SPARE PARTS INVENTORY MANAGEMENT WITH DELIVERY LEAD TIMES AND RATIONING

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Yaşar Levent Koçağa

May, 2004

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Alper Şen (Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Ülkü Gürler

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Osman Alp

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet B. Baray
Director of the Institute

ABSTRACT

SPARE PARTS INVENTORY MANAGEMENT WITH DELIVERY LEAD TIMES AND RATIONING

Yaşar Levent Koçağa
M.S. in Industrial Engineering
Supervisor: Assist. Prof. Dr. Alper Şen
May, 2004

We study the spare parts service system of a major semiconductor equipment manufacturer facing two kinds of orders of different criticality. The more critical down orders need to be supplied immediately, whereas the less critical maintenance orders allow a given demand lead time to be fulfilled. For this system, we propose a policy that rations the maintenance orders. Under a one-for-one replenishment policy with backordering and for Poisson demand arrivals for both classes, we first derive expressions for the service levels of both classes and then conduct a computational study to illustrate superior system performance compared to a system without rationing. We also conduct a case study with 64 representative parts and show that significant savings are possible through incorporation of demand lead times and rationing.

Keywords: Inventory models, spare parts planning, multiple demand classes, rationing, demand lead time.

ÖZET

TALEP TEDARİK SÜRESİ VE KRİTİK SEVİYE POLİTİKASI İLE YEDEK PARÇA ENVANTER YÖNETİMİ

Yaşar Levent Koçağa
Endüstri Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Yrd. Doç. Dr. Alper Şen
Mayıs, 2004

Bu tez çalışmasında iki tip talep sınıfının gözlemlendiği yarı-iletken üreten makinaları imal eden bir firmanın yedek parça envanter sistemi incelenmiştir. Bu sistemde müşterilerdeki parça arızalarından kaynaklanan acil siparişlerin anında karşılanmalısı zorunluluğu varken, daha az kritik olan ve müşterilerin düzenli bakım aktivitelerinden kaynaklanan siparişler, sabit bir talep tedarik süresi sonrasında karşılanmaktadır. Bu sistemdeki envanter kontrolü için müşterilerdeki parça arızalarından kaynaklanan siparişlerin kritik talep sınıfı olduğu bir kritik seviye envanter kontrol politikasının kullanılması önerilmektedir. Her iki talep sınıfına ait talebin Poisson tipi rassal değişken olduğu ve zamanında karşılanmayan talebin kaybedilmediği varsayımları altında ve envanter seviyesinin birebir sipariş verme ile kontrol edildiği durum için her iki talep sınıfının servis seviyeleri belirlenmiş ve bu seviyeler yapılan eniyileştirme çalışmasında kullanılmıştır. Yapılan bu eniyileştirme çalışmasının sonucunda kritik seviye kontrol politikasının kullanılmadığı bir sisteme göre belirgin performans artışları saptanmıştır. Bu sonuçlar 64 parçanın kullanıldığı bir vaka analizi ile de desteklenmiştir.

Anahtar sözcükler: Envanter sistemleri, yedek parça planlaması, çoklu talep sınıfları, kritik seviye kontrol politikası, talep tedarik süresi.

Acknowledgement

I would like to express my most sincere gratitute to my advisor and mentor, Asst. Prof. Alper Şen for all the trust and encouragement during my graduate study. He has been supervising me with everlasting interest and great patience for this research and has helped me to shape my future research career.

I am also grateful to Prof. Ülkü Gürler for her invaluable guidance, remarks and recommendations not only for this thesis but also for my future career.

I am also indepted to Asst. Prof. Osman Alp for excepting to read and review this thesis and for his invaluable suggestions.

I would also like to thank to my officemate Banu Yüksel Özkaya for her kind help and guidance during my graduate studies.

Last but not the least, I would like to thank my numerious friends with whom I shared joy and sorrow over the last six years in Bilkent. Destabilize 61, without you life would be less joyful . . .

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The primary motivation behind this research is our experience with a leading semiconductor equipment manufacturer. The company manufactures systems that perform most of the primary steps in the chip fabrication process. The main customers of the company are semiconductor wafer manufacturers and semiconductor integrated circuit manufacturers, which either use the chips they manufacture in their own products or sell them to other companies downstream. The company owns research, development and manufacturing facilities in the United States, Europe and Far East and distributes its systems across the globe to world's leading semiconductor companies. The company is at the top of the supply chain for most personal computers and other high technology products.

Semiconductor systems are very expensive investments and are very critical to operations of many high technology companies. Unused semiconductor manufacturing capacity due to equipment failures is very costly. In order to provide spare parts and service to customers for equipment failures and scheduled maintenances, the company has an extensive spare parts network. The network consists of more than 70 locations across the globe, that consists of company owned distribution centers and depots. In addition, the company also has agreements with its leading customers where it manages the stock rooms (for all or a group of spare parts) in customer facilities (some of these are consignments). 3 continental distribution centers: one in North America, one in Asia and one in Europe constitute the backbone of the network and are primarily responsible for procuring and distributing spare parts to depots and customer locations. The depot locations are such that they can provide a 4-hour service to customers (those who do not have stock rooms operated by the company) for equipment failures ("down orders"). However, the continental distribution centers may also be used as a primary source for down orders for certain customers. In addition, the continental distribution center provides a second level of support for down orders that cannot be satisfied from the local depots. The customers also demand spare parts to be used in their scheduled maintenance activities ("lead time orders"). The primary source to meet these demands are usually the continental distribution centers. However local depots can also be used for this purpose for certain customers.

Both types of customer orders (down and lead time) go through an order fulfillment engine which searches for available inventory in different locations according to a search sequence specific to each customer. However the down orders need to be satisfied immediately (their request date is the date of order creation), while the lead time orders need to be satisfied at a future date. A depot may be facing down and lead time demand from a variety of customers, while a continental distribution center may be facing down and lead time demand from external customers in addition to the "replenishment orders" requested by internal customers: the depots and stock rooms managed by the company. The operations of this complex network is further complicated by a vast number of parts composed of consumables and non-consumables (more than 50,000 active parts need to be managed) and varying service level requirements by different customers.

While providing an implementable and "good" solution for the whole spares network is a proven challenge, we focus on an important issue where improvements can provide immediate and significant benefits. In the existing practice, for those locations that are facing different types of demand (down, lead time or replenishment), the company targets to achieve the maximum of the service level requirements while considering the aggregated demand. Moreover, the company does not recognize the possible demand lead times (the difference between requested date and ship date in excess of transportation time) for lead time orders and possible slacks (the difference between the replenishment lead time the company uses for planning downstream locations and transportation lead time) for replenishment orders. Obviously, this approach is inefficient. We suggest an inventory model that recognizes both the demand lead times and multiple demand classes, and allows for providing differentiated service levels through rationing. In Chapter 5, we use representative data from the company to show that our model generates significant savings.

Inventory systems have received extensive attention since the first half of the twentieth century. Effective management of inventory using Operations Research tools has been a major concern both in the literature and the industry. Basic, yet crucial questions such as when to replenish and how much to replenish have been the focus of inventory management. Since inventory costs constitute a significant portion of the costs a firms faces, the objective of inventory management has been ensuring a high level of customer service by holding the minimum possible amount of inventory. Although the depth of the focus of inventory management has extended from single locations to multiple locations (multi-echelon theory) and from a single product to customized products (product differentiation), in most cases demand from multiple sources is handled in a uniform way. However, just as different customers may require different product specifications, they may also require different service levels. Particularly, for a single product, different customers may have different stockout costs and/or different minimum service level requirements or different customers may simply be of different importance to the supplier by similar measures. Therefore, it can be imperative to distinguish between classes of customers thereby offering them different service. In this setting, different product demand from different customers can no longer be handled in a uniform way. This, in turn, gives rise to multiple demand classes and customer differentiation.

Multiple demand classes occur naturally in many inventory systems. Consider a two-echelon supply network consisting of a warehouse at the upstream and a number of retailers at the downstream. If the retailers are located in say, different regions and have different demand characteristics, it may be beneficial to assign retailers different priorities and differentiate demand accordingly. A similar example can be a two echelon supply network where the upstream is a warehouse which supplies customers (directly) and the downstream retailers (in the form of replenishment orders). In such a case, the stockout cost resulting from not being able to supply customers is usually much higher than that of the retailers since the latter one causes only a delay in the replenishment orders which usually results a lower cost.

Another example regarding inventory systems is a spare parts system. In a production system, a part may be installed in various equipment some of which being crucial to the continuum of production. Thus the demand for this spare part can be differentiated into several demand classes. Again, in a production system where the same component is used in multiple end products of different criticality (based on measures such as profitability) the demand of the end products can be differentiated accordingly. Observe that, in both examples, the demand does not come from different end customers. Yet, multiple demand classes occur naturally in both examples either in the form of demand for a spare part from equipment of different criticality or demand for a common component from different end products.

Multiple demand classes can also be observed in other systems. Revenue management is a celebrated example. The underlying assumption here is that some customers are willing to pay more for a room or seat than others. Therefore it can be optimal to refuse a low-price customer in anticipation of a future request from a high-price customer. It is indeed optimal if the customers arrive sequentially (first the low-price than the high-price customers) and the optimal policy has shown to be characterized by a set of protection levels which essentially are the minimum number of rooms reserved for future (high-price) classes. Observe that, in these problems the inventory is perishable and this leads to non-stationary control policies which adjust as time to expiration (i.e., flight date of the plane) approaches. Another distinguishing fact is that inventory level (capacity) is fixed. Thus, as opposed to most classical inventory systems, the replenishment decisions are irrelevant.

Given a system with multiple demand classes the easiest policy would be to use different stockpiles for each demand class. This way, it would be very easy to assign a different service level to each class. Also the practical implementation of this policy would be relatively easy. But the drawback of this policy is that no advantage would be taken from the so-called portfolio effect. In other words, the advantage of pooling demand from different demand sources together would no longer be utilized. Therefore, as a result of the increasing variability of the demand, more safety stock would be needed to ensure a minimum required service level which in turn means more inventory. On the other side, one could simply use the same pool of inventory to satisfy demand from various customer classes without differentiating them. In this case, the highest required service level would determine the total inventory needed and thus the inventory cost. The drawback of this policy is that we would be offering higher service levels to the rest of the demand classes, a deficiency that would lead to increased inventory costs.

Rationing or the so called critical level policy essentially lies between these two extremes. Rationing has proved to be effective to handle different demand classes with different stockout costs or service levels. Kleijn and Dekker [17] provide a comprehensive study illustrating various examples where multiple demand classes arise together with a literature review about the applications of rationing in such environments. We will explain this policy assuming there are two demand classes but the extension to several demand classes is straightforward. In this setting, certain part of the stock is reserved for high priority demand. This amount is called the critical level and once inventory level reaches this level, demand from lower priority demand class is no longer satisfied. If demand not satisfied immediately is backordered, how to handle replenishment orders is another problem. Obviously, if there is a backorder for a high priority customer upon the arrival of a replenishment order, it is optimal to use this replenishment order to satisfy this backorder. In addition, if there is a backorder for a low priority customer upon the arrival of a replenishment order and the inventory level is at or above the critical level, one should use this replenishment order to satisfy this backorder. However, if there is a low priority backorder and the inventory level is below the critical level one can either satisfy this backorder or increase the inventory level.

The latter one is referred to as the priority cleaning mechanism and has been proven to be optimal for specific conditions. Under general conditions, however, whichever of these is optimal depends on the problem settings. Notice that the service level of the low priority class is not affected by the way replenishment orders are handled. The drawback of the priority clearing mechanism is that it increases the average backorder length of a low priority customer.

Except for very specific cases, a simple critical level policy with a static critical level will not be optimal. An optimal policy should take into account the remaining time until the arrival of the next replenishment arrival. As the booking limits adjust to the remaining time until expiration in revenue management, the critical level in a rationing policy should also adjust dynamically. For example, if the inventory level is below the critical level, but it is known that a replenishment order will arrive within a short period of time, it may not be optimal to refuse a low priority demand arrival, especially if the probability of a high priority demand arrival within this time is very small. But employing such a dynamic rationing policy would be extremely difficult from a practical point of view. Thus, we prefer to focus on a static rationing policy where the critical level does not change over time.

Obviously the structure of the firm we study by itself inhibits different demand classes (down orders vs lead time orders) thereby creating an environment where rationing can be applied. Thus our approach in this research is to incorporate rationing to the current practice of the firm with two demand classes differentiated by their demand lead-time. Our motivation in taking this approach is that we believe it will result in better system performance given certain service level requirements. We consider the down orders as the high priority (or critical) class and the maintenance orders as low priority (or non-critical). But we must note, at this point, that if no commitment is made to the orders with zero demand lead time whereas orders with positive demand lead time are subject to a contract, the reverse could also be considered and the orders with the positive demand lead-time could be the critical (high priority) class.

We will first model the system as a single location system facing a Poisson demand in both critical and non-critical classes with rates λ_c and λ_n respectively. The spare part inventory is replenished according to a $(S-1, S)$ policy, S being the order-up-to level. For simplicity we consider a deterministic replenishment lead-time, L. The non-critical orders have a deterministic demand lead-time of T while the critical orders must be satisfied immediately. The service level we consider in modeling will be the type I service level, the probability of no stockout. Under these circumstances the policy works as follows: Once a critical order comes it is either immediately satisfied or backlogged if there is no inventory. On the other hand, a non-critical order is accepted at the time it arrives, and at its due date is satisfied if the inventory level is above a critical level, S_c, otherwise it is backlogged. Our aim will be to find the optimum S and S_c such that the given service levels requirements $\bar{\beta}_c$ and $\bar{\beta}_n$ are satisfied.

