lead times must be short enough so that the customer always gets what she ordered, i.e. the lead time should be short enough so that no real lost sales occur. • Q must be bigger than R, so that at most one order is outstanding at a time.

5.1.3 Axsäter (2007)

Axsäter (2007) presents a model that builds on a continuous review (R, Q) policy for the single supplier case. To this an emergency ordering heuristic is added. The costs considered are holding costs, backorder costs, normal and emergency ordering costs and an additional unit cost for emergency ordering. When making the emergency ordering decision, the total extra cost of the emergency ordering compared to cost incurred if no emergency order is placed, is considered. At the decision point, no further emergency replenishments in the future are taken into account. Furthermore, the actual purchase value is not considered when calculating the holding costs, but this cost is only proportional to the number of units in stock. Finally, demand is assumed to have a compound Poisson distribution and there is no service level constraint.

The Axsäter method includes quite complex computations. This together with its restriction to the compound Poisson demand and its dependence on a backorder cost are the major drawbacks.

Review	Continuous review for normal orders Continuous review for emergency orders
Demand	Poisson/Compound Poisson
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	(R, Q) policy with opportunity to place an emergency order every time demand is registered. R and Q can be determined from the single supplier model (this is not

necessarily optimal for the emergency ordering case). When deciding whether to initiate an emergency order or not we assume that there are no other emergency orders. When demand is registered the emergency opportunity is evaluated, if cost savings are achieved we place an emergency order.

Information needed	• Variable cost (normal) • Variable cost (emergency) • Ordering cost (normal) • Ordering cost (emergency) • Holding cost • Backorder cost • Lead time (normal) • Lead time (emergency)
Constraints	• Emergency order quantity is assumed to be big enough to bring IP over R. • A normal order cannot be triggered at the same time as an emergency order. • Complete pipeline information is required

5.1.4 Axsäter (2014)

Axsäter (2014) builds on Axsäter (2007). It has the same modeling assumptions but the emergency ordering decision is now reviewed periodically, with the added improvement that one can now consider to place an emergency order a fix integer k review periods later in time. As in Axsäter (2007) an emergency order is only placed if the least expensive alternative is to place the emergency order at the time of the review.

5.1.5 Chiang and Gutierrez (1996)

Chiang and Gutierrez (1996) provide a periodic review order up to policy where the decision is whether to use a faster delivery mode with higher ordering cost or a slower delivery mode with lower ordering cost. The unit cost is assumed to be the same for both delivery modes. The other costs considered are holding costs and backorder costs. The emergency lead time is restricted to be shorter than the review period and no order crossing is allowed. The parameters to determine for a given review period are the order up to level and an indifference level of the inventory position at which the decision to use the emergency or regular order is altered. The total cost is minimized without a service level constraint and the parameters are determined with step-wise search on analytically determined intervals.

One major drawback with this model in our context is that the variable costs are equal for both ordering alternatives.

Review	Periodic review for regular orders Periodic review for emergency orders
Demand	Normal/Poisson
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	Order up to policy, (S, T). Each period either a normal or an emergency order is placed to reach the order up to level S (denoted R in the article). For non-negative order up to levels either only the normal order will be used or there will exist an indifference inventory level where the replenishment mode is changed. If the inventory position is higher than the indifference level a normal order is placed, otherwise an emergency order is placed. There are thus three parameters to be decided (review time T, indifference level I and order up to level S).
Information needed	• Ordering cost (normal) • Ordering cost (emergency) • Holding cost • Backorder cost • Lead time (normal) • Lead time (emergency)
Constraints	• The unit cost for emergency and normal orders are the same • Non-negative order up to level: $S \geq 0$ • No cross overs: normal lead time − emergency lead time < review period

5.1.6 Johansen and Thorstenson (1998)

Johansen and Thorstenson (1998) present two continuous review ordering policies based on an (R, Q) policy for normal orders and (s, S)-policy for emergency ordering. The first policy does not take time left until a normal order arrives into account while the second does. The reorder points are based on net stock (the same as inventory position for normal orders when no backorders are outstanding) and the normal lead time is considered to be an integer multiple of the emergency lead time. Both lead times are considered to be constant. Poisson demand is assumed, which is approximated by a Bernoulli process during the time that a normal order is outstanding. A normal order can only be placed when there is no other order outstanding and an emergency order can only be placed when there is already an outstanding normal order.

The optimal parameters for the second policy are determined through an iterative algorithm that finds the lowest cost given a normal order quantity Q. The total cost is a quasi-convex function of Q which means that an optimal Q can be determined. The difference in performance between the two policies is quite small.

Review	Continuous review for normal orders Continuous review for emergency orders
Demand	Poisson
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	(R, Q) policy for normal orders. The normal orders are only being placed if no orders (neither normal nor emergency) are outstanding. The emergency orders are controlled with a reorder point r(j) and an order-up-to level u(j), where j is the remaining time until the normal orders arrives, thus controlled with an (s, S)-policy. Because normal orders are only placed when no other orders are outstanding one can say that both reorder points are based on the net stock.
Information needed	• Variable cost (normal) • Variable cost (emergency) • Ordering cost (normal) • Ordering cost (emergency) • Holding cost • Backordering cost as a fixed cost per unit backordered or as a cost per unit and time unit • Lead time (normal) • Lead time (emergency)
Constraints	• Only one outstanding normal order • Normal lead time is an integer multiple of the emergency one • Emergency orders only allowed when one normal order

is outstanding

- Backorders are restricted to a maximum number B

5.1.7 Johansen and Thorstenson (2014)

The model provided by Johansen and Thorstenson (2014) is an extension of their earlier work from 1998. An (R, Q) policy is used for the normal orders and a state dependent (s, S) policy is used for emergency orders. The difference compared to their 1998 method is that the inventory position (instead of net stock), defined as the sum of the net stock and number of already ordered units that have a remaining lead time shorter than the emergency lead time, is periodically reviewed. Furthermore, lead times are not restricted to multiples of the review period and multiple normal orders can be outstanding as long as only one has a remaining lead time that is longer than the emergency lead time. The costs considered are the same as for the 1998 continuous review model except that a cost per backordered unit is not considered anymore. The lead times are assumed constant and the demand has a compound Poisson distribution. The optimal parameters are found with a policy iteration algorithm that according to the authors is faster than Axsäter's from 2007. The objective is to minimize the long range average cost per review period, without any service level constraint. Another difference compared to the 1998 model is that backorders are not restricted to a maximum number B anymore.

Review	Periodic review for normal orders Periodic review for emergency orders
Demand	Compound Poisson
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	Discrete-time Markov decision model. Normal orders are placed based on an (R, Q) policy while emergency orders are controlled with a state dependent reorder point s and an order up to level S i.e. an (s, S) policy. The reorder points are based on the inventory position defined as the sum of the net stock and number of already ordered units that have a remaining lead time shorter than the emergency lead time. A rapid policy iteration algorithm is used to find the optimal parameters.
Information needed	• Variable cost (normal) • Variable cost (emergency)

- Ordering cost (normal)
- Ordering cost (emergency)
- Holding cost
- Backorder cost
- Lead time (normal)
- Lead time (emergency)

Constraints	• Only one normal order with remaining lead time longer than the emergency lead time is allowed at any time • The time difference in lead time between normal and emergency orders must be a multiple of the review period

5.1.8 Moinzadeh and Nahmias (1988)

Moinzadeh and Nahmias (1988) provide an approximately optimal ordering policy for continuous review, with two reorder points and two order quantities. The model has similarities with the (R, Q) policy, not the least in terms of its simplicity, but compares on-hand inventory instead of the inventory position to the reorder points. Variable costs and ordering costs for both supply modes are included and a backorder cost per unit is required. All in all, it is a fairly general model that can handle both Gaussian (normal) and Poisson demand. The policy parameters are derived through an extensive algorithm, where the normal reorder point is determined from single mode optimization. When optimizing, it is assumed that only one order of each kind can be outstanding at any time. When the parameters have been determined, the policy will however work just as a normal reorder point policy, meaning that several orders of each type can be outstanding at the same time.

Review	Continuous review for normal orders Continuous review for emergency orders
Demand	Poisson/Normal
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	(R1, R2, Q1, Q2) policy based on on-hand inventory. When on-hand inventory reaches R1, we order Q1 units with the normal supply option and when on-hand inventory reaches R2 we order Q2 units with the emergency supply option.

Information needed	• Variable cost (normal) • Variable cost (emergency) • Ordering cost (normal) • Ordering cost (emergency) • Holding cost • Backorder cost per unsatisfied demand • Lead time (normal) • Lead time (emergency)
Constraints	• Never more than a single order outstanding of each type

5.1.9 Song and Zipkin (2009)

The basic idea in the model by Song and Zipkin (2009) is to use information regarding the timing of outstanding orders in the best possible way. The authors make extensive use of queueing theory to create a model that has a defined distribution for the inventory level. Under the assumptions of one for one ordering and Poisson demand for both normal and emergency orders, the policy is cost minimizing. Backorder costs are however needed. Song and Zipkin also derive solutions to different variations of the problem. The two most interesting are the inclusion of batch ordering and the assumption of stochastic lead times respectively. The first has the restriction that the same order quantity q must be used for normal and emergency orders. The second is very complex, but it was shown that it had significant benefits compared to a constant lead time model when lead times have high variability.

Review	Continuous review for normal orders Continuous review for emergency orders
Demand	Poisson/Compound Poisson
Lead time (L_n/L_e)	Stochastic/Stochastic or deterministic/deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	(S, S-1) continuous review ordering policy making use of two inventory positions for normal and emergency ordering. Their policy therefore maintains $IP_1 = s_1$ and $IP_2 \geq s_2$.
Information needed	• Variable cost (normal)

- Variable cost (emergency)
- Holding cost
- Backorder cost per unit and time unit
- Lead time (normal)
- Lead time (emergency)

Constraints	• For the batch case transition probabilities for the Markov chain must be defined

5.1.10 Veeraraghavan and Scheller - Wolf (2008)

Veeraraghavan's and Scheller-Wolf's (2008) model is similar in its assumptions to the ones in Whittemore & Saunders (1977). They construct a heuristic where the key idea is that the decision on the amount to order at the emergency supplier should not be based on information about orders that will arrive after this order. They allow orders to overshoot and undershoot the inventory position and employ simulation to obtain the distribution of these over- and undershoots. The authors intend their model to be practical and therefore use very general conditions so that their model can be used in practice. They provide results that show that these results perform well for small problems. The main drawback is that their method for finding optimal policies becomes harder to obtain when the problem grows in size.

Review	Periodic review for normal orders Periodic review for emergency orders
Demand	In theory every discrete or discretized continuous demand distribution
Lead time (L_n/L_e)	Deterministic/Deterministic
Unsatisfied demand	Backordered
Policy/decision mechanism	Uses a dual index policy that makes use of emergency and regular order up to levels, an (S, T) policy. Lead times are constant.
Information needed	• Variable cost (normal) • Variable cost (emergency) • Holding cost • Backorder cost • Lead time (normal)

- Lead time (emergency)

Constraints
- The lead times are integer multiples of the review period

5.2 Classification of models

To make our choice of model easier we decided to classify them based on their approach to the dual sourcing problem. There was one clear divider that we found fundamental.

Minner (2003) concludes that the problem of finding optimal policies for dual sourcing systems grows quickly when batch size and lead times increase because an optimal policy would have to incorporate the exact timing and size of outstanding orders. The state vector therefore grows quickly in comparison to a single source (R, Q) system where it can be defined by the inventory position alone. Finding a good heuristic therefore requires some compromise with regards to batch or lead time assumptions.

This lead us to divide the articles between those that take the timing of outstanding orders into account and those that don´t. We refer to this as time dependency. To implement such a policy in practice requires tracking of orders in real time. This can be done with modern information technology. We refer to Song and Zipkin (2009, pp. 362-363) for a summary of tracking technologies. A second sub classification was then added based on the number of parameters needed for control of each policy.

5.2.1 Policies with time dependency

Class 1: Dual index periodic review

These policies track two inventory positions that are updated under periodic review.

- Veeraraghavan and Scheller-Wolf (2008)
- Arts (2009)
- Johansen and Thorstenson (2014)

Class 2: Dual index continuous review

These policies track two inventory positions that are updated under continuous review.

- Moinzadeh and Schmidt (1991)
- Song and Zipkin (2009)
- Johansen and Thorstenson (1998)

Class 3: Single index periodic review

A single reorder point under continuous review with periodic inspection for emergency orders.

- Axsäter (2014)

Class 4: Single index continuous review

A single reorder point under continuous review with continuous inspection for emergency orders.

- Axsäter (2007)

5.2.2 Policies built without time dependency

For policies covering normally distributed demand we did not find any satisfying models that consider the time dependency between orders. Thus we have to look at approximately optimal approaches.

Class 5: Single index dual reorder point

The policies only track the inventory level / inventory position and normal and emergency orders are placed when the inventory level / inventory position hits the respective reorder point.

- Moinzadeh and Nahmias (1988)

Class 6: Single index single reorder point

Works like the lost sales policy described earlier in section 3.6.2 that tracks one inventory position and uses a single reorder point.

- Lost sales (Axsäter, 2006) with modification

6 Choosing models to evaluate

We have divided our recommendation into two categories, one for larger problems using normal demand that disregards the time dependency of orders for finding an optimal policy, and one for smaller problems using one for one ordering that takes the time dependency into account. To cover these aspects, we have chosen three models.

6.1 Factors influencing choice

Referring again to the purpose of this thesis, *to investigate the challenges of introducing emergency ordering policies in GIM and try to identify and evaluate such policies,* we set the following goal for our choice. The policy should be feasible to implement in GIM given some modification and should cover the largest possible set of demand types for Syncron. Based on informal meetings with Syncron employees we have identified the following minimum requirements.

Minimum Requirements

1. It must be possible to optimize the policies after a target service level.
2. It must be possible to calculate the average inventory level a policy will result in.
3. At least one policy must be able to handle normally distributed demand. This due to the fact that this is a common assumption in practice.

Set of items

The policy should cover as many items as possible in GIM in order to be relevant. Items are classified based on the demand type that GIM has assigned to them. The division of demand types in our data sample is presented in section 7.5 but we will first present a brief overview. From Figure 6 on page 59 we see that slow items constitute 72 % of all items, followed by lumpy and fast items with 21 % and 5 % respectively. If we look at the distribution of items based on the total value for each class, Figure 8 page 59, it is remarkably similar. A different picture emerges when we look at the total demand for each class, see Figure 7 page 59. Here the largest category is fast followed by lumpy and slow, the reverse order compared to the previous figures. This is explained by the higher average demand for fast items.

Implementation

The implementation aspect is represented by how different the policy is compared to the current version of GIM and hence the effort needed to implement it. Is it possible to make alterations so that the policy can work with GIM? Or are the assumptions underlying it too restrictive to make it possible to use in practice?

6.2 Choice of policies to evaluate

Based on the set of items we would ideally like to cover all three categories slow, lumpy and fast. At a minimum we decided to cover slow items since they represented both a majority of the total number and value. We also wanted our policies to handle continuous review since this was the most common option used in GIM. This required us to choose one article under continuous review that can handle Poisson demand. This narrowed us down to Moinzadeh and Schmidt (1991), Song and Zipkin (2009), Johansen and Thorstenson (1998) and Axsäter (2007). All of these do take time dependency into account, an indication of that they would perform better than the current heuristic. The two first of these are very similar with Song and Zipkin (2009) building on and expanding the results of the first. We therefore eliminated the Moinzadeh and Schmidt (1991). Axsäter (2007) requires complex computations to be made every time emergency ordering is evaluated. This was of some concern to us since it would be harder to create a simulation model for such a policy. It would also require more processing capacity in GIM, something that drives cost for Syncron. Compared to Song and Zipkin (2009) that only sets two control parameters it seemed unnecessarily complex. We discarded it based on our implementation factor. Finally Song and Zipkin (2009) had a closed form expression for the distribution of the inventory level that was easier to evaluate than Johansen and Thorstenson (1998). The later also had a constraint on the number of outstanding orders. Therefore, we chose Song and Zipkin (2009) to be our policy for slow items.

We also wanted a policy that could handle fast items due to our third minimum requirement. This section contained fewer policies and none with time dependency. The two approximate policies we could choose from where Moinzadeh and Nahmias (1988) and the lost sales model (Axsäter, 2006). Of these two we found the former to be the most promising. It can be thought of as a double (R, Q) policy with one set of parameters for normal and one for the emergency orders. The lost sales model was attractive due to its simplicity but had more constraints attached. Unfortunately, Moinzadeh and Nahmias (1988) provides no simple algorithm for how to optimize the four policy parameters in their model. They employ simulation on a small set of combinations to find the best values. This would not be possible in GIM across thousands of items so due to this we discarded the policy based on our implementation factor. This left us with the lost sales model, which was chosen for fast items. One could argue that this model violates the first minimum requirement, but since customers are immediately satisfied with an emergency order in case of a stock-out, we have chosen to ignore this violation.

When we examined the policy closer we found that its constraints would be too restrictive in many cases. Due to this we also decided to develop a modified version of Syncron's current heuristic. This policy would be used for items with normally distributed demand and contain less restrictions.

6.3 Models to evaluate for Syncron

Both the model by Song and Zipkin (2009) and the lost sales model will be evaluated with real data from a large Swedish company. Both will be explained in greater detail in this section. When modifying the heuristic we came up with two versions that both took the costs of emergency orders into account. Since we were unsure about which one would perform better it was decided to evaluate both. They will be described in the following.

6.4 Song and Zipkin (2009)

Consider an inventory system in continuous time, with Poisson demand, linear ordering costs and two replenishment sources. The replenishment lead times are deterministic. Complete information regarding the time remaining for outstanding orders is known. The system information is summarized in two inventory positions, one for each source. By setting two policy variables the normal inventory position is kept constant while the emergency inventory position is kept above a fixed level. We can view the system as a network of queues with two nodes and with a state dependent routing mechanism. One of the nodes has a limit on the occupancy, and is by-passed if the limit is exceeded. (Song and Zipkin, 2009)

6.4.1 Denotation

λ	Demand rate (Poisson)
T_1'	Normal order lead time
T_2'	Emergency order lead time
T_1	$T_1' - T_2'$
T_2	T_2'
N	All outstanding orders
N_2	All outstanding orders that will arrive within T_2'
N_1	$N - N_2$
IN	Net inventory = on-hand inventory – backorders
s_1	Policy parameter 1
s_2	Policy parameter 2, $s_2 \leq s_1$
u_1	$s_1 - s_2$
IP_1	$IN + N = IN + N_2 + N_1$ = net inventory + outstanding orders N
IP_2	$IN + N_2$ = net inventory + all outstanding orders arriving before the emergency lead time.
c_1	Unit purchasing cost from the normal supplier
c_2	Unit purchasing cost from the emergency supplier, $c_1 < c_2$
h	Holding cost per unit and time unit
b	Backorder cost per unit and time unit

6.4.2 Policy

The policy triggers orders so that IP_1 is kept at s_1 and IP_2 at or over s_2. If we start at inventory position IP_1 at time 0, an order will be triggered every time a unit is demanded. In case IP_2 is or becomes strictly lower than s_2 this triggers an emergency order, otherways a normal order will be triggered. The result is that emergency orders are triggered when too much pipeline stock is too far away (Song and Zipkin, 2009).

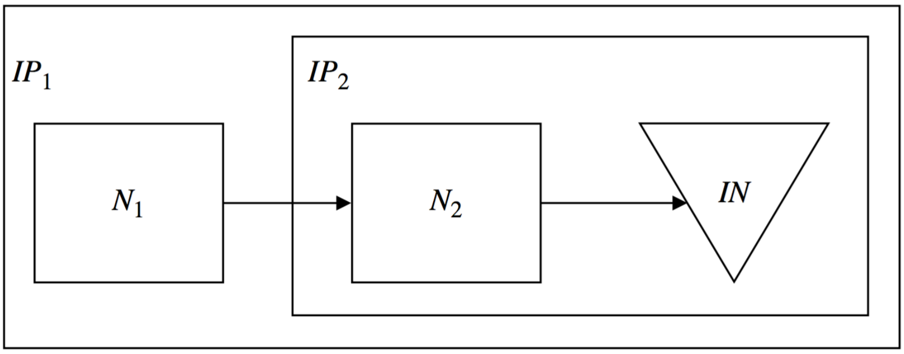


Figure 4: Visualization of relations between inventory position, net inventory and outstanding orders (Song and Zipkin, 2009)

We can describe the system according to Figure 5, where we have three nodes, 0, 1 and 2. Node 0 is the order generating node, node 1 encompasses all outstanding orders that will not arrive within the emergency lead time, and node 2 encompasses all outstanding orders that will arrive within the emergency lead time. Note that for pure Poisson demand the unit generating rate in node 0 will equal the arrival rate of customers to the system. In case $IP_2 \geq s_2$ or equivalently $N_1 < u_1$ we will not place an emergency order and the units will be routed to node 1. In case we place an emergency order the units are routed directly to node 2, by-passing node 1. This can also be interpreted as a capacity limit on node 1. (Song and Zipkin, 2009)

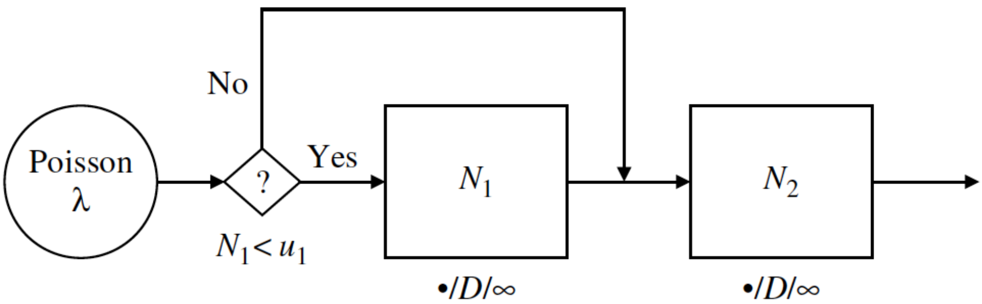


Figure 5: Visualization of the system (Song and Zipkin, 2009)

The net inventory in this model can be expressed as:

$$IN = s_1 - N. \tag{6.1}$$

This means that if we can derive the distribution for N, we can determine the key performance measures, e.g. average inventory, average backorders and probability of stock out. (Song and Zipkin, 2009)

In order to compute the distribution of N Song and Zipkin (2009) first compute the joint distribution of (N_1, N_2). We define the equilibrium probabilities as

$$p(n_1, n_2) = P(N_1 = n_2, N_2 = n_2), \tag{6.2}$$

the Poisson probabilities as

$$\phi_k(n) = \frac{(\lambda T_k)^n}{n!} e^{-\lambda T_k}, \qquad k \in \{1, 2\} \tag{6.3}$$

and the set of feasible states *($IP_2 \geq s_2$)* as

$$\mathcal{N}(u_1) = \{(n_1, n_2) : n_1 \leq u_1\}. \tag{6.4}$$

We can then write $p(n_1, n_2)$ on the product form

$$p(n_1, n_2) = \frac{1}{G(u_1)} \phi_1(n_1) \phi_2(n_2), \qquad (n_1, n_2) \in \mathcal{N}(u_1), \tag{6.5}$$

where

$$G(u_1) = \sum_{\mathcal{N}(u_1)} \phi_1(n_1) \phi_2(n_2) = \sum_{n_1 \leq u_1} \phi_1(n_1) \tag{6.6}$$

is a normalizing constant. $\phi_1(n_1)\phi_2(n_2)$ can be interpreted as the probability that n_1 customers arrive during a time period (normal lead time – emergency lead time) times the probability that n_2 customers arrive during an emergency lead time, where we restict to the cases when $IP_2 \geq s_2$. Song and Zipkin (2009) further define

$$p(n) = P(N = n) \tag{6.7}$$

and

$$\phi(n) = \frac{(\lambda T_1)^n}{n!} e^{-\lambda T_1}. \tag{6.8}$$

Now the distribution of N can be formulated as

$$p(n) = \sum_{m=0}^{n} p(m, n-m) = \frac{1}{G(u_1)} \begin{cases} \phi(n), & n \le u_1 \\ \sum_{m=0}^{u_1} \phi_1(m)\phi_2(n-m), & n > u_1 \end{cases} = \tag{6.9}$$

$$= \frac{1}{G(u_1)} \begin{cases} \phi(n), & n \le u_1 \\ \phi(n) - \sum_{m=u_1+1}^{n} \phi_1(m)\phi_2(n-m), & n > u_1 \end{cases} \tag{6.10}$$

(Song and Zipkin, 2009).

These results can be found in Moinzadeh and Schmidt (1991) as well. Let us now define $\eta(u_1)$ to be the fraction of orders that are ordered from the emergency source. Thus we have

$$\eta(u_1) = P(N_1 = u_1) = \frac{\phi(u_1)}{G(u_1)}. \tag{6.11}$$

$\eta(u_1)$ decreases in u_1 and approaches one when u_1 goes to zero.

The total costs can now be expressed as

$$C(s_1, u_1) = c_1\lambda + (c_2 - c_1)\lambda\eta(u_1) + hE[(s_1 - N)^+] + bE[(N - s_1)^+] \tag{6.12}$$

where

$$(IL)^+ = (s_1 - N)^+ \tag{6.13}$$

and

$$(IL)^- = (N - s_1)^+. \tag{6.14}$$

C is sub-modular and convex in s_1 for any fix u_1 (shown by Moinzadeh and Schmidt (1991)). This means that "the best s_1 is non decreasing in u_1" (Song and Zipkin, 2009, p. 365) and that a local optimum will also be the global optimum. The optimum can be found by performing an outer search over u_1 and an inner search over s_1. In other words, in the outer loop we search over u_1 and in the inner loop we find the s_1 that for a given u_1 minimizes the costs.

6.5 (R, Q)-LS

We will now adapt the lost sales model described in section 3.6.2. In the following we will refer to it as (R, Q)-LS. A complete mathematical description has already been provided in that section. The following model translations are made:

- The penalty cost will be set to correspond to the extra cost of using an emergency order, i.e. $b = c_e - c_n$.
- Since most stock outs will occur at the end of the order cycle, there is a possibility that a normal order will arrive before the placed emergency order. This implies that the emergency lead time must be small enough so the probability for this is reduced.
- The emergency orders will not be included in the normal inventory position since they are already "lost" i.e. dedicated to a customer.

6.6 Add-on emergency ordering heuristics

In this section we will present two emergency ordering heuristics with the intention to improve Syncron's (R, Q)-RH, presented in section 4.2.1. A requirement is that they must make use of the single supplier (R, Q) policy for normal orders and to supplement this with an independent decision mechanism for emergency orders. The main idea, of both these heuristics, will be to compare the extra direct and indirect costs of placing an emergency order with the improved service level that this will achieve. The two policies will be derived in the following sections.

6.6.1 Notation

We start with introducing necessary notation.

t_0	Point in time at which the emergency ordering option is evaluated.
t_e	Point in time at which an emergency order will arrive if placed at time t_0.
t_n	Point in time when the next normal order (that arrives after t_e) arrives.
q_n	Normal order quantity (fix).
q_e	Emergency order quantity (variable).
c_n	Unit cost if using the normal mode.
c_e	Unit cost if using the emergency mode.
h	Holding cost rate, per unit and time unit.
μ	Expected demand per time unit.
σ	Standard deviation of demand during time unit.
A	Normal ordering cost.
a	Emergency ordering cost.
T	Review period.
L	Limit set by the user (cost/service level).
SP	Maximum stock out probability.
$C(q_e)$	Extra cost of ordering q_e units.
$\Delta SL(q_e)$	Improved service level by ordering q_e units.

In contrary to Axsäter (2007 and 2014) we will not consider an infinite time horizon, but will limit the focus to a single period, defined as the time until the next normal order arrives. The improved service level is defined as the increased probability of no stock out at the end of this period.

6.6.2 (R, Q)-SH – Short time horizon heuristic

In this first heuristic, that we hereafter will refer to as (R, Q)-SH, we look at the probability of stock out during the time period $[t_e, t_e + 1]$ or in a more general case $[t_e, t_e + T]$ (the time unit here is days and 1 is chosen because GIM is updated on a daily basis). How much must we order to bring the stock out probability down to an acceptable level? To simplify let's say: What is the probability of stock out at time $t_e + 1$? If this is above the maximum stock out probability SP, set by the user, we order enough to reduce this probability to SP. If the stock out probability is less than SP we don't place an emergency order. Next day we make the same evaluation again. The reason that we look at the stock out probability at time $t_e + 1$ is that if we review our inventory on a daily basis, this is the first point in time when we can affect the stock out probability by placing an emergency order today. In a pure continuous review system however, we will instead evaluate the stock out probability at time t_e. This will of course be evaluated every time a customer places an order or when the forecast is updated.

Assumptions

Before we can express the heuristic mathematically we constrain the problem to the for Syncron most applicable context. Demand is thus expected to be normally distributed with mean μ and standard deviation σ. The inventory level at the time of evaluation we denote IL_0 and the corresponding inventory position we denote IP_0. The outstanding orders expected to arrive within t_e time units are included in M, $i \in M = \{1,2, \ldots, I\}$ and are indexed in order of their arrival. These have order quantity $q_i \in \{q_e,\ q_n\}$. We denote the inventory level at time $t_e + 1$ as IL'.

