STRUCTURED POLICIES FOR COMPLEX PRODUCTION AND INVENTORY MODELS

By

Daning Sun

B. Sc. (Mathematics) Fudan University
M. Eng. (Decision Science & Computer Science) Shanghai Jiao-Tong University

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

in THE FACULTY OF GRADUATE STUDIES
FACULTY OF COMMERCE AND BUSINESS ADMINISTRATION

We accept this thesis as conforming to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

August 1990

© Daning Sun, 1990
In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Faculty of Commerce and Business Administration

The University of British Columbia

6224 Agricultural Road

Vancouver, Canada

V6T 1W5

Date:  

Aug. 30, 1990
Abstract

For inventory models minimizing the long-run average cost over an infinite horizon, the existence of optimal policies was an open question for a long time. Consider a deterministic, continuous time inventory system satisfies the following conditions: the production network is acyclic, the joint setup cost function is monotone, the holding cost and the backlogging cost rates are nonnegative, the demand rates are constant over time, the production rates are infinite or finite non-increasing, and backlogging may be allowed or not. For this very general extension of the Wilson–Harris EOQ model, we prove the existence of optimal policies. Very few properties of optimal policies have been discovered since the 1950’s. Restricting the above inventory model to infinite production rates, we present some new properties of optimal policies, such as the Latest Ordering Property, and explicit expressions for echelon inventories and order quantities in terms of ordering instants.

An assembly production system with $n$ facilities has a constant external demand occurring at the end facility. Production rates at each facility are finite and non-increasing along any path in the assembly network. Associated with each facility are a set-up cost and positive echelon holding cost rate. The formulation of the lot-sizing problem is developed in terms of integer-ratio lot size policies. This formulation provides a unification of the integer-split policies formulation of Schwarz and Schrage [34] (1975) and the integer-multiple policies formulation of Moily [20] (1986), allowing either assumption to be operative at any point in the system. A relaxed solution to this unified formulation provides a lower bound to the cost of any feasible policy. The derivation of this Lower Bound Theorem is novel and relies on the notion of path holding costs, a generalization of
echelon holding costs. An optimal power-of-two lot size policy is found by an \( O(n^3 \log n) \)
algorithm and its cost is within 2% of the optimum in the worst case.

Mitchell [18] (1987) extended Roundy's 98%-effectiveness results for one-warehouse multi-retailer inventory systems with backlogging. We extend this 98%-effectiveness result for series inventory systems with backlogging. The nearly-integer-ratio policies still work. The continuous relaxation provides a lower bound on the long-run average cost of any feasible policy. The backlogging model is also reduced in \( O(n) \) time to an equivalent model without backlogging. Roundy's results [27] (1983) are then applied for finding a 98%-effective backlogging policy in \( O(n \log n) \) time.

In an EOQ model with \( n \) products, joint setup costs provide incentives for joint replenishment. These joint setup costs may be modelled as a positive, nondecreasing, submodular set function. A grouping heuristic partitions the \( n \) products into groups, and all products in the same group are always jointly replenished. Each group is then considered as a single "aggregate product" being replenished independently of the other groups, and therefore according to the EOQ formula. As a result, possible savings when several groups are simultaneously replenished are simply ignored. Our main result is that the cost of the best such grouping solution cannot be worse than 44.8% above the optimum cost. Known examples show that it can be as bad as 22.4% above the optimum cost. These results contrast with earlier results for power-of-two policies, the best of which never being worse than about 2% above the optimum cost.
Table of Contents

Abstract ii

List of Tables vii

List of Figures viii

Acknowledgement x

1 Introduction and Summary 1

1.1 Optimal Inventory Policies 2

1.2 Generalization of Previous Work 3

1.3 Performance Bounds on the Cost of Grouping Policies 6

1.4 Joint Replenishment with Subadditive Joint Setup Costs: a Bad Example for Power-of-Two policies 7

2 Optimal Inventory Policies 11

2.1 Introduction 11

2.2 Inventory Models 12

2.3 Existence of Optimal Policies 17

2.4 Properties of Optimal Inventory Policies 28

2.5 Conclusions 56

2.6 Appendix to Chapter 2 57

2.6.1 An Inventory Problem That Has no Optimal Policy 57

2.6.2 Some misunderstanding of the properties of optimal policies 58
2.6.3 A proof of lemma 2.3.4 by using a “remove” Algorithm 61
2.6.4 Calculating $H'$: An Example 62
2.6.5 Notation and Definitions 64

3 Lot Sizing Policies for Finite Production Rate Assembly Systems 75
3.1 Introduction 75
3.2 Notation 78
3.3 Properties of Finite Horizon Optimal Policies 81
3.4 Integer-Ratio Lot Size Policies 82
3.5 Lower Bound Theorem 88
3.6 Effectiveness of Integer-Ratio and Power-of-Two Lot Size Policies 98
3.7 Finding Optimal Power-of-Two Lot Size Policies 99
3.8 Conclusions 106
3.9 Appendix to Chapter 3 107
  3.9.1 Effectiveness of Integer-Multiple and Integer-Split Lot Size Policies 107
  3.9.2 Constructing a Production Schedule from Integer-Ratio Lot Size Policies 111

4 Series Systems With Backlogging 116
4.1 Introduction 116
4.2 Best Integer Frequency Policies 118
4.3 Lower Bound Theorem 132
4.4 Reduction to an Equivalent No-Backlog Problem and Scheduling Algorithm 138
4.5 Conclusions 140

5 The Performance Ratio of Grouping Policies for the Joint Replenishment Problem 142
5.1 Introduction ................................................. 142
5.2 Lower Bound on the Average Cost of All Feasible Policies ............. 146
5.3 Grouping Policies and Performance Ratio .................................. 152
5.4 Upper Bound on the Worst Case Performance Ratio of Grouping Policies 158
5.5 Lower Bound on the worst Case Performance Ratio and Bad Examples 166
5.6 Conclusions ................................................... 169
5.7 Appendix to Chapter 5 ............................................. 171
  5.7.1 The Proofs of Lemmas in the Text ................................. 171
  5.7.2 Notation .................................................... 185
  5.7.3 Performance Ratios of Power-of-β Policies .......................... 189

Bibliography ....................................................... 190
List of Tables

2.1 Expressing $\theta_{r_3}, \theta_{r_4}, \theta_{r_5}, \theta_{r_6}$ in terms of $t_3, t_4$ .......... 42
2.2 Values of functions $\theta_{r_j}(t)$ and $\tau_{r_j}(t)$ ........................................ 43
2.3 Definition of routes ................................................. 62
2.4 Holding costs $h_i$ and $h_{fr}$ .................................. 62
2.5 $\lambda_r$ and $\lambda_{ij}$ ........................................... 63
2.6 Echelon inventory $E_r$ and actual route inventory $I_r$ ...................... 63
2.7 Node inventory $I_i$ and actual route inventory $I_r$ ...................... 63
2.8 Holding cost rate $h_i'$ and echelon holding cost rate $h_j$ ............. 63
3.1 Slopes of the path echelon inventory level ............................... 89
3.2 Production starting instants and lot sizes ............................... 113
5.1 Values of $\rho_\beta, W_\beta$ and $r^*$ .................................. 166
5.2 Performance Ratios of Power-of-$\beta$ Policies .......................... 189
List of Figures

1.1 A cyclic heuristic for three products ........................................ 9

2.1 Time intervals $T_m$, $\tau_m$ and $\tau'$ ........................................ 21
2.2 The non-uniqueness of route ordering quantities. .......................... 31
2.3 Modification Step $\Delta(i, t', t'', I_i^{t''} - k)$ and how it may introduce new non-zero-inventory ordering instants at predecessor node $j \in p(i)$ ............ 33
2.4 The Latest Ordering Property and the Zero Inventory Ordering Property do not imply each other .................................................. 36
2.5 Order instants $\theta_r$ and $\tau_r$. .................................................. 39
2.6 The echelon inventory level $E_i^t$ is a function of time $t$ ............... 44
2.7 Decomposing all the routes passing through node $i$ ......................... 49
2.8 Predecessor Modification Step $\Delta_R(j, \theta_i^{t''}, \theta_i^t, x)$ ............. 52
2.9 Example showing that no optimal policy exists .............................. 58
2.10 The order instants at warehouse do not coincide with the order instants at retailers ......................................................... 60
2.11 Algorithm Remove $(I, i, t)$ .................................................. 61
2.12 The production network of six products ....................................... 62
2.13 The correspondence between the two inventory systems ................. 66

3.1 Example of an assembly system ................................................ 79
3.2 Path $(u, v)$. .................................................. 83
3.3 Cumulated Path Echelon Inventory on path $(u, v)$ ........................ 84
3.4 Route $(u, 1)$ is divided into path $(u, i)$ and route $(s(i), 1)$ ............ 87
3.5 Illustrating the proof of Lemma 3.5.1 — facility $u$ produces four lots over the time interval $[0,T)$.

3.6 Illustrating the proof of Lemma 3.5.2 — decomposition of the path echelon holding cost on $(u,v)$.

3.7 Path $(u,v)$ is divided into path $(u,i)$ and path $(s(i),v)$

3.8 Illustrating the proof of Lemma 3.5.4 — assembly system $G(N,A)$ is partitioned into leaf-routes.

3.9 Graph $G(N,A)$

3.10 Graph $G(N',A')$

3.11 Graph $G(N_1,A_1)$, which is embedded in graph $G(N_2,A_2)$

3.12 Graph $G(N_2,A_2) = G(N,A) \times G(N',A')$

3.13 Three facilities in series

3.14 Network corresponding to the constraints of the 3-facility problem

3.15 Scheduling Algorithm

3.16 Procedure $Schedule(t,v)$

3.17 Six facilities in series

3.18 Echelon inventories $E_{(i,j)}^{t}$ for the example in section 3.9.2.

4.1 The Actual Inventory and Echelon Inventory Level for Series System with backlogging

4.2 The Power-of-Two Frequency Policy

5.1 Grouping Heuristic $H_b$
Acknowledgement

I wish to express my deepest indebtedness to my co-supervisor, Professor Maurice Queyranne, for introducing this research area to me, guiding the research and improving every aspect of my dissertation. Without his invaluable input, wisdom, encouragement, and financial support during my Ph.D. studies in the Management Science Division, my dissertation would never have taken its present form.

I am indebted to my co-supervisor, Professor Derek Atkins, for calling the problems to my attention, for directing my research and for reading and checking every version of my dissertation. His insights, encouragement and personal support have been invaluable to me.

I express my thanks to Professor Robin Roundy of Cornell University for agreeing to be the external examiner for this dissertation. I also thank him for suggesting the improved algorithm for finding a best power-of-two policy for a finite production rate assembly system.

Thanks are also due to my committee members Professors Thomas McCormick, Daniel Granot and Tae Oum for their careful reading of my dissertation and their valuable suggestions.

Throughout the period of my Ph.D. program I was supported by the Government of Canada through the Canadian International Development Agency. I would like to express my profound gratitude to them for the opportunity they have given me.

Finally I must thank my wife Yanyun Zeng and my son James Sun for their patience and understanding when I could not be with them most of the time because of this research.
Chapter 1

Introduction and Summary

There is no doubt about the importance of production planning and inventory control for the manufacturing and distribution industries. Billions of dollars in Canada alone have been invested in inventories. A significant proportion of the workload for computers in manufacturing companies is spent for updating and maintaining production and inventory records. Dozens of production and inventory control packages are commercially available ranging from a small system costing a few hundred dollars to large implementations costing close to one million dollars. The costs of all these activities are not just measured in terms of the current assets tied up, software and hardware costs and the costs of personnel, but also in the loss of national and international competitiveness engendered by poor delivery and service. Thus, a small percentage savings in inventory costs means millions of dollars and better service, and enhances chances of survival in the fiercely competitive world of the 1990's.

Since the classic Economic Order Quantity (EOQ) model was investigated by Harris in 1915, thousands of articles have been published in the production and inventory area. Most of the serious works has been done since the 1950's. It could be claimed that the most significant progress has been made in the 1980's.

The EOQ model elegantly solves the one product inventory problem. The EOQ solution is optimal among not only periodic policies but also all feasible policies. To extend this model to finite production rates and allowing backlogging is straightforward.

However, it is a completely different story for multi-product inventory systems: the
optimal policy is so difficult to get (except for few very simple cases) that we may even doubt the existence of an optimal policy. The difficulties result from the fact that the production decisions are dependent on each other: decisions for one product may depend on decisions made for other products. Although many MRP software packages claim to handle complex bills of materials, in fact most treat each component or product as an isolated single facility. Perhaps the most significant step forward in our ability to sensibly model multiechelon inventory systems was made by Roundy in his two papers [28] (1985) and [30] (1986). Most of my thesis is dedicated to making extensions to the theory that has been built up on his initial work. Because these extensions, both in this thesis and by others prior to this thesis, depend crucially on a clear understanding of properties of optimal policies including even the existence of optimal policies, the first part of the thesis is devoted to a careful understanding and development of such properties. These two aspects of the thesis are now discussed in more detail.

1.1 Optimal Inventory Policies

Even though it is difficult to get an optimal policy, it is natural to ask whether an optimal policy exists and what the properties of such an optimal policy might be. The first question was an open question up until now. Inspired by the work of Hassin and Megiddo [12] (1989), we prove the existence of the optimal policies in Chapter 2.

Turning now to properties of an optimal policy, investigation of such properties was started by Wagner and Whitin [38] (1958). They demonstrated that an optimal policy has the Zero Inventory Ordering Property, i.e., each product orders only at the zero inventory level. Besides the Zero Inventory Ordering Property, Schwarz [33] (1973) proved that two other properties of an optimal policy also hold for one-warehouse $n$-retailer inventory systems: (1) Nestedness Property, i.e., the warehouse orders only when at least one
Chapter 1. Introduction and Summary

retailer orders, and (2) Stationary Property, i.e., order quantities at any retailer between successive orders at warehouse are of equal size. Zheng [39] (1987) extended the Zero Inventory Ordering Property to general acyclic inventory systems with submodular setup cost function (see Chapter 2 herein for details), and the Nestedness Property (which he called the Last Minute Ordering Property) to general acyclic inventory systems but with a separable setup cost function. Zheng also demonstrated that the echelon inventories in an inventory system without backlogging may be expressed as a linear function of the ordering instants. In Chapter 2, we expand this list of properties further. We prove that there is an optimal policy which satisfies the Latest Ordering Property, i.e., every product is ordered as late as possible. We also show that if policy satisfies the optimality properties mentioned above, the echelon inventories as well as order quantities can be explicitly expressed as functions of the ordering instants even for an inventory system with backlogging. These quantitative relationships are very useful in deriving the formulation for specific policies. For example, in Chapter 4 we use them to calculate the total holding cost of any feasible policy for an inventory system with backlogging.

1.2 Generalization of Previous Work

Because optimal policies are difficult to find and have properties that are so general as to make implementation of them an impossibly complex task, it is not surprising that most research has been restricted to some special classes of policies. For a long time, nested policies (i.e., when the warehouse orders, each retailer also has to order) have been implicitly assumed to be near optimal for 1-warehouse n-retailer systems. Unfortunately, it turned out that these policies could be arbitrarily bad, (see Roundy [27] (1983)). Other authors concentrated on designing ingenious and sophisticated heuristics. They then compared their heuristics with others demonstrating improvements by either examples
or simulation. The optimal solutions were just not available for comparison. As a result, nothing was known of the worst case performance of such heuristics.

In his two papers [28] (1985) and [30] (1986), Roundy cleverly avoided this comparison difficulty by comparing with a lower bound to the (unknown) optimum. The lower bound is obtained by solving a continuous relaxation associated with a class of heuristics known as power-of-two policies. He proved that the long-run average cost given by a power-of-two heuristic is within 2% of this lower bound in the worst case. This means that the lower bound is really very close to the optimal value. If the production network is a one-warehouse n-retailer system, an $O(n \log n)$ algorithm gives the optimal power-of-two solution. In short, from the theoretical and practical point of view, this problem is essentially solved as data accuracy is unlikely to justify closing that 2% gap. Since then, a variety of extensions to his basic work have been studied.

For ease of discussion, we classify inventory systems in terms of infinite production rate or finite production rate, and without backlogging or with backlogging.

Most extensions to the basic work were for infinite production rate inventory systems without backlogging. For example, Queyranne [24] (1985) first extended the result to a submodular setup cost function. Zheng [39] (1987) extended the results to a general acyclic production network with a submodular setup cost function. These results probably reach the “boundary” of possible extensions for power-of-two heuristics for infinite production rate inventory systems. As evidence for this view, we have constructed an example in Section 1.4 of this chapter showing that if the setup cost function is not submodular but rather subadditive, the continuous relaxation of power-of-two heuristics is no longer a lower bound on the average cost of inventory systems. For inventory systems in this category, Roundy [23] (1988) summarized several important reasons for using reorder intervals, that is, the time between placement of orders, rather than lot sizes to formulate these inventory models.
Chapter 1. Introduction and Summary

For the finite production rate inventory systems without backlogging, there were no such results prior to this thesis. Several papers proposed heuristics for inventory models with finite production rates, but none of them attempted to find provable performance bounds. For example, for finite production rate assembly systems, Crowston, Wagner, and Henshaw [6] (1972), and Schwarz and Schrage [34] (1975) studied integer-multiple lot size policies, wherein the lot size at each facility is an integer-multiple of its successor’s lot size. Moily [20] (1986) investigated integer-split lot size policies, wherein the lot size at each facility is an integer-split (divisor) of its successor’s lot size. Neither integer-multiple lot size policies nor integer-split lot size policies dominate each other. As shown in Chapter 3, both integer-multiple and integer-split lot size policies may be simultaneously arbitrarily bad. In Chapter 3, we propose an integer-ratio lot size policy and show that the average cost given by the best power-of-two lot size policy is within 2% of the optimal value of the assembly inventory system. As summarized in Chapter 3, we find that using lot sizes rather than reorder intervals to formulate the inventory model is more effective in the finite production rate case.

For the case of infinite production rate inventory model with backlogging, except for one paper by Mitchell [18] (1987) for the one-warehouse $n$-retailer backlogged problem, no generalizations to include backlogging appear to have been made. In Chapter 4, we show that the worst case performance results can be extended to series systems. For inventory systems in this category, it is not near-optimal to use constant reorder intervals. Following Mitchell [18] (1987), we formulate the problem in terms of the reciprocal of (long-run average) order frequencies, i.e., allowing reorder intervals to vary.

Thus the original work of Roundy used power-of-two order intervals, the “finite production rate” results in this thesis use power-of-two lot sizes, and the “backlogging” results in this thesis use power-of-two order frequencies.
1.3 Performance Bounds on the Cost of Grouping Policies

Until now, all known performance bounds on inventory systems were for the integer ratio or power-of-two type replenishment policies. In fact, other policies, such as *grouping policies*, are also attractive and interesting in real world applications. A grouping policy partitions the products into groups, and all products in the same group are always jointly replenished. Each group is then considered as a single "aggregate product" being replenished independently of the other groups, and therefore the EOQ formula is applicable. As a result, possible savings when several groups are simultaneously replenished are simply ignored. The grouping policies are easy to implement and are potentially more appropriate when inventory is a subproblem, such as inventory problems combined with vehicle routing problems (Anily [2] (1986)). Grouping policies sometimes outperform power-of-two policies. For example, if two products are independent of each other and have the reorder intervals of 1 and \( \sqrt{2} \), respectively, then the (trivial) grouping policy gives the optimal solution, but the power-of-two policy will never be optimal. A two-products example in Chapter 5 shows that a best grouping policy can be much worse than a power-of-two policy. Therefore, a natural question to ask is whether best grouping policies have provable performance bounds or could be arbitrarily bad. In Chapter 5, we prove that grouping policies do have a provable performance bound, that is, the cost of the best such grouping solution cannot be worse than 44.8% above the optimum cost.

As the subsequent chapters in this thesis are relatively independent, a detailed literature review is found at the beginning of each chapter.
1.4 Joint Replenishment with Subadditive Joint Setup Costs: a Bad Example for Power–of–Two policies

For the joint replenishment inventory problem with a submodular setup cost function (see the definition in Chapter 2), the continuous relaxation of the problem of finding optimal power–of–two policies yields a lower bound for all feasible policies. However, the following example shows that when the setup cost function is subadditive (see the definition in Chapter 2) but not submodular, the continuous relaxation of this problem no longer yields a lower bound for all feasible policies.

**Example.** The joint replenishment inventory model has three products. Setup costs \( K_S \), holding cost rates \( h_i \) and demand rates \( d_i \) are given below:

\[
K_1 = K_2 = K_3 = K_{12} = K_{13} = K_{23} = 1, \quad K_{123} = 2,
\]

\[
h_1 = h_2 = h_3 = 1,
\]

\[
d_1 = d_2 = d_3 = 2,
\]

where setup cost \( K_S \) means all products in set \( S \) are replenished at the same time. Note that \( K_S \) is subadditive but not submodular.

As the setup cost for replenishing two products is the same as for replenishing one product, we will always replenish at least two products at the same time. By symmetry, there are only two different power–of–two policies, where \( t_i \) is the replenishment cycle of product \( i \):

1. Products 1 and 2 are replenished at the same time, and product 3 is replenished less frequently. The following nonlinear integer programming solves this problem.

\[
Z_{POT}^1 = \min \frac{K_{123} - K_{12}}{t_3} + \frac{K_{12}}{t_1} + (2t_1 + t_3) = \frac{1}{t_3} + \frac{1}{t_1} + (2t_1 + t_3)
\]

s.t. \( t_3 \geq t_1 \),

\[ t_1 = 2^{m_1} t_0, \quad t_3 = 2^{m_3} t_0, \]
where \( t_0 \) is the base period. The solution to the continuous relaxation of the problem above is \( t_1 = t_2 = \frac{1}{\sqrt{2}}, \ t_3 = 1 \), and

\[
\tilde{Z}_{POT}^1 = 2 + 2\sqrt{2} = 4.82843
\]

(2) Products 1, 2 and 3 are replenished at the same time.

\[
Z_{POT}^2 = \min \frac{K_{123}}{t_3} + 3t_3 = \frac{2}{t_3} + 3t_3
\]

s.t. \( t_3 \geq 0 \),

\[ t_3 = 2^{m_3} t_0. \]

where \( t_0 \) is the base period. The solution to the continuous relaxation of the problem above is \( t_1 = t_2 = t_3 = \sqrt{\frac{2}{3}} \), and

\[
\tilde{Z}_{POT}^2 = 2\sqrt{6} = 4.89898.
\]

Therefore, the power-of-two lower bound is

\[
\hat{Z}_{POT} = \min \{ \tilde{Z}_{POT}^1, \tilde{Z}_{POT}^2 \} = 2 + 2\sqrt{2} = 4.82843.
\]

However, the following cyclic heuristic defines a feasible policy with an average cost less than the Power-of-Two lower bound \( \hat{Z}_{POT} \). Let \( t \) be the cycle period of the heuristic. At equal time intervals \( t/3 \), we replenish two products at the same time in the cyclic sequence \( \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2\}, \ldots \) as shown in Figure 1.1.

The total setup costs in any time interval \( t \) is

\[ K_{12} + K_{13} + K_{23} = 3, \]

and the holding costs for product \( i \) (\( i = 1, 2 \) and 3) in the same time period is

\[ h_i \left[ \frac{d_i}{2} \left( \frac{t}{3} \right)^2 + \frac{d_i}{2} \left( \frac{2t}{3} \right)^2 \right] = \frac{5}{9} t^2. \]
Figure 1.1: A cyclic heuristic for three products

Therefore, the average cost for cyclic heuristic is

\[ C(t) = \frac{3}{t} + \frac{5}{3}t. \]

For \( t^* = \frac{3}{\sqrt{5}} = 1.34164 \), the average cost is

\[ C(t^*) = 2\sqrt{5} = 4.47214. \]

Note that

\[ \frac{C(t^*)}{Z_{POT}} = \frac{2\sqrt{5}}{2(1 + \sqrt{2})} = 0.92621, \]

i.e., the average cost for this cyclic heuristic is less than 93% of the Power-of-Two lower bound.
In fact, this cyclic heuristic can easily be extended to the following $n$–product problem:

$$K_S = \left\lfloor \frac{|S|}{n-1} \right\rfloor = \begin{cases} 0, & |S| = 0, \\ 1, & |S| = 1, 2, \ldots, n-1, \\ 2, & |S| = n, \end{cases}$$

$$h_i = 1, \ d_i = 2, \ \forall i \in N,$$

where $N = \{1, 2, \ldots, n\}$. It is not difficult to find that the Power–of–Two lower bound is $Z_{POT} = 2(1 + \sqrt{n - 1})$. The average cost given by the cyclic heuristic is $C(t^*) = 2\sqrt{n + 2}$ with $t^* = \sqrt{n + 2}/n$. For $n = 10$, $C(t^*)/Z_{POT} = \sqrt{3}/2 = 0.8660$, i.e., the average cost for this cyclic heuristic is less than 87% of the Power–of–Two lower bound.
Chapter 2

Optimal Inventory Policies

2.1 Introduction

Lots of the work in production and inventory area has dealt with the case of minimization of the long-run average cost for infinite horizon problems. In this case, the existence of optimal policies was an open question. In Section 2.6.1 of this chapter, we show that an inventory problem could have no optimal policy. This means that the existence of optimal policies is a meaningful problem. The first objective of this chapter is to prove the existence of optimal policies for very general inventory systems. Our proof is inspired by the work of Hassin and Megiddo [12] (1989). We correct a deficiency in their proof, and extend their result to a more considerably general setting. An immediate consequence of this theorem is that the qualifying phrase "if there exists an optimal policy", found (or which ought to be found) in many theorems about inventory systems with long-run average cost criterion, may now be dropped.

Various properties of optimal inventory policies have been investigated since the late 1950's. The classic Wagner-Whitin [38] (1958) paper was one of the first works discussing such properties. But very few new properties have been discovered since the early 1960's. Schwarz [33] (1973) and Zheng [39] (1987) summarized the known properties. These properties seem simple and easy to accept, but they may also be very easily misunderstood. See concrete examples in section 2.6.2. To alleviate the potential for confusion, we associate these properties with different inventory models. All known
properties of optimal policies will be expressed in term of this classification. In addition, some properties not discussed before, such as those related to route inventories, will also be presented. These properties reveal some deeper insights into the structure of inventory systems.

The structure of this chapter is as follows. In Section 2.2, we provide a taxonomy of deterministic inventory models. In Section 2.3, we prove the existence of optimal policies. In Section 2.4, properties of optimal policies for various inventory systems are listed.

2.2 Inventory Models

In this thesis, we deal with different inventory models. We use five parameters — network, setup cost function, holding and backlogging cost, production rate, backlogging — to describe these inventory models. This makes it easy to rigorously specify the models and their properties. In the following, we first describe these five parameters, then introduce inventory models.

The production network $G(N, A)$ is an acyclic directed graph, where $N = \{1, 2, \cdots, n\}$ is the node set and $A$ is the arc set. The nodes are indexed such that $i > j$ if $(i, j) \in A$. Each node represents a particular product, at a particular location and/or production stage. A directed arc $(i, j) \in A$ from node $i$ to node $j$ represents the fact that the output of location or operation $i$ directly supplies location or operation $j$. We say that $i$ is an immediate predecessor of $j$, and that $j$ is an immediate successor of $i$, whenever are $(i, j) \in A$. Let $p(i)$ be the set of all immediate predecessors of product $i$, and $s(i)$ be the set of all immediate successors of node $i$. Let $D_N \triangleq \{i \in N | s(i) = \emptyset\}$ be the set of all nodes without successor. Without loss of generality, we may assume that $D_N = \{i \in N | \text{node } i \text{ has external demand}\}$ — the external demand set of $G(N, A)$. See the details why we may make such an
assumption in section 2.6.5.

Let $\lambda_{ij}$ be the number of units of product $i$ required to produce 1 unit of product $j$.

We use the following symbols to denote different types of networks:

$G$ – General acyclic network: there is no directed cycle in network $G(N, A)$.

$A$ – Assembly Systems (a special case of $G$): each node has exactly one successor in the network $G(N, A)$, except for one node, which has no successor and is called the root node.

$D$ – Distribution Systems (a special case of $G$): each node has exactly one predecessor in the network $G(N, A)$, except for one node, which has no predecessor.

$S$ – Series Systems (the intersection of $A$ and $D$): each node has exactly one successor and one predecessor in the network $G(N, A)$, except for two nodes, one with only a successor and the other one with only a predecessor.

$O$ – One-warehouse Multi-retailer Systems (a special case of $D$): one node, termed the warehouse, has successors but no predecessor in network $G(N, A)$, and all other nodes, the retailers, have no successor and exactly one predecessor, the warehouse.

$I$ – Isolated nodes (a special case of $G$): $A = \emptyset$.

The setup (or joint replenishment) cost function is a real-valued set function $K^t: 2^N \mapsto R^+$ such that for all $S \subseteq N$, $K(S)$ is the cost associated with replenishing all the products $i \in S$ at the same instant $t$. Without loss of generality, it is assumed that $K^t(\emptyset) = 0$, and $K^t(S) \geq 0$, $\forall S \neq \emptyset, S \subseteq N$. In this thesis, we will assume, for simplicity, that setup costs are stationary, that is $K^t(S) = K(S)$ for all $t, S$. Let $k_i$ be a real number associated with each node $i \in N$. The following symbols are used to denote different properties of the setup cost function:
MT - Monotonicity: \( K(S) \leq K(T), \ \forall S \subseteq T \subseteq N \).

SA - Monotonicity and Subadditivity: \( K(S \cup T) \leq K(S) + K(T), \ \forall S, T \subseteq N \).

SM - Monotonicity and Submodularity: \( K(S \cup T) + K(S \cap T) \leq K(S) + K(T), \ \forall S, T \subseteq N \).

MD - Modularity or First Order Interaction (a special case of SM): \( K(\emptyset) = 0, \ K(S) = K_0 + \sum_{i \in S} k_i, \ \forall S \neq \emptyset, S \subseteq N \), where \( K_0 \geq 0 \) is a fixed constant.

SP - Separability (a special case of MD): \( K(S) = \sum_{i \in S} k_i, \ \forall S \subseteq N \).

Backlogging: The following symbols are used to denote different assumptions about backlogging.

GB - General Backlogging: Backlogging may be allowed for some or all external demands, but is not allowed for any internal (induced) demand.

B - Always backlogging (a special case of GB): Backlogging is allowed for every external demand.

NB - No backlogging (a special case of GB): There is no backlogging at any place.

**Holding and Backlogging Costs** are inventory related cost, charged in terms of the units of products stored and backlogged and the length of time it is stored and backlogged, respectively. Let \( h_i^t \) (resp. \( b_i^t \)) be the **holding** (resp. **backlogging** cost rate) of product \( i \) at instant \( t \): it is the cost of holding (resp. backlogging) one unit of product \( i \) for one unit of time at instant \( t \). Let \( h_i^t = h_i^t - \sum_{j \in P(i)} \lambda_{ij} h_j^t \) be the **echelon holding cost rate** of node \( i \) at instant \( t \): it is the "added" holding cost rate for product \( i \). Let \( I_i^t \) be the **actual (node) inventory** level of product \( i \) at instant \( t \), \( I_i^{-t} = \lim_{t' \nearrow t} I_i^{t'} \) be the inventory level just before instant \( t \), \( I_i^{t+} = \lim_{t' \searrow t} I_i^{t'} \) be the inventory level just after instant \( t \). Let \( I_i^t \) be the **physical inventory**.
level of product $i$ at instant $t$ and $I_i^t \triangleq \max(-I_i^t, 0)$, be the backlogged inventory level of product $i$ at instant $t$. Let the total holding and backlogging cost rate at instant $t$ be $H^t \triangleq \sum_{i=1}^{n}(T_i h_i^t + I_i^t b_i^t)$ which is the total holding and backlogging cost in one unit of time at instant $t$. The total holding and backlogging cost: $C_H(T) \triangleq \int_0^T H^t \, dt$ is the total holding and backlogging cost accrued over time interval $[0, T)$.

The following symbols are used to denote different types of holding and backlogging cost functions:

**NN** – Nonnegative Holding and backlogging cost rates: $h_i^t \geq 0$ and $b_i^t \geq 0$ for each $i \in N$. Thus holding and backlogging costs are increasing functions of the number of units stored and backlogged in the holding and backlogging duration.

**CT** – Constant holding and backlogging cost rates (a special case of $NN$): $h_i^t = h_i \geq 0$ and $b_i^t = b_i \geq 0$, which are independent of time $t$ for each product $i$.

**NE** – Nonnegative Echelon holding cost rates (a special case of $CT$): $h_i \geq 0$ and constant.

**Production Rate:** Let production rate $\pi_i$ be the number of units that can be produced at node $i$ per time unit. The following symbols are used to denote different types of production rates:

**F** – Finite production rate: the production rate $\pi_i$ is finite for each node in $N$. It is assumed that production at node $i$ can only be turned on, at rate $\pi_i$, or off, at rate 0. Every turn on in this production status incurs a setup cost, as described above.
Chapter 2. Optimal Inventory Policies

**FF** – Fast finite production rate (a special case of F): the production rate $\pi_i$ of each node $i \in N$ is fast enough that its output could supply the total demands from all its immediate successors without the need for building inventory beforehand: $\pi_i \geq \sum_{j \in s(i)} \lambda_{ij} \pi_j$, for all $i \in N$. Recall that $s(i)$ is the set of all immediate successors of node $i$.

**IF** – Infinite production rate: the production rate $\pi_i$ of each node is infinite. This corresponds to “instantaneous” or negligible production times, and to deliveries in distribution networks.

Let “⊂” mean “is a special case of”. Because “a property holds for a general case” implies that “it also holds for the special case”, the following relationships are worth noticing. For the network types, $S \subseteq A \subseteq G$, $S \subseteq D \subseteq G$ and $O \subseteq D$, $I \subseteq G$; for the setup cost functions: $SP \subseteq MD \subseteq SM \subseteq SA \subseteq MT$; for backlogging: $B \subseteq GB$ and $NB \subseteq GB$; for the holding and backlogging cost functions: $NE \subseteq CT \subseteq NN$; for the production rates: $FF \subseteq F$.

Now we can use these parameters to define our inventory model by $IM(network, setup\ cost\ function, holding\ and\ backlogging\ cost, production\ rate, backlogging)$. When we don’t care about the assumptions on some of the parameters, sometimes we will use “•” to represent the situation. Note that because we always assume that the external demand rates are constant, demand is not a parameter in our models. We also assume that the transfer of a lot is instantaneous, and that supplies arrive at nodes without predecessor instantaneously and in batch. Our objective is to minimize the long-run average cost over an infinite horizon (see definition in next section).

As an example, inventory model $IM(G,MT,NE,FF,GB)$ means:

1. The production network $G(N,A)$ is a general acyclic network.
2. The setup cost function satisfies the monotonicity property: \( K(S) \leq K(T), \forall S \subseteq T \subseteq N \).

3. The echelon holding cost rates are constant and non-negative. The backlogging cost rates are also constant.

4. The production rate at any node is finite and fast enough such that its output could supply the total demands from all its successors without building inventory beforehand. This requirement means that \( \pi_i \geq \sum_{j \in s(i)} \pi_j \), for each node \( i \in N \).

5. The external demand has to be satisfied and backlogging some of the external demands is allowed.

As another example inventory model \( IM(G,MT,NE,FF,NB) \) is the same as inventory model \( IM(G,MT,NE,FF,GB) \) except that backlogging is not allowed anywhere in the network.

### 2.3 Existence of Optimal Policies

Even though the results in this chapter are not difficult to be extended to more general inventory models, we derive them only for simple and important inventory models \( IM(G,MT,NN,IF,GB) \) and \( IM(G,MT,NN,FF,NB) \).

In this section, we prove that an optimal policy exists for these two inventory systems. For this, we need a few concepts about policies and costs. We use two assumptions: the Zero Initial Inventory Assumption and the Nonpositive Ending Inventory Assumption. They are important because they guarantee that an infinite horizon policy constructed from a series of finite horizon policies is feasible, i.e., satisfies all constraints. This construction will be used in the proof of the existence of optimal policies. Based on these prerequisites, we proceed to proving the existence of optimal policies.
Chapter 2. Optimal Inventory Policies

A replenishment policy $P \triangleq (t_P, Q_P)$ is a specification of ordering instants $t_P$ and order quantities $Q_P$ for all products over a planning horizon. That is,

\[
\begin{align*}
  t_P &= (t_{1P}, t_{2P}, \cdots, t_{nP}), \\
  Q_P &= (Q_{1P}, Q_{2P}, \cdots, Q_{nP}), \\
  t_{iP} &= (t_{1iP}, t_{2iP}, \cdots, t_{kiP}, \cdots), \\
  0 &\leq t_{1iP} < t_{2iP} < \cdots < t_{kiP} < \cdots \\n  Q_{iP} &= (Q_{1iP}, Q_{2iP}, \cdots, Q_{kiP}, \cdots) \geq 0,
\end{align*}
\]

where $n$ is the number of products, quantity $Q^k_{iP}$ is the amount of product $i$ ordered at instant $t^k_{iP}$. Sometimes we use $Q^t_{iP}$ to represent the amount ordered for product $i$ at instant $t$, and if $t = t^k_{iP}$ the quantity $Q^t_{iP}$ may be replaced by $Q^k_{iP}$, whenever it is convenient. Note that for the finite product rate case the ordering instant $t$ is the production starting time. When replenishment policy $P$ is not emphasized, subscript $P$ in $t$ and $Q$ may be dropped without risk of confusion.

An infinite/finite horizon policy: If the planning horizon is infinite (resp. finite), the policy $P$ is called an infinite (resp. a finite) horizon policy. If $P$ is an infinite horizon policy, the restriction of $P$ to a finite horizon $[0, T)$ is a finite horizon policy, which is also called policy $P$ for simplicity.

A feasible policy $P$ is a policy which satisfies all the requirements of the model under consideration. These requirements include satisfying external demands for specified products, without backlogging if it is prohibited.

We consider different costs incurred in an inventory system by this replenishment policy $P$:

The total cost $C_P(T)$ of policy $P$ over finite horizon $[0, T)$ is the sum of total setup or ordering cost $C_{SP}(T)$ and total holding and backlogging cost $C_{HP}(T)$ over horizon
Chapter 2. Optimal Inventory Policies

[0, T), that is, \( C_P(T) = C_{SP}(T) + C_{HP}(T) \). Note that we exclude the ordering cost at instant \( T \). This means that backlogging may exist at instant \( T \). Thus if we concatenate two policies together, the ordering cost of the first policy at ending instant will be taken into consideration by the second policy at the initial instant. Over a finite horizon, the objective is to minimize the total cost \( C_P(T) \) over all feasible policies \( P \). Assuming all setup costs are positive (except for the empty set) and that a feasible policy with finite cost exists, we are only interested in policies with a finite number of replenishments over finite horizon \([0, T)\).