The remainder of the thesis is organized as follows:

In Chapter 2, we will provide a review of the literature in inventory systems with demand lead time and inventory systems with rationing.

In Chapter 3, we first derive the service levels for both customer classes. Although the service level of the non-critical class can be calculated analytically the service level of the critical class can only be approximated. Thus we present our approximation and prove that it is a lower bound for the actual service level under priority clearing mechanism. Having proved our approximation is a lower bound for the actual service level we go one step further and conduct a simulation study to see how our approximation works under reasonable service levels. The model of this simulation study is also explained in this section. Lastly, we present the service level optimization model that we consider and its algorithm.

In Chapter 4, we present the results of our simulation study which indicates that our approximation for the service level of the critical class works extremely well for high service levels of the critical class. In addition, we present the results of the optimization study that we conducted using our justified approximation for the critical service level.

In Chapter 5, we present our results from a case study that we conducted using 64 parts from the semiconductor equipment manufacturer that we described earlier.

In Chapter 6, we conclude the thesis giving an overall summary of what we have done, our contribution to the existing literature and its practical implications.

Chapter 2

Literature Survey

In this chapter, we will first review the literature in inventory systems with demand lead time. Then we will elaborate on the literature about rationing. We find it useful to distinguish between the periodic review literature and continuous review literature. Therefore we will first focus on the periodic review models and then proceed with the continuous review models. We will conclude this section with a table which essentially summarizes the literature about rationing.

A single location service parts system was first considered by Scarf [24] where there exists only one service class. Scarf, efficiently solved the model by observing that the replenishment process is equivalent to an $M/G/\infty$ queue. This fact makes Palm's theorem [23] applicable which states that the steady state number of customers waiting in the queue, which are the outstanding orders in our case, is Poisson distributed with a mean equal to the arrival rate multiplied by the average service time. Using the outstanding order distribution and the standard inventory balance equation (on-hand inventory = base-stock level - outstanding orders + backorders), it becomes easy to derive performance metrics such as on-hand inventory distribution and random customer delay. Later, Sherbrooke and Feeney [9] extend this model to include compound Poisson arrivals. Beginning with the seminal METRIC [25], many researchers have studied service parts systems in the context of multi-echelon distribution systems. Other research in this area include [11], [21], [2] and [4]. We note that, as a result of the introduction of the non-emergency service class, the standard inventory balance equation is no longer valid for the model we consider.

The concept of demand lead-time was first introduced by Simpson [26] by the term ”service time” for base-stock, multi-stage production systems. Hariharan and Zipkin [15] then coined the name “demand lead-time” to describe inventory-distribution systems where customers do not require immediate delivery of orders and allow for a fixed delay. The key observation of both papers is that a demand lead-time works just as the opposite of supply lead-time reducing the inventory required for achieving a required service level. Obviously this fact also applies to the system we consider but the existence of the two service classes makes the system more complex requiring a different analysis. Moinzadeh and Aggarwal [19] consider a two echelon system with two modes of inventory replenishment. However, in their case all orders are satisfied on a FCFS basis while the two order classes differ only in their transportation lead-times between the echelons. On the other hand, in the system we consider, orders are satisfied on a FDFS (first-due-first-serve) basis. Wang, Cohen and Zheng [30] analyze a similar two echelon system in order to derive the transient and steady performance metrics of the system. This work is actually the most relevant to ours in terms of the presence of two classes of service differentiated by a demand lead-time. Therefore we prefer to explore their work profoundly.

Wang, Cohen and Zheng [30] first study a single location system and derive expressions for the inventory level distribution and random customer delay. As a result, an expected yet crucial observation is made: the service level of customers with positive demand lead times is higher than service level for customers with zero demand lead time as long as there is a positive probability that the replenishment order corresponding to a customer with positive demand lead time arrives before its demand due date is made. After deriving the steady state performance metrics for the single location system, the model is extended to a two echelon system. By following an approach similar to the well-known METRIC, the multi-echelon network is decomposed into single location subsystems. After the analysis of the two-echelon setting, an optimization study is conducted to see the effects of the introduction of a non-emergency service class. As a result it is seen that the system with two service classes results in significant cost savings in terms of inventory as a result of the non-zero demand lead-time.

In the system we consider, the customers with positive demand lead times constitute the non-critical demand class, while the customers with zero demand lead times constitute the critical demand class. Therefore, it is imperative that we use a policy that could provide a higher service level to the demand class with zero demand lead times. Rationing is such a policy. In the standard policy, whenever on-hand inventories drop below a certain level - usually called critical level, rationing level or threshold level of the associated customer class- the demands of the lower priority classes are not satisfied with the expectation of future high priority class customer demands.

The literature about rationing begins with Veinott [29] who was the first to consider the problem of several demand classes in inventory systems. He analyzed a periodic review inventory model with n demand classes and zero lead-time with limited ordering, and introduced the notion of a critical level policy. Topkis [28] proved the optimality of this policy both for the case of backordering and for the case of lost sales. The problem was analyzed by breaking down the period until the next ordering opportunity into a finite number of subintervals. In any given interval the optimal rationing policy is such that demand from a given class is satisfied from existing stock as long as there remains no unsatisfied demand from a higher class and the stock level does not drop below a certain critical level for that class. The critical levels are generally decreasing with the remaining time until the next ordering opportunity. Independent of Topkis, Evans [8] and Kaplan [16] fundamentally derived the same results for two demand classes. In his paper Kaplan [16] suggested to let the critical level depend on the time until next replenishment. A single period inventory model where demand occurs at the end of a period is presented by Nahmias and Demmy [22] for two demand classes. This work was later generalized by Moon and Kang [20]. Nahmias and Demmy [22] generalized their results to a multi-period model with zero lead-times and an (s, S) inventory policy. Atkins and Katircioglu [1] analyzed a periodic review inventory system with several demand classes, backordering and a fixed lead-time; where for each class a minimum service level was required. For this model they presented a heuristic rationing policy. Cohen, Kleindorfer and Lee [3] also considered the problem of two demand classes, in the setting of a periodic review (s, S) policy with lost sales. However, they did not use a critical level policy. At the end of each period the inventory is issued with priority such that stock is used to satisfy high-priority demand first, followed by low-priority demand.

Frank, Zhang and Duenyas [10] considered a periodic review inventory system with two priority demand classes, one deterministic and the other stochastic. The deterministic demand must be supplied immediately while stochastic demand not satisfied is lost. Thus at each decision epoch, one has to decide how much demand to fill from the stochastic source along with the usual replenishment decisions. They first characterize the optimal policy and show that it has a complex state dependent structure. Therefore they proposed a simpler policy, called (s, k, S) policy, k being the static critical level determining how much stochastic demand to satisfy, and provided a numerical study which shows that this simpler policy works very well.

Nahmias and Demmy [22] were the first to consider multiple demand classes in a continuous review inventory model. They analyzed a (Q, r) inventory model, with two demand classes, Poisson demand, backordering, a fixed lead-time and a critical level policy, under the crucial assumption that there is at most one outstanding order. This assumption implies that whenever a replenishment order is triggered, the net inventory and the inventory position are identical. The model of Nahmias and Demmy is analyzed in a lost sales context by Melchiors, Dekker and Klein [18].

Ha [12] considered a lot-for-lot model with two demand classes, backordering and exponentially distributed lead-times and showed that this model can be formulated as a queuing model. He showed that in this setting a critical level policy is optimal, with the critical level decreasing in the number of backorders of the low-priority class. Moreover he proved that it is optimal to increase the stock level upon arrival of a replenishment order, even if there are backorders for low-priority customers when the inventory level is below the critical level.

A critical level policy for two demand classes where the critical level depends on the remaining time until the next stock replenishment was discussed by Teunter and Klein Haneveld [27]. A so-called remaining time policy is characterized by a set of critical stocking times $(L_1, L_2, ...)$; if the remaining time until the next replenishment is at most L_1, no items are reserved for the high-priority customers; if the time is between L_1 and $L_1 + L_2$ then one item should be reserved, and so on. They first analyze a model, which is the continuous equivalent of the periodic review models by [8] and [16]. Teunter and Klein Haneveld [27] also presented a continuous review (s, Q) model with deterministic lead-times. Under the assumption that an arriving replenishment order is large enough to satisfy all outstanding orders for high-priority customers, they derived a method to find near optimal critical stocking times. They showed such a remaining time policy outperforms a simple critical level policy where all critical levels are stationary.

Ha [13] considered a single item, make-to-stock production system with n demand classes, lost sales, Poisson demand and exponential production times. He modeled the system as an $M/M/1/S$ queuing system and proved that a lot-for-lot production policy and a critical level rationing policy is optimal. Moreover, it is also shown that the optimal policy stationary. For two demand classes, he presented expressions for the expected inventory level and the stockout probabilities. To determine the optimal policy, he used an exhaustive search, and made the assumption that the average cost is unimodal in the order-up-to level. Ha [14] generalized his policy for Erlang distributed lead times where he stated that the critical level policy would also provide good results under generally distributed lead-times.

Dekker et. al. [5] analyzed a similar model, with n demand classes, lost sales, Poisson demand but generally distributed lead-times. They modeled this system to derive expressions for the average cost and service levels. In addition, the authors derived efficient algorithms to determine the optimal critical level, order-up-to level policy, both for systems with and without service level constraints. Moreover they presented a fast heuristic approach for the model without service level constraints. In this model the different demand classes are characterized by different lost sales costs. Deshpande, Cohen and Donohue [7] considered a rationing policy for two demand classes differing in delay and shortage penalty costs with Poisson demand arrivals under a continuous review (Q, r) environment. They did not make the assumption of at most one outstanding order which makes the allocation of arriving orders a major issue to consider. They defined a so-called threshold clearing mechanism to overcome the difficulty of allocating arriving orders and provided an efficient algorithm for computing the optimal policy parameters which are defined by (Q, r, K), K being the threshold level.

Dekker, Kleijn and de Rooj [6] discussed a case study on the inventory control of slow moving spare parts in a large petrochemical plant, where parts were installed in equipments of different criticality. They studied a lot-for-lot inventory model with two demand classes, but without the assumption of at most one outstanding order. Demand for both classes is assumed to be Poisson while the replenishment lead-time is assumed to be deterministic. The primary contribution of this paper is the derivation of service levels for both classes in the form of probability of no stockout. However, the service level for the critical demand is only an approximation since it depends on how incoming replenishment orders are handled in a complicated way, while the service level for non-critical demand class which is exact, since it is not effected by the way incoming orders are handled.

We again note that the primary difference between our model and earlier research is that we simultaneously consider demand lead times and rationing.

We conclude the section with Table 2.1 which essentially summarizes the literature about rationing. We have classified the research based on several attributes. The first one is the demand process which is divided as being Poisson, general or deterministic. The second one is the number of demand classes considered and it is either 2 or n which stands for multiple demand classes. The third one is the type of review policy and it is either periodic or continuous review. The fourth one classifies the research based on whether demand not satisfied at its due date is backlogged or lost. The fifth and the last one is the lead time and it is classified as zero or positive. If there is a positive lead time it is further classified as being exponential, generally distributed or fixed. Lastly observe that some references occur more than once in Table 2.1. This is because these references include more than one model and each such model is dedicated a separate row.

Reference	Demand Process			Number of Demand Classes		Review Pol.		Backorders		Lead time			
	Poisson	General	Deter.	2	n	Per.	Cont.	Yes	No	Zero	Positive		
											Expo.	General	Fixed
Atkins and Katircioglu [1]					√	√		√					√
Dekker et. al.[5]	√				√		√		√			√	
Dekker et. al.[6]	√			√			√	√					√
Deshpande et. al.[7]	√			√			√	√					√
Evans [8]		√		√		√		√	√	√			
Frank et. al.[10]		√	√	√		√		√					√
Ha [12]	√			√			√	√			√		
Ha [13]	√				√						√		√
Ha [14]			√	√							√		
Kaplan [16]		√		√		√		√	√	√			
Melchiors et.al.[18]	√			√			√		√				√
Moon and Kang [20]	√				√	√		√					
Moon and Kang [20]		√		√			√	√					√
Nahmias and Demmy [22]	√			√			√	√					√
Nahmias and Demmy [22]	√			√		√		√		√			
Nahmias and Demmy [22]	√			√		√		√		√			
Teunter and Klein Haneveld [27]	√			√			√	√					√
Topkis [28]		√			√	√		√	√	√			
Veinott [29]		√			√	√				√			

Table 2.1: Summary of studies on inventory rationing

Chapter 3

Model

We consider a single location spare part inventory system which faces two classes of demand arrivals with different criticality. The down orders which result from the equipment failures of customers are assumed to constitute the high priority, i.e., critical class, whereas the maintenance orders are assumed to constitute the low priority, i.e., non-critical class. Demand arrivals of the critical and non-critical class are both assumed to be Poisson with rates of λ_c and λ_n, respectively. Both arrivals are satisfied from the same pool of inventory which is controlled by a base stock policy with a base stock level S. Therefore, each demand arrival triggers a replenishment order with a deterministic lead time of L. In addition, the demand from the non-critical class allows a deterministic demand lead time of T, which is called the demand lead time. Before proceeding with the description of our rationing policy, we provide the following notation which will be used throughout the rest of this thesis:

λ_c = Arrival rate in the critical demand class;

λ_n = Arrival rate in the non-critical demand class;

L = Replenishment (supply) lead time;

T = Demand lead time;

$\bar{\beta}_c$ = Service level requirement for critical class;

$\bar{\beta}_n$ = Service level requirement for non-critical class;

S = Base stock level;

S_c = Critical level;

$\beta_c(S, S_c)$ = Service level for critical class for a given S, S_c;

$\beta_n(S, S_c)$ = Service level for non-critical class for a given S, S_c;

$I(a)$ = Inventory level net of backorders for non-critical class at time a;

$B_n(a)$ = Backorders for the non-critical class at time a;

$D_c(a, b]$ = Critical demand due in interval $(a, b]$;

$D_n(a, b]$ = Non-critical demand due in interval $(a, b]$;

$R(a, b]$ = Replenishments that are received in interval $(a, b]$;

H = Hitting time, i.e., arrival time of the $(S - S_c)$th total demand.