Mathematical model

The probability of backorders at time $t_e + 1$ is

$$P(IL' < 0) = P\left(D(t_e + 1) > IL_0 + \sum_{i \in M} q_i\right). \tag{6.15}$$

The cumulative distribution function of the demand can, with the standardized normal distribution, be expressed as

$$P(D < x) = \Phi\left(\frac{x - \mu * (t_e + 1)}{\sqrt{t_e + 1} * \sigma}\right) = P(IL' > 0) \tag{6.16}$$

Consequently we get

$$P(IL' < 0) = 1 - \Phi\left(\frac{IL_0 + \sum_{i \in M} q_i - \mu * (t_e + 1)}{\sqrt{t_e + 1} * \sigma}\right). \tag{6.17}$$

Now let us express the emergency ordering quantity. If $P(IL' < 0)$ is bigger than SP, we place an emergency order such that

$$1 - \Phi\left(\frac{IL_0 + q_e + \sum_{i \in M} q_i - \mu * (t_e + 1)}{\sqrt{t_e + 1} * \sigma}\right) \leq SP \Rightarrow \tag{6.18}$$

$$\Rightarrow 1 \leq SP + \Phi\left(\frac{IL_0 + q_e + \sum_{i \in M} q_i - \mu * (t_e + 1)}{\sqrt{t_e + 1} * \sigma}\right) \Rightarrow \tag{6.19}$$

$$\Rightarrow 1 - SP \leq \Phi\left(\frac{IL_0 + q_e + \sum_{i \in M} q_i - \mu * (t_e + 1)}{\sqrt{t_e + 1} * \sigma}\right) \Rightarrow \tag{6.20}$$

$$\Rightarrow \Phi^{-1}(1 - SP) \leq \frac{IL_0 + q_e + \sum_{i \in M} q_i - \mu * (t_e + 1)}{\sqrt{t_e + 1} * \sigma} \Rightarrow \tag{6.21}$$

$$\Rightarrow \Phi^{-1}(1 - SP) * \sqrt{t_e + 1} * \sigma \leq IL_0 + q_e + \sum_{i \in M} q_i - \mu * (t_e + 1) \Rightarrow \tag{6.22}$$

$$\Rightarrow \Phi^{-1}(1 - SP) * \sqrt{t_e + 1} * \sigma - IL_0 + \mu * (t_e + 1) - \sum_{i \in M} q_i \leq q_e. \tag{6.23}$$

So we just have to find the smallest integer q_e such that

$$q_e \geq \Phi^{-1}(1 - SP) * \sqrt{t_e + 1} * \sigma - IL_0 + \mu * (t_e + 1) - \sum_{i \in M} q_i. \tag{6.24}$$

Note that $(1 - SP)$ would naturally correspond to the target service level. In fact, this policy is just a base stock policy, or order up to policy, where we use the emergency inventory position, defined as the inventory level plus all outstanding orders arriving within the emergency lead time, as the basis for our calculation. Removing q_e we can restructure 6.24 so that it tells us when to order. Thus, if the below relation is fulfilled we place an order.

$$IL_0 + \sum_{i \in M} q_i \leq \Phi^{-1}(1 - SP) * \sqrt{t_e + 1} * \sigma + \mu * (t_e + 1) \tag{6.25}$$

If we use the emergency inventory position definition from above,

$$IP_e = IL_0 + \sum_{i \in M} q_i, \tag{6.26}$$

and the usual $\sigma' = \sqrt{t_e + 1} * \sigma$ and $\mu' = \mu * (t_e + 1)$ we can rewrite 6.25 as

$$IP_e \leq \Phi^{-1}(TSL) * \sigma' + \mu'. \tag{6.27}$$

The ordered quantity should of course fulfill

$$IP_e + q_e \geq \Phi^{-1}(TSL) * \sigma' + \mu' \tag{6.28}$$

And therefore the order up to level would be the right hand side of 6.28.

Including Cost

In case we have an overage inventory holding cost and an underage backorder cost this emergency heuristic would just be the classical newsvendor problem. Syncron does not work with backorder cost so we need another solution. In this heuristic we will weigh the cost of emergency ordering against the improved service. The user is asked to set a limit, L, for how much every unit of service improvement is allowed to cost. We know of course that as we approach a service level of 100 % the marginal cost for improving the service level increases. Thus we start with the smallest q_e that satisfies 6.28, and decrease one integer step at a time until $C(q_e)/\Delta SL(q_e)$ is smaller than the aforementioned limit. In case we reach $q_e = 0$, we stop and do not place an order. Thus we have $q_e + 1$ different possibilities, to place an order of size 0, 1, 2,..., q_e. We order the highest quantity that satisfy 6.30 below.

The emergency ordering cost, $C(q_e)$, is the sum of the extra direct cost of placing the emergency order and the indirect cost of holding more stock. We assume that the emergency order inflicts extra stock corresponding to the order quantity, from the time of arrival until the next normal order arrives. This can of course be computed with more detail for more accurate results, but this assumption is made to keep calculations relatively simple. In case no normal order is outstanding, the next normal order will be assumed to arrive in one normal lead time. The cost of placing an emergency order of size q_e can thus be expressed as

$$C(q_e) = a - \frac{q_e * A}{q_n} + (c_e - c_n) * q_e + q_e * h * (t_n - t_e) \tag{6.29}$$

and our solution should satisfy

$$\frac{\Delta SL(q_e)}{C(q_e)} = \frac{\Phi\left(\frac{IP_e + q_e - \mu'}{\sigma'}\right) - \Phi\left(\frac{IP_e - \mu'}{\sigma'}\right)}{a - \frac{q_e * A}{q_n} + (c_e - c_n) * q_e + q_e * h * (t_n - t_e)} \geq L. \tag{6.30}$$

In 6.30 the numerator is the expected improved service level at time $t_e + 1$ by placing en emergency order of size q_e right now, and the denominator is the assumed extra cost of placing the order. L is the limit set by the user, of how much each unit of improved service level is allowed to cost.

6.6.3 (R, Q)-LH – Long time horizon heuristic

The difference between this second heuristic, that we hereafter refer to as (R, Q)-LH, and (R, Q)-SH is at what future point in time we investigate the stock out probability. In (R, Q)-SH, we only looked one emergency lead time into the future. In (R, Q)-LH, we consider the stock out probability when the next normal order arrives. The notations and the mathematical derivation is analogous to the ones for (R, Q)-SH. We will only give the final results. We want to find the smallest q_e that satisfies below relation. Note again that $1 - SP$ can be interpreted as the target service level.

$$q_e \geq \Phi^{-1}(1 - SP) * \sqrt{t_n} * \sigma - IL_0 + \mu * t_n - \sum_{i \in M} q_i \tag{6.31}$$

By rewriting 6.31 as for (R, Q)-SH we get our decision rule. Place an emergency order if

$$IP_e \leq \Phi^{-1}(TSL) * \sigma' + \mu' \tag{6.32}$$

such that

$$IP_e + q_e \geq \Phi^{-1}(TSL) * \sigma' + \mu'. \tag{6.33}$$

Of course $\sigma' = \sqrt{t_n} * \sigma$ and $\mu' = \mu * t_n$. If we want to include the cost evaluation into the decision rule we do this exactly the same way as for (R, Q)-SH. If the ratio

$$\frac{\Phi\left(\frac{IP_e + q_e - \mu'}{\sigma'}\right) - \Phi\left(\frac{IP_e - \mu'}{\sigma'}\right)}{a - \frac{q_e * A}{q_n} + (c_e - c_n) * q_e + q_e * h * (t_n - t_e)} \tag{6.34}$$

is smaller than the given limit L, we place an emergency order of size q_e. Thus we start with the smallest quantity that satisfies 6.33 and decrease it with integer steps until 6.34 is smaller than our given value.

Both heuristics are simple order up to policies that take cost into consideration by comparing the improved service level to the cost of placing the emergency order. A natural further improvement of these two models, that would fit well with the multi-item TSL-decision algorithm that Syncron uses, see chapter 4.1.5, would be to compare expected extra lines or units covered by an emergency order to the extra emergency costs. This way one could make sure that improvement in service level over a group of items is performed where it is least costly.

However, if we want to control a single, perhaps very important item, a warehouse manager would be able to set a maximum cost per improved SL-unit and the two proposed heuristics would automatically handle the situation. (R, Q)-SH is likely to be more sensitive to variations in the lead time but has the advantage of postponing the decision as long as possible. (R, Q)-LH will most likely result in a higher service level but to a higher cost.

7 Data analysis

This chapter describes our selection of items for the simulation study from the complete data sample obtained from Syncron's database. A sample of 7 items from a total of 511 have been selected.

7.1 Data description

In order to make an assumption about the demand patterns that occur at our case company's dealer network, historical transaction data has been extracted.

The data comes from two retailers, A and B, located in Sweden in close proximity of each other. Both are supplied from the same central warehouse in the USA through either maritime or air freight. Their items therefore have the same lead times. The normal lead times were in all cases 42 days and the emergency lead times were 14 days. Of these 14 days, only 2 are needed for the airfreight, thus 2 days naturally makes up the lower bound of the emergency lead time. The information collected for each item is illustrated in Table 1 and Table 2.

Table 1: Item data

Column header	*Format*
Item Code	XXXXXXXX
Reorder Point	Integer number
Order Quantity	Integer number
Target Service level	$0 \leq x \leq 1$
Stocked	Yes/No
Delivery Mode	Normal/Emergency
Lead time	Days
Weight	Grams
Unit Cost	SEK
Demand Type	Fast/Erratic/Lumpy/Slow

Table 2: Transaction data

Column header	*Format*
Item Code	XXXXXXXX
Request Date	YYYY-MM-DD
Order Quantity	Integer number
Sales order Number	XXXXXX

The *Request Date* format provided to us did not include the exact time of the request, only its date. This made it impossible to study the time between arrivals of customer orders. Extend, our simulation software, sets customer arrivals based on a distribution for time between arrivals. This information had to be deduced by other means, for example by fitting an empirical compounding Poisson distribution.

7.2 Data selection

In order to narrow down our original sample of 511 articles the following criteria were selected;

- **Only demand registered during the time period 10 November 2013 to 10 November 2015 was examined**. Both retailers had only been operational since early 2013. When looking at the data on a yearly basis it became clear that the first part of 2013 was a trial phase with demand two thirds lower than either 2014 or 2015. Because of this it was disregarded.
- **Only items currently held in stock were considered**. The non stocked items had very low yearly demand and were requested once or twice per year. Even if these items would have been stocked, the possible inventory reductions would have been small.
- **Items held at retailer B were not considered.** At closer examination, 95% of all items were registered at retailer A and an even higher percentage of the transactions came from the same retailer. All of the stocked items at retailer B had very low yearly demand. Due to this retailer B was of little relevance and removed from our sample.

This selection limited us to 258 items. Since the setting of policy parameters in GIM depends on the demand class of the item we decided to pick two items from each of the two demand classes fast and erratic and three from class slow. The reasoning behind this was threefold.

First, since each simulation run would be time consuming we wanted to limit them to a small sample of seven items. Second, by picking one item with low and one with high total demand within the classes fast and erratic, the outcome from our simulations would indicate the results for the items in between.

Third, when picking items from the slow demand class we limited the set of items to items that were strictly ordered in quantities of one, i.e. pure Poisson demand. This narrowed down our sample to 112 out of 179. From these items two items with high demand were chosen since the larger part of the sample consisted of items that had been ordered only once. The final third item was picked from the 67 items that did not have pure Poisson demand. Its purpose would be to illustrate what the outcome would be if the assumption of pure Poisson demand is false. Thus we had our final seven items.

Note that no lumpy items were tested in our simulation study. The reason behind this is that we have not presented a specific emergency ordering policy for such items.

7.3 Distribution fitting

Distribution fitting was performed to examine how closely the demand class set by GIM actually resembled the data sample. Our aim here was to model the data sample as closely as possible regardless of how GIM had classified it.

For each item, the transaction data was run through a module in Extend called StatFit. StatFit is a statistical software that gives suggestions on continuous and discrete distributions that describe a data set and gives corresponding probabilities based on statistical tests. Recall here that demand data is aggregated on a daily basis. None of our seven items had a good fit. The same analysis was performed on a data sample of order quantity for all orders to check for a distribution for order size. Here two out of out seven items had a fit and both of them for a geometric distribution. Unfortunately this information could not be used since we could not study time between arrivals.

The discrete event simulation model, built in Extend, that we intended to use needed two stochastic variables as input. One to control time between customer arrivals and another for order size. Since we did not have a good fit for the demand size of all of our items and no information regarding time between arrivals it was decided to use an empirical distribution for the demand sizes of our items. Hence, our simulations used compounding Poisson distributions where the compounding distribution was empirical for the order size. Items with pure Poisson demand thus had a 100% probability of order size one. The customer arrival intensity was then fitted to the average order size so that the mean demand per day remained the same in both sample and fitted distributions.

One problem with this approach was that there would be a difference in standard deviation between our sample data and the fitted empirical distributions on the days where two different demand transactions had occurred in our sample. This was accepted.

A short summary of the picked items is presented in Table 3. More details about the items are presented in Appendix II.

Table 3: Items and demand distributions. The difference in standard deviation of our fitted distributions compared to the sample data is shown in the last column Std Diff.

Item	*Demand Class*	*Fitted Distribution*	*Std Diff*
1	Erratic	Empirical	-4%
2	Erratic	Empirical	-15%
3	Fast	Empirical	-37%
4	Fast	Empirical	-22%
5	Slow	Empirical	-13%
6	Slow	Empirical	0%
7	Slow	Empirical	-21%

7.4 Additional data and assumptions

We did not get a holding cost from our case company. Instead, we used a holding cost rate of 25% which is a common assumption in literature, see Alford et al. (1948) for an example. Setting a correct holding cost rate is notoriously difficult (Berling 2005). so we note that this probably contains a large margin of error. The holding cost is computed as

$$h = r * c_n \tag{7.1}$$

where r denotes the holding cost rate and c_n denotes the normal unit cost.

The air freights used for computing our emergency unit costs were extracted from a freight rate table that was obtained from the case company. The freight rates were given as a variable cost per weight unit, but orders had to be of at least a minimum value. The minimum order value requirement was neglected since it was assumed that this would always be fulfilled by combining emergency orders for many items. The emergency unit costs, c_e, were ultimately computed by adding the air freight unit rate, c_{af}, to the normal unit cost provided by the case company:

$$c_e = c_n + c_{af}. \tag{7.2}$$

The currency used in our simulation study is SEK and all results presented in this thesis, that include costs, are presented in SEK. The freight rates were given in USD and were therefore converted to SEK, at the 2015-12-09 rate of 8,48 SEK/USD.

7.5 Distribution of demand types in sample

The selected sample of 258 items had been divided into different demand classes by GIM. In Figure 6, Figure 7 and Figure 8 we see how items, demand and value are distributed over the different demand classes. This information was used in chapter 6.1 to show what demand types in GIM that proposed ordering policies must be compatible with. Each demand type is based on an assumption about the statistical distribution of demand. Any proposed ordering policy should therefore cover as large a fraction of items, demand and value as possible.

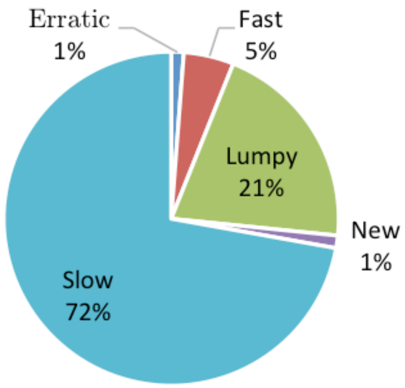


Figure 6: All items by demand type

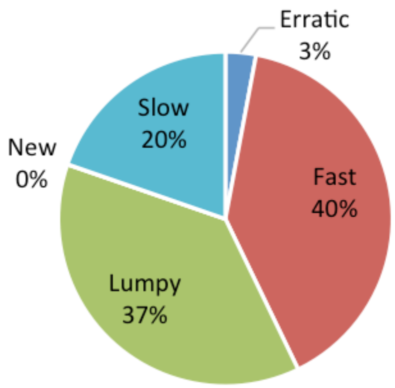


Figure 7: Total item demand by demand type

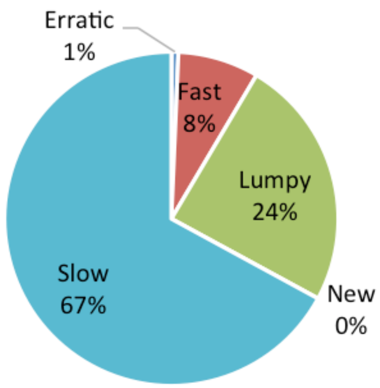


Figure 8: Total value of demand by demand type

8 Simulations

This section describes the analytical models and Extend software models that were used during the simulation. A brief overview of the models will be provided together a validity study and tables covering input parameters.

8.1 Analytical models for determining policy parameters

Before being able to test the models identified in the previous chapters with simulation on real data, we had to build analytical models for determining the policy parameters. These were all written in visual basic, using excel spread sheets for inputs and outputs. The models programming code will not be presented in detail but is available on request.

8.1.1 (R, Q) based models

Four of our policies, (R, Q), (R, Q)-RH, (R, Q)-SH and (R, Q)-LH, used the same policy parameters, R and Q. Given holding costs and ordering costs the order quantity Q can be determined analytically, e.g. with the EOQ formula (section 3.2.1), but since we did not have these costs we chose to use the order quantities provided by our case company.

The reorder point R was then determined by increasing R until the service level was above a target service level. Here it is important to mention that we for slow items, used the S1 service level, since this is the way Syncron handles these items. For more details see section 4.1.

For the fast and erratic items we did actually make some corrections to the way Syncron computes the fill rate. As explained in section 4.1.3 Syncron seems to disregard the term $G(K + \frac{Q}{\sigma'})$. Since this is not always a good approximation we chose to include this term.

8.1.2 (R, Q)-LS

For the lost sales model we started by using the Q provided by the case company, and performed a linear search over R to find the R that minimized the costs. See chapter 3 for details. We had troubles however, to find items where the resulting R was smaller than Q, which is an important constraint (to assure high precision) of the lost sales model. We therefore had to increase Q step by step and perform new optimizations of R, until we found an R that satisfied the condition. Obviously much of the real data aspect of the research was not valid anymore, but this gave us the opportunity to at least test the policy.

8.1.3 S&Z

The final excel-model was built for slow items only, assuming a pure Poisson demand. As described in detail in section 6.4 the parameters s_1 and u_1 were determined with an inner and an outer loop, in order to minimize the costs given that we know the backorder costs.

8.2 Simulation models for testing the chosen policies

Before we present the simulation models we define and denote the input parameters that are used in the chosen policies, see Table 4. Note that all ordering costs A are set to 0 in our simulations. The base case values for item 1 to 7 can be found in Appendix II.

Table 4: Input parameters needed for simulation runs

Parameter	Description
c_n	Cost per unit for the for the normal replenishments
c_e	Cost per unit for the for the emergency replenishments
A_n	Ordering cost for the for the normal replenishments.
A_e	Ordering cost for the for the emergency replenishments
h	Holding cost per unit and day
b	Backorder cost per unit and day. Only used for slow items.
R	Reorder point. Not used for S&Z
Q	Batch quantity for the normal replenishments. Not used for S&Z.
S_1	Policy parameter 1 for S&Z, order up to level for IP_1
S_2	Policy parameter 2 for S&Z, order up to level for IP_2
Int. Time	Time between customer arrivals
Demand/Day	Mean daily demand. Only for (R, Q)-RH, (R, Q)-SH and (R, Q)-LH.
Std. Dev.	Standard Deviation of daily demand. Only for (R, Q)-SH and (R, Q)-LH
L_n	Lead time for normal replenishments
L_e	Lead time for emergency replenishments
Empirical distribution table	Empirical probabilities for order size
Simulation time	Length in days of the simulation
ROL	Run out level for (R, Q)-RH
Search Start	The upper bound for the service level improvement. Only for (R, Q)-SH and (R, Q)-LH
Ratio	Maximum cost for service level per unit improved fill rate. Only for (R, Q)-SH and (R, Q)-LH.

All in all, we wanted to model six different ordering policies. To build representations of the different ordering processes we chose the simulation software Extend. For each of the six models we will below briefly explain the final simulation models. See more details in the Appendix V and VI. The complete models are also available upon request from the authors.

In Figure 9 the general form of the demand generation for all our models is depicted. The *Create* block to the left releases customers according to a Poisson process (many different processes can be chosen here) with exponentially distributed interarrival times. When the customer arrives at the *Set* block to the right, a random number is generated in the *Random Number* block according to an empirical distribution. This number is the order quantity.

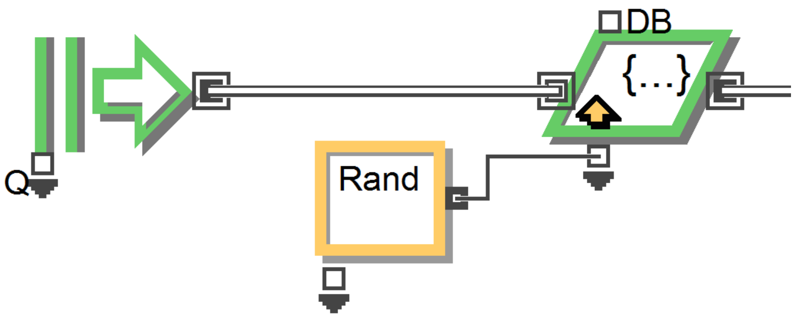


Figure 9: Demand generator

8.2.1 (R, Q)

The simulation model for the single supplier (R, Q) policy is shown in Figure 10. This is the base for the three emergency ordering policies that build on the (R, Q) policy. There are separate flows for sales- and customer orders, and backorders and stock on hand serve as buffers on either side of the flow. Demand is generated as explained above. Purchase orders of a fix quantity Q are generated as soon as we receive orders that bring the inventory position to or below R, that is on a continuous basis. Not depicted in Figure 10 are all computations that give us the results as well as other supporting computations for determining when to place orders. These are shown in the Appendix V.

8.2.2 (R, Q)-RH

The second simulation model is the one for Syncron's existing solution for emergency orders, (R, Q)-RH, see Figure 11. The model is similar to the (R, Q) model but the purchase order flow has been extended to include a separate flow for emergency orders. Normal orders are generated the same way as for the (R, Q) model, but they are now tracked through a customized block that can be seen in Appendix V. This block takes care of the emergency order generation, emulating the rush ordering heuristic in GIM. The code of the customized block is shown in the Appendix VI. Decisions to place normal or emergency orders are taken each time demand occurs, thus on a continuous basis.

8.2.3 S&Z

The Song and Zipkin simulation model is illustrated in Figure 12. This model does not have the (R, Q) model as a base and is thus built differently.

We still have the sales order flow at the top of the model and purchase order flow at the bottom, but the mechanism for triggering purchase orders differs from the other models. We have tried to emulate the mechanism by Song and Zipkin, visualized in Figure 5 in section 6.4. This corresponds to the bottom left part of our model. Once again, the supporting computations are not shown in Figure 12, but in Appendix V.

8.2.4 (R, Q)-LS

Figure 13 shows the Lost Sales simulation model. This model works according to the same principles as the (R, Q) model but is built separately. The top flow carries customers and the bottom flow generates orders. The difference lies in that the lost sales model has a counter for the inventory position and when it reaches zero new customers are diverted to the emergency source. Note that according to 3.6.2 there can be no backorders in this model since demand during stock out is satisfied immediately from the emergency source. We do however measure backorders and emergency source utilization in the models since in reality customer will have to wait for deliveries from the fast source too.

8.2.5 (R, Q)-SH

The simulation model for (R, Q)-SH is shown in Figure 14. It is similar to the model for (R, Q)-RH, but the customized block that handles the creation of emergency orders now works like described in section 6.6.2 instead. See further details in Appendix V and VI.

8.2.6 (R, Q)-LH

The last simulation model is depicted in Figure 15. The model has basically no visual differences from the model for (R, Q)-SH. The only difference is in the customized block, where we now consider a longer time horizon. See further details in Appendix V and VI.