The long-run average cost \( \overline{C}_P \) of an infinite horizon policy \( P \) is defined by

\[
\overline{C}_P \triangleq \limsup_{T \to \infty} \frac{1}{T} C_P(T).
\]

(2.1)

The optimal average cost \( \overline{C}^* \) over all feasible policies is defined by

\[
\overline{C}^* \triangleq \inf_{\text{all feasible } P} \overline{C}_P.
\]

(2.2)

Sometimes, we are also interested in the relationship between two policies. Domination is one of the simplest relationship between two policies.

**Dominate:** Policy \( P' \) dominates policy \( P \) over interval \([0, T)\), if \( C_{P'}(t) \leq C_P(t), \ \forall t \in [0, T) \).

To construct a feasible infinite horizon policy from a series of finite horizon policies, we need the Zero Initial Inventory Assumption and Nonpositive Ending Inventory Assumption defined below.

**Zero Initial Inventory Assumption:** The initial inventory of each product is zero and all the products are replenished at the initial instant \( 0 \).
Nonpositive Ending Inventory Policy: A finite horizon feasible policy \( P \) over \([0, T)\) is a Nonpositive Ending Inventory policy if the inventory level of each product is nonpositive at instant \( T \). When backlogging is not allowed, a Nonpositive Ending Inventory policy is a Zero Ending Inventory policy.

Nonpositive Ending Inventory Assumption: For any feasible policy \( P \) over finite horizon \([0, T)\) there exists a Nonpositive Ending Inventory policy \( P' \), which dominates policy \( P \). When backlogging is not allowed, the Nonpositive Ending Inventory Assumption reduces to the Zero Ending Inventory Assumption.

As the total cost over any finite horizon has no effect on the long-run average cost, without loss of generality we assume that the Zero Initial Inventory Assumption holds for inventory models \( \text{IM}(G,\text{MT},\text{NN},\text{IF},\text{GB}) \) and \( \text{IM}(G,\text{MT},\text{NN},\text{FF},\text{NB}) \). In Lemma 2.3.4, we can show that the Nonpositive Ending Inventory Assumption also holds for both inventory models.

Given a sequence of finite Nonpositive Ending Inventory policies \( P^m \) over time interval \([0, T_m)\), satisfying the Zero Initial Inventory Assumption, the concatenation method \( CM \) constructs an infinite horizon policy \( P \triangleq (P^1P^2\cdots) \) as follows. For all \( m \), let policy \( P^m = (t^m, Q^m) \) be defined by

\[
\begin{align*}
\left. \begin{array}{l}
t^m_i = (t^1_i, t^2_i, \ldots, t^m_i),
0 = t^1_i < t^2_i < \cdots < t^m_i < T_m
Q^m_i = (Q^1_i, Q^2_i, \ldots, Q^m_i) \geq 0,
\end{array} \right\} & i = 1, 2, \ldots, n
\end{align*}
\]

Let

\[
\begin{align*}
T_0 & \triangleq 0,
\tau_m & \triangleq \sum_{i=0}^{m} T_i, \quad m = 0, 1, 2, \ldots
\end{align*}
\]
Figure 2.1: Time intervals $T_m$, $\tau_m$ and $\tau'$(

\[
\begin{array}{ccccccc}
T_1 & T_2 & \ldots & T_m & T_{m+1} \\
0 & \tau_1 & \ldots & \tau_{m-1} & \tau_m & \tau' & \tau_{m+1} \\
\end{array}
\]

time

For index $k = \ell + \sum_{u=1}^{m-1} j_u$ with $1 \leq \ell \leq j_{im}$, let policy $P = (t_P, Q_P)$ be defined by

\[
ti_P^k = t_m-1 + t_{ipm},
\]

and

\[
Qi_P^k = \begin{cases} 
Q_{ipm}' - T_{ipm-1}^{\ell}, & \text{if } \ell = 1, \\ 
Q_{ipm}', & \text{if } \ell = 2, \ldots, j_{im}
\end{cases}
\]

where $T_{ipm-1}^{\ell}$ is the ending inventory of product $i$ under policy $P_m$. Recall that $T_{ipm-1}^{\ell} \leq 0$ by the Nonpositive Ending Inventory Assumption.

That is, policy $P = (P_1P_2 \cdots)$ over interval $(\tau_{m-1}, \tau_m)$ is identical to replenishment policy $P_m$ over $(0, T_m)$, shifted forward $\tau_{m-1}$ time units.

Lemma 2.3.1 (Property of Infinite Horizon Policy $P$)

Consider inventory models IM($G$,MT,NN,IF,GB) and IM($G$,MT,NN,FF,NB). Suppose $P_m$ satisfies the Zero Initial Inventory and the Nonpositive Ending Inventory Assumptions for each $m$. Suppose $\tau \in [\tau_m, \tau_{m+1})$ and $\tau' = \tau - \tau_m$ (see the Figure 2.1).

Then policy $P = (P_1P_2 \cdots)$ constructed by concatenation method CM is feasible, and its total cost over interval $[0, \tau)$ is

\[
C_P(\tau) = C_P(\tau_m) + C_{Pm+1}(\tau') = \sum_{i=1}^{m} C_{P_i}(T_i) + C_{Pm+1}(\tau').
\]
Proof. For the inventory model IM(G,MT,NN,FF,NB), the Zero Initial Inventory Assumption and the Nonpositive Ending Inventory Assumption ensure that the initial inventory and ending inventory of each product under each finite horizon policy $P^m$ is zero. Therefore, policy $P$ is feasible. For the inventory model IM(G,MT,NN,IF,GB), the initial inventory of each product is zero, each product is ordered at the initial instant, and the ending inventory of each product is nonpositive. To make policy $P$ feasible, we only need to include the backlogged quantities into the order quantities of policy $P$ at all instants $\tau_m$, as defined above. Thus feasibility obtains. The total cost equations follow directly by the construction.

Theorem 2.3.2 (Existence of an Optimal Policy)

If the Zero Initial Inventory and the Nonpositive Ending Inventory Assumptions hold for the inventory models IM(G,MT,NN,IF,GB) and IM(G,MT,NN,FF,NB), then there exists an optimal infinite horizon policy $P^*$, i.e., satisfying

$$\overline{C}_{P^*} = \overline{C}^*. \quad (2.4)$$

Proof.

By definition of $\overline{C}^*$, there exists a sequence of infinite horizon policies $\{P^1, P^2, \ldots, P^m, \ldots\}$ such that

$$\overline{C}_{P^m} \leq \overline{C}^* + \frac{1}{2^{m+1}}, \quad \forall \ m = 1, 2, \ldots \quad (2.5)$$

By definition of $\overline{C}_{P^m}$, there exists $T'_m > 0$ such that

$$\frac{1}{T} C_{P^m}(T) \leq \overline{C}_{P^m} + \frac{1}{2^{m+1}}, \quad \forall \ T \geq T'_m, \ m = 1, 2, \ldots \quad (2.6)$$
Let
\[ C_m \triangleq C_{P^m}(T'_m). \] (2.7)

Recursively define
\[ T_m \triangleq \max \left\{ T'_m, \ 3 \sum_{i=1}^{m-1} T_i, \ 2^{m+1} C_{m+1} \right\}. \] (2.8)

(Note that the third argument in the maximization is expressed in time units, because in inequality (2.6), the term \((1/2^{m+1})\) has the units of a cost rate: dollars per time unit.)

Using the concatenation method CM given above, we may define a feasible policy \( P^* = (P^1 P^2 \cdots) \). By Lemma 2.3.1, the total cost of policy \( P^* \) over interval \([0, \tau]\) is
\[ C_{P^*}(\tau) = \sum_{i=1}^{m} C_{P^i}(T_i) + C_{P^{m+1}}(\tau'). \]

In the following, we show that the infinite horizon policy \( P^* \) defined above is an optimal policy.

First, let's estimate the total cost of policy \( P^* \) over \([0, \tau_m]\):
\[
C_{P^*}(\tau_m) = \sum_{i=1}^{m} C_{P^i}(T_i)
\leq \sum_{i=1}^{m} T_i \left( \bar{C}_{P^i} + \frac{1}{2^{i+1}} \right) \quad \text{ (by inequality (2.6))}
\leq \sum_{i=1}^{m} T_i \left( \bar{C}^* + \frac{1}{2^i} \right) \quad \text{ (by inequality (2.5))}
= \tau_m \bar{C}^* + \sum_{i=1}^{m} \frac{T_i}{2^i} \quad \text{ (by definition of } \tau_m \text{)} \] (2.9)

The following inequality is proved by induction:
\[
\sum_{i=1}^{m} \frac{T_i}{2^i} \leq \frac{1}{2^{m-1}}. \] (2.10)

For \( m = 1 \), it follows from the definition \( \tau_1 = T_1 \).
Suppose that the inequality (2.10) holds for \( m \). For \( m + 1 \), we have
\[
\sum_{i=1}^{m+1} \frac{T_i}{2^i} \tau_{m+1} = \sum_{i=1}^{m} \frac{T_i}{2^i} + \frac{T_{m+1}}{2^m} + \sum_{i=1}^{m+1} \frac{m+1}{\sum_{i=1}^{m+1} T_i} \geq \frac{1}{4 \sum_{i=1}^{m} T_i} + \frac{1}{2^{m+1}} \quad (\text{by definition of } T_{m+1} \text{ in (2.8)})
\]
\[
\leq \frac{1}{4 \sum_{i=1}^{m} T_i} + \frac{1}{2^{m+1}} \quad (\text{by induction})
\]
\[
= \frac{1}{2^m}.
\]

This completes the proof of inequality (2.10).

From inequalities (2.9) and (2.10),
\[
\frac{1}{\tau_m} C_{P^*}(\tau_m) \leq \bar{C}^* + \frac{1}{2^{m-1}}, \quad \text{for } m = 1, 2, \ldots.
\]
(2.11)

Then
\[
\frac{1}{\tau} C_{P^*}(\tau) = \frac{1}{\tau} C_{P^*}(\tau_m) + \frac{1}{\tau} C_{P_{m+1}}(\tau') \quad (\text{by Lemma 2.3.1})
\]
\[
\leq \frac{\tau_m}{\tau} \left( \bar{C}^* + \frac{1}{2^{m-1}} \right) + \frac{1}{\tau} C_{P_{m+1}}(\tau'). \quad (\text{by inequality (2.11)})
\]
(2.12)

There are two cases to consider:

Case 1. If \( \tau' \leq T'_{m+1} \), then
\[
C_{P_{m+1}}(\tau') \leq C_{P_{m+1}}(T'_{m+1})
\]
\[
= C_{m+1} \quad (\text{by definition of } C_{m+1} \text{ in (2.7)})
\]
\[
\leq \frac{T_m}{2^{m+1}}. \quad (\text{by definition of } T_m \text{ in (2.8)})
\]

By inequality (2.12),
\[
\frac{1}{\tau} C_{P^*}(\tau) \leq \frac{\tau_m}{\tau} \left( \bar{C}^* + \frac{1}{2^{m-1}} \right) + \frac{1}{\tau} \frac{T_m}{2^{m+1}}
\]
Chapter 2. Optimal Inventory Policies

\[ C^* \leq \frac{1}{2^{m-1}} + \frac{1}{2^{m+1}} \] (from \( \tau_m \leq \tau \) and \( T_m \leq \tau \))

\[ C^* \leq \frac{1}{2^{m-2}} \]

Case 2. If \( \tau' \geq T_{m+1}' \), then

\[ C_{P_{m+1}}(\tau') \leq \tau' \left( C^* + \frac{1}{2^{m+1}} \right) \] (by inequality (2.6))

\[ \leq (\tau - \tau_m) \left( C^* + \frac{1}{2^{m+1}} \right) \] (by inequality (2.5) and definition of \( \tau' \))

By inequality (2.12),

\[ \frac{1}{\tau} C_{P*}(\tau) \leq \frac{1}{\tau} \left( \tau_m \left( C^* + \frac{1}{2^{m-1}} \right) + \frac{\tau - \tau_m}{\tau} \left( C^* + \frac{1}{2^{m+1}} \right) \right) \]

\[ \leq C^* + \frac{1}{2^{m-1}} + \frac{1}{2^{m+1}} \]

\[ \leq C^* + \frac{1}{2^{m-2}}. \]

In summary,

\[ \frac{1}{\tau} C_{P*}(\tau) \leq C^* + \frac{1}{2^{m-2}}, \quad \forall \tau \in [\tau_m, \tau_{m+1}], \ m = 1, 2, \ldots \]

That is,

\[ C_{P*} = \limsup_{\tau \to \infty} \frac{1}{\tau} C_{P*}(\tau) \leq C^*. \]

The proof is completed by recalling that \( C_{P*} \geq C^* \).

Remark: This proof extends and corrects a proof by Hassin and Megiddo [12] (1989) for a related inventory problem. The original proof in Hassin and Megiddo established results only for all \( t = \tau_m \), but not for other values of \( \tau \), which invalidates the result. Our proof corrects this by lengthening the times during which policies \( P_m \) are used (equation (2.8)) so as to achieve the result for all \( t \geq 0 \).
Note that the proof of this theorem does not require the existence of an optimal finite horizon policy, although this is easy to prove for the models under consideration.

Sometimes, we are not only interested in the existence of an optimal infinite horizon policy, but also in properties satisfied by an optimal policy such as the Nonpositive Inventory Ordering Property and the Latest Ordering Property (defined in the next section). The following corollary gives us a vehicle to extend the properties satisfied by finite horizon policies to an optimal infinite horizon policy.

**Corollary 2.3.3 (Existence of Optimal Solutions Satisfying Property \( \mathcal{P} \))**

Suppose that the Zero Initial Inventory and the Nonpositive Ending Inventory Assumptions hold for inventory models IM\((G,MT,NN,IF,GB)\) and IM\((G,MT,NN,FF,NB)\). Consider a property \( \mathcal{P} \) satisfying the following conditions:

1. **(Nonpositive Ending Inventory under Property \( \mathcal{P} \))**
   
   For any finite horizon \([0,T)\) and any feasible policy \( P \) over this horizon, there exists a feasible policy \( P' \) satisfying:
   
   1.1. property \( \mathcal{P} \);
   
   1.2. Nonpositive Ending Inventory Assumption; and
   
   1.3. \( P' \) dominates \( P \).

2. **(Stability of property \( \mathcal{P} \))**
   
   If all finite horizon feasible policies \( P^1, P^2, \ldots \) satisfy property \( \mathcal{P} \), then the infinite horizon policy \( P = (P^1P^2\ldots) \) defined by concatenation method CM also satisfies property \( \mathcal{P} \).

Then there exists an optimal policy satisfying property \( \mathcal{P} \).

**Proof.** Follows immediately from the previous proof by assuming that all the components \( P^m \) satisfy property \( \mathcal{P} \).

\[ \square \]
Remark: Condition 2, the Stability Condition is satisfied by many useful properties, such as "Zero Inventory Ordering" property, as we shall see later in this thesis. It is not satisfied, however, by some other desirable properties of optimal policies such as that of being periodic. Every finite horizon policy is periodic, with period at most the length of the horizon. Yet, this does not imply that infinite horizon policies need to be periodic (consider for instance the case of two unrelated products, where one EOQ frequency is an irrational multiple of the other). The question of finding sufficient properties for the existence of an optimal policy which is periodic is a very interesting one, but it is beyond the scope of this thesis (See Hassin and Megiddo [12] (1989) for one such example).

We conclude this section by proving a lemma mentioned earlier.

Lemma 2.3.4
For the inventory models $IM(G,MT,NN,IF,GB)$ and $IM(G,MT,NN,FF,NB)$ the Nonpositive Ending Inventory Assumption holds.

Proof. Whenever there is a positive actual inventory level $I_i^T$ of product $i$ at the ending instant $T$, this "extra" inventory $I_i^T$ may be removed as well as all the inventories of $i$'s predecessors, which supplies the "extra" inventory $I_i^T$. By doing so, the actual inventory level of each node at any instant does not increase, and neither does the total holding cost. The monotonicity of the setup cost function $K$ ensures that the total setup cost does not increase. Therefore, the total inventory cost does not increase either. This shows that the Zero Ending Inventory Assumption holds. (For a more detailed proof based on the "Remove" algorithm see section 2.6.3.)
2.4 Properties of Optimal Inventory Policies

As the behaviour of inventory models is quite different when the production rate is finite or infinite, for simplicity we consider only the infinite production rate inventory model in this section.

We demonstrate that for the inventory model IM(G,MT,NN,IF,GB), there exists an optimal infinite horizon policy which satisfies the following three properties: Nonpositive Inventory Ordering, Latest Ordering and Nonnegative Filling. Therefore, from the perspective of minimizing the long-run average cost, we may restrict ourselves to policies satisfying these properties. For a replenishment policy satisfying the Nonpositive Inventory Ordering Property and the Latest Ordering Property, we show that there is an explicit expression of echelon inventory levels and route order quantities in terms of ordering instants. This formulation implies that the ordering instants uniquely determine the order quantities. Therefore, the replenishment policy needs only be described by the ordering instants \( t_p \). The definitions of these concepts appear below.

We first introduce network-related concepts, such as paths, routes, and inventory-related concepts, such as actual inventory levels and echelon inventory levels. We then present properties of optimal inventory policies.

Path \( r = (i_1, i_2, i_3, \ldots, i_m) \) is a sequence of nodes \( i_1, i_2, \ldots, i_m \in N \) such that all arcs \( (i_1, i_2), (i_2, i_3), \ldots, (i_{m-1}, i_m) \in A \). The first node of \( r \) is \( f_r \triangleq i_1 \), the second node of \( r \) is \( s_r \triangleq i_2 \), and the last node of \( r \) is \( \ell_r \triangleq i_m \). Set \( \{i_1, i_2, \ldots, i_m\} \) is called the node set of path \( r \), and \( i \in r \) means that node \( i \in \{i_1, i_2, \ldots, i_m\} \). Let \( |r| \triangleq m \) denote the length of path \( r \). Whenever convenient, we also say that arc \( (i_j, i_{j+1}) \) belongs to path \( r \), for all \( j = 1, \ldots, m - 1 \).

Route \( r = (i_1, i_2, i_3, \ldots, i_m) \) is a path terminating at a node \( i_m \in D_N \). If there is no further specification, \( r \) generally denotes a route instead of a path. Say
Chapter 2. Optimal Inventory Policies

\[ r' = (i_j, \ldots, i_m) \] with \( j \in \{1, \ldots, m\} \) is a sub-route of \( r \), denoted by \( r' \subseteq r \). Let \( r \setminus f_r \triangleq (i_2, \ldots, i_m) \) be the immediate sub-route of \( r \).

\( R_i \) is the set of all routes \( r \) starting at node \( i \in N \), i.e., such that \( f_r = i \).

\[ R \triangleq \bigcup_{i \in N} R_i \] is the set of all routes \( r \) in \( G(N,A) \).

\( \lambda_{ij} \) is the number of units of product \( i \) required to produce 1 unit of product \( j \), for \((i,j) \in A \).

\[ \lambda_r \triangleq \begin{cases} 1 & \text{if } |r| = 1, \\ \lambda_{i_1i_2}\lambda_{i_2i_3}\ldots\lambda_{i_{m-1}i_m} & \text{if } |r| \geq 2 \end{cases} \] is the number of units of product \( f_r \) required to produce 1 unit of product \( \ell_r \) through path \( r \).

\[ d_r \triangleq \lambda_r d_{\ell_r} \] is the induced route demand rate at node \( f_r \) from route \( r \).

\[ d_i \triangleq \sum_{r \in R_i} d_r \] is the induced (node) demand rate of node \( i \). Note that \( d_r \) is the "long-run average" demand rate of node \( f_r \) from route \( r \), as is induced demand rate \( d_i \). The "actual" induced route demand at an internal node \( f_r \notin D_N \) is generated by the orders from its immediate successor on route \( r \) and varies over time. The "actual" induced demand at node \( i \) is the sum of the orders from "all" its successors.

**A Route Order Quantity** \( Q^t_r \) is that part of order quantity \( Q^t_i \) at node \( i \) and ordering instant \( t \) which is used to satisfy the (induced) demand from node \( \ell_r \) along route \( r \).

By definition, \( Q^t_i = \sum_{r \in R_i} Q^t_r \).

\( I^t_r \) is the actual route inventory at node \( f_r \) for route \( r \) at instant \( t \). By definition, \( I^t_i = \sum_{r \in R_i} I^t_r \) for all \( t \). Similarly, \( I^{-}_r = \lim_{t' \downarrow t} I^t_r \) and \( I^{+}_r = \lim_{t' \uparrow t} I^t_r \).

\( E^t_r \) is the echelon inventory on route \( r \) at instant \( t \). It is the total inventory of good \( f_r \).
Chapter 2. Optimal Inventory Policies

on route \( r \), defined as follows:

\[
E_r^t \triangleq \sum_{r' \subseteq r} \left( \frac{\lambda_{r'}}{\lambda_{r'}} \right) I_{r'}^t
\]

where routes \( r' \) are subroutes of \( r \) having the same last node. Similarly, \( E_r^{t-} = \lim_{t' \nearrow t} E_r^{t'} \) and \( E_r^{t+} = \lim_{t' \searrow t} E_r^{t'} \).

Given a replenishment policy \( P = (t_P, Q_P) \), the actual (node) inventories \( I_i^t \) are uniquely defined by policy \( P \). However, the actual route order quantities \( Q_r^t \) with \( r \in R_i \) are not uniquely defined by \( Q_i^t \), neither are the actual route inventories \( I_r^t \) nor the echelon inventories \( E_r \). See the example in Figure 2.2 on page 31.

**Nonpositive Inventory Ordering Property:** If a replenishment policy \( P \) orders quantity \( Q_i^t > 0 \) at node \( i \) and ordering instant \( t \) with actual inventory level \( I_i^{t-} \leq 0 \), it is said that policy \( P \) satisfies the Nonpositive Inventory Ordering Property for node \( i \) at instant \( t \).

Note that if node \( i \) does not allow backlogging, then the Nonpositive Inventory Ordering Property reduces to the well-known Zero Inventory Ordering Property, that is, policy \( P \) orders positive quantities at node \( i \) only at instants \( t \) where \( I_i^{t-} = 0 \). This is the case for all internal nodes under either \( GB \), \( B \) or \( NB \) model.

If policy \( P \) satisfies the Nonpositive Inventory Ordering Property for every node \( i \in N \) at every ordering instant, then policy \( P \) is said to satisfy the Nonpositive Inventory Ordering Property.

The concepts defined below are used in the proofs of the following lemmas.

**Node-instant point** \((i, t)\) is a pair of node \( i \) and ordering instant \( t \). Over any finite horizon \([0, T)\), if \( i < j \) or, if \( i = j \) and \( t < u \), we say that node-instant point \((i, t)\) is less than node-instant point \((j, u)\), written as \((i, t) < (j, u)\). (This is the usual lexicographic order on \( N \times [0, T)\).)
Figure 2.2: The non-uniqueness of route ordering quantities.
Modification Step: Given a replenishment policy $P$, policy $P' = (t_P, Q_P')$ is defined by applying the following Modification Step $\Delta(i, t', t'', x)$, which reduces the order quantity at node $i$ and instant $t'$ by quantity $x$ and increases that at instant $t''$ by the same quantity $x$, that is,

$$Q_{jP'}^v = \begin{cases} Q_{iP}^v - x & \text{if } j = i \text{ and } v = t', \\ Q_{iP}^v + x & \text{if } j = i \text{ and } v = t'', \\ Q_{jP}^v & \text{otherwise.} \end{cases}$$

See Figure 2.3 on page 33.

Lemma 2.4.1 (Property of the Modification Step)

Given a feasible policy $P$ for the inventory model IM(G,MT,NE,IF,GB). Let $P$ be a feasible policy such that $I_{iP}^{t''-} > 0$ at ordering instant $t''$. Let $t' = \max \{v | Q_{iP}^v > 0, 0 < v < t''\}$ be the last ordering instant before $t''$ at node $i$ and $x = \min \{I_{iP}^{t''-}, Q_{iP}^t\} > 0$. Then policy $P'$ generated from policy $P$ by executing of the Modification Step $\Delta(i, t', t'', x)$ is feasible and dominates policy $P$. Besides, the only inventory levels affected by this change are those of $i$ and its immediate predecessors, and these changes are restricted to time interval $(t', t'')$.

Proof. First, observe that, by definition of $t'$ and $x$, one has $I_{iP}^t \geq I_{iP}^{t''-} > 0$ for all $t \in (t', t'')$. Thus for policy $P'$, $I_{iP'}^t \geq I_{iP}^{t''-} - x \geq 0$ for all $t \in (t', t'')$. If $i$ is an internal node, all orders from $i$'s successors during time interval $(t', t'')$ may still be satisfied. Thus, the only nodes affected by the Modification Step are $i$ itself and its predecessors. Each immediate predecessor $j \in p(i)$ has a quantity at least $\lambda_{ij}Q_{ij}^t \geq x$ available at instant $t'$, so the only change is an increase in its inventory level during time interval $(t', t'')$. Therefore, the non–immediate predecessors of $i$ are not affected by the Modification Step. The total holding and backlogging costs $C_{HP}(t)$ and $C_{HP'}(t)$ of policies $P$ and $P'$,
Figure 2.3: Modification Step $\Delta(i, t', t'', I''_{iP})$ and how it may introduce new non-zero-inventory ordering instants at predecessor node $j \in p(i)$

Before executing $\Delta(i, t', t'', I''_{iP})$

After executing $\Delta(i, t', t'', I''_{iP})$
respectively, during time interval $[0,t)$ satisfy, for all $t \in [0,T]$: 
\[
C_{HP}(t) - C_{HP'}(t) = \int_t^{\min\{t,t''\}} x \left( h^t_i - \sum_{j \in p(i)} \lambda_{ij} h^t_j \right) dt
\]
(\text{where } \int_a^b f(t)dt = 0, \text{ if } a > b). \text{ The only change in setup costs occurs at node } i \text{ and instant } t', \text{ and only if } x = Q^t_i. \text{ Because of the monotonicity assumption } MT, \text{ total setup costs will satisfy } C_{SP}(t) \geq C_{SP'}(t) \text{ for all } t \in [0,T). \text{ This completes the proof of the lemma.} \square

Note that this Lemma, and the next two results, requires both the monotonicity assumption $MT$ on setup costs and the nonnegativity assumption $NE$ on echelon inventory costs.

The Nonpositive Inventory Ordering Property is probably one of the best known properties of optimal policies. We prove it below for a rather general model.

**Proposition 2.4.2 (Finite Horizon Nonpositive Inventory Ordering Property)**

For any finite cost, feasible replenishment policy $P = (t_P, Q_P)$ over a finite horizon $[0,T)$ for the inventory model $IM(G,MT,NE,IF,GB)$, there exists a replenishment policy $P' = (t_P, Q_P')$, which satisfies the Nonpositive Inventory Ordering Property and dominates policy $P$.

**Proof.** If node $i$ violates the Nonpositive Inventory Ordering Property at instant $t$, then we call node–instant point $(i,t)$ a bad point.

Let $(i,t'')$ be the (lexicographically) smallest bad point of policy $P$, then $Q^t_{iP} > 0$ and $l^t_{iP} > 0$. Let $t' = \max \{v | Q^v_{iP} > 0, 0 \leq v < t''\}$ as defined in Lemma 2.4.1. (Note that $t'$ exists, as $Q^0_{iP} > 0$.) Because $t'$ is not a bad point, we have $l^{t'}_{iP} = 0$. Because there are no positive order quantities for node $i$ in time interval $(t',t'')$, the order quantity $Q^t_{iP}$ must satisfy all the demands to node $i$ in time interval $[t',t'')$ and leave the actual inventory
level $I_{i,p}^{t''} > 0$ at instant $t''$. Hence, the Modification Step $\Delta(i,t',t'',I_{i,p}^{t''})$ generates a feasible replenishment policy $P'$ from $P$, and it reduces the inventory level $I_{i,p}^{t''} > 0$ to zero, that is, point $(i,t'')$ is not a bad point of policy $P'$. By Lemma 2.4.1, policy $P'$ dominates policy $P$.

Note however that the Modification Step may generate new bad points $(j,s)$ for node $j \in p(i)$, i.e., $j > i$, therefore, $(j,s) > (i,t'')$, see also Figure 2.3 on page 33. In other words, the smallest bad point of policy $P'$ is strictly greater than the smallest bad point of policy $P$. As there are only a finite number of node-instant points in the finite horizon $[0,T)$, all the bad points in $[0,T)$ will be removed by a finite number of Modification Steps. This produces a policy which has no bad point and dominates policy $P$. □

Note that, for the purpose of studying infinite horizon policies, we need not prove the existence of an optimal policy for finite horizon. This result, however, follows easily.

**Corollary 2.4.3 (Infinite Horizon Nonpositive Inventory Ordering Property)**

*Given the inventory model IM(G,MT,NE,IF,GB), there exists an optimal infinite horizon policy satisfying the Nonpositive Inventory Ordering Property.*

**Proof.** Let property $\mathcal{P}$ be the Nonpositive Inventory Ordering Property. From the previous Proposition, it follows that all the conditions in Corollary 2.3.3 hold. Applying Corollary 2.3.3 completes the proof. □

**Latest Ordering Property:** A replenishment policy $P = (t, Q)$ satisfies the Latest Ordering Property, if the following holds for every node $i$ and every ordering instant $t = t_i^k$: the order quantity $Q_i^t$ is positive iff there exists a positive order quantity $Q_j^u > 0$ of some node $j \in s(i)$ at some ordering instant $u \in [t_i^k, t_i^{k+1})$. \(\Box\)
Figure 2.4: The Latest Ordering Property and the Zero Inventory Ordering Property do not imply each other

Latest Ordering means that the positive quantity ordered at the possible ordering instants (given by the replenishment policy) are delayed as late as possible. This does not mean that the corresponding ordering instants have to coincide with ordering instants of the successors. See the example in Figure 2.10 on page 60 of section 2.6.2, where warehouse 3 orders product 2 at the instants at which retailer 2 does not order.

Note that in this definition, the Latest Ordering Property and the Nonpositive Inventory Ordering Property do not imply each other. See the example in Figure 2.4 on page 36. In this figure, $t_2^1, \cdots, t_2^6$ are all the possible ordering instants of policy $P$ at node 2. At ordering instant $t_2^1$, policy $P$ satisfies the Zero Inventory Ordering Property, but does not satisfy the Latest Ordering Property. At ordering instant $t_2^5$, policy $P$ satisfies the Latest Ordering Property, but does not satisfy the Zero Inventory Ordering Property.
Proposition 2.4.4 (Finite Horizon Latest Ordering Property)

For any finite cost, feasible replenishment policy \( P = (t, Q) \) over a finite horizon \([0, T)\) for an inventory model \( IM(G,MT,NE,IF,GB) \), there exists a finite horizon replenishment policy \( P' = (t, Q') \), such that \( P' \) satisfies the Latest Ordering Property and dominates policy \( P \).

Proof. The proof is similar to that for Proposition 2.4.2. If node \( i \) violates the Latest Ordering Property at instant \( t \), then we call the node-instant point \((i, t)\) a bad point.

Let node-instant point \((i, t')\) be the (lexicographically) smallest bad point. Then \( Q_{iP}' > 0 \). Let ordering instant \( t'' \) be the next ordering instant at node \( i \) after \( t' \). Because \((i, t')\) is a bad point, there is no demand from any node \( j \in s(i) \) during time interval \([t', t'')\). The Modification Step \( \Delta(i, t', t'', Q_{iP}') \), which shifts the order quantity \( Q_{iP}' \) for ordering instant \( t' \) to ordering instant \( t'' \), generates a feasible policy \( P' \) without order quantity at instant \( t' \), i.e., \((i, t'')\) is not a bad point for policy \( P' \). Therefore, policy \( P' \) has a larger smallest bad point than policy \( P \) and dominates policy \( P \). By executing such Modification Step a finite number of times produces a policy which has no bad point and dominates policy \( P \).

Corollary 2.4.5 (Infinite Horizon Latest Ordering Property)

For inventory model \( IM(G,MT,NE,IF,GB) \), there exists an optimal infinite horizon policy satisfying the Latest Ordering Property.

Proof. The proof is similar to that for Corollary 2.4.3.

The following concepts related to ordering instants are useful in expressing echelon inventories and order quantities in terms of ordering instants. Below, the ordering instants \( \theta_r \) are related to the first node \( f_r \) of route \( r \), and the ordering instants \( \tau_r \) are
related to the last node \( \ell_r \) of route \( r \). Note that these ordering instants depend not only on nodes, such as \( f_r \) and \( \ell_r \), but also on route \( r \). Given all the possible ordering instants \( t^k_i \) of a policy \( P \), \( \theta_r \) and \( \tau_r \) are recursively defined by \( t^k_i \) along route \( r \). The parameter \( t \) in the following definition reflects the relationship between any instant \( t \) (not necessarily an ordering instant) and the original ordering instant. To ease the illustration, we may assume that policy \( P \) satisfies Nonpositive Ending Inventory and Latest Ordering Properties, even though the following concepts do not rely on these properties. To simplify indexing, we assume that \( Q^k_i > 0 \) at every ordering instant \( t^k_i \) if \( i \in D_N \). To understand the following four concepts, it is better to refer to the example shown in Figure 2.5 on p. 39 at the same time.

\[
\bar{\theta}^k_r \triangleq \begin{cases} 
 t^k_{f_r}, & \text{if } |r| = 1 \\
 \max\{t^k_{f_r}, t^k_{r \setminus f_r}, \ell = 1, 2, \ldots\}, & \text{if } |r| \geq 2
\end{cases}
\]

is the latest ordering instant at node \( f_r \), at which the order quantity can be used to satisfy the demand from node \( \ell_r \) at instant \( t^k_{\ell_r} \) through route \( r \). If policy \( P \) satisfies Nonpositive Ending Inventory and Latest Ordering Properties, then route order quantity \( Q^k_i \) is positive only at these ordering instants \( \bar{\theta}^k_r \). For simplicity, we say that latest ordering instant \( \bar{\theta}^k_r \) of node \( f_r \) corresponds to latest ordering instant \( \bar{\theta}^k_{\ell_r} \) at node \( \ell_r \) through route \( r \).

Consider the example in Figure 2.5. By definition \( \bar{\theta}^k_{r_1} = t^k_1 \) for all \( k \). Therefore, we have \( \bar{\theta}^3_{r_3} = \max\{t^3_3 \leq \theta^3_{r_1}\} = \max\{t^3_3 \leq t^3_1\} = t^3_3 \) and \( \bar{\theta}^4_{r_3} = \max\{t^4_3 \leq \theta^4_{r_1}\} = \max\{t^4_3 \leq t^4_1\} = t^4_3 \), that is, \( \bar{\theta}^3_{r_3} \) and \( \bar{\theta}^4_{r_3} \) have the same value. Because of this, the following concept is introduced to eliminate the duplicated indices.

\[
\Theta_r \triangleq \{\theta^k_r|k = 1, 2, \ldots\}
\]

is the latest ordering instant set at node \( f_r \) for route \( r \). Because some of the instants \( \theta^k_r \) may coincide, we let \( \Theta_r = \{\theta^\ell_r|\ell = 1, 2, \ldots\} \) where \( \theta^1_r < \theta^2_r < \cdots < \theta^\ell_r < \cdots \) are all the distinct values of the elements in \( \Theta_r \). For example,
Figure 2.5: Order instants $\theta_r$ and $\tau_r$.

Part I, Actual Inventory Levels

$r_1 = 1, r_2 = 2, r_3 = (3, 1), r_4 = (3, 2), r_5 = (4, 3, 1), r_6 = (4, 3, 2),$
Figure 2.5: Order instants $\theta_r$ and $\tau_r$.

Part II, Echelon Inventory Levels
in Figure 2.5, we have $\theta^2_{r_3} = t^3_3 = \bar{\theta}^3_{r_3} = \bar{\theta}^4_{r_3}$.

Based on these ordering instants, we may establish two functions of any instant $t$. The value of $\theta_r(t)$ is taken from $\Theta_r$, which is the subset of the ordering instants at first node $f_r$ of route $r$, and the value of $\tau_r(t)$ is on the set of the ordering instants at last node $\ell_r$ of route $r$.

$$\theta_r(t) \triangleq \min \{\theta^k_r | \theta^k_r \geq t, \ k = 1, 2, \ldots \},$$

for any $t > 0$, is the first node earliest ordering instant after instant $t$, i.e., the earliest ordering instant of first node $f_r$ of route $r$ after instant $t$. For example, in Figure 2.5, if $t \in (\theta^2_{r_3}, \theta^3_{r_3}]$, then we have $\theta_{r_3}(t) = \theta^2_{r_3} = t^4_3$.

$$\tau_r(t) \triangleq \begin{cases} \theta_r(t), & \text{if } |r| = 1, \\ \tau_{r \setminus f_r}(\theta_r(t)), & \text{if } |r| \geq 2 \end{cases}$$

for any $t > 0$, is the last node earliest ordering instant for instant $t$, which is the earliest ordering instant of the last node $\ell_r$ of route $r$ after instant $t$. For example, in Figure 2.5, if $t \in (\theta^2_{r_3}, \theta^3_{r_3}]$, then we have $\tau_{r_3}(t) = \tau_{r_3}(\theta_{r_3}(t)) = \tau_{r_1}(\theta^3_{r_3}) = \theta_{r_1}(t^4_3) = t^5_1$.

In the following, we list the all the values of $\theta_r$ and $\tau_r$ in terms of $t^k_r$ for the example in Figure 2.5. As $|r_1| = |r_2| = 1$, we have $\theta^k_{r_1} = t^k_1$ and $\theta^k_{r_2} = t^k_2$. The four tables in Table 2.1 show how to express $\theta^k_r$ in terms of $t^k_r$, for $|r| \geq 2$. And the six tables in Table 2.2 show that $\theta_r(t)$ and $\tau_r(t)$ as the functions of $t$.

Recall that $d_r \triangleq \lambda_r d_{\ell_r}$ is the induced demand on route $r$, and $I_{r(t)}^\ell$ is the actual inventory at instant $\tau_r(t)$ used for external demand of node $\ell_r$. If a policy satisfies the Nonpositive Inventory Ordering Property, then $I_{r(t)}^\ell \leq 0$, with equality if backlogging is not allowed.