Note that $D_c(a, b]$ is a Poisson random variable with rate $\lambda_c \times (b - a)$ and $D_n(a, b]$ is a Poisson random variable with rate $\lambda_n \times (b - a)$. H is an Erlang $S - S_c$ random variable with rate $\lambda_c + \lambda_n$. In our model, we will use type I service level, i.e., the probability of no stock out, as our service level measure. We note that because of the PASTA (Poisson Arrivals See Time Averages) property, this is also the type II service level, i.e., the fill rate.

In this setting, our proposed policy shall work as follows: whenever a critical order arrives, it is immediately satisfied if the on-hand inventory is positive or backlogged if the on-hand inventory is zero. A non-critical order is accepted as it arrives, and at its due date, that is, T time units after its arrival, it is satisfied only if the on-hand inventory is above the critical level, S_c, otherwise it is backlogged. Note again that whether critical or non-critical, each demand arrival triggers a replenishment order which will arrive after L time units. Incoming replenishment orders are allocated according to a priority clearing mechanism. Under this mechanism, replenishment arrivals are allocated as follows: if there is a critical backorder at the time of a replenishment arrival it is immediately cleared, if there is a non-critical order it is cleared only if the on-hand inventory has reached S_c. In other words, incoming replenishment orders are used to clear backorders of the non-critical class only if the on-hand inventory is at the critical level, S_c. Given our rationing policy, the service level for the critical and non-critical classes clearly depend on S and S_c (as well as parameters of the system: λ_c, λ_n, L, T).

We assume that $\bar{\beta}_n < \bar{\beta}_c$, which means that the demand class with demand lead time has a service level requirement lower than the demand class without demand lead time. This assumption is valid for the semiconductor equipment manufacturer that motivated this research. However, we note that in other applications, the demand class with demand lead time can in fact be the demand class that needs prioritized service. For example, in a retail setting, the customers in the demand class with demand lead time (these could be online orders) submit their orders in advance, and a commitment is made upon the acceptance of these orders, whereas no prior commitment is made to the customers in the demand class without demand lead time, who ask for inventory upon their arrival to the store.

We also assume that $T \leq L$. This is a reasonable assumption since replenishment lead times are usually long and spare part providers cannot quote a demand lead time longer than the replenishment lead times. This assumption is also valid for the semiconductor equipment manufacturer that we analyze.

Given this system, our purpose is to determine the minimum inventory investment which satisfies the service requirements for both classes. Furthermore, we assume the ownership of on-order inventory and minimize expected inventory on hand plus on expected inventory on order. Note that unlike the case in a standard continuous review $(S-1, S)$ policy, the inventory position is not always equal to S in this system with demand lead times. The expected inventory position is in fact equal to $S + \lambda_n \times T$, where the second term is due to the outstanding replenishment orders for the non-critical demand class that are yet not due. When we assume that fill rates are reasonably high, we can approximate the expected inventory on hand plus expected inventory on order by the inventory position. Thus we select our objective as minimizing S (since $\lambda_n \times T$ is a constant). Our optimization problem for given λ_c, λ_n, L, T, and minimum service level requirements $\bar{\beta}_c$ and $\bar{\beta}_n$ is given as follows:

$$\begin{aligned}
&\min_{S,S_c} S \\
&s.t \\
&\beta_c(S, S_c) \geq \bar{\beta}_c \\
&\beta_n(S, S_c) \geq \bar{\beta}_n \\
&S, S_c \geq 0
\end{aligned}$$

Observe that the service level for the critical class is closely related to the way incoming orders are handled and thus the arrival process. Therefore finding a closed form expression for the service level of the critical class is extremely difficult and for this reason we have to resort to approximations. In the next section, we will derive the service level of the non-critical class and an approximation for the service level of the critical class.

3.1 Deriving the Service Levels

In this section, we derive the resulting service levels for a given set of policy parameters: S, S_c. The service level that we derive is exact for the non-critical demand class. The service level that we derive for the critical demand class, however, is an approximation. But, we show analytically that the approximation constitutes a lower bound for the actual service level for the critical demand class, when we use a priority clearing mechanism to clear the backorders.

First consider the service level for the non-critical demand class and consider the interval $(t, t + L]$. Since all outstanding orders at time t would arrive by time $t + L$, the inventory level at time $t + L$ would be S, if no demand occurred during the interval. In order for a non-critical demand arriving at $t+L-T$ to be fulfilled at its due date $t + L$, the inventory level at time $t + L$ must be at least S_c+1 and this would happen if and only if the sum of the critical demand during $(t, t + L]$ and the non-critical demand due in $(t + T, t + L]$ is less than $S - S_c$. Observe that we are not considering the non-critical demand due in $(t, t + T]$ as the replenishments for these demands are already received by time $t + L$, and hence they do not impact the inventory level at time $t + L$. Thus, the service level of the non-critical demand class is given by:

$$\beta_n(S, S_c) \;=\; P\left\{D_c(t, t + L] + D_n(t + T, t + L] \le S - S_c - 1\right\}.$$

Thus, we have the following expression for the service level of the non-critical demand class

$$\beta_n(S, S_c) = \sum_{i=0}^{S-S_c-1} \frac{e^{-[(\lambda_c+\lambda_n)L-\lambda_n T]}\left[(\lambda_c + \lambda_n)L - \lambda_n T\right]^i}{i!} \qquad (3.1)$$

We again note that the expression in Equation (3.1) is an exact expression for the non-critical demand class.

Now consider the service level for the critical demand and again consider the time interval $(t, t+L]$. Since all outstanding orders at time t would arrive by time $t+L$, the inventory level at time $t+L$ would be S, if no demand occurred during the interval. In order to satisfy a critical demand arriving at $t + L$, there must be at least one unit of inventory at $t + L$. Note that the replenishment orders corresponding to the non-critical demands that are due in the interval $(t, t + T]$ are received in the interval $(t+L-T, t+L]$. In order to calculate the probability that there is at least one unit of inventory at $t + L$, we condition on whether the hitting time, i.e., first $S - S_c$ units of total demand occurs in the interval $(t, t + L - T]$ or in the interval $(t + L - T, t + L]$. If the hitting time is in the interval $(t, t + L - T]$, then there should be at most $S_c - 1$ critical demands after the hitting time. If the hitting time is in the interval in $(t + L - T, t + L]$, say at time $t + L - T + z$, we need to consider non-critical demands that are due only in the interval $(t + z, t + L - T + z]$, as the replenishment orders corresponding to the non-critical demands that are due in period $(t, t + z]$ will arrive before

$t+L-T+z$. Therefore, regardless of what z is, we can use $D_n(t,t+L-T]$ to represent non-critical demands that have a net impact on inventory. Thus, the approximation for the service level for the critical demand class is given by:

$$\begin{aligned}\beta_c(S,S_c) &= P\left\{D_c(t+H,t+L] \leq S_c-1, H \leq T-L\right\} \\ &+ P\left\{D_c(t,t+L-T]+D_n(t,t+L-T] \leq S-S_c-1,\right. \\ &\qquad \left. D_c(t,t+L]+D_n(t,t+L-T] \leq S-1\right\}\end{aligned}$$

Realizing that H is an Erlang $S-S_c$ random variable with rate $\lambda_c+\lambda_n$, we have:

$$\begin{aligned}\beta_c(S,S_c) &= \int_0^{L-T} (\lambda_c+\lambda_n)^{S-S_c} \frac{y^{S-S_c-1}}{(S-S_c-1)!} \times (\sum_{i=0}^{S_c-1} \frac{e^{-\lambda_c(L-y)}\left[\lambda_c(L-y)\right]^i}{i!})dy \\ &+ \sum_{i=0}^{S-S_c-1} \sum_{x=0}^{S-i-1} \frac{e^{-(\lambda_c+\lambda_n)(L-T)}\left[(\lambda_c+\lambda_n)(L-T)\right]^i}{i!} \times \frac{e^{-\lambda_c T}\left[\lambda_c T\right]^x}{x!} \quad (3.2)\end{aligned}$$

Note again that the expression in Equation (3.2) is an approximation for the service level of the critical demand class. This is due to the following reasons. First note that rationing may not start exactly at the hitting time since the inventory level at time t may not be S or all outstanding orders at time t may not arrive before the hitting time. Also the expression assumes that once the rationing starts, we will keep on rationing until $t+L$, which may not be the case. Though the expression in Equation (3.2) is an approximation, we next show that it is a lower bound for the actual service level when the incoming replenishment orders are handled according to a priority clearing mechanism.

Theorem 1 *The approximation for the critical service level given in Equation (3.2) is a lower bound for the actual critical service level, given that the priority clearing mechanism is employed, that is, all incoming replenishment orders are allocated to the critical class until the inventory on-hand reaches S_c.*

Proof: Since all outstanding replenishments at t will arrive at time $t+L$, we have the following

$$\begin{aligned}I(t)-B_n(t)+R(t,t+H]+R(t+H,t+L] &= S, \text{ or} \\ I(t)+R(t,t+H] = S-R(t+H,t+L]+B_n(t). & \quad (3.3)\end{aligned}$$

In order to write the inventory level at time $t+H$, consider the worst case, i.e., no rationing has ever been performed during the interval $(t, t+H]$ and all non-critical backorders at time t are cleared by time $t+H$. Thus,

$$I(t+H) \geq I(t) + R(t, t+H] - D_c(t, t+H] - D_n(t, t+H] - B_n(t). \tag{3.4}$$

From Equations 3.3 and 3.4, we have

$$I(t+H) \quad \geq \quad S - R(t+H, t+L] - D_c(t, t+H] - D_n(t, t+H]$$

But, by definition, $D_c(t, t+H] - D_n(t, t+H] = S - S_c$. Therefore, we have,

$$I(t+H) = S_c - R(t+H, t+L] + x, \;\; x \geq 0 \tag{3.5}$$

The maximum level of inventory inventory level during the interval $(t+H, t+L]$ is $S_c + x$. Therefore, under a priority clearing mechanism, x is the maximum amount of inventory that could be used to satisfy non-critical demands or to clear non-critical backorders. Hence, we have

$$I(t+L) \quad \geq \quad I(t+H) + R(t+H, t+L] - D_c(t+H, t+L] - x, \;\; \text{or},$$

$$I(t+L) \geq S_c - D_c(t+H, t+L] \tag{3.6}$$

Since, we are conditioning on the event $\{D_c(t+H, t+L] \leq S - S_c - 1\}$, we have,

$$I(t+L) \geq 1 \tag{3.7}$$

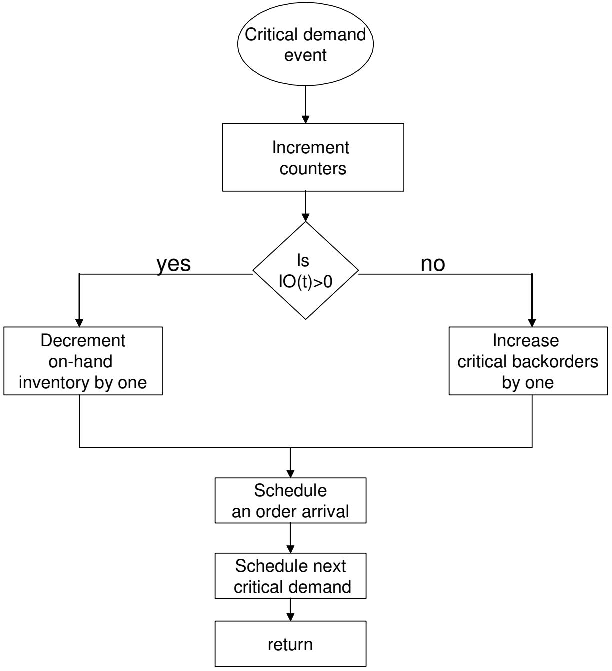
□

Having established this proof, we will test the performance of this approximation with a simulation study in Chapter 4.

3.2 Simulation Model

In this section we present the model of our simulation study. We coded a discrete event simulation algorithm in C with the next-event time advance mechanism to advance the simulation clock. The input parameters are S, S_c, λ_c, λ_n, L and T and the random output parameters are β_c and β_n.

Figure 3.1: Flow diagram of the critical demand arrival event

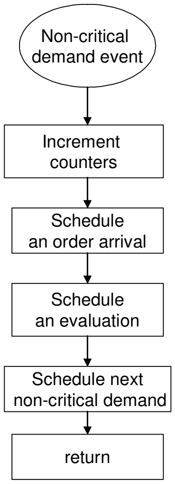

Critical demand event
Increment counters
Is IO(t)>0
yes
no
Decrement on-hand inventory by one
Increase critical backorders by one
Schedule an order arrival
Schedule next critical demand
return

We model the simulation with five events. Besides the end simulation event which terminates the simulation run, we have four other events which are represented by the associated functions in the C code. Next we present the flow charts of these events.