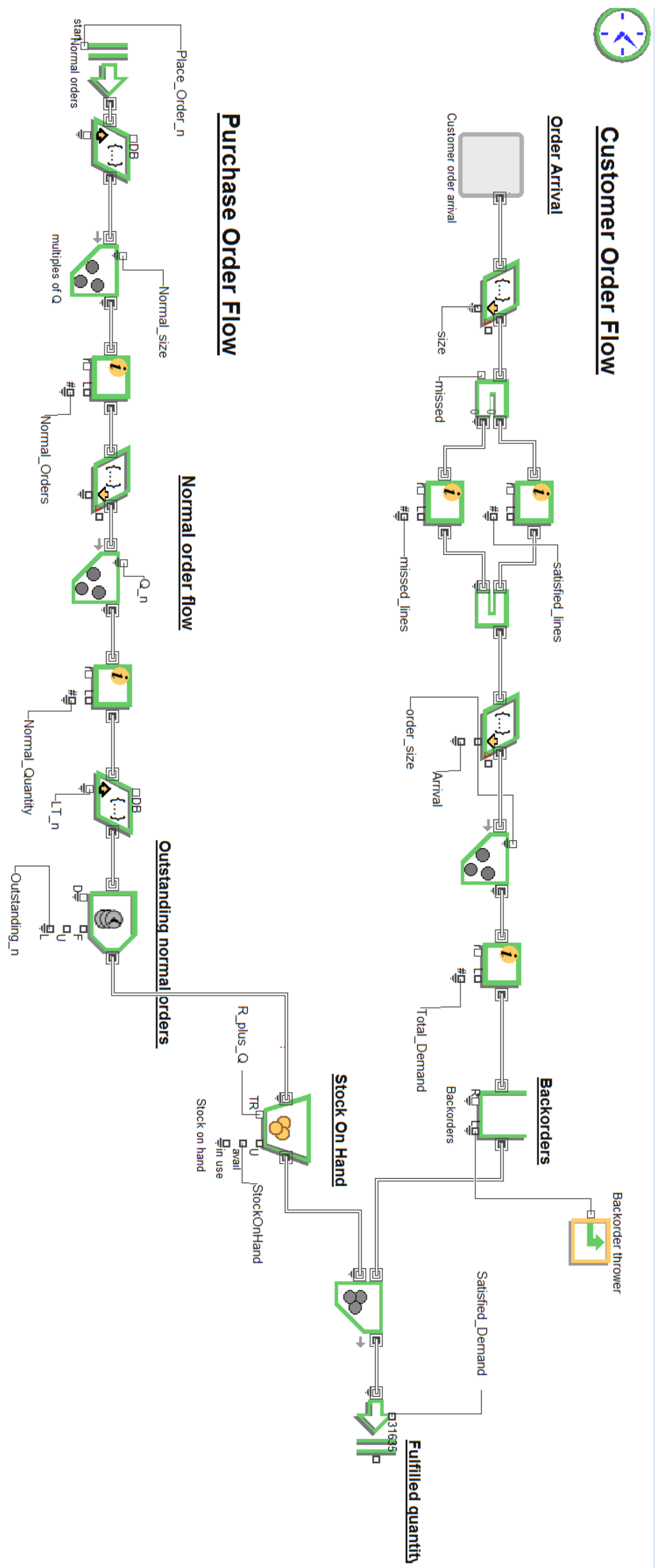


Figure 10: (R, Q) simulation model

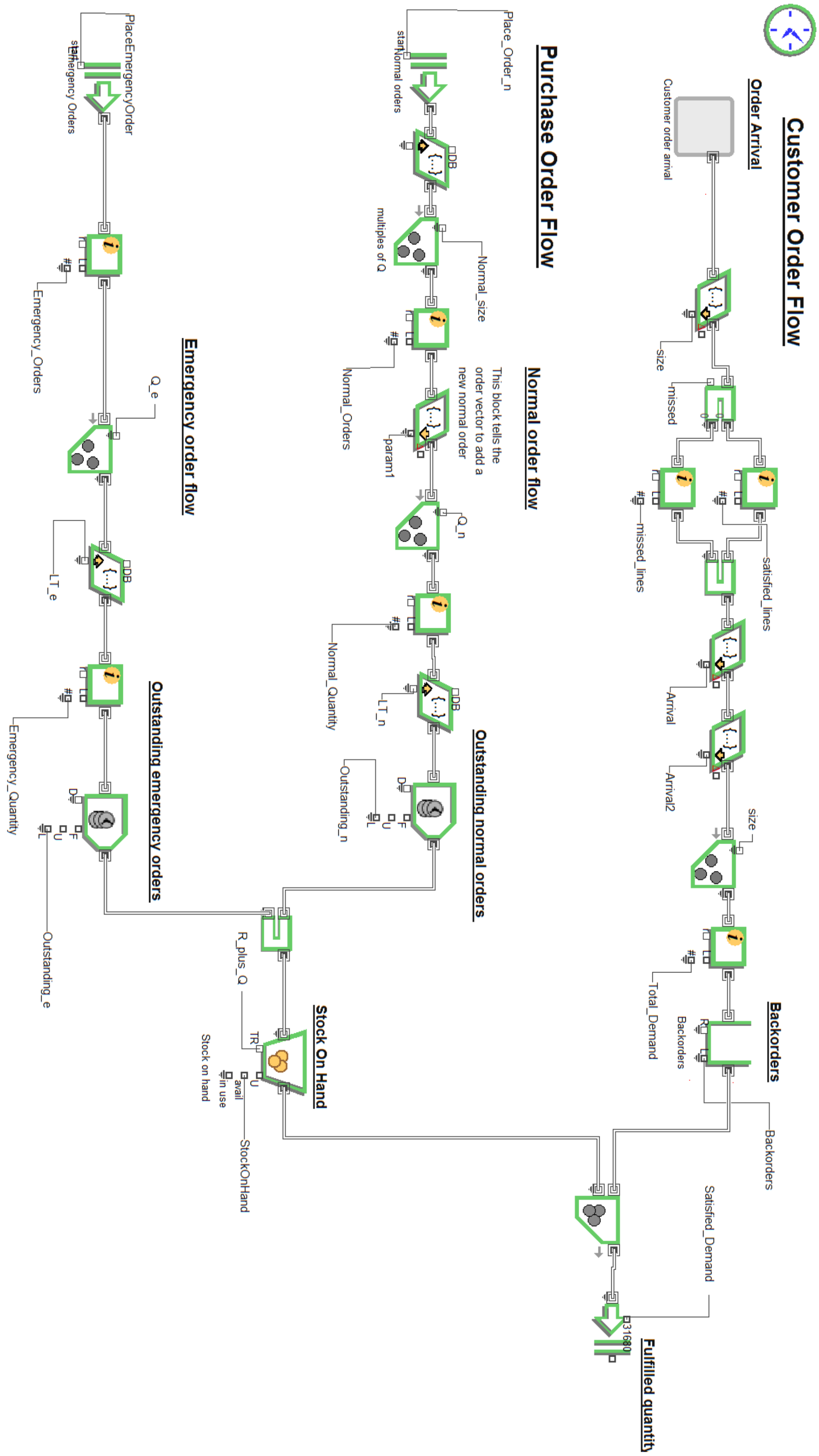


Figure 11: (R, Q)-RH simulation model

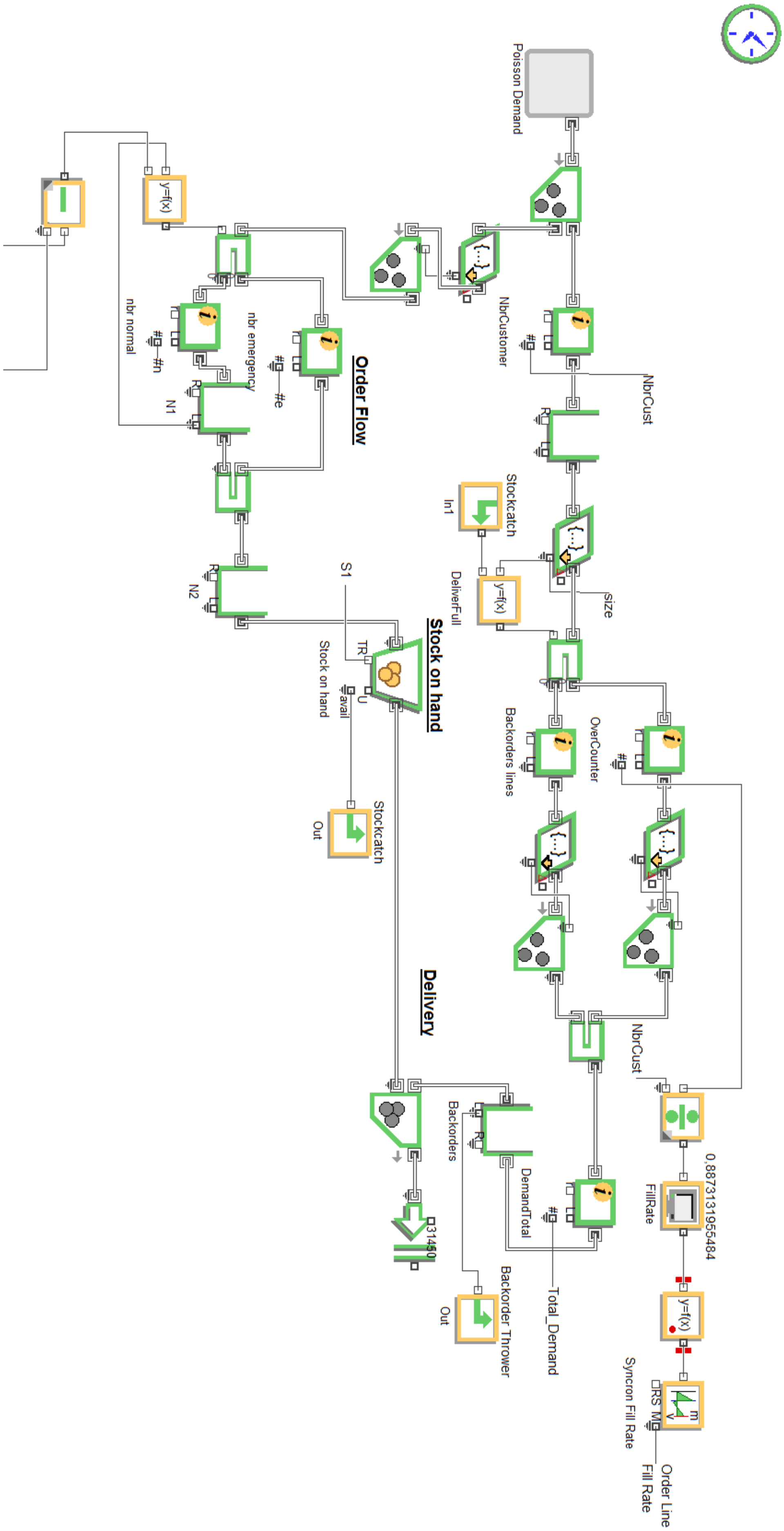


Figure 12: S&Z simulation model

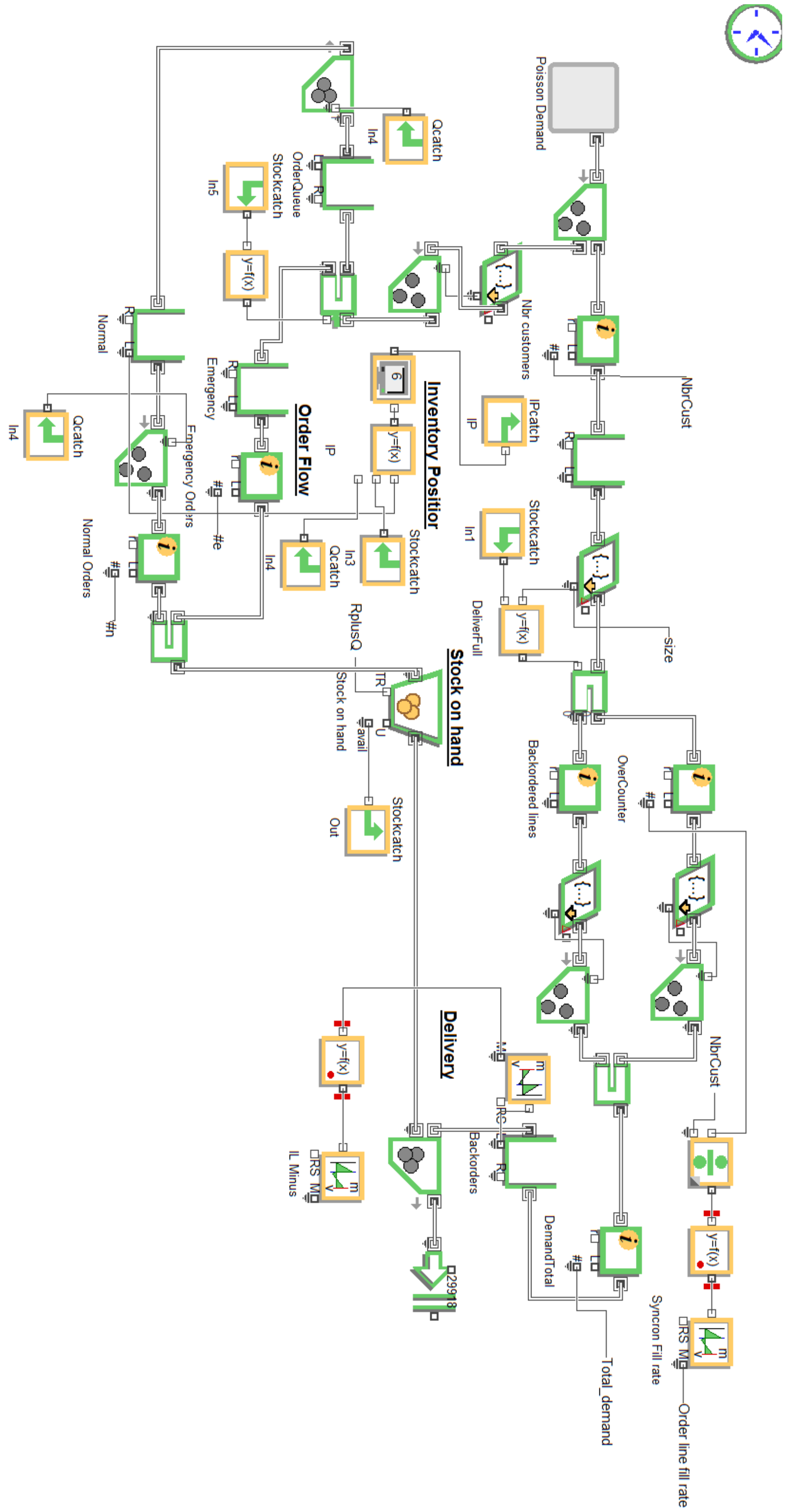


Figure 13: (R, Q)-LS simulation model

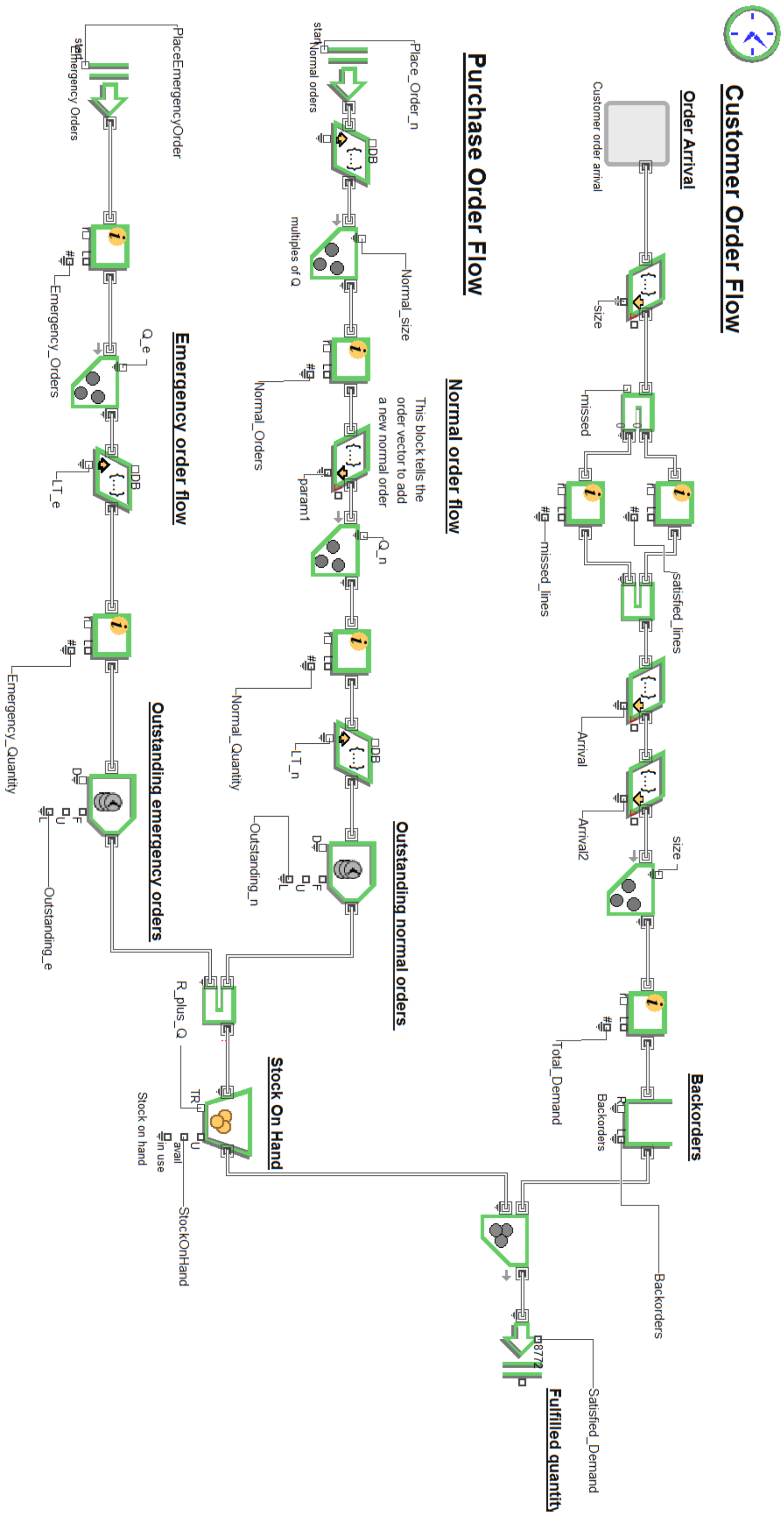


Figure 14: (R, Q)-SH simulation model

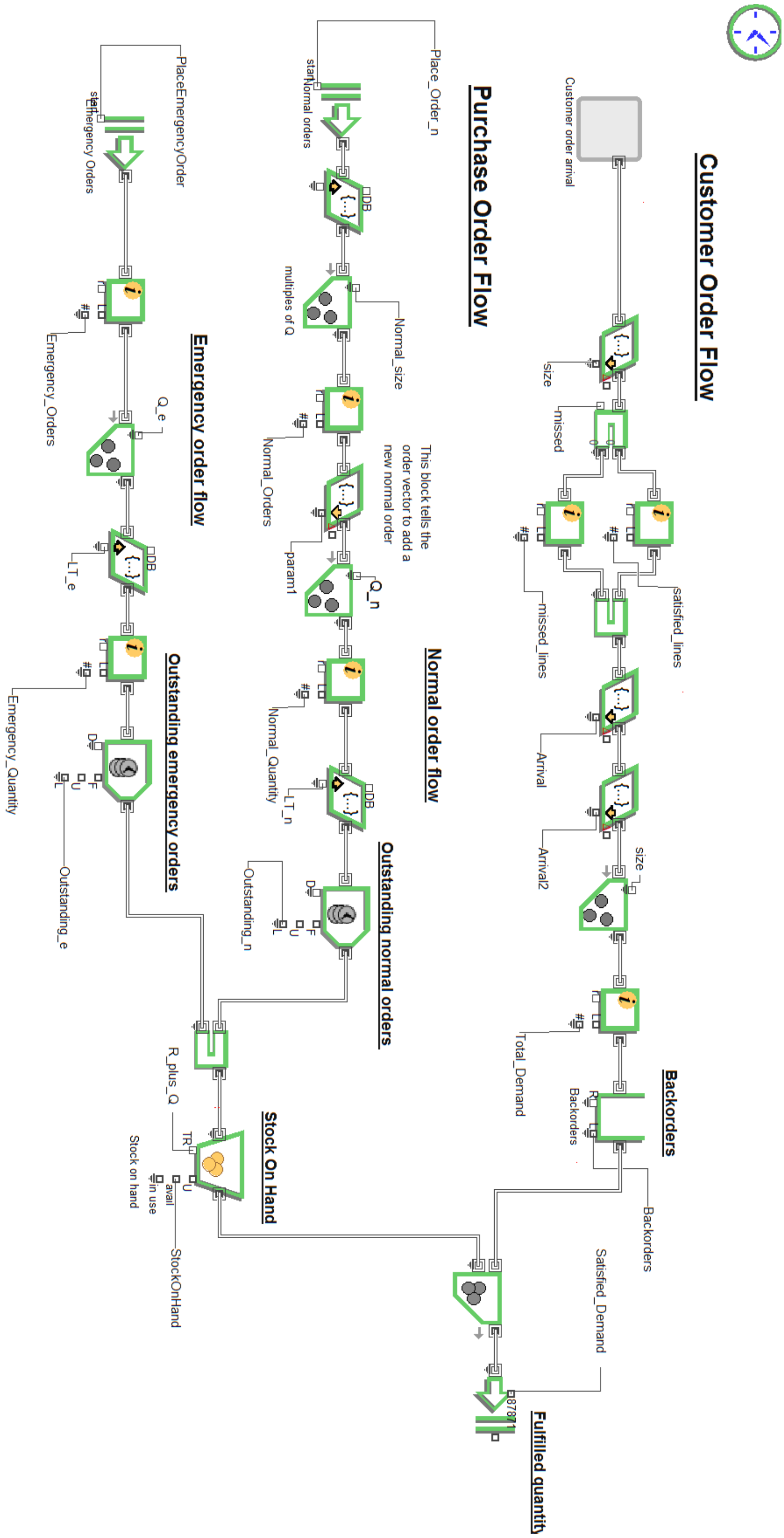


Figure 15: (R, Q)-LH simulation model

8.3 Validity of analytical and Extend software models

To prove the validity of the models covered above we will provide numerical examples for the base models (R, Q) and S&Z. Since the analytical models are implementations of analytical expressions their validity can be proven by getting the same output from identical input values.

8.3.1 (R, Q) – Normally distributed demand

The input values and resulting fill rate for item 4 can be viewed in Table 5 below. Then a numerical example taken from equation 3.22 follows.

Table 5: (R, Q) model validation for normally distributed demand

Item 4	Input values					Output
Parameter	R	Q	LT	D	STD	Fill rate (S2)
Equation 3.22	11	2	42	0,13133	0,507131	0,974
Analytical model	11	2	42	0,13133	0,507131	0,974
Simulation model (25 * 500000 days)	11	2	42	0,13133	0,507131	0,974

$$P(IL \leq 0) = \frac{\sigma'}{Q}\left[G\left(\frac{R - x - \mu'}{\sigma'}\right) - G\left(\frac{R + Q - x - \mu'}{\sigma'}\right)\right] \Rightarrow \tag{8.1}$$

$$S2 = 1 - P(IL \leq 0) = 1 -$$

$$\left[\frac{\sqrt{42} * 0{,}507131}{2}\left[G\left(\frac{11 - 0 - 42 * 0{,}13133}{\sqrt{42} * 0{,}507131}\right) - G\left(\frac{11 + 2 - 0 - 42 * 0{,}13133}{\sqrt{42} * 0{,}507131}\right)\right]\right]$$

$$= 0{,}974 \tag{8.2}$$

8.3.2 (R, Q) – Pure and compound Poisson distributed demand

The validation was done in two parts, first for pure Poisson demand and then for compounding Poisson demand. By inserting the values for item 6, pure Poisson demand, in equation 3.16 we get the results shown in Table 6. Observe that the achieved service levels were the same for both the analytical model and the Extend software as in the numerical example.

Table 6: (R, Q) model validation for pure Poisson demand

Item 6	Input values					Output
Parameters	R	Q	LT	λ	STD	Fill rate (S2)
Equation 3.16	2	1	42	0,02189	-	0,934
Analytical model	2	1	42	0,02189	-	0,934
Simulation model (25 * 500000 days)	2	1	42	0,02189	-	0,934

$$P(IL = j) = \frac{1}{Q} \sum_{k=max\{R+1,j\}}^{R+Q} \frac{(\lambda L)^{k-j}}{(k-j)!} e^{-\lambda L} \quad \Longrightarrow \tag{8.3}$$

$$S2 = P(IL > 0) =$$

$$P(IL > 0) = \sum_{j=1}^{2+1} \frac{1}{1} \sum_{k=max\{2+1,j\}}^{2+1} \frac{(0{,}02189 * 42)^{k-j}}{(k-j)!} e^{-(0{,}02189*42)} = 0{,}934 \tag{8.4}$$

Then in order to also test the compound Poisson demand generator we tested the model for item 4, which had demand sizes 1 and 2 with probabilities 0,08 and 0,92 respectively, see Appendix II. Recall from section 8.1.1 that the models we have constructed assume normally distributed or pure Poisson demand while the input values to the simulation models always are compound Poisson distributed. We therefore don't have an analytical model to compare with here. In order the check if the resulting fill rate was correct we needed to compute the theoretical fill rate with compound Poisson demand referred to in equation 3.18. The resulting fill rate with R = 12 and Q = 2 was 0,969 both in the equation and with our simulation model. The results are also presented in Table 7. The simulation was run 25 times over 100 000 days.

Table 7: (R, Q) model validation for compound Poisson demand

Item 4	Input values					Output
Parameter	R	Q	LT	λ	-	Fill rate (S2)
Equation (3.18)	11	2	42	0,0684	-	0,969
Simulation model (25 * 100000 days)	11	2	42	0,0684	-	0,969

$$P(IL = j) = \frac{1}{Q} \sum_{k=max\{R+1,j\}}^{R+Q} \sum_{i=0}^{\infty} \frac{(\lambda L)^i}{i!} e^{-\lambda L} f_{k-j}^i \quad \Rightarrow \tag{8.5}$$

$$S2 = P(IL > 0) = \sum_{j=1}^{R+Q} P(IL = j) = \tag{8.6}$$

$$\sum_{j=1}^{12+2} \frac{1}{2} \sum_{k=max\{12+1,j\}}^{12+2} \sum_{i=0}^{\infty} \frac{(*\,42)^i}{i!} e^{-(0{,}13133*42)} f_{k-j}^i \quad = 0{,}969 \tag{8.7}$$

8.3.3 (R, Q) - LS

Item 4 was used again for the numerical example of the lost sales model but the parameters were updated after the model had been run. Again we can see that both the cost expression and fill rate of the analytical model are the same as the output of our numerical example. Since we didn't have an analytical expression to compute the expected fill rate for the lost sales model, given compound Poisson demand, we were not able to apply the same validation method as for the other models above. Instead the flows were scrutinized step by step testing different scenarios in the slow motion running setting. The results from the simulation model and equations are presented in Table 8.

Table 8: (R, Q) - LS model validation

Item 4	Input values							Output	
Parameters	R	Q	LT	D	STD	h	b	C/t	Fill rate
Equation 3.38, 3.43	7	7	42	0,131 33	0,507 131	1,93	273	13,223	0,909
Simulation model	7	7	42	0,131 33	0,507 131	1,93	273	13,223	0,909

From equation 3.38 we get

$$C = \frac{h * \left(\frac{Q}{2\mu} + \frac{R - \mu' + \sigma' * G(\frac{R-\mu'}{\sigma'})}{\mu}\right) * Q + b * \sigma' * G(\frac{R-\mu'}{\sigma'})}{\frac{Q}{\mu} + \frac{\sigma' * G(\frac{R-\mu'}{\sigma'})}{\mu}} = \tag{8.6}$$

$$\left(1{,}93 * \left(\frac{7}{2 * 0{,}13133} + \frac{7 - 0{,}507131 * 42 + 0{,}507131 * \sqrt{42} * G\left(\frac{7 - 0{,}507131 * 42}{0{,}507131 * \sqrt{42}}\right)}{0{,}13133}\right) * 7\right.$$

$$+271 * 0{,}507131 * \sqrt{42} * G\left(\frac{7 - 0{,}507131 * 42}{0{,}507131 * \sqrt{42}}\right)\Bigg)$$

$$/\left(\frac{Q}{0{,}13133} + \frac{0{,}507131 * \sqrt{42} * G\left(\frac{7 - 0{,}507131 * 42}{0{,}507131 * \sqrt{42}}\right)}{0{,}13133}\right)$$

$$= 13{,}223. \tag{8.7}$$

And from equation 3.42 we get

$$S2 = \frac{Q}{Q + \sigma' * G\left(\frac{R - \mu'}{\sigma'}\right)} = \frac{7}{7 + 0{,}507131 * \sqrt{42} * G\left(\frac{7 - 0{,}507131 * 42}{0{,}507131 * \sqrt{42}}\right)}$$

$$= 0{,}909. \tag{8.8}$$

8.3.4 S&Z

A slightly different approach was chosen to validate S&Z. To spare the reader from needing to check an even larger number of equation lines than the previous case we choose to use the numerical values from the illustrative example Song & Zipkin (2009) present on p. 367 and plot our result against theirs. Figure 16 presents the results from our analytical and Extend simulation models against their results. Figure 17 presents their results alone. As we can see our results are close to identical to theirs using the analytical model. For the Extend model the values start to deviate when the probability of stock out approaches zero. This is because the random variations in a simulation model have a larger impact when we measure small probabilities.

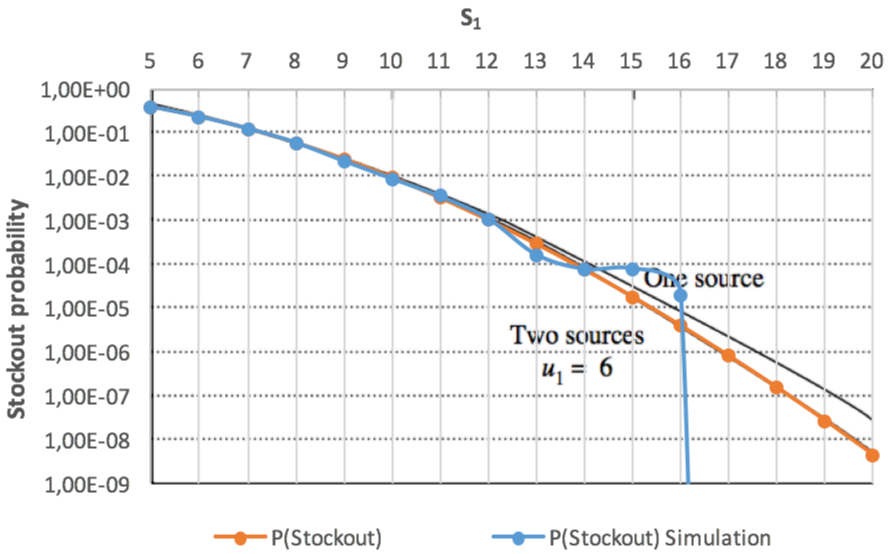


Figure 16: Stock out probability for S&Z. *P(Stockout)* shows observation points from our analytical model. *P(Stockout) Simulation* shows observation points from Extend simulations. The two black lines are from Song and Zipkin (2009) p. 367.

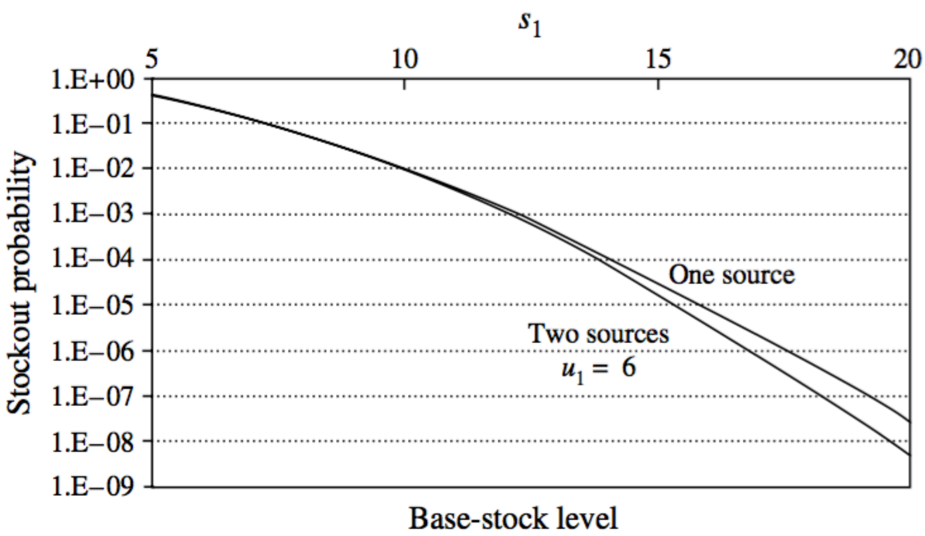


Figure 17: Stock out probability from Song & Zipkin (2009), p. 367 figure 6

9 Results & analysis

This section presents the results from our simulation runs together with analysis of the results. The simulations are divided into two sections, the first covering comparisons for slow demand items, especially evaluating Song and Zipkin (2009). The second part covers items with fast and erratic demand.

9.1 Comments with regards to cost comparisons

One of the problems that we had to confront in our comparison between the policies is that they are optimized according to different parameters, Target Service Level and cost respectively, see section 3.5 for a recap. S&Z and (R, Q)-LS are optimized based on cost while all other emergency ordering policies are optimized with regards to a TSL. Our solution here was to convert each TSL to the corresponding backorder cost, based on the R* that closest fulfilled the TSL, using equation 3.27. Note that the equation gives an interval. We chose to use the lower part of the interval when reformulating it to a backorder cost.

$$b = \frac{S_2(R^*)h}{1 - S_2(R^*)} \tag{9.1}$$

Note that equation 3.27 which 9.1 is derived from assumes an (R, Q) policy with compound Poisson demand. The backorder cost b will therefore result in a different service level when used as input to S&Z then the TSL its derived from.

9.2 Slow items

We have studied three slow items, two items that depicted a pure Poisson demand pattern, item 5 and item 6, and one item with compound Poisson demand, item 7. Recall from section 7.2 that item 7 was chosen to test the results if the pure Poisson assumption was false. Three policies were tested:

1. (R, Q)
2. (R, Q)-RH
3. S&Z

Item data used are the same as presented in Appendix II and the run out limit (ROL) of (R, Q)-RH was set to 0, equivalent to saying that emergency orders are to be placed if there is 50% probability of stock out.

9.2.1 Test approach

Three simulation runs were performed. Before each run the analytical models were re-run based on new backorder costs.

Analytical computation of policy parameters

In the first step we computed the optimal policy parameters analytically, for item 5 – 7, assuming pure Poisson demand. First R and Q were set for (R, Q). The same parameters were used for (R, Q)-RH. Finally s_1 and s_2 were set for S&Z using the derived backorder cost.

Initial simulation run

Using the now determined parameters we ran the initial simulations for all three policies and all three items. The backorder costs used were computed according to formula 9.1, using their respective TSLs.

Second simulation run, reassessing backorder costs

When using (R, Q)-RH the result is an increase in both achieved service and cost compared to (R, Q). This lead us to reset our target service level and backorder cost for the analytical models during the second run based on the service level achieved by (R, Q)-RH in the previous round. We then ran the simulations again with new parameter values for (R, Q) and S&Z. For (R, Q)-RH only the backorder cost parameter was updated. The reasoning behind this was to compare S&Z to (R, Q)-RH based on the service level that the latter policy achieved during simulations and let S&Z use the corresponding backorder cost as input.

Third simulation run, adjusting policy parameters to reach the same service levels

In the third simulation our aim was to compare S&Z and (R, Q)-RH at the same service level. Backorder costs were excluded from the cost calculations. Since they are optimized with different objectives, cost or TSL, they find different service levels to be optimal. If, however, they achieved the same service level the unit costs would be an indication of what policy was more efficient in its timing of emergency orders. Especially we wanted to see if the mechanism in S&Z, by which it authorizes emergency orders when too many outstanding orders are too far away in time, would be better at placing orders than (R, Q)–RH. Thus we adjusted the parameters R and ROL in (R, Q)-RH so that the same fill rate was attained for both policies. This was done by trying different parameter values in the simulation model.

9.2.2 Results

All simulation results that are presented here can be found in Appendix 4 in full detail, including mean values and standard deviations.

Initial Simulation Run

In Table 9 we see the results of the initial runs for items 5-7. For item 5 and 6 (R, Q) and (R, Q)-RH managed to reach their TSL (90%) by a margin while S&Z missed it with some distance for both items. Even though S&Z did not reach the TSL, it resulted in the lowest unit cost, backorder costs included. The cost increases for (R, Q) compared to S&Z were however only 0.35% for item 5 and 0,3% for item 6, while improving the service level with 15 and 9,2 percentage points respectively.

The analytical fill rate computed for item 5 and 6 with (R, Q) were 0,955 and 0,934 respectively. This was very close to the fill rates achieved with simulation, at 0,955 and 0,933 respectively. S&Z also reached the exact same fill rates during simulation as predicted analytically.

Table 9: Item 5-7, initial run with backorder costs based on TSL. Holding cost rate = 25%. Emergency lead time = 14 days.