The next Proposition shows that the route echelon inventory of an optimal policy can be explicitly expressed in terms of ordering instants and inventory levels at the end products only.
Table 2.1: Expressing $\theta_{r3}, \theta_{r4}, \theta_{r5}, \theta_{r6}$ in terms of $t_3, t_4$

<table>
<thead>
<tr>
<th>$\theta^1_{r3}$</th>
<th>$\theta^2_{r3}$</th>
<th>$\theta^3_{r3}$</th>
<th>$\theta^4_{r3}$</th>
<th>$\theta^5_{r3}$</th>
<th>$\theta^6_{r3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_3$</td>
<td>$t^2_3$</td>
<td>$t^3_3$</td>
<td>$t^4_3$</td>
<td>$t^5_3$</td>
<td>$t^6_3$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\bar{\theta}^1_{r3}$</th>
<th>$\bar{\theta}^2_{r3}$</th>
<th>$\bar{\theta}^3_{r3}$</th>
<th>$\bar{\theta}^4_{r3}$</th>
<th>$\bar{\theta}^5_{r3}$</th>
<th>$\bar{\theta}^6_{r3}$</th>
<th>$\bar{\theta}^7_{r3}$</th>
<th>$\bar{\theta}^8_{r3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_3$</td>
<td>$t^2_3$</td>
<td>$t^3_3$</td>
<td>$t^4_3$</td>
<td>$t^5_3$</td>
<td>$t^6_3$</td>
<td>$t^7_3$</td>
<td>$t^8_3$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\theta^1_{r4}$</th>
<th>$\theta^2_{r4}$</th>
<th>$\theta^3_{r4}$</th>
<th>$\theta^4_{r4}$</th>
<th>$\theta^5_{r4}$</th>
<th>$\theta^6_{r4}$</th>
<th>$\theta^7_{r4}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_3$</td>
<td>$t^2_3$</td>
<td>$t^3_3$</td>
<td>$t^4_3$</td>
<td>$t^5_3$</td>
<td>$t^6_3$</td>
<td>$t^7_3$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\bar{\theta}^1_{r4}$</th>
<th>$\bar{\theta}^2_{r4}$</th>
<th>$\bar{\theta}^3_{r4}$</th>
<th>$\bar{\theta}^4_{r4}$</th>
<th>$\bar{\theta}^5_{r4}$</th>
<th>$\bar{\theta}^6_{r4}$</th>
<th>$\bar{\theta}^7_{r4}$</th>
<th>$\bar{\theta}^8_{r4}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_3$</td>
<td>$t^2_3$</td>
<td>$t^3_3$</td>
<td>$t^4_3$</td>
<td>$t^5_3$</td>
<td>$t^6_3$</td>
<td>$t^7_3$</td>
<td>$t^8_3$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\theta^1_{r5}$</th>
<th>$\theta^2_{r5}$</th>
<th>$\theta^3_{r5}$</th>
<th>$\theta^4_{r5}$</th>
<th>$\theta^5_{r5}$</th>
<th>$\theta^6_{r5}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_4$</td>
<td>$t^2_4$</td>
<td>$t^3_4$</td>
<td>$t^4_4$</td>
<td>$t^5_4$</td>
<td>$t^6_4$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\bar{\theta}^1_{r5}$</th>
<th>$\bar{\theta}^2_{r5}$</th>
<th>$\bar{\theta}^3_{r5}$</th>
<th>$\bar{\theta}^4_{r5}$</th>
<th>$\bar{\theta}^5_{r5}$</th>
<th>$\bar{\theta}^6_{r5}$</th>
<th>$\bar{\theta}^7_{r5}$</th>
<th>$\bar{\theta}^8_{r5}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_4$</td>
<td>$t^2_4$</td>
<td>$t^3_4$</td>
<td>$t^4_4$</td>
<td>$t^5_4$</td>
<td>$t^6_4$</td>
<td>$t^7_4$</td>
<td>$t^8_4$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\theta^1_{r6}$</th>
<th>$\theta^2_{r6}$</th>
<th>$\theta^3_{r6}$</th>
<th>$\theta^4_{r6}$</th>
<th>$\theta^5_{r6}$</th>
<th>$\theta^6_{r6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_4$</td>
<td>$t^2_4$</td>
<td>$t^3_4$</td>
<td>$t^4_4$</td>
<td>$t^5_4$</td>
<td>$t^6_4$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\bar{\theta}^1_{r6}$</th>
<th>$\bar{\theta}^2_{r6}$</th>
<th>$\bar{\theta}^3_{r6}$</th>
<th>$\bar{\theta}^4_{r6}$</th>
<th>$\bar{\theta}^5_{r6}$</th>
<th>$\bar{\theta}^6_{r6}$</th>
<th>$\bar{\theta}^7_{r6}$</th>
<th>$\bar{\theta}^8_{r6}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t^1_4$</td>
<td>$t^2_4$</td>
<td>$t^3_4$</td>
<td>$t^4_4$</td>
<td>$t^5_4$</td>
<td>$t^6_4$</td>
<td>$t^7_4$</td>
<td>$t^8_4$</td>
</tr>
</tbody>
</table>
### Table 2.2: Values of functions $\theta_{r_j}(t)$ and $\tau_{r_j}(t)$

<table>
<thead>
<tr>
<th>$t \in$</th>
<th>$(t_1^1, t_1^2)$</th>
<th>$(t_2^1, t_2^2)$</th>
<th>$(t_3^1, t_3^2)$</th>
<th>$(t_4^1, t_4^2)$</th>
<th>$(t_5^1, t_5^2)$</th>
<th>$(t_6^1, t_6^2)$</th>
<th>$(t_7^1, t_7^2)$</th>
<th>$(t_8^1, t_8^2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{r_1}(t)$ =</td>
<td>$t_1^1$</td>
<td>$t_2^1$</td>
<td>$t_3^1$</td>
<td>$t_4^1$</td>
<td>$t_5^1$</td>
<td>$t_6^1$</td>
<td>$t_7^1$</td>
<td>$t_8^1$</td>
</tr>
<tr>
<td>$\tau_{r_1}(t)$ =</td>
<td>$t_1^2$</td>
<td>$t_2^2$</td>
<td>$t_3^2$</td>
<td>$t_4^2$</td>
<td>$t_5^2$</td>
<td>$t_6^2$</td>
<td>$t_7^2$</td>
<td>$t_8^2$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t \in$</th>
<th>$(t_2^1, t_2^2)$</th>
<th>$(t_3^1, t_3^2)$</th>
<th>$(t_4^1, t_4^2)$</th>
<th>$(t_5^1, t_5^2)$</th>
<th>$(t_6^1, t_6^2)$</th>
<th>$(t_7^1, t_7^2)$</th>
<th>$(t_8^1, t_8^2)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{r_2}(t)$ =</td>
<td>$t_2^1$</td>
<td>$t_3^1$</td>
<td>$t_4^1$</td>
<td>$t_5^1$</td>
<td>$t_6^1$</td>
<td>$t_7^1$</td>
<td>$t_8^1$</td>
</tr>
<tr>
<td>$\tau_{r_2}(t)$ =</td>
<td>$t_2^2$</td>
<td>$t_3^2$</td>
<td>$t_4^2$</td>
<td>$t_5^2$</td>
<td>$t_6^2$</td>
<td>$t_7^2$</td>
<td>$t_8^2$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t \in$</th>
<th>$(\theta_1^{r_3}, \theta_2^{r_3})$</th>
<th>$(\theta_2^{r_3}, \theta_3^{r_3})$</th>
<th>$(\theta_3^{r_3}, \theta_4^{r_3})$</th>
<th>$(\theta_4^{r_3}, \theta_5^{r_3})$</th>
<th>$(\theta_5^{r_3}, \theta_6^{r_3})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{r_3}(t)$ =</td>
<td>$t_3^3$</td>
<td>$t_4^3$</td>
<td>$t_5^3$</td>
<td>$t_6^3$</td>
<td>$t_7^3$</td>
</tr>
<tr>
<td>$\tau_{r_3}(t)$ =</td>
<td>$t_3^4$</td>
<td>$t_4^4$</td>
<td>$t_5^4$</td>
<td>$t_6^4$</td>
<td>$t_7^4$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t \in$</th>
<th>$(\theta_1^{r_4}, \theta_2^{r_4})$</th>
<th>$(\theta_2^{r_4}, \theta_3^{r_4})$</th>
<th>$(\theta_3^{r_4}, \theta_4^{r_4})$</th>
<th>$(\theta_4^{r_4}, \theta_5^{r_4})$</th>
<th>$(\theta_5^{r_4}, \theta_6^{r_4})$</th>
<th>$(\theta_6^{r_4}, \theta_7^{r_4})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{r_4}(t)$ =</td>
<td>$t_4^3$</td>
<td>$t_5^3$</td>
<td>$t_6^3$</td>
<td>$t_7^3$</td>
<td>$t_8^3$</td>
<td></td>
</tr>
<tr>
<td>$\tau_{r_4}(t)$ =</td>
<td>$t_4^4$</td>
<td>$t_5^4$</td>
<td>$t_6^4$</td>
<td>$t_7^4$</td>
<td>$t_8^4$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t \in$</th>
<th>$(\theta_1^{r_5}, \theta_2^{r_5})$</th>
<th>$(\theta_2^{r_5}, \theta_3^{r_5})$</th>
<th>$(\theta_3^{r_5}, \theta_4^{r_5})$</th>
<th>$(\theta_4^{r_5}, \theta_5^{r_5})$</th>
<th>$(\theta_5^{r_5}, \theta_6^{r_5})$</th>
<th>$(\theta_6^{r_5}, \theta_7^{r_5})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{r_5}(t)$ =</td>
<td>$t_5^4$</td>
<td>$t_6^4$</td>
<td>$t_7^4$</td>
<td>$t_8^4$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\tau_{r_5}(t)$ =</td>
<td>$t_5^5$</td>
<td>$t_6^5$</td>
<td>$t_7^5$</td>
<td>$t_8^5$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$t \in$</th>
<th>$(\theta_1^{r_6}, \theta_2^{r_6})$</th>
<th>$(\theta_2^{r_6}, \theta_3^{r_6})$</th>
<th>$(\theta_3^{r_6}, \theta_4^{r_6})$</th>
<th>$(\theta_4^{r_6}, \theta_5^{r_6})$</th>
<th>$(\theta_5^{r_6}, \theta_6^{r_6})$</th>
<th>$(\theta_6^{r_6}, \theta_7^{r_6})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\theta_{r_6}(t)$ =</td>
<td>$t_6^4$</td>
<td>$t_7^4$</td>
<td>$t_8^4$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\tau_{r_6}(t)$ =</td>
<td>$t_6^5$</td>
<td>$t_7^5$</td>
<td>$t_8^5$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Figure 2.6: The echelon inventory level $E_r^t$ is a function of time $t$

![Diagram](image)

**Proposition 2.4.6 (Echelon Inventory)**

Suppose that replenishment policy $P$ for inventory model IM($G, r, \cdot, IF, GB$) satisfies the Nonpositive Inventory Ordering Property and the Latest Ordering Property, then the echelon inventory for each route $r$ can be represented as

$$E_r^t = d_r \left[ (\tau_r(t) - t) + \frac{1}{d_r} I_r^{\tau_r(t)} \right] \quad \forall r \in R. \quad (2.13)$$

For the non-backlogging case, the formula simplifies to $E_r^t = d_r [\tau_r(t) - t]$.

**Proof.** By induction on the length of route $r$:

(i) For $|r| = 1$, we have $r = \ell_r$, $d_r = d_{\ell_r}$, and $\theta_r(t) = \tau_r(t)$. Therefore,

$$E_r^t = I_r^t = d_r (\theta_r(t) - t) + I_r^{\theta_r(t)}$$

$$= d_r \left[ (\tau_r(t) - t) + \frac{1}{d_r} I_r^{\tau_r(t)} \right].$$

See Figure 2.6 on page 44.

(ii) Suppose that equation 2.13 holds for $|r| = m$. Let $|r| = m + 1 \geq 2$ and $t_1 = \theta_r(t)$. Then product $f_r$ orders positive quantity for route $r$ at instant $t_1$. Recall that for the
internal demands we do not allow backlogging. The Nonpositive Inventory Ordering Property thus reduces to the Zero Inventory Ordering Property. Therefore, the route inventory level \( I_{r}^{t_1} = 0 \). Note that \( |r \setminus f_r| = m, \ell_r = \ell_{r \setminus f_r}, \lambda_{r, s_r} d_{r \setminus f_r} = d_r \), and \( \tau_{r \setminus f_r} (t_1) = \tau_{r \setminus f_r} (\theta_r (t)) = \tau_r (t) \). We have

\[
E_{r}^{t_1} = \sum_{r' \subseteq r} \lambda_{r', r} I_{r'}^{t_1} - \quad \text{(by definition)}
\]

\[
= \sum_{r' \subseteq r \setminus f_r} \lambda_{r', r} I_{r'}^{t_1} - \quad \text{(because } I_{r}^{t_1} = 0 \text{)}
\]

\[
= \lambda_{f_r, s_r} E_{r \setminus f_r}^{t_1} \quad \text{(by definition)}
\]

\[
= \lambda_{f_r, s_r} d_{r \setminus f_r} \left( (\tau_{r \setminus f_r} (t_1) - t_1) + \frac{1}{d_{r \setminus f_r}} I_{r \setminus f_r}^{\tau_r (t_1)} \right) \quad \text{(by induction)}
\]

\[
= d_r \left( (\tau_r (t) - t_1) + \frac{1}{d_{r \setminus f_r}} I_{r \setminus f_r}^{\tau_r (t)} \right).
\]

Because there is no order during time interval \([t, t_1)\), we have

\[
E_{r}^{t} = (t_1 - t) d_r + E_{r}^{t_1} - \quad \text{(by definition)}
\]

\[
= d_r \left( (\tau_r (t) - t) + \frac{1}{d_{r \setminus f_r}} I_{r \setminus f_r}^{\tau_r (t)} \right).
\]

This completes the proof.

Note that Proposition 2.4.6 extends the Lemma 5.8 on p. 154 in Zheng's Thesis [39] (1987) to the backlogging case.

By previous lemmas, there exist optimal policies satisfying the Nonpositive Inventory Ordering and the Latest Ordering Properties. The following Proposition also holds for such optimal policies. This new result reveals the relationship between route order quantities for such policies.

**Proposition 2.4.7 (Route Order Quantities)**

*Suppose that replenishment policy \( P = (t, Q) \) for the inventory model IM(G,MT,NE,IF,GB)*
satisfies the Nonpositive Inventory Ordering Property and the Latest Ordering Property. Then for any route \( r \in R \) order quantity \( Q_r^t \) satisfies:

\[
(a) \quad Q_r^t = \begin{cases} 
(t_{k+1}^t - t_{j_r}^k) d_{j_r} + I_r^{k+1} - I_r^k, & \text{if } |r| = 1, \\
\lambda_{f_{r,s}} \sum_{s \in [t_{j_r}^k, t_{j_r}^{k+1})} Q_{r \setminus f_r}^s, & \text{if } |r| \geq 2, \\
0, & \text{otherwise}
\end{cases}
\]  

(2.14)

(2.15)

where \( I_r^j = I_{r, f_r}^j \) (and \( I_r^j = I_{r, f_r}^j \) resp.) is the inventory level of node \( f_r \) at instant \( t = t_{j_r}^j \) (and just before \( t = t_{j_r}^j \) resp.) for \( j = k, k+1 \).

(b) \( Q_r^t > 0 \) iff \( t \in \Theta_r \).

(2.16)

Proof. (a) Note that \( |r| = 1 \) implies \( r = f_r \) and the first part of equation (2.14) is simply the inventory balance equations.

Suppose that \( |r| \geq 2 \). From the inventory balance equation, if \( t = t_{k}^t \) and \( r \in R_i \), we have

\[
Q_r^k = \lambda_{f_{r,s}} \sum_{s \in [t_{j_r}^k, t_{j_r}^{k+1})} Q_{r \setminus f_r}^s + I_r^{k+1} - I_r^k,
\]

By definition, both order quantities \( Q_r^k \) and \( Q_r^{k+1} \) are positive. Because node \( i \notin D_N \), no backlogging is allowed at node \( i \). By the Nonpositive Inventory Ordering Property, which reduces the Zero Inventory Ordering Property, we have \( I_r^k = I_r^{k+1} = 0 \). Therefore, \( I_r^k = I_r^{k+1} = 0 \). This completes the proof of part (a).

(b) We prove part (b) by induction on the length of route \( r \). For \( |r| = 1 \), it is true by the definition of \( \tilde{b}_r^k \). Suppose (b) holds for \( |r| = m \). For \( |r| = m + 1 \), we have to show two implications.

Case 1: Suppose \( Q_r^t > 0 \), let \( t = t_{j_r}^k \). From equation (2.14), we know that there exists \( s \in [t_{j_r}^k, t_{j_r}^{k+1}) \) such that \( Q_{r \setminus f_r}^s > 0 \). By induction, \( s \in \Theta_{f_r} \), i.e., \( s = \tilde{b}_r^\ell \) for some \( \ell \). Therefore, by definition, \( t = \tilde{b}_r^\ell \), i.e., \( t \in \Theta_r \). That is, \( Q_r^t > 0 \) implies \( t \in \Theta_r \).
Case 2: Suppose $t \in \Theta_r$, let $t = t_{f_r}^k = \bar{t}_r^l$ for some $l$. By the definition of $\bar{t}_r^l$, we know that $s = \bar{t}_r^l \in [t_{f_r}^k, t_{f_r}^{k+1})$. By induction, $Q_{f_r}^s > 0$. From equation (2.14), we know that $Q_{f_r}^r > 0$. That is, $t \in \Theta_r$ implies $Q_r^t > 0$.

Therefore, by induction, (b) holds for any route.

Suppose that replenishment policy $P = (t, Q)$ for the inventory model IM(G,MT,NE,IF,GB) satisfies the Nonpositive Inventory Ordering and the Latest Ordering Properties. We make the following observations:

Observation 1. The route demand induced by route $r$ at node $f_r \notin D_N$ come from the orders of its sub-route $r \setminus f_r$. They are instantaneous, and occur at discrete points in time. Case (a) in Proposition 2.4.7 implies that every induced route demand at node $f_r$ is satisfied by "one" order of node $f_r$ from its predecessor. Therefore, the number of orders at node $f_r$ for route $r$ is always no greater than the number of orders at node $s_r$ for its sub-route $r \setminus f_r$ over any finite horizon.

Observation 2. Case (b) in Proposition 2.4.7 implies that the Latest Ordering Property holds not only for nodes but also for "routes". That is, we order each product for a route as late as possible: the Latest Ordering instant $t = \bar{t}_r^k$ of node $f_r$ is the only ordering instant at which the route order quantity $\bar{Q}_r^k \triangleq Q_r^t$ is positive. And each route order quantity $\bar{Q}_r^k$ is satisfied by only one route order quantity $\bar{Q}_r^{k'}$ of any its super-route $r' = (p_r, r)$.

Observation 3. For the inventory model IM(G,MT,NE,IF,NB), a special case of the above where backlogging is not allowed, the order quantities are uniquely defined by the ordering instants.

Observation 4. If $Q_{f_r}^k > 0$, it is still possible that $Q_r^k = 0$, for some route $r$ with $|r| \geq 2$. 
Chapter 2. Optimal Inventory Policies

**Observation 5.** Note that Proposition 2.5 (p.37) in Zheng [39] (1987) is an immediate result of this Proposition: There exists an optimal power–of–two policy $T$, where $T = (T_1, \ldots, T_n)$ is the replenishment cycle time vector for a power–of–two policy, with Last Minute Ordering Property, which means node nested property in our thesis, see definition later: If $d_i = 0$ (i.e., if $i$ is not an end–product), then $T_i \geq \min \{T_j : j \in S(i)\}$.

In the following, we express the holding and backlogging cost rates in terms of *echelon* inventory levels instead of *actual* inventory levels. Recall that $f_r$ denotes the first node of route $r$, and $R$ is the set of all routes in graph $G(N, A)$.

**Proposition 2.4.8 (First Expression of $H^t$)**

For the inventory model IM($G, \cdot, NN, IF, GB$), the total holding and backlogging cost rate at any instant $t$ is given by:

$$H^t = \sum_{r \in R} h_r^t E_r^t + \sum_{i \in D_N} L_i^t(b_i^t + h_i^t).$$

**Proof.** By induction on the maximum length from initial nodes (the node without predecessors) to node $i$, we verify that the following equation holds:

$$h_i^{t'} = h_i^t + \sum_{j \in P(i)} \lambda_{ij} h_j^{t'} = \sum_{r \in G_i} \lambda_r h_{f_r}, \quad (2.17)$$

where $G_i$ is the set of all “paths” which start from a node $j \in P(i) \cup \{i\}$ and end at node $i$. Note that $I_i^t = T_i^t - I_i^t$, $I_i^t = I_i^t + L_i^t$ and $I_i^t = 0$ if node $i \notin D_N$, then we have:

$$H^t = \sum_{i \in D_N} (T_i^t h_i^{t'} + L_i^t b_i^t) + \sum_{i \in N \setminus D_N} I_i^t h_i^{t'}$$

$$= \sum_{i \in D_N} \left[ (T_i^t h_i^{t'} + L_i^t (b_i^t + h_i^{t'})) + \sum_{i \in N \setminus D_N} I_i^t h_i^{t'} \right]$$

$$= \sum_{i \in N} I_i^t h_i^{t'} + \sum_{i \in D_N} I_i^t (b_i^t + h_i^{t'}),$$

and

$$\sum_{i \in N} h_i^{t'} I_i^t = \sum_{i \in N} \left[ \left( \sum_{r_1 \in G_i} \lambda_{r_1} h_{f_{r_1}} \right) \left( \sum_{r' \in R_i} I_{r'}^t \right) \right] = \sum_{i \in N} \sum_{r_1 \in G_i} \sum_{r' \in R_i} (\lambda_{r_1} h_{f_{r_1}} I_{r'}^t).$$

Figure 2.7: Decomposing all the routes passing through node $i$

For any route $r = (f_r, \ldots, \ell_r) \in R$, let $i \in r$, $r_1 = (f_r, \ldots, i)$ and $r' = (i, \ldots, \ell_r)$. Then $r_1 \in G_i$, $r' \in R_i$ and $\lambda_{r_1} = \lambda_r / \lambda_{r'}$. Rearrange the summation in the previous equation by first going through all the routes $r \in R$, then collecting all terms with $i \in r$ symbolically, and reassigning to subroutes:

$$\sum_{i \in N} \sum_{r_1 \in G_i} \sum_{r' \in R_i} = \sum_{r \in R} \sum_{i \in r} \sum_{r' \subset r} \sum_{r' \in R}.$$

See Figure 2.7.

Therefore,

$$\sum_{i \in N} h_i^t I_i^t = \sum_{r \in R} h_r^t \sum_{r' \subset r} \left( \frac{\lambda_r}{\lambda_{r'}} I_r^t \right) = \sum_{r \in R} h_r^t E_r^t.$$

This completes the proof.

Consider inventory model $IM(G, \cdot, CT, IF, GB)$, where the holding and backlogging cost rates are constant. If a replenishment policy satisfies the Nonpositive Inventory Ordering and the Latest Ordering Properties, we obtain an explicit expression of holding and backlogging cost rates in terms of ordering instants.
Corollary 2.4.9 (Second Expression of $H^t$)

Suppose that replenishment policy $P$ for inventory model $IM(G, \cdot, CT, IF, GB)$ satisfies the Nonpositive Inventory Ordering and the Latest Ordering Properties. Then the total holding and backlogging cost rate at instant $t$ is given by

$$H^t = \sum_{r \in R} h_r d_r \left[ (r_r(t) - t) + \frac{1}{d_r} T_r^r(t) \right] + \sum_{i \in D_N} I_i^t \left( b_i + h_i^t \right),$$

where for $i \in D_N$

$$I_i^t = \begin{cases} -d_i(\theta_i(t) - t) + \left| I_i^{\theta_i(t)} \right| \geq 0, & \text{if } d_i(\theta_i(t) - t) \leq \left| I_i^{\theta_i(t)} \right|, \\ 0, & \text{otherwise.} \end{cases}$$

Proof. By definition, for $i \in D_N$, we have $I_i^t = \max \left\{ -d_i(\tau_i(t) - t) - T_i^r(t), 0 \right\}$. By Nonpositive Inventory Ordering Property, we have $T_i^r(t) \leq 0$. Applying Propositions 2.4.8 and 2.4.6, completes the proof.

The following Proposition matches the intuition that optimal policies should satisfy the Nonnegative Filling Property. However, the proof is not trivial for a general inventory system. This property was implicitly assumed in Mitchell’s paper [18] (1987), when he calculated the inventory level of the inventory system, without proof. This result is also used in Chapter 4 of this thesis for the backlogging case.
We will use the following notation in Proposition 2.4.10: consider a replenishment policy \( P \) and node–instant point \((j, \theta'_j)\), where node \( j \in D_N \) and \( \theta'_j \) is a latest ordering instant. For any route \( r \ni j \) (including node \( j \) itself), let \( t = \theta'_r \) be the latest ordering instant of node \( f_r \) incurred by the instant \( \theta'_j \) of node \( j \) through route \( r \). Let \( Q^t_{rP} \) be the corresponding route order quantity. (Refer to the definition of \( \bar{\theta}^k_r \).) Then define \( Q'_rP \) and \( I'_rP \) as follows:

\[
Q'_rP \triangleq Q^t_{rP}, \quad I'_rP \triangleq I^t_{rP}, \quad \text{if } t = \theta'_r, \quad \text{for } r \ni j \quad \text{(including } r = j). 
\]

Similarly, let the latest ordering instant \( \theta''_j \) be the next latest ordering instant at node \( j \) after \( \theta'_j \). Define \( Q''_jP \) and \( I''_rP \) as follows:

\[
Q''_jP \triangleq Q^t_{rP}, \quad I''_rP \triangleq I^t_{rP}, \quad \text{if } t = \theta''_r, \quad \text{for } r \ni j \quad \text{(including } r = j). 
\]

**Proposition 2.4.10 (Finite Horizon Nonnegative Filling Property)**

For any finite-cost, feasible replenishment policy \( P = (t_P, Q_P) \) for an inventory model \( \text{IM}(G, MT, CT, IF, GB) \) over a finite interval \([0, T)\), there exists a replenishment policy \( P' = (t_P, Q'_P) \) which satisfies the Nonnegative Filling Property, and dominates policy \( P \) over interval \([0, T)\).

**Proof.** Based on Proposition 2.4.2 and Proposition 2.4.4, it may be assumed that replenishment policy \( P \) satisfies the Nonpositive Inventory Ordering and the Latest Ordering Properties.

If node \( j \in D_N \) violates the Nonnegative Filling Property at ordering instant \( t \), we call node–instant point \((j, t)\) a bad point. Let \((j, \theta'_j)\) be a bad point under policy \( P \). Then \( I_{jP'}^+ < 0 \) and \( Q_{jP'}^+ > 0 \). Let \( \theta''_j \) be the next latest ordering instant of node \( j \) after ordering instant \( \theta'_j \), that is, \( Q''_{jP} > 0 \), which is defined as the same as \( Q'_jP \). See Figure 2.8 on page 52.
Figure 2.8: Predecessor Modification Step \( \Delta_R(j, \theta''_r, \theta'_r, x) \)

\[ r = (f_r, s_r, j), \ r_1 = (s_r, j) \]
For each route $r \ni j$, let $\theta'_r$ and $\theta''_r$ be the latest ordering instants of node $f_r$ corresponding to latest ordering instants $\theta'_j$ and $\theta''_j$ through route $r$, respectively. Let $Q'_{rP}$ and $Q''_{rP}$ be the corresponding route order quantities.

Policy $P' = (t_P, Q_{P'})$ is defined by applying the following Predecessor Modification Step $\Delta_R(j, \theta''_r, \theta'_r, x)$ as follows:

$$Q^t_{rP'} = \begin{cases} Q'_{rP} + \lambda_r x, & \text{if } t = \theta'_r \\ Q''_{rP} - \lambda_r x, & \text{if } t = \theta''_r \\ Q^t_{rP}, & \text{otherwise} \end{cases} \quad \forall r \ni j \quad \text{(including } r = j),$$

and

$$Q^t_{iP'} = \sum_{r \ni i} Q^t_{rP'}, \quad \text{for all } i \in N.$$

By Proposition 2.4.7, we have $Q'_{rP} \geq \lambda_{f_{rs}} Q'_{r_{j_{f_r}}}$, and $Q''_{rP} \geq \lambda_{f_{rs}} Q''_{r_{j_{f_r}}}$. Therefore, policy $P'$ resulting from the application of Predecessor Modification Step $\Delta_R(j, \theta''_r, \theta'_r, x)$ is feasible for any $x \in [-Q'_{jP}, Q''_{jP}]$. However, if $x \leq |I_{jP}^+|$, the only backlogging cost to be changed is that at node $j$. Therefore, we require $x \in [-Q'_{jP}, \min \{|I_{jP}^+|, Q''_{jP}\}]$. Note that the holding cost of every node $i \in P(j)$ may be changed.

Recall that $b_j$ is the backorder cost rate of node $j$. The total change of holding and backlogging cost is $C_{HP'}(T) - C_{HP}(T) = \alpha \times x$, where

$$\alpha \triangleq \left\{-(\theta'_r - \theta'_j)b_j + \sum_{r \ni j, r \neq j} \left[ (\theta'_{r_{j_{f_r}}}, \theta'_r) - (\theta''_{r_{j_{f_r}}}, \theta''_r) \right] \lambda_r h'_r \right\}$$

is a constant independent of $x$ as long as $x \in [-Q'_{jP}, \min \{|I_{jP}^+|, Q''_{jP}\}]$.

Let

$$x^* = \begin{cases} -Q'_{j} & \text{if } \alpha > 0, \\
\min\{|I_{jP}^+|, Q''_{jP}\} & \text{if } \alpha \leq 0, \end{cases}$$

and let policy $P'$ result from policy $P$ by application of Predecessor Modification Step $\Delta_R(j, \theta''_r, \theta'_r, x^*)$. Thus $C_{HP'}(T) \leq C_{HP}(T)$. We claim that policy $P'$ has fewer bad points.
than policy $P$ in either case: If $\alpha > 0$, then there is no order for $j$ at instant $\theta_r$, i.e., $(j, \theta_r')$ is not a bad point. If $\alpha \leq 0$, we have two sub-cases to consider. First, if $|\Gamma_j^+| > Q_j''$, then $x^* = Q_j''$, and there is no order for $j$ at instant $\theta_r''$. Note that $(j, \theta_r'')$ must also be a bad point of policy $P$, but is not a bad point of policy $P'$. Second, if $|\Gamma_j^+| \leq Q_j''$, then $x^* = |\Gamma_j^+|$ and $(j, \theta_r')$ is no longer a bad point of policy $P'$.

Because of the monotonicity of the set-up cost function, the total set-up cost cannot increase, i.e., $C_{SP}(T) \leq C_{SP}(T)$. Therefore, $C_{P'}(T) \leq C_{P}(T)$.

In summary, policy $P'$ dominates policy $P$ and has fewer bad points over finite horizon $[0, T)$. By executing Predecessor Modification Step $\Delta_R(j, \theta_r'', \theta_r', x^*)$ a finite number of times, we construct a policy $P'$ without any bad point and dominating policy $P$.

The proof of Proposition 2.4.10 is completed by repeating this construct a finite number of times, until all bad points are eliminated.

**Corollary 2.4.11 (Infinite Horizon Nonnegative Filling Property)**

For inventory model $IM(G, MT, NE, IF, GB)$ there exists an optimal infinite horizon policy satisfying the Nonnegative Filling Property.

**Proof.** By applying Corollary 2.3.3 and the previous Proposition, it is easy to verify this Corollary.

**Node Nestedness Property:** A replenishment policy $P$ is said to satisfy the Node Nested Property, if every node $i \in N$ orders only at instants which coincide with an ordering instant of one of its successors.

**Every Node Nestedness Property:** A replenishment policy $P$ is said to satisfy the Every Node Nested Property, if every node $i \in N$ orders only at instants which
coincide with an ordering instant of every of its successors. If policy \( P \) satisfies Every Node Nestedness Property, we may say that \( P \) is a nested policy.


Theorem 2.4.12 (Node Nestedness Property for Separable Setup Function)

For inventory model \( IM(G,SP,CT,IF,GB) \) there exists an optimal policy satisfies Node Nested Property.

On pp.28–30 of his thesis [39] (1987), Zheng showed that Node Nestedness Property does not hold for inventory model with non-separable setup cost by an inventory model of only three nodes.

If an optimal policy could be a nested policy, it is easier to deal with such as in Chapter 4. Therefore, it is interested to known for which inventory model there is an optimal nested policy. Lemma 2.4.13 below specifies a sufficient condition for it.

Lemma 2.4.13 (Optimal Nested Policy for Assembly Systems)

For inventory model \( IM(A,SP,CT,IF,GB) \) there exists an optimal nested policy.

\[ \text{Proof.} \quad \text{This follows from the fact that there is only one successor and Theorem 2.4.12.} \]
2.5 Conclusions

In this chapter, we prove the existence of an optimal policy for a large class of inventory models. This had been an open question for a long time.

We also present properties of optimal policies in this chapter. For simplicity, we only consider inventory models with infinite production rate. In fact, these properties may be extended to inventory models with finite production rates. As an example, we present some of the properties for the assembly systems in Chapter 3.

These properties of optimal policies are useful in probing optimal policies per se. They are also useful for special classes of policies, such as power-of-two policies, under which we may further simplify the average cost formulation. For example, in Chapter 4, we consider a series inventory model with backlogging, for which an optimal policy is nested. If we consider integer frequency policies, the expression of holding and backlogging cost is further simplified. Based on this formulation, we derive heuristic and a lower bound. These developments rely in part on properties of optimal policies.
2.6 Appendix to Chapter 2

2.6.1 An Inventory Problem That Has no Optimal Policy

The following example illustrates that an inventory problem could have no optimal policy.

**Example.** A two-product joint replenishment inventory problem over finite horizon \([0, 2)\) does not allow backlogging. Suppose its initial inventory levels \(I_i(0) = 0\), holding cost rates \(h_i = 1\), demand rates \(d_i = 2\), \((i = 1, 2)\) and the joint setup cost function is

\[
K(S) = \begin{cases} 
1, & \text{if } |S| = 1, \\
3, & \text{if } |S| = 2.
\end{cases}
\]

Because no backlogging is allowed, any feasible policy has to replenish both products 1 and 2 at instant 0, therefore incurring an initial setup cost of 3. Taking this into account and using the setup cost allocation of Atkins–Iyogum [3] (1987), it can be shown that a lower bound on the total cost is 9. This lower bound could only be achieved if both product were jointly replenished at instants 0 and 1, but the joint replenishment at instant 1 increases the total cost to 10. Consider the following sequence of policies depending on parameter \(\epsilon\) satisfying \(0 < \epsilon < 1\):

Product 1 is replenished at instants 0 and 1, and product 2 is replenished at instants 0 and \(1 - \epsilon\) (see Figure 2.9). The total setup cost over \([0, 2)\) is \(3 + 2 = 5\), the total holding cost over \([0, 2)\) is \(2 + (1 - \epsilon)^2 + (1 + \epsilon)^2 = 4 + 2\epsilon^2\), and the total cost over \([0, 2)\) is \(C_\epsilon = 9 + 2\epsilon^2\).

Therefore, \(\lim_{\epsilon \to 0^+} C_\epsilon = 9\), but no policy can achieve a total cost of 9. \(\Box\)
2.6.2 Some misunderstanding of the properties of optimal policies

Various properties of optimal inventory policies have been investigated since the late 1950's. The classic Wagner–Whitin [38] (1958) paper was one of the first works discussing such properties. But very few new properties have been discovered since the early 1960's. Schwarz [33] (1973) and Zheng [39] (1987) summarized the known properties. These properties seem simple and easy to accept, but they may also be very easily misunderstood.

For example, for 1–warehouse $n$–retailer systems, nested policies (i.e., such that when the warehouse orders, each retailer also has to order) have long been implicitly assumed to be near optimal. Unfortunately, nested policies may be arbitrarily bad, as Roundy [27] (1983) pointed out.

Another example is in Mitchell's [18] (1987) paper. He considered 1–warehouse $n$–retailer systems with backlogging. He misinterpreted the last minute ordering property as follows: “By last minute ordering (see Schwarz 1973) we know that retailer $i$ will also
order at those instants when the warehouse orders products \( i \). In fact, this interpretation is wrong. It is possible that the warehouse orders product \( i \) at an instant that the retailer \( i \) does not. See the example in Figure 2.10 on page 60, where warehouse 3 orders product 2 at the instant that retailer 2 does not order. This misinterpretation, however, does not invalidate his Lower Bound Theorem, but the latter needs a correction. This correction uses the latest ordering property demonstrated in this chapter. (1) Because of the Observation 1 of Proposition 2.4.7, the number of orders at warehouse for product \( i \) is no greater than the number of orders at retailer \( i \). Therefore, the inequality in Mitchell's [18] (1987) still holds. (2) It requires to consider an additional simple case where the warehouse orders product \( i \) at an instant when the retailer \( i \) does not: in Figure 2.10, we know that the ordering instants of retailer 2 do not coincide with the warehouse 3. As each retailer order will be satisfied by one warehouse order, the warehouse may delay this ordering instant to match the corresponding ordering instant of the retailer. This will reduce the holding and backlogging cost. As the setup cost function is separable, we do not change the setup costs of warehouse and the retailers. This modification will generate a lower bound for the policy, although it is not necessarily a feasible policy.
Figure 2.10: The order instants at warehouse do not coincide with the order instants at retailers.

\[ r_1 = 1, r_2 = 2, r_3 = (3, 1), r_4 = (3, 2) \]
2.6.3 A proof of lemma 2.3.4 by using a “remove” Algorithm

Lemma 2.3.4

For the inventory models IM(G,MT,IN,IF,GB) and IM(G,MT,IN,FF,NB) the Nonpositive Ending Inventory Assumption holds.

Proof. We only show that the result holds for the inventory model IM(G,MT,IN,IF,GB). The proof for the other model is similar. The following algorithm “Remove(I,i,t)” in Figure 2.11 removes the “extra” inventory amount $I$ at node $i$ and instant $t$, as well as all the inventories of $i$’s predecessors which supply the “extra” inventory $I$. The proof is completed by calling algorithm “Remove($I,i,T$)” for each node $i \in N$ at ending instant $T$.