Figure 3.1 describes the critical demand event function. After a critical demand arrival first the counter for the cumulative number of critical demand arrivals is incremented by one. Then the on-hand inventory is checked to see whether or not this arrival can be satisfied immediately. If on-hand inventory is greater than zero and the critical arrival can be satisfied immediately the counter for satisfied critical customers is incremented by one while the on-hand inventory is decremented by one. Otherwise, the counter for critical backorders is incremented by one. Observe that every critical demand arrival event schedules a replenishment order arrival event for L time units after since the inventory is controlled by a base stock policy. Also, it schedules the next critical arrival event.

A non-critical arrival event is similar to a critical arrival event. However the

Figure 3.2: Flow diagram of the non-critical demand arrival event


Non-critical demand event
Increment counters
Schedule an order arrival
Schedule an evaluation
Schedule next non-critical demand
return

only counter that is updated is the counter for the cumulative number of non-critical arrivals (since all non-critical demand arrivals are accepted as they arrive). This is because the due date of such an arrival is T time units after its arrival. Thus another difference of the non-critical arrival event is that it also schedules this evaluation event.

A replenishment event merely represents the arrival of a replenishment order. Thus if there are any critical backorders the counter for critical backorders is decremented by one. If there is a non-critical backorder and the inventory on hand is at S_c the counter for non-critical backorders is decremented by one. Otherwise this replenishment order is used to increment the on-hand inventory by one. A replenishment arrival event also schedules the next replenishment arrival event.

An evaluation event merely determines whether a non-critical arrival from T time units before (i.e., one whose due date has arrived) will be satisfied or not. If inventory on-hand is above S_c the counter for satisfied non-critical customers is incremented by one while the on-hand inventory is decremented by one. Otherwise the counter for non-critical backorders is incremented by one. An evaluation

Figure 3.3: Flow diagram of the replenishment order arrival event

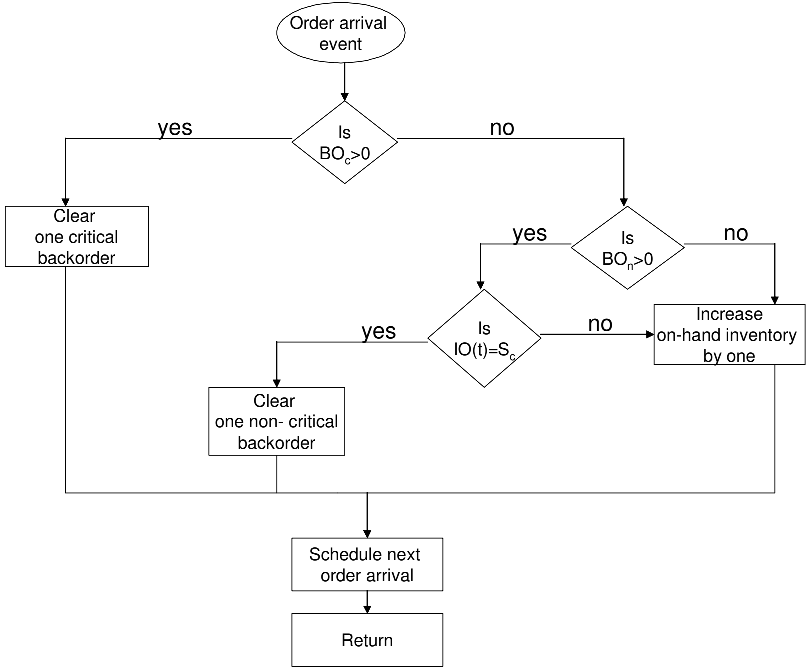
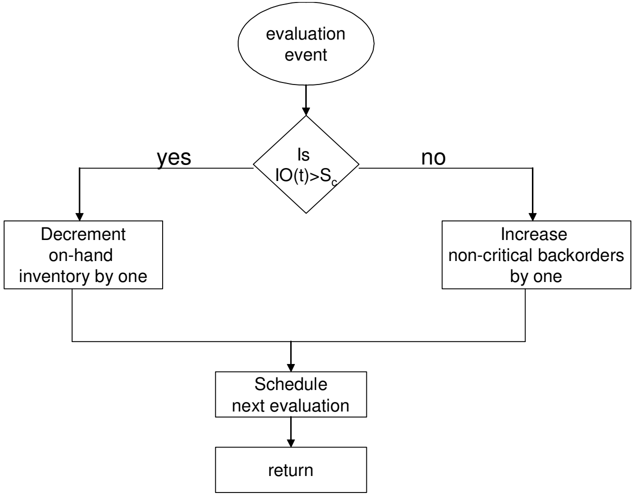

Order arrival event
Is $BO_c>0$
yes
no
Clear one critical backorder
yes
Is $BO_n>0$
no
yes
Is $IO(t)=S_c$
no
Increase on-hand inventory by one
Clear one non- critical backorder
Schedule next order arrival
Return

Figure 3.4: Flow diagram of the evaluation event


evaluation event
Is $IO(t)>S_c$
yes
no
Decrement on-hand inventory by one
Increase non-critical backorders by one
Schedule next evaluation
return

event also schedules the next evaluation event.

The run time of the simulation is 10^7 time units and there is one replication. We also test our simulation model with batch-mean method with 10^5 run time and 100 replications and show that the confidence intervals of our related output parameters are in the order of 10^{-5}. To verify the accuracy of our simulation with a single replication we chose $S = 12$, $S_c = 3$, $\lambda_c = 4$, $\lambda_n = 8$, $L = 0.5$ and $T = 0.1$ with the batch-mean method. To do this we divided the simulation into 100 replications of 10^5 each. We assume independence of successive simulation runs which is acceptable considering the relatively long individual replication lengths of 10^5. As a result we see that the associated confidence intervals of our simulation outputs are 3.26×10^{-5} and 5.60×10^{-4} for the critical and non-critical service levels respectively. Having verified that these confidence intervals are indeed small we conclude that we can confidently use our output from the simulation model with one replication as an approximation for the associated service levels.

3.3 Service Level Optimization

Having established that our approximation is a lower bound for the actual critical service level our approach will be to use this approximation for service level optimization. (In the simulation study section we will show that the results of the simulation study indicate that our approximation for the critical service level works well especially for very high service levels). Thus we will use the approximation for critical service level to solve the following optimization problem:

$$\begin{aligned}
&\min_{S,S_c} S \\
&S.t \\
&\beta_c(S, S_c) \geq \bar{\beta}_c \\
&\beta_n(S, S_c) \geq \bar{\beta}_n \\
&S, S_c \geq 0
\end{aligned}$$

Algorithm 1 The service level optimization algorithm

```
Set $S_{max} := \arg\min\left\{x \geq 0 : \beta_c(x, 0) \geq \bar{\beta}_c\right\}$
Set $S_{min} := \arg\min\left\{x \geq 0 : \beta_n(x, 0) \geq \bar{\beta}_n\right\}$
for $S = S_{min} + 1$ to $S_{max} - 1$ do
    $S_c = S - S_{min}$
    if $\beta_c(S, S_c) \geq \bar{\beta}_c$ then
        $S^* = S$
        $S_c^* = S_c$
        break
    end if
end for
```

The algorithm for the optimization model is presented in Algorithm 1. The algorithm starts by determining S_{max}, the minimum amount of inventory needed to ensure $\bar{\beta}_c$, the minimum service level requirement for the critical demand class. This is the maximum amount of inventory which would satisfy both service level requirements and is found by setting the critical level, S_c equal to zero. Similarly we find S_{min}, the minimum amount of inventory needed to ensure $\bar{\beta}_n$, the minimum service level requirement for the non-critical demand class. We know from Wang et. al. [30], if $S_c = 0$ and $T \leq L$ that $\beta_c = \beta_n$. Thus for $\bar{\beta}_c > \bar{\beta}_c$ we will have $S_{max} > S_{min}$. Knowing this, we enumerate all possible S values from $S_{min} + 1$ to $S_{max} - 1$ for $S_c = S - S_{min}$ to seek a value less than S_{max}. In other words while holding a common pooled inventory of S_{min} and thereby ensuring $\beta_n \geq \bar{\beta}_n$, we search for a possible $S < S_{max}$ which also satisfies $\beta_c \geq \bar{\beta}_c$. Since we know our approximation for the critical service level is only a lower bound, there exist an opportunity to further reduce the base stock level found using the approximated critical service level in the optimization study. To do this we conduct a simulation optimization study for possible (S, S_c) pairs. The results of this study together with the output of the optimization study are provided in Section 4.2.

Chapter 4

Numerical Study

Our numerical study is composed of two parts. In Section 4.1, we test the performance of the approximation for the critical service level that is suggested in Section 3.1 and identify the cases where it can estimate the actual service level with reasonable accuracy. To accomplish this, we use the simulation model that is presented in Section 3.2 which is coded in C and compare the simulated service level with service level calculated through the approximation. Having confirmed that the approximation works well in most cases, we use the approximation in the optimization model to demonstrate the impact of various factors on base stock levels and critical levels in Section 4.2.

4.1 Simulation Study

In this section, we analyze the performance of the approximation for the critical service level with respect to the actual (simulated) service level. This is done again in two steps. First we test the performance of the approximation when the required service level is high, specifically at 99 % and 95 %. Testing the approximation specifically at these levels is useful as high service levels are quite common in industry, especially for critical parts or critical demand classes. Specific testing around 99 % is performed in Section 4.1.1 and specific testing around 95 % is performed in Section 4.1.2. In Section 4.1.3, we allow the critical service level to vary and we test the performance of the approximation by varying a single parameter such as base stock level, arrival rate for the critical demand class, arrival rate for the non-critical demand class and demand lead time. All tables represent the simulated non-critical service level, the exact non-critical service level calculated from the Equation 3.1, the simulated critical service level, the approximation for the critical service level calculated from Equation 3.2, the difference between the simulated service level and the approximation for the critical service level and the percentage difference. The percentage difference is given by the percentage of the difference between the simulated critical service level and the approximation for the critical service level with respect to the simulated service level, that is, 100× (simulation-approximation)/simulation.

4.1.1 Accuracy of the approximation around 99 percent

In Table 4.1, we start with a dataset ($S = 5, S_c = 3, \lambda_n = 4, \lambda_c = 1, L = 0.5, T = 0.1$) that provides a critical service level around 99 %. At each step, the base stock, S, and the critical arrival rate, λ_c, are both increased by a unit to keep the critical service level around 99 percent. As seen from the data, both the simulated and approximated critical service level first decrease and then increase. What is more interesting is that the difference between the simulated and approximated service levels, which is the error of our approximation behaves the opposite way. Furthermore, the difference attains its smallest value where the critical service level attains its highest value. The maximum difference is 0.0085 which confirms that the approximation performs well in this scenario. We also note that the maximum difference between the service level obtained from simulation for the non-critical demand class and the service level calculated using the exact formula presented in Equation 3.1 is 0.0004, which shows that our simulation results can accurately describe the system.

In Table 4.2, we start from another dataset ($S = 8, S_c = 1, \lambda_n = 4, \lambda_c = 2, L = 0.5, T = 0.1$) that provides a critical service level around 99 %. This time, at each

Table 4.1: Performance of the approximation for a fixed critical service level of 99 percent ($S_c = 3$, $\lambda_n = 4$, L=0.5 and T=0.1)

S	λ_c	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
5	1	0.3791	0.3796	0.9995	0.9976	0.0019	0.19
6	2	0.5180	0.5184	0.9981	0.9927	0.0054	0.54
7	3	0.6249	0.6248	0.9968	0.9892	0.0076	0.76
8	4	0.7066	0.7064	0.9962	0.9877	0.0085	0.85
9	5	0.7693	0.7693	0.9958	0.9876	0.0082	0.82
10	6	0.8178	0.8180	0.9958	0.9884	0.0074	0.74
11	7	0.8561	0.8560	0.9960	0.9896	0.0064	0.64
12	8	0.8858	0.8857	0.9964	0.9909	0.0055	0.55
13	9	0.9091	0.9090	0.9967	0.9922	0.0045	0.45
14	10	0.9274	0.9274	0.9971	0.9934	0.0037	0.37
15	11	0.9421	0.9420	0.9975	0.9945	0.0030	0.30
16	12	0.9537	0.9536	0.9978	0.9954	0.0024	0.24

step, the critical level, S_c, and the critical arrival rate, λ_c are both increased by one unit to keep the critical service level around 99 percent. As seen from the data, the approximation works the best when the critical service level is highest. The maximum difference between the approximation and simulation is 0.0553. which still can be considered reasonable.