Policy	R	Q	S_1	S_2	B1	ROL	Fill rate	Cost / unit	Emergency order utilization
					Item 5				
(R, Q)	3	1	N/A	N/A	5,59	N/A	0,955	412,70	N/A
S&Z	N/A	N/A	3	0	5,59	N/A	0,863	411,25	0,048
(R, Q)-RH	3	1	N/A	N/A	5,59	0	0,971	414,47	0,056
					Item 6				
(R, Q)	2	1	N/A	N/A	4,28	N/A	0,933	474,72	N/A
S&Z	N/A	N/A	2	0	4,28	N/A	0,783	473,08	0,105
(R, Q)-RH	2	1	N/A	N/A	4,28	0	0,959	478,57	0,094
					Item 7				
(R, Q)	3	1	N/A	N/A	10,50	N/A	0,832	1572,63	N/A
S&Z	N/A	N/A	3	1	10,50	N/A	0,783	1562,36	0,319
(R, Q)-RH	3	1	N/A	N/A	10,50	0	0,908	1586,29	0,272

For item 7 neither (R, Q) nor S&Z managed to achieve the target service level, at 90%, while (R, Q)-RH just made it. In terms of costs, S&Z was still the best. The cost increase of (R, Q) compared to S&Z is 0,66% and (R, Q)-RH comes with a cost increase of 1,53% compared to S&Z.

The analytical fill rate for (R, Q) was 0,912 and for S&Z 0,853. The corresponding results from the simulation were significantly lower, 0,832 and 0,783 respectively.

Second simulation run

Adjusting backorder costs and updating policy parameters gives the results presented in Table 10. S&Z now achieves a much higher fill rate for all items. Furthermore, S&Z is still the cost optimal policy (backorder costs included) even though the cost difference compared to (R, Q)-RH is very small for item 5 (0,06%) and item 6 (0,09%). For item 7 S&Z still does not reach the TSL at 90%, but the cost difference is bigger than for the other two items (0,94% cost increase for (R, Q)-RH and 0,96% for (R, Q)).

Table 10: Item 5-7, backorder costs based on fill rate achieved with (R, Q)-RH in the initial run. Holding cost rate = 25%. Emergency lead time = 14 days.

Policy	R	Q	S_1	S_2	B1	ROL	Fill rate	Cost / unit	Emergency order utilization
Item 5									
(R, Q)	4	1	N/A	N/A	23,04	N/A	0,989	420,83	N/A
S&Z	N/A	N/A	4	1	23,04	N/A	0,966	417,31	0,049
(R, Q)-RH	3	1	N/A	N/A	23,04	0	0,971	417,58	0,055
Item 6									
(R, Q)	3	1	N/A	N/A	20,70	N/A	0,986	487,81	N/A
S&Z	N/A	N/A	3	1	20,70	N/A	0,957	482,28	0,104
(R, Q)-RH	2	1	N/A	N/A	20,70	0	0,959	482,80	0,094
Item 7									
(R, Q)	4	1	N/A	N/A	35,88	N/A	0,918	1608,58	N/A
S&Z	N/A	N/A	4	1	35,88	N/A	0,877	1593,26	0,158
(R, Q)-RH	3	1	N/A	N/A	35,88	0	0,907	1608,30	0,272

Third simulation run

Table 11 shows the results when aiming to reach the same service level for S&Z and (R, Q)-RH. Note that for (R, Q)-RH the ROL parameter is now used, meaning that an order is only placed if there is 50% probability for the inventory level to fall below this limit. Furthermore the backorder costs are now excluded from the cost per unit. For item 5 and 6 the differences in cost per unit are small. For item 6 the cost for (R, Q)-RH is actually lower than for S&Z. For item 7 the difference is more apparent, the unit cost for S&Z is 0,57% lower.

Note that S&Z and (R, Q)-RH are not directly comparable with (R, Q) here since the service level achieved with (R, Q) is much higher.

Table 11: Item 5 – 7, achieving the same fill rate for S&Z and (R, Q)-RH with backorder costs excluded. Holding cost rate = 25%. Emergency lead time = 14 days.

Policy	R	Q	S1	S2	B1	ROL	Fill rate	Cost / unit	Emergency order utilization
Item 5									
(R, Q)	4	1	N/A	N/A	0	N/A	0,989	419,15	N/A
S&Z	N/A	N/A	4	1	0	N/A	0,967	412,32	0,048
(R, Q)-RH	3	1	N/A	N/A	0	-0,43	0,967	412,65	0,043
Item 6									
(R, Q)	3	1	N/A	N/A	0	N/A	0,986	484,92	N/A
S&Z	N/A	N/A	3	1	0	N/A	0,959	477,31	0,094
(R, Q)-RH	2	1	N/A	N/A	0	0	0,958	476,49	0,10
Item 7									
(R, Q)	4	1	N/A	N/A	0	N/A	0,919	1569,08	N/A
S&Z	N/A	N/A	4	1	0	N/A	0,876	1554,77	0,158
(R, Q)-RH	3	1	N/A	N/A	0	-1,0	0,875	1563,65	0,161

9.3 Fast and erratic items

The fast and erratic items are both modeled as having normally distributed demand in GIM. For this kind of demand there were few existing suitable models in the academic literature. We therefore relied on our own heuristics as well as an adaption of the lost sales model, described by Axsäter (2006). For the simulation study we consequently had five different policies to compare:

1. (R, Q)
2. (R, Q)-RH
3. (R, Q)-LS
4. (R, Q)-SH
5. (R, Q)-LH

9.3.1 Test approach

Analytical computation of policy parameters

All policies that are compared here are variations of the decision rule for how to place emergency orders given an (R, Q) policy. The optimal policy parameters R* and Q* are derived analytically from a given TSL for the four first policies. Therefore, they will use the same initial R* and Q* for every item. The difference between them will lie in the utilization of the emergency source and the timing of these orders. For (R, Q)-LS, R* is derived using its own analytical model.

(R, Q) vs. (R, Q)-RH

The first step was to compare (R, Q) to (R, Q)-RH. This was performed by computing R analytically given a Q from our data set and assuming normally distributed demand with mean and standard deviation computed from the transaction history. In the simulation model we used a compound Poisson process as input that corresponded to the transaction history. After initial simulations of both (R, Q) and (R, Q)-RH we then increased R in (R, Q) and re-ran the analytical models until the same or higher fill rate was achieved with (R, Q) as with (R, Q)-RH. This was performed for item 1 to 4. Our aim here was to see if raising the reorder point for the simple (R, Q) policy would achieve a higher service level more cost efficiently than enabling emergency orders through (R, Q)-RH.

Sensitivity analysis (R, Q) vs. (R, Q)-RH

For item 1 (erratic) and item 3 (fast) we performed a sensitivity analysis to see what would happen if we increased/decreased the parameters emergency lead time and holding cost rate.

(R, Q)-LH and (R, Q)-SH vs. (R, Q)-RH

In the second step we aimed to investigate which of the emergency ordering policies that performed best. For item 1 and item 3 we compared the emergency ordering policies (R, Q)-SH, (R, Q)-LH and (R, Q)-RH for two different scenarios. The first scenario represented a situation where (R, Q) was better than (R, Q)-RH while the other scenario represented the opposite situation. The scenarios were derived by comparing the results of (R, Q) and (R, Q)-RH for different combinations of the parameters holding cost rate and emergency lead time.

We also experimented with different Search Start and Ratio settings for (R, Q)-SH and (R, Q)-LH.

Decreasing R with (R, Q)-LS

Our third investigation regarded the possibility to lower the reorder point using the (R, Q)-LS. Recall that the lost sales model requires $R \leq Q$ to remain valid. When running the analytical model to obtain parameters we found that the optimal values always had $R > Q$. This was expected since R is set to cover demand during lead time which was 42 days. Since we wanted to simulate using parameters that were optimal we had to make a compromise here with our data. We increased the value of Q incrementally and re-ran the analytical models, step wise increasing R until an $R \leq Q$ was found optimal. Q was not adjusted for (R, Q) and (R, Q)-RH.

Since (R, Q)-LS also builds on the assumption that demand occurring during a stock out is satisfied with an emergency delivery we wanted to reflect this with a shorter lead time. The emergency option was therefore set to 2 days, according to the reasoning in section 7.1.

Finally, a sensitivity analysis was made for item 2 where Q was adjusted for all models.

9.3.2 Results

(R, Q) vs. (R, Q)-RH

The results of the simulations for our base scenario are presented in Table 12. The holding cost rate was 25% (holding cost equals 25% of the stocked value on a yearly basis) and the emergency lead time 14 days. For all four items we managed to achieve an equal or higher fill rate at a lower cost per unit with (R, Q) than with (R, Q)-RH.

Table 12: Item 1 - 4, (R, Q) vs. (R, Q)-RH. Holding cost rate = 25%. Emergency lead time = 14 days.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
Item 1								
(R, Q)	11	2	N/A	0,859	N/A	N/A	1593,80	N/A
(R, Q)-RH	11	2	0	0,903	N/A	N/A	1631,97	0,192
(R, Q)	13	2	N/A	0,912	N/A	N/A	1615,90	N/A
Item 2								
(R, Q)	8	2	N/A	0,963	N/A	N/A	279,97	N/A
(R, Q)-RH	8	2	0	0,974	N/A	N/A	285,74	0,101
(R, Q)	9	2	N/A	0,980	N/A	N/A	282,32	N/A
Item 3								
(R, Q)	38	8	N/A	0,933	N/A	N/A	619,61	N/A
(R, Q)-RH	38	8	0	0,967	N/A	N/A	627,27	0,101
(R, Q)	42	8	N/A	0,967	N/A	N/A	622,40	N/A
Item 4								
(R, Q)	11	2	N/A	0,950	N/A	N/A	2917,37	N/A
(R, Q)-RH	11	2	0	0,966	N/A	N/A	2956,56	0,101
(R, Q)	12	2	N/A	0,969	N/A	N/A	2932,42	N/A

Sensitivity analysis (R, Q) vs. (R, Q)-RH

The sensitivity of the previous results was tested in two cases for item 1 (erratic) and 3 (fast). In the first case the inventory holding cost rate was increased to 100%. The results are presented in Table 13. We can see that the cost difference between (R, Q) and (R, Q)-RH is now smaller but (R, Q) still retains the lowest cost per unit. It is thus evident that (R, Q) is superior even when the holding cost has increased by a factor of 4. In fact, the holding cost rate had to be increased to 330% for item 1 and 138% for item 3 before the cost for (R, Q)-RH was lower than the cost for (R, Q).

Table 13: Item 1 and 3, (R, Q) vs. (R, Q)-RH. Emergency lead time = 14 days. Holding cost rate = 100%.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
				Item 1				
(R, Q)	11	2	N/A	0,859	N/A	N/A	1910,43	N/A
(R, Q)-RH	11	2	0	0,903	N/A	N/A	2010,90	0,192
(R, Q)	13	2	N/A	0,912	N/A	N/A	1998,78	N/A
				Item 3				
(R, Q)	38	8	N/A	0,933	N/A	N/A	656,22	N/A
(R, Q)-RH	38	8	0	0,967	N/A	N/A	668,93	0,110
(R, Q)	42	8	N/A	0,967	N/A	N/A	667,30	N/A

In the second case the emergency lead time was decreased from 14 to 2 days. The results are presented in Table 14. The same pattern as in the previous remains. The (R, Q) policy is still better at achieving a service level at lower cost.

Table 14: Item 1 and 3, (R, Q) vs. (R, Q)-RH. Emergency lead time = 2 days. Holding cost rate = 25%.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
				Item 1				
(R, Q)	11	2	N/A	0,859	N/A	N/A	1593,80	N/A
(R, Q)-RH	11	2	0	0,914	N/A	N/A	1637,87	0,208
(R, Q)	14	2	N/A	0,932	N/A	N/A	1626,84	N/A
				Item 3				
(R, Q)	38	8	N/A	0,933	N/A	N/A	619,61	N/A
(R, Q)-RH	38	8	0	0,989	N/A	N/A	628,55	0,121
(R, Q)	48	8	N/A	0,990	N/A	N/A	626,36	N/A

(R, Q)-SH and (R, Q)-LH vs (R, Q)-RH

In this section we compare (R, Q)-RH with the heuristics that do take cost into consideration. In table Table 15 the results for our base case are presented for all four items. Note that the Cost/SL-ratio is set arbitrarily but according to the same formula for all four items.

Table 15: Item 1 - 4, Comparison between emergency ordering policies. Emergency lead time = 14 days. Holding cost rate = 25%.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
				Item 1				
(R, Q)-RH	11	2	0	0,903	N/A	N/A	1631,97	0,192
(R, Q)-SH	11	2	0	0,887	951,47	94%	1614,49	0,11
(R, Q)-LH	11	2	0	0,868	951,47	94%	1600,62	0,04
				Item 2				
(R, Q)-RH	8	2	0	0,974	N/A	N/A	285,74	0,101
(R, Q)-SH	8	2	0	0,975	388,64	97%	286,25	0,114
(R, Q)-LH	8	2	0	0,982	388,64	97%	291,10	0,199
				Item 3				
(R, Q)-RH	38	8	0	0,967	N/A	N/A	627,27	0,101
(R, Q)-SH	38	8	0	0,980	2108,51	95%	631,22	0,165
(R, Q)-LH	38	8	0	0,994	2108,51	95%	644,11	0,343
				Item 4				
(R, Q)-RH	11	2	0	0,966	N/A	N/A	2956,56	0,109
(R, Q)-SH	11	2	0	0,972	2601,23	95%	2958,92	0,112
(R, Q)-LH	11	2	0	0,977	2601,23	95%	2966,77	0,132

The results indicate a few things. Firstly, we note that from these simulations we cannot tell which of the policies that is superior, since the policies that achieve a higher service level consistently do that at a higher cost. Secondly we see that (R, Q)-LH tends to attain a higher service level than (R, Q)-SH if we use the same Cost/SL-ratio. Here it is also important to emphasize that with (R, Q)-RH emergency orders are only placed if there is a 50% risk of stock out while (R, Q)-LH and (R, Q)-SH can, if it is cost efficient, place emergency orders even when the risk is as small as 1 - Search Start.

Sensitivity Analysis

Next we investigated the efficiency of the policies by changing the policy-parameters such that the same fill rate was achieved for all policies. Note that R and Q remained the same for all emergency ordering policies, and that only the ROL and Ratio parameters were altered to achieve comparable fill rates. This was performed for two scenarios for both item 1 (erratic) and 3 (fast).

In the first scenario we used our base case, where it was previously shown that (R, Q) outperformed (R, Q)-RH. See the results in Table 16. The results are inconclusive. For item 1 it seems that (R, Q)-LH is clearly better than the other two, while for item 3 we have a marginal advantage for the (R, Q)-SH.

Table 16: Sensitivity case 1. (R, Q)-LH and (R, Q)-SH vs. (R, Q)-RH. Emergency lead time = 14 days. Holding cost rate = 25%.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
Item 1								
(R, Q)-RH	11	2	1,32	0,912	N/A	N/A	1638,96	0,226
(R, Q)-SH	11	2	N/A	0,913	1520	94%	1641,48	0,234
(R, Q)-LH	11	2	N/A	0,913	1535	94%	1637,24	0,223
Item 3								
(R, Q)-RH	38	8	0	0,967	N/A	N/A	627,27	0,101
(R, Q)-SH	38	8	N/A	0,967	1000	95%	627,00	0,104
(R, Q)-LH	38	8	N/A	0,967	1050	95%	627,89	0,119

The second scenario is when the emergency lead time is cut to 2 days and the holding cost rate is increased to 100%. This was deemed to be a representative case for a scenario where (R, Q)-RH is superior to (R, Q), although this could not be conclusively shown for item 1. The results for this scenario are presented in Table 17. We can see for both items that (R, Q)-SH seems to be slightly more efficient than the other two policies. For item 1 we also note that the single supplier (R, Q) policy (R=12, Q=2) was able to achieve the fill rate 0,888 at a cost of 1954,37 SEK/unit, thus only beaten by (R, Q)-SH. These results can be seen in Appendix IV.

Table 17: Sensitivity Case 2. (R, Q)-LH and (R, Q)-SH vs. (R, Q)-RH. Emergency lead time = 2 days. Holding cost rate = 100%.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
Item 1								
(R, Q)-RH	11	2	-3,68	0,887	N/A	N/A	1967,24	0,092
(R, Q)-SH	11	2	N/A	0,887	1063,83	94%	1953,96	0,091
(R, Q)-LH	11	2	N/A	0,888	1492,54	94%	1964,30	0,098
Item 3								
(R, Q)-RH	38	8	-2	0,981	N/A	N/A	668,20	0,091
(R, Q)-SH	38	8	N/A	0,981	Infinity	95%	666,53	0,078
(R, Q)-LH	38	8	N/A	0,981	880	95%	669,81	0,104

(R, Q)-LS vs. (R, Q) policies

We present two scenarios for (R, Q)-LS. One where results using the adjusted Q are compared to the previous results from Table 12 and one where all policies use the adjusted Q.

The results for the first scenario are presented in Table 18. By comparing (R, Q)-LS to (R, Q) for item 2 – 4, we see that costs are higher and fill rates lower for (R, Q)-LS. This lower efficiency of (R, Q)-LS is mainly due to the fact that the restriction on the size of Q has made us adjust it too far up. A smaller Q would have worked better but was not possible due to this restriction. It seems that (R, Q)-LS fits poorly with items that have low demand and small batch sizes.

Table 18: Item 1-4, (R, Q) vs. (R, Q)-LS. Emergency lead time = 2 days. Holding cost rate = 25%.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
				Item 1				
(R, Q)	11	2	N/A	0,859	N/A	N/A	1593,80	N/A
(R, Q)-RH	11	2	0	0,914	N/A	N/A	1637,87	0,208
(R, Q)-LS	3	3	N/A	0,495	N/A	N/A	1552,42	0,306
				Item 2				
(R, Q)	8	2	N/A	0,963	N/A	N/A	279,97	N/A
(R, Q)-RH	8	2	0	0,980	N/A	N/A	286,57	0,101
(R, Q)-LS	5	5	N/A	0,915	N/A	N/A	279,97	0,066
				Item 3				
(R, Q)	38	8	N/A	0,933	N/A	N/A	619,61	N/A
(R, Q)-RH	38	8	0	0,989	N/A	N/A	628,55	0,121
(R, Q)-LS	29	29	N/A	0,926	N/A	N/A	625,29	0,060
				Item 4				
(R, Q)	11	2	N/A	0,950	N/A	N/A	2917,37	N/A
(R, Q)-RH	11	2	0	0,981	N/A	N/A	2964,30	0,123
(R, Q)-LS	7	7	N/A	0,903	N/A	N/A	2926,54	0,081

In the second scenario we have adjusted Q for the other two models to see if the models could have worked if given higher batch quantities. Table 19 gives us an indication of this for item 2. Here the lost sales model finds a 5 percentage points lower service level to be optimal and a correspondingly lower cost.

Table 19: Item 2, (R, Q) vs. (R, Q)-LS. Emergency lead time = 2 days. Holding cost rate = 25%. Q = 5 for all policies.

Policy	R	Q	ROL	Fill Rate	Ratio	Search Start	Cost / unit	Emergency order utilization
Item 2								
(R, Q)	7	5	N/A	0,965	N/A	N/A	281,18	N/A
(R, Q)-RH	7	5	N/A	0,980	N/A	N/A	288,92	0,139
(R, Q)-LS	5	5	N/A	0,915	N/A	N/A	279,97	0,066

9.4 Analysis

9.4.1 Slow items

We have analyzed three slow items. Two that had pure Poisson demand and one that had a compounding distribution. As expected both the (R, Q) policy and the Song and Zipkin policy were very close to their analytical fill rates for the items with Poisson demand (5 and 6). This result confirms that our analytical and simulation models work as intended. However, none of the policies managed to hit the TSL set by the case company exactly. There are different reasons for this. In the case of (R, Q), the TSL was used as a minimum service level constraint, meaning that we aimed to find the R that would at least result in a fill rate equal to the TSL. Since the demand is low, an integer change in R will have a big impact on the expected service level. This makes it more or less impossible to hit a fill rate with precision.

In contrast to (R, Q), S&Z achieved a fill rate far below the TSL. The reason for this is that the two are optimized to different parameters, TSL and cost, and find different levels of service to be optimal. The fact that S&Z achieved a lower fill rate indicates that our service level to backorder cost translation underestimates the backorder costs. This was expected since the translation was made using the analytical model for (R, Q). Another explanation for the big difference between the fill rates of (R, Q) and S&Z is the big changes that come with an integer step in parameters. There does not necessarily exist parameters such that $SL(R, Q) = SL(S_1, S_2)$.

Even though S&Z did not manage to achieve the TSL it managed to find the best solution from a cost perspective for all items and scenarios, when including backorder costs. In run 3, when backorder costs were disregarded and the efficiency of the policies was tested, S&Z performed better than (R, Q)-RH for item 5 and 7, but was outperformed for item 6. However, it was hard to achieve comparable service levels and the cost differences were small. We can therefore not conclude that the order timing mechanism in S&Z is better than the corresponding mechanism for (R, Q)-RH.

For item 7, where demand was not a pure Poisson process, neither (R, Q) nor S&Z was close to achieving its analytical service level. This tells us that we should be careful to use the policies when demand is not actually Poisson. For item 7, (R, Q)-RH was the only policy that managed to achieve the TSL. (R, Q)-RH thus has the ability to compensate for erroneous assumptions, but with added cost.

The conclusion from our simulation study is that, when the aim is to minimize total costs, S&Z outperforms both (R, Q) and (R, Q)-RH given that the backorder costs are known and correctly determined. Furthermore, the translation between fill rate and backorder costs used here seems to be inaccurate for S&Z. Both S&Z and (R, Q) are highly sensitive to false assumptions regarding demand. Finally (R, Q)-RH can compensate for bad assumptions, but this seems to come at unnecessarily high cost.

9.4.2 Fast and erratic items

The first conclusion that can be drawn is that, if the aim of using the emergency option is to achieve a higher fill rate cost efficiently, this is better done by adjusting the parameters of (R, Q). Raising the reorder point resulted in significantly lower costs for all four items while achieving at least the same fill rate as (R, Q)-RH. One should therefore be careful before enabling emergency ordering. If a company wants to satisfy a key customer quickly this is better done by placing emergency orders manually outside the regular ordering system.

The second conclusion, which follows intuition, is that the usefulness of the emergency source increases with shorter lead time and higher holding cost (or lower emergency unit cost). Adding to this the fact that a few assumptions were made regarding the different costs and lead times meant that a sensitivity analysis was necessary. Although we found parameter combinations where it was advantageous to use an emergency ordering option, major changes were required to achieve this. A holding cost of 100% of the unit purchase value on an annual basis, or a decrease of the emergency order lead time from 14 to 2 days, was not enough to favor (R, Q)-RH. This strongly supports the removal of the emergency ordering option for the fast and erratic items, until a thorough analysis concludes that it is actually beneficial.

Regarding our heuristics for improving (R, Q)-RH there seems to be only minor differences between them. By adding a Cost/Service-ratio our heuristics place fewer emergency orders but also achieve lower service. They seem to have a slight advantage over the current policy when service is held at the same level. But in general the holding cost was small in comparison to the extra unit cost for the emergency source, so the main predictor of cost was the utilization of that source. Under these conditions (R, Q)-LH performed slightly better than (R, Q)-SH.

If the holding cost had been higher in comparison to the cost for the emergency source and if the emergency lead time had been lower, the results could have been different. Our sensitivity analysis indicates that (R, Q)-SH could perform better than both (R, Q)-RH and (R, Q)-LH as well as (R, Q) under such circumstances. It remains however to investigate how the policies perform when lead times are stochastic. Intuitively the policy with the shorter horizon should suffer more when lead times are non-constant.

Finally, for (R, Q)-LS the $R \leq Q$ requirement makes it unsuitable for the articles we have studied. The analytical computations could not find optimal values that did not violate this restriction. When increasing the order quantity and decreasing the emergency lead time, we could see that it was, as expected, possible to reduce the reorder point with (R, Q)-LS while lowering costs. This shows that if all the assumptions for (R, Q)-LS are fulfilled it works well.

The (R, Q)-LS also relies on the assumption that customers are satisfied if they get their products after one emergency lead time instead of over the counter. This certainly adds to the complexity of the concept service level. If items outside of the sample in this study have emergency lead times that are short enough to satisfy otherwise lost sales, a more detailed study could be initiated. For the items studied here we find it to be too restrictive for Syncron to use.

9.4.3 Implementation aspects

The results from our study only cover single item optimization. As we have described in section 4.1.5 Syncron optimizes service levels for many different items at the same time. In this thesis we have suggested two types of emergency ordering policies. Those that are added on top of a single supplier (R, Q) policy for the sake of improving the service level, (R, Q)-RH, (R, Q)-SH and (R, Q)-LH, and those that utilize the difference in unit cost between the two supply options to optimally determine policy parameters, S&Z and (R, Q)-LS.

A problem for Syncron is that the latter kind is not compatible with their optimization procedure for groups of items, where the marginal increase in stock value to cover another pick is compared for different items. Here we identify one of the major drawbacks of Syncron's inventory optimization procedure: one could never incorporate a cost minimizing dual sourcing policy into their optimization procedure for a group of items.

If you only have one single supply option and a fix order quantity, the cost of sold goods will be independent of how you set your remaining policy parameters. Therefore it is rational to ignore the cost of sold goods when optimizing the policy parameters for a group of items, just as Syncron does. When you have multiple options however, the cost of sold goods depends on the utilization of the different options, and thus on how you set the policy parameters. The added cost variables of a multiple sourcing problem are consequently not supported by Syncron's multi-item optimization scheme.

If you disregard the more complex dynamics of multiple sourcing, you cannot make use of the inventory reducing capacity that a multi-supplier ordering system provides. This leaves Syncron with emergency ordering policies that are added on top of single supplier ordering policies. These policies can have the ability to improve the service level, but there will be no guarantee that this is cost optimal. On the contrary, as we have seen in our simulation study, it often turns out to be less efficient than to increase the reorder point for a simple, and easily understandable, (R, Q) policy.

Our guess is that Syncron wants to stick with their successful multi-item optimization scheme. The question is therefore how the add-on emergency ordering policy can be improved. We have developed and tested two emergency ordering heuristics in this thesis. Non of them show compelling and conclusive improvements on the single item level. We argued, in section 6.6, that a slight adaption of the Cost/SL-ratio definition could be made in order to measure the improvement of service level in quantity rather than in percent. Such an adaption would provide the opportunity to set a cost limit per pick covered for a group of items. Consequently, one would only place emergency orders for items where the extra cost per satisfied unit demand is lower than a given limit.

This solution would leave the original multi-item optimization procedure in tact and add another optimization procedure for emergency ordering on top of it. The utility of such an idea would of course have to be tested in a multi-item setting.

9.4.4 Credibility of our results

The credibility of our results has already been discussed in previous sections and chapters. Our greatest concern regarding the results is the data accuracy. We have made some assumptions regarding costs, lead times and demand distributions and will discuss these below.

The emergency ordering costs were extracted from the transport cost rate table attained from our case company. The costs were built up by a fix cost and a variable cost per weight unit. Depending on the amount of goods sent with an emergency order the costs can vary substantially. Furthermore, we did not receive any freight rates for normal shipments, these were therefore assumed to be negligible in comparison to the emergency shipment rates. Moreover, we did not receive any holding cost. Therefore we used a holding cost rate of 25%, a common assumption in the literature, see section 7.4.

Since there was a lot uncertainty regarding both the extra cost for emergency ordering and the cost of holding stock, we performed sensitivity analyses where the holding cost was increased drastically. Even with a holding cost rate well above 100% our results did not change. Thus we do not deem our holding cost assumptions to have an impact on our conclusions.

The lead times that we received directly from the system were always 42 and 14 days for normal and emergency orders respectively. According to the information received from our case company, emergency shipments could be made twice a week, sending the goods from USA to Stockholm. These shipments would all in all take two days. Since our customer facing retailers were located in northern Sweden, a quite long transportation distance remained. From this scarce information it was impossible to predict the actual lead time. Therefore the original 14 days were used. The information about the transportation routes and departure and arrival times instead highlighted our concern about the assumption of constant lead times.

Sensitivity analyses were performed where the emergency lead time was shortened significantly and the results from these simulations have been discussed and taken into account. We did not perform simulations with stochastic lead times and therefore uncertainty remains in that area.

There was also some uncertainty concerning the demand distribution used in our simulation models. These were derived from real data, but it was not possible to compute the time between customer arrivals since only the day of the arrival was recorded. Some of the observed demand transactions occurred on the same dates. These observations where added together when measuring demand per day. Thus we had a higher standard deviation of the demand per day in our data sample than for our fitted distributions, which took all transactions into account. This was shown in section 7.3. Since different items were tested, different demand processes were also tested and our conclusions remain credible from that point of view.

A final problem that we had was that it was hard to achieve the same service levels for different policies. When trying to reach the same fill rate for different policies, policy parameters had to be adjusted. For some policies, a certain fill rate could also be achieved with different parameter combinations and at different costs. We therefore tried to keep the reorder point R the same for all emergency ordering heuristics and to the extent possible change a parameter, e.g. Ratio, in the same direction for all alternatives. This problem of course affects the credibility of our simulation results, but we have been careful to only draw conclusions from unambiguous results.

To conclude we find that the validity of our results to be the weakest part of our analysis. It has been hard to measure all policies in an equal manner given that policies are optimized with respect to two different parameters, cost and service level, in their original form. We have tried to make the comparisons better by measuring results at similar service levels but the results are not directly comparable. The reliability of the results is however much stronger given thorough validation of our models based on analytical expressions and documented input values. Our sample of seven items is smaller then we would have wished but given the time consuming nature of the simulations a much larger sample would not have been feasible. The case company that the sample originates from was also reluctant to hand out more information.

10 Conclusion and recommendation

In this chapter we try to answer the questions stated in the problem formulation and provide Syncron with a recommendation for future actions.

What are the theoretical and practical problems that prevent Syncron from implementing an emergency ordering policy in GIM based on research within the field?

We have identified three main aspects that prevent Syncron from implementing an emergency ordering policy based on the research in the field.

First of all, we find that no policies exist that can optimize the general problem with varying demand patterns, ordering quantities and lead times. As stated by previous research this problem quickly grows to a size that cannot be solved in reasonable time. A policy with some attached constraints must therefore be used.