Figure 2.11: Algorithm Remove ($I,i,t$)

Input: $I$, $i$, $t$, $Q$;
Output: $Q$.

begin
if $I > 0$ then
begin
/* find the instant $u$ of the order which supplies inventory $I$. */
$u \leftarrow \max \{v \leq t|Q^u_i > 0,\}$;
if $Q^u_i \geq I$ then
begin
/* reduce the order quantity at node $i$ and instant $u$ by amount $I$. */
$Q^u_i \leftarrow Q^u_i - I; \ I \leftarrow 0;$
/* increase the inventory at node $j \in p(i)$. */
for all $j \in p(i)$ do Remove($\lambda_{ji}I,j,u$);
end
else /* $Q^u_i < I$ */
begin
for all $j \in p(i)$ do Remove($\lambda_{ji}Q^u_i,j,u$);
$I \leftarrow I - Q^u_i; \ Q^u_i \leftarrow 0;$
/* extra inventory $I$ is not reduced to zero, */
/* call the remove algorithm again */
Remove($I,i,t$);
end
end
end
2.6.4 Calculating $H^i$: An Example

Figure 2.12: The production network of six products

![Production network diagram]

Table 2.3: Definition of routes

<table>
<thead>
<tr>
<th>$r_1$</th>
<th>$r_2$</th>
<th>$r_3$</th>
<th>$r_4$</th>
<th>$r_5$</th>
<th>$r_6$</th>
<th>$r_7$</th>
<th>$r_8$</th>
<th>$r_9$</th>
<th>$r_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>(3,1)</td>
<td>(3,2)</td>
<td>(4,2)</td>
<td>(5,3,1)</td>
<td>(5,3,2)</td>
<td>(6,3,1)</td>
<td>(6,3,2)</td>
<td>(6,4,2)</td>
</tr>
</tbody>
</table>

Table 2.4: Holding costs $h_i$ and $h_{fr}$

<table>
<thead>
<tr>
<th>$h_{fr_1}$</th>
<th>$h_{fr_2}$</th>
<th>$h_{fr_3}$</th>
<th>$h_{fr_4}$</th>
<th>$h_{fr_5}$</th>
<th>$h_{fr_6}$</th>
<th>$h_{fr_7}$</th>
<th>$h_{fr_8}$</th>
<th>$h_{fr_9}$</th>
<th>$h_{fr_{10}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_1$</td>
<td>$h_2$</td>
<td>$h_3$</td>
<td>$h_3$</td>
<td>$h_4$</td>
<td>$h_5$</td>
<td>$h_5$</td>
<td>$h_6$</td>
<td>$h_6$</td>
<td>$h_6$</td>
</tr>
</tbody>
</table>
Table 2.5: $\lambda_r$ and $\lambda_{ij}$

<table>
<thead>
<tr>
<th>$\lambda_{r1}$</th>
<th>$\lambda_{r2}$</th>
<th>$\lambda_{r3}$</th>
<th>$\lambda_{r4}$</th>
<th>$\lambda_{r5}$</th>
<th>$\lambda_{r6}$</th>
<th>$\lambda_{r7}$</th>
<th>$\lambda_{r8}$</th>
<th>$\lambda_{r9}$</th>
<th>$\lambda_{r10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>$\lambda_{31}$</td>
<td>$\lambda_{32}$</td>
<td>$\lambda_{42}$</td>
<td>$\lambda_{53}\lambda_{31}$</td>
<td>$\lambda_{53}\lambda_{32}$</td>
<td>$\lambda_{63}\lambda_{31}$</td>
<td>$\lambda_{63}\lambda_{32}$</td>
<td>$\lambda_{64}\lambda_{42}$</td>
</tr>
</tbody>
</table>

Table 2.6: Echelon inventory $E_r$ and actual route inventory $I_r$

<table>
<thead>
<tr>
<th>$E_{r1}$</th>
<th>$E_{r2}$</th>
<th>$E_{r3}$</th>
<th>$E_{r4}$</th>
<th>$E_{r5}$</th>
<th>$E_{r6}$</th>
<th>$E_{r7}$</th>
<th>$E_{r8}$</th>
<th>$E_{r9}$</th>
<th>$E_{r10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{r1}$</td>
<td>$I_{r2}$</td>
<td>$I_{r3} + \lambda_{31}I_{r1}$</td>
<td>$I_{r4} + \lambda_{32}I_{r3}$</td>
<td>$I_{r5} + \lambda_{42}I_{r2}$</td>
<td>$I_{r6} + \lambda_{53}\lambda_{31}I_{r1}$</td>
<td>$I_{r7} + \lambda_{53}\lambda_{32}I_{r2}$</td>
<td>$I_{r8} + \lambda_{63}\lambda_{31}I_{r1}$</td>
<td>$I_{r9} + \lambda_{63}\lambda_{32}I_{r2}$</td>
<td>$I_{r10} + \lambda_{64}\lambda_{42}I_{r2}$</td>
</tr>
</tbody>
</table>

Table 2.7: Node inventory $I_i$ and actual route inventory $I_r$

<table>
<thead>
<tr>
<th>$I_1$</th>
<th>$I_2$</th>
<th>$I_3$</th>
<th>$I_4$</th>
<th>$I_5$</th>
<th>$I_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{r1}$</td>
<td>$I_{r2}$</td>
<td>$I_{r3} + I_{r4}$</td>
<td>$I_{r5}$</td>
<td>$I_{r6} + I_{r7}$</td>
<td>$I_{r8} + I_{r9} + I_{r10}$</td>
</tr>
</tbody>
</table>

Table 2.8: Holding cost rate $h'_i$ and echelon holding cost rate $h_j$

<table>
<thead>
<tr>
<th>$h'_1$</th>
<th>$h'_2$</th>
<th>$h'_3$</th>
<th>$h'_4$</th>
<th>$h'_5$</th>
<th>$h'_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_1 +$</td>
<td>$h_2 +$</td>
<td>$h_3 +$</td>
<td>$h_4 +$</td>
<td>$h_5$</td>
<td>$h_6$</td>
</tr>
<tr>
<td>$\lambda_{31}h_3 +$</td>
<td>$\lambda_{32}h_3 +$</td>
<td>$\lambda_{42}h_4 +$</td>
<td>$\lambda_{53}\lambda_{31}h_5 +$</td>
<td>$\lambda_{53}\lambda_{32}h_5 +$</td>
<td>$\lambda_{63}\lambda_{31}h_6$</td>
</tr>
</tbody>
</table>
$H^t = \sum_{i=1}^{6} h'_i I_i$

$= (h_1 + \lambda_3 h_3 + \lambda_5 \lambda_3 h_5 + \lambda_6 \lambda_3 h_6) I_{r_1}$

$+ (h_2 + \lambda_3 h_3 + \lambda_4 h_4 + \lambda_5 \lambda_3 h_5 + \lambda_6 \lambda_3 h_6 + \lambda_6 \lambda_4 h_6) I_{r_2}$

$+ (h_3 + \lambda_5 h_5 + \lambda_6 h_6)(I_{r_3} + I_{r_4}) + (h_4 + \lambda_6 h_6) I_{r_5}$

$+ h_5(I_{r_6} + I_{r_7}) + h_6(I_{r_8} + I_{r_9} + I_{r_{10}})$

$= h_1 I_{r_1} + h_2 I_{r_2} + (\lambda_3 I_{r_1} + I_{r_3}) h_3 + (\lambda_3 I_{r_2} + I_{r_4}) h_3 + (\lambda_4 I_{r_2} + I_{r_5}) h_4$

$+ (I_{r_6} + \lambda_5 I_{r_3} + \lambda_5 \lambda_3 I_{r_1}) h_5 + (I_{r_7} + \lambda_5 I_{r_4} + \lambda_5 \lambda_3 I_{r_2}) h_5$

$+ (I_{r_8} + \lambda_6 I_{r_3} + \lambda_6 \lambda_3 I_{r_1}) h_6 + (I_{r_9} + \lambda_6 I_{r_4} + \lambda_6 \lambda_3 I_{r_2}) h_6$

$+ (I_{r_{10}} + \lambda_6 I_{r_5} + \lambda_6 \lambda_4 I_{r_2}) h_6$

$= \sum_{j=1}^{10} h_{f_{r_j}} E_{r_j}$

### 2.6.5 Notation and Definitions

In this section, all the notation and definitions are collected for reference.

**A** – Assembly Systems (a special case of G): each node has exactly one successor in the network $G(N, A)$, except for one node, which has no successor and is called the root node.

**B** – Always backlogging (a special case of GB): Backlogging is allowed for every external demand.

$b'_i$ is the backlogging cost rate of product $i$ at instant $t$: it is the cost of backlogging one unit of product $i$ for one unit of time at instant $t$.

$C_H(T) \triangleq \int_0^T H^t dt$ is the total holding and backlogging cost accrued during the time interval $[0, T]$.

**CT** – Constant holding and backlogging cost rates (a special case of NN): $h'_i = h'_i \geq 0$ and $b'_i = b_i \geq 0$, which are independent of time $t$ for each product $i$. 
Chapter 2. Optimal Inventory Policies

**D** – Distribution Systems (a special case of G): each node has exactly one predecessor in the network $G(N, A)$, except for one node, which has no predecessor.

$$d_i \triangleq \sum_{r \in R_i} d_r$$ is the induced (node) demand rate of node $i$.

$D_N \triangleq \{i \in N | s(i) = \emptyset\}$ is the set of all nodes without successor. Without loss of generality, we may assume that $D_N = \{i \in N | \text{node } i \text{ has external demand}\}$ — the external demand set of $G(N, A)$. Otherwise, if node $i$ in an inventory model has both “external demand” $d_i$ and “internal demand” induced by its successors, we may construct a new inventory model from the original inventory model by adding a pseudo node $i_e$, arc $(i, i_e)$, moving the external demand $d_i$ from node $i$ to node $i_e$ and letting setup cost $K_{i_e} = 0$, $K(S \cup i_e) = K(S)$, actual holding cost rate $h_{i_e}' = h_i'$, backlogging cost rate $b_{i_e}' = b_i$ and not allowing backlogging at node $i$. It is easy to establish a correspondence between feasible policies for these two inventory systems, preserving total cost. For example, given any feasible policy for the original inventory model, we may construct a feasible policy for the new inventory model by immediately delivering all the inventories at node $i$ for external demand $d_i$ to node $i_e$ (see the example in Figure 2.13 on page 66).

It is easy to check that the total cost of the policy for the new inventory model is exactly the same as the original one. For the reverse, given any feasible policy for the new inventory model, construct a policy for the original inventory system by merging the inventory at node $i$ and $i_e$ for the new inventory model. This gives a feasible policy for the original inventory model, which has the same total cost as the new inventory model unless the following situation arises: node $i$ has positive inventory and node $i_e$ has negative inventory (backlogging). However, such a policy is dominated in the new inventory model, because shifting (pushing) the product
Figure 2.13: The correspondence between the two inventory systems.
from $i$ to $i_e$ incurs no additional setup cost and has the same holding cost, but eliminates the backlogging cost.

**Dominate:** Policy $P'$ dominates policy $P$ over interval $[0,T)$, if $C_{P'}(t) \leq C_P(t), \ \forall t \in [0,T)$.

$d_r \triangleq \lambda_r d_{t_r}$ is the induced route demand rate at node $f_r$ from route $r$.

**Every Node Nestedness Property:** A replenishment policy $P$ is said to satisfy the Every Node Nested Property, if every node $i \in N$ orders only at instants which coincide with an ordering instant of every of its successors. If policy $P$ satisfies Every Node Nestedness Property, we may say that $P$ is a nested policy.

$E^t_r$ is the echelon inventory on route $r$ at instant $t$. It is the total inventory of good $f_r$ on route $r$, defined as follows:

$$E^t_r \triangleq \sum_{r' \subseteq r} \left( \frac{\lambda_r}{\lambda_{r'}} \right) I^t_{r'}$$

where routes $r'$ are subroutes of $r$ having the same last node.

$$E_{r^-}^t = \lim_{t' \searrow t} E_{r'}^{t'}.$$  

$$E_{r^+}^t = \lim_{t' \nearrow t} E_{r'}^{t'}.$$  

**F** – Finite production rate: the production rate $\pi_i$ is finite for each node in $N$.

**A feasible policy** $P$ is a policy which satisfies all the requirements of the model under consideration. These requirements include satisfying external demands for specified products, without backlogging if it is prohibited.

**FF** – Fast finite production rate (a special case of F): $\pi_i \geq \sum_{j \in \sigma(i)} \lambda_{ij} \pi_j$, for all $i \in N$.  

Chapter 2. Optimal Inventory Policies

\[ f_r \triangleq i_1 \] is the first node of \( r \).

\( G \) – General acyclic network: there is no directed cycle in network \( G(N, A) \).

\( GB \) – General Backlogging: Backlogging may be allowed for some or all external demands, but is not allowed for any internal (induced) demand.

\( G(N, A) \) is an acyclic direct graph, where \( N \triangleq \{1, 2, \cdots, n\} \) is the node set, and \( A \) is the arc set. The nodes are indexed such that \( i > j \) if \((i, j) \in A\).

\[ h_i^t = h_i'^t - \sum_{j \in p(i)} \lambda_{ij}h_j'^t \] is the echelon holding cost rate of node \( i \) at instant \( t \): it is the “added” holding cost rate for product \( i \).

\( h_i'^t \) is the holding cost rate of product \( i \) at instant \( t \): it is the cost of holding one unit of product \( i \) for one unit of time at instant \( t \).

\[ h_i' \triangleq \text{the actual inventory holding cost rate at node } i. \]

\[ H_i^t \triangleq \sum_{i=1}^{n}(T_i h_i'^t + I_i b_i'^t) \] is the total holding and backlogging cost rate at instant \( t \). which is the total holding and backlogging cost in one unit of time at instant \( t \).

**Holding and Backlogging Costs** are inventory related cost, charged in terms of the units of products stored and backlogged and the length of time it is stored and backlogged, respectively. Let

\( I \) – Isolated nodes (a special case of \( G \)): \( A = \emptyset \).

\( i \in r \) means that node \( i \in \{i_1, i_2, \cdots, i_m\} \).

\( \{i_1, i_2, \cdots, i_m\} \) is the node set of path \( r \).

**IF** – Infinite production rate: the production rate \( \pi_i \) of each node is infinite.
Chapter 2. Optimal Inventory Policies

$I_i^t$ is the actual (node) inventory level of product $i$ at instant $t$.

$I_i^- = \lim_{t' \uparrow t} I_i^{t'}$ is the inventory level just before instant $t$.

$I_i^+ = \lim_{t' \downarrow t} I_i^{t'}$ is the inventory level just after instant $t$.

$I_i^t \triangleq \max(-I_i^t, 0)$ is the backlogged inventory level of product $i$ at instant $t$.

$I_i^t \triangleq \max(I_i^t, 0)$ is the physical inventory level of product $i$ at instant $t$.

$I_r^t$ is the actual route inventory at node $f$, for route $r$ at instant $t$. By definition, $I_r^t = \sum_{r \in R_i} I_r^t$ for all $t$. Similarly,

$I_r^- = \lim_{t' \uparrow t} I_r^{t'}$.

$I_r^+ = \lim_{t' \downarrow t} I_r^{t'}$.

IM(network, setup cost, holding/backlogging cost, production rate, backlogging) is the inventory model.

An infinite/finite horizon policy: If the planning horizon is infinite (resp. finite), the policy $P$ is called an infinite (resp. a finite) horizon policy. If $P$ is an infinite horizon policy, the restriction of $P$ to a finite horizon $[0, T]$ is a finite horizon policy, which is also called policy $P$ for simplicity.

$K^t : 2^N \mapsto R^+$ is the setup (or joint replenishment) cost function. It is assumed that $K^t(\emptyset) = 0$, and $K^t(S) \geq 0$, $\forall S \neq \emptyset, S \subseteq N$. Let $k_i$ be a real number associated with each node $i \in N$.

Latest Ordering Property: A replenishment policy $P = (t, Q)$ satisfies the Latest Ordering Property, if the following holds for every node $i$ and every ordering instant
Chapter 2. Optimal Inventory Policies

$t = t_i^k$: the order quantity $Q_i^t$ is positive iff there exists a positive order quantity $Q_j^u > 0$ of some node $j \in s(i)$ at some ordering instant $u \in [t_i^k, t_i^{k+1})$.

The long-run average cost $\overline{C}_P$ of an infinite horizon policy $P$ is defined by

$$\overline{C}_P \triangleq \limsup_{T \to \infty} \frac{1}{T} C_P(T).$$

$\ell_r \triangleq i_m$ is the last node of $r$.

**MD** – Modularity or First Order Interaction (a special case of SM): $K(\emptyset) = 0$, $K(S) = K_0 + \sum_{i \in S} k_i$, $\forall S \neq \emptyset, S \subseteq N$, where $K_0 \geq 0$ is a fixed constant.

**MT** – Monotonicity: $K(S) \leq K(T)$, $\forall S \subseteq T \subseteq N$.

**NB** – No backlogging (a special case of GB): There is no backlogging at any place.

**NE** – Nonnegative Echelon holding cost rates (a special case of CT): $h_i \geq 0$ and constant.

**NN** – Nonnegative Holding and backlogging cost rates: $h_i^t \geq 0$ and $b_i^t \geq 0$ for each $i \in N$. Thus holding and backlogging costs are increasing functions of the number of units stored and backlogged in the holding and backlogging duration.

**Node–instant point** $(i, t)$ is a pair of node $i$ and ordering instant $t$. Over any finite horizon $[0, T)$, if $i < j$ or, if $i = j$ and $t < u$, we say that node–instant point $(i, t)$ is less than node–instant point $(j, u)$, written as $(i, t) < (j, u)$. (This is the usual lexicographic order on $N \times [0, T)$.)

**Node Nestedness Property**: A replenishment policy $P$ is said to satisfy the Node Nested Property, if every node $i \in N$ orders only at instants which coincide with an ordering instant of one of its successors.
Nonnegative Filling Property: Given replenishment policy $P$ for inventory model $IM(G,\cdot,\cdot,IF,GB)$, if order quantity $Q_i^t$ at node $i \in D_N$ and ordering instant $t$ reduces the actual route inventory to a non-negative level (i.e., $I_i^{t^+} \geq 0$), then the Nonnegative Filling Property holds for node $i$ at ordering instant $t$. If the Nonnegative Filling Property holds at every node in $D_N$ and every ordering instant, then replenishment policy $P$ satisfies the Nonnegative Filling Property.

Nonpositive Ending Inventory Assumption: for any feasible policy $P$ over finite horizon $[0,T)$ there exists a Nonpositive Ending Inventory policy $P'$, which dominates policy $P$. When backlogging is not allowed, the Nonpositive Ending Inventory Assumption reduces to the Zero Ending Inventory Assumption.

Nonpositive Ending Inventory Policy: A finite horizon feasible policy $P$ over $[0,T)$ is a Nonpositive Ending Inventory policy if the inventory level of each product is nonpositive at instant $T$. When backlogging is not allowed, a Nonpositive Ending Inventory policy is a Zero Ending Inventory policy.

Nonpositive Inventory Ordering Property: If a replenishment policy $P$ orders quantity $Q_i^t > 0$ at node $i$ and ordering instant $t$ with actual inventory level $I_i^{t^-} \leq 0$, it is said that policy $P$ satisfies the Nonpositive Inventory Ordering Property for node $i$ at instant $t$.

O – One-warehouse Multi-retailer Systems (a special case of D): one node, termed the warehouse, has successors but no predecessor in network $G(N,A)$, and all other nodes, the retailers, have no successor and exactly one predecessor, the warehouse.

The optimal average cost $\bar{C}^*$ over all feasible policies is defined by

$$\bar{C}^* \overset{\Delta}{=} \inf_{\text{all feasible } P} \bar{C}_P.$$

(2.19)
Path $r = (i_1, i_2, i_3, \ldots, i_{m-1}, i_m)$ is a sequence of nodes $i_1, i_2, \ldots, i_m \in N$ such that all arcs $(i_1, i_2), (i_2, i_3), \ldots, (i_{m-1}, i_m) \in A$.

$\pi_i$ is the production rate at node $i$, i.e., the number of units that can be produced at node $i$ per time unit.

$p(i) \triangleq$ the set of immediate predecessors of node $i$.

$P(i) \triangleq$ the set of all predecessors of node $i$.

$R \triangleq \bigcup_{i \in N} R_i$ is the set of all routes $r$ in $G(N, A)$.

$|r| \triangleq m$ is the length of path $r$.

A replenishment policy $P \triangleq (t_P, Q_P)$ is a specification of ordering instants $t_P$ and order quantities $Q_P$ for all products over a planning horizon. That is,

$$t_P = (t_{1P}, t_{2P}, \ldots, t_{nP}),$$

$$Q_P = (Q_{1P}, Q_{2P}, \ldots, Q_{nP}),$$

$$t_{iP} = (t_{1iP}, t_{2iP}, \ldots, t_{kiP}, \ldots),$$

$$0 \leq t_{1iP} < t_{2iP} < \ldots < t_{kiP} < \ldots$$

$$Q_{iP} = (Q_{1iP}, Q_{2iP}, \ldots, Q_{kiP}, \ldots) \geq 0,$$

where $n$ is the number of products, quantity $Q_{kiP}$ is the amount of product $i$ ordered at instant $t_{kiP}$. Sometimes we use $Q_{iP}$ to represent the amount ordered for product $i$ at instant $t$, and if $t = t_{kiP}$ the quantity $Q_{kiP}$ may be replaced by $Q_{kiP}$, whenever it is convenient. Note that for the finite product rate case the ordering instant $t$ is the production starting time. When replenishment policy $P$ is not emphasized, subscript $P$ in $t$ and $Q$ may be dropped without risk of confusion.

Route $r = (i_1, i_2, i_3, \ldots, i_{m-1}, i_m)$ is a path terminating at a node $i_m \in D_N$. 
Chapter 2. Optimal Inventory Policies

A Route Order Quantity $Q_r^t$ is that part of order quantity $Q_i^t$ at node $i$ and ordering instant $t$ which is used to satisfy the (induced) demand from node $l_r$ along route $r$.

By definition, $Q_i^t = \sum_{r \in R_i} Q'_r$.

$r' = (i_j, \cdots, i_m)$ with $j \in \{1, \cdots, m\}$ is a sub-route of $r$, denoted by $r' \subseteq r$.

$r \setminus f_r \triangleq (i_2, \cdots, i_m)$ is the immediate sub-route of $r$.

$R_i$ is the set of all routes $r$ starting at node $i \in N$, i.e., such that $f_r = i$.

S - Series Systems (the intersection of A and D): each node has exactly one successor and one predecessor in the network $G(N, A)$, except for two nodes, one with only a successor and the other one with only a predecessor.

SA - Monotonicity and Subadditivity: $K(S \cup T) \leq K(S) + K(T)$, $\forall S, T \subseteq N$.

$s(i) \triangleq$ the set of immediate successors of node $i$.

$S(i) \triangleq$ the set of all successors of node $i$.

SM - Monotonicity and Submodularity: $K(S \cup T) + K(S \cap T) \leq K(S) + K(T)$, $\forall S, T \subseteq N$.

SP - Separability (a special case of MD): $K(S) = \sum_{i \in S} k_i$, $\forall S \subseteq N$.

$s_r \triangleq i_2$ is the second node of $r$.

The total cost $C_P(T)$ of policy $P$ over finite horizon $[0, T)$ is the sum of total setup or ordering cost $C_{SP}(T)$ and total holding and backlogging cost $C_{HF}(T)$ over horizon $[0, T)$, that is, $C_P(T) = C_{SP}(T) + C_{HF}(T)$.

Zero Initial Inventory Assumption: The initial inventory of each product is zero and all the products are replenished at the initial instant 0.
Chapter 2. Optimal Inventory Policies

Zero Inventory Ordering Property: Policy $P$ orders positive quantities at node $i$ only at instants $t$ where $I^+_i = 0$.

$\lambda_{ij}$ is the number of units of product $i$ required to produce 1 unit of product $j$, for $(i,j) \in A$.

$\lambda_r \triangleq \begin{cases} 1 & \text{if } |r| = 1, \\ \lambda_{i_1 i_2} \cdots \lambda_{i_{m-1} i_m}, & \text{if } |r| \geq 2 \end{cases}$ is the number of units of product $f_r$ required to produce 1 unit of product $\ell_r$ through path $r$.

$\tau_r(t) \triangleq \begin{cases} \theta_r(t), & \text{if } |r| = 1, \\ \tau_{r \setminus f_r}(\theta_r(t)), & \text{if } |r| \geq 2 \end{cases}$ for any $t > 0$, is the last node earliest ordering instant for instant $t$, which is the earliest ordering instant of the last node $\ell_r$ of route $r$ after instant $t$.

$\Theta_r \triangleq \{ \bar{\theta}_r^k | k = 1, 2, \cdots \}$ is the latest ordering instant set of node $f_r$ for route $r$. Because some of the instants $\bar{\theta}_r^k$ may coincide, we let $\Theta_r = \{ \theta_r^\ell | \ell = 1, 2, \cdots \}$ where $\theta_r^1 < \theta_r^2 < \cdots < \theta_r^\ell < \cdots$ are all the distinct values of the elements in $\Theta_r$.

$\theta_r(t) \triangleq \min \{ \theta_r^k | \theta_r^k \geq t, \ k = 1, 2, \cdots \}$ for any $t > 0$, is the first node earliest ordering instant after instant $t$, i.e., the earliest ordering instant of first node $f_r$ of route $r$ after instant $t$.

$\bar{\theta}_r^k \triangleq \begin{cases} t_{f_r}^k, & \text{if } |r| = 1 \\ \max \{ t_{f_r}^\ell, t_{f_r} \leq \bar{\theta}_r^k, \ell = 1, 2, \cdots \}, & \text{if } |r| \geq 2 \end{cases}$ for each $k = 1, 2, \cdots$ is the latest ordering instant at node $f_r$, which can be used to satisfy the demand from node $\ell_r$ at instant $t_{f_r}^k$ through route $r$. 
Chapter 3

Lot Sizing Policies for Finite Production Rate Assembly Systems

3.1 Introduction

This chapter considers lot sizing policies in the inventory model IM(A,SP,NE,FF,NB), i.e., a production/inventory assembly system with finite production rates and satisfying the following assumptions:

1. External demand for the single final product is known and occurs at a constant rate in continuous time.

2. Production rates at each facility are finite, non-increasing along any path in the assembly system and greater than the external demand. The importance of the latter assumption is that, in an optimal policy, there is no need for a facility to build extra inventory to meet future withdrawals from its successor.

3. Proportional echelon holding costs are incurred at each facility, and no back-orders are permitted. A fixed set-up cost is charged at each facility when production starts. There are no cost savings on joint production at different facilities.

4. Lot transfer times are zero throughout the system, that is, the finished product of any facility is immediately available to its successor. (Szendrovits [36] (1975) considers a finite production rate series system with non-zero lot transfer times, but with identical lot sizes throughout the system).

5. We are considering policies which minimize long-run average costs.
We elaborate on these assumptions by comparing them to previous work. Most work to date has concentrated on the case when production is instantaneous, i.e., infinite production rates. Roundy [27] (1983), [30] (1986) demonstrates that with infinite production rate, a relatively simple heuristic delivers a high quality (effective) solution to the multi-stage production/inventory lot sizing problem. The heuristic restricts replenishment time intervals for all components to a power-of-two multiple of a base interval. Roundy proves that there exists a power-of-two policy which is guaranteed to be 94%-effective. Put another way, the cost of power-of-two policies is within about 6% of the unknown optimum. By optimizing the base interval, this can be improved to within about 2%. Several generalizations, such as to more general cost structures and acyclic networks (e.g., [24] (1985), [30] (1986), [39] (1987)), extend the class of problems for which such guaranteed effectiveness results hold.

The literature on finite production rates is less comprehensive. The main differences between finite production rate systems and infinite production rate systems arise in the following ways.

First, in infinite production rate systems, an optimal policy requires that facilities start producing at zero actual and echelon inventories. Echelon inventories, therefore, follow saw-tooth patterns, i.e., all the troughs touch the zero inventory line. However, in finite production rate systems, optimal policies do not require echelon inventories to be zero when production starts. Neither actual nor echelon inventories follow simple saw-tooth patterns, see Figure 3.17 on page 116.

Second, optimal policies for infinite production rate systems are nested. Nested means that each production starting time of a facility has to coincide with a production starting time of its successor. If production rates are finite, optimal policies may be non-nested. Crowston, Wagner, and Henshaw [6] (1972), Schwarz and Schrage [34] (1975) (see also Szendrovits [37] (1981)), studied integer-multiple lot size policies, wherein the lot size at
each facility is an integer-multiple of its successor's lot size. Such policies are actually nested. Moily [20] (1986) investigated integer-split lot size policies, wherein the lot size at each facility is an integer-split of its successor's lot size. This is, in a sense, the "reverse" of being nested. We will see that neither integer-multiple lot size policies nor integer-split lot size policies dominate each other. As shown in Section 3.9.1, both integer-multiple and integer-split lot size policies may be arbitrarily bad, i.e., their effectiveness is (asymptotically) zero. In fact, no heuristics for finite production rate assembly systems have, to our knowledge, been shown to have a nonzero effectiveness.

Third, it is not necessarily effective in finite production rate systems to use constant reorder time intervals, as it is done for the infinite production rate case. Actually, reorder intervals at a facility need not be the same, even for a same lot size. This is illustrated in Section 3.9.2. In addition, a formulation of the problem is simplified when couched in terms of lot sizes rather than reorder intervals. Although the production schedule will not be as simple as with constant reorder intervals, it can be readily identified using the scheduling algorithm described in Section 3.9.2.

If we let all the production rates go to infinity, we obtain infinite production rate assembly systems as special cases. Therefore the Lower Bound Theorem of this chapter holds for infinite production rate assembly systems. Yet, our proof is distinctly different from Roundy's [27] (1983).

The three major contributions in this chapter are as follows. First, for assembly systems with non-increasing finite production rates, a formulation of the lot-sizing problem is developed in terms of integer-ratio lot size policies, expressed in terms of lot sizes rather than reorder time intervals. This formulation unifies the integer-split policies formulation of Schwarz and Schrage [34] (1975) and the integer-multiple policies formulation of Moily [20] (1986).
Second, the solution to the continuous relaxation of this unified formulation provides a lower bound to the cost of any feasible policy. The derivation of this Lower Bound Theorem, a key result in this chapter, is novel and relies on the notion of path holding costs, a generalization of echelon holding costs. This proof is significantly different from earlier proofs of similar results, which rely on convex duality. The proposed proof technique may perhaps itself lead to further generalizations.

Third, an optimal power-of-two lot size policy is found by a $O(n^5)$ algorithm and its cost is within 6% or 2% of the optimal in the worst case, depending on whether the base lot-size is given or not. These results are the first such results, we believe, for the finite production rate case.

The structure of the chapter is as follows. Section 3.2 contains all the requisite notation for reference. Section 3.3 lists some properties of optimal policies. A basic formulation for integer-ratio lot size policies is introduced in Section 3.4. The Lower Bound Theorem is given in Section 3.5. Section 3.6 demonstrates the effectiveness of integer-ratio and power-of-two lot size policies. Section 3.7 presents an algorithm for obtaining 94% and 98% effective power-of-two lot size policies. Section 3.8 summarizes the main contribution of this chapter, discusses some of their implications, and suggests directions for further research. Section 3.9.1 illustrates that both integer-multiple and integer-split lot size policies can simultaneously be arbitrarily worse than integer-ratio lot size policies. Section 3.9.2 contains a scheduling algorithm for converting a policy described by lot sizes into a workable schedule.

3.2 Notation

The following definitions (some of which are repeated from Chapter 2) are collected here for convenience.
Chapter 3. Finite Production Rate

Figure 3.1: Example of an assembly system.

\[ G(N, A) \]

is a network with node set \( N = \{1, \ldots, n\} \), the facilities, and arc set \( A \). Product \( i \) is the output of facility \( i \). Arc \( (i, j) \in A \) means that facility \( j \) uses the output of facility \( i \) directly. The network is an **Assembly System**, if for each node \( i \neq 1 \), there is exactly one arc \( (i, j) \in A \). The corresponding graph is a directed tree, where each node (facility), except the final facility, is the tail of exactly one arc.

\( s(i) \) is the unique **immediate successor** of facility \( i \). We assume that \( i > s(i) \) for each \( i \), and \( s(1) \triangleq 0 \) and 1 is the final facility. \( S(i) \) is the set of all the successors of facility \( i \). Product \( i \) will be used by all facilities \( j \in S(i) \). Let \( \overline{S}(i) \triangleq S(i) \cup \{i\} \).

\( p(i) \) is the set of all **immediate predecessors** of facility \( i \). \( P(i) \) is the set of all the predecessors of facility \( i \). Let \( \overline{P}(i) \triangleq P(i) \cup \{i\} \).

\( \langle i, j \rangle \) is the unique (directed) **path** from \( i \) to \( j \in S(i) \) in \( G(N, A) \). \( R \) is the set of all the paths in the assembly system. Clearly, \( |R| \leq \binom{n}{2} \). For convenience, let \( k \in \langle i, j \rangle \) mean that node \( k \) is on the path \( \langle i, j \rangle \) (allowing \( k = i \) and \( k = j \)). A path \( \langle i, 1 \rangle \) finishing at facility 1 will be termed a **route**. The length \( |\langle i, j \rangle| \) of path \( \langle i, j \rangle \) is the number of nodes in path \( \langle i, j \rangle \), including \( i \) and \( j \). We have \( S(i) = \langle s(i), 1 \rangle \) and \( \overline{S}(i) = \langle i, 1 \rangle \) for all \( i \).
\(\pi_i\) is the production rate of facility \(i\), in the units of output of facility \(i\) per time unit. We are assuming that \(\pi_i \geq \pi_{s(i)}\). The demand rate at facility 1 is denoted \(\pi_0\). Without loss of generality, we can assume that one unit of product \(i\), the output of facility \(i\), is needed to produce one unit of product \(s(i)\).

\(K_i\) is the set-up cost of facility \(i\). It is incurred whenever facility \(i\) starts producing.

\(h'_i\) is the actual holding cost rate of product \(i\). The echelon holding cost rate \(h_i\) of product \(i\) reflects the "added" value at facility \(i\): \(h_i = h'_i - \sum_{j \in p(i)} h'_j\). It is assumed that \(h_i > 0\), for all \(i\).

\(h_{(u,v)} \triangleq (h_i| i \in (u,v))\) is the holding cost rate vector on path \((u,v)\). The holding cost rate vector on set \(N\) is \(h_N \triangleq (h_i| i \in N)\). The pseudo-holding cost rate of product \(i\) at facility \(j\) is \(H_{ij} \triangleq \pi_0 h_i/2 \left(1/\pi_{s(j)} - 1/\pi_j\right)\).

\(I^t_i\) is the actual inventory level of product \(i\) at facility \(i\) at instant \(t\). It is the cumulative output of facility \(i\) that has not yet been used by facility \(s(i)\).

\(E^t_{(u,v)}\) is the path echelon inventory of any product \(k \in \mathcal{P}(u)\) on path \((u,v)\) at instant \(t\). Indeed, \(E^t_{(u,v)} \triangleq \sum_{i \in (u,v)} I^t_i\), independent of product \(k\). For an integer-ratio lot size policy, the average path echelon inventory (of product \(k \in \mathcal{P}(u)\)) on path \((u,v)\) is \(E_{(u,v)} \triangleq \liminf_{T \to \infty} \frac{1}{T} \int_0^T E^t_{(u,v)} dt\).

\(H^t_{(u,v)}(h_{(u,v)}) \triangleq \sum_{i \in (u,v)} h_i E^t_{(i,v)}\) is the total holding cost on path \((u,v)\) at instant \(t\), using echelon holding cost rate \(h_{(u,v)}\). The total holding cost on graph \(G(N,A)\) at instant \(t\) using echelon holding cost rate \(h_N\) is \(H^t_G(h_N) \triangleq \sum_{j \in N} h_j E^t_{(j,i)}\).

\(N^*\) is the set of positive integers: \(N^* \triangleq \{1, 2, \ldots\}\).

\(Z\) is the set of integers: \(Z \triangleq \{0, \pm 1, \pm 2, \ldots\}\).
3.3 Properties of Finite Horizon Optimal Policies

First, observe that optimal policies exist for any finite horizon: there exists a policy with bounded total cost; any policy with cost not exceeding that one must contain a bounded number of setups; such policies form a compact set; and therefore an optimal policy exists.

As we mentioned before, we will focus on assembly systems with non-increasing production rates. Some elementary properties of optimal policies over a finite horizon are listed below.

Lemma 3.3.1 (Production Starting Time)
For an optimal policy, a facility $i$ will not start producing at instant $t$ unless facility $s(i)$ is (or starts) producing at instant $t$, and the actual inventory $I_{t}^i$ of facility $i$ is zero.

Proof. By contradiction, if facility $i$ starts producing with a non-zero inventory $I_{t}^i$, or during the idle time of facility $s(i)$, then facility $i$ can delay its starting time while still meeting the demand from facility $s(i)$. Because $\pi_i \geq \pi_{s(i)}$, no extra inventory has to build before facility $s(i)$ starts producing. The effect on the holding cost of this new policy is that the inventory at facility $i$ is reduced by some amount $\Delta I_{t}^i$ during a time interval $[t_1, t_2]$, and the inventory at each facility $j \in p(i)$ is increased by the same amount $\Delta I_{t}^j$. The total change in holding cost is

$$-h_i' \int_{t_1}^{t_2} \Delta I_{t}^i \, dt + \sum_{j \in p(i)} h_j' \int_{t_1}^{t_2} \Delta I_{t}^j \, dt = -h_i \int_{t_1}^{t_2} \Delta I_{t}^i \, dt < 0.$$

This contradicts the optimality of the policy.

Lemma 3.3.2 (Zero Final Inventory Property)
Any optimal finite horizon policy over the interval $[0, T)$ has zero final inventories:

$$I_{T}^i = 0, \quad \text{for all } i.$$
Chapter 3. Finite Production Rate

Proof. Similar to that of Lemma 3.3.1 with \( t_2 = T \).

If the horizon \( T > 0 \) is large enough, costs associated with non-zero initial inventories have a vanishing effect on long-run average costs and the zero final inventory property holds for any initial inventories. Therefore, zero initial inventories are assumed without loss of generality.