In Table 4.3, we start from a third dataset ($S = 5, S_c = 3, \lambda_n = 1, \lambda_c = 4, L = 0.5, T = 0.1$) that provides a critical service level around 99 %. At each

Table 4.2: Performance of the approximation for a fixed critical service level of 99 percent ($S = 8$, $\lambda_n = 4$, L=0.5 and T=0.1)

S_c	λ_c	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
1	2	0.9828	0.9828	0.9983	0.9963	0.0020	0.20
2	3	0.9059	0.9057	0.9974	0.9928	0.0046	0.46
3	4	0.7066	0.7064	0.9962	0.9877	0.0085	0.85
4	5	0.4140	0.4142	0.9943	0.9802	0.0141	1.42
5	6	0.1623	0.1626	0.9923	0.9697	0.0226	2.28
6	7	0.0370	0.0372	0.9910	0.9554	0.0356	3.59
7	8	0.0037	0.0037	0.9921	0.9368	0.0553	5.57

Table 4.3: Performance of the approximation for a fixed critical service level of 99 percent ($S_c = 3$, $\lambda_c = 4$, L=0.5 and T=0.1)

S	λ_n	β_n	β_n	β_c	β_c	Difference	%
		(sim)	(exact)	(sim)	(approx)	(sim-approx)	difference
5	1	0.3084	0.3084	0.9499	0.9295	0.0204	2.15
6	2	0.4697	0.4695	0.9801	0.9625	0.0176	1.80
7	3	0.6022	0.6025	0.9915	0.9789	0.0126	1.27
8	4	0.7066	0.7064	0.9962	0.9877	0.0085	0.85
9	5	0.7849	0.7851	0.9981	0.9925	0.0056	0.56
10	6	0.8438	0.8436	0.9990	0.9954	0.0036	0.36
11	7	0.8868	0.8867	0.9995	0.9971	0.0024	0.24
12	8	0.9181	0.9181	0.9997	0.9981	0.0016	0.16
13	9	0.9410	0.9409	0.9999	0.9988	0.0011	0.11
14	10	0.9575	0.9574	0.9999	0.9992	0.0007	0.07

step, the base stock, S, and the non-critical arrival rate, λ_n are both increased by one unit to keep the critical service level around 99 percent. The results are similar to those in Tables 4.1 and 4.2. The approximation still works the best when the critical service level is highest. The maximum difference between the approximation and the simulation is 0.0204.

In Table 4.4, we start from a fourth dataset ($S = 8, S_c = 1, \lambda_n = 4, \lambda_c = 2, L = 0.5, T = 0.1$) that provides a critical service level around 99 %. At each step, the critical level, S_c, and the non-critical arrival rate, λ_n are both increased by one unit to keep the critical service level around 99 percent. The results are similar to those in Tables 4.1, 4.2 and 4.3. The maximum difference between the simulation and the approximation is 0.0085.

4.1.2 Accuracy of the approximation around 95 percent

We repeat the analysis above for a critical service level around 95 %. In Table 4.5, we start from a dataset ($S = 5, S_c = 2, \lambda_c = 4, \lambda_n = 1, L = 0.5, T = 0.1$) that provides a critical service level around 95 %. At each step, the base stock, S, and the critical arrival rate, λ_c are both increased by one unit to keep the critical service level around 95 % this time. Similar to the case with service levels

Table 4.4: Performance of the approximation for a fixed critical service level of 99 percent ($S = 8$, $\lambda_c = 4$, L=0.5 and T=0.1)

S_c	λ_n	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
1	2	0.9755	0.9756	0.9947	0.9925	0.0022	0.22
2	3	0.8946	0.8946	0.9945	0.9885	0.0060	0.60
3	4	0.7066	0.7064	0.9962	0.9877	0.0085	0.85
4	5	0.4333	0.4335	0.9980	0.9898	0.0082	0.82
5	6	0.1856	0.1851	0.9991	0.9929	0.0062	0.62
6	7	0.0479	0.0477	0.9997	0.9957	0.0040	0.40
7	8	0.0056	0.0055	0.9999	0.9977	0.0022	0.22

Table 4.5: Performance of the approximation for a fixed critical service level of 95 percent ($S_c = 2$, $\lambda_n = 1$, L=0.5 and T=0.1)

S	λ_c	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
5	4	0.5696	0.5697	0.9380	0.9190	0.0190	2.03
6	5	0.6705	0.6696	0.9481	0.9339	0.0142	1.50
7	6	0.7440	0.7442	0.9573	0.9467	0.0106	1.11
8	7	0.8000	0.8006	0.9652	0.9573	0.0079	0.82
9	8	0.8433	0.8436	0.9718	0.9658	0.0060	0.62
10	9	0.8765	0.8769	0.9772	0.9726	0.0046	0.47

around 99 %, the the approximation works the best when the critical service level is highest. However, observe that the differences between simulated and approximated service levels attain higher values compared to those for 99 percent due to the decreased critical service level.

In Table 4.6, we start from another dataset ($S = 7, S_c = 1, \lambda_c = 5, \lambda_n = 1, L = 0.5, T = 0.1$) that provides a critical service level around 95 %. At each step, this time, the critical level, S_c, and the critical arrival rate, λ_c are both increased by one unit to keep the critical service level around 95 %. Again, the difference between the simulated and approximated service levels attains its smallest value where the critical service level attains its highest value.

Table 4.6: Performance of the approximation for a fixed critical service level of 95 percent ($S = 7$, $\lambda_n = 1$, L=0.5 and T=0.1)

S_c	λ_c	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
1	5	0.9262	0.9258	0.9761	0.9722	0.0039	0.40
2	6	0.7440	0.7442	0.9573	0.9467	0.0106	1.11
3	7	0.4535	0.4532	0.9321	0.9118	0.0203	2.18
4	8	0.1855	0.1851	0.9040	0.8671	0.0369	4.08
5	9	0.0438	0.0439	0.8832	0.8134	0.0698	7.90
6	10	0.0047	0.0045	0.8864	0.7524	0.1340	15.12

Table 4.7: Performance of the approximation with respect to S ($S_c = 2$, $\lambda_c = 6$, $\lambda_n = 2$, L=0.5 and T=0.1)

S	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
3	0.0225	0.0224	0.6178	0.3642	0.2536	41.05
4	0.1078	0.1074	0.7089	0.5506	0.1583	22.33
5	0.2693	0.2689	0.8124	0.7187	0.0937	11.53
6	0.4742	0.4735	0.8953	0.8437	0.0516	5.76
7	0.6684	0.6678	0.9486	0.9225	0.0261	2.75
8	0.8160	0.8156	0.9773	0.9655	0.0118	1.21
9	0.9094	0.9091	0.9909	0.9861	0.0048	0.48
10	0.9600	0.9599	0.9967	0.9949	0.0018	0.18
11	0.9840	0.9840	0.9989	0.9983	0.0006	0.06
12	0.9942	0.9942	0.9997	0.9995	0.0002	0.02

4.1.3 Accuracy of the approximation with varying system parameters

Tables 4.7 and 4.8 show the impact of the base stock level, S on the critical and non-critical service levels for two different scenarios. As seen from the data in both tables, critical and non-critical service levels both increase as the base stock level increases. We also note that the difference between actual and approximated service level decreases confirming the performance of our approximation for high critical service levels.

Table 4.8: Performance of the approximation with respect to S ($S_c = 1$, $\lambda_c = 1$, $\lambda_n = 5$, L=0.5 and T=0.08)

S	β_n	β_n	β_c	β_c	Difference	%
	(sim)	(exact)	(sim)	(approx)	(sim-approx)	difference
3	0.2672	0.2674	0.9197	0.8244	0.0953	10.36
4	0.5181	0.5184	0.9564	0.9051	0.0513	5.36
5	0.7359	0.7360	0.9799	0.9556	0.0243	2.48
6	0.8776	0.8774	0.9919	0.9818	0.0101	1.02
7	0.9512	0.9510	0.9972	0.9934	0.0038	0.38

Table 4.9: Performance of the approximation with respect to λ_c ($S = 5$, $S_c = 2$, $\lambda_n = 1$, L=1 and T=0.5)

λ_c	β_n	β_n	β_c	β_c	Difference	%
	(sim)	(exact)	(sim)	(approx)	(sim-approx)	difference
1	0.8090	0.8088	0.9950	0.9860	0.0090	0.90
2	0.5439	0.5438	0.9481	0.9008	0.0473	4.99
3	0.3218	0.3208	0.8377	0.7378	0.0999	11.93
4	0.1739	0.1736	0.6961	0.5438	0.1523	21.88
5	0.0886	0.0884	0.5614	0.3668	0.1946	34.66

In Tables 4.9 and 4.10, we study the impact of the critical arrival rate and the non-critical arrival rates, respectively. As we increase both rates, we see that both critical and non-critical service levels deteriorate. As we already observe before, the performance of the approximation also deteriorates as we begin to see low service levels. The difference between the simulated and approximated critical service levels are at unacceptable levels for service levels around 60 %. However, note that these service levels are hardly observed in practice, especially for critical items or for critical demand classes.

In Table 4.11, we study the impact of demand lead time, T. The demand lead time, T starts at 0.10 and is increased by 0.05 at each step, until it is equal to the lead time. This increases both the critical and non-critical service levels. Again, the difference behaves as expected, attaining its smallest value when the critical service level is the highest. We also note that the non-critical service level is quite sensitive to the demand lead time, while the critical service level is not.

Table 4.10: Performance of the approximation with respect to λ_n ($S = 5$, $S_c = 2$, $\lambda_c = 1$, L=0.5 and T=0.1)

λ_n	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
1	0.8090	0.8088	0.9950	0.9860	0.0090	0.90
2	0.9770	0.6767	0.9936	0.9686	0.0250	2.52
3	0.5436	0.5438	0.9928	0.9484	0.0444	4.47
4	0.4227	0.4232	0.9923	0.9274	0.0649	6.54
5	0.3207	0.3208	0.9921	0.9072	0.0849	8.56

Table 4.11: Performance of the approximation with respect to T ($S = 14$, $S_c = 3$, $\lambda_c = 10$, $\lambda_n = 4$, and L=0.5)

T	β_n (sim)	β_n (exact)	β_c (sim)	β_c (approx)	Difference (sim-approx)	% difference
0.10	0.9274	0.9274	0.9971	0.9934	0.0037	0.37
0.15	0.9387	0.9386	0.9978	0.9943	0.0035	0.35
0.20	0.9486	0.9486	0.9983	0.9953	0.0030	0.30
0.25	0.9574	0.9574	0.9987	0.9964	0.0023	0.23
0.30	0.9651	0.9651	0.9990	0.9973	0.0017	0.17
0.35	0.9717	0.9718	0.9993	0.9980	0.0013	0.13
0.40	0.9775	0.9775	0.9994	0.9986	0.0008	0.08
0.45	0.9823	0.9823	0.9995	0.9990	0.0005	0.05
0.50	0.9946	0.9863	0.9995	0.9993	0.0002	0.02

Having tested the performance of the approximation in a variety of settings, we can conclude that, with a reasonable accuracy, our approximation can be used to estimate the actual service levels for the critical demand class when a priority clearing mechanism is used. We also show computationally that the service level obtained through approximation is always lower than the actual service level for the critical demand class, which confirms our analytical proof in Chapter 3. We finally note that the performance of the approximation improves as the service level for the critical demand class increases which is in line with high service level needs for critical demand classes. This can be explained as follows: when the service level for the critical class is high, the impact of the way incoming replenishment orders are handled is less pronounced as there are not many backorders for the critical class. When the service level for the critical class decreases the performance of our approximation deteriorates, as it fails to capture the effect of incoming replenishment orders exactly.

4.2 Optimization Study

In this section, we present the output of our optimization and simulation optimization study with the aim of demonstrating that a system with rationing (although using our approximation for the critical service level) can result in significant inventory savings compared to a one without rationing. Through Tables 4.12-4.22 we present our results for various input parameters. In all tables, the first column represents the input parameter in consideration. The second column represents the required base stock level if no rationing is used (demand lead times are still recognized). Observe that this base stock level will be determined by the higher service level requirement which is the critical service level requirement in our case (although we recognize demand lead times the policy without rationing is still a round-up policy where demand from multiple classes is pooled). The third and fourth columns represent the base stock level and the critical level that are found through the optimization study using the approximation for the critical service level. The fifth column represents the percentage saving resulting from using a rationing policy that uses the approximation for the critical service level compared to a policy where no rationing is used (100×(column2-column3)/column2). The sixth and seventh columns represent the base stock level and the critical level that are found through the optimization study that uses the simulation results for the service level for the critical demand class. As our simulation runs are shown to represent the system accurately, the sixth and seventh columns are in fact true optimal values of the base stock level and the critical level. The eighth column represents the percentage saving resulting from using a rationing policy that uses the simulation results for the critical service level compared to a policy where no rationing is used (100×(column2-column6)/column2). The ninth and eleventh columns represent the service levels for the critical and non-critical demand classes obtained from simulation for the optimal base stock and critical levels. The tenth column represents the service level estimated by the approximation, again for the optimal base stock and critical levels.

In Table 4.12, λ_n is increased by one unit from 1 to 10 while λ_c is kept fixed

Table 4.12: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 1$, L=0.5, T=0.1, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.80$)

λ_n	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
1	5	4	1	20.00	4	1	20.00	0.9933	0.9903	0.9372
2	6	5	2	16.67	5	2	16.67	0.9988	0.9972	0.8570
3	6	6	0	0.00	5	1	16.67	0.9926	0.9852	0.9067
4	7	6	2	14.29	6	2	14.29	0.9992	0.9971	0.8386
5	8	7	2	12.50	6	1	25.00	0.9930	0.9830	0.8912
6	8	7	2	12.50	7	2	12.50	0.9994	0.9972	0.8317
7	9	8	2	11.11	7	1	22.22	0.9937	0.9820	0.8828
8	10	8	2	20.00	7	1	30.00	0.9908	0.9731	0.8301
9	10	9	2	10.00	8	1	20.00	0.9943	0.9817	0.8786
10	11	9	2	18.18	8	1	27.27	0.9921	0.9739	0.8310

at 1. Note that we can reach the optimal solution for four cases using our approximation, whereas in other cases there is only a single unit difference between the optimal base stock level and the result from our optimization study. The fine performance of the optimization study that uses approximation is attributed to the relatively small arrival rates and lead time demands. In addition, observe that rationing tends to create more savings when the arrival rate in the non-critical demand class is significantly higher than the arrival rate in the critical demand class, although there is no uniformity in this behavior.