Second, the multi-item optimization procedure used at Syncron disregards the different costs between two supply options and only compares stock value. It therefore uses too few cost parameters to be able to compare emergency ordering policies. Consequently, emergency ordering policies must either be added on top of the current single supplier ordering policies or the multi-item optimization procedure must be changed.

Third, presented in the previous work by Schmidt (2015), most of the heuristics found in literature are optimized with regards to cost. This includes the cost of backorders. As we have shown there are ways to translate a target service level to a backorder cost given a certain policy and demand pattern. For many policies and demand parameters these relations are however still to be derived, which makes backorder cost including policies unsuited for Syncron for the time being.

If an emergency ordering policy could be implemented in GIM under what circumstances should the policy be used?

We have shown that by restricting the problem scope there are models that can be used. We have evaluated four such policies. The first is a dual index policy by Song and Zipkin (2009) that uses two order up to levels to control when normal and emergency orders should be placed. The model takes the timing of outstanding orders into account and thus requires continuous tracking of orders. The model comes with the following restrictions. Demand is assumed to be a pure Poisson process with customers ordering one item at a time. Our results indicate that the performance of the model deteriorates quickly if this assumption is false. The model is optimized with regards to backorder costs that must be determined correctly. We have provided one approximation for doing this.

The second policy is a reinterpretation of the classical lost sales model as presented by Axsäter (2006). We have set the cost of lost sales to equal the cost difference of ordering between two suppliers. The model comes with the following restriction. Demand is assumed to be normally distributed. The emergency lead time must be short enough so that no customers are lost, i.e. cancel their orders. The batch size Q must be greater than the reorder point R and consequently only one normal order can be outstanding at a time.

The final two policies are heuristics that we have developed to improve the emergency ordering policy that Syncron currently uses. They offer no performance guarantee. Their main improvement compared to the current heuristic is that they take the cost of ordering into account and thus filter out emergency orders that would be too costly. However our results indicate that the gains from this are negligible. Both can be implemented with minor changes to the current heuristic, but should only be used when the emergency ordering costs are low enough to not erase the gain from reduced inventory. Our results indicate that employing our heuristics becomes relevant when the holding cost is four times higher and lead times are down to two days compared to our base case scenario with 14 days lead time.

If an emergency ordering policy could be implemented in GIM, how would such an ordering policy perform in comparison to their current heuristic? Against a policy without emergency ordering?

The first conclusion to be drawn is that the heuristic used by Syncron (R, Q)-RH, was outperformed by a simple (R, Q) policy under real conditions. Our suggested improvements of this heuristic, (R, Q)-SH and (R, Q)-LH, only achieved slightly better results. Increasing the holding cost and decreasing the emergency lead times showed, however, that there are situations where even the simple rush ordering heuristic used by Syncron can be better than the (R, Q) policy. Under these conditions the short horizon heuristic, which postponed the emergency ordering decision as long as possible, was the top performer among the mentioned policies. The average cost decrease compared to (R, Q)-RH was 0,46%.

The second conclusion to be drawn is that the policy by Song and Zipkin outperforms all other policies when demand follows a pure Poisson process and we include backorder costs. This means that if we set the backorder costs correctly, S&Z can provide significant cost reductions compared to both the simple (R, Q) policy and Syncron's rush ordering heuristic. The average cost reduction was 0,45% compared to (R, Q) and 1,14% compared to (R, Q)-RH when backorder costs were derived from the target service level stated by the case company.

Finally, the adapted lost sales model (R, Q)-LS, outperformed both (R, Q) and (R, Q)-RH when the order quantity was increased to satisfy the constraints of (R, Q)-LS. The cost reduction was 0,43% compared to (R, Q) and 3,1% compared to (R, Q)-RH. The conclusion to be drawn is that (R, Q)-LS can reduce stock and thus costs for items where the normal ordering cost is high and when the emergency lead time is short enough.

What are our recommendations?

Our overall recommendation to Syncron is to use more caution when enabling emergency ordering. In most cases analyzed it has been proven more efficient to increase the reorder point in a simple (R, Q) policy than enabling emergency orders with their current rush ordering heuristic.

For slow items we further suggest separating the items that do not depict pure Poisson demand from those that do. Including full-optimizing ordering policies in the multi-item optimization procedure that Syncron uses today is not feasible, and the value of S&Z would be relatively small even if it was possible. We therefore believe that modeling non-Poisson items with a compound Poisson distribution is a more promising direction to go.

At this stage we do not recommend Syncron to implement any of our tailor-made heuristics for fast and erratic items. The results of our simulation study are not convincing enough to implement these heuristics. We instead recommend Syncron to investigate if the advantages of these policies are more apparent in a multi-item setting, where they could mimic the current way of optimizing single supplier systems in that they allow service improvements where it is the most efficient.

The lost sales model offers interesting opportunities for the future, but only for a limited amount of items. Given that and that it is incompatible with the multi-item optimization procedure, we believe that its current coverage area is far to small to be attractive to Syncron´s customers.

11 References

Alford, Leon Pratt, and John Robert Bangs, eds. (1948), *Production handbook*. Ronald Press Co.

Arts, J. J. (2009), *Efficient optimization of the Dual-Index policy using Markov Chain approximations*, Master's Thesis, Eindhoven University of Technology, Eindhoven, Netherlands.

Axsäter, S. (2006), *Inventory Control*, 2nd edn, Springer, New York, NY.

Axsäter, S. (2007), A heuristic for triggering emergency orders in an inventory system. *European Journal of Operational Research*, vol. 176, pp. 880-891.

Axsäter, S. (2014), An improved decision rule for emergency replenishments, *International Journal of Production Economics*, vol. 157, pp. 313-317.

Berling, P. (2005). *On determination of inventory cost parameters*. Lund University.

Chiang, C. & Gutierrez, G. J. (1996), A periodic review inventory system with two supply modes, *European Journal of Operational Research*, vol. 94, no. 3, pp. 527–547.

Doyle, C. (2011), *In A Dictionary of Marketing*, Oxford University Press, Retrieved 14 Sep. 2015, from http://www.oxfordreference.com.ludwig.lub.lu.se/view/10.1093/acref/9780199590230.001.0001/acref-9780199590230-e-1606.

Golicic, Susan L., Donna F. Davis, & Teresa M. McCarthy. (2005)"A balanced approach to research in supply chain management.", *Research methodologies in supply chain management*, Physica-Verlag HD, pp. 15-29.

Hillier, F. S., & Lieberman, G. J. (2005). *Introduction to operations research*, 8th edn, McGraw-Hill, Boston, MA.

Johansen, S.G., Thorstenson, A. (1998) An inventory model with Poisson demands and emergency orders, *International Journal of Production Economics*, vol. 56-57, pp. 275–289.

Johansen, S.G. & Thorstenson, A. (2014) Emergency orders in the periodic-review inventory system with fixed ordering costs and compound Poisson demand, *International Journal of Production Economics*, vol. 157, pp. 147-157.

Kotzab, H., Müller, M., Reiner, G. & Seuring, S. (2005), *Research Methodologies in Supply Chain Management*, Learning Materials in pdf, Lund University, 20th January 2015, viewed 7 October 2015.

Laguna, M. & Marklund, J. (2005), Business process modeling simulation and design, Pearson/Prentice Hall, Upper Saddle River, NJ.

Mayring, P. (2003), *Qualitative Inhaltanalyse – Grundlagen und Techniken* (Qualitative Content Analysis – Basics and Techniques), 8th edn, Beltz Verlag, Weinheim, Germany.

Minner, S., (2003), Multiple-supplier inventory models in supply chain management: A review, *International Journal of Production Economics,* vol. 81–82, pp. 265–279.

Moinzadeh, K., Nahmias, S., (1988), A continuous review model for an inventory system with two supply modes, *Management Science*, vol. 34, no 6, pp. 761–773.

Moinzadeh, K., & Schmidt, C. P. (1991), An (S− 1, S) inventory system with emergency orders, *Operations Research*, vol. 39, no. 2, pp. 308-321.

Rosling, K. (2002) The (r, Q) Inventory Model with Lost Sales, Vaxjo University, Sweden.

Schmidt, C. (2015), *Multiple supplier/delivery inventory models - a review from an after-market perspective*, Master's Thesis, Umeå University, Umeå, Sweden.

Song, J.S. & Zipkin, P. (2009), Inventories with Multiple Supply Sources and Networks of Queues with Overflow Bypasses, *Management Science*, vol. 55, no. 3, pp. 362-372.

Veeraraghavan, S. & Scheller-Wolf, A. (2008), Now or later: A simple policy for effective dual sourcing in capacitated systems, *Operations Research*, vol. 56, no. 4, pp. 850–864.

Whittemore, A.S. & Saunders, S.C. (1977), Optimal inventory under stochastic demand with two supply options, SIAM Journal on Applied Mathematics, vol. 32, no. 2, pp. 293-305.

Appendix I

Articles from literature review

Abginehchi et al. (2015)

Allon and Van Mieghem (2010)

Arts (2009)

Arts and Kiesmüller (2013)

Arts et al. (2011)

Axsäter (2007)

Axsäter (2014)

Axsäter et al. (2013)

Barankin (1961)

Çapar et al. (2011)

Chiang (2002)

Chiang (2010)

Chiang and Gutierrez (1996)

Fu et al. (2013)

Gallego et al. (2007)

Groenevelt and Rudi (2003)

Howard et al. (2015)

Huggins and Olsen (2010)

Jain et al. (2010)

Janakiraman et al. (2015)

Johansen and Thorstenson (1998)

Johansen and Thorstenson (2014)

Kim et al. (2009)

Lawson & Porteus (2000)

Mamami and Moinzadeh (2014)

Minner (2003)

Moinzadeh and Nahmias (1988)

Moinzadeh and Schmidt (1991)

Muharremoglu and Tsitsiklis (2003)

Olsson (2015)

Özsen (2014)

Özsen and Thonemann (2012)

Özsen and Thonemann (2014)

Pérez and Geunes (2014)

Roni et al. (2015)

Rosenshine and Obee (1976)

Sheopuri et al. (2010)

Shuo et al. (2011)

Song and Zipkin (2009)

Tagaras and Vlachos (2001)

Veeraraghavan and Scheller - Wolf (2008)

Whittemore and Saunders (1977)

Zhang and Hua (2013)

Zhang et al. (2012)

Zheng (1992)

Zhou and Chao (2010)

Zhou and Zhao (2009)

Zhou et al. (2011)

Zipkin (2008)

Additional references

Abginehchi, S., Larsen, C. & Thorstenson, A. (2015), Exploring the economic consequences of letting a supplier hold reserve storage, *Computers & Operations Research*, vol. 53, pp. 223-233.

Allon, G. & Van Mieghem, J. A. (2010), Global dual sourcing: Tailored base-surge allocation to near-and offshore production. *Management Science*, vol. 56, no. 1, pp. 110-124.

Arts, J. & Kiesmüller, G. P. (2013), Analysis of a two-echelon inventory system with two supply modes. *European Journal of Operational Research*, vol. 225, no. 2, pp. 263-272.

Arts, J., Van Vuuren, M. & Kiesmüller, G. (2011). Efficient optimization of the dual-index policy using Markov chains. *IIE Transactions*, vol. 43, no. 8, 604-620.

Axsäter, S., Howard, C. & Marklund, J. (2013), A distribution inventory model with transshipments from a support warehouse, *IIE Transactions,* vol. 45, no 3, pp. 309-322.

Barankin, E. W. (1961), A delivery-lag inventory model with an emergency provision (the single-period case), *Naval Research Logistics Quarterly*, vol. 8, pp. 285–311.

Çapar, İ., Ekşioğlu, B. & Geunes, J. (2011), A decision rule for coordination of inventory and transportation in a two-stage supply chain with alternative supply sources, *Computers & Operations Research*, vol. 38, no. 12, pp. 1696-1704.

Chiang, C., (2002), Ordering policies for inventory systems with expediting, *Journal of the Operational Research Society*, vol. 53, no. 11, pp. 1291–1295.

Chiang, C. (2010), An order expediting policy for continuous review systems with manufacturing lead-time, *European Journal of Operational Research*, vol. 203, no. 2, pp. 526–531.

Fu, K., Xu, J. & Miao, Z. (2013), Newsvendor with multiple options of expediting, *European Journal of Operational Research*, vol. 226, no. 1, pp. 94-99.

Gallego, G., Jin, Y., Muriel, A., Zhang, & A. Yildiz, T. (2007), Optimal ordering policies with convertible lead- times, *European Journal of Operational Research,* vol. 176, pp. 892–910.

Groenevelt, H. & Rudi, N. (2003), *A base stock inventory model with possibility of rushing part of an order*, Working paper, University of Rochester, Rochester, NY.

Howard, C. Marklund, J. Tan T. & Reijnen I., (2015), Inventory Control in a spare parts distribution system with emergency stocks and pipeline information, *Manufacturing & Service Operations Management,* vol. 17, no. 2, pp. 142-156.

Huggins, E. L. & Olsen, T. L. (2010), Inventory control with generalized expediting, *Operations research*, vol. 58, no. 5, pp. 1414-1426.

Jain, A., Groenevelt, H. & Rudi, N. (2010), Continuous review inventory model with dynamic choice of two freight modes with fixed costs, *Manufacturing & Service Operations Management*, vol. 12, no. 1, pp. 120–139.

Janakiraman, G., Seshadri, S., & Sheopuri, A. (2014), Analysis of tailored base-surge policies in dual sourcing inventory systems, *Management Science*, vol. 61, no. 7, pp. 1547-1561.

Kim, C. Klabjan, D. & Simchi-Levi, D. (2009), *Optimal policy for expediting orders in transit*, Working paper, Massachusetts Institute of Technology, Cambridge, MA.

Lawson, D. & Porteus, E. (2000), Multistage inventory management with expediting, *Operations Research*, vol. 48, no. 6, pp. 878-893.

Mamani, H. & Moinzadeh, K. (2014), Lead time management through expediting in a continuous review inventory system, *Production and Operations Management*, vol 23, no. 1, pp. 95-109.

Muharremoglu, A. & Tsitsiklis, J. N. (2003), *Dynamic leadtime management in supply chains*, Working paper, Massachusetts Institute of Technology, MA.

Olsson, F. (2015), Emergency lateral transshipments in a two-location inventory system with positive transshipment leadtimes, *European Journal of Operational Research*, vol. 242, no. 2, pp. 424-433.

Özsen, R. (2014), *Order Expediting in Supply Chains: Models Solution Approaches and Applications,* 7th edn, LIT Verlag, Münster, Germany.

Özsen, R. & Thonemann, U. W. (2012), *An optimal expediting policy for periodic-review inventory systems*, Working Paper, University of Cologne, Cologne, Germany.

Pérez, C. & Geunes, J. (2014), A (Q, R) inventory replenishment model with two delivery modes, *European Journal of Operational Research*, vol. 237, pp. 528–545.

Roni, M., Mingzhou, J. & Eksioglu, S. (2015), A hybrid inventory management system responding to regular demand and surge demand, *Omega*, vol. 52, pp. 190-200.

Rosenshine, M. & Obee, D. (1976), Analysis of a standing order inventory system with emergency orders, *Operations Research*, vol. 24, no. 6, pp. 1143-1155.

Sheopuri, A., Janakiraman, G. & Seshadri, S. (2010), New policies for stochastic inventory control problem with two supply sources, *Operations Research*, vol. 58, no. 3, pp. 734-745.

Shuo, H., Axsäter, S., Yifan, D. & Jian, C. (2011), A real-time decision rule for an inventory system with committed service time and emergency orders, *European Journal of Operational Research*, vol. 215, no. 1, pp. 70-79.

Tagaras, G. & Vlachos, D. (2001), A periodic review inventory system with emergency replenishments, *Management Science*, vol. 47, no. 3, pp. 415–429.

Zhang, W., & Hua, Z. (2013), Optimal inventory policy for capacitated systems with two supply sources, *Operations Research Letters*, vol. 41, no. 1, pp. 12-18.

Zhang, W., Hua, Z., & Benjaafar, S. (2012), Optimal inventory control with dual-sourcing, heterogeneous ordering costs and order size constraints, *Production and Operations Management*, vol. 21, no. 3, pp. 564-575.

Zheng, Y. S. (1992), On properties of stochastic inventory systems, *Management Science*, vol. 38, no. 1, pp. 87–103.

Zhou, S. X. & Chao, X. (2010), Newsvendor bounds and heuristics for serial supply chains with regular and expedited shipping, Naval Research Logistics (NRL), vol. 57, no. 1, pp. 71-87.

Zhou, Y. & Zhao, X. (2009), A lost-sale inventory system with dual supply and multiple demand classes, In *Industrial Engineering and Engineering Management, 2009, IEEM 2009, IEEE International Conference on* (pp. 533-537), IEEE.

Zhou, D., Leung, L. C. & Pierskalla, W. P. (2011), Inventory management of platelets in hospitals: optimal inventory policy for perishable products with regular and optional expedited replenishments, *Manufacturing & Service Operations Management*, vol. 13, no. 4, pp. 420-438.

Zipkin, P. (2008), Old and new methods for lost-sales inventory systems, *Operations Research*, vol. 56, no. 5, pp. 1256-1263.

Appendix II

Item data

Item	Type	C_n	C_e	H	Int. Time	Demand / Day	Std. Dev.	Q	L_n	L_e	TSL
1	Erratic	1488	1579	1,02	56,23	0,08755	0,737875	2	42	14	0,94
2	Erratic	265	308	0,18	26,11	0,07524	0,389675	2	42	14	0,97
3	Fast	607	662	0,42	5,62	0,59781	1,481763	8	42	14	0,95
4	Fast	2814	3086	1,93	14,62	0,13133	0,507131	2	42	14	0,95
5	Slow	387	431	0,27	31,78	0,03146	0,17738	1	42	14	0,9
6	Slow	442	486	0,30	45,69	0,02189	0,147945	1	42	14	0,9
7	Slow	1482	1526	1,02	33,23	0,03967	0,248112	1	42	14	0,9

Empirical distributions

Item 1 - Erratic			Item 2 - Erratic		
Demand	**Observations**	**Probability**	**Demand**	**Observations**	**Probability**
1	2	15,4%	1	2	7,1%
2	1	7,7%	2	25	89,3%
3	1	7,7%	3	1	3,6%
4	2	15,4%			
5	1	7,7%			
6	1	7,7%			
7	2	15,4%			
8	3	23,1%			
Total	**13**	**100,0%**	**Total**	**28**	**100,0%**

Item 3 - Fast			Item 4 - Fast		
Demand	**Observations**	**Probability**	**Demand**	**Observations**	**Probability**
1	14	10,8%	1	4	8,00%
2	7	5,4%	2	46	92,00%
3	31	23,8%			
4	74	56,9%			
5	4	3,1%			
Total	**130**	**100,0%**	**Total**	**50**	**100,00%**

Item 5 - Slow			Item 6 - Slow		
Demand	**Observations**	**Probability**	**Demand**	**Observations**	**Probability**
1	23	100%	1	16	100,0%
Total	**23**	**100%**	**Total**	**16**	**100,0%**

Item 7 - Slow		
Demand	**Observations**	**Probability**
1	16	72,727%
2	5	22,727%
3	1	4,545%
Total	**22**	**100,0%**

Appendix III

Translation between G and K

G	K	G	K	G	K	G	K
0,3990	0,0	0,1400	0,7	0,0370	1,4	0,0070	2,1
0,3500	0,1	0,1200	0,8	0,0290	1,5	0,0050	2,2
0,3070	0,2	0,1000	0,9	0,0230	1,6	0,0040	2,3
0,2700	0,3	0,0830	1,0	0,0180	1,7	0,0030	2,4
0,2300	0,4	0,0700	1,1	0,0140	1,8	0,0020	2,5
0,2000	0,5	0,0560	1,2	0,0110	1,9	0,0010	2,6
0,1790	0,6	0,0500	1,3	0,0090	2,0	0,0010	2,7

Appendix IV

Simulation results

Slow items

Base case. Emergency lead time = 14 days. Holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters						Simulation data		Fill rate		Ready rate		Order line fill rate		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	S1	S2	B1	ROL	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
5	(R, Q)	3	1	N/A	N/A	5,59	N/A	1000000	25	0,955	0,00141	0,955	0,00093	0,955	0,00141	12986038	64838	31466	168	412,70	N/A	N/A
5	S&Z	N/A	N/A	3	0	5,59	N/A	1000000	25	0,863	0,00210	0,863	0,00199	0,863	0,00210	12937697	59922	31460	144	411,25	0,048	0,0010
5	(R, Q)-RH	3	1	N/A	N/A	5,59	0	1000000	25	0,971	0,00169	0,971	0,00085	0,971	0,00169	13032935	74579	31445	194	414,47	0,056	0,0014
5	(R, Q)	4	1	N/A	N/A	23,04	N/A	1000000	25	0,989	0,00010	0,989	0,00064	0,989	0,00010	13214639	72626	31401	187	420,83	N/A	N/A
5	S&Z	N/A	N/A	4	1	23,04	N/A	1000000	25	0,966	0,00118	0,967	0,00062	0,966	0,00118	13149746	74247	31511	185	417,31	0,049	0,0012
5	(R, Q)-RH	3	1	N/A	N/A	23,04	0	1000000	25	0,971	0,00126	0,970	0,00063	0,971	0,00126	13114876	84211	31407	212	417,58	0,055	0,0012
5	(R, Q)	4	1	N/A	N/A	0,00	N/A	1000000	25	0,989	0,00082	0,989	0,00055	0,989	0,00082	13170043	81381	31421	215	419,15	N/A	N/A
5	S&Z	N/A	N/A	4	1	0,00	N/A	1000000	25	0,967	0,00117	0,967	0,00064	0,967	0,00117	12969095	62320	31454	164	412,32	0,048	0,0011
5	(R, Q)-RH	3	1	N/A	N/A	0,00	-0,43	1000000	25	0,967	0,00108	0,967	0,00060	0,967	0,00108	12997601	57604	31498	152	412,65	0,043	0,0010
Item nbr	Ordering Policy	Policy parameters						Simulation data		Fill rate		Ready rate		Order line fill rate		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	S1	S2	B1	ROL	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std
6	(R, Q)	2	1	N/A	N/A	4,28	N/A	1000000	25	0,933	0,00222	0,933	0,00121	0,933	0,00222	10407353	56810	21923	125	474,72	N/A	N/A
6	S&Z	N/A	N/A	2	0	4,28	N/A	1000000	25	0,783	0,00289	0,785	0,00184	0,783	0,00289	10358128	58549	21895	123	473,08	0,105	0,0023
6	(R, Q)-RH	2	1	N/A	N/A	4,28	0	1000000	25	0,959	0,00165	0,959	0,00066	0,959	0,00165	10465752	67673	21869	150	478,57	0,094	0,0019
6	(R, Q)	3	1	N/A	N/A	20,70	N/A	1000000	25	0,986	0,00123	0,986	0,00057	0,986	0,00123	10654718	67418	21842	13	487,81	N/A	N/A
6	S&Z	N/A	N/A	3	1	20,70	N/A	1000000	25	0,957	0,00179	0,957	0,00089	0,957	0,00089	10535541	75845	21845	168	482,28	0,104	0,0022
6	(R, Q)-RH	2	1	N/A	N/A	20,70	0	1000000	25	0,959	0,00135	0,959	0,00086	0,959	0,00135	10590806	78243	21936	170	482,80	0,094	0,0018
6	(R, Q)	3	1	N/A	N/A	0,00	N/A	1000000	25	0,986	0,00100	0,986	0,00060	0,986	0,00103	10598268	50780	21856	118	484,92	N/A	N/A
6	S&Z	N/A	N/A	3	1	0,00	N/A	1000000	25	0,959	0,00142	0,959	0,00060	0,959	0,00142	10461026	64072	21917	144	477,31	0,094	0,0018
6	(R, Q)-RH	2	1	N/A	N/A	0,00	0	1000000	25	0,958	0,00190	0,957	0,00076	0,958	0,00190	10446301	67116	21923	151	476,49	0,104	0,0018
Item nbr	Ordering Policy	Policy parameters						Simulation data		Fill rate		Ready rate		Order line fill rate		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	S1	S2	B1	ROL	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std
7	(R, Q)	3	1	N/A	N/A	10,50	N/A	2000000	25	0,832	0,00198	0,870	0,00148	0,826	0,00200	124772307	452977	79340	296	1572,63	N/A	N/A
7	S&Z	N/A	N/A	3	1	10,50	N/A	2000000	25	0,783	0,00202	0,855	0,00087	0,772	0,00087	124066998	445078	79410	286	1562,36	0,319	0,0006
7	(R, Q)-RH	3	1	N/A	N/A	10,50	0	2000000	25	0,908	0,00125	0,935	0,00082	0,903	0,00137	125762816	570228	79281	378	1586,29	0,272	0,0014
7	(R, Q)	4	1	N/A	N/A	35,88	N/A	2000000	25	0,918	0,00158	0,939	0,00092	0,914	0,00184	127758037	548099	79423	143	1608,58	N/A	N/A
7	S&Z	N/A	N/A	4	1	35,88	N/A	2000000	25	0,877	0,00166	0,919	0,00098	0,870	0,00165	126210158	645772	79215	407	1593,26	0,158	0,0018
7	(R, Q)-RH	3	1	N/A	N/A	35,88	0	2000000	25	0,907	0,00145	0,935	0,00065	0,903	0,00149	127695511	527508	79398	319	1608,30	0,272	0,0015
7	(R, Q)	4	1	N/A	N/A	0,00	N/A	2000000	25	0,919	0,00163	0,939	0,00097	0,915	0,00166	124574348	621650	79393	431	1569,08	N/A	N/A
7	S&Z	N/A	N/A	4	1	0,00	N/A	2000000	25	0,876	0,00015	0,919	0,00070	0,870	0,00147	123410060	453508	79375	309	1554,77	0,158	0,0015
7	(R, Q)-RH	3	1	N/A	N/A	0,00	-1	2000000	25	0,875	0,00166	0,906	0,00089	0,870	0,00159	124013921	482026	79311	326	1563,65	0,161	0,0012

Sensitivity analysis: emergency lead time = 2 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters						Simulation data		Fill rate		Ready rate		Order line fill rate		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	S1	S2	B1	ROL	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std
5	(R, Q)	3	1	N/A	N/A	5,59	N/A	1000000	25	0,955	0,0014	0,955	0,00093	0,955	0,00141	12986038	64838	31466	168	412,70	N/A	N/A
5	S&Z	N/A	N/A	3	0	5,59	N/A	1000000	25	0,886	0,0021	0,887	0,00130	0,886	0,00210	12840646	56373	31468	140	408,05	0,099	0,0019
5	(R, Q)-RH	3	1	N/A	N/A	5,59	N/A	1000000	25	0,980	0,0008	0,980	0,00050	0,980	0,00080	13273392	73396	31413	185	422,54	0,186	0,0021
5	(R, Q)	4	1	N/A	N/A	23,04	N/A	1000000	25	0,989	0,0008	0,989	0,00050	0,989	0,00080	13235901	83927	31453	217	420,81	N/A	N/A
5	S&Z	N/A	N/A	3	0	23,04	N/A	1000000	25	0,886	0,0018	0,886	0,00150	0,886	0,00180	12978655	72051	31523	179	411,72	0,099	0,0017
5	(R, Q)-RH	4	1	N/A	N/A	23,04	N/A	1000000	25	0,993	0,0005	0,993	0,00030	0,993	0,00050	13321023	61611	31442	158	423,67	0,073	0,0019
5	(R, Q)	4	1	N/A	N/A	0,00	N/A	500000	25	0,989	0,0010	0,989	0,00060	0,989	0,00100	13147723	63520	31399	169	418,73	N/A	N/A
5	S&Z	N/A	N/A	3	0	0,00	N/A	500000	25	0,887	0,0021	0,887	0,00130	0,887	0,00210	12799800	61268	31461	158	406,84	0,098	0,0018
5	(R, Q)-RH	3	1	N/A	N/A	0,00	-1,322	500000	25	0,879	0,0022	0,879	0,00140	0,879	0,00220	12820060	83773	31447	211	407,67	0,091	0,0025

(R, Q) vs. (R, Q)-RH: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters						Simulation data		Fill rate		Ready rate		Order line fill rate		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	S1	S2	B1	ROL	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std
5	(R, Q)	3	1	N/A	N/A	0,00	N/A	1000000	25	0,955	0,0019	0,955	0,00100	0,955	0,00192	12894043	69944	31434	186	410,19	N/A	N/A
5	(R, Q)-RH	2	1	N/A	N/A	0,00	-1,6	1000000	25	0,956	0,0016	0,956	0,00090	0,956	0,00156	12957285	51650	31501	135	411,32	0,020	0,0009
6	(R, Q)	3	1	N/A	N/A	0,00	N/A	1000000	25	0,986	0,0010	0,986	0,00060	0,986	0,00103	10598268	50780	21857	140	484,92	N/A	N/A
6	(R, Q)-RH	2	1	N/A	N/A	0,00	1,05	1000000	25	0,986	0,0010	0,986	0,00030	0,986	0,00102	11096477	58551	21857	171	507,60	0,509	0,0008
7	(R, Q)	3	1	N/A	N/A	0,00	N/A	2000000	25	0,833	0,0021	0,871	0,00140	0,826	0,00232	122381727	534925	79201	370	1545,20	N/A	N/A
7	(R, Q)-RH	2	1	N/A	N/A	0,00	-0,66	2000000	25	0,842	0,0014	0,881	0,00070	0,836	0,00144	124043449	352282	79176	233	1566,68	0,382	0,0012