Corollary 3.3.3 (Proportion of Working Time)

Let \( W_i \) be the total working time of facility \( i \) over the interval \( [0, T) \). For any optimal finite horizon policy over the interval \( [0, T) \) with zero initial inventories, the proportions of working time satisfy

\[
\frac{W_i}{T} = \frac{\pi_0}{\pi_i}, \quad \text{for all } i \in N.
\]  

Proof. For optimal finite horizon policies, only the amount \( \pi_i W_i \) is needed to satisfy the total demand \( \pi_0 T \) over the interval \( [0, T) \).

3.4 Integer-Ratio Lot Size Policies

In this section, we define integer-ratio lot size and power-of-two lot size policies. Lemma 3.4.1 lists some basic properties of integer-ratio lot size policies. Then we derive a formulation of average route echelon inventories for integer-ratio lot size policies. This is achieved first for a special situation in Lemma 3.4.2, and then extended to the general case in Theorem 3.4.3. Finally, Lemma 3.4.4 gives a non-linear mixed integer programming formulation for minimizing the average cost of integer-ratio lot size policies. Recall the assumption that all initial inventories are zero.

An Integer-Ratio Lot Size Policy (IRLSP) is a policy which satisfies:
Figure 3.2: Path \( (u, v) \).

1. for each facility \( u \in N \), the lot size \( Q_u \) is stationary;
2. for each pair of facilities \( u \in N \) and \( v \in S(u) \), either \( Q_u/Q_v \) or \( Q_v/Q_u \in N^* \);
3. each facility \( u \in N \) starts producing at instant \( t \) iff its actual inventory \( I_u^t \) is zero and its immediate successor \( s(u) \) already produces or starts producing at instant \( t \);
4. all the facilities start producing at instant \( t = 0 \).

A **Power-of-Two Lot Size Policy (PoTLSP)** is an integer-ratio lot size policy with each lot size \( Q_i (i \in N) \) of the form \( Q_i = Q_0 2^{m_i} \) for some positive base lot size \( Q_0 \), where \( m_i \in \mathbb{Z} = \{0, \pm1, \pm2, \cdots \} \), the set of integers.

We list the following facts about integer-ratio lot size policies without proof.

**Lemma 3.4.1 (Production Runs on a Path)**

For any integer-ratio lot size policy, suppose \( Q_i = \max_{j \in (u,v)} Q_j \) is the maximum lot size on path \( (u, v) \),

then

1. Each time facility \( i \) starts producing, so does each facility \( j \in (u, v) \).
2. Each facility \( j \in (u, i) \) produces only when facility \( i \) does.

We first give a formul\( \) for average route echelon inventory in a special situation.
Figure 3.3: Cumulated Path Echelon Inventory on path \((u,v)\).

\[ E^t_{(u,v)} \]

Lemma 3.4.2 (Average Route Echelon Inventory – A Special Case)

For any integer-ratio lot size policy, if

\[ Q_u \geq Q_j, \quad \text{for all} \quad j \in \langle u, v \rangle, \quad \text{and} \]

\[ Q_u \leq Q_s(v), \quad \text{and} \]

then

\[ E_{(u,v)} = \frac{1}{2} \pi_0 \left( \frac{1}{\pi_s(v)} - \frac{1}{\pi_u} \right) Q_u, \]

where \( E_{(u,v)} \triangleq \liminf_{T \to \infty} \frac{1}{T} \int_0^T E^t_{(u,v)} \, dt \) is the average path echelon inventory on path \((u,v)\) for an integer-ratio lot size policy, and \( Q_0 = Q_{\max} \triangleq \max_{j \in N} Q_j \).

Note that if \( v = u \), then (3.2) always holds. And if \( v = 1 \), then (3.3) always holds.

Proof. For any integer-ratio lot size policy, (3.3) means \( Q_s(v)/Q_u = m \in N^* \), that is, facility \( u \) makes \( m \) lots over the time interval that facility \( s(v) \) is making one lot \( Q_s(v) \). Combining with condition (3.2), this means that the path echelon inventory \( E^t_{(u,v)} \) has a saw-tooth shape when each lot \( Q_u \) is producing, as shown in Figure 3.3.

From this figure, it is not difficult to compute the cumulated path echelon inventory
of \((u, v)\) for each lot \(Q_u\):

\[
S(Q_u, \pi_u, \pi_{s(v)}) = \frac{1}{2} \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) Q_u^2.
\]

Because an integer-ratio lot size policy is a periodic policy, there is a finite regeneration instant \(T\) of the system such that all the inventories in the system will be zero at instant \(T\).

By Corollary 3.3.3, \(W_{s(v)}/T = \pi_0/\pi_{s(v)}\). Observe that the average path echelon inventory over the working time \(W_{s(v)}\) of facility \(s(v)\) is \(S(Q_u, \pi_u, \pi_{s(v)})/(Q_u/\pi_{s(v)})\). Therefore, the average path echelon inventory of \((u, v)\) on the entire horizon \([0, T)\) is

\[
E_{(u,v)} = \frac{S(Q_u, \pi_u, \pi_{s(v)}) W_{s(v)}}{Q_u/\pi_{s(v)}} = \frac{S(Q_u, \pi_u, \pi_{s(v)}) \pi_0}{Q_u/\pi_{s(v)}} \pi_{s(v)} = \frac{1}{2} \pi_0 \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) Q_u.
\]

This completes the proof. \(\square\)

More intuitively, to be optimal on a finite horizon, each lot \(Q_u\) has to be used up over a time interval of length \(Q_u/\pi_0\) in order to meet the constant demand rate \(\pi_0\). Therefore the average path echelon inventory of \((u, v)\) is

\[
E_{(u,v)} = \frac{S(Q_u, \pi_u, \pi_{s(v)})}{Q_u/\pi_0} = \frac{1}{2} \pi_0 \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) Q_u.
\]

The following theorem extends the formula of average route echelon inventory to a general situation.

**Theorem 3.4.3 (Average Route Echelon Inventory – the General Case)**

*For an integer–ratio lot size policy, the average route echelon inventory on \((u, 1)\) is*

\[
E_{(u,1)} = \Phi(Q; u, 1), \quad \text{for all} \quad u \in N,
\]

*where \(\Phi(Q; u, v) \overset{\Delta}{=} \frac{1}{2} \pi_0 \sum_{k \in (u,v)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{t \in (u,k)} Q_t.\)*
Proof. We prove it by induction on $|(u, 1)|$, the length of $(u, 1)$.

1. For $|(u, 1)| = 1$, we have $u = 1$. By Lemma 3.4.2 (with $u = v = 1$), (3.5) holds.

2. Suppose (3.5) holds for $|(u, 1)| \leq \ell$. We show that (3.5) holds for $|(u, 1)| = \ell + 1$.

There are two cases:

2.1 $Q_u \leq Q_{s(u)}$

By Lemma 3.4.2 (with $u = v$), the average inventory of product $u$ is

$$E_{(u, u)} = \frac{1}{2\pi_0} \left( \frac{1}{\pi_{s(u)}} - \frac{1}{\pi_u} \right) Q_u.$$  

Therefore, the average echelon inventory $E_{(u, 1)}$ on route $(u, 1)$ is, by induction,

$$E_{(u, 1)} = E_{(u, u)} + E_{(s(u), 1)}$$

$$= \frac{1}{2\pi_0} \left( \frac{1}{\pi_{s(u)}} - \frac{1}{\pi_u} \right) Q_u + \Phi(Q; s(u), 1)$$

$$= \frac{1}{2\pi_0} \left( \frac{1}{\pi_{s(u)}} - \frac{1}{\pi_u} \right) Q_u + \frac{1}{2\pi_0} \sum_{k \in (s(u), 1)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (s(u), k)} Q_{\ell}.$$  

Using $Q_u \leq Q_{s(u)}$ and noting $\Phi(Q; u, v) \triangleq \frac{1}{2\pi_0} \sum_{k \in (u, v)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (u, k)} Q_{\ell}$,

$$E_{(u, 1)} = \frac{1}{2\pi_0} \left( \frac{1}{\pi_{s(u)}} - \frac{1}{\pi_u} \right) Q_u + \frac{1}{2\pi_0} \sum_{k \in (s(u), 1)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (u, k)} Q_{\ell}$$

$$= \Phi(Q; u, 1).$$

2.2 $Q_u > Q_{s(u)}$

Let $i \in (s(u), 1)$ be the facility such that $Q_u > Q_j$ for all $j \in (s(u), i)$, and $Q_u \leq Q_{s(i)}$, (where $Q_{s(1)} \triangleq \max_{j \in N} Q_j$). If $i \neq 1$, then $s(i)$ is the first facility on route $(s(u), 1)$ such that $Q_u \leq Q_{s(i)}$. See Figure 3.4.

It is easy to check that the conditions in Lemma 3.4.2 hold on path $(u, i)$. They also hold trivially if $i = 1$. Therefore, the average path echelon inventory on $(u, i)$ is

$$E_{(u, i)} = \frac{1}{2\pi_0} \left( \frac{1}{\pi_{s(i)}} - \frac{1}{\pi_u} \right) Q_u.$$
Figure 3.4: Route \((u, 1)\) is divided into path \((u, i)\) and route \((s(i), 1)\)

\[
\begin{array}{c}
\circ u \\
\circ s(u) \\
\cdots \\
\circ i \\
\circ s(i) \\
\cdots \\
\circ 1
\end{array}
\]

\[
\langle u, i \rangle \\
\langle s(i), 1 \rangle
\]

\[
= \frac{1}{2} \pi_0 \sum_{k \in \langle u, i \rangle} \left( \frac{1}{\pi_s(k)} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u, k \rangle} Q_\ell.
\]

By induction on path \((s(i), 1)\), we have

\[
E_{\langle u, 1 \rangle} = E_{\langle u, i \rangle} + E_{\langle s(i), 1 \rangle}
\]

\[
= \frac{1}{2} \pi_0 \left( \frac{1}{\pi_u} - \frac{1}{\pi_s(i)} \right) Q_u + \Phi(Q; s(i), 1)
\]

\[
= \frac{1}{2} \pi_0 \sum_{k \in \langle u, i \rangle} \left( \frac{1}{\pi_s(k)} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u, k \rangle} Q_\ell
\]

\[
+ \frac{1}{2} \pi_0 \sum_{k \in \langle s(i), 1 \rangle} \left( \frac{1}{\pi_s(k)} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle s(i), k \rangle} Q_\ell
\]

\[
= \frac{1}{2} \pi_0 \sum_{k \in \langle u, i \rangle} \left( \frac{1}{\pi_s(k)} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u, k \rangle} Q_\ell
\]

\[
+ \frac{1}{2} \pi_0 \sum_{k \in \langle s(i), 1 \rangle} \left( \frac{1}{\pi_s(k)} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u, k \rangle} Q_\ell
\]

\[
= \frac{1}{2} \pi_0 \sum_{k \in \langle u, 1 \rangle} \left( \frac{1}{\pi_s(k)} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u, k \rangle} Q_\ell
\]

\[
= \Phi(Q; u, 1).
\]

In summary, (3.5) holds for \(|\langle u, 1 \rangle| = \ell + 1\). This completes the proof.

The following lemma on minimum average cost follows immediately.

Lemma 3.4.4 (Minimum Average Cost)

The minimum average cost \(C\) of integer-ratio lot size policies results from the following
Chapter 3. Finite Production Rate

mixed nonlinear integer programming problem (P),

\[
C = \min_Q \sum_{i \in N} \left\{ \frac{K_i}{Q_i/\pi_0} + h_i \Phi(Q_i; i, 1) \right\}
= \min_Q \sum_{i \in N} \left\{ \frac{K_i}{Q_i/\pi_0} + h_i \frac{1}{2} \pi_0 \sum_{k \in (i,1)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (i,k)} Q_{\ell} \right\},
\]  
\text{s.t.} \quad Q_i \geq 0, \quad \text{for all } i \in N, \quad (3.6a)
\frac{Q_u}{Q_v}, \text{ or } \frac{Q_v}{Q_u} \in N^*, \quad \text{for all } u \in N, \text{ and } v \in S(u). \quad (3.6b)

A continuous relaxation (RP) to (P) yields the lower bound \( C^* \) on the average cost of an integer-ratio lot size policy:

\[
C^* = \min_Q \sum_{i \in N} \left\{ \frac{K_i}{Q_i/\pi_0} + h_i \frac{1}{2} \pi_0 \sum_{k \in (i,1)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (i,k)} Q_{\ell} \right\},
\]  
\text{s.t.} \quad (3.6b).

### 3.5 Lower Bound Theorem

In this section, the Lower Bound Theorem (Theorem 3.5.6) shows that \( C^* \) is not only a lower bound on the average cost for integer-ratio lot size policies but also a lower bound on the average cost for all feasible policies. This main result of the chapter is built upon a series of lemmas as follows. A lower bound on average path echelon inventory, which is based on average lot sizes, is given below.

**Lemma 3.5.1 (Lower Bound on Average Path Echelon Inventory)**

For any feasible policy over the interval \([0,T]\) and any path \((u,v)\), the average path echelon inventory satisfies

\[
\frac{1}{T} \int_0^T E_{(u,v)}^t dt \geq \frac{1}{2} \pi_0 \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) \overline{Q}_u, \quad (3.8)
\]

where \( \overline{Q}_u \triangleq \pi_0 T/n_u \) is the average lot size of product \( u \) over the interval \([0,T]\), and \( n_u \) is the number of lots over the interval \([0,T]\).
Table 3.1: Slopes of the path echelon inventory level

<table>
<thead>
<tr>
<th>case #</th>
<th>status of $s(v)$</th>
<th>status of $u$</th>
<th>slope of $E_{(u,v)}^t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Working</td>
<td>Working</td>
<td>$\pi_u - \pi_{s(v)}$</td>
</tr>
<tr>
<td>2</td>
<td>Working</td>
<td>Idle</td>
<td>$-\pi_{s(v)}$</td>
</tr>
<tr>
<td>3</td>
<td>Idle</td>
<td>Working</td>
<td>$\pi_u$</td>
</tr>
<tr>
<td>4</td>
<td>Idle</td>
<td>Idle</td>
<td>0</td>
</tr>
</tbody>
</table>

Proof. For any feasible policy and any instant $t$, there are four possible cases for the slope of $E_{(u,v)}^t$:

Figure 3.5 on page 91 shows an example of inventory level $E_{(u,v)}^t$ on path $(u,v)$. It shows that the cumulated path echelon inventory on $(u,v)$ over the interval $[0,T)$. Its total area is no less the area of cases 1 and 2, i.e., during the working time $W_{s(v)}$ of facility $s(v)$. The area of cases 1 and 2 is shown under the thick lines in Figure 3.5 (A) and (B). During the idle times (cases 3 and 4) of facility $s(v)$, there is no demand on path $(u,v)$, i.e., the slope of $E_{(u,v)}^t$ is non-negative. Therefore the inventory level of $E_{(u,v)}^t$ before any idle time will be no less than that after the idle time. The total inventory on path $(u,v)$ with up-slope $\pi_u - \pi_{s(v)}$ and down-slope $-\pi_{s(v)}$, as shown in Figure 3.5 (C).

By convexity, this area is no less than that of $n_u$ equally spaced triangles, each with the base line length $W_{s(v)}/n_u$. Hence we have

$$\frac{1}{T} \int_0^T E_{(u,v)}^t \, dt \geq \frac{11}{T2} \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) \left( \frac{\pi_{s(v)} W_{s(v)}}{n_u} \right)^2 n_u$$

$$\geq \frac{11}{T2} \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) \frac{(\pi_0 T)^2}{n_u} \quad \text{(By Corollary 3.3.3)}$$

$$= \frac{11}{2} \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) \pi_0 \bar{Q}_u.$$
Figure 3.5: Illustrating the proof of Lemma 3.5.1 — facility $u$ produces four lots over the time interval $[0,T)$.

$E_{(u,v)}^r$  
(A) Echelon inventory on path $(u,v)$ over time interval $[0,T)$:

$E_{(u,v)}^c$  
(B) The cumulated echelon inventory is no less than the total area of cases 1 and 2:

$E_{(u,v)}^e$  
(C) The cumulated echelon inventory is no less than four saw-tooth areas over time interval $W_{s(v)}$: 
The next Lemma shows that path holding costs can be rewritten in a recursive form which is needed later.

**Lemma 3.5.2 (Decomposition of Path Holding Costs)**

Let \( H_{(u,v)}^t(h_{(u,v)}) \) be the total holding cost on path \((u,v)\) at instant \(t\) with echelon holding cost rate \(h_{(u,v)}\). Let path \((u,v)\) be decomposed as \((u,v) = ((u,i), (s(i),v))\), then \( H_{(u,v)}^t(h_{(u,v)}) \) can be decomposed as

\[
H_{(u,v)}^t(h_{(u,v)}) = \begin{cases} 
  h_u E_{(u,i)}^t(h_{(s(u),i)}) + H_{(s(i),v)}^t(h''_{(s(i),v)}), & \text{if } i \neq v, \\
  h_v E_{(u,v)}^t(h_{(s(u),v)}), & \text{if } i = v,
\end{cases}
\]  

(3.9)

where

\[
h''_{s(i)} \triangleq h_{s(i)} + \sum_{j \in (u,i)} h_j,
\]

\[
h''_{(s(i),v)} \triangleq (h''_{s(i)}, h_{(s(i),v)}).
\]

**Proof.** Note that

\[
H_{(u,v)}^t(h_{(u,v)}) \triangleq \sum_{j \in (u,v)} h_j E_{(j,v)}^t = \sum_{j \in (u,v)} \sum_{k \in (j,v)} h_j I_k,
\]

i.e., \( H_{(u,v)}^t(h_{(u,v)}) \) is the sum of the values of all the blocks in Figure 3.6 on page 93. The value of each block is the product of the vertical label and the horizontal label. Thus (3.9) is just another way to calculate the total value of \( H_{(u,v)}^t(h_{(u,v)}) \). \( \square \)

The following main lemma estimates the average path holding cost based on average lot sizes by using two lemmas above.

**Lemma 3.5.3 (Lower Bound on Average Path Holding Costs)**

\[
\frac{1}{T} \int_0^T H_{(u,v)}^t(h_{(u,v)}) \, dt \geq \sum_{j \in (u,v)} h_j \Phi(Q; j, v)
\]

\[
= \sum_{j \in (u,v)} h_j \frac{1}{2} \pi_0 \sum_{k \in (j,v)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (j,k)} Q_{\ell}.
\]  

(3.10)
Figure 3.6: Illustrating the proof of Lemma 3.5.2 — decomposition of the path echelon holding cost on \( (u, v) \).

Proof. We prove it by induction on \(|(u, v)|\), the length of path \( (u, v) \).

1. For \(|(u, v)| = 1\), we have \( u = v \). By definition, \( H_u^t(h_u) = h_u E_u^t \). By Lemma 3.5.1, 

\[
\frac{1}{T} \int_0^T H_u^t(h_u) \, dt \geq \ h_u \, \frac{1}{\pi_0} \left( \frac{1}{\pi_{s(u)}} - \frac{1}{\pi_u} \right) \bar{Q}_u \triangleq h_i \Phi(\bar{Q}; u, u),
\]

that is, (3.10) holds.

2. Suppose (3.10) holds for \(|(u, v)| \leq \ell\). We show that (3.10) also holds for \(|(u, v)| = \ell + 1\).

There are two cases to be considered.

2.1. If there exists \( i \in (u, v) \) such that \( \bar{Q}_{s(i)} \geq \bar{Q}_u \) and \( \bar{Q}_u \geq \bar{Q}_j \) for all \( j \in (s(u), i) \), then path \( (u, v) \) can be divided into two subpaths \( (u, i) = (u, i, s(i), v) \), see Figure 3.7.

By Lemma 3.5.2, we have

\[
\frac{1}{T} \int_0^T H_{(u,v)}^t(h_{(u,v)}) \, dt
= \frac{1}{T} \int_0^T h_u E_{(u,i)}^t \, dt + \frac{1}{T} \int_0^T H_{(s(u),i)}^t(h_{(s(u),i)}) \, dt + \frac{1}{T} \int_0^T H_{(s(i),v)}^t(h_{(s(i),v)}) \, dt.
\]
By Lemma 3.5.1, we have

\[
\frac{1}{T} \int_0^T h_u E^t_{(u,i)} \ dt \geq h_u \frac{1}{2} \pi_0 \left( \frac{1}{\pi_{s(i)}} - \frac{1}{\pi_u} \right) \overline{Q}_u
\]

\[
= h_u \frac{1}{2} \pi_0 \sum_{k \in (u,i)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \overline{Q}_u
\]

\[
= A_1,
\]

where

\[
A_1 = h_u \frac{1}{2} \pi_0 \sum_{k \in (u,i)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (u,k)} \overline{Q}_\ell.
\]

Because \(|(s(u),i)| \leq \ell\), and \(|(s(i),v)| \leq \ell\), we have, by induction, the following two inequalities,

\[
\frac{1}{T} \int_0^T H^t_{(s(u),i)}(h_{s(u),i}) \ dt \geq A_2,
\]

and

\[
\frac{1}{T} \int_0^T H^t_{(s(i),v)}(h''_{s(i),v}) \ dt \geq A_3 + A_4,
\]

where

\[
A_2 = \sum_{j \in (s(u),i)} h_j \frac{1}{2} \pi_0 \sum_{k \in (j,i)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (j,k)} \overline{Q}_\ell,
\]

\[
A_3 = \sum_{j \in (s(i),v)} h_j \frac{1}{2} \pi_0 \sum_{k \in (j,v)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (j,k)} \overline{Q}_\ell,
\]

\[
A_4 = \left( \sum_{j \in (u,i)} h_j \right) \frac{1}{2} \pi_0 \sum_{k \in (s(i),v)} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in (s(i),k)} \overline{Q}_\ell,
\]
That is,

the LHS of (3.10) $\geq A_1 + A_2 + A_3 + A_4$.

However, the RHS of (3.10) can be rewritten as,

$$\sum_{j \in \langle u,v \rangle} h_j \frac{1}{2} \pi_0 \sum_{k \in \langle j,v \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle j,k \rangle} \overline{Q}_\ell = B_1 + B_2 + B_3 + B_4 + B_5,$$

where

$$B_1 = h_u \frac{1}{2} \pi_0 \sum_{k \in \langle u,i \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u,k \rangle} \overline{Q}_\ell,$$

$$B_2 = h_u \frac{1}{2} \pi_0 \sum_{k \in \langle s(i),v \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle u,k \rangle} \overline{Q}_\ell,$$

$$B_3 = \sum_{j \in \langle s(u),i \rangle} h_j \frac{1}{2} \pi_0 \sum_{k \in \langle j,i \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle j,k \rangle} \overline{Q}_\ell,$$

$$B_4 = \sum_{j \in \langle s(u),i \rangle} h_j \frac{1}{2} \pi_0 \sum_{k \in \langle s(i),v \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle j,k \rangle} \overline{Q}_\ell,$$

$$B_5 = \sum_{j \in \langle s(u),i \rangle} h_j \frac{1}{2} \pi_0 \sum_{k \in \langle j,i \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle j,k \rangle} \overline{Q}_\ell.$$

It is easy to check that $A_1 = B_1, A_2 = B_3,$ and $A_3 = B_5$. As for $B_2$ and $B_4$, because $k \in \langle s(i),v \rangle$, we have $s(i) \in \langle u,k \rangle$ and $\overline{Q}_{s(i)} \geq \overline{Q}_j$, for all $j \in \langle u,i \rangle$. Thus, we get

$$\max_{\ell \in \langle u,k \rangle} \overline{Q}_\ell = \max_{\ell \in \langle j,k \rangle} \overline{Q}_\ell = \max_{\ell \in \langle s(i),k \rangle} \overline{Q}_\ell.$$ Hence,

$$B_2 + B_4 = \sum_{j \in \langle u,i \rangle} h_j \frac{1}{2} \pi_0 \sum_{k \in \langle s(i),v \rangle} \left( \frac{1}{\pi_{s(k)}} - \frac{1}{\pi_k} \right) \max_{\ell \in \langle s(i),k \rangle} \overline{Q}_\ell = A_4.$$

That is, $A_1 + A_2 + A_3 + A_4 = B_1 + B_2 + B_3 + B_4 + B_5$, i.e., (3.10) holds in this case.

2.2. If there is no such $i \in \langle u,v \rangle$, that is, $\overline{Q}_u > \overline{Q}_j$ for all $j \in \langle s(u),v \rangle$, then

$$\frac{1}{T} \int_0^T H^t_{\langle u,v \rangle}(h_{\langle u,v \rangle}) \, dt = \frac{1}{T} \int_0^T h_u E^t_{\langle u,v \rangle} \, dt + \frac{1}{T} \int_0^T H^t_{\langle s(u),v \rangle}(h_{\langle s(u),v \rangle}) \, dt.$$

Because

$$\frac{1}{T} \int_0^T h_u E^t_{\langle u,v \rangle} \, dt \geq h_u \frac{1}{2} \pi_0 \left( \frac{1}{\pi_{s(v)}} - \frac{1}{\pi_u} \right) \overline{Q}_u \quad (\text{by Lemma 3.5.1})$$
and by induction,
\[
\frac{1}{T} \int_0^T H_{(s(u),v)}^t(h_{(s(u),v)}) \, dt \\
\geq \sum_{j \in (s(u),v)} h_j \frac{1}{2} \pi_0 \sum_{k \in (s(k),v)} \left( \frac{1}{\Lambda_{s(k)}} - \frac{1}{\Lambda_k} \right) \max_{\epsilon \in (s(k),v)} \bar{Q}_\epsilon,
\]
(3.10) also holds in this case.

In summary, (3.10) holds for \(|(u,v)| = \ell + 1\). This completes the proof. 

All results so far have concerned paths. They are now generalized to assembly systems. The following Lemma describes an allocation of the given echelon holding costs to so-called leaf-routes, that is, routes from the leaf-nodes to the root of the assembly system.

**Lemma 3.5.4 (Holding Costs Allocation in Assembly Systems)**

Let \(L = \{i_1, \cdots, i_q\}\) be the set of all leaves (facilities with no predecessors) in the assembly system. Call \((i_\ell, 1)\), for all \(\ell = 1, \cdots, q\), the leaf-routes of the assembly system. Let \(H_G^t(h_N) \triangleq \sum_{j \in N} h_j E_{(i_1,1)}^t\) be the total holding cost on graph \(G(N, A)\) at instant \(t\) with echelon holding cost rate \(h_N\). Define
\[
h_j^\ell \triangleq \begin{cases} 
    h_j, & \text{if } j \in (i_1,1), \text{ for } \ell = 1, \\
    h_j, & \text{if } j \in (i_\ell,1) \setminus \bigcup_{k=1}^{\ell-1} (i_k,1), \text{ for } \ell = 2, \cdots, q, \\
    0, & \text{otherwise.}
\end{cases}
\]
as the leaf-route-allocated holding cost rate on \((i_\ell, 1)\). And let
\[
h^\ell \triangleq \{h_j^\ell | j \in (i_\ell, 1)\}.
\]
Figure 3.8: Illustrating the proof of Lemma 3.5.4 — assembly system $G(N, A)$ is partitioned into leaf-routes.

be the leaf-route-allocated holding cost rate vector on $(i_t, 1)$. Then the total holding cost at instant $t$ on the assembly system is the sum of holding costs at instant $t$ on all leaf-routes with leaf-route-allocated holding cost rate vector $h^t$, i.e.,

$$H_G^t(h_N) = \sum_{t=1}^q H_{(i_t, 1)}^t(h^t).$$  \hfill (3.11)

Proof. By the definition of $H_G^t(h_N)$, and noting that $h_j = \sum_{(i, j, 1) \ni j} h_j^t$, we have:

$$H_G^t(h_N) = \sum_{j \in N} h_j E_{(j, 1)}^t = \sum_{j \in N} \sum_{(i_t, 1) \ni j} h_j^t E_{(j, 1)}^t = \sum_{t=1}^q \sum_{j \in (i_t, 1)} h_j^t E_{(j, 1)}^t = \sum_{t=1}^q H_{(i_t, 1)}^t(h^t).$$

This completes the proof. \hfill \Box

The lower bound lemma on average holding costs can now be extended from paths (Lemma 3.5.3) to assembly systems:
Lemma 3.5.5 (Lower Bound on Average Holding Cost for Assembly Systems)

\[
\frac{1}{T} \int_0^T H_G^t(h_N) \, dt \geq \sum_{j \in N} h_j \Phi(\overline{Q}_t; j, 1)
\]
\[
= \sum_{j \in N} h_j \frac{1}{2} \pi_0 \max_{\ell \in \{j, k\}} \max_{\ell \in \{j, k\}} \overline{Q}_\ell.
\]

**Proof.** We prove it by induction on \(|L|\), the number of leaves.

1. For \(|L| = 1\), (3.12) holds by Lemma 3.5.3 with \(G = (n, 1)\).
2. Suppose (3.12) holds for \(|L| = q\), we show that (3.12) also holds for \(|L| = q + 1\). By Lemma 3.5.4 and the notation defined there,

\[
\frac{1}{T} \int_0^T H_G^t(h_N) \, dt = \frac{1}{T} \int_0^T H_G^t(h_{N'}) \, dt + \frac{1}{T} \int_0^T H_G^t(h_{q+1}) \, dt.
\]

By induction,

\[
\frac{1}{T} \int_0^T H_G^t(h_{N'}) \, dt \geq \sum_{j \in N'} h_j \frac{1}{2} \pi_0 \max_{\ell \in \{j, k\}} \max_{\ell \in \{j, k\}} \overline{Q}_\ell.
\]

By Lemma 3.5.3,

\[
\frac{1}{T} \int_0^T H_G^t(h_{q+1}) \, dt \geq \sum_{j \in \{q+1, 1\}} h_j^{q+1} \sum_{k \in \{j, 1\}} \frac{1}{2} \pi_0 \max_{\ell \in \{j, k\}} \overline{Q}_\ell
\]
\[
= \sum_{j \in N \setminus N'} h_j \frac{1}{2} \pi_0 \max_{\ell \in \{j, k\}} \overline{Q}_\ell.
\]

Combining these two inequalities, we conclude that (3.12) also holds for \(|L| = q + 1\). This completes the proof. \(\square\)

The Lower Bound Theorem below, one of the key results in this chapter, now follows directly from Lemma 3.5.5:

**Theorem 3.5.6 (Lower Bound Theorem)**

For any feasible policy, the long run average cost
That is, the optimal value of continuous relaxation for integer-ratio lot size policies is also the lower bound on the average cost of all feasible policies.

Proof. For any feasible policy over the interval \([0, T)\), the average total set-up cost is 
\[ \sum_{i \in N} K_i n_i / T = \sum_{i \in N} K_i / (\overline{Q}_i / \pi_0). \] By Lemma 3.5.5, the average cost over the interval \([0, T)\) is
\[ \sum_{i \in N} K_i / (\overline{Q}_i / \pi_0) + \frac{1}{T} \int_0^T H_G(h_N) \, dt \geq \sum_{i \in N} K_i / (\overline{Q}_i / \pi_0) + \sum_{i \in N} h_i \Phi(\overline{Q}_i; i, 1). \]
Therefore, by the definition of \(C^*\),
\[ \text{the long run average total cost} = \liminf_{T \to \infty} \left\{ \sum_{i \in N} \frac{K_i}{Q_i / \pi_0} + \frac{1}{T} \int_0^T H_G(h_N) \, dt \right\} \geq C^*. \]
This completes the proof. \(\square\)

3.6 Effectiveness of Integer-Ratio and Power-of-Two Lot Size Policies

Given an optimal solution \(Q^* = (Q_1^*, \ldots, Q_n^*)\) to the continuous relaxation \((RP)\), an \textit{Order-Preserving} Solution \(Q = (Q_1, \ldots, Q_n)\) is an integer-ratio lot size policy to problem \((P)\) such that \(Q_i = Q_j\) (resp., \(Q_i \leq Q_j\); resp., \(Q_i \geq Q_j\)), whenever \(Q_i^* = Q_j^*\) (resp., \(Q_i^* < Q_j^*\); resp., \(Q_i^* > Q_j^*\)), for all \(i, j \in N\).

The following Lemma repeats the result in Roundy [27] (1983), and we state it without proof.

\textbf{Lemma 3.6.1 (Effectiveness of an Order-Preserving Policy)}

If \(Q\) is an order-preserving integer-ratio lot size solution to \((P)\), then
\[ \frac{C(Q^*)}{C(Q)} \geq \min_{i \in N} e\left( \frac{Q_i}{Q_i^*} \right), \]
where $e(q) \triangleq \frac{2}{q + 1/q}$.

The next Rounding Lemma repeats the result in Roundy [27] (1983), and we state it again without proof.

**Lemma 3.6.2 (Rounding Lemma)**

If $Q^*$ is an optimal solution to the continuous relaxation (RP) and $Q$ is defined by

$$Q_i \triangleq Q_0 2^{m_i}, \quad \text{for all } i \in N,$$

$$m_i \triangleq \lfloor \log_2(Q_i/Q_0) + 1/2 \rfloor,$$

then $Q$ is an optimal power-of-two lot size solution to (P) for a given positive lot size $Q_0$.

Using these two Lemmas, we can get the following two effectiveness theorems, which are proved in Roundy [27] (1983). The effectiveness of a policy, also defined in [27] (1983), is the ratio of the infimum of the average cost over all policies to the average cost of the policy in question.

**Theorem 3.6.3 (94%-Effectiveness Theorem)**

For every fixed $Q_0 > 0$ there exists an optimal power-of-two lot size policy with base lot size $Q_0$, whose effectiveness is at least 94%.

**Theorem 3.6.4 (98%-Effectiveness Theorem)**

There exists an optimal power-of-two lot size policy with base lot size $Q^*_0$ in the interval $[\sqrt{5}, \sqrt{2}]$, whose effectiveness is at least 98%.

3.7 Finding Optimal Power-of-Two Lot Size Policies

We sincerely thank Roundy for suggesting the following method, which cuts down the $O(n^5)$ running time of our original algorithm to $O(n^3 \log n)$ for the new algorithm obtained below.
Chapter 3. Finite Production Rate

First, we rewrite the original relaxation problem \((RP)\) to a new equivalent problem \((RP_1)\). Then, we present a mapping from this model to the model presented in Roundy [31] (1987). Finally, we show an algorithm solving the original problem in \(O(n^3 \log n)\).

Lemma 3.7.1 (Equivalent Formulation)

Problem \((RP)\):

\[
C^* = \max_Q f(Q) \triangleq \sum_{i \in N} \left[ \frac{K_i}{Q_i/\pi_0} + \sum_{j \in \langle i, 1 \rangle} H_{ij} \max_{t \in \langle i, j \rangle} Q_t \right]
\]

s.t. \(Q_i > 0\) \quad \forall i \in N

is equivalent to problem \((RP_1)\):

\[
C_1^* = \max_q f_1(q) \triangleq \sum_{i \in N} \left[ \frac{K_i}{q_{ii}/\pi_0} + \sum_{j \in \langle i, 1 \rangle} H_{ij} q_{ij} \right]
\]

s.t. \(q_{ij} \geq 0\) \quad \forall \langle i, j \rangle \in R,

(3.14a)

\(q_{ij} \leq q_{i,s(j)}\) \quad \forall \langle i, s(j) \rangle \in R,

(3.14b)

\(q_{ij} \geq q_{s(i), j}\) \quad \forall \langle s(i), j \rangle \in R,

(3.14c)

where \(R\) is the set of all paths in \(G(N, A)\).

Proof. Suppose that \(Q = (Q_1, \cdots, Q_n)\) is a feasible solution to \((RP)\). Let

\[
q_{ij} \triangleq \max_{t \in \langle i, j \rangle} Q_t, \quad \forall \langle i, j \rangle \in R.
\]

Then inequalities (3.14a); (3.14b) and (3.14c) hold, that is., \(q = (q_{ij}|\langle i, j \rangle \in R)\) is also a feasible solution to \((RP_1)\). Note also that \(q_{ii} = Q_i, \forall i \in N\). Therefore, \(f_1(q) = f(Q)\), and \(C_1^* \leq C^*\).

Now suppose that \(q\) is a feasible solution to \((RP_1)\). Let \(Q_i = q_{ii}, \forall i \in N\). If \(\ell \in \langle i, j \rangle \in R\), then by (3.14c) \(q_{ij} \geq q_{s(i), j} \geq \cdots \geq q_{\ell, j}\), and by (3.14b) \(q_{\ell, q} \leq q_{\ell, s(\ell)} \leq \cdots \leq q_{\ell, j}\).
Therefore, $Q_t = q_{it} \leq q_{ij}, \forall t \in (i,j)$, and $\max_{t \in (i,j)} Q_t \leq q_{ij}$. Hence, $f(Q) \leq f_1(q)$, and $C^* \leq C_1^*$.

This implies $C^* = C_1^*$, i.e., problem $(RP)$ is equivalent to problem $(RP_1)$. 

Let $G(N_1, A_1)$ be a network corresponding to problem $(RP_1)$:

\[ N_1 \triangleq \{(i,j) \in R\}, \]
\[ A_1 \triangleq \{ \{(i, s(j)), (i, j)\} | (i, s(j)) \in R \} \cup \{(i, j), (s(i), j)\} | (s(i), j) \in R \} \]

Let $D \triangleq \max_{i \in L} |(i, 1)|$ be the length of a longest leaf-route in $G(N, A)$, where $L$ is the set of all leaves. Let $G(N', A')$ be the series system with node set $N' \triangleq \{0, 1, 2, \cdots, D-1\}$ and arc set $A' \triangleq \{(i, i-1) | i = 1, 2, \cdots, D-1\}$. Let $G(N_2, A_2)$ be a graph defined by

\[ N_2 \triangleq \{(i, k) | i \in N, k = 0, 1, 2, \cdots, D-1\} \]
\[ A_2 \triangleq \{((i, k), (s(i), k)) | s(i) \in N, k = 0, 1, 2, \cdots, D-1\} \]
\[ \cup \{((i, k), (i, k-1)) | i \in N, k = 1, 2, \cdots, D-1\} \]

The problem of minimizing the relaxed average cost of a nested policy on this network can be solved in $O(|R| D \log |R|)$ time, by the algorithm in Roundy [31] (1987). Note that $|R| \leq n(n-1)/2$ and $D \leq n$, so $O(|R| D \log |R|) \leq O(n^3 \log n)$.