In Table 4.13, λ_n is increased by one unit from 1 to 10, while λ_c is kept fixed at 1. Note that the non-critical service level requirement is 90 percent now. We can reach the optimal solution (base stock level) for six cases using our approximation whereas the difference is only a single unit in other cases. Again, the fine performance of the optimization study that uses approximation is due to the relatively small arrival rates and lead time demands. The improved performance of our optimization algorithm can be attributed to the fact that a higher non-critical service level requirement of 90 percent increases S_{min} and thus decreases the number of possible (S, S_c) pairs. As a result, our algorithm becomes more precise by attaining the optimal values for two more cases. Again, observe that rationing tends to create more savings when the arrival rate in the

Table 4.13: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 1$, L=0.5, T=0.1, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.90$)

λ_n	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
1	5	4	1	20.00	4	1	20.00	0.9933	0.9903	0.9372
2	6	5	1	16.67	5	1	16.67	0.9965	0.9937	0.9568
3	6	6	0	0.00	5	1	16.67	0.9926	0.9852	0.9068
4	7	6	1	14.29	6	1	14.29	0.9959	0.9907	0.9378
5	8	7	1	12.50	7	1	12.50	0.9976	0.9940	0.9580
6	8	8	0	0.00	7	1	12.50	0.9959	0.9890	0.9258
7	9	8	1	11.11	8	1	11.11	0.9975	0.9928	0.9490
8	10	9	2	10.00	8	1	20.00	0.9961	0.9880	0.9182
9	10	9	1	10.00	9	1	10.00	0.9976	0.9920	0.9427
10	11	10	2	9.09	9	1	18.18	0.9964	0.9875	0.9134

non-critical demand class is significantly higher than the arrival rate in the critical demand class.

In Table 4.14, λ_n is increased one unit from 10 to 20, while λ_c is kept fixed at 5. As a result of the relative increase in the arrival rates and lead time demands, the performance of the approximation deteriorates and the base stock levels that are determined through approximation are always higher than the optimal base stock levels. Again, observe that rationing tends to create more savings when the arrival rate in the non-critical demand class is significantly higher than the arrival rate in the critical demand class.

In Table 4.15, λ_n is increased one unit from 10 to 20 while λ_c is kept fixed at 5. Note that that the service level for the non-critical demand class is set at 90 percent now. Despite this difference, the results remain same as in Table 4.14. As a result of the relative increase in the arrival rates and lead time demands, the optimization algorithm attains the optimal value only in two cases; for all other cases, the difference between the optimal values and our optimization output is only one unit. In addition, observe that savings are decreased compared to Table 4.14 due to the increase in $\bar{\beta}_n$. This effect will be clearer in the next two tables.

In Table 4.16, $\bar{\beta}_c$ is increased from 90 percent to 99.5 percent while $\bar{\beta}_n$ is kept

Table 4.14: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, L=0.5, T=0.1, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.80$)

λ_n	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
10	14	13	3	7.14	12	2	14.29	0.9962	0.9824	0.8775
11	15	13	3	13.33	12	2	20.00	0.9952	0.9758	0.8406
12	15	14	3	6.67	13	2	13.33	0.9969	0.9827	0.8789
13	16	14	3	12.50	13	2	18.75	0.9961	0.9767	0.8445
14	16	14	3	12.50	13	2	18.75	0.9952	0.9694	0.8058
15	17	15	3	11.76	14	2	17.65	0.9968	0.9776	0.8486
16	18	15	3	16.67	14	2	22.22	0.9961	0.9710	0.8124
17	18	16	3	11.11	15	2	16.67	0.9973	0.9785	0.8527
18	19	16	3	15.79	15	2	21.05	0.9967	0.9724	0.8189
19	19	17	3	10.53	16	2	15.79	0.9977	0.9793	0.8569
20	20	17	3	15.00	16	2	20.00	0.9973	0.9738	0.8252

Table 4.15: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, L=0.5, T=0.1, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.90$)

λ_n	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
10	14	13	2	7.14	13	2	7.14	0.9982	0.9915	0.9332
11	15	14	3	6.67	13	2	13.33	0.9976	0.9876	0.9085
12	15	14	2	6.67	14	2	6.67	0.9985	0.9913	0.9320
13	16	15	3	6.25	14	2	12.50	0.9980	0.9877	0.9085
14	16	15	2	6.25	14	1	12.50	0.9902	0.9714	0.9312
15	17	16	3	5.88	15	2	11.76	0.9983	0.9878	0.9091
16	18	16	2	11.11	15	1	16.67	0.9909	0.9712	0.9311
17	18	17	3	5.56	16	2	11.11	0.9985	0.9880	0.9098
18	19	17	2	10.53	16	1	15.79	0.9915	0.9711	0.9313
19	19	18	3	5.26	17	2	10.53	0.9987	0.9882	0.9112
20	20	18	2	10.00	17	1	15.00	0.9921	0.9712	0.9317

Table 4.16: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, $\lambda_n = 10$, L=2, $T = 0.5$ and $\bar{\beta}_n = 0.80$)

$\bar{\beta}_c$	S	S	S_c	% saving	$\mathbf{S}^*$	$\mathbf{S}_c^*$	% saving	β_c	β_c(app)	β_n
0.900	33	32	2	3.03	31	1	6.06	0.9544	0.8179	0.8650
0.925	33	33	0	0.00	31	1	6.06	0.9544	0.8179	0.8650
0.950	34	34	0	0.00	31	1	8.82	0.9544	0.8179	0.8650
0.970	36	35	5	2.78	32	2	11.11	0.9889	0.9061	0.8179
0.980	37	35	5	5.41	32	2	13.51	0.9889	0.9061	0.8179
0.985	37	36	6	2.70	32	2	13.51	0.9889	0.9061	0.8179
0.990	38	36	6	5.26	33	3	13.16	0.9973	0.9401	0.8179
0.995	40	37	7	7.50	33	3	17.50	0.9973	0.9401	0.8179

Table 4.17: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 10$, $\lambda_n = 5$, L=2, $T = 0.5$ and $\bar{\beta}_n = 0.80$)

$\bar{\beta}_c$	S	S	S_c	% saving	$\mathbf{S}^*$	$\mathbf{S}_c^*$	% saving	β_c	β_c(app)	β_n
0.900	35	35	0	0.00	34	1	2.86	0.9148	0.8722	0.8309
0.925	36	36	0	0.00	35	2	2.78	0.9580	0.9057	0.8308
0.950	37	37	0	0.00	35	2	5.41	0.9580	0.9057	0.8308
0.970	39	39	0	0.00	36	3	7.69	0.9799	0.9322	0.8309
0.980	40	40	0	0.00	37	4	7.50	0.9905	0.9526	0.8309
0.985	40	40	0	0.00	37	4	7.50	0.9905	0.9526	0.8309
0.990	41	41	0	0.00	37	4	9.76	0.9905	0.9526	0.8309
0.995	43	42	9	2.33	38	5	11.63	0.9956	0.9679	0.8309

fixed at 80 percent. It can easily be seen that rationing becomes more effective as the critical service level requirement increases. Table 4.17 is the same as Table 4.16, except that the critical and non-critical arrival rates are reversed. Again rationing becomes more instrumental and results in more savings at higher critical service levels. But this time the savings are less compared to the previous savings since the critical arrival rate is higher compared to the non-critical arrival rate.

In Table 4.18, the demand lead time T is increased from 0.4 to 2.0 (which is equal to the replenishment lead time). The obvious result of this is that all base stock parameters decrease as a result of the positive effect of demand lead

Table 4.18: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, $\lambda_n = 10$, L=2, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.80$)

T	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
0.4	40	37	6	7.50	34	3	15.00	0.9972	0.9438	0.8136
0.8	35	34	7	2.86	29	2	17.14	0.9903	0.9136	0.8324
1.2	30	30	0	0.00	25	2	16.67	0.9924	0.9318	0.8550
1.6	24	24	0	0.00	20	2	16.67	0.9914	0.9235	0.8271
2.0	19	19	0	0.00	17	3	10.53	0.9971	0.9730	0.9165

Table 4.19: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, $\lambda_n = 20$, L=2, $\bar{\beta}_c = 0.995$ and $\bar{\beta}_n = 0.80$)

T	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
0.1	68	60	5	11.76	57	2	16.18	0.9955	0.9503	0.8270
0.2	65	58	5	10.77	55	2	15.38	0.9957	0.9343	0.8319
0.3	63	57	6	9.52	53	2	15.87	0.9959	0.9235	0.8370
0.4	61	55	7	9.84	50	2	18.03	0.9950	0.8950	0.8041
0.5	58	53	7	8.62	48	2	17.24	0.9952	0.8920	0.8096
0.6	56	52	8	7.14	44	2	21.43	0.9955	0.8925	0.8156
0.7	53	50	8	5.66	42	2	20.75	0.9957	0.8953	0.8218
0.8	51	48	8	5.88	42	2	17.65	0.9959	0.8997	0.8284
0.9	48	47	9	2.08	40	2	16.67	0.9962	0.9052	0.8353
1.0	46	45	9	2.17	38	2	17.39	0.9964	0.9113	0.8427

time on inventory. However as the demand lead time increases, we also observe that rationing becomes less effective and reductions in base stock levels through rationing are limited. The same effect is also observed in Table 4.19, but here the savings related to rationing are higher due to the increased non-critical arrival rate.

Tables 4.20, 4.21 and 4.22 demonstrate the same effect of demand lead time on savings that could be achieved as a result of rationing, however this time this effect is not that noticeable due to the decreased arrival rates.

Table 4.20: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, $\lambda_n = 10$, L=0.5, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.80$)

T	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
0.1	14	13	3	7.14	12	2	14.29	0.9962	0.9824	0.8773
0.2	13	12	4	7.69	10	2	23.08	0.9940	0.9593	0.8095
0.3	11	11	0	0.00	9	2	18.18	0.9955	0.9628	0.8310
0.4	9	9	0	0.00	8	2	11.11	0.9965	0.9734	0.8576
0.5	8	8	0	0.00	7	2	12.50	0.9947	0.9858	0.9580

Table 4.21: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 5$, $\lambda_n = 5$, L=0.5, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.80$)

T	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
0.1	11	10	3	9.09	10	3	9.09	0.9979	0.9922	0.8311
0.2	10	10	0	0.00	9	2	10.00	0.9946	0.9813	0.8894
0.3	9	9	0	0.00	8	2	11.11	0.9929	0.9741	0.8577
0.4	9	9	0	0.00	8	3	11.11	0.9975	0.9882	0.8153
0.5	8	8	0	0.00	7	2	12.50	0.9917	0.9858	0.9579

Table 4.22: Optimal Parameters: Approximation vs Simulation ($\lambda_c = 10$, $\lambda_n = 5$, L=0.5, $\bar{\beta}_c = 0.99$ and $\bar{\beta}_n = 0.80$)

T	S	S	S_c	%	$\mathbf{S}^*$	$\mathbf{S}_c^*$	%	β_c	β_c(app)	β_n
				saving			saving			
0.1	15	14	4	6.67	13	3	13.33	0.9920	0.9793	0.8305
0.2	14	14	0	0.00	13	3	7.14	0.9952	0.9848	0.8774
0.3	13	13	0	0.00	12	3	7.69	0.9936	0.9800	0.8473
0.4	13	13	0	0.00	11	3	15.38	0.9906	0.9748	0.8095
0.5	12	12	0	0.00	11	3	8.33	0.9922	0.9863	0.9319

Through this optimization study, we have two main observations: one is related to the performance of our optimization algorithm which uses our approximation for the critical service level and the other is related to the cases where rationing is more effective. We first note that the optimization algorithm that uses the approximation can reach the optimal values for cases where the arrival rates and thus lead time demands are relatively low (slow moving items). In these specific cases, the optimal amount of base stock required to ensure the required service levels is low and there is a relatively small opportunity to save from rationing (one or two units) and our optimization algorithm can capture those savings. However as the demand rates become larger, more inventory is required to ensure the required service levels and hence there is more opportunity to incur inventory savings from rationing. In these cases, our optimization algorithm with approximation usually cannot obtain the optimal base stock levels as it fails to capture the important effect of incoming replenishment orders. Despite this fact, our extremely faster optimization algorithm with approximation usually misses the optimal base stock level by only one or two units: an impressive performance. In addition, observe that the results of our optimization algorithm with approximation serves another important role: the base stock levels that are obtained through approximation is used as an upper bound for the simulation optimization which would otherwise be even slower.

Another important observation is regarding cases where rationing (both our approximate results and the optimal) is more effective in saving inventory (in the form of base stock since we assume ownership of on-order inventory). Rationing is more effective in cases where the non-critical arrivals are dominant in the arrival mix. That is, rationing tends to be more effective in cases where λ_n is large compared to λ_c. This is intuitive, because there are more opportunities to ration when you have more non-critical arrivals in the arrival mix. In this case, a policy without rationing becomes more inefficient, since a large fraction of customers will be supported by a higher service level than required. On the other hand a rationing policy will utilize the increased proportion of customers who tolerate a lower service level through its ability to differentiate service and save inventory. A similar observation is made regarding the service level requirements, $\bar{\beta}_c$ and $\bar{\beta}_n$.

When all other parameters are kept the same, rationing tends to be more effective in cases where $\bar{\beta}_c$ is large as compared to $\bar{\beta}_n$. In other words, rationing becomes more effective by differentiating service when the service level requirements are significantly different. On the other hand, a policy without rationing is ineffective when the difference between $\bar{\beta}_c$ and $\bar{\beta}_n$ is high, as non-critical demand class gets a service level much higher than required.