Fast and erratic items

Item 1 - base case: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	11	2	N/A	N/A	N/A	1000000	25	0,8587	0,00257	0,9305	0,00125	0,8020	0,003	139361477	941974	87440	648	1593,80	N/A	N/A
1	(R, Q)-RH	11	2	0	N/A	N/A	1000000	25	0,9029	0,00190	0,9566	0,00085	0,8604	0,002	142826571	1052190	87518	698	1631,97	0,1920	0,0022
1	(R, Q)	13	2	N/A	N/A	N/A	1000000	25	0,9124	0,00250	0,9550	0,00120	0,8708	0,000	141328035	1001063	87461	691	1615,90	N/A	N/A
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)-SH	11	2	N/A	94%	Infinity	1000000	25	0,9238	0,00250	0,9687	0,00064	0,8860	0,003	144068433	1082209	87358	705	1649,18	0,28	0,00
1	(R, Q)-LH	11	2	N/A	94%	Infinity	1000000	25	0,9684	0,00140	0,9898	0,00036	0,9487	0,002	148575376	1000958	87488	656	1698,24	0,52	0,00
1	(R, Q)-SH	11	2	N/A	50%	Infinity	1000000	25	0,8932	0,00260	0,9511	0,00105	0,8468	0,003	142407712	1329656	87794	880	1622,07	0,15	0,00
1	(R, Q)-LH	11	2	N/A	50%	Infinity	1000000	25	0,9030	0,00240	0,9565	0,00077	0,8605	0,003	142710456	1339931	87444	897	1632,03	0,19	0,00
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)-SH	11	2	N/A	94%	951,47	1000000	25	0,8867	0,00290	0,9486	0,00130	0,8383	0,003	141201552	1005377	87459	687	1614,49	0,11	0,00
1	(R, Q)-LH	11	2	N/A	94%	951,47	1000000	25	0,8680	0,00320	0,9369	0,00132	0,8136	0,003	140218640	1072789	87603	735	1600,62	0,04	0,00
1	(R, Q)-SH	11	2	N/A	50%	951,47	1000000	25	0,8715	0,00260	0,9392	0,00121	0,8185	0,003	140088679	926745	87421	629	1602,47	0,05	0,00
1	(R, Q)-LH	11	2	N/A	50%	951,47	1000000	25	0,8625	0,00250	0,9332	0,00132	0,8063	0,003	139830523	1338604	87625	922	1595,78	0,01	0,00
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)-LS	3	3	N/A	N/A	N/A	1000000	25	0,495	0,00226	0,6943	0,00211	0,3410	0,004	135831902	903720	87497	593	1552,42	0,3055	0,0027

Item 2 - base case: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
2	(R, Q)	8	2	N/A	N/A	N/A	1000000	25	0,9629	0,0011	0,9754	0,0006	0,9517	0,0014	21074946	102476	75277	397	279,97	N/A	N/A
2	(R, Q)-RH	8	2	0	N/A	N/A	1000000	25	0,9743	0,0011	0,9828	0,0006	0,9668	0,0013	21499305	106130	75240	393	285,74	0,1009	0,0016
2	(R, Q)	9	2	N/A	N/A	N/A	1000000	25	0,9801	0,0010	0,9860	0,0006	0,9750	0,0012	21252233	109376	75276	426	282,32	N/A	N/A
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
2	(R, Q)-SH	8	2	N/A	97%	Infinity	1000000	25	0,9812	0,00090	0,9873	0,00050	0,9756	0,001	21598779	95112	75124	350	287,51	0,1322	0,0018
2	(R, Q)-LH	8	2	N/A	97%	Infinity	1000000	25	0,994	0,00050	0,9966	0,00020	0,9917	0,001	22630647	101810	75080	367	301,42	0,3842	0,0011
2	(R, Q)-SH	8	2	N/A	50%	Infinity	1000000	25	0,9706	0,00100	0,9802	0,00060	0,9619	0,001	21304218	115527	75297	436	282,94	0,0518	0,0014
2	(R, Q)-LH	8	2	N/A	50%	Infinity	1000000	25	0,9744	0,00080	0,9828	0,00050	0,9668	0,001	21501338	86634	75250	332	285,73	0,1005	0,0016
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
2	(R, Q)-SH	8	2	N/A	97%	388,64	1000000	25	0,9753	0,00090	0,9852	0,00040	0,9667	0,001	21548858	99122	75279	363	286,25	0,1135	0,0017
2	(R, Q)-LH	8	2	N/A	97%	388,64	1000000	25	0,9823	0,00070	0,9893	0,00040	0,9761	0,001	21913073	119330	75278	440	291,10	0,1985	0,0021
2	(R, Q)-SH	8	2	N/A	50%	388,64	1000000	25	0,9708	0,00130	0,9804	0,00060	0,9622	0,002	21262682	96123	75177	356	282,84	0,0496	0,0016
2	(R, Q)-LH	8	2	N/A	50%	388,64	1000000	25	0,9738	0,00120	0,9823	0,00050	0,9661	0,001	21449398	109786	75337	409	284,71	0,0839	0,0014
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
2	(R, Q)-LS	5	5	N/A	N/A		1000000	25	0,9154	0,00141	0,9342	0,00084	0,8986	0,001	21067650	113468	75248	424	279,98	0,0658	0,0014

Item 3 - base case: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9326	0,00410	0,9468	0,00270	0,9204	0,0046	74256488	366815	119845	620	619,61	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9669	0,00160	0,9763	0,00090	0,9587	0,0017	75037976	357422	119626	583	627,27	0,1099	0,0024
3	(R, Q)	42	8	N/A	N/A	N/A	200000	25	0,9668	0,00270	0,9746	0,00180	0,9600	0,0030	74414112	388598	119561	660	622,40	N/A	N/A

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-SH	38	8	N/A	95%	Infinity	200000	25	0,9817	0,00130	0,9875	0,00080	0,9763	0,002	75601778	454591	119660	718	631,80	0,1738	0,0033
3	(R, Q)-LH	38	8	N/A	95%	Infinity	200000	25	0,9973	0,00050	0,9984	0,00030	0,9964	0,001	78203869	360348	119696	571	653,35	0,4787	0,0016
3	(R, Q)-SH	38	8	N/A	50%	Infinity	200000	25	0,9521	0,00280	0,9646	0,00180	0,9418	0,003	74591709	418661	119714	692	623,08	0,0501	0,0016
3	(R, Q)-LH	38	8	N/A	50%	Infinity	200000	25	0,9667	0,00150	0,9761	0,00110	0,9586	0,002	74898320	447066	119502	735	626,75	0,1021	0,0020

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-SH	38	8	N/A	95%	2108,51	200000	25	0,9795	0,00160	0,9860	0,00090	0,9739	0,002	75539105	417243	119672	667	631,22	0,1651	0,0028
3	(R, Q)-LH	38	8	N/A	95%	2108,51	200000	25	0,9938	0,00070	0,9960	0,00040	0,9918	0,001	77181748	322137	119827	510	644,11	0,3431	0,0025
3	(R, Q)-SH	38	8	N/A	50%	2108,51	200000	25	0,9531	0,00230	0,9653	0,00140	0,9427	0,003	74520574	419753	119602	690	623,07	0,0497	0,0021
3	(R, Q)-LH	38	8	N/A	50%	2108,51	200000	25	0,9662	0,00200	0,9755	0,00120	0,9578	0,002	74899617	414270	119566	679	626,43	0,0979	0,0019

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-LS	29	29	N/A	N/A	N/A	200000	25	0,9264	0,00196	0,9399	0,00169	0,9155	0,002	74830973	429321	119674	701	625,29	0,0603	0,0018

Item 4 - base case: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
4	(R, Q)	11	2	N/A	N/A	N/A	500000	25	0,9496	0,00230	0,9599	0,00140	0,9403	0,002	191950606	1020291	65796	374	2917,37	N/A	N/A
4	(R, Q)-RH	11	2	0	N/A	N/A	500000	25	0,9656	0,00170	0,9740	0,00120	0,9578	0,002	194518232	1235528	65792	429	2956,56	0,1094	0,0029
4	(R, Q)	12	2	N/A	N/A	N/A	500000	25	0,9693	0,00160	0,9776	0,00120	0,9617	0,002	192279650	1028305	65570	376	2932,42	N/A	N/A
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
4	(R, Q)-SH	11	2	N/A	95%	0	500000	25	0,9759	0,00110	0,9847	0,00050	0,9678	0,001	194604983	921602	65637	331	2964,89	0,1249	0,0017
4	(R, Q)-LH	11	2	N/A	95%	0	500000	25	0,9934	0,00050	0,9960	0,00020	0,9912	0,001	201822868	509935	65904	181	3062,38	0,3926	0,0017
4	(R, Q)-SH	11	2	N/A	50%	0	500000	25	0,9567	0,00200	0,9654	0,00140	0,9488	0,002	192251256	930356	65607	333	2930,35	0,0370	0,0013
4	(R, Q)-LH	11	2	N/A	50%	0	500000	25	0,9661	0,00170	0,9743	0,00090	0,9585	0,002	193970851	980249	65621	346	2955,95	0,1061	0,0015
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
4	(R, Q)-SH	11	2	N/A	95%	2601,23	500000	25	0,9718	0,00130	0,9790	0,00070	0,9648	0,002	194408856	1113895	65703	391	2958,92	0,1120	0,0024
4	(R, Q)-LH	11	2	N/A	95%	2601,23	500000	25	0,9774	0,00100	0,9839	0,00050	0,9713	0,001	194795929	905669	65659	313	2966,77	0,1317	0,0021
4	(R, Q)-SH	11	2	N/A	50%	2601,23	500000	25	0,9564	0,00150	0,9653	0,00100	0,9484	0,002	192099919	1090062	65583	391	2929,12	0,0337	0,0015
4	(R, Q)-LH	11	2	N/A	50%	2601,23	500000	25	0,9627	0,00130	0,9727	0,00100	0,9538	0,002	192899290	866803	65761	310	2933,35	0,0437	0,0012
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
4	(R, Q)-LS	7	7	N/A	N/A	N/A	500000	25	0,9034	0,00158	0,9191	0,00135	0,8891	0,002	192349758	974710	65726	348	2926,54	0,0809	0,0016

Item 3 - Sensitivity analysis: emergency lead time = 14 days & holding rate = 50%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9326	0,00330	0,9472	0,00250	0,9206	0,004	75469337	377249	119446	656	631,83	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9675	0,00190	0,9766	0,00110	0,9592	0,002	76670604	279045	119617	461	640,97	0,1099	0,0022
3	(R, Q)	43	8	N/A	N/A	N/A	200000	25	0,9723	0,00230	0,9790	0,00130	0,9665	0,002	76365862	289518	119607	504	638,47	N/A	N/A

Item 3 - Sensitivity analysis: emergency lead time = 14 days & holding rate = 100%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9334	0,00300	0,9477	0,00220	0,9213	0,003	78474898	328088	119582	608	656,24	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9667	0,00220	0,9761	0,00150	0,9585	0,002	80128299	381349	119865	647	668,49	0,1105	0,0024
3	(R, Q)	42	8	0	N/A	N/A	200000	25	0,9667	0,00240	0,9744	0,00160	0,9599	0,003	79870299	361489	119794	668	666,73	N/A	N/A
3	(R, Q)	43	8	N/A	N/A	N/A	200000	25	0,9727	0,00240	0,9793	0,00160	0,9670	0,003	79986293	368181	119402	683	669,89	N/A	N/A

Item 3 - Sensitivity analysis: emergency lead time = 14 days. Holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9326	0,00410	0,9468	0,0027	0,9204	0,005	74256488	366815	119845	620	619,61	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9888	0,0006	0,9941	0,0003	0,9839	0,001	74998457	428307	119319	690	628,55	0,1214	0,0024
3	(R, Q)	48	8	N/A	N/A	N/A	200000	25	0,9898	0,0012	0,9927	0,0008	0,9875	0,001	74889774	375075	119563	635	626,36	N/A	N/A

Item 3 - Sensitivity analysis: emergency lead time = 2 days & holding rate = 100%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9334	0,0030	0,9477	0,0022	0,9213	0,003	78474898	328088	119582	608	656,24	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9884	0,0009	0,9938	0,0003	0,9834	0,001	80435479	273623	119659	488	672,21	0,1217	0,0018
3	(R, Q)	46	8	N/A	N/A	N/A	200000	25	0,9851	0,0018	0,9889	0,0012	0,9817	0,002	81072393	361430	119595	670	677,89	N/A	N/A
3	(R, Q)	47	8	0	N/A	N/A	200000	25	0,9875	0,0014	0,9908	0,0010	0,9847	0,002	81324459	393449	119453	731	680,80	N/A	N/A
3	(R, Q)	48	8	N/A	N/A	N/A	200000	25	0,9896	0,0014	0,9924	0,0009	0,9872	0,002	81723969	371236	119580	688	683,43	N/A	N/A
3	(R, Q)	1	2	0	N/A	N/A	200000	25	0,4953	0,0014	0,7342	0,0015	0,2006	0,002	79797874	446517	119722	677	666,53	1,0000	1,0000
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-SH	38	8	N/A	95%	infinity	200000	25	0,9805	0,00100	0,9909	0,00050	0,9722	0,001	79725393	452177	119627	756	666,45	0,0781	0,0031
3	(R, Q)-LH	38	8	N/A	95%	infinity	200000	25	0,9996	0,00010	0,9998	0,00004	0,9993	0,000	85069732	381205	119314	644	712,99	0,3975	0,0034
3	(R, Q)-SH	38	8	N/A	50%	infinity	200000	25	0,9674	0,00120	0,9782	0,00070	0,9578	0,002	79208194	315625	119500	546	662,83	0,0518	0,0017
3	(R, Q)-LH	38	8	N/A	50%	infinity	200000	25	0,9873	0,00070	0,9931	0,00030	0,9819	0,001	80155067	342214	119649	586	669,92	0,1046	0,0021
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-SH	38	8	N/A	95%	3860,61	200000	25	0,9806	0,00110	0,9910	0,00040	0,9724	0,001	79643940	461637	119491	787	666,53	0,0777	0,0023
3	(R, Q)-LH	38	8	N/A	95%	3860,61	200000	25	0,9994	0,00010	0,9998	0,00004	0,9991	0,000	83963223	350754	119589	579	702,10	0,3220	0,0024
3	(R, Q)-SH	38	8	N/A	50%	3860,61	200000	25	0,9669	0,00150	0,9782	0,00080	0,9571	0,002	79227538	360201	119510	624	662,94	0,0526	0,0018
3	(R, Q)-LH	38	8	N/A	50%	3860,61	200000	25	0,9873	0,00070	0,9931	0,00030	0,9818	0,001	80123678	306324	119729	522	669,21	0,1006	0,0021
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-SH	38	8	N/A	95%	482,58	200000	25	0,9657	0,00170	0,9767	0,00100	0,9567	0,002	79435466	298563	119745	526	663,37	0,0564	0,0014
3	(R, Q)-LH	38	8	N/A	95%	482,58	200000	25	0,9468	0,00260	0,9591	0,00160	0,9365	0,003	78717638	390339	119386	702	659,35	0,0224	0,0014
3	(R, Q)-SH	38	8	N/A	50%	482,58	200000	25	0,9512	0,00220	0,9638	0,00150	0,9402	0,003	78850943	358007	119535	638	659,65	0,0274	0,0011
3	(R, Q)-LH	38	8	N/A	50%	482,58	200000	25	0,9384	0,00390	0,9521	0,00270	0,9267	0,004	78643857	487002	119656	896	657,25	0,0087	0,0005
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-SH	38	8	N/A	95%	3860,61	200000	25	0,9806	0,00110	0,9910	0,00040	0,9724	0,001	79643940	461637	119491	787	666,53	0,0777	0,0023
3	(R, Q)-LH	38	8	N/A	95%	880	200000	25	0,9806	0,00120	0,9866	0,00100	0,9752	0,001	80267950	260975	119836	441	669,81	0,1038	0,0019
3	(R, Q)-RH	38	8	-2,3	N/A	N/A	200000	25	0,9792	0,00090	0,9874	0,00050	0,9718	0,001	79787321	383925	119592	658	667,16	0,0847	0,0019
3	(R, Q)-RH	38	8	-2	N/A	N/A	200000	25	0,9810	0,00120	0,9887	0,00040	0,9741	0,001	79809855	370015	119441	644	668,20	0,0910	0,0017

Item 3 - Sensitivity analysis: emergency lead time = 2 days & holding rate = 50%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9326	0,00330	0,9472	0,0025	0,9206	0,004	75469337	377249	119446	656	631,83	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9887	0,0008	0,9939	0,0003	0,9837	0,001	76970912	458222	119692	743	643,08	0,1221	0,0030
3	(R, Q)	48	8	N/A	N/A	N/A	200000	25	0,9897	0,0014	0,9925	0,0010	0,9874	0,002	77126798	379953	119503	664	645,40	N/A	N/A

Item 3 - Sensitivity analysis: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9669	0,00160	0,9763	0,00090	0,9587	0,002	75037976	357422	119626	583	627,27	0,1099	0,0024
3	(R, Q)	42	8	N/A	N/A	N/A	200000	25	0,9668	0,00270	0,9746	0,00180	0,9600	0,003	74414112	388598	119561	660	622,40	N/A	N/A
3	(R, Q)-SH	38	8	N/A	95%	1000	200000	25	0,9666	0,00180	0,9759	0,00120	0,9585	0,002	75029976	475306	119665	766	627,00	0,1041	0,0026
3	(R, Q)-LH	38	8	N/A	95%	1050	200000	25	0,9667	0,00220	0,9759	0,00140	0,9590	0,003	75207700	327592	119779	525	627,89	0,1185	0,0024

Item 1 - Sensitivity analysis: emergency lead time = 14 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	13	2	N/A	N/A	N/A	1000000	25	0,9124	0,0025	0,9550	0,0012	0,8708	0,000	141328035	1001063	87461	691	1615,90	N/A	N/A
1	(R, Q)-RH	11	2	1,3	N/A	N/A	1000000	25	0,9115	0,0023	0,9616	0,0008	0,8709	0,000	143397446	1022139	87493	687	1638,96	0,2262	0,0022
1	(R, Q)-SH	11	2	N/A	94%	1520	1000000	25	0,9126	0,0023	0,9628	0,0009	0,8723	0,000	143037572	1258073	87139	836	1641,48	0,2343	0,0025
1	(R, Q)-LH	11	2	N/A	94%	1535	1000000	25	0,9125	0,0023	0,9644	0,0007	0,8701	0,000	143279614	839545	87513	557	1637,24	0,2228	0,0028

Item 1 - Sensitivity analysis: emergency lead time = 2 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	11	2	N/A	N/A	N/A	1000000	25	0,8587	0,0026	0,9310	0,0013	0,8020	0,003	139361477	941974	87440	648	1593,80	N/A	N/A
1	(R, Q)-RH	11	2	0	N/A	N/A	1000000	25	0,9137	0,0021	0,9642	0,0008	0,8748	0,002	143016508	1084482	87319	728	1637,87	0,2080	0,0025
1	(R, Q)	14	2	N/A	N/A	N/A	1000000	25	0,9318	0,0025	0,9660	0,0012	0,9050	0,003	142552299	1191433	87625	822	1626,84	N/A	N/A

Item 1 - Sensitivity analysis: emergency lead time = 14 days & holding rate = 50%.

Item nbr	**Ordering Policy**	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	11	2	N/A	N/A	N/A	1000000	25	0,8591	0,0032	0,9308	0,0014	0,8020	0,004	148313410	1315163	87242	929	1700,03	N/A	N/A
1	(R, Q)-RH	11	2	0	N/A	N/A	1000000	25	0,9036	0,0019	0,9567	0,0006	0,8612	0,002	153515022	911685	87273	648	1759,02	0,1912	0,0024
1	(R, Q)	13	2	N/A	N/A	N/A	1000000	25	0,9119	0,0028	0,9546	0,0013	0,8700	0,003	152483226	1197659	87451	847	1743,65	N/A	N/A
1	(R, Q)-SH	11	2	N/A	94%	1724,14	1000000	25	0,9115	0,0024	0,9620	0,0007	0,8704	0,003	154857752	1284754	87496	850	1769,89	0,2290	0,0029
1	(R, Q)-LH	11	2	N/A	94%	1851,85	1000000	25	0,9112	0,0025	0,9638	0,0009	0,8677	0,003	154174460	1032294	87595	698	1760,08	0,2054	0,0031
1	(R, Q)-RH	12	2	-1,9	N/A	N/A	1000000	25	0,9104	0,0017	0,9579	0,0007	0,8676	0,002	153654011	938908	87520	638	1755,65	0,1134	0,0021

Item 1 - Sensitivity analysis: emergency lead time = 2 days & holding rate = 100%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	11	2	N/A	N/A	N/A	1000000	25	0,8577	0,0029	0,9305	0,0012	0,8009	0,004	167175729	854463	87523	632	1910,07	N/A	N/A
1	(R, Q)-RH	11	2	0	N/A	N/A	1000000	25	0,9141	0,0027	0,9639	0,0010	0,8747	0,003	177554227	1364379	87539	864	2028,29	0,2081	0,0033
1	(R, Q)	13	2	N/A	N/A	N/A	1000000	25	0,9110	0,0032	0,9546	0,0013	0,8693	0,004	175138156	1062153	87686	802	1997,34	N/A	N/A
1	(R, Q)	14	2	N/A	N/A	N/A	1000000	25	0,9308	0,0023	0,9660	0,0009	0,9046	0,003	179065905	878175	87677	659	2042,34	N/A	N/A
1	(R, Q)	12	2	N/A	N/A	N/A	1000000	25	0,8880	0,0030	0,9440	0,0011	0,8354	0,003	170968207	844155	87480	635	1954,37	N/A	N/A
1	(R, Q)-SH	11	2	N/A	94%	1063,83	1000000	25	0,8873	0,0025	0,9498	0,0011	0,8383	0,003	171182856	943575	87608	665	1953,96	0,0912	0,0023
1	(R, Q)-LH	11	2	N/A	94%	1492,54	1000000	25	0,8881	0,0022	0,9508	0,0009	0,8378	0,003	172101588	839073	87615	567	1964,30	0,0983	0,0028
1	(R, Q)-RH	11	2	-3,7	N/A	N/A	1000000	25	0,8874	0,0026	0,9473	0,0010	0,8402	0,003	172524832	1080270	87699	693	1967,24	0,1058	0,0029

Item 1 - Sensitivity analysis: emergency lead time = 2 days, holding cost = 100% and air freight rate reduced 50%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	11	2	N/A	N/A	N/A	1000000	25	0,8589	0,0031	0,9309	0,0014	0,8034	0,004	166959745	823235	87366	610	1911,04	N/A	N/A
1	(R, Q)-RH	11	2	0	N/A	N/A	1000000	25	0,9144	0,0020	0,9639	0,0009	0,8751	0,002	176652646	1247575	87476	833	2019,43	0,2077	0,0031
1	(R, Q)	13	2	N/A	N/A	N/A	1000000	25	0,9117	0,0023	0,9546	0,0011	0,8702	0,003	174881397	871908	87493	663	1998,80	N/A	N/A
1	(R, Q)	14	2	N/A	N/A	N/A	1000000	25	0,9312	0,0027	0,9662	0,0012	0,9052	0,003	178668507	1010666	87375	758	2044,84	N/A	N/A
1	(R, Q)	12	2	N/A	N/A	N/A	1000000	25	0,8873	0,0033	0,9439	0,0016	0,8348	0,004	170979292	1169321	87496	877	1954,15	N/A	N/A
1	(R, Q)-SH	11	2	N/A	94%	813,01	1000000	25	0,8875	0,0030	0,9502	0,0010	0,8379	0,003	170675999	880008	87381	615	1953,24	0,0917	0,0024
1	(R, Q)-LH	11	2	N/A	94%	1010,10	1000000	25	0,8878	0,0025	0,9503	0,0013	0,8380	0,003	171706094	1017781	87533	725	1961,62	0,0985	0,0034
1	(R, Q)-RH	11	2	-3,7	N/A	N/A	1000000	25	0,8878	0,0024	0,9468	0,0012	0,8401	0,003	172278535	1167396	87541	821	1967,98	0,1053	0,0024

Results for lost sales comparison: emergency lead time = 2 days & holding rate = 25%.

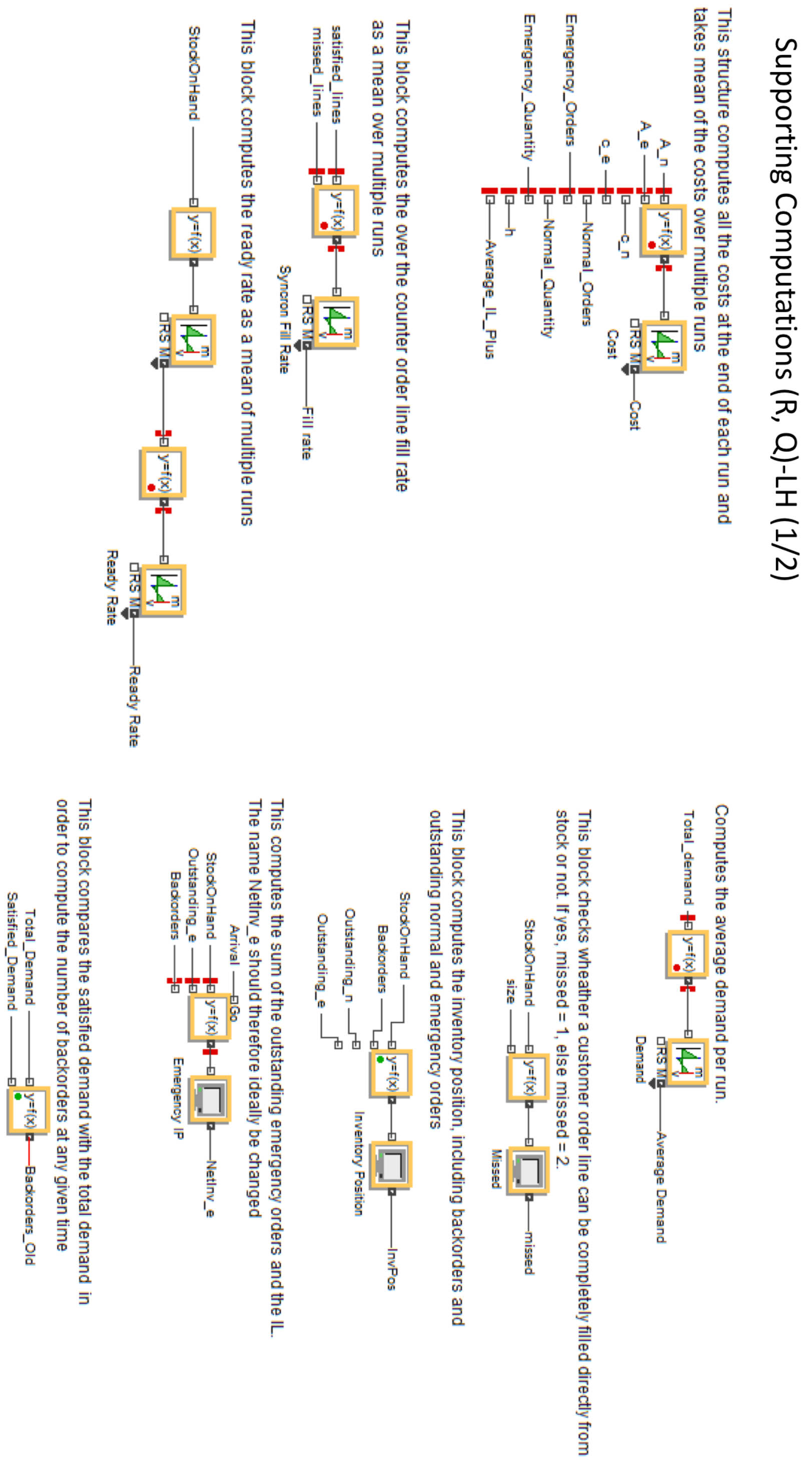
Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
1	(R, Q)	11	2	N/A	N/A	N/A	1000000	25	0,8587	0,0026	0,9310	0,0013	0,8020	0,003	139361477	941974	87440	648	1593,80	N/A	N/A
1	(R, Q)-RH	11	2	0	N/A	N/A	1000000	25	0,9137	0,0021	0,9642	0,0008	0,8748	0,002	143016508	1084482	87319	728	1637,87	0,2080	0,0025
1	(R, Q)-LS	3	3	N/A	N/A	N/A	1000000	25	0,4950	0,0023	0,6943	0,0021	0,3410	0,004	135831902	903720	87497	593	1552,42	0,3055	0,0027
2	(R, Q)	8	2	N/A	N/A	N/A	1000000	25	0,9629	0,0011	0,9754	0,0006	0,9517	0,001	21074946	102476	75277	397	279,97	N/A	N/A
2	(R, Q)-RH	8	2	0	N/A	N/A	1000000	25	0,9801	0,0007	0,9888	0,0004	0,9725	0,001	21581569	112328	75309	407	286,57	0,1137	0,0019
2	(R, Q)-LS	5	5	N/A	N/A	N/A	1000000	25	0,9154	0,0014	0,9342	0,0008	0,8986	0,001	21067650	113468	75248	424	279,98	0,0658	0,0014
3	(R, Q)	38	8	N/A	N/A	N/A	200000	25	0,9326	0,0041	0,9468	0,0027	0,9204	0,005	74256488	366815	119845	620	619,61	N/A	N/A
3	(R, Q)-RH	38	8	0	N/A	N/A	200000	25	0,9888	0,0006	0,9941	0,0003	0,9839	0,001	74998457	428307	119319	690	628,55	0,1214	0,0024
3	(R, Q)-LS	29	29	N/A	N/A	N/A	200000	25	0,9264	0,0020	0,9399	0,0017	0,9155	0,002	74830973	429321	119674	701	625,29	0,0603	0,0018
4	(R, Q)	11	2	N/A	N/A	N/A	500000	25	0,9496	0,0023	0,9599	0,0014	0,9403	0,002	191950606	1020291	65796	374	2917,37	N/A	N/A
4	(R, Q)-RH	11	2	0	N/A	N/A	500000	25	0,9808	0,0008	0,9880	0,0004	0,9740	0,001	194634074	832937	65659	294	2964,30	0,1229	0,0020
4	(R, Q)-LS	7	7	N/A	N/A	N/A	500000	25	0,9034	0,0016	0,9191	0,0013	0,8891	0,002	192349758	974710	65726	348	2926,54	0,0809	0,0016

For item 2 - same Q for (R, Q) and (R, Q)-RH as for (R, Q)-LS. Emergency lead time = 2 days & holding rate = 25%.