The way to embed $G(N_1, A_1)$ into $G(N_2, A_2)$ is as follows:

\[ \text{node } (i, j) \in N_1 \rightarrow \text{node } (i, k) \in N_2, \quad \text{with } k = D - |(j, 1)|. \]

Note that if $k \leq D - 1 - |(i, 1)|$ then node $(i, k) \in N_2$ has no corresponding node $(i, j) \in N_1$. In that case, we set $K_{(i,k)} = 0$ and $h_{(i,k)} = 0$ so that it does not effect the final solution. After embedding $G(N_1, A_1)$ into $G(N_2, A_2)$, the problem can be solved on $G(N_2, A_2)$ and the solution to our problem can be extracted. The number of nodes and arcs in $G(N_2, A_2)$ is at most twice the number of nodes and arcs, respectively, in either $G(N_1, A_1)$ or $G(N', A')$. 

The algorithm for finding an optimal power-of-two lot size policy consists of two stages:

Stage 1 Solve relaxation \((RP)\) in \(O(|R|D \log |R|) \leq O(n^3 \log n)\), by using the algorithm above suggested by Roundy.

Stage 2 Using the "Rounding Lemma" with the complexity of \(O(n \log n)\), generate an optimal power-of-two solution to \((P)\), with at least 94% effectiveness, from a solution to \((RP)\) for a fixed base lot size \(Q_0\). Using Roundy’s method in [27] (1983), derive an optimal base lot size \(Q_0\) and a corresponding optimal power-of-two lot size policy, which has at least 98% effectiveness.

Example. The following example of an assembly system \(G(N, A)\) in Figure 3.9 with 7 facilities illustrates the embedding procedure. The length of the longest leaf-route, which corresponds to a series network \(G(N', A')\) in Figure 3.10, is four. Graph \(G(N_1, A_1)\) in Figure 3.11 is a network corresponding to problem \((RP_1)\), and is embedded in graph \(G(N_2, A_2)\) in Figure 3.12, which is the Cartesian product of graph \(G(N, A)\) and graph \(G(N', A')\).
Figure 3.9: Graph $G(N, A)$

Figure 3.10: Graph $G(N', A')$
Figure 3.11: Graph $G(N_1, A_1)$, which is embedded in graph $G(N_2, A_2)$
Figure 3.12: Graph $G(N_2, A_2) = G(N, A) \times G(N', A')$
3.8 Conclusions

This chapter has considered a lot-sizing problem in an assembly system with finite production rates. A key assumption is that the production rates are non-increasing along any path in the system. We show that the best integer-split policy defined by Schwarz and Schrage [34] (1975) and the best integer-multiple policy defined by Moily [20] (1986) can be arbitrarily bad in Section 3.9.1. To overcome this difficulty, we introduced a class of integer-ratio lot size policies. The non-increasing production rates assumption allows us to formulate the problem of finding a minimum cost integer-ratio lot size policy as a mixed nonlinear integer programming problem. A continuous relaxation of this formulation yields a lower bound on the cost of any feasible policy. The derivation of this Lower Bound Theorem is novel, and relies on the concept of path holding costs, which is a generalization of echelon holding costs. Using the methods introduced by Roundy ([27] (1983) and [30] (1986)), we then construct an optimal power-of-two integer-ratio lot size policy with cost within 2% of the lower bound. Following a suggestion from Roundy, this policy is found by a $O(n^3 \log n)$ algorithm.

The results in this chapter have several possible implications. First, it demonstrates that when studying production systems, one should not restrict attention to overly simplified classes of policies (here, integer-split and integer-multiple lot size policies) unless they can be proven optimal or near-optimal. Second, we have shown that the class of power-of-two policies introduced by Roundy extends to finite production rate systems, provided we consider lot sizes, instead of reorder intervals as in Roundy’s original work. Third, this class of power-of-two lot size policies is a class of well-structured, provably very near-optimal and implementable policies which can be determined in polynomial time.

Further directions for research may include relaxing the non-increasing production
Figure 3.13: Three facilities in series

\[
\begin{array}{c}
3 \rightarrow 2 \rightarrow 1 \rightarrow \pi_0
\end{array}
\]

rate assumption, and studying other production networks such as distribution systems, and general acyclic networks. Research along these lines is being undertaken.

3.9 Appendix to Chapter 3

3.9.1 Effectiveness of Integer-Multiple and Integer-Split Lot Size Policies

From the analysis of integer-multiple lot size policies in Crowston, Wagner, and Henshaw [6] (1972) and integer-split lot size policies in Moily [19] (1982) and [20] (1986), it can be shown that either class of policies may be arbitrarily worse than the other, depending on the ratio of setup costs to holding cost rates. The example below shows that both classes of policies can simultaneously be arbitrarily worse than integer-ratio lot size policies.

The system has 3 facilities in a series, see Figure 3.13.

The parameters are \( K_1 = 16m^2, K_2 = 4, K_3 = 8m^2, h_1 = 2, h_2 = 8m^2, h_3 = 2, \pi_0 = 1, \pi_1 = (16m^2 + 8)/(16m^2 + 7), \pi_2 = (16m^2 + 8)/(12m^2 + 5), \pi_3 = (16m^2 + 8)/(8m^2 + 3) \), where \( m \) is a positive integer, large enough for our purpose — see below.

Consider the following three classes of policies (policies in 1 are integer-multiple lot size policies, in 2 are integer-split lot size policies, and in 3 are integer-ratio lot size policies):


Using actual inventory, and noting that \( 1/\pi_0 - 1/\pi_1 = 1/(16m^2 + 8) \), \( 1/\pi_1 - 1/\pi_2 = \)
Figure 3.14: Network corresponding to the constraints of the 3-facility problem

\[ \text{1} \rightarrow \text{2} \rightarrow \text{3} \]

1/4, 1/\pi_2 - 1/\pi_3 = 1/4, the cost of an integer-ratio lot size policy \( Q \) is:

\[
C_{\text{IMLSP}}(Q) = \frac{K_1}{Q_1} + \frac{K_2}{Q_2} + \frac{K_3}{Q_3} + \frac{1}{2}(h_1 + h_2 + h_3)\left(\frac{1}{\pi_0} - \frac{1}{\pi_1}\right)Q_1 + \frac{1}{2}(h_2 + h_3)\left(\frac{1}{\pi_1} - \frac{1}{\pi_2}\right)Q_2 + \frac{1}{2}h_3\left(\frac{1}{\pi_2} - \frac{1}{\pi_3}\right)Q_3
\]

\[
= \frac{16m^2}{Q_1} + \frac{4}{Q_2} + \frac{8m^2}{Q_3} + \frac{1}{4}Q_1 + \left(m^2 + \frac{1}{4}\right)Q_2 + \frac{1}{4}Q_3
\]

The Minimum Violators Algorithm of Roundy [27] (1983) will solve the following continuous relaxation

\[
\min_{Q > 0} C_{\text{IMLSP}}(Q)
\]

s.t. \( Q_3 \leq Q_2 \leq Q_1 \),

and produce a lower bound of for this class of policies. The algorithm uses a graph in Figure 3.14, which is the reverse of the production graph.

At the beginning of the algorithm, each group has one element, and

\[
\frac{K^{(1)}}{h^{(1)}} = \frac{16m^2}{1/4} = 64m^2, \quad \frac{K^{(2)}}{h^{(2)}} = \frac{4}{m^2 + 1/4}, \quad \frac{K^{(3)}}{h^{(3)}} = \frac{8m^2}{1/4} = 32m^2.
\]

For \( m \geq 1 \), \( K^{(2)}/h^{(2)} \) is the smallest ratio and \( s(2) = 3 \). Therefore we collapse set \( \{2\} \) into set \( \{3\} \) and have

\[
\frac{K^{(2,3)}}{h^{(2,3)}} = \frac{8m^2 + 4}{m^2 + 1/4 + 1/4} = 8.
\]

Because \( K^{(2,3)}/h^{(2,3)} < K^{(1)}/h^{(1)} \), the partition \( \{\{1\}, \{2,3\}\} \) is optimal. Therefore,

\[
Q_1^* = \sqrt{\frac{K^{(1)}}{h^{(1)}}} = 8m, \quad Q_2^* = Q_3^* = \sqrt{\frac{K^{(2,3)}}{h^{(2,3)}}} = 2\sqrt{2},
\]
and a lower bound on the average cost of integer-multiple lot size policies is,

\[
C_{IMLSP}^* = C_{IMLSP}(Q^*) = 2 \left( \sqrt{16m^2 \times 1/4} + \sqrt{(8m^2 + 4)(m^2 + 1/4 + 1/4)} \right) = 2 \left[ 2m + \sqrt{2(2m^2 + 1)} \right] = O(m^2).
\]


Using echelon inventories, the cost of an integer-split lot size policy \(Q\) is:

\[
\min_{Q > 0} C_{ISLSP}(Q) = K_1 + \frac{K_2}{Q_1} + \frac{K_3}{Q_2} + \frac{K_4}{Q_3} + \frac{h_1}{2} \left( \frac{1}{\pi_0} - \frac{1}{\pi_1} \right) Q_1 + \frac{h_2}{2} \left( \frac{1}{\pi_0} - \frac{1}{\pi_2} \right) Q_2 + \frac{h_3}{2} \left( \frac{1}{\pi_0} - \frac{1}{\pi_3} \right) Q_3
\]

\[
= \frac{16m^2}{Q_1} + \frac{4m^2}{Q_2} + \frac{8m^2}{Q_3} + \frac{1}{16m^2 + 8} Q_1 + \frac{m^2(4m^2 + 3)}{4m^2 + 2} Q_2 + \frac{8m^2 + 5}{16m^2 + 8} Q_3.
\]

The same Minimum Violators Algorithm will solve the following continuous relaxation

\[
\min_{Q > 0} C_{ISLSP}(Q)
\]

\[
\text{s.t. } Q_3 \geq Q_2 \geq Q_1,
\]

and produce a lower bound for this class of policies. The algorithm uses a graph which is the same as the production graph.

At the beginning of the algorithm, each group has one element, and

\[
\frac{K^{(1)}}{h^{(1)}} = \frac{16m^2}{1/8(2m^2 + 1)} = 108(m^2 + 1),
\]

\[
\frac{K^{(2)}}{h^{(2)}} = \frac{4}{m^2(4m^2 + 3)/2(2m^2 + 1)} = \frac{8(2m^2 + 1)}{m^2(4m^2 + 3)},
\]

\[
\frac{K^{(3)}}{h^{(3)}} = \frac{8m^2}{(8m^2 + 5)/8(2m^2 + 1)} = \frac{64m^2(2m^2 + 1)}{8m^2 + 5}.
\]

For \(m \geq 1\), \(K^{(2)}/h^{(2)}\) is the smallest ratio and \(s(2) = 1\). Therefore we collapse set \(\{2\}\) into set \(\{1\}\) and have

\[
\frac{K^{(1,2)}}{h^{(1,2)}} = \frac{16m^2 + 4}{1/8(2m^2 + 1) + m^2(4m^2 + 3)/2(2m^2 + 1)}
\]
Chapter 3. Finite Production Rate

\[
\frac{4(4m^2 + 1)}{1 + 12m^2 + 16m^4 / (2m^2 + 1)} = \frac{32(4m^2 + 1)}{16m^4 + 12m^2 + 1},
\]

If \(m\) is large enough, then \(K^{(1,2)}/h^{(1,2)} < K^{(3)}/h^{(3)}\), and the partition \(\{\{1, 2\}, \{3\}\}\) is optimal. Therefore,

\[
Q_1^* = Q_2^* = \sqrt{\frac{K^{(1,2)}}{h^{(1,2)}}} = 4\sqrt{\frac{2(4m^2 + 1)}{16m^4 + 12m^2 + 1}}, \quad Q_3^* = \sqrt{\frac{K^{(3)}}{h^{(3)}}} = 8m\sqrt{\frac{2m^2 + 1}{8m^2 + 5}},
\]

and a lower bound on the average cost of integer-split lot size policies is,

\[
C_{ISLSP}^* = C_{ISLSP}(Q^*) = 2 \left[ \sqrt{\frac{4(4m^2 + 1)}{16m^4 + 12m^2 + 1}} - \frac{2(4m^2 + 1)}{8(2m^2 + 1)} + \sqrt{\frac{8m^2}{8m^2 + 5}} \right] = O(m^2).
\]

3. Integer-ratio lot size policies.

Using the results in this chapter, the cost of an integer-ratio lot size policy \(Q\) is:

\[
C_{IRLSP}(Q) = \frac{K_1}{Q_1} + \frac{K_2}{Q_2} + \frac{K_3}{Q_3} + \frac{1}{2} h_1 \left( \frac{1}{\pi_0} - \frac{1}{\pi_1} \right) Q_1 + \frac{1}{2} h_2 \left[ \left( \frac{1}{\pi_1} - \frac{1}{\pi_2} \right) Q_2 + \left( \frac{1}{\pi_0} - \frac{1}{\pi_1} \right) (Q_2 \lor Q_1) \right] + \frac{1}{2} h_3 \left[ \left( \frac{1}{\pi_2} - \frac{1}{\pi_3} \right) Q_3 + \left( \frac{1}{\pi_1} - \frac{1}{\pi_2} \right) (Q_3 \lor Q_2) + \left( \frac{1}{\pi_0} - \frac{1}{\pi_1} \right) (Q_3 \lor Q_2 \lor Q_1) \right]
\]

\[
= \frac{16m^2}{Q_1} + \frac{4}{Q_2} + \frac{8m^2}{Q_3} + \frac{1}{16m^2 + 8} Q_1 + m^2 \left[ Q_2 + \frac{1}{4m^2 + 2} (Q_2 \lor Q_1) \right] + \frac{1}{4} Q_3 + \frac{1}{4} (Q_3 \lor Q_2) + \frac{1}{16m^2 + 8} (Q_3 \lor Q_2 \lor Q_1)
\]

where \(a \lor b \triangleq \max\{a, b\}\).

Applying the algorithm in Section 3.7 yields \(Q_2 \leq Q_3 \leq Q_1\) and

\[
\min_{Q \geq 0} C_{IRLSP}(Q) = \frac{16m^2}{Q_1} + \frac{4}{Q_2} + \frac{8m^2}{Q_3} + \frac{1}{4} Q_1 + m^2 Q_2 + \frac{1}{2} Q_3.
\]
We can verify that
\[
Q_1^* = \sqrt{\frac{16m^2}{1/4}} = 8m, \quad Q_2^* = \sqrt{\frac{4}{m^2}} = \frac{2}{m}, \quad Q_3^* = \sqrt{\frac{8m^2}{1/2}} = 4m
\]
defines a feasible integer-ratio lot size policy and its average cost is:
\[
C_{IRLSP}(Q^*) = 2 \left( \sqrt{16m^2 \times 1/4} + \sqrt{4m^2} + \sqrt{8m^2 \times 1/2} \right) = 12m.
\]

Therefore,
\[
\frac{C_{IRLSP}(Q^*)}{\min\{C_{IMLS}, C_{ISLSP}\}} = \frac{12m}{O(m^2)} \rightarrow 0, \quad (m \to \infty)
\]
That is, the effectiveness of optimal integer-multiple and integer-split lot size policies is zero in the worst case, i.e., these two policies can be arbitrarily worse than integer-ratio lot size policies.

3.9.2 Constructing a Production Schedule from Integer-Ratio Lot Size Policies

Given integer-ratio lot sizes \(Q_1, \ldots, Q_n\), we illustrate how to obtain a corresponding production schedule. As indicated in Section 3.3, we assume that the initial inventories at all facilities are zero. As the actual working time \(T_i = Q_i/\pi_i\) for each facility \(i\) to produce one lot is constant, specifying the starting time \(t_u(j)\) of each production run \(j\) at each facility \(u\) is enough to define the production schedule. Let \(L \triangleq \{i_1, \ldots, i_q\}\) be the set of all leaves (facilities without predecessors) in assembly system \(G(N,A)\), where \(q\) is the number of leaves in the system. Let \(Q_{\text{max}}^i \triangleq \max_{j \in \{i_1, 1\}} Q_j\) be the maximum lot size on route \((i_\ell, 1)\) for all \(i_\ell \in L\). Let \(r_\ell \triangleq Q_{\text{max}}^\ell / Q_1\) be the maximum ratio on route \((i_\ell, 1)\) at facility 1, and \(r(1) \triangleq \text{lcm}\{r_1, \ldots, r_q\}\), where \(\text{lcm}\{r_1, \ldots, r_q\}\) denotes the least common multiple of integers \(r_1, \ldots, r_q\). Let \(T = r(1)Q_1/\pi_0\) be the cycle period, and \(r(u) \triangleq r(1)(Q_1/Q_u)\) be the number of production lots at facility \(u\) over the interval \([0, T)\). Note that \(r(u)\) is
an integer: suppose \( u \in (i, 1) \), then \( Q_{\text{max}} / Q_u \) is an integer, and therefore

\[
r(u) = \frac{Q_1}{Q_u} r(1) = \frac{Q_1}{Q_u} \frac{r(1)}{r_\ell} = \frac{Q_1}{Q_u} \frac{Q_{\text{max}}}{Q_1} \frac{r(1)}{Q_{\text{max}}} = \frac{Q_{\text{max}} r(1)}{Q_u r_\ell}
\]

is an integer.

Scheduling Algorithm in Figure 3.15 will generate a periodic production schedule.

Procedure Schedule\((t, v)\) called by Scheduling Algorithm is defined in Figure 3.16. Given schedule \( t_u(j) \) over interval \([0, T)\) at node \( v \), Schedule\((t, v)\) will generate a schedule for each node \( u \) in \( p(v) \), and the recursive calls Schedule\((t, u)\) generate a schedule for each node \( u \) in \( P(v) \). Recall that \( [a] \) denotes the largest integer less than or equal to \( a \).
Chapter 3. Finite Production Rate

Figure 3.15: Scheduling Algorithm

Input : Assembly system $G(N, A)$ and integer-ratio lot sizes $Q_1, \ldots, Q_n$ for all facility in $N$.
Output : Production run starting time $t_u(1), t_u(2), \ldots$ for all facilities $u$ over the time interval $[0, T)$.

Step 1. For $\ell = 0, 1, \ldots, q$ do begin $Q^\ell_{\text{max}} \leftarrow \max_{j \in (u, 1)} Q_j$; $r_{\ell} \leftarrow Q^\ell_{\text{max}} / Q_1$ end;

$r(1) \leftarrow \text{lcm}\{r_1, \ldots, r_q\}$, using the Euclidean Algorithm.

(If $Q_1, \ldots, Q_n$ is a power-of-two lot size policy solution, then simply $r(1) \leftarrow \max\{r_1, \ldots, r_q\}$.)

Step 2. For $u = 0, 1, \ldots, n$ do $r(u) \leftarrow r(1)Q_1 / Q_u$.

Step 3. For $j = 0, 1, \ldots, r(1) - 1$ do $t_u(j) \leftarrow jQ_1 / \pi_0$;

Step 4. $\text{Schedule}(t, 1)$;

Step 5. Stop.

Figure 3.16: Procedure $\text{Schedule}(t, v)$

begin

If $p(v) = \emptyset$ then return

else for each $u \in p(v)$ do

begin

if $Q_u / Q_v \geq 1$ then for $j = 0, 1, \ldots, r(u) - 1$ do $t_u(j) \leftarrow t_v \left( \frac{Q_u}{Q_v} \right)^j$

else begin

$r \leftarrow \frac{Q_v}{Q_u}$;

for $j = 0, 1, \ldots, r(u) - 1$ do $t_u(j) \leftarrow t_v \left( \left\lfloor \frac{j}{r} \right\rfloor \right) + (j \mod r) \frac{Q_u}{\pi_v}$;

end;

$\text{Schedule}(t, u)$;

end;

end.
The following is a numerical example for an integer-ratio lot size policy with 6 facilities in series (See Figure 3.17).

**Figure 3.17:** Six facilities in series

![Diagram of six facilities in series]

Production rates $\pi_i$ and lot sizes $Q_i$ are given in Table 3.2. Recall that $T'_i = Q_i / \pi_i$ is the actual production time of facility $i$, $r(i) = r(1)(Q_1 / Q_i)$ is the number of production lots at facility $i$ over the interval $[0, T)$, where $T = r(1)(Q_1 / \pi_0)$ is the production cycle period. We have $\pi_0 = 1$, $Q_{\text{max}} = 12$, $r_1 = r(1) = 4$ and $T = 12$. Using the "Scheduling Algorithm", we get the results in Table 3.2. Note that $Q_2 = \max_j Q_j$, $Q_4 = \max_{j \in \{4,1\}} Q_j$, and $Q_5 = \max_{j \in \{5,6,1\}} Q_j$. Three graphs of path echelon inventories are shown in Figure 3.18 on page 115.

**Table 3.2:** Production starting instants and lot sizes

<table>
<thead>
<tr>
<th>$i$</th>
<th>$\pi_i$</th>
<th>$Q_i$</th>
<th>$T'_i$</th>
<th>$r(i)$</th>
<th>$t_i(j)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$j = 0$</td>
</tr>
<tr>
<td>1</td>
<td>1$\frac{1}{6}$</td>
<td>3</td>
<td>2$\frac{4}{7}$</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1$\frac{1}{3}$</td>
<td>6</td>
<td>4$\frac{1}{2}$</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>1$\frac{1}{2}$</td>
<td>1</td>
<td>2$\frac{2}{3}$</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1$\frac{1}{2}$</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>12</td>
<td>4</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>6</td>
<td>1$\frac{1}{3}$</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>
Figure 3.18: Echelon inventories $E_{(i,j)}^t$ for the example in section 3.9.2.
Chapter 4

Series Systems With Backlogging

4.1 Introduction

In this chapter we consider the inventory model IM(S,SP,NE,IF,B), i.e., the series inventory model with backlogging, or more specifically:

1. The production network $G(N, A)$ is a series system, i.e., each node has exactly one successor and one predecessor in the network $G(N, A)$. One node without successor is the only node with external demand. One node without predecessor is the only node with external supply. Let $N \triangleq \{1, 2, \ldots, n\}$ be the node set and such that the node without successor is node 1, $p(i) = i + 1$ if $p(i) \neq \emptyset$.

2. The setup cost function is separable: $K(S) = \sum_{i \in S} K_i$, $\forall S \subseteq N$.

3. The echelon holding cost rate $h_i = h_i' - h_{i+1}' \geq 0$ for each $i \in N$ is constant and non-negative. The backlogging cost rate $b_i \geq 0$ is also constant.

4. The production rate at any node is infinite.

5. The external demand is allowed to be backlogged.

For the inventory model IM(S,SP,NE,IF,NB), i.e., without backlogging for the external demand, Roundy [28] (1985) has shown in more general cases that a simple heuristic delivers a high quality solution, one that is guaranteed to be within 6% of the optimal. Further than this, without backlogging we would know that this worst case performance

However, apart from one paper by Mitchell [18] (1987) for the one-warehouse \( n \)-retailer backlogged problem no generalizations to include backlogging appear to have been made. The reasons to undertake such a generalization are twofold. Firstly, there is the obvious reason to continue to explore the extent to which these rather remarkable worst-case performance results remain valid. In addition to the generalizations listed above for which the results remain valid, the results fail to be true for many obvious generalizations such as to subadditive joint costs. Thus to more carefully delineate the boundaries of where the result is true or not is worthwhile in itself.

The second, and perhaps more important, reason to consider the backlog case is rather different. The real target in much inventory/production research is ultimately to learn useful properties about the case of stochastic demand. To consider the case of series systems under stochastic demand it is important that we are confident about our understanding of the deterministic case. As the stochastic case must include the risk of shortages, so ought the deterministic comparisons to include the shortage case as well.

The results of the paper are as follows. Section 4.2 derives the total holding and backlogging cost over a finite horizon for feasible policies and the best Integer Frequency Policies, which are essentially identical to the average integer ratio policies explored by Mitchell [18] (1987). Section 4.3 proves that the continuous relaxation of this problem is a lower bound on any feasible policy and the worst-case result follows directly from this. Section 4.4 illustrates that the backlog problem may be reduced to an equivalent no-backlog problem and presents an algorithm to solve this problem.
4.2 Best Integer Frequency Policies

As a route (and path) in a series system is uniquely defined by its two ending nodes \( i \) and \( j \), we may use \( \langle i, j \rangle \) to represent this path.

For the inventory model \( IM(S, SP, NE, IF, B) \), we know from Chapter 2 that given any replenishment policy \( P = (t_P, Q_P) \) over any finite horizon \([0, T)\), there exists an optimal replenishment policy \( P' = (t_{P'}, Q_{P'}) \) such that \( P' \) dominates \( P \) and satisfies the Nonpositive Inventory Ordering Property, the Latest Ordering Property, the Nonnegative Filling Property and the Route Nestedness Property. Therefore, without loss of generality, we consider only policies satisfying these properties.

Because of the nestedness property, we know that if node \( i \geq 2 \) orders at instant \( t \) then node \( j \in \{1, \cdots, i-1\} \) also orders at instant \( t \). Because of the Nonpositive Inventory Ordering Property, the inventory level of node \( i \) before instant \( t \) is zero. Considering the Nonnegative Filling Property and the nestedness property, no node orders when node 1 is backlogged. Therefore, in this time interval before instant \( t \), the inventory of node \( i \geq 2 \) is zero and their inventories equal the inventory of node 1. For simplicity of description, we say that node \( i \) is backlogged, if the echelon inventory of a node \( i \) is negative. Then the property described above may be rephrased as there is an time interval before the ordering instant \( t \) of node \( i \) in which node \( i \) and all its successors are backlogged. This property allows us to conveniently label the time epochs in \([0, T)\) by two indices. See Figure 4.1 on page 120 in this chapter. The first index refers to the node \( i \) and labelling is started with \( i = n \) and proceeds to \( i = 1 \). As upstream (larger \( i \)) nodes order less frequently than downstream (smaller \( i \)) nodes there are fewer backlogging periods to label for upstream node. The second index labels all backlogging periods of node \( i \). Thus in Figure 4.1, node \( n = 3 \) has two backlogging periods labelled \((3,1)\) and \((3,2)\). All the nodes \( i \) with \( i \leq 2 \) share these periods, which are labelled as \((i,1)\) and \((i,2)\) for all
Figure 4.1: The Actual Inventory and Echelon Inventory Level for Series System with backlogging

\[
\begin{array}{c}
I^1_1 \quad 3 \quad 2 \quad 1 \quad d_i \quad B_{11} \quad B_{12} \\
I^2_2 \\
E^2_2 \\
I^3_3 \\
E^3_3 \\
A_{31} \quad A_{32} \quad B_{31} \quad B_{32} \quad \eta^3_{31} \quad \eta^3_{32} \quad T
\end{array}
\]
Now label the rest of the backlogging periods of node 2 from (2,3) to (2,6). This continues until all periods of all nodes are labelled.

Let $A_{ij}$ (resp. $B_{ij}$) be the areas shown in Figure 4.1 and $\eta_{ij}'$ (resp. $\eta_{ij}''$) the base length (time interval) of triangle $A_{ij}$ (resp. $B_{ij}$). For $i \in \{1, \cdots, n-1\}$ and $j = 1, \cdots, m_{i+1}$, we have $B_{i+1,j} = B_{ij}$ or $\eta_{i+1,j}'' = \eta_{ij}''$. Refer to Figure 4.1. Without loss of generality, we may assume that $\lambda_r = 1$ for all $r \in R$ and $d_i = 2$, then $A_{ij} = \eta_{ij}^2$ and $B_{ij} = \eta_{ij}''^2$.

Let $m_i$ be the number of replenishment of node $i \in N$ over finite horizon $[0,T)$. Because of nestedness, $m_1 \geq m_2 \geq \cdots \geq m_n \geq m_{n+1} \triangleq 0$.

**Lemma 4.2.1**

For the inventory model $IM(S,SP,NE,IF,B)$, the total holding and backlogging cost for any feasible policy $P$ over the finite horizon $[0,T)$ is

$$C_{HP}(T) \triangleq \int_0^T H t \, dt = \sum_{i=1}^{n} \left\{ h_i \sum_{j=1}^{m_i} \eta_{ij}'^2 + (b_1 + h_1') \sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}''^2 \right\},$$

(4.1)

where $h_{i+1}' = 0$ by convention. The total setup cost over the finite horizon $[0,T)$ is

$$C_{SP}(T) = \sum_{i=1}^{n} m_i K_i,$$

where $K_i$ is the setup cost of product $i$. The total cost over the finite horizon $[0,T)$ is

$$C_P(T) = C_{SP}(T) + C_{HP}(T).$$

**Proof.** By Corollary 2.4.9,

$$C_{HP}(T) = \int_0^T H t \, dt$$

$$= \sum_{i \in N} h_i \int_0^T \left[ 2(\tau_{i,1}(t) - t) + I_{1,i}(t) \right] dt + (b_1 + h_1') \int_0^T I_{1,i} dt$$

$$= \sum_{i \in N} h_i \left[ \sum_{j=1}^{m_i} (A_{ij} - B_{ij}) \right] + (b_1 + h_1') \sum_{j=1}^{m_1} B_{1,j}$$

$$= \sum_{i \in N} h_i \left[ \sum_{j=1}^{m_i} (\eta_{ij}'^2 - \eta_{ij}''^2) \right] + (b_1 + h_1') \sum_{j=1}^{m_1} \eta_{1,j}''^2.$$
Chapter 4. Series Systems With Backlogging

\[= \sum_{i=1}^{n} h_i \sum_{j=1}^{m_i} \eta_{ij}^2 + (b_1 + h'_1 - h_1) \sum_{j=1}^{m_1} \eta_{1j}^2 - \sum_{i=2}^{n} h_i \sum_{j=1}^{m_i} \eta_{ij}^2.\]

Note that for \(i = 1, 2, \ldots, n - 1\), we have \(\eta_{i+1,j}^\prime = \eta_{i,j}^\prime\), therefore,

\[= (b_1 + h'_1) \sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}^2 + (b_1 + h'_{i+2}) \sum_{j=1}^{m_{i+1}} \eta_{i+1,j}^\prime.\]

By convention \(h'_{n+1} = 0\) and \(m_{n+1} = 0\), we have

\[= (b_1 + h'_1 - h_1) \sum_{j=1}^{m_1} \eta_{1j}^\prime - \sum_{i=2}^{n} h_i \sum_{j=1}^{m_i} \eta_{ij}^2 = \sum_{i=1}^{n} (b_1 + h'_{i+1}) \sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}^\prime,\]

and

\[
C_{HF}(T) = \sum_{i=1}^{n} h_i \sum_{j=1}^{m_i} \eta_{ij}^\prime + (b_1 + h'_{i+1}) \sum_{i=1}^{n} \sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}^\prime.
\]

It is obvious for the total setup cost. This completes the proof. \(\square\)

Now let's consider following special replenishment policies:

**An Integer Frequency Policy** is a nested replenishment policy where the number of replenishments of node \(i \in \{1, \ldots, n - 1\}\) in each cycle of node \(i + 1\) (which is the time interval between two consecutive replenishments of node \(i + 1\)), is a fixed integer \(\ell_i\), which is called the replenishment ratio.

Note that the total number of replenishments of node \(i \in N\) over the finite horizon \([0, T]\) is \(m_i\). Therefore, \(m_i / m_{i+1} = \ell_i\) and \(\prod_{j=i}^{n-1} \ell_j = \frac{m_i}{m_n}\) for \(i \in \{1, 2, \ldots, n - 1\}\). Denote the set of all Integer Frequency Policy over the finite horizon \([0, T]\) with replenishment numbers \(m_1, \ldots, m_n\) as \(A(m_1, \ldots, m_n)\).

**A Best Integer Frequency Policy** is an Integer Frequency Policy, which minimizes the holding and backlogging cost \(C_{HF}(T)\) among all the policies \(P \in A(m_1, \ldots, m_n)\).
over the finite horizon \([0, T)\). Denote this minimum holding and backlogging cost as \(C_{HA(m_1, \cdots, m_n)}(T)\) or \(C_{HA}(T)\), and minimum total cost as \(C_A(T) = \sum_{i=1}^{n} m_i K_i + C_{HA}(T)\).

Similarly, we may define

A **Power—of—two frequency policy** is an Integer Frequency Policy where all the replenishment ratios \(\ell_1, \cdots, \ell_{n-1}\) are power—of—two integers.

A **Best power—of—two frequency policy** is a power—of—two frequency policy which minimizes the holding and backlogging cost \(C_{HP}(T)\) among all the power—of—two frequency policies.

Given a replenishment policy \(P \in A(m_1, \cdots, m_n)\) over the finite horizon \([0, T)\), there are \(m_i\) replenishments of node \(i\) over \([0, T)\). As in each cycle of node \(i + 1\) there are \(\ell_i\) replenishments of node \(i\), we may partition the set of indices \(\{1, \cdots, m_i\}\) into \(m_{i+1}\) equal size sets \(L_{i1}, \cdots, L_{im_{i+1}}\), such that

\[
\begin{align*}
|L_{ij}| &= \ell_i, \\
\ell_i &\in L_{ij}, \\
\sum_{k \in L_{ij}} \eta'_{ik} + \sum_{k \in L_{ij}\{j\}} \eta''_{ik} &= \eta'_{i+1,j},
\end{align*}
\]

\(i = 1, \cdots, n - 1, \quad j = 1, \cdots, m_{i+1}.

Recall that

\[\eta''_{i+1,j} = \eta''_{ij}, \quad \text{for } i = 1, \cdots, n - 1, \quad j = 1, \cdots, m_{i+1}.

Now we can express \(C_{HA}(T)\) as follows:

\[
C_{HA}(T) = \min_{P \in A(m_1, \cdots, m_n)} C_{HP}(T) = \min \sum_{i=1}^{n} \left\{ h_i m_i \sum_{j=1}^{m_i} \eta'_{ij}^2 + (b_1 + h_{i+1}'') \sum_{j=m_{i+1}+1}^{m_i} \eta''_{ij}^2 \right\}, \quad (4.2a)
\]
Chapter 4. Series Systems With Backlogging

\[
\sum_{k \in L_{ij}} \eta'_{ik} \quad \sum_{k \in L_{ij} \setminus \{j\}} \eta''_{ik} = \eta'_{i+1,j}, \quad i = 1, \ldots, n-1, \quad j = 1, \ldots, m_{i+1},
\]

\[
\eta''_{i+1,j} = \eta''_{ij},
\]

\[
\sum_{k=1}^{m_{i}} (\eta'_{nk} + \eta''_{nk}) = T, \quad i = 1, \ldots, n, \quad j = 1, \ldots, m_{i},
\]

The following lemma is useful in solving the quadratic programming problems below.

**Lemma 4.2.2**

The optimal value of the problem

\[
f = \min_{x_1, \ldots, x_m \geq 0} \left\{ \sum_{i=1}^{m} a_i x_i'^2 + \sum_{i=1}^{m-1} b_i x_i'' \middle| \sum_{i=1}^{m} x_i' + \sum_{i=1}^{m-1} x_i'' = X' \right\}
\]

is

\[
f = \frac{X'^2}{D},
\]

where

\[
D = \sum_{i=1}^{m} \frac{1}{a_i} + \sum_{i=1}^{m-1} \frac{1}{b_i},
\]

and the optimal solution to the problem is given by:

\[
x_i' = \frac{1}{a_i} \frac{X'}{D}, \quad i = 1, 2, \ldots, m,
\]

\[
x_i'' = \frac{1}{b_i} \frac{X'}{D}, \quad i = 1, 2, \ldots, m-1.
\]

**Proof.** An easy computation.

In the following we will use recursive equations to derive an explicit expression for \(C_{HA}(T)\) for fixed \(\ell_1, \ldots, \ell_{n-1}\).
First we introduce the recursive relations for variable $D_i$, which depends on $D_{i-1}$ and $\ell_{i-1}$:

\[
\begin{align*}
D_1 & \triangleq \infty, \\
D_i & \triangleq \left( \frac{1}{D_{i-1} + h_{i-1}} + \frac{1}{b_1 + h_i'} \right) \ell_{i-1} - \frac{1}{b_1 + h_i'}, 
\end{align*}
\]  

for $i = 2, 3, \cdots, n$. (4.3b)

We define $C_{i+1}(\eta', \ell_1, \cdots, \ell_i)$ in terms of $D_i$:

\[
C_1(\cdot) = \frac{1}{D_1} = 0, \quad \text{for } i = 1, \cdots, n - 1. \quad (4.4b)
\]

\[
C_{i+1}(\eta', \ell_1, \cdots, \ell_i) = \min \left\{ \left( \frac{1}{D_i} + h_i \right) \sum_{j=1}^{\ell_i} \eta_j'^2 + (b_1 + h_{i+1}') \sum_{j=1}^{\ell_{i-1}} \eta_j'^2 \right\}, \quad \text{for } i = 1, \cdots, n - 1. \quad (4.4b)
\]

In fact, $C_{i+1}(\eta', \ell_1, \cdots, \ell_i)$ may be explained as the minimum total holding and backlogging cost of node 1 to $i$ inclusive over time interval $\eta'$, in which the echelon inventory level of "node $i + 1$" (not node $i$!) is positive.

By Lemma 4.2.2, the following lemma solves the quadratic programming problem of $C_i(\eta', \ell_1, \cdots, \ell_{i-1})$:

**Lemma 4.2.3**

The optimal value of problem (4.4b) is:

\[
C_{i+1}(\eta', \ell_1, \cdots, \ell_i) = \frac{1}{D_{i+1}} \eta'^2, \quad \text{for } i = 1, 2 \cdots, n - 1. \quad (4.5a)
\]
Chapter 4. Series Systems With Backlogging

and the optimal solution to problem (4.4) is:

\[ \eta'_j = \frac{1}{D_i + h_i D_{i+1}} \eta', \quad \text{for } j = 1, 2, \ldots, \ell_i, \] (4.5b)

\[ \eta''_j = \frac{1}{b_1 + h_{i+1}' D_{i+1}} \eta', \quad \text{for } j = 1, 2, \ldots, \ell_i - 1, \] (4.5c)

where \( D_i \) is defined in equation (4.3).