Finally, note that our optimization algorithm which uses the approximation for the critical service level is significantly faster than the simulation optimization. On a 2 Ghz Pentium 4 processor, the run time of the optimization algorithm is 5 minutes on the average, while the average run time for a single simulation is 60 minutes. Because the simulation optimization runs several simulations to find the actual critical service levels for different (S, S_c) pairs, this duration can increase to several hours depending on system parameters. Considering the small differences between the results of our optimization algorithm and the optimal output parameters, we conclude that our optimization algorithm indeed performs very well by capturing most of the savings due to rationing while the simulation optimization can be a computational burden especially in systems where there is an extensive number of items to manage.

Chapter 5

Case Study

In this chapter, we attempt to verify the significance of our results through a case study at the semiconductor equipment manufacturer which we briefly described in Chapter 1. We have selected a depot in North America that is serving a number of customers for both down demand and lead time demand. As described earlier, the depot is positioned to provide a 4-hour service to a specific list of customer locations for their down orders. Considering the shipment time around 4 hours to its customer locations, this means that the down orders need to be satisfied immediately from on stock inventory (i.e., demand lead time is zero). The depot is also used to support maintenance orders from the same list of customers. However, in this case, the parts do not have to be shipped right away. The customers usually plan such maintenance activities in advance, and demand lead times of 2 weeks are common and acceptable for such orders.

We selected 64 parts for our study. A summary of characteristics for these parts is given in Table 5.1. In order to ensure the appropriateness of the $(S-1, S)$ inventory policy and the validity of the Poisson demand assumption, we include rather expensive and slow moving parts in our study. We used the demand history of a 12 months period in years 2001 and 2002 and include all requested orders (these could include orders that were not satisfied or canceled later) whose primary source is the depot we have selected. The ratio of critical orders to total orders vary at the part level. On the average, 52.2 % of a part's demand is from down orders (i.e., critical demand).

Table 5.1: Part Characteristics

	Min	Max	Average
Part Cost ($)	1,104	40,451	8,681
Critical Annual Demand	1	166	43.19
Non-critical Annual Demand	2	120	39.59
Total Annual Demand	41	212	82.78
Percentage of Critical Demand	1.18	96.77	52.20
COGS ($)	94,985	3,318,438	643,600
Lead Time (days)	19	120	68.06
Lead Time Demand	10.06	19.65	13.99

In the same 12 months period, these 64 parts had a sales volume of $ 41.2 million (in cost). $ 24.3 million (59.1 %) of this is generated by orders that are denoted by customers as down orders; $ 16.9 million (40.9 %) of this is generated by orders that are denoted by customers as lead time orders. We note again that with company's current practice, the demand lead times are not recognized by the company and the safety stocks are set to satisfy service level requirements for the down orders while considering the total demand (down orders and lead time orders).

We test a number of service level targets for down demand as the company may change these targets depending on its negotiation with its customers that are served through this particular depot. We also note that setting service level targets for lead time demand alone is not an established practice for the company, as the current practice provides a service level which is same as the service level targeted for down demand (as noted in [30]). Therefore, we also test a number of service level targets for lead time demand. However, we set the service level targets for the lead time demand always less than the service level targets for the down demand which is in line with company's and customers' expectations.

The analysis is done in three steps. In the first step, we do not recognize the demand lead time for lead time orders and we do not apply any rationing. We simply calculate the minimum base stock levels that will satisfy the target service level requirement for the down orders considering the total demand (down demand plus lead time demand). This reflects the current practice in the company. In the second step, we recognize the demand lead time for lead time orders, however do not use any rationing to provide differentiated service to the two type of demand classes. We calculate the minimum base stock levels that will satisfy the target service level requirement for the down orders. This is similar to the model in [30] and in fact lead time orders and down orders get the same service level in this case. Finally, in the third step, we recognize the demand lead times and also use rationing to provide differentiated service to two demand classes. In this analysis, we use the approximation that is derived in Section 3.1, as this is proven to be an effective procedure in Section 4.1. The procedure is also easy to implement which is important for a company that needs to manage 50,000 or more parts across more than 70 locations in the world.

We demonstrate our analysis for a particular part in Table 5.2. The third column in the table shows the minimum base stock levels to satisfy the service level requirement for the critical demand class when one does not recognize the demand lead times and does not apply any rationing. The fourth column in the table shows the minimum base stock levels to satisfy the service level requirement for the critical demand class when one recognizes the demand lead times, but does not apply any rationing. The fifth column in the table shows the percentage saving for this case over the no demand lead time case, that is, 100×(column3-column4)/column3. The sixth and seventh columns in the table show the minimum base stock levels and the corresponding critical levels to satisfy the service level requirements for the critical demand class and non-critical demand class, individually when one recognizes the demand lead times and also uses rationing (which uses the approximation for the critical service level). The eighth column in the table shows the percentage saving for this case over the no demand lead time case, that is, 100×(column3-column6)/column3. This part is a $ 3,530 part with a lead time of 86 days. For the 12 month period in this analysis, the down demand was 12 units, and the lead time demand was 41 units.

When the critical and non-critical service levels are 90 % and 80 %, respectively, recognizing demand lead time for the non-critical demand class generates

Table 5.2: Part Example

Service Level (%)		No Demand Lead Time	No Rationing		Rationing		
Critical	Non-critical	S	S	% saving	S	S_c	% saving
90	80	18	16	11.11	16	0	11.11
95	80	20	18	10.00	17	2	15.00
97	80	21	19	9.52	17	2	19.05
99	80	23	20	13.04	18	3	21.74
99	85	23	20	13.04	18	3	21.74
99	90	23	20	13.04	19	3	17.39
99	95	23	20	13.04	20	0	13.04

2 units savings in base stock levels. The use of rationing is not very useful here as the service level difference is not significant in this setting. However as the service level is increased for the critical demand class, we see savings through rationing and continue to see savings through recognition of demand lead times for the non-critical class. The maximum saving through recognition of demand lead times is achieved when the service level for the critical demand classes is highest. The maximum saving through rationing is achieved when the service level difference between critical and non-critical demand classes is highest. After we reach the critical and non-critical service levels of 99 % and 80 %, respectively, we start to increase the service level of the non-critical demand class, while keeping the critical service level constant. We observe that base stock levels are same if we do not apply any rationing, and the impact of rationing disappears as the non-critical service level approaches critical service level.

A similar analysis is done for all 64 parts. Tables 5.3 and 5.4 show the dollar value of base stock levels (in thousands dollars) for three different approaches. We see that recognizing the demand lead times and using rationing to differentiate service levels generate significant savings to the company for these 64 parts. When the critical service level is 99 % and the non-critical service level is 80 %, recognizing demand lead times saves 7.4 % on base stock levels (which is equal to the inventory investment, once we assume that the pipeline stocks are owned by the company); additional 4.1 % is saved once the company starts rationing (even

Table 5.3: Impact of Critical Service Level

SL (%)		No Demand	No	%	Rationing			
		Lead Time	Rationing	saving	Approx	%	Exact	%
$\bar{\beta}_c$	$\bar{\beta}_n$	(\$000)	(\$000)		(\$000)	saving	(\$000)	saving
99	80	14,050	13,009	7.41	12,432	11.52	11,778	16.17
97	80	12,930	12,007	7.14	11,652	9.88	11,041	14.61
95	80	12,294	11,450	6.87	11,233	8.63	10,762	12.46
90	80	11,449	10,636	7.10	10,563	7.74	10,197	10.94

Table 5.4: Impact of Non-critical Service Level

SL (%)		No Demand	No	%	Rationing			
		Lead Time	Rationing	saving	Approx	%	Exact	%
$\bar{\beta}_c$	$\bar{\beta}_n$	(\$000)	(\$000)		(\$000)	saving	(\$000)	saving
99	80	14,050	13,009	7.41	12,432	11.52	11,778	16.17
99	85	14,050	13,009	7.41	12,591	10.38	11,952	14.93
99	90	14,050	13,009	7.41	12,691	9.67	12,140	13.59
99	95	14,050	13,009	7.41	12,804	8.87	12,511	10.95

though we use an approximation for the service level of the critical demand class) to provide differentiated services to two types of demand. As the critical service level declines to approach the non-critical service level, we see that savings due to the recognition of demand lead times are still significant, while the impact of rationing is less pronounced. In the last two columns of Tables 5.3 and 5.4, we also report the dollar value of base stocks when we use rationing with the exact values of the service level for the critical demand class derived through simulation and percentage savings over the no demand lead time case.

We conclude that the recognition of the demand lead times and the use of rationing create significant savings for the company. This is true even when we use an approximation to estimate the service level for the critical demand class. More savings are obviously possible if we can accurately determine the service level for the critical demand class. However, the approximation is easy to implement (which is necessary for this particular company) and as it is shown here, its performance is quite reasonable.

Chapter 6

Conclusion

In this thesis, we consider a single echelon spare part distribution system. Our research is motivated by our experience with a leading semiconductor equipment manufacturer. This manufacturer faces two kinds of orders: down orders that result from the equipment failures of the customers and lead time orders that result from the scheduled maintenance activities of customers. The down orders must be supplied immediately while the lead time orders are needed to be supplied at a future date (demand lead time). Currently, the company uses a common pool of inventory controlled by a base stock policy. However, it neither recognizes the demand lead times for the lead time demand, nor it treats the down demand and lead time demand differently, in order to provide different service levels. Since unused semiconductor manufacturing capacity in customers is very costly, we need a policy that could favor down orders. Therefore, we model the system as a single echelon inventory model where down orders are considered as the critical demand class, while the lead time orders are considered as the non-critical class and propose a static rationing policy that would ration the demand from non-critical class. The model aims to satisfy the minimum service level requirements for both demand classes, while using less inventory than the policy the company currently uses.

We develop an approximation for the critical service level and prove that this approximation is essentially a lower bound for the critical service level. Furthermore we conduct a simulation study to test the performance of our approximation versus the actual (simulated) critical service level and show that our approximation performs well specifically for high critical service levels which is in line with the needs of a critical demand class. As a result of our optimization study, we show that rationing the non-critical orders indeed results in significant savings in terms of base stock inventory. Savings are also verified in a case study where we use real data from the semiconductor equipment manufacturer we mentioned earlier.

We also present the situations where the rationing policy is most useful. The rationing policy is more effective (by saving more base stock compared to a policy without rationing) when the non-critical arrival rate dominates the critical arrival rate and the critical service level requirement dominates the non-critical service level requirement. Obviously more savings are possible through using the actual service levels from simulation. However, our optimization algorithm which is extremely faster than simulation optimization captures most of the savings due to rationing and hence will be more effective especially in systems where the number of parts is extensive (as in the semiconductor equipment manufacturer we consider).

During this research, we combine the concept of demand lead time and inventory rationing, both of which have proven to be cost effective for inventory systems. However, to our knowledge our study is the first to simultaneously consider rationing and demand lead time. Our numerical results indicate that such a practice indeed results in significant inventory and cost savings.

Future research can extend the analysis here in many directions. Although we were motivated by a system with two demand classes, considering the possibility of more demand classes would be an appropriate extension. In addition, considering several deterministic demand lead times would be interesting. In some cases the supply or demand lead time might be stochastic, hence this could be a logical extension. Obviously most inventory systems have a multi-echelon structure and therefore extending the analysis to a multi-echelon setting would be a realistic approach. Another extension is concerning the definition of the critical demand class. In some inventory systems, orders with a positive demand lead time might constitute the critical class, especially if a commitment is made to such orders. In that case, rationing the orders with zero lead time would be more appropriate. Finally, a cost optimization scheme could be considered which would require the derivation of expressions for long run on-hand inventory and average backorders for both customer classes.

Bibliography

[1] D. Atkins and K. K. Katircioglu. Managing inventory for multiple customers requiring different levels of service. Working Paper 94-MSC-015, University of British Columbia, Vancouver, BC, 1995.

[2] S. Axsater. Continuous review policies for multi-level inventory systems with stochastic demand. In S. C. Graves, A. R. Kan, and P. Zipkin, editors, *Logistics of Production and Inventory*, volume 4 of *Handbooks in OR & MS*. Elsevier, Amsterdam, 1993.

[3] M. A. Cohen, P. Kleindorfer, and H. L. Lee. Service constrained (s, s) inventory systems with priority demand classes and lost sales. *Management Science*, 34:482–499, 1989.

[4] M. A. Cohen, Y. S. Zheng, and Y. Wang. Identifying opportunities for improving teradyne's service-parts logistics system. *Interfaces*, 29(4):1–18, 2000.

[5] R. Dekker, R. M. Hill, M. J. Kleijn, and R. H. Teunter. On the (s-1,s) lost sales inventory model with priority demand classes. *Naval Research Logistics*, 49:593–610, 2002.

[6] R. Dekker, M. J. Kleijn, and P. J. D. Rooij. A spare parts stocking system based on equipment criticality. *Int J. Prod. Econ.*, 56–57:69–77, 1998.

[7] V. Deshpande, M. A. Cohen, and K. Donohue. A threshold inventory rationing policy for service-differentiated demand classes. *Management Science*, 49:683–703, 2003.

[8] R. V. Evans. Sales and restocking policies in a single item inventory system. *Management Science*, 14:463–472, 1968.

[9] G. J. Feeney and C. C. Sherbrooke. The (s -1, s) inventory policy under compound poisson demand. *Management Science*, 12:391–411, 1966.

[10] K. C. Frank, R. Q. Zhang, and I. Duenyas. Optimal policies for inventory systems with priority demand classes. *Operations Research*, 51:993–1002, 2003.

[11] S. C. Graves. A multi-echelon inventory model for a repairable item with one-for-one replenishment. *Management Science*, 31:1247–1256, 1985.