Item nbr	Ordering Policy	Policy parameters					Simulation data		Fill rate		Ready rate		Order line fill		Cost (SEK)		Demand		Cost/unit	Utilization	
		R	Q	ROL	TSL	Ratio	Length (days)	# of runs	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Mean	Std.
2	(R, Q)	7	5	0	N/A	N/A	1000000	25	0,9650	0,0016	0,9742	0,0007	0,9569	0,002	21148365	108670	75214	423	281,18	N/A	N/A
2	(R, Q)-RH	7	5	0	N/A	N/A	1000000	25	0,9795	0,0009	0,9871	0,0004	0,9728	0,001	21737825	86139	75237	309	288,92	0,1387	0,0017

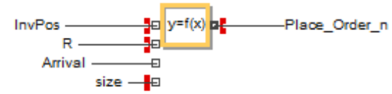
Appendix V

Extend software simulation blocks

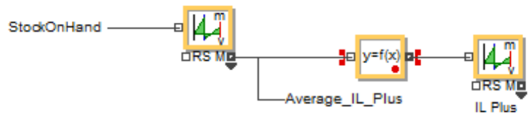


Supporting Computations (R, Q)-LH (2/2)

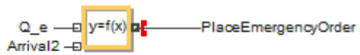
This block is basically the normal order generator. If InvPos is smaller or equal to R directly after we get an order we send a pulse to the normal order create block, which realeases an order.



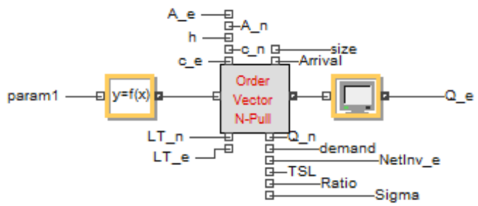
This block computes the mean stock on hand over 1(!) run



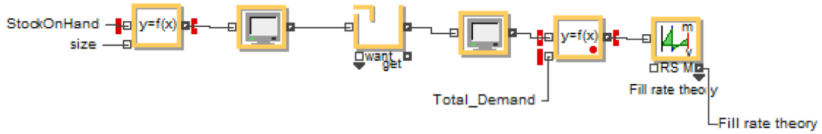
This sends a pulse to the emergency orders create block every time an emergency order should be placed. If a customer order arrives and this leads to a positive emergencyOrderQuantity from the order vector block, a pulse is sent.



This block tracks the arrival times of all outstanding normal orders and computes the emergency quantity every time it gets a signal from the update connector. This particular block computes emergency quantity based on the (R,Q)-LH Algorithm.

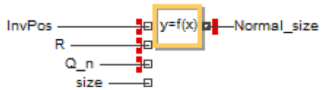


Computes theoretical fill rate, where also fractions of customer orders can be fulfilled

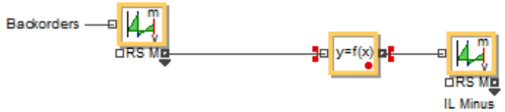


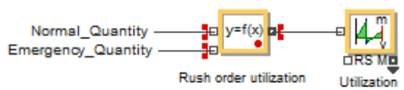
This block computes R + Q

Determines how many multiples of the order size to order

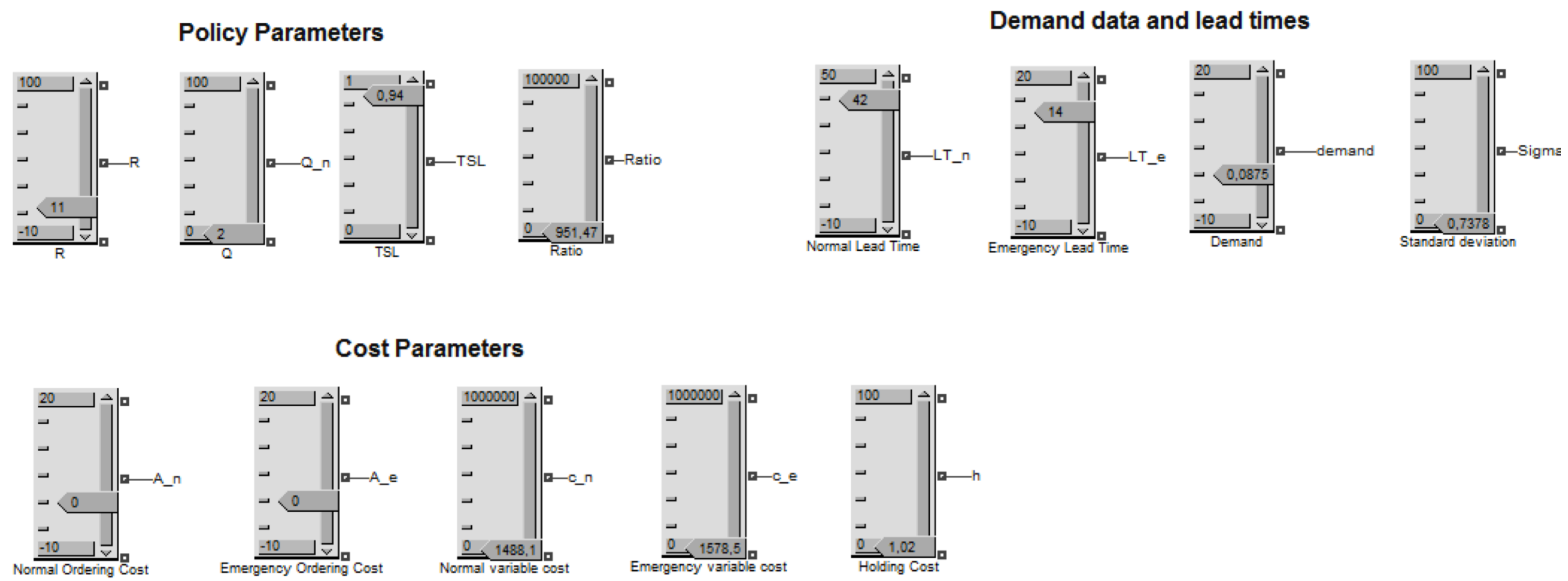


This structure computes the average IL-



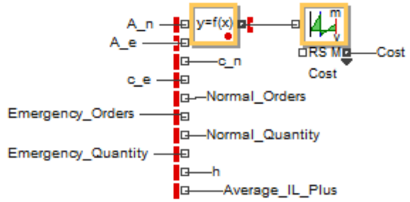


Input Parameters (R, Q)-LH

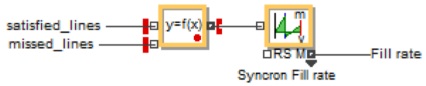


Supporting Computations (R, Q)-SH (1/2)

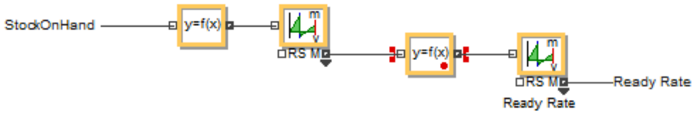
This structure computes all the costs at the end of each run and takes mean of the costs over multiple runs



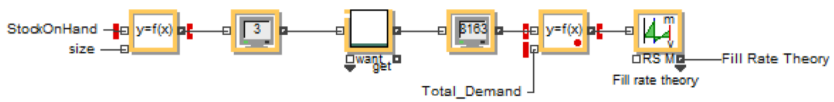
This block computes the over the counter order line fill rate as a mean over multiple runs



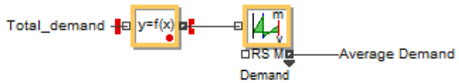
This block computes the ready rate as a mean of multiple runs



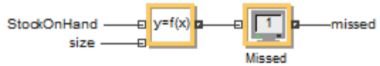
Computes theoretical fill rate, where also fractions of customer orders can be fulfilled



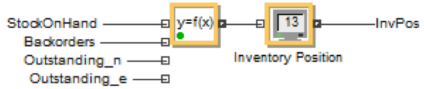
Computes the average demand per run.



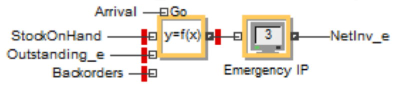
This block checks wheather a customer order line can be completely filled directly from stock or not. If yes, missed = 1, else missed = 2.



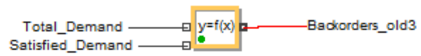
This block computes the inventory position, including backorders and outstanding normal and emergency orders



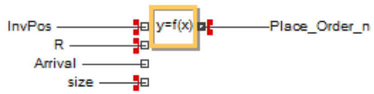
This computes the sum of the outstanding emergency orders and the IL. The name NetInv_e should therefore ideally be changed



This block compares the satisfied demand with the total demand in order to compute the number of backorders at any given time

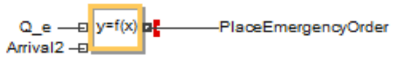


This block is basically the normal order generator. If InvPos is smaller or equal to R directly after we get an order we send a pulse to the normal order create block, which realeases an order.

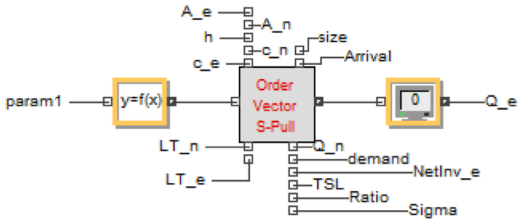


Supporting Computations (R, Q)-SH (2/2)

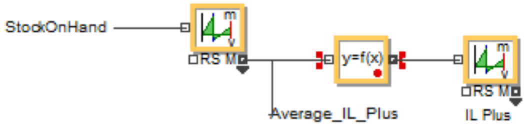
This sends a pulse to the emergency orders create block every time an emergency order should be placed. If a customer order arrives and this leads to a positive emergencyOrderQuantity from the order vector block, a pulse is sent.



This block tracks the arrival times of all outstanding normal orders and computes the emergency quantity every time it gets a signal from the update connector. This particular block computes the emergency order quantity with the (R,Q)-SH Algorithm.

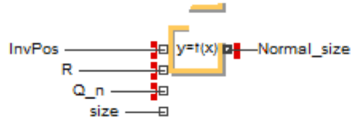


This block computes the mean stock on hand over 1(!) run

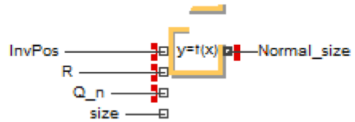


This block computes R + Q

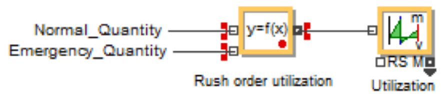
Determines how many multiples of the order size to order



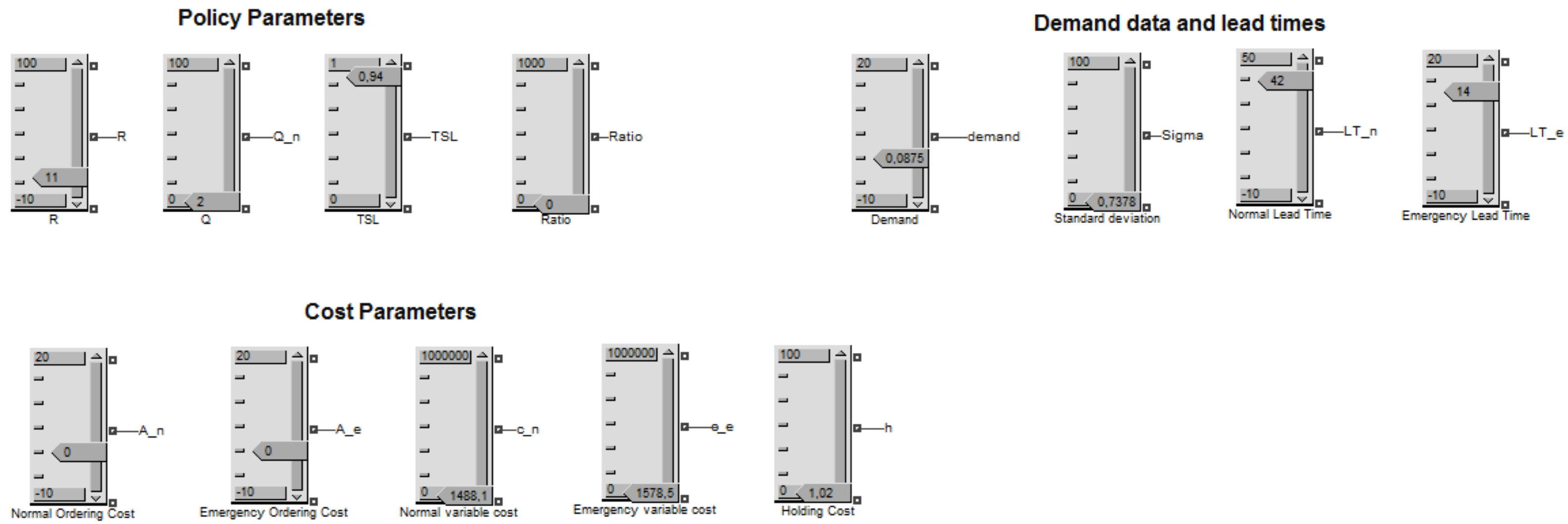
Determines how many multiples of the order size to order



This structure computes the average IL-

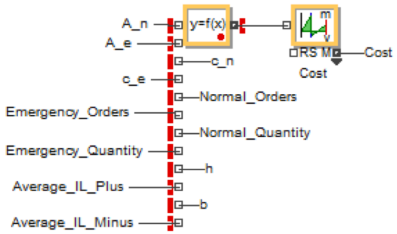


Input Parameters (R,Q)-SH

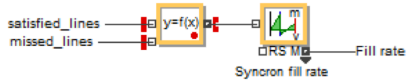


Supporting Computations (R, Q)-RH (1/2)

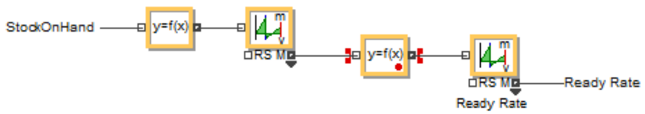
This structure computes all the costs at the end of each run and takes mean of the costs over multiple runs



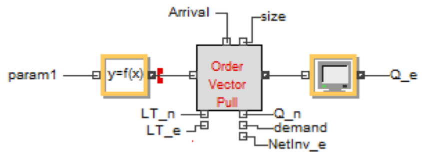
This block computes the over the counter order line fill rate as a mean over multiple runs



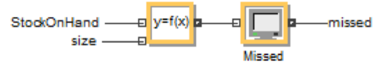
This block computes the ready rate as a mean of multiple runs



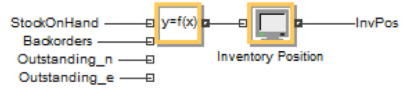
This block tracks the arrival times of all outstanding normal orders and computes the emergency quantity every time it gets a signal from the update connector. This particular block aims to emulate the GIM, rush order heuristic.



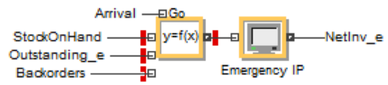
This block checks wheather a customer order line can be completely filled directly from stock or not. If yes, missed = 1, else missed = 2.



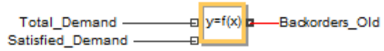
This block computes the inventory position, including backorders and outstanding normal and emergency orders



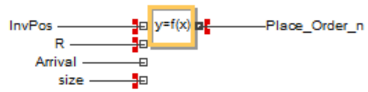
This computes the sum of the outstanding emergency orders and the IL. The name NetInv_e should therefore ideally be changed



This block compares the satisfied demand with the total demand in order to compute the number of backorders at any given time

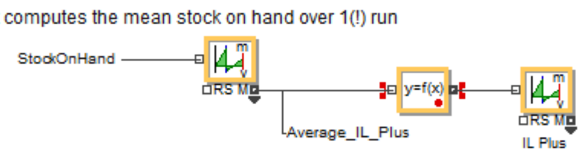


This block is basically the normal order generator. If InvPos is smaller or equal to R directly after we get an order we send a pulse to the normal order create block, which realeases an order.

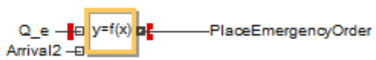


Supporting Computations (R, Q)-RH (2/2)

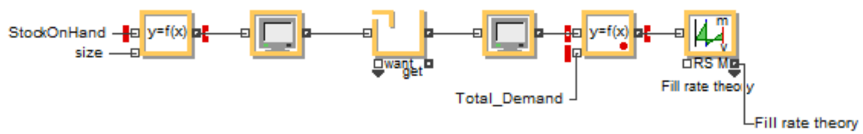
This block computes the mean stock on hand over 1(!) run



This sends a pulse to the emergency orders create block every time an emergency order should be placed. If a customer order arrives and this leads to a positive emergencyOrderQuantity from the order vector block, a pulse is sent.

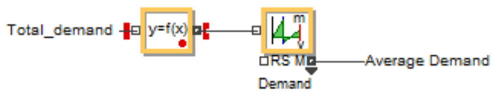


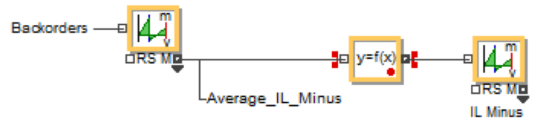
Computes theoretical fill rate, where also fractions of customer orders can be fulfilled

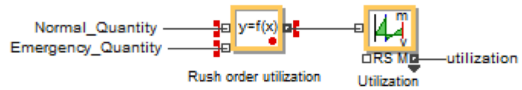


This block computes R + Q

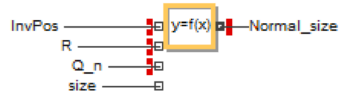
Computes the average demand per run.



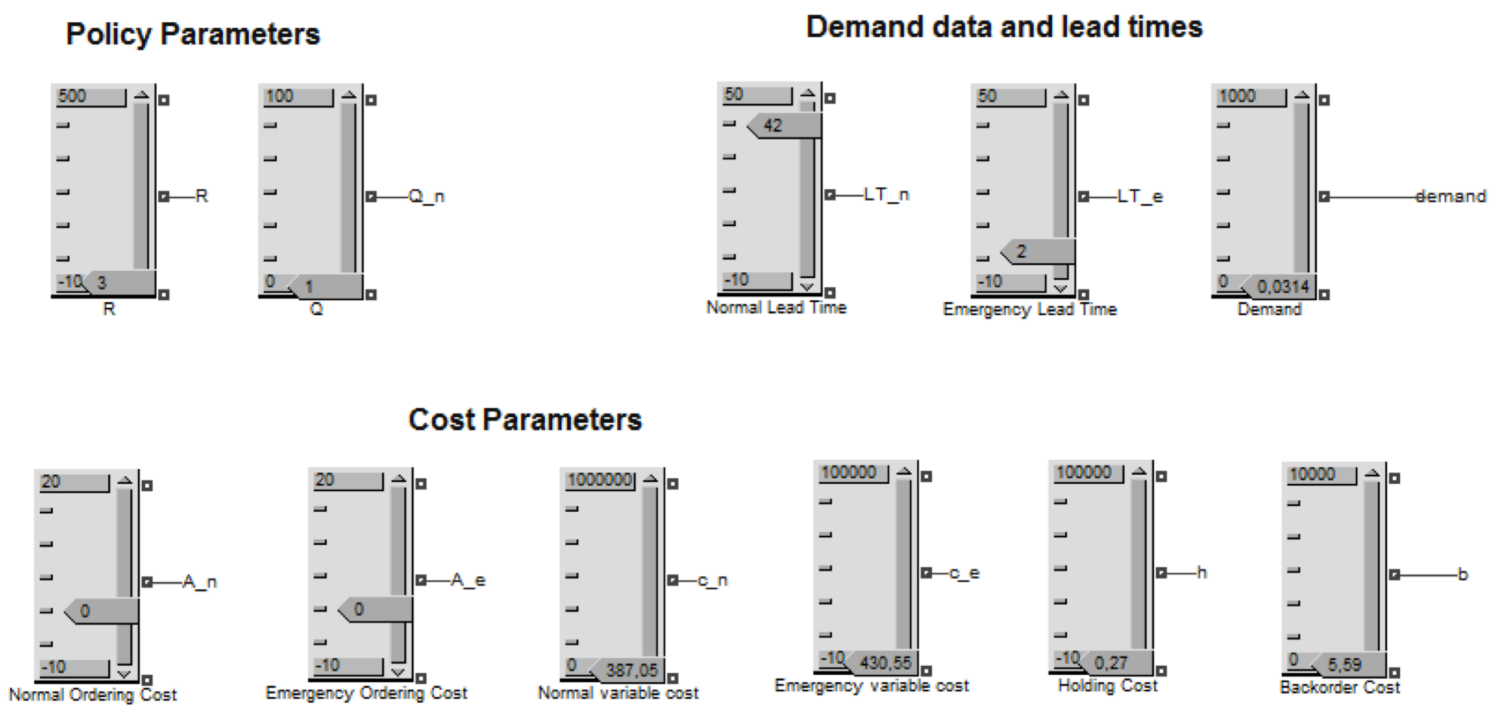




Determines how many multiples of the order size to order

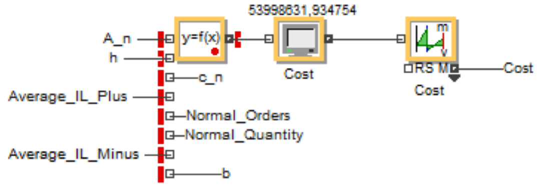


Input Parameters (R, Q)-RH

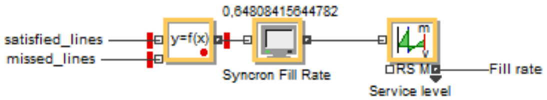


Supporting Computations (R, Q) (1/2)

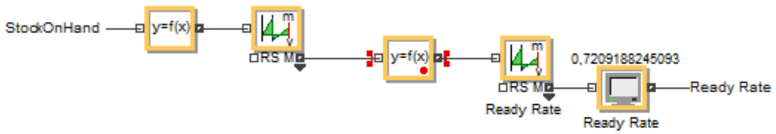
This structure computes all the costs at the end of each run and takes mean of the costs over multiple runs



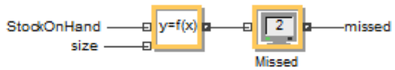
This block computes the over the counter order line fill rate as a mean over multiple runs



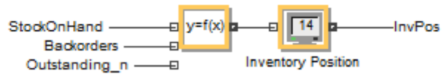
This block computes the ready rate as a mean of multiple runs



This block checks wheather a customer order line can be completely filled directly from stock or not. If yes, missed = 1, else missed = 2.

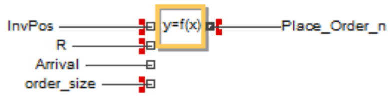


This block computes the inventory position, including backorders and outstanding normal and emergency orders

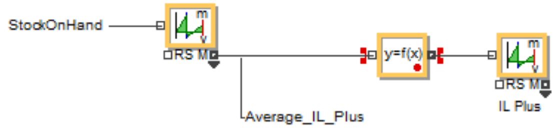


This block compares the satisfied demand with the total demand in order to compute the number of backorders at any given time

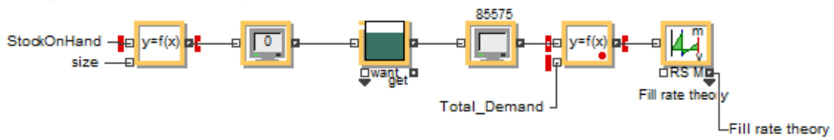
This block is basically the normal order generator. If InvPos is smaller or equal to R directly after we get an order we send a pulse to the normal order create block, which realeases an order.



This block computes the mean stock on hand over 1(!) run



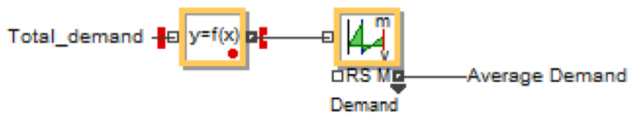
Computes theoretical fill rate, where also fractions of customer orders can be fulfilled

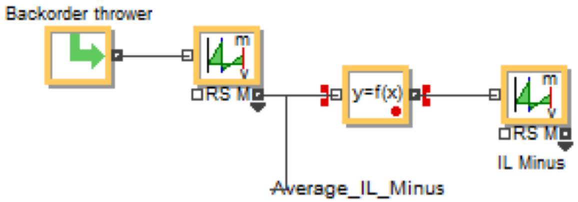


Supporting Computations (R, Q) (2/2)

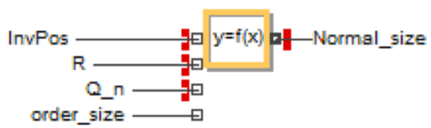
This block computes R + Q

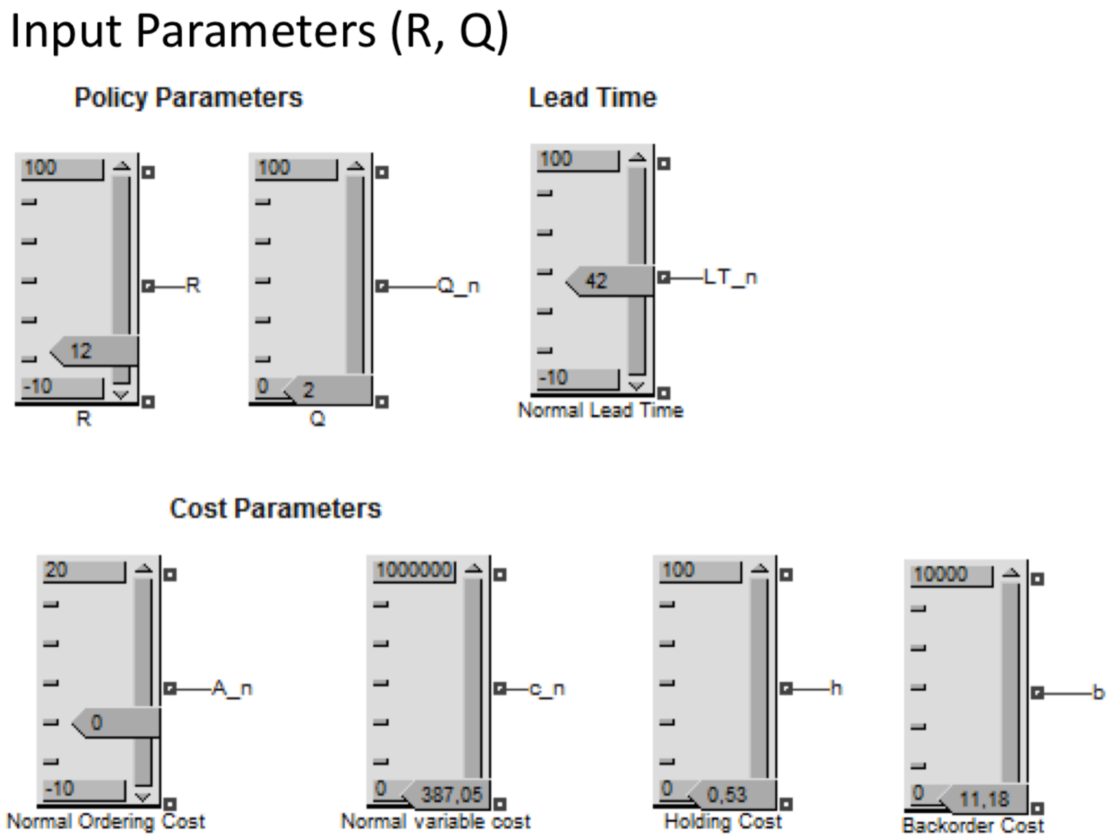
Computes the average demand per run.





Determines how many multiples of the order size to order

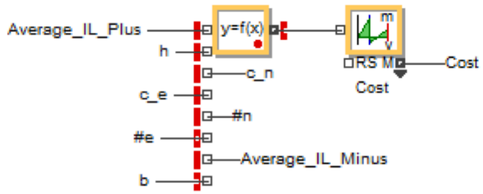




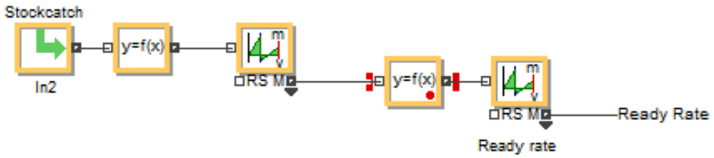
Input Parameters (R, Q)
Policy Parameters
Lead Time
100
12
-10
R
100
2
0
Q
Q_n
100
42
-10
Normal Lead Time
LT_n
Cost Parameters
20
0
-10
Normal Ordering Cost
A_n
1000000
0
387,05
Normal variable cost
c_n
100
0
0,53
Holding Cost
h
10000
0
11,18
Backorder Cost
b

Supporting Computations S&Z (1/1)

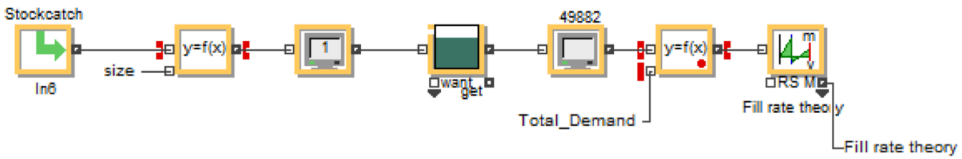
This structure computes all the costs at the end of each run and takes mean of the costs over multiple runs



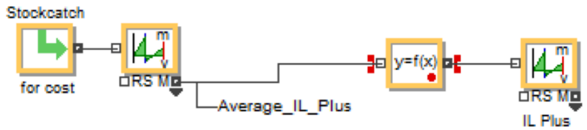
This block computes the ready rate as a mean of multiple runs



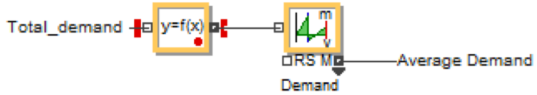
Computes theoretical fill rate, where also fractions of customer orders can be fulfilled

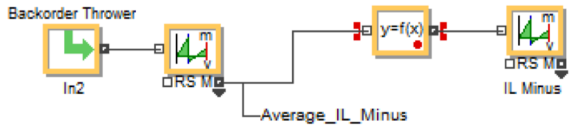


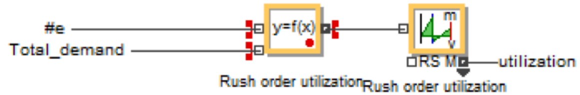
This block computes the mean stock on hand over 1(!) run



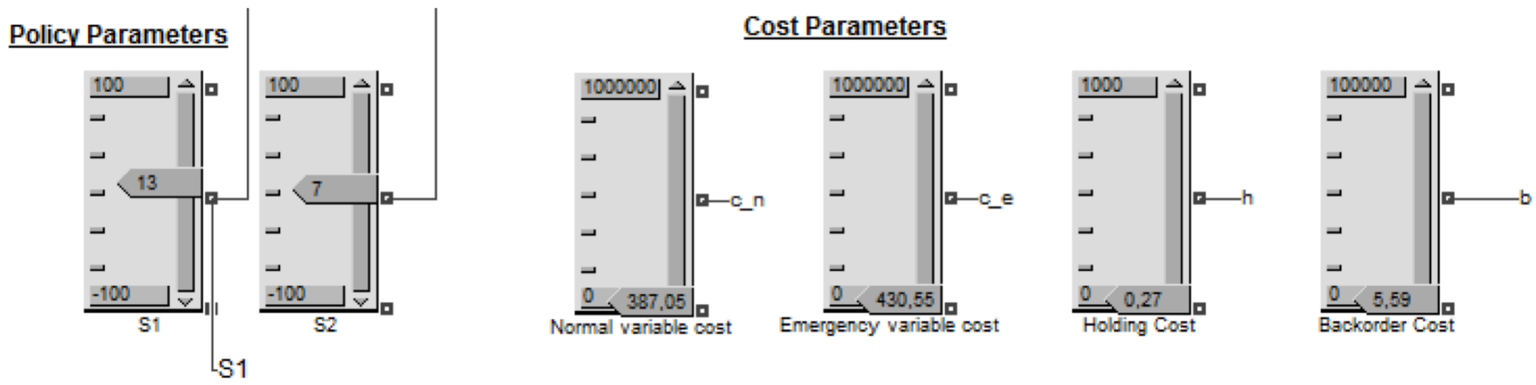
Computes the average demand per run.