Proof. By Lemma 4.2.2, it is obvious. \( \square \)

Lemma 4.2.4

For \( s = 1, \ldots, n \), \( C_{HA}(T) \) may also be expressed as:

\[ C_{HA}(T) = \min \left\{ \sum_{j=1}^{m_s} \frac{1}{D_s} \eta_{sj}^2 + \sum_{i=s}^{n} \left\{ h_i \sum_{j=1}^{m_i} \eta_{ij}^2 + (b_1 + h_{i+1}') \sum_{j=m_{i+1}+1}^{m_i} \eta''_{ij} \right\} \right\}, \] (4.6a)

\[ \begin{aligned}
&\sum_{k \in L_{ij}} \eta'_{ik} + \sum_{k \in L_{ij} \setminus \{j\}} \eta''_{ik} = \eta'_{i+1,j}, &i = s, \ldots, n-1, \\
&\eta''_{i+1,j} = \eta''_{ij}, &j = 1, \ldots, m_{i+1}, \\
&\sum_{k=1}^{m_n} (\eta'_{nk} + \eta''_{nk}) = T, \\
&\eta'_{ij}, \eta''_{ij} \geq 0, &i = s, \ldots, n, \quad j = 1, \ldots, m_i.
\end{aligned} \] (4.6b)

and the optimal solution to problem (4.2) is:

\[ \eta'_{sk} = \frac{1}{D_s + h_s D_{s+1}} \eta'_{s+1,j}, \quad \text{for } k \in L_s, \] (4.7)

\[ \eta''_{sk} = \frac{1}{b_1 + h_{s+1}' D_{s+1}} \eta''_{s+1,j}, \quad \text{for } k \in L_s \setminus \{j\}, \]

where \( D_s \) is defined in equation (4.3).

Proof. For \( s = 1 \), two expressions (4.2) and (4.6) of \( C_{HA}(T) \) are exactly the same.
Now suppose (4.6) holds for $s(\leq n - 1)$. By definition of $L_{sk}$, we have

$$\sum_{j=1}^{m_s} \frac{1}{D_s} \eta'_{sj}^2 + h_s \sum_{j=1}^{m_s} \eta'_{sj}^2 + (b_1 + h'_{s+1}) \sum_{j=m_{s+1}+1}^{m_s} \eta''_{sj}^2$$

$$= \sum_{k=1}^{m_{s+1}} \left\{ \left( \frac{1}{D_s} + h_s \right) \sum_{j \in L_{sk}} \eta'_{sj}^2 + (b_1 + h'_{s+1}) \sum_{j \in L_{sk} \setminus \{k\}} \eta''_{sj}^2 \right\}$$

Therefore,

$$C_{HA}(T) = \min \sum_{j=1}^{m_s} \frac{1}{D_s} \eta'_{sj}^2 + \sum_{i=s}^{n} \left\{ h_i \sum_{j=1}^{m_i} \eta'_{ij}^2 + (b_1 + h'_{i+1}) \sum_{j=m_{i+1}+1}^{m_i} \eta''_{ij}^2 \right\}$$

s.t. (4.6b)

$$= \min \sum_{k=1}^{m_{s+1}} \left\{ \left( \frac{1}{D_s} + h_s \right) \sum_{j \in L_{sk}} \eta'_{sj}^2 + (b_1 + h'_{s+1}) \sum_{j \in L_{sk} \setminus \{k\}} \eta''_{sj}^2 \right\}$$

$$+ \min \sum_{j=1}^{m_{s+1}} \frac{1}{D_{s+1}} \eta'_{s+1,j}^2 + \sum_{i=s+1}^{n} \left\{ h_i \sum_{j=1}^{m_i} \eta'_{ij}^2 + (b_1 + h'_{i+1}) \sum_{j=m_{i+1}+1}^{m_i} \eta''_{ij}^2 \right\},$$

s.t. (4.6b)

For $k = 1, \ldots, m_{s+1}$, each term in the summation of the first part of the above formulation plus the relevant constraints in (4.6b) may be formulated as in problem (4.4):

$$C_{s+1}(\eta'_{s+1,k}, \ell_1, \ldots, \ell_s) = \min \sum_{j \in L_{sk}} \left( \frac{1}{D_s} + h_s \right) \eta'_{sj}^2 + (b_1 + h'_{s+1}) \sum_{j \in L_{sk} \setminus \{k\}} \eta''_{sj}^2$$

s.t. \quad \sum_{j \in L_{sk}} \eta'_{sj} + \sum_{j \in L_{sk} \setminus \{k\}} \eta''_{sj} = \eta'_{s+1,k}, \quad \eta''_{s+1,k} = \eta''_{sk}.$$

By Lemma 4.2.3, for $k = 1, \ldots, m_{s+1}$, the optimal value of the problem is

$$C_{s+1}(\eta'_{s+1,k}, \ell_1, \ldots, \ell_s) = \frac{1}{D_{s+1}} \eta'_{s+1,k}^2$$

and the optimal solution to the problem is:

$$\eta'_{sj} = \frac{1}{D_s + h_s} \eta'_{s+1,k}, \quad \text{for } j \in L_{sk},$$

$$\eta''_{sj} = \frac{1}{b_1 + h'_{s+1}} \eta'_{s+1,k}, \quad \text{for } j \in L_{sk} \setminus \{k\}. $$
Therefore, the problem becomes

\[ C_{HA}(T) = \min \sum_{j=1}^{m+1} \frac{1}{D_{s+1}} \eta'_{s+1,j}^2 + \sum_{i=s+1}^{n} \left\{ h_i \sum_{j=1}^{m_i} \eta''_{ij}^2 + (b_1 + h'_{i+1}) \sum_{j=m_i+1}^{m_i} \eta''_{ij}^2 \right\}, \]

s.t. \hspace{1cm} (4.6b) \hspace{1cm} \text{(change } s \text{ to } s+1) \]

That is, the lemma holds for \( s+1 \). By induction, the expression (4.6) holds for \( s = 1, \cdots, n \), and the optimal solution (4.7) holds for \( s = 1, \cdots, n - 1 \). \( \square \)

The following Corollary is an immediate result of Lemma 4.2.4

Corollary 4.2.5

The total holding and backlogging cost for the best Integer Frequency Policy over time interval \([0, T)\) is

\[ C_{HA}(T) = \frac{T^2}{D_H}, \]

where

\[ D_H \triangleq \left( \frac{1}{D_n} + \frac{1}{b_1} \right)^{-1} m_n, \]

and the optimal solution is given by

\[
\eta'_{nj} = \frac{1}{D_H} \left( \frac{T}{D_n + h_n} \right), \hspace{1cm} \eta''_{nj} = \frac{1}{b_1 D_H}, \]

for \( j = 1, 2, \cdots, m_n \).

Proof. Let \( s = n \) in Lemma 4.2.4, we have

\[ C_{HA}(T) = \min \sum_{j=1}^{m_n} \left\{ \left( \frac{1}{D_i} + h_n \right) \eta'_{nj}^2 + b_1 \eta''_{nj}^2 \right\}, \]

s.t. \hspace{1cm} \sum_{j=1}^{m_n} (\eta'_{nj} + \eta''_{nj}) = T, \]

\[ \eta'_{nj}, \eta''_{nj} \geq 0, \hspace{1cm} j = 1, \cdots, m_n. \]
By Lemma 4.2.2, the result is immediate.

To derive an explicit expression for $D_i$ is not easy; therefore we introduce a new variable $F_i$ which is a function of $D_{i+1}$ and easier to manipulate.

\[
F_0 \triangleq 0 \quad (4.11a)
\]
\[
F_i \triangleq \frac{1}{D_{i+1} + \frac{1}{b_1 + h'_{i+1}}}, \quad \text{if } i = 1, 2, \ldots, n - 1. \quad (4.11b)
\]

Let

\[
\alpha_i \triangleq \frac{b_1 + h'_{i+1}}{b_1 + h_i}, \quad \text{for } i = 1, 2, \ldots, n,
\]

where $h'_{n+1} = 0$. It is obvious that $\alpha_i \in (0, 1]$.

**Lemma 4.2.6**

(1) The following recursive relations holds for $F_i$:

\[
F_i = (h_i + \alpha_i F_{i-1}) \frac{\alpha_i}{\ell_i}, \quad \text{for } i = 1, 2, \ldots, n - 1. \quad (4.12)
\]

(2) In addition, for $i = 1, 2, \ldots, n - 1$:

\[
F_i = (b_1 + h'_{i+1})^2 \sum_{j=1}^{i} \left[ \left( \prod_{k=j}^{i} \frac{1}{\ell_k} \right) \frac{h_j}{(b_1 + h'_j)(b_1 + h'_{j+1})} \right]. \quad (4.13)
\]

**Proof.** (1) First note that

\[
\frac{1}{F_{i-1}} = D_i + \frac{1}{b_1 + h'_i},
\]

and

\[
D_i + \frac{1}{h_i} = \frac{1}{F_{i-1}} - \frac{1}{b_1 + h'_i} + \frac{1}{h_i} = \frac{1}{F_{i-1}} + \frac{b_1 + h'_{i+1}}{(b_1 + h'_i)h_i} = \frac{1}{F_{i-1}} + \frac{\alpha_i}{h_i}.
\]
Therefore, we have

\[
\frac{1}{D_i} + h_i + \frac{1}{b_1 + h'_{i+1}} = \frac{D_i(b_1 + h'_{i+1}) + 1 + h_iD_i}{(1 + h_iD_i)(b_1 + h'_{i+1})} = \frac{D_i(b_1 + h'_i) + 1}{(1 + h_iD_i)(b_1 + h'_i)}
\]

\[
= \frac{(D_i + \frac{1}{b_1 + h'_i})(b_1 + h'_i)}{(1 + h_iD_i)(b_1 + h'_i)}
\]

\[
= \frac{\frac{1}{F_{i-1}}}{\frac{h_i}{F_{i-1}} + \alpha_i} \frac{b + h'_{i+1}}{b + h'_i} = \frac{1}{(h_i + \alpha_iF_{i-1})\alpha_i},
\]

and

\[
\frac{1}{F_i} = D_{i+1} + \frac{1}{b_1 + h'_{i+1}} \quad \text{(by definition of } F_i \text{ in (4.11b))}
\]

\[
= \left(\frac{1}{D_i} + h_i + \frac{1}{b_1 + h'_{i+1}}\right) \ell_i \quad \text{(by definition of } D_i \text{ in (4.3b))}
\]

\[
= \frac{1}{(h_i + \alpha_iF_{i-1})\alpha_i} \ell_i.
\]

That is, equation (4.12) holds.

(2) For \(i = 1\), by equation (4.12), we have

\[
F_1 = h_1 \frac{\alpha_1}{\ell_1} = h_1 \frac{1}{\ell_1} \frac{b_1 + h'_1}{b_1 + h'_1} = \frac{1}{\ell_1} \frac{h_1}{(b_1 + h'_1)(b_1 + h'_2)},
\]

i.e., equation (4.13) holds.

Suppose equation (4.13) holds for \(i\). For \(i + 1\), we have

\[
F_{i+1} = (h_{i+1} + \alpha_{i+1}F_i)\frac{\alpha_{i+1}}{\ell_{i+1}} \quad \text{(by equation (4.12))}
\]

\[
= h_{i+1} \frac{b_1 + h'_{i+2}}{b_1 + h'_{i+1}} \frac{1}{\ell_{i+1}} + \left(\frac{b_1 + h'_{i+2}}{b_1 + h'_{i+1}}\right)^2 \frac{1}{\ell_{i+1}} \sum_{j=1}^{i} \left[\left(\prod_{k=j}^{i} \frac{1}{\ell_k}\right) \frac{h_j}{(b_1 + h'_j)(b_1 + h'_{j+1})}\right]
\]

(by induction)

\[
= (b_1 + h'_{i+2})^2 \frac{1}{\ell_{i+1}} \frac{h_{i+1}}{(b_1 + h'_{i+1})(b_1 + h'_{i+2})}
\]
\[+(b_1 + h_{i+2}')^2 \sum_{j=1}^{i} \left[ \left( \prod_{k=j}^{i+1} \frac{1}{\ell_k} \right) \frac{h_j}{(b_1 + h_j')(b_1 + h_{j+1}')} \right] = (b_1 + h_{i+2}')^2 \sum_{j=1}^{i+1} \left[ \left( \prod_{k=j}^{i+1} \frac{1}{\ell_k} \right) \frac{h_j}{(b_1 + h_j')(b_1 + h_{j+1}')} \right] \]
i.e., equation (4.13) holds for \(i + 1\).

**Corollary 4.2.7**

\[
\frac{1}{D_H} = b_1^2 \sum_{j=1}^{n} \frac{1}{m_j} \frac{h_j}{(b_1 + h_j')(b_1 + h_{j+1}')} \tag{4.14}
\]

(recall that \(h_{n+1}' = 0\)).

**Proof.** Note that \(h_n = h_n'\) and

\[
D_n + \frac{1}{h_n} = \frac{1}{F_{n-1}} - \frac{1}{b_1 + h_n} + \frac{1}{h_n} = \frac{1}{F_{n-1}} + \frac{b_1}{(b_1 + h_n)h_n}.
\]

We have, by equation (4.9),

\[
\frac{D_H}{m_n} = \frac{1}{D_n + h_n} + \frac{1}{b_1} = \frac{b_1 + \frac{1}{D_n} + h_n}{(\frac{1}{D_n} + h_n)b_1} = \frac{b_1 + h_n}{D_n} \frac{D_n + \frac{1}{b_1 + h_n}}{D_n + \frac{1}{h_n} \frac{b_1}{D_n}} = \frac{b_1 + h_n}{D_n} \frac{1}{F_{n-1} + \frac{b_1}{b_1 + h_n} \frac{1}{h_n} \frac{b_1}{h_n}} \frac{1}{b_1} = \frac{b_1 + h_n}{b_1} \frac{1}{h_n + \frac{b_1}{b_1 + h_n} F_{n-1}}
\]

Note that for \(i = n - 1\) equation (4.13) becomes

\[
F_{n-1} = (b_1 + h_n')^2 \sum_{j=1}^{n-1} \frac{m_j}{m_j} \frac{h_j}{(b_1 + h_j')(b_1 + h_{j+1}')}.
\]
\[
\frac{1}{D_H} = \frac{b_1}{b_1 + h_n} \left( h_n + \frac{b_1}{b_1 + h_n} F_{n-1} \right) \frac{1}{m_n} \\
= \frac{1}{m_n} \frac{b_1 h_n}{b_1 + h_n} + b_1^2 \sum_{j=1}^{n-1} \frac{1}{m_j (b_1 + h'_j)(b_1 + h'_{j+1})} \\
= b_1^2 \sum_{j=1}^{n} \frac{1}{m_j (b_1 + h'_j)(b_1 + h'_{j+1})}.
\]

Based on Corollary 4.2.5 and Corollary 4.2.7, the following corollary is immediate.

**Corollary 4.2.8**

Let the average replenishment interval length for node \( i \) be \( T_j = \frac{T}{m_j} \). Then the average holding and backlogging cost is

\[
\frac{1}{T} C_{HA}(T) = b_1^2 \sum_{j=1}^{n} \left[ \frac{h_j T_j}{(b_1 + h'_j)(b_1 + h'_{j+1})} \right],
\]

the average setup cost is

\[
\frac{1}{T} C_{SA}(T) = \sum_{j=1}^{n} \frac{K_j}{T_j},
\]

and the average cost is

\[
\frac{1}{T} C_A(T) = \sum_{i=1}^{n} \left\{ \frac{K_i}{T_i} + \frac{b_1^2 h_i T_i}{(b_1 + h'_i)(b_1 + h'_{i+1})} \right\}.
\]

Therefore, we have the following integer programming problem to get the optimal value of best Integer Frequency Policies:

**Lemma 4.2.9**

The minimum average cost of the series system for best Integer Frequency Policies is

\[
Z_B \triangleq \min \sum_{i=1}^{n} \left\{ \frac{K_i}{T_i} + \frac{b_1^2 h_i T_i}{(b_1 + h'_i)(b_1 + h'_{i+1})} \right\},
\]

s.t. \( T_n \geq T_{n-1} \geq \cdots \geq T_1 \geq 0, \)

\[
T_i/T_{i-1} = \text{an integer}.
\]
Its continuous relaxation is

\[
\tilde{Z}_B \triangleq \min \left\{ \sum_{i=1}^{n} \left( \frac{K_i}{T_i} + \frac{b_i h_i T_i}{(b_1 + h_i')(b_1 + h_{i+1}')} \right) \right\},
\]

s.t. \( T_n \geq T_{n-1} \geq \ldots \geq T_1 \geq 0. \) \hfill (4.17a)

This can be compared to the single facility case where

\[
\tilde{Z}_B = \min \left\{ \frac{K}{T} + \frac{b h T}{(b + h)} \right\},
\]

note that \( h_{n+1}' = 0 \) and \( h_i' = h_n. \)

4.3 Lower Bound Theorem

In this section, we show that the optimal solution \( \tilde{Z}_B \) to the continuous relaxation problem (4.17) is a lower bound on the average cost of all feasible policies over any finite horizon \([0, T)\).

First we derive a lower bound on the holding and backlogging cost of any feasible policy \( \mathcal{P}. \) Then we prove that this lower bound in fact has the same form as the holding and backlogging cost of best Integer Frequency Policies. Finally, the lower bound theorem follows immediately.

Lemma 4.3.1

We recursively define \( B_i \) and \( \overline{B}_i(x_i) \) as follows:

\[
B_0 \triangleq 0, \hfill (4.18a)
\]

\[
\begin{align*}
\overline{B}_i(x_i) & \triangleq x_i^2 \left( B_{i-1} + \frac{h_i}{m_i} \right) + \frac{b_i + h_{i+1}'}{m_i - m_{i+1}} (1 - x_i)^2, \\
B_i & \triangleq \min_{x_i \in [0, 1]} \overline{B}_i(x_i),
\end{align*}
\]

for \( i = 1, 2 \ldots, n. \) \hfill (4.18b)
Then for the inventory model IM(S,SP,NE,IF,B), the holding cost of any feasible policy \( P \) over the finite horizon \([0, T]\) satisfies the following inequality:

\[
C_{HP}(T) \geq T^2 B_n.
\]

Proof. By equation (4.1) in Lemma 4.2.1:

\[
C_{HP}(T) = \sum_{i=1}^{n} \left\{ h_i \sum_{j=1}^{m_i} \eta_{ij}^2 + (b_1 + h'_{i+1}) \sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}^2 \right\},
\]

where \( h'_{n+1} = 0 \) and \( m_{n+1} = 0 \).

Let

\[
\eta_i' = \sum_{j=1}^{m_i} \eta_{ij}', \quad \text{for } i = 1, 2, \ldots, n,
\]

i.e., \( \eta_i' \) is the total time during which the echelon inventory of node \( i \) is not in backlogging.

Then

\[
\sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}'' = \eta_{i+1}' - \eta_i', \quad \text{for } i = 1, 2, \ldots, n,
\]

where \( 0 \leq \eta_1' \leq \eta_2' \leq \cdots \leq \eta_n' \leq \eta_{n+1}' = T \).

Then it is obvious that

\[
\sum_{j=1}^{m_i} \eta_{ij}'^2 \geq \frac{\eta_i'^2}{m_i},
\]

\[
\sum_{j=m_{i+1}+1}^{m_i} \eta_{ij}''^2 \geq \frac{(\eta_{i+1}' - \eta_i')^2}{m_i - m_{i+1}}.
\]

Therefore,

\[
C_{HP}(T) \geq \sum_{i=1}^{n} \left[ h_i \frac{\eta_i'^2}{m_i} + (b_1 + h'_{i+1}) \frac{(\eta_{i+1}' - \eta_i')^2}{m_i - m_{i+1}} \right],
\]

where \( h'_{n+1} = 0 \) and \( m_{n+1} = 0 \).

Let recursively define function \( B_i(x_1, \ldots, x_i) \) as follows:

\[
B_0(\cdot) = 0,
\]
and for \( i = 1, \cdots, n, \)

\[
B_i(x_1, \cdots, x_i) = x_i^2 \left( B_{i-1}(x_1, \cdots, x_{i-1}) + \frac{h_i}{m_i} \right) + \frac{b_1 + h_{i+1}}{m_i - m_{i+1}} (1 - x_i)^2.
\]

Then by induction it is easy to verify that

\[
B_k(x_1, \cdots, x_k) = \sum_{i=1}^{k} \left( \prod_{j=i+1}^{k} x_j \right)^2 \left[ \frac{h_i}{m_i} x_i^2 + \frac{b_1 + h_{i+1}}{m_i - m_{i+1}} (1 - x_i)^2 \right],
\]

where \( \prod_{j=k+1}^{k} x_j \triangleq 1. \)

Let \( x_i = \frac{i}{\eta_{i+1}} \in [0, 1]. \) Then \( \prod_{j=i}^{n} x_j = \prod_{j=i}^{n} \frac{\eta_j}{\eta_{j+1}} = \frac{\eta_i}{\eta_{n+1}} \), and inequality (4.19) becomes

\[
\int_0^T H^n \, dt \geq T^2 \sum_{i=1}^{n} \left[ \frac{h_i}{m_i} \left( \prod_{j=i}^{n} x_j \right)^2 + \frac{b_1 + h_{i+1}}{m_i - m_{i+1}} \left( \prod_{j=i+1}^{n} x_j \right)^2 (1 - x_i)^2 \right]
= T^2 \sum_{i=1}^{n} \left( \prod_{j=i+1}^{n} x_j \right)^2 \left[ \frac{h_i}{m_i} x_i^2 + \frac{b_1 + h_{i+1}}{m_i - m_{i+1}} (1 - x_i)^2 \right]
= T^2 B_k(x_1, x_2, \cdots, x_n).
\]

By induction we prove

\[
B_i \leq B_i(x_1, \cdots, x_i), \quad \text{for all } x_j \in [0, 1]. \tag{4.20}
\]

For \( i = 0, B_0 = B_0(\cdot). \) Suppose inequality (4.20) holds for \( i, \) then by induction

\[
B_{i+1} \leq \bar{B}_i(x_{i+1})
= x_{i+1}^2 \left( B_i + \frac{h_{i+1}}{m_{i+1}} \right) + \frac{b_1 + h_{i+2}}{m_{i+1} - m_{i+2}} (1 - x_{i+1})^2
\leq x_{i+1}^2 \left( B_i(x_1, \cdots, x_i) + \frac{h_{i+1}}{m_{i+1}} \right) + \frac{b_1 + h_{i+2}}{m_{i+1} - m_{i+2}} (1 - x_{i+1})^2
= B_{i+1}(x_1, \cdots, x_{i+1}).
\]

This completes the proof.
In order to estimate $C_{HP}(T)$, we only need to derive a direct expression for $B_i$.

Lemma 4.3.2 below gives us a recursive formulation of $B_i$ based only on the previous value $B_{i-1}$.

**Lemma 4.3.2**

For $i = 1, 2, \cdots, n$, $B_i$ satisfies the following recursive relationship:

$$
\frac{1}{B_i} = \left( \frac{1}{m_i B_{i-1} + h_i} + \frac{1}{b_1 + h'_{i+1}} \right) m_i = \frac{m_{i+1}}{b_1 + h'_{i+1}},
$$

(4.21)

**Proof.** By differentiation, we have

$$
\frac{d \overline{B}_i(x_i)}{d x_i} = \left( B_{i-1} + \frac{h_i}{m_i} \right) 2x_i - \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}} 2(1 - x_i) = 0.
$$

The solution $\bar{x}_i$ to this equation satisfies

$$
\left( B_{i-1} + \frac{h_i}{m_i} \right) \bar{x}_i = \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}}(1 - \bar{x}_i),
$$

(4.22)

and

$$
\left( B_{i-1} + \frac{h_i}{m_i} + \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}} \right) \bar{x}_i = \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}}.
$$

(4.23)

From equation (4.23), it is obvious that $\bar{x}_i \in [0,1]$, i.e., $\bar{x}_i$ is the optimal solution to $\overline{B}(x_i)$. Therefore,

$$
B_i = \overline{B}(\bar{x}_i)
$$

$$
= \bar{x}_i^2 \left( B_{i-1} + \frac{h_i}{m_i} \right) + \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}}(1 - \bar{x}_i)^2
$$

$$
= \bar{x}_i^2 \left( B_{i-1} + \frac{h_i}{m_i} \right) + \left( B_{i-1} + \frac{h_i}{m_i} \right) \bar{x}_i(1 - \bar{x}_i) \quad \text{(by equation (4.22))}
$$

$$
= \bar{x}_i \left( B_{i-1} + \frac{h_i}{m_i} \right)
$$

$$
= \frac{\left( B_{i-1} + \frac{h_i}{m_i} \right) \left( \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}} \right)}{B_{i-1} + \frac{h_i}{m_i} + \frac{b_1 + h'_{i+1}}{m_i - m_{i+1}}} \quad \text{(by equation (4.22))}
$$
Finally, we have
\[
\frac{1}{B_i} = \frac{1}{B_{i-1} + h_i/m_i} + \frac{m_i - m_{i+1}}{b_1 + h'_{i+1}} \\
= \left( \frac{1}{m_i B_{i-1} + h_i} + \frac{1}{b_1 + h'_{i+1}} \right) m_i - \frac{m_{i+1}}{b_1 + h'_{i+1}}.
\]
This completes the proof. \qed

**Lemma 4.3.3**

\[
B_n = b_1^2 \sum_{j=1}^{n} \frac{h_j}{m_j (b_1 + h'_j)(b_1 + h'_{j+1})}
\]
(recall that \(h'_{n+1} = 0\)).

**Proof.** Let
\[
D_1' \triangleq \frac{1}{B_0}, \\
D_i' = \frac{1}{m_i B_{i-1}}, \quad \text{for } i = 2, \ldots, n, \\
\ell_i' \triangleq \frac{m_i}{m_{i+1}}, \quad \text{for } i = 1, \ldots, n, \\
D_H' \triangleq \frac{1}{B_n}.
\]
Note that \(\ell_i'\) is not necessarily an integer. By Lemma 4.3.2, we have
\[
D_1' = \infty, \\
D_i' = \left( \frac{1}{D_{i-1}'} + \frac{1}{b_1 + h'_i} \right) \ell_{i-1}' - \frac{1}{b_1 + h'_i}, \quad \text{for } i = 2, 3, \ldots, n, \\
D_H' = \left( \frac{1}{D_n'} + \frac{1}{b_1} \right) m_n.
\]
Comparing $D'_i$, $D'_{H}$ with the definition of $D_i$, $D_{H}$, it is obvious that $D'_{H} = D_{H}$. Note that the integrality of $\ell_{i}$ is not required in the proof of Corollary 4.2.7. Therefore, $D'_{H}$ has the same expression as $D_{H}$ even for noninteger $\ell_{i}$.

Based on the previous lemma, the following lower bound theorem follows directly.

**Theorem 4.3.4 (Lower Bound Theorem)**

*For the inventory model IM(S,SP,NE,IF,B), the long-run average cost $\bar{C}_P$ of any feasible policy $P$ satisfies*

$$\bar{C}_P \geq \bar{Z}_B.$$  

*That is, $\bar{Z}_B$ is a lower bound on all feasible policies.*

**Proof.** By Lemma 4.3.1 and Lemma 4.3.3, the total cost $C_P(T)$ of any feasible policy $P$ over the finite horizon $[0, T)$ is bounded below by

$$C_P(T) = \sum_{i=1}^{n} m_i K_i + C_{HP}(T) \geq \sum_{i=1}^{n} m_i K_i + T^2 B_n = \sum_{i=1}^{n} m_i K_i + T^2 b_i^2 \sum_{i=1}^{n} \frac{1}{m_i (b_1 + h'_i)(b_1 + h'_{i+1})}.$$  

Recall that $T_i = T/m_i$ is the average replenishment interval length for $i \in N$ and we have

$$\frac{1}{T} C_P(T) \geq \sum_{i=1}^{n} \left\{ \frac{K_i}{T_i} + \frac{b_i^2 h_i T_i}{(b_1 + h'_i)(b_1 + h'_{i+1})} \right\}.$$  

Because $m_1 \geq m_2 \geq \cdots \geq m_n$, we have $T_n \geq T_{n-1} \geq \cdots \geq T_1 \geq 0$. By the definition of $\bar{Z}_B$, we get $\frac{1}{T} C_P(T) \geq \bar{Z}_B$, and this completes the proof.
4.4 Reduction to an Equivalent No-Backlog Problem and Scheduling Algorithm

For an instance of the inventory model IM(S,SP,NE,IF,B), we can easily define a corresponding instance of the inventory model IM(S,SP,NE,IF,NB) as follows. All the parameters of two instances are exactly the same except the holding and backlogging cost rate. Suppose the holding cost rate of node $i \in N$ for the instance of the inventory model IM(S,SP,NE,IF,B) is $h_i'$ and the backlogging cost rate of node 1 is $b_1$. The actual holding cost of node $i \in N$ for the instance of the inventory model IM(S,SP,NE,IF,NB) is defined as $\bar{h}_i = \frac{b_1 h_i'}{b_1 + h_i'}$ and there is no backlogging at node 1. Then it is easy to verify that the echelon holding cost rate $\bar{h}_i$ satisfies

$$\bar{h}_i = \bar{h}_i' - \bar{h}_{i+1}' = \frac{b_1^2 h_i}{(b_1 + h_i')(b_1 + h_{i+1}')} \geq 0.$$ 

We can formulate this problem by power-of-two policies as in Roundy [27] (1983) who solved this no-backlog problem by the minimum violating algorithm in $O(n \log n)$ time. The formulation of this problem for power-of-two policies is

$$\min \sum_{i=1}^{n} \left\{ \frac{K_i}{T_i} + \bar{h}_iT_i \right\},$$

s.t. $T_n \geq T_{n-1} \geq \cdots \geq T_1 \geq 0$,

$$T_i/T_{i-1} = \text{power-of-two integer},$$

which is the same as the equivalent backlogging problem. Therefore they have the same average cost and the same lower bound $\bar{Z}_B$. Thus, the results of the no-backlogging problem can be easily applied to the corresponding backlogging problem. We state only two consequences of Roundy's results here.

**Theorem 4.4.1 (94% Effectiveness of Power-of-Two Frequency Policies)**

For each fixed base period $T_B$, there is an optimal best power-of-two frequency policy $P$
with the cycle time \( T_n = 2^n T_B \) of product \( n \) for some integer \( q_n \) and the average cost \( \bar{C}_P \) of policy \( P \) satisfies \( \bar{C}_P \geq 94\% \bar{Z}_B \) for the inventory model IM(S,SP,NE,IF,B). Moreover, the optimal best power-of-two frequency policy \( P \) can be found in \( O(n \log n) \) time.

If allowing the base period \( T_B \) varies, we have a better estimate.

**Theorem 4.4.2 (98% Effectiveness of Power-of-two Frequency Policies)**

There is an optimal best power-of-two frequency policy \( P \) whose average cost \( \bar{C}_P \) satisfies \( \bar{C}_P \geq 98\% \bar{Z}_B \) for the inventory model IM(S,SP,NE,IF,B). Moreover, the optimal best power-of-two frequency policy \( P \) can be found in \( O(n \log n) \) time.

In the following, we illustrate how we can get an optimal best power-of-two frequency policy \( P \), if the optimal average replenishment intervals are \( T_1^*, T_2^*, \ldots, T_n^* \) and \( \bar{Z}_B = C_A(T^*) \) are the solution to the continuous relaxation (4.17):

Consider a power-of-two policy \( P \) with a base period \( T_B \) and the optimal average power-of-two interval of node \( i \in N \) is \( T_i = 2^{q_i} T_B \), where \( q_i = \left\lfloor \log_2 \frac{T_i}{T_B} - \frac{1}{2} \right\rfloor \). Let \( m_i = 2^{n-q_i} \) for \( i = 1, \ldots, n \) and \( \ell_i = \frac{m_i}{m_{i+1}} = 2^{a+1-q_i} \) for \( i = 1, \ldots, n-1 \).

Let \( T = T_n \) be the cycle time of product \( n \) and

\[

\begin{align*}
\eta_{ij} &= \zeta_i' \quad \text{for } j = 1, \ldots, m_i, \\
\eta_{ij} &= \zeta_i'' \quad \text{for } j = m_{i+1}, \ldots, m_i.
\end{align*}
\]

where \( \zeta_i' \) and \( \zeta_i'' \) are recursively (from \( n \) to \( 1 \)) given by following equations:

\[
\begin{align*}
\xi_n' &= \frac{1}{D_n + h_n} \frac{T}{D_H}, \\
\xi_n'' &= \frac{1}{b_1} \frac{T}{D_H}, \\
\xi_i' &= \frac{1}{D_i-1 + h_{i-1}} \xi_{i+1}', \\
\xi_i'' &= \frac{1}{b_1 + h_i'} \xi_{i+1}' \quad \text{for } i = n-1, \ldots, 2, 1,
\end{align*}
\]
where $D_i$ (for $i = 1, \ldots, n$) is defined in equations (4.3) and $D_H$ is defined in equations (4.9).

Let $\xi_i = \xi'_i + \xi''_i$, then the schedule for product $i \in N$ is to order products at instants $t_i(k_i, \ldots, k_{n-1}, k_n) \triangleq \left\{ T \times k_n + \sum_{j=i}^{n-1} \xi_j \times k_j \right\}$ such that the echelon inventory level of product $i$ reaches $2\xi'_i$, where $k_j = 0, 1, \cdots, \ell_j - 1$ for $j = 1, \cdots, n - 1$ and $k_n$ is any non-negative integer. Alternatively, the order quantities of node $i$ at instant $t_i(k_i, \cdots, k_{n-1}, k_n)$ is

$$Q_i(k_i, \cdots, k_n) = \begin{cases} 2\xi'_i + 2\xi''_{j+1}, & \text{if } k_i = \cdots = k_j = 0, k_{j+1} > 0, \text{ for } i \leq n - 1, \\ 2T, & \text{for } i = n. \end{cases}$$

See the example in Figure 4.2.

### 4.5 Conclusions

In this chapter, we derive a best power-of-two frequency policy for the series systems with backlogging. This policy is guaranteed to be within 6% of the optimal, and can be found in $O(n \log n)$ time. We also present an algorithm to show how to get the replenishment schedule for the best power-of-two frequency policy in linear time to calculate all the parameters.
Figure 4.2: The Power-of-Two Frequency Policy

\[ \begin{align*}
&\ 3 \rightarrow 2 \rightarrow 1 \rightarrow d_1 \\
&\ell_1 = 4, \ell_2 = 2
\end{align*} \]

\[ \begin{align*}
&k_1 = 0 \ 1 \ 2 \ 3 \ 0 \\
&\xi_1' \xi_1'' \xi_2' \xi_2'' \xi_3' \xi_3''
\end{align*} \]

\[ \begin{align*}
&k_2 = 0 \ 1 \ 0 \\
&E_2^x
\end{align*} \]

\[ \begin{align*}
&k_3 = 0 \ 1 \\
&E_3^x
\end{align*} \]
Chapter 5

The Performance Ratio of Grouping Policies for the Joint Replenishment Problem

5.1 Introduction

We consider the inventory model IM(I,SM,CT,IF,NB), i.e., a multi-product extension of the traditional Economic Order Quantity (EOQ) model where there is a cost incentive for simultaneous replenishment of several products. There are \( n \) products, and \( N = \{1, \ldots, n\} \) denotes the set of products. Except for the setup costs (see below), the products are independent: there are no joint or dependent demands, no substitution opportunities, and products in \( N \) are not being used in making other products in \( N \). As in the EOQ model, we make the following assumptions:

1. We consider a continuous time, infinite horizon model, with stationary data (demand and cost rates) and no discounting. As a result, we focus on minimizing the long run average cost per time unit, while satisfying demands for all products.

2. The demand for each product is deterministic and occurs at a constant rate. By rescaling the units for each product, we are assuming that the demand rate is of two units per time unit, for each product.

3. The demand for each product is satisfied by continuously withdrawing from the inventory of that product. No shortages or backlogs are allowed. The inventories are replenished at times and in quantities to be determined. Replenishment of each product is instantaneous and lead times can be assumed to be zero, w.l.o.g. (without loss of
4. The total cost is the sum of all holding costs and setup costs.

5. The holding cost for product $i$ accumulates at a constant rate $h_i$ dollars per unit of product and per unit time. For each product $i$ ($i = 1, \ldots, n$), rate $h_i$ is a positive real number.

6. At each replenishment, a positive setup cost $K(S)$ is incurred, depending only on the set $S \subseteq N$ of products being replenished.

In the traditional $EOQ$ model, we have $K(S) = \sum_{i \in S} k_i$, where the $k_i$'s are given separate setup costs. In this case, there is no incentive for joint replenishment, and an optimal policy has each product being independently replenished according to the familiar $EOQ$ (or square root, or Wilson's) formula.

We depart from the traditional $EOQ$ model by allowing the joint setup cost $K(S)$ to be less than the sum of the separate setup costs of the products in $S$, therefore making joint replenishment costwise attractive. The model for joint setup costs we use here is that of a submodular setup cost introduced by Queyranne [24] (1985). We assume the set function $K : 2^N \to R_+$ satisfies the following conditions:

1. $K(\emptyset) = 0$;
2. $K(S) > 0$, for all $S \subseteq N, S \neq \emptyset$;
3. $K(S) \leq K(T)$, whenever $S \subseteq T$;
4. $K(S \cup T) \leq K(S) + K(T) - K(S \cap T)$, for all $S, T$.

Conditions 1 and 2 are necessary for the model to be meaningful (otherwise optimal total costs for any finite period may go to $\pm \infty$). The nondecreasing property of condition 3 may be assumed w.l.o.g. (otherwise, set $S$ is never used in an optimal solution, and function $K$ can then be redefined so as to satisfy condition 3). The submodularity property of condition 4 is fairly general, and allows the derivation of very tight bounds on the optimal cost. For example, the most popular joint setup cost function, defined by
Chapter 5. Grouping Policies

\[ K(S) \triangleq k_0 + \sum_{i \in S} k_i \] and sometimes called the major/minor, or quasi-linear, or modular, setup cost function, is submodular. We refer to Goyal and Şatir [10] (1989) for a survey on models using this function. The “family model” of Roundy [30] (1986) and the example in Rosenblatt and Kaspi [26] (1985) are also submodular. We refer to Queyranne [24] (1985) and Zheng [39] (1987) for further discussion of submodular setup costs. Section 2 below recalls how very tight bounds on the optimal cost may be obtained for submodular setup costs.

A replenishment policy is a specification of all replenishment epochs and quantities, for all products, over an infinite horizon. A feasible policy is a replenishment policy whereby all demands are satisfied, that is, inventory levels are never negative. The joint replenishment problem is to find a feasible policy with lowest possible long run average total cost per time unit. The only known fact about optimal policies is that it must satisfy the following zero inventory property: the inventory of every product drops to zero just before any replenishment of that product. This property implies that replenishment quantities are directly implied by replenishment times (and conversely).

Much attention has focused on stationary policies, that is, policies where each product is replenished at constant time intervals (and therefore, by the zero-inventory property, in constant quantities). Nonstationary policies may have lower cost than any stationary policy (Andres and Emmons, [1] (1976)). For submodular setup costs, however, it is known that the cost of a particular stationary policy (a “power-of-two” policy) cannot be more than 2% above the cost of any feasible policy (see Section 5.2).