[12] A. Y. Ha. Inventory rationing in a maketostock production system with several demand classes and lost sales. *Management Science*, 43:1093–1103, 1997.

[13] A. Y. Ha. Stockrationing policy for a maketostock production system with two priority classes and backordering. *Naval Reserach Logistics*, 44:457–472, 1997.

[14] A. Y. Ha. Stock rationing in an m/ek/1 maketostock queue. *Management Science*, 46:77–87, 2000.

[15] R. Hariharan and P.Zipkin. Customer-order information, lead-times, and inventory. *Management Science*, 41:1599–1607, 1995.

[16] A. Kaplan. Stock rationing. *Management Science*, 15:260–267, 1969.

[17] M. J. Kleijn and R. Dekker. An overview of inventory systems with several demand classes. Technical Report 9838/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, September 1998.

[18] P. M. Melchiors, R. Dekker, and M. J. Kleijn. Inventory rationing in an (s, q) inventory model with lost sales and two demand classes. *J. Oper Res. Soc.*, 51(1):111–122, 1998.

[19] K. Moinzadeh and P. K. Aggarwal. An information-based multi-echelon inventory system with emergency orders. *Operations Research*, 45:694–701, 1997.

[20] I. Moon and S. Kang. Rationing policies for some inventory systems. *J. Oper. Res. Soc.*, 49(5):509–518, 1998.

[21] S. Nahmias. Managing repairable item inventory systems. In L. B. Schawarz, editor, *Multi-Level Production/Inventory Control Systems*, volume 16 of *TIMS Studies in Management Science*, pages 253–277. North-Holland, Amsterdam, 1981.

[22] S. Nahmias and W. Demmy. Operating characteristics of an inventory system with rationing. *Management Science*, 27:1236–1245, 1981.

[23] C. Palm. Analysis of the erlang tra. formula for busy-signal arrangements. *Ericsson Technics*, 4:39–58, 1938.

[24] H. E. Scarf. Stationary operating characteristics of an inventory model with time lag. In K. J. Arrow, S. Karlin, and H. Scarf, editors, *Studies in the Mathematical Theory of Inventory and Production*, chapter 16. Stanford University Press, Stanford, California, 1958.

[25] C. C. Sherbrooke. Metric: A multi-echelon technique for recoverable item control. *Operations Research*, 16:122–141, 1968.

[26] K. F. Simpson. In-process inventories. *Operations Research*, 6:863–872, 1958.

[27] R. H. Teunter and W. K. K. Haneveld. Reserving spare parts for critical demand. Technical report, Graduate School/Research Institute System, Organizations and Management (SOM), University of Groningen, 1996.

[28] D. M. Topkis. Optimal ordering and rationing policies in a non-stationary dynamic inventory model with n demand classes. *Management Science*, 15:160–176, 1968.

[29] A. F. J. Veinott. Optimal policy in a dynamic, single product, non-stationary inventory model with several demand classes. *Operations Research*, 13:761–778, 1965.

[30] Y. Wang, M. A. Cohen, and Y. S. Zheng. Differentiating customer service on the basis of delivery lead times. *IIE Transactions*, 34:979–989, 2002.

Appendix A

Code

A.1 Code of the simulation study

```
/* This C program simulates a one-for-one replenishment inventory system with two
under a static rationing policy and a priority clearing mechanism.
It calculates the type I service levels for each class*/

/*external definitions for the program*/

#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include "lcgrand.h" /* Header file for random-number generator. */
#include "approximation.h" /* Header file for the service level approximation of

int    bigs,inv_level,base_stock, next_event_type, num_events,critical;
double L,T, lambda_c, lambda_n, sim_time,
c_num_cus, n_num_cus, c_num_sat, n_num_sat,
c_backorders, n_backorders, time_last_event, time_next_event[6];
```

```
double replenish_queue[1000],evaluation_queue[1000];

FILE  *outfile;

void  initialize(void);
void  timing(void);
void c_demand(void);
void n_demand(void);
void evaluate (void);
void  order_arrival(void);
void  report(void);
double expon(double mean);


/* The main function runs the discrete event simulation by the event-scheduling/t
main ()
{
/* Open the output file. */

outfile = fopen("ration.out", "w");

/* Specify the number of events for the timing function. */

num_events = 5;

/* Ask for input parameters. */

printf(" \n Enter the order-up-to level: ");
scanf("%d",&inv_level);
printf(" \n Enter the treshold level: ");
scanf("%d",&critical);
printf(" \n Enter the arrival rate for the critical class: ");
scanf("%lf", &lambda_c);
```

```
printf(" \n Enter the arrival rate for the non-critical class: ");
scanf("%lf", &lambda_n);
printf(" \n Enter the replenishment lead-time: ");
scanf("%lf", &L);
printf(" \n Enter the demand lead-time for the non-critical class: ");
scanf("%lf", &T);

bigs=inv_level;
base_stock=inv_level;

printf(" \n order-up-to level is %d", inv_level);
printf(" \n threshold level is %d", critical);
printf(" \n critical arrival rate is %f", lambda_c);
printf(" \n non-critical arrival rate is %f", lambda_n);
printf(" \n leadtime is is %f", L);
printf(" \n demand leadtime is is %f", T);

printf(" \n Simulating the system");

/* initialize statistical counters*/

initialize();

/* Run the simulation until it terminates after an end-simulation event
(type 5) occurs. */

do {

/* Determine the next event. */

timing();

/* Invoke the appropriate event function. */
switch (next_event_type) {
                case 1:
                    order_arrival();
                    break;
                case 2:
                    c_demand();
                    break;
                case 3:
                    n_demand();
                    break;
                case 4:
                    evaluate();
                    break;
                case 5:
                    report();
                    break;
       }

        /* If the event just executed was not the end-simulation event (type 3),
           continue simulating.  Otherwise, end the simulation for the current
           (s,S) pair and go on to the next pair (if any). */

} while (next_event_type != 5);


/* End the simulation. */

/*close the output file*/

fclose(outfile);

return 0;
```

```
}

void initialize(void)  /* Initialization function. */
{
/* Initialize the simulation clock. */

sim_time = 0.0;


/* Initialize the statistical counters. */

c_num_cus = 0;
c_num_sat = 0;
n_num_cus = 0;
n_num_sat = 0;
c_backorders = 0;
n_backorders = 0;



/* Initialize the event list.  Since no order is outstanding, the order-
       arrival event and the evaluation event are eliminated from consideration.

time_next_event[1] = 1.0e+30;
time_next_event[2] = sim_time + expon(lambda_c);
time_next_event[3] = sim_time + expon(lambda_n);
time_next_event[4] = 1.0e+30;
time_next_event[5] = 10000000.0;

}

void timing(void)  /* Timing function. */
```

```
{
    int   i;
    double min_time_next_event = 1.0e+29;

    next_event_type = 0;

    /* Determine the event type of the next event to occur. */

    for (i = 1; i <= num_events; ++i)
    {
      if (time_next_event[i] < min_time_next_event)
     {
       min_time_next_event = time_next_event[i];
       next_event_type     = i;
      }
    }

  /*advance the simulation clock to the time of the next event.*/

  sim_time = min_time_next_event;
}

void c_demand(void)  /* Critical Demand event function. */
{
  int i;

  /*increase the number of critical demand arrivals by one unit.*/

  c_num_cus++;

  /* Decrement the inventory or increase critical backorders. */

  if (inv_level>0)
  {
    inv_level--;
    c_num_sat++;
  }

  else {
    c_backorders++;
  }

  /* Schedule the time of the next critical demand and replensihment arrival. */

  time_next_event[2] = sim_time + expon(lambda_c);

  /*schedule the next replenishment arrival by putting the one associated with th
  demand arrival in the appropriate place in the replenishment queue.*/

  for (i=0; i<= 999; i++)
  {
     if(replenish_queue[i] == 0)
     {
        replenish_queue[i] = sim_time + L;
        break;
     }
  }

  time_next_event[1] = replenish_queue[0];
}


void n_demand(void) /*Non-critical demand event function*/
{
  int i;
  int j;
```

```
/*increase number of non-critical demand arrivals by one unit.*/

n_num_cus++;

/*schedule the next non-critical demand arrival.*/

time_next_event[3] = sim_time + expon(lambda_n);


/*schedule the next replenishment arrival by putting the one associated with th
demand arrival in the appropriate place in the replenishment queue.*/

for (i=0; i<= 999; i++)
{
  if(replenish_queue[i] == 0)
  {
    replenish_queue[i] = sim_time + L;
    break;
  }
}

time_next_event[1] = replenish_queue[0];



/*schedule the next evaluation of inventory by putting the one associated with
demand arrival in the appropriate place in the evaluation queue.*/


for (j=0; j<= 999; j++)
{
  if(evaluation_queue[j] == 0)
```

```
    {
      evaluation_queue[j] = sim_time + T;
      break;
    }
  }

  time_next_event[4] = evaluation_queue[0];
}

void evaluate (void) /*Evaluation function for non-critical demand*/
{
  int i;

  /*ration non-critical demand if inventory level is less than or equal to critic

  if (inv_level <= critical )
  {
    n_backorders++;
  }
  else
  {
    inv_level--;
    n_num_sat++;
  }



  /*schedule the next evaluation event*/

  for (i=0; i<= 998; i++)
  {
    evaluation_queue[i] = evaluation_queue[i+1];
  }
  if (evaluation_queue[0] != 0)
  time_next_event[4] = evaluation_queue[0];

  else{
    time_next_event[4] = 1.0e+30;
  }
}

void order_arrival(void)  /* Order arrival event function. */
{
  int i;

  /*use the priority clearing mechanism, i.e., clear non-critical backorders only
  inv level is at or above critical*/

  if (c_backorders > 0)

    c_backorders --;

  else if (inv_level >= critical && n_backorders > 0)

    n_backorders --;

  else

    inv_level++;

  /*schedule the next replenishment arrival*/

  for (i=0; i<= 998; i++)
  {
    replenish_queue[i] = replenish_queue[i+1];
  }

  if (replenish_queue[0] != 0)
  time_next_event[1] = replenish_queue[0];

  else{
    time_next_event[1] = 1.0e+30;
  }
}

void report(void)  /* Report generator function. */
{
  /* Compute and write the estimates of desired measures of performance
  which are the type I service levels of both demand classes.*/

  double beta_c,beta_n;

  beta_c = c_num_sat / c_num_cus;

  beta_n  = n_num_sat/ n_num_cus;

  printf("\n critical satisfied %f", c_num_sat);
  printf("\n critical customers %f", c_num_cus);
  printf("\n non-critical satisfied %f", n_num_sat);
  printf("\n non-critical customers %f", n_num_cus);
  printf("\n critical service level is %f", beta_c);
  printf("\n non-critical service level is %f", beta_n);
  printf("\n approximation for critical service level is %f", approximate());
  fprintf(outfile, "%d %d % lf %lf %lf %lf %lf \n ", bigs,critical, lambda_c, lam
  printf(" \n End of Simulation ... ");
}

double expon(double mean)  /* Exponential variate generation function. */
```

```
{
  /* Return an exponential random variate with mean "mean". */

  return log(lcgrand(1)) / (- mean) ;
}


/***************************End of program*******************************/
```

A.2 Code of the optimization study

```
#include <math.h>
#include <stdlib.h>
#include <stdio.h>
#include "approximation.h"
#include "nc_service.h"

int    base_stock,critical, delta_max,delta_min,S_opt,S_c_opt;
double L,T,lambda_c,lambda_n, beta_c, beta_n,result;


main ()
{
  int num1,num2,num3;

  printf(" \n Enter the arrival rate for the critical class: ");
  scanf("%lf", &lambda_c);
  printf(" \n Enter the arrival rate for the non-critical class: ");
  scanf("%lf", &lambda_n);
  printf(" \n Enter the replenishment lead-time: ");
```

```
scanf("%lf", &L);
printf(" \n Enter the demand lead-time for the non-critical class: ");
scanf("%lf", &T);
printf(" \n Enter the service level requirement for the critical class: ");
scanf("%lf", &beta_c);
printf(" \n Enter the service level requirement for the non-critical class: ");
scanf("%lf", &beta_n);

printf(" \n critical arrival rate is %f", lambda_c);
printf(" \n non-critical arrival rate is %f", lambda_n);
printf(" \n leadtime is is %f", L);
printf(" \n demand leadtime is is %f", T);
printf(" \n beta_c is %f", beta_c);
printf(" \n beta_n is %f", beta_n);

for(num2=1;; num2++)
{
  base_stock=num2;

  if(nc_service() >= beta_n)
  {
    delta_min = base_stock;
    printf("\n The non-critical service level is %f", nc_service() );
    break;
  }
}
printf("\n delta_min is  %d", delta_min);

for(num1=delta_min; ; num1++)
{
  base_stock=num1;
  if(approximate() >= beta_c)
  {
      delta_max = base_stock;
      S_opt = base_stock;
      printf("\n The critical service level is %f", approximate() );
      break;
    }
  }
  printf("\n delta_max is  %d", delta_max);

  for (num3=delta_max-1; num3>= delta_min; num3--)
  {
    base_stock=num3;
    critical=base_stock-delta_min;
    if(approximate() >= beta_c )
    {
      S_opt = base_stock;
      S_c_opt= critical;
      printf("\n The critical service level is %f", approximate() );
      printf("\n The non-critical service level is %f", nc_service() );
    }

  }
  printf("\n The optimal order-up-to level for these parameters is %d", S_opt);
  printf("\n The optimal critical(threshold) level for these parameters is %d", S
}
```