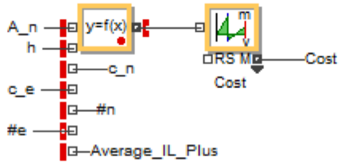


Input Parameters S&Z

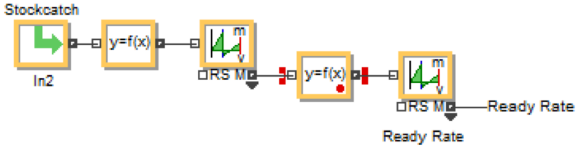


Supporting Computations (R, Q)-LS (1/1)

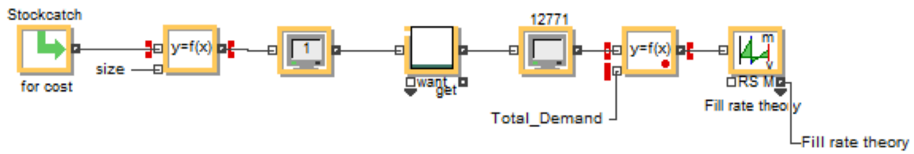
This structure computes all the costs at the end of each run and takes mean of the costs over multiple runs



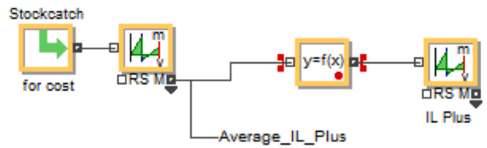
This block computes the ready rate as a mean of multiple runs



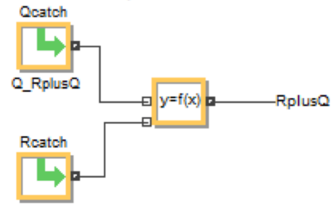
Computes theoretical fill rate, where also fractions of customer orders can be fulfilled



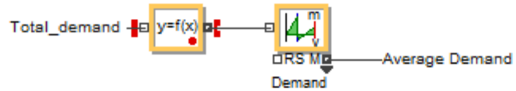
This block computes the mean stock on hand over 1(!) run

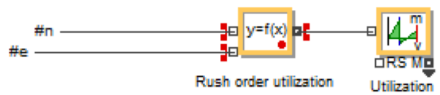


This block computes R + Q

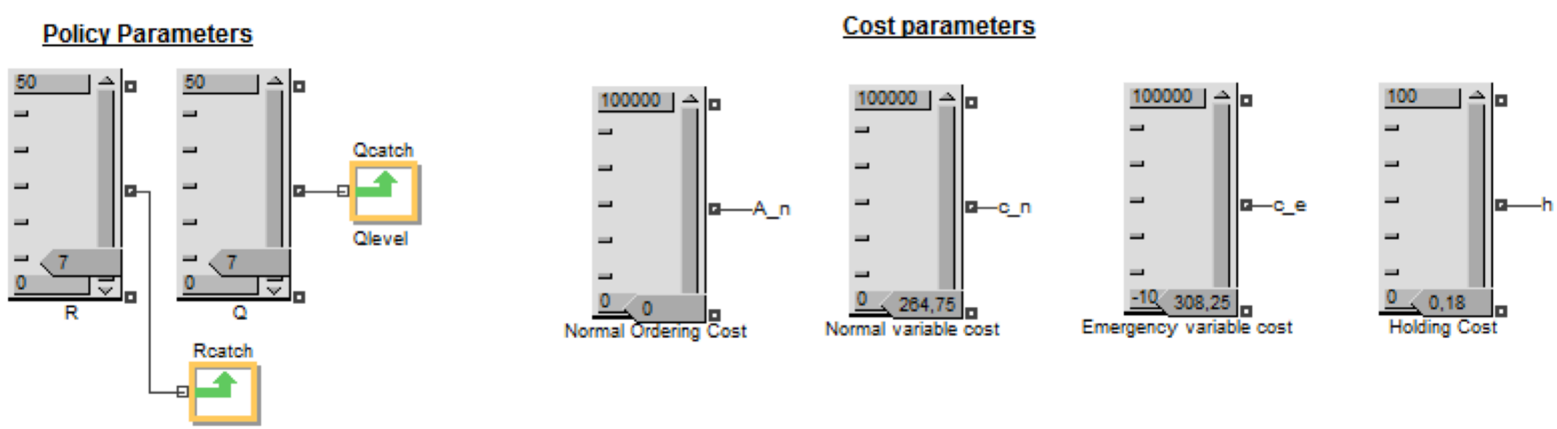


Computes the average demand per run.





Input Parameters (R, Q)-LS



Appendix VI

(R, Q)-LH – Custom block code in Extend software

```
//This is a customized block that keeps track of outstanding normal orders.
Only the time of arrival is tracked. It also calculates an emergency ordering
quantity based on the Long Horizon Algorithm.

// Declare constants and static variables here.
real Orders[100];
real Temp[100];
integer IP_e; //this should include IL and outstanding emergency orders
real normalLeadTime;
real emergencyLeadTime;
integer normalOrderQuantity;
real demand;
real sigma;
real sigma_prim;
real demand_prim;
real emergencyHorizon;
real ServiceLevel;
real ratio;
real c_e;
real c_n;
real A_e;
real A_n;
real h;
integer vectorSize;

procedure updateOrderVector()
{
   integer i, k;
   k=0;

   //Restore Temp
   for (i=0;i<vectorSize;i++)
   {
      Temp[i] = 0;
   }

   for (i=0; i <vectorSize; i++) {// Sets all incoming orders to 0.
      if (Orders[i]-currentTime <0)
      {
         Orders[i] = 0;
      }
      else
      {
         break;
```

```
        }
    }
    for (i=0; i < vectorSize; i++) {
        if (Orders[i]-currentTime > 0)
        {
            Temp[k] = Orders[i];
            k++;
        }
    }
    for (i=0; i <vectorSize; i++)
    {
        Orders[i] = Temp[i];
    }
}
on NormOrderIn
{
    integer i, k;

    k=0;
    //Restore Temp
    for (i=0;i<vectorSize;i++)
    {
        Temp[i] = 0;
    }
    if(NormOrderIn == 0)
    {
    }
    else{
        for (i=0; i <vectorSize; i++) {// Sets all incoming orders to 0.
            if (Orders[i]-currentTime <0)
            {
                Orders[i] = 0;
            }
        }
        for (i=0; i < vectorSize; i++) {
            if (Orders[i]-currentTime > 0)
            {
                Temp[k] = Orders[i];
                k++;
            }
        }
        for (i=0; i <vectorSize; i++)
        {
            Orders[i] = Temp[i];
        }
        Orders[k] = NormOrderIn + normalLeadTime;
    }
}
```

```
real abs(real nbr)
{
   if(nbr < 0)
   {
      return -nbr;
   }
   else
   {
      return nbr;
   }
}


//Cumulative Distribution Function (CDF) for standardized normal
distribution
real normalDist(real input)
{
   integer i;
   real sum, value, output;

   sum = input;
   value = input;

   if(input > 5)
   {
      output = 1;
   } Else if(input < -5)
   {
      output = 0;
   } Else
   {
      //This approximation can be found at Wikipedia. It has been tested.
      for (i=1; i<101; i++)
      {
         value = (value*input*input/(2*i+1));
         sum=sum+value;
      }
      output = 0.5+(sum/sqrt(2*pi))*exp(-(input*input)/2);
   }
   return output;
}


real inverseNormalDist(real input)
{
```

```
    //This one is suggested by Bell on Wikipedia (should give an error less
than 0,003 (maximum absolute error)):
    real output;
    if(input>0.5)
    {
        output=sqrt(-1.57079632679*log(1-(2*input-1)^2));
    }
    else
    {
        output = -sqrt(-1.57079632679*log(1-(1-2*input)^2));
    }
    if (abs(0.5-input)>0.24216)
    {
        if(input>0.5)
        {
            output = 1.010304*(output)^1.023630;
        }
        else
        {
            output = -1.010304*(- output)^1.023630;
        }
    }
    return output;
}

//This calculates the sum of the quantities of the outstanding normal orders
that arrive within a given time.
integer calculateIncomingQuantity(real time)
{
    integer i, quantity;
    real orderTime;
    quantity = 0;

    for (i=0; i< vectorSize; i++)
    {
        orderTime = Orders[i];

        if (orderTime == 0)
        {
            break;
        }
        if (orderTime < currentTime + time)
        {
            quantity += normalOrderQuantity;
        }
    }
    return quantity;
```

```
}

//This returns the index of the next outstanding normal order that will arrive
after one emergency lead time
//In case there is no normal order that meets this criteria, vectorSize will be
returned.
integer orderInQuestion()
{
   integer i;
   real orderTime;

   for (i = 0; i< vectorSize; i++)
   {
      orderTime = Orders[i];

      if (orderTime == 0) {
         break;
      }
      if(orderTime > currentTime + emergencyLeadTime)
      {
         return i;
      }
   }

   return vectorSize;
}

//This calculates the arrival time of the next outstanding normal order that
will arrive after one emergency lead time.
//In case there is no outstanding order fulfilling this criteria, current time ´+
normal lead time will be returned
real orderArrivalTime() {
   real time;
   integer index;

   index = orderInQuestion();
   if (index == vectorSize)
   {
      time = currentTime + normalLeadTime;
   } else
   {
      time = Orders[index];
   }
   return time;
}

//This procedure updates sigma and mu (called demand here) so that we
get these for the time in question.
```

```
procedure updateSigmaAndDemand() {
   real orderTime;
   orderTime = orderArrivalTime()-currentTime;
   sigma_prim = sigma*sqrt(orderTime);
   demand_prim = demand*orderTime;
}


//This calculates the emergency quantity that will bring back the expected
service level to the TSL
integer calculateEmergencyQuantity()
{
   integer q, t;
   real value;

   q=0;
   t = emergencyLeadTime;

   updateSigmaAndDemand();

   value = inverseNormalDist(ServiceLevel) * sigma_prim - IP_e +
demand_prim - calculateIncomingQuantity(t);

   if (value > 0)
   {
      //basically round up
      q = value + 1;
   }
   return q;
}

//This calculates the extra holding cost incurred by placing an emergency
order.
real calculateHoldingCost(integer quantity)
{
   real time;

   time = orderArrivalTime() - (currentTime + emergencyLeadTime);

   ctr1_prm = time;
   return quantity * h * time;

}

//This calculates the total extra cost incurred by placing an emergency
order
real calculateCost(real quantity)
{
```

```
   real directCost, indirectCost, normalQuantity;
   directCost = quantity*(c_e - c_n) + A_e -
(quantity/normalOrderQuantity)*A_n;
   indirectCost = calculateHoldingCost(quantity);

   return directCost + indirectCost;
}

//Every time we get an input to the updateIn-connector we will calculate the
emergency order quantity.
on UpdateIn
{
   integer i, emergencyQuantity, incomingQuantity;
   real SL_No_Action, SL_Action, delta_SL, SL_Cost_Ratio, delta_Cost;

   updateOrderVector();
   IP_e = NetInvIn - IncCustomerQuantityIn;
   incomingQuantity = calculateIncomingQuantity(emergencyLeadTime);

   updateSigmaAndDemand();

   SL_No_Action = normalDist((IP_e + incomingQuantity-
demand_prim)/(sigma_prim));

   QuantityOut = 0; //This makes sure that the gate, that lets emergency
orders pass, is closed whenever we don't want to place an emergency
order

   if(updateIn == 1)
   {

   //This is the actual algorithm for determining if to place an emergency
order and the size of it
   //In case you set ratio to 0, it will work just like a base stock policy based
on the emergency inventory position (Not to be confused with the IP_E
which only includes emergency orders)
      if(SL_No_Action < ServiceLevel)
      {
         emergencyQuantity = calculateEmergencyQuantity();
         for(i=emergencyQuantity; i>0; i--)
         {
            SL_Action = normalDist((i + IP_e + incomingQuantity-
demand_prim)/(sigma_prim));
            delta_SL = SL_Action-SL_No_Action;
            delta_Cost = calculateCost(i);
            SL_Cost_Ratio = delta_SL / delta_Cost;
            if (SL_Cost_Ratio > ratio) {
               QuantityOut=i;
```

```
                break;
            }
        }
    } else
    {
        QuantityOut=0;
    }
}
else
{
    QuantityOut = 0;
}
}

// This message occurs for each step in the simulation.
on simulate
{
    OutNum1_prm = Orders[0];
    OutNum2_prm = Orders[1];
    OutNum3_prm = Orders[2];
    OutNum4_prm = Orders[3];
    OutNum5_prm = Orders[4];
    OutNum6_prm = Orders[5];
    OutNum7_prm = Orders[6];
    OutNum8_prm = Orders[7];
    OutNum9_prm = Orders[8];
    OutNum10_prm = Orders[9];
}
// If the dialog data is inconsistent for simulation, abort.
on checkdata
{
}
// Initialize any simulation variables.
on initsim
{
    normalLeadTime = LTnIn;
    emergencyLeadTime = LTeIn;
    normalOrderQuantity = Qin;
    demand = DemandIn;
    ServiceLevel = SLIn;
    ratio = 1/ratioIn;
    sigma = sigmaIn;
    c_e = UnitCost_e_In;
    c_n = UnitCost_n_In;
    h = HoldingCost_In;
    A_e = OrderingCost_e_In;
    A_n = OrderingCost_n_In;
```

```
  vectorSize = 100; //OBS Don't forget to change the sizes of the order
vector and temp arrays
  integer i;
  for (i = 0; i< vectorSize; i++)
  {
    Orders[i] = 0;
    Temp[i] = 0;
  }
  //For testing normal distribution function (and inverse). Just printing the
variables in the block dialog
  inv_prm = inverseNormalDist(0.1);
  inv_prm_1 = inverseNormalDist(0.2);
  inv_prm_2 = inverseNormalDist(0.3);
  inv_prm_3 = inverseNormalDist(0.4);
  inv_prm_4 = inverseNormalDist(0.5);
  inv_prm_5 = inverseNormalDist(0.6);
  inv_prm_6 = inverseNormalDist(0.7);
  inv_prm_7 = inverseNormalDist(0.8);
  inv_prm_8 = inverseNormalDist(0.9);
  inv_prm_9 = inverseNormalDist(0.95);

  inv_prm_10 = normalDist(-1.28);
  inv_prm_11 = normalDist(-0.84);
  inv_prm_12 = normalDist(-0.52);
  inv_prm_13 = normalDist(-0.25);
  inv_prm_14 = normalDist(0);
  inv_prm_15 = normalDist(0.25);
  inv_prm_16 = normalDist(0.52);
  inv_prm_17 = normalDist(0.84);
  inv_prm_18 = normalDist(1.28);
  inv_prm_19 = normalDist(5);
}
```

(R, Q)-SH – Custom block code in Extend software

```
// This is a customized block that keeps track of outstanding normal orders.
Only the time of arrival is tracked. It also calculates an emergency ordering
quantity based on the Short Horizon Algorithm.

// Declare constants and static variables here.
real Orders[100];
real Temp[100];
integer IP_e; //this should include IL and outstanding emergency orders
real normalLeadTime;
real emergencyLeadTime;
integer normalOrderQuantity;
real demand;
real sigma;
real sigma_prim;
real demand_prim;
real emergencyHorizon;
real ServiceLevel;
real ratio;
real c_e;
real c_n;
real A_e;
real A_n;
real h;
real periodTime;
integer vectorSize;
procedure updateOrderVector()
{
   integer i, k;
   k=0;

   //Restore Temp
   for (i=0;i < vectorSize; i++)
   {
      Temp[i] = 0;
   }

   for (i=0; i < vectorSize; i++) {//Sets all incoming orders to 0.
      if (Orders[i]-currentTime <0)
      {
         Orders[i] = 0;
      }
      else
      {
         break;
      }
   }
   for (i=0; i < vectorSize; i++) {
```

```
        if (Orders[i]-currentTime > 0)
        {
            Temp[k] = Orders[i];
            k++;
        }
    }
    for (i=0; i < vectorSize; i++)
    {
        Orders[i] = Temp[i];
    }
}
on NormOrderIn
{
    integer i, k;

    k=0;
    //Restore Temp
    for (i=0;i < vectorSize; i++)
    {
        Temp[i] = 0;
    }
    if(NormOrderIn == 0)
    {
    }
    else{
        for (i=0; i < vectorSize; i++) {// Sets all incoming orders to 0.
            if (Orders[i]-currentTime <0)
            {
                Orders[i] = 0;
            }
        }
        for (i=0; i < vectorSize; i++) {
            if (Orders[i]-currentTime > 0)
            {
                Temp[k] = Orders[i];
                k++;
            }
        }
        for (i=0; i < vectorSize; i++)
        {
            Orders[i] = Temp[i];
        }
        Orders[k] = NormOrderIn + normalLeadTime;
    }
}

real abs(real nbr)
{
```

```
    if(nbr < 0)
    {
        return -nbr;
    }
    else
    {
        return nbr;
    }
}

//Cumulative Distribution Function (CDF) for standardized normal
distribution
real normalDist(real input)
{
    integer i;
    real sum, value, output;

    sum = input;
    value = input;

    if(input > 5)
    {
        output = 1;
    } Else if(input < -5)
    {
        output = 0;
    } Else
    {
        //This approximation can be found at Wikipedia. It has been tested.
        for (i=1; i<101; i++)
        {
            value = (value*input*input/(2*i+1));
            sum=sum+value;
        }
        output = 0.5+(sum/sqrt(2*pi))*exp(-(input*input)/2);
    }
    return output;
}


real inverseNormalDist(real input)
{
    //This one is suggested by Bell on Wikipedia (should give an error less
than 0,003 (maximum absolute error):
    real output;
    if(input>0.5)
    {
        output=sqrt(-1.57079632679*log(1-(2*input-1)^2));
```

```
    } else
    {
        output = -sqrt(-1.57079632679*log(1-(1-2*input)^2));
    }
    if (abs(0.5-input)>0.24216)
    {
        if(input>0.5)
        {
            output = 1.010304*(output)^1.023630;
        } else
        {
            output = -1.010304*(- output)^1.023630;
        }
    }
    return output;
}

//This calculates the sum of the quantities of the outstanding normal orders
that arrives within a given time.
integer calculateIncomingQuantity(real time)
{
    integer i, quantity;
    real orderTime;
    quantity = 0;

    for (i=0; i< vectorSize; i++)
    {
        orderTime = Orders[i];

        if (orderTime == 0)
        {
            break;
        }
        if (orderTime<currentTime+time) // If the time for the outstanding order
is before the time period when the emergency can impact, it should be
included.
        {
            quantity += normalOrderQuantity;
        }

    }
    return quantity;
}

//Returns the index of the next outstanding order that will arrive after one
emergency lead time. In case there is no, the vectorSize is returned.
integer orderInQuestion()
{
```

```
    integer i;
    real orderTime;

    for (i = 0; i< vectorSize; i++)
    {
        orderTime = Orders[i];
        if (orderTime == 0) {
            break;
        }
        if(orderTime > currentTime + emergencyLeadTime)
        {
            return i;
        }
    }
    return vectorSize;
}


//Calculates the arrival time of the next normal order
//If there is no such order outstanding it returns currentTime + 1 normal lead
time.
real orderArrivalTime() {
    real time;
    integer index;

    index = orderInQuestion();
    if (index == vectorSize)
    {
        time = currentTime + normalLeadTime;
    } else
    {
        time = Orders[index];
    }
    return time;
}

//This calculates the emergency quantity
integer calculateEmergencyQuantity()
{
    integer q, t;
    real value;

    q=0;
    t = emergencyLeadTime + periodTime;

    value = inverseNormalDist(ServiceLevel) * sigma_prim - IP_e +
demand_prim - calculateIncomingQuantity(t);

    if(value > 0) {
```

```
        //basically round up
        q = value + 1;
    }
    return q;
}

//This calculates the extra holding cost incurred by placing an emergency
order.
real calculateHoldingCost(integer quantity)
{
    real time;

    time = orderArrivalTime() - (currentTime + emergencyLeadTime);

    return quantity * h * time;

}

//This calculates the total extra cost incurred by placing an emergency
order
real calculateCost(real quantity)
{
    real directCost, indirectCost, normalQuantity;

    directCost = quantity*(c_e - c_n) + A_e -
(quantity/normalOrderQuantity)*A_n;
    indirectCost = calculateHoldingCost(quantity);

    return directCost + indirectCost;
}


on UpdateIn
{
    integer i, emergencyQuantity, incomingQuantity;
    real SL_No_Action, SL_Action, delta_SL, SL_Cost_Ratio, delta_Cost;

    updateOrderVector();
    IP_e = NetInvIn - IncCustomerQuantityIn;
    incomingQuantity = calculateIncomingQuantity(emergencyLeadTime +
periodTime);

    SL_No_Action = normalDist((IP_e + incomingQuantity-
demand_prim)/(sigma_prim));

    QuantityOut = 0;

    if(updateIn == 1)
```

```
  {
     //This is the actual algorithm for determining if to place an emergency
order and the size of it
  //In case you set ratio to 0, it will work just like a base stock policy based
on the emergency inventory position (Not to be confused with the IP_E
which only includes emergency orders)
     if(SL_No_Action < ServiceLevel)
     {
        emergencyQuantity = calculateEmergencyQuantity();
        for(i=emergencyQuantity; i>0; i--)
        {
           SL_Action = normalDist((i + IP_e + incomingQuantity-
demand_prim)/(sigma_prim));
           delta_SL = SL_Action-SL_No_Action;
           delta_Cost = calculateCost(i);
           SL_Cost_Ratio = delta_SL / delta_Cost;
           if (SL_Cost_Ratio > ratio) {
              QuantityOut=i;
              break;
           }
        }
     } else
     {
        QuantityOut=0;
     }
  } else
  QuantityOut = 0;
}


// This message occurs for each step in the simulation.
on simulate
{
  OutNum1_prm = Orders[0];
  OutNum2_prm = Orders[1];
  OutNum3_prm = Orders[2];
  OutNum4_prm = Orders[3];
  OutNum5_prm = Orders[4];
  OutNum6_prm = Orders[5];
  OutNum7_prm = Orders[6];
  OutNum8_prm = Orders[7];
  OutNum9_prm = Orders[8];
  OutNum10_prm = Orders[9];
}

// If the dialog data is inconsistent for simulation, abort.
on checkdata
{
```

```
}
// Initialize any simulation variables.
on initsim
{
   normalLeadTime = LTnIn;
   emergencyLeadTime = LTeIn;
   normalOrderQuantity = Qin;
   demand = DemandIn;
   ServiceLevel = SLIn;
   ratio = 1/ratioIn;
   sigma = sigmaIn;
   periodTime = 0; //Truly continuous review means that we can set this to 0
   sigma_prim = sqrt(emergencyLeadTime + periodTime)*sigma;
   demand_prim = (emergencyLeadTime + periodTime)*demand;
   c_e = UnitCost_e_In;
   c_n = UnitCost_n_In;
   h = HoldingCost_In;
   A_e = OrderingCost_e_In;
   A_n = OrderingCost_n_In;
   vectorSize = 100;
   integer i;
   for (i = 0; i< vectorSize; i++)
   {
      Orders[i] = 0;
      Temp[i] = 0;
   }
   inv_prm = inverseNormalDist(0.1);
   inv_prm_1 = inverseNormalDist(0.2);
   inv_prm_2 = inverseNormalDist(0.3);
   inv_prm_3 = inverseNormalDist(0.4);
   inv_prm_4 = inverseNormalDist(0.5);
   inv_prm_5 = inverseNormalDist(0.6);
   inv_prm_6 = inverseNormalDist(0.7);
   inv_prm_7 = inverseNormalDist(0.8);
   inv_prm_8 = inverseNormalDist(0.9);
   inv_prm_9 = inverseNormalDist(0.95);

   inv_prm_10 = normalDist(-1.28);
   inv_prm_11 = normalDist(-0.84);
   inv_prm_12 = normalDist(-0.52);
   inv_prm_13 = normalDist(-0.25);
   inv_prm_14 = normalDist(0);
   inv_prm_15 = normalDist(0.25);
   inv_prm_16 = normalDist(0.52);
   inv_prm_17 = normalDist(0.84);
   inv_prm_18 = normalDist(1.28);
   inv_prm_19 = normalDist(5);
}
```

(R, Q)-RH – Custom block code in Extend software model

```
//This is a customized block that keeps track of outstanding normal orders.
Only the time of arrival is tracked. Furthermore it calculates the emergency
ordering quantity, emulating the Syncron's GIM, at any given point
and returns it.
// Declare constants and static variables here.
integer vectorSize;
real Orders[100];
real Temp[100];
integer IP_e; //this should include outstanding emergency orders
integer normalLeadTime;
integer emergencyLeadTime;
integer normalOrderQuantity;
real demand;

procedure updateOrderVector()
{
   integer i, k;
   k=0;

   //Restore Temp
   for (i=0;i< vectorSize;i++)
   {
      Temp[i] = 0;
   }

   for (i=0; i <vectorSize; i++) {// Sets all incoming orders to 0.
      if (Orders[i]-currentTime <0)
      {
         Orders[i] = 0;
      }
      else
      {
         break;
      }
   }
   for (i=0; i < vectorSize; i++) {
      if (Orders[i]-currentTime > 0)
      {
         Temp[k] = Orders[i];
         k++;
      }
   }
   for (i=0; i <vectorSize; i++)
   {
      Orders[i] = Temp[i];
   }
}
```

```
on NormOrderIn
{
   IP_e = NetInvIn;
   integer i;
   integer k;
   k=0;
   //Restore Temp
   for (i=0;i< vectorSize;i++)
   {
      Temp[i] = 0;
   }
   if(NormOrderIn == 0)
   {
   }
   else{
      for (i=0; i < vectorSize; i++) {// Sets all incoming orders to 0.
              if (Orders[i]-currentTime <0)
         {
            Orders[i] = 0;
         }
      }
      for (i=0; i < vectorSize; i++) {
         if (Orders[i]-currentTime > 0)
         {
            Temp[k] = Orders[i];
            k++;
         }
      }
      for (i=0; i < vectorSize; i++)
      {
         Orders[i] = Temp[i];
      }
      Orders[k] = NormOrderIn+normalLeadTime;
   }
}

on UpdateIn
{
   real quantity, incQuant, tempQuantity, time, ROL;
   integer output, i, nbr;
   quantity = 0;
   incQuant = 0;
   tempQuantity = 0;
   output = 0;
   i=0;
   QuantityOut=0;
   ROL = 0;
   IP_e = NetInvIn-IncCustomerQuantityIn;
```

```
updateOrderVector();

if(updateIn == 1)
{
   while(orders[i] > 0)
   {
      time = Orders[i]-currentTime;
      if (time > emergencyLeadTime)
      {
         tempQuantity = IP_e + ROL + incQuant-time*demand;
         if (tempQuantity < quantity)
         {
            quantity = tempQuantity;
         }
      }
      incQuant = incQuant + normalOrderQuantity;
      i += 1;

      if (i == vectorSize)
      {
         break;
      }
   }

   tempQuantity = IP_e + incQuant + ROL - NormalLeadTime * demand;
   if (tempQuantity < quantity)
   {
      quantity = tempQuantity;
   }
   if(-quantity > 0)
   {
      nbr = quantity;
      ctr1_prm = -(quantity-nbr);
      ctr2_prm = -quantity;
      ctr3_prm = -nbr;

   //Basically round up
      If (quantity - nbr == 0)
      {
         output = -quantity;
         control_prm = output;
      } else
      {
         output = -quantity+1;
      }
      QuantityOut = output;
      control_prm = output;
   }
   else
```

```
        {
            QuantityOut = 0;
        }
    }

    else
    {
        QuantityOut = 0;
    }
}

// This message occurs for each step in the simulation.
on simulate
{
    OutNum1_prm = Orders[0];
    OutNum2_prm = Orders[1];
    OutNum3_prm = Orders[2];
    OutNum4_prm = Orders[3];
    OutNum5_prm = Orders[4];
    OutNum6_prm = Orders[5];
    OutNum7_prm = Orders[6];
    OutNum8_prm = Orders[7];
    OutNum9_prm = Orders[8];
    OutNum10_prm = Orders[9];
}
// If the dialog data is inconsistent for simulation, abort.
on checkdata
{
}
// Initialize any simulation variables.
on initsim
{
    normalLeadTime = LTnIn;
    emergencyLeadTime = LTeIn;
    normalOrderQuantity = Qin;
    demand = DemandIn;
    vectorSize = 100; //OBS Change the size of the ordervector and
tempvector to the same value as here!
    integer i;
    for (i = 0; i<vectorSize; i++)
    {
        Orders[i] = 0;
        Temp[i] = 0;
    }
    }
```