One class of stationary policies that has been considered in the literature, is that of grouping policies (or fixed partition policies), whereby the set \( N \) of products is partitioned into groups of products, and all items in the same group are always jointly
replenished. Each group is then considered as a single “aggregate product” being replenished independently of the other groups, and therefore according to the EOQ formula. As a result, possible savings when several groups are simultaneously replenished are simply ignored. Grouping policies are considered by Chakravarty et al. [4] (1982), [5] (1985), by Rosenblatt and Kaspi [26] (1985) and by Queyranne [25] (1987).

An apparent advantage of grouping policies is that an optimal grouping may be relatively straightforward to compute. Chakravarty et al. show that, for special types of setup costs, the products can be indexed such that an optimal grouping is consecutive, that is, each group contains only products with consecutive indices. They also provide an $O(n^2)$ shortest path algorithm for finding a corresponding optimal grouping policy. Rosenblatt and Kaspi propose to find an optimal grouping by Dynamic Programming, for an arbitrary (not necessarily submodular) setup cost function. A correct Dynamic Programming algorithm runs in $O(3^n)$, Queyranne [25] (1987). This might be acceptable when $n$ is fairly small and the setup cost function is complicated, rendering other approaches (see Section 5.2) more cumbersome. Another apparent advantage of grouping policies is that they may be very easy to implement in practice, by permanently “tying together” products in a same group.

In terms of cost, unfortunately, a best grouping policy can be somewhat worse than some other good feasible policy. An example in Zheng [39] (1987) shows that the cost of a best grouping policy can be worse than 20% above the cost of another feasible policy. The objective of this chapter is to study, for submodular setup costs, how bad can best grouping policies be in the worst case. We use submodular costs here because, as mentioned above, they provide a fairly general model and a very tight estimate on the optimal cost of any feasible policy is available for such setup cost functions.

The organization of the chapter is as follows. In Section 5.2, we recall the properties of another class of stationary policies, namely, the integer ratio policies, which allow the
derivation of a very tight lower bound on the optimal cost of a feasible policy. This lower bound will be used in the definition of the performance ratio for grouping policies. A special class of stationary policies, the power-of-$\beta$ policies, is also discussed, and will be used later for deriving an upper bound on that performance ratio. Section 5.3 defines grouping policies and their performance ratio. Three lemmas in this section reduce the space of problem instances that has to be searched for determining the worst case performance ratio of grouping policies. It is interesting to note that the resulting instances are such that the consecutiveness property holds for an optimal grouping (Lemma 5.3.3), which can therefore be found by an $O(n^2)$ shortest path algorithm. The last lemma in this section presents a useful transformation of variables for estimating the performance ratio. Section 5.4 gives an upper bound on the worst case performance ratio of grouping policies, which is the main results of the chapter. Bad examples in Section 5.5 give us a lower bound on the worst case performance ratio of grouping policies. Section 5.7.1 collects the proofs of lemmas in the text. Section 5.7.2 includes all the main notation in the chapter for reference. Section 5.7.3 is the table of performance ratios of power-of-$\beta$ policies.

5.2 Lower Bound on the Average Cost of All Feasible Policies

Besides grouping policies, another class of stationary policies has been widely studied, that is, base period policies, whereby all products are replenished at constant intervals ("cycles") which are integral multiples of a common base period. Base period policies are surveyed by Goyal and Şatir [10] (1989) for the case of major/minor setup costs. Integer ratio policies are base period policies where, for any two products, the cycle of one product is an integer multiple of the cycle of the other one. A special class of
integer ratio policies is that of power-of-two policies, where each interval is a power-of-two times the base period. An extension to power-of-\(\beta\) policies, where \(\beta\) is an arbitrary integer greater than one, is discussed below. Power-of-two policies were introduced for joint replenishment problems by Jackson et al. [13] (1985) and Roundy [28] (1985). One of Roundy's major contributions was to show that finding a best integer ratio policies also yields, as a by-product, a lower bound on the cost of any feasible policy.

Power-of-two policies, and the corresponding lower bound result, have also been extended by Roundy [30] (1986) to his "family model" of setup costs, and by Queyranne [24] (1985) and Zheng [39] (1987) to general submodular setup cost functions. For all these cases, the lower bound is also a very tight estimate (within about 2%) of the cost of a feasible (power-of-two) policy. As mentioned in the Section 5.1, this very tight bound is the main tool used here for assessing the performance ratio of grouping heuristics.

We now introduce the requisite notations and definitions. Let

\[ R_+ \triangleq (0, \infty) \] be the set of positive real numbers;

\[ R^n_+ \triangleq \prod_{i=1}^{n} R_+ \] be the set of \(n\)-dimensional positive real numbers;

**power-of-\(\beta\) policy** \(t\) be a sequence \(t = (t_1, t_2, \ldots, t_n) \in R^n_+\) with the following three properties, see Roundy [28] (1985):

1. Orders for product \(i\) are placed once every \(t_i > 0\) units of time, beginning at time zero;
2. \(t_i = \beta^{m_i} \beta_0, m_i \in Z, \forall i \in N\) and for some \(\beta_0 \in [1/\sqrt{\beta}, \sqrt{\beta})\), where \(Z \triangleq \{0, \pm1, \pm2, \ldots\}\) is the set of integers, and \(N = \{1, 2, \ldots, n\}\) is the set of products;
3. The zero-inventory property holds, i.e., an order is placed for a product only when the inventory of that product is zero.
For a joint replenishment inventory model, we can easily derive a formulation of the average cost for a given power-of-\( \beta \) policy, see Queyranne [24] (1985) and Zheng [39] (1987) for details. Power-of-\( \beta \) policy \( t \) induces a permutation \( \alpha = (\alpha_1, \alpha_2, \ldots, \alpha_n) \) of the indices in \( N \) such that

\[
t_{\alpha_1} \leq t_{\alpha_2} \leq \cdots \leq t_{\alpha_n}.
\]

Let

\[
U_i \triangleq \{\alpha_1, \alpha_2, \cdots, \alpha_i\}, \quad \forall i \in N
\]

be the sets associated with permutation \( \alpha \). (Ties are broken arbitrarily). Observe that under a power-of-\( \beta \) policy \( t \), whenever product \( \alpha_i \) is replenished, all the products in set \( U_{i-1} \) are also replenished. The average set-up cost for power-of-\( \beta \) policy \( t \) is

\[
K[t] = \sum_{i=1}^{n} K(U_i) \left( \frac{1}{t_{\alpha_i}} - \frac{1}{t_{\alpha_{i+1}}} \right) = \sum_{i=1}^{n} \frac{K(U_i) - K(U_{i-1})}{t_{\alpha_i}},
\]

where \( t_{\alpha_{n+1}} = \infty \), \( U_0 = \emptyset \) and \( K(\emptyset) = 0 \).

The optimal average cost for power-of-\( \beta \) policies (for fixed \( \beta \) and \( \beta_0 \)) is given by a non-linear integer programming problem:

\[
(JR)_\beta \left\{ \begin{array}{l} 
C_{\beta}(n, K, h, \beta_0) \triangleq \min_{t \geq 0} \sum_{i=1}^{n} \left[ \frac{K(U_i) - K(U_{i-1})}{t_{\alpha_i}} + h_{\alpha_i} t_{\alpha_i} \right], \\
\text{s.t.} \quad t_i = \beta^{m_i} \beta_0, \quad m_i \in \mathbb{Z}, \quad \forall i \in N,
\end{array} \right.
\]

where \( \alpha \) satisfies (5.1a) and the \( U_i \)'s are defined by (5.1b).

The following non-linear programming problem \((RJR)\), independent of \( \beta \) and \( \beta_0 \), is a continuous relaxation of \((JR)_\beta\).

\[
(RJR) : \quad LB(n, K, h) \triangleq \min_{t \geq 0} \sum_{i=1}^{n} \left[ \frac{K(U_i) - K(U_{i-1})}{t_{\alpha_i}} + h_{\alpha_i} t_{\alpha_i} \right],
\]

where \( \alpha \) satisfies (5.1a) and the \( U_i \)'s are defined by (5.1b).
Sequence \((S_1, S_2, \ldots, S_q)\) of subsets of \(N\) is a **nested path** in \(N\), if \(\emptyset = S_0 \subset S_1 \subset S_2 \subset \cdots \subset S_q = N\). Note that the inclusions are proper in this definition. Note also that \((U_1, U_2, \ldots, U_n)\) defined above forms a nested path in \(N\). Let

\[
S^n \triangleq \begin{cases} 
K : 2^N \mapsto R_+ & \begin{align*}
K(\emptyset) &= 0; \\
K(S) &= 0, \text{ and } K(N) > 0 \text{ (non-negativity)}; \\
K(S) &\leq K(T), \text{ if } S \subseteq T \text{ (non-decreasing)}; \\
K(S \cap T) + K(S \cup T) &\leq K(S) + K(T), \\
\forall S, T \subseteq N \text{ (submodularity)}
\end{align*}
\end{cases}
\]

denotes the set of submodular set functions on \(N\).

The following characterization theorem in Zheng [39, Theorem 4.5] (1987) solves \((RJR)\).

**Theorem 5.2.1 (Optimal Solution to \((RJR)\))**

Assume that \(K \in S^n\) and the components of \(t = (t_1, t_2, \ldots, t_n)\) take on \(q\) distinct values \(t(1) < t(2) < \cdots < t(q)\), \(q \leq n\), and \((S_1, S_2, \ldots, S_q)\) is a nested path in \(N\) with \(S_\ell \backslash S_{\ell-1} \triangleq \{ i \in N : t_i = t(\ell) \}\). Then \(t\) is optimal for \((RJR)\) iff the following two conditions hold for each \(\ell = 1, 2, \ldots, q\):

1. \(t(\ell) = \sqrt{[K(S_\ell) - K(S_{\ell-1})]/h(S_\ell \backslash S_{\ell-1})},\)
   where \(h(S) \triangleq \sum_{i \in S} h_i, \) for any set \(S \in N\).
2. \(\sqrt{[K(S) - K(S_{\ell-1})]/h(S \backslash S_{\ell-1})} \geq t(\ell), \quad \forall S, \text{ such that } S_{\ell-1} \subseteq S \subseteq S_\ell.\)

Besides, the optimal value of \((RJR)\) is

\[
LB(n, K, h) = 2 \sum_{\ell=1}^{q} \sqrt{[K(S_\ell) - K(S_{\ell-1})]/h(S_\ell \backslash S_{\ell-1})}.
\]

We introduce two sets of problem instances related to \((K, h)\). They will be used in the sequel:

\[
\mathcal{M}^n \triangleq \left\{ K : 2^N \rightarrow R_+ \mid 0 < K_1 \leq K_2 \leq \cdots \leq K_n, \text{(non-decreasing)} \right\}
\]

\[
\mathcal{M}^n \triangleq \left\{ K : 2^N \rightarrow R_+ \mid K(S) = \max_{i \in S} K_i, \forall S \subseteq N, \text{(maximum)} \right\}
\]

denotes the set of ‘maximum’ submodular set functions on \(N\). Note that \(K_i \triangleq K(\{i\}) \forall i \in N\).

\[
\Omega^n \triangleq \left\{ (K, h) \in (\mathcal{M}^n, R_+) \left| \sqrt{\frac{K_1}{h_1}} < \sqrt{\frac{K_2 - K_1}{h_2}} < \cdots < \sqrt{\frac{K_n - K_{n-1}}{h_n}} \right. \right\}
\]

denotes the set of ‘monotone path’ instances. Note the strict inequalities in the definition.

The following corollary is an immediate result of Theorem 5.2.1.

**Corollary 5.2.2 (Optimal Solution to (RJR) for monotone paths)**

If \((K, h) \in \Omega^n\), then \(LB(n, K, h) = 2 \sum_{i=1}^{n} \sqrt{(K_i - K_{i-1})h_i}\) is the optimal value to problem (RJR).

The Lower Bound Theorem, Roundy [28] (1985), is a fundamental result for the analysis of heuristics for the joint replenishment problem:

**Theorem 5.2.3 (Lower Bound Theorem)**

\(LB(n, K, h)\) is a lower bound on the average cost of any feasible policy over any finite horizon.

A worst case performance ratio for a class of solutions to a problem is defined as the supremum, over all instances, of the ratio of the cost of the best policy in that class, to the optimum cost.
Chapter 5. Grouping Policies

The following lemma is a direct extension of a rounding lemma in Roundy [28] (1985). There, for $x \in R_+$, $e(x) \triangleq \frac{1}{2} \left( x + \frac{1}{x} \right)$, and $\log_\beta x$ denotes the logarithm of $x$ with base $\beta$.

**Lemma 5.2.4 (Performance Ratio $R_{\beta}(\beta_0)$ of Power-of-$\beta$ Policies)**

For any fixed base period $\beta_0$, let

$$t_i^* \triangleq \beta_0 \beta \left[ \log_\beta \left( \frac{\mu_i \beta_0}{\beta} \right) \right], \quad \forall i \in N,$$

where $t = (t_1, t_2, \cdots, t_n)$ is the optimal solution to (RJR). Then $t^* = (t_1^*, t_2^*, \cdots, t_n^*)$ is an optimal power-of-$\beta$ policy for (JR) with $\beta_0$ fixed. The worst case performance ratio $R_{\beta}(\beta_0)$ of power-of-$\beta$ policies satisfies

$$R_{\beta}(\beta_0) \triangleq \frac{C_\beta(n, K, h, \beta_0)}{LB(n, K, h)} \leq e \left( \sqrt{\beta} \right).$$

**Proof.** see Section 5.7.1. \qed

The next result, also a direct extension of Roundy, allows base period $\beta_0$ to vary.

**Lemma 5.2.5 (Performance Ratio $R_{\beta}^*$ of Power-of-$\beta$ Policies)**

Let $C_\beta(n, K, h) \triangleq \inf_{\beta_0 > 0} C_\beta(n, K, h, \beta_0)$ be the optimal average cost of power-of-$\beta$ policies for instance $(K, h)$ of $n$-product. The worst case performance ratio $R_{\beta}^*$ of power-of-$\beta$ policies satisfies

$$R_{\beta}^* \triangleq \inf_{\beta_0 > 0} R_{\beta}(\beta_0) = \frac{C_\beta(n, K, h)}{LB(n, K, h)} \leq \rho_{\beta},$$

where $\rho_{\beta} \triangleq \frac{1}{\ln \beta} \frac{\beta - 1}{\sqrt{\beta}}$ is the upper bound of worst case performance ratio of power-of-$\beta$ policies.

**Proof.** see Section 5.7.1. \qed
5.3 Grouping Policies and Performance Ratio

Say \( G \triangleq \{G_1, G_2, \cdots, G_p\} \) is a grouping of \( N \), if \( \{G_1, G_2, \cdots, G_p\} \) forms a partition of \( N \), i.e.,

\[
\bigcup_{i=1}^{p} G_i = N; \quad G_i \cap G_j = \emptyset, \text{ iff } i \neq j, \forall i, j = 1, 2, \cdots, p.
\]

Set \( G_i \) is called a group (of products). The corresponding grouping policy \( G \) will replenish all the products in each group \( G_i \) at the same time \( T_i \), the replenishment period of group \( G_i \). As we mentioned in the introduction, we ignore the possible cost saving due to replenishing different groups at the same time. Therefore, the optimal average cost \( g(G_i) \) of group \( G_i \) is

\[
g(G_i) \triangleq 2\sqrt{K(G_i)h(G_i)},
\]

the corresponding optimal replenishment time \( T_i^* \) of group \( G_i \) is

\[
T_i^* = \sqrt{K(G_i)/h(G_i)},
\]

and the optimal average cost \( C_{GP}(n, K, h, G) \) of grouping policy \( G \) is the sum of optimal average costs of all groups in \( G \):

\[
C_{GP}(n, K, h, G) = \sum_{i=1}^{p} g(G_i),
\]

where \(|G| = p\), and the corresponding optimal replenishment period vector is \( T^* = (T_1^*, T_2^*, \cdots, T_p^*) \).

If we use the (not necessarily optimal) replenishment period vector \( T = (T_1, T_2, \cdots, T_p) \) for grouping policy \( G \), the corresponding average cost is

\[
C_{GP}(n, K, h, G, T) \triangleq \sum_{i=1}^{p} \left\{ \frac{K(G_i)}{T_i} + h(G_i)T_i \right\}.
\]

Therefore, the optimal average cost of grouping policy \( G \) can also be expressed as

\[
C_{GP}(n, K, h, G) = \inf_{T \in R_+^p} C_{GP}(n, K, h, G, T).
\]
Let $\mathcal{D}^n$ be the set of all partitions on $N$, and

$$C_{GP}(n, K, h, \mathcal{D}^n) \triangleq \min_{G \in \mathcal{D}^n} C_{GP}(n, K, h, G),$$

be the average cost of optimal grouping policies for $n$-product instance $(K, h)$.

Because we are mainly concerned with finding an upper bound on the performance ratio of grouping policies, we replace the optimal average cost by its lower bound $LB(n, K, h)$ in our definition of performance ratio of grouping policies. This substitution cannot decrease the performance ratio. Therefore, the upper bound obtained here will also be an upper bound on the 'real' performance ratio. However, because $LB(n, K, h)$ is a very tight lower bound—within 2% of the optimum cost, see Roundy [28] (1985) (by letting $\beta = 2$ in Lemma 5.2.5), we do not really lose much by this substitution.

Now we are ready to define various performance ratios of grouping policies. Let

$$r(n, K, h, G, T) \triangleq \frac{C_{GP}(n, K, h, G, T)}{LB(n, K, h)}$$

be the performance ratio of grouping policy $G$ with replenishment period vector $T$ for $n$-product instance $(K, h)$. Let

$$r(n, K, h, G) \triangleq \frac{C_{GP}(n, K, h, G)}{LB(n, K, h)}$$

be the performance ratio of grouping policy $G$ for $n$-product instance $(K, h)$. Let

$$r(n, K, h, \mathcal{D}^n) \triangleq \frac{C_{GP}(n, K, h, \mathcal{D}^n)}{LB(n, K, h)}$$

be the performance ratio of an optimal grouping policy for $n$-product instance $(K, h)$. Let

$$r^*(n) \triangleq \sup_{(K, h) \in (S^n, R^n)} r(n, K, h, \mathcal{D}^n)$$

be the worst case performance ratio of grouping heuristics for $n$-product, and let

$$r^* \triangleq \sup_{n \in N} r^*(n).$$
be the worst case performance ratio of grouping heuristics (over all problem instances). Note that \( N \overset{\Delta}{=} \{1, 2, 3, \ldots\} \) is the set of natural numbers. Estimating an upper bound on ratio \( r^* \) is the main objective of this chapter.

A partition \( G \in \mathcal{D}^n \) is a consecutive partition iff the indices of the products in each group \( G_i \) are consecutive integers. Let \( \mathcal{C}^n \) be the set of all consecutive partitions on \( N \). Lemmas 5.3.1 to 5.3.5 imply that, to find the worst case performance ratio \( r^* \), we may limit the problem instances to \((K, h) \in \Omega^n ('monotone path' instances), and grouping policies to \( G \in \mathcal{C}^n \).

First, we may limit problem instances from submodular set functions \((\mathcal{S}^n)\) to ‘maximum’ submodular set functions \((\mathcal{M}^n)\):

**Lemma 5.3.1 (First Reduction)**

\[
  r^*(n) = \sup_{(K, h) \in (\mathcal{M}^n, \mathbb{R}_+^n)} r(n, K, h, \mathcal{D}^n).
\]

**Proof.** See Section 5.7.1. \(\square\)

Because of Lemma 5.3.1, we will assume \( K \in \mathcal{M}^n \) hereafter. The following lemma shows that we may restrict attention to grouping policies using only consecutive partitions of \( N \), which we call the consecutive grouping heuristic. Let

\[
  C_{GP}(n, K, h, \mathcal{C}^n) \overset{\Delta}{=} \min_{G \in \mathcal{C}^n} C_{GP}(n, K, h, G)
\]

be the average cost of consecutive grouping heuristics, for \( n \)-product instance \((K, h)\). Let

\[
  r(n, K, h, \mathcal{C}^n) \overset{\Delta}{=} \frac{C_{GP}(n, K, h, \mathcal{C}^n)}{LB(n, K, h)},
\]

be the corresponding performance ratio.

The following inequality is useful in proving Lemma 5.3.3 - consecutive grouping heuristics are optimal.
Lemma 5.3.2 (Consecutive Grouping of Three Products)

If \( \xi_3 \geq \xi_2 > 0, \eta_1, \eta_2, \eta_3 > 0 \), then

\[
\sqrt{\xi_2 \eta_2} + \sqrt{\xi_3 (\eta_1 + \eta_3)}
\geq \min \left\{ \sqrt{\xi_2 (\eta_1 + \eta_2)} + \sqrt{\xi_3 \eta_3}, \sqrt{\xi_3 (\eta_1 + \eta_2 + \eta_3)} \right\}.
\]

Proof. See Section 5.7.1. \( \square \)

Lemma 5.3.3 (Consecutive Grouping Heuristics are Optimal)

If \( K \in \mathcal{M}^n \), then \( C_{GP}(n, K, h, \mathcal{C}^n) = C_{GP}(n, K, h, \mathcal{D}^n) \).

Proof. See Section 5.7.1. \( \square \)

Therefore, using the method proposed in Chakravarty, Orlin, and Rothblum [4] (1982), the optimal grouping policy can be found by an \( O(n^2) \) shortest path algorithm.

The second reduction follows directly from Lemmas 5.3.1 and 5.3.3:

Corollary 5.3.4 (Second Reduction)

The worst case performance ratio of consecutive grouping heuristics for all problem instances \( K \in \mathcal{M}^n \) is \( r^*(n) \), i.e.,

\[
r^*(n) = \sup_{(K, h) \in (\mathcal{M}^n, \mathbb{R}_+^2)} r(n, K, h, \mathcal{C}^n).
\]

The next lemma shows that we may further restrict our attention to problem instances \( (K, h) \in \Omega^n \). Recall that \( \Omega^n \) is the set of 'monotone path' instances.

Lemma 5.3.5 (Third Reduction)
The worst case performance ratio of consecutive grouping heuristics for all problem instances \((K, h) \in \Omega^n\) is \(r^*(n)\), i.e.,

\[
r^*(n) = \sup_{(K, h) \in \Omega^n} r(n, K, h, C^n) = \sup_{(K, h) \in \Omega^n} \frac{\min_{G \in G^n} 2 \sum_{k=1}^{p} \sqrt{K_k h_G}}{2 \sum_{i=1}^{n} \sqrt{(K_i - K_{i-1}) h_i}},
\]

where

\[
G_k = \{ \ell_{k-1} + 1, \ldots, \ell_{k} \}
\]
\[
h_{G_k} = \sum_{i \in G_k} h_i \quad \text{for } p = 1, 2, \ldots, p.
\]

**Proof.** See Section 5.7.1. \(\Box\)

Now we only need to consider problem instances \((K, h) \in \Omega^n\). We find it convenient to replace variables \((K, h)\) by \((x, t)\) as follows:

\[
x_i = \sqrt{(K_i - K_{i-1}) h_i}, \quad t_i = \frac{\sqrt{(K_i - K_{i-1}) h_i}}{h_i}, \quad \forall i \in N.
\] (5.2)

Let \(R^n_+ \triangleq \{ t \in R^n_+ | t_1 < t_2 < \cdots < t_n \} \) be the 'monotone cone' in \(R^n_+\).

It is easy to verify that \((K, h) \in \Omega^n\) is equivalent to \((x, t) \in (R^n_+, R^n_+)\). For simplicity, we do not change function names when variables \((K, h)\) are replaced by \((x, t)\).

The following lemma establishes the correspondence between variables \((K, h)\) and \((x, t)\).

**Lemma 5.3.6 (Transformation of Variables)**

Let \(G = \{ G_1, G_2, \ldots, G_p \} \in \mathcal{C}^n\), \((x, t)\) be defined by (5.2), and

\[
f_{jk}(t, T) \triangleq e \left( \frac{t_j}{T_k} \right) + \frac{1}{2} \sum_{i=k+1}^{p} \frac{t_j}{T_i}, \quad \forall j \in G_k, \ k = 1, 2, \ldots, p.
\]
then for any \((K, h) \in \Omega^n\), i.e., \((x, t) \in (R^n_+, R^n_-)\), we have,

\[
C_{GP}(n, K, h, G, T) = 2 \sum_{k=1}^{p} \sum_{j \in G_k} f_{jk}(t, T) x_j, \quad \text{and}
\]

\[
LB(n, K, h) = 2 \sum_{i=1}^{n} x_i.
\]

Proof. Clearly,

\[
h_i = x_i/t_i, \quad K_i - K_{i-1} = x_i t_i, \quad \text{for} \ i = 1, 2, \ldots, n.
\]

Therefore,

\[
K_i = K_i - K_0 = \sum_{j=1}^{i} (K_j - K_{j-1}) = \sum_{j=1}^{i} x_j t_j, \quad \text{for} \ i = 1, 2, \ldots, n.
\]

As \(G = \{G_1, G_2, \ldots, G_p\} \in C^n\), suppose

\[
G_i = \{\ell_{i-1} + 1, \ldots, \ell_i\}, \quad \text{for} \ i = 1, 2, \ldots, p.
\]

Then

\[
K(G_i) = K_{\ell_i}, \quad h(G_i) = \sum_{j \in G_i} h_j, \quad \text{for} \ i = 1, 2, \ldots, p.
\]

Therefore,

\[
C_{GP}(n, K, h, G, T)
= \sum_{i=1}^{p} \left\{ \frac{K(G_i)}{T_i} + h(G_i) T_i \right\}
= \sum_{i=1}^{p} \left\{ \sum_{j=1}^{\ell_i} \frac{T_i}{T_i} x_j t_j + \sum_{j=1}^{\ell_i} \frac{x_j}{t_j} \right\}
= \sum_{i=1}^{p} \sum_{j=1}^{\ell_i} \frac{T_i}{T_i} x_j + \sum_{i=1}^{p} \sum_{j=\ell_{i-1}+1}^{\ell_i} \frac{T_i}{t_j} x_j
= \sum_{k=1}^{p} \sum_{j=\ell_{k-1}+1}^{\ell_k} \sum_{i=k}^{p} \frac{T_i}{T_i} x_j + \sum_{k=1}^{p} \sum_{j=\ell_{k-1}+1}^{\ell_k} \frac{T_k}{t_j} x_j
= \sum_{k=1}^{p} \sum_{j=\ell_{k-1}+1}^{\ell_k} \left\{ \frac{T_i}{T_k} + \frac{t_j}{T_i} + \sum_{i=k+1}^{p} \frac{t_j}{T_i} \right\} x_j
= \sum_{k=1}^{p} \sum_{j \in G_k} f_{jk}(t, T) x_j.
\]
Chapter 5. Grouping Policies

This completes the proof.

5.4 Upper Bound on the Worst Case Performance Ratio of Grouping Policies

Based on the transformation provided in the previous section, we may rewrite the worst case performance ratio of grouping heuristics \( r^*(n) \) in the following form:

\[
\begin{align*}
\min_{G \in C^n} & \quad 2 \sum_{k=1}^{p} \left( \sum_{j \in G_k} x_j t_j \right) \left( \sum_{j \in G_k} \frac{x_j}{t_j} \right) \\
\sup_{(x,t) \in (R_+^p,R_+^n)} & \quad \frac{2 \sum_{i=1}^{n} x_i}{2 \sum_{i=1}^{n} x_i}
\end{align*}
\]

\[
= \sup_{(x,t) \in (R_+^p,R_+^n)} \left( \min_{G \in C^n} \inf_{T \in R_{[n]}^p} 2 \sum_{k=1}^{p} \sum_{j \in G_k} f_{jk}(t,T) x_j, \right) \frac{2 \sum_{i=1}^{n} x_i}{2 \sum_{i=1}^{n} x_i}
\]

This problem is much simpler than the original problem defined on submodular joint setup cost functions. It has only \( O(n) \) variables. The minimization problem on a set of consecutive grouping policies can be solved by an \( O(n^2) \) shortest path algorithm. However, as a max–min problem, it is still too difficult to find the value of \( r^* = \sup_n r^*(n) \) for the following reasons:

1. The objective function of this problem is neither quasi-concave nor quasi-convex in variables \( x \) and \( t \). It has many local minima and maxima. Therefore, the classic convex analysis methods does not apply.

2. This problem is first to minimize on a set of consecutive grouping policies, then to maximize with respect to variables \( x \) and \( t \). The problem cannot be solved by exchanging the max and min operators.

3. This problem also has many constraints, which further complicate the situation.

Therefore, in this section, we will use a different approach to estimate an upper bound
on $r^*$. First, we reduce the search space from set $\Omega^n$ to a subset of it — $B^n_\beta$, the set of 'root-of-$\beta$ path' instances defined next. Although the worst case performance ratio $r^*_\beta$ of grouping heuristics over problem instances in $B^n_\beta$ used not equal to $r^*$, they yield an upper bound on $r^*$: $r^* \leq \rho_\beta r^*_\beta$. As we know the value of $\rho_\beta$, the next step is to estimate an upper bound on $r^*_\beta$. To do this, we construct a grouping heuristic $H_\beta$, which uses at most two products in each group. An upper bound $W_\beta$ on the performance ratio for the heuristic $H_\beta$ is also an upper bound on $r^*_\beta$: $r^*_\beta \leq W_\beta$ for $\beta = 2, 3, \cdots$. Finally, we use nonlinear programming technique to get the upper bounds on $W_\beta$ and therefore on $r^*$.

First, let us define the following concepts,

\[ B^n_\beta \triangleq \left\{ (K, h) \in \Omega^n \left| \begin{array}{c}
m_1(\beta_0) < m_2(\beta_0) < \cdots < m_n(\beta_0), \\
\text{for some } \beta_0 \in \left[1/\beta, \sqrt{\beta}\right]
\end{array} \right. \right\} \]

be the set of 'root-of-$\beta$ paths'.

Note that $m_i(\beta_0) \triangleq \left\lfloor \log_\beta \left( \sqrt{\frac{K_i - K_{i-1}}{h_i}} \frac{\sqrt{\beta}}{\beta_0} \right) \right\rfloor$, $\forall i$.

\[ R^n_\beta \triangleq \left\{ t \in R^n_+ \left| \begin{array}{c}
m_1(\beta_0) < m_2(\beta_0) < \cdots < m_n(\beta_0), \\
\text{for some } \beta_0 \in \left[1/\sqrt{\beta}, \sqrt{\beta}\right]
\end{array} \right. \right\} \]

be the 'root-of-$\beta$ cone'. Note that

\[ m_i(\beta_0) \triangleq \left\lfloor \log_\beta \left( \frac{\sqrt{\beta}}{\beta_0} \right) \right\rfloor \text{, } \forall i.\]

It is clear that $(K, h) \in B^n_\beta$ is equivalent to $(x, t) \in (R^n_+, R^n_\beta)$. Note that the strict inequalities in the definition imply that all the replenishment times $t_i = \sqrt{(K_i - K_{i-1})/h_i}$ are in distinct 'root-of-$\beta$' intervals $\left[1/\sqrt{\beta}, \sqrt{\beta}\right] \beta_0^m$. Therefore, we have $t_i/t_j \geq \beta^{i-j-1}$, for all $i, j, j \geq i$.

Theorem 5.4.1 (Upper Bound on Performance Ratio of Grouping Policies)

Let $r^*_\beta \triangleq \sup_{n \in \mathbb{N}} \sup_{(K, h) \in B^n_\beta} r(n, K, h, C^n)$ be the worst case performance ratio of consecutive
grouping heuristics over problem instances \((K, h) \in B_\beta^n\) — "root-of-\(\beta\) paths”, then the worst case performance ratio of grouping heuristics \(r^*\) satisfies:

\[
r^* \leq \inf_{\beta \in \mathbb{N}\setminus\{1\}} \rho_\beta r_\beta^*.
\]

**Proof.** See Section 5.7.1. \(\square\)

We now use variables \((x, t)\) to replace \((K, h)\) defined in (5.2). The following Lemma gives us an upper bound on the performance ratio of any grouping \(G\).

**Lemma 5.4.2 (Upper Bound on Performance Ratio of Grouping \(G\))**

Suppose \((K, h) \in \Omega^n\), or equivalently \((x, t) \in (R_+^n, R^n_\infty)\) defined by (5.2). For any grouping \(G\), let

\[
W(n, t, G) = \inf_{T \in R_+^{|G|}} \max_{j \in \mathbb{G}_k} f_{jk}(t, T)
\]

where \(f_{jk}(t, T)\) is defined in Lemma 5.3.6. Then we have

\[
\sup_{x \in R_+^n} \inf_{T \in R_+^{|G|}} \frac{C_{GP}(n, K, h, G, T)}{LB(n, K, h)} \leq W(n, t, G).
\]

**Proof.** By definition, we have:

\[
W(n, t, G) = \inf_{T \in R_+^{|G|}} \max_{j \in \mathbb{G}_k} f_{jk}(t, T)
\]

\[
\geq \inf_{T \in R_+^{|G|}} \frac{\sum_{k=1}^{|G|} \sum_{j \in \mathbb{G}_k} f_{jk}(t, T) x_j}{\sum_{j=1}^n x_j}
\]

\[
= \inf_{T \in R_+^{|G|}} \frac{C_{GP}(n, K, h, G, T)}{LB(n, K, h)}.
\]

This completes the proof. \(\square\)

We can now define a grouping heuristic \(H_b\) in Figure 5.1, where \(b \in R_+\) is a parameter which satisfies \(1 < b \leq \beta\) and whose precise value will be determined later.
Figure 5.1: Grouping Heuristic $H_b$

Input: instance $(K, h) \in \Omega^n$, real numbers $\beta > 1$ and $b \in \left(1, \sqrt{\beta}\right]$.

Output: number $p$ of groups, and grouping $G(b) = (G_1, G_2, \cdots, G_p)$.

Step 1: for $i := 1$ to $n$ do $t_i := \sqrt{(K_i - K_{i-1})/h_i};$

$\quad t_0 := 0; t_{n+1} := +\infty;$

Step 2: $i := 1; k := 0;$

repeat

$\quad k := k + 1;$

if $t_{i+1} > b t_i$

then begin $G_k := \{i\}; i := i + 1$ end

else begin $G_k := \{i, i + 1\}; i := i + 2$ end

until $i > n; p := k.$

As each group $G_k (k = 1, 2, \cdots, p)$ contains at most two products, let $\ell_k \triangleq \max\{i : i \in G_k\}$, so that $G_k = \{\ell_k\}$ or $G_k = \{\ell_k - 1, \ell_k\}.$

Lemma 5.4.3

Suppose $t$ is generated by heuristic $H_b$. Then we have $t_{\ell_k + 1} > b t_{\ell_k} \ \forall k.$

Proof. If $G_k = \{i\}$, then $t_{\ell_k + 1} = t_{i+1} > b t_i = b t_{\ell_k}$. Otherwise, $G_k = \{i, i + 1\}$, and using $i_{i+2} > \beta t_i$ and $b \leq \sqrt{\beta}$, we have $t_{\ell_k + 1} = t_{i+2} > \beta t_i \geq \beta (t_{i+1}/b) \geq b t_{i+1} = b t_{\ell_k}$.

The following technical Lemma is used in the proof of Lemma 5.4.5:

Lemma 5.4.4

If $1 \leq y \leq b$ and $0 < a_2 \leq \sqrt{b}$, then we have

$$e\left(\frac{y}{a_2}\right) \leq e\left(\frac{b}{a_2}\right).$$
Proof. See Section 5.7.1.

To get an upper bound on $f_{jk}(t, T)$, we introduce two additional parameters $a_1$ and $a_2$, which will be optimized later, satisfying $0 < a_1, a_2 \leq \sqrt{b}$. For any grouping policy $G = \{G_1, G_2, \ldots, G_p\}$, let

$$ T_k \triangleq t_{\ell_k} / a_{|G_k|}, \quad \text{for } k = 1, 2, \ldots, p, $$

$$ f_{jk}^b(t, T) \triangleq e \left( a_{|G_k|} \frac{t_j}{t_{\ell_k}} \right) + \frac{t_j}{2} \sum_{i=k+1}^{p} \frac{a_{|G_i|}}{t_{\ell_i}}. $$

**Lemma 5.4.5 (Upper Bound on $f_{jk}^b(t, T)$)**

For any $\beta > 1$ and $b$ satisfying $0 < b \leq \sqrt{\beta}$, let:

$$ c \triangleq \max \left( \frac{a_1}{\beta^2}, \frac{a_2}{\beta^2} \right), $$

$$ g_1(a_1, a_2, b) \triangleq \frac{1}{2} \left( a_1 + \frac{1}{a_1} + \frac{a_1}{b} + \frac{c\beta}{\beta - 1} \right), $$

$$ g_2(a_1, a_2, b) \triangleq \frac{1}{2} \left( a_1 + \frac{1}{a_1} + \frac{a_2}{\beta} + \frac{c}{\beta - 1} \right), $$

$$ g_3(a_1, a_2, b) \triangleq \frac{1}{2} \left( a_2 + \frac{1}{a_2} + \frac{a_1}{b} + \frac{c\beta}{\beta - 1} \right), $$

$$ g_4(a_1, a_2, b) \triangleq \frac{1}{2} \left( a_2 + \frac{1}{a_2} + \frac{a_2}{\beta} + \frac{c}{\beta - 1} \right), $$

$$ g_5(a_1, a_2, b) \triangleq \frac{1}{2} \left( \frac{a_2}{b} + \frac{b}{a_2} + \frac{a_1}{\beta} + \frac{c}{\beta - 1} \right), $$

$$ g_6(a_1, a_2, b) \triangleq \frac{1}{2} \left( \frac{a_2}{b} + \frac{b}{a_2} + \frac{a_2}{\beta^2} + \frac{c}{\beta (\beta - 1)} \right). $$

Then, for $t \in R^3_0$ and any $a_1, a_2$ satisfying $0 < a_1, a_2 \leq \sqrt{b}$, we have, for $j = 1, 2, \ldots, n$:

$$ f_{jk}^b(t, T) \leq \max_{i=1, \ldots, 6} g_i(a_1, a_2, b), $$

where $T$ and $f_{jk}^b(t, T)$ are defined by equation (5.3) and (5.4) respectively.
Proof. Observe that

\[ \ell_i - \ell_k = \sum_{r=k+1}^{i} |G_r|, \quad \text{for } i = k+1, \ldots, p, \]

\[ \frac{t_{\ell_i}}{t_{\ell_j}} \geq \beta^{\ell_i - \ell_j - 1}, \quad \text{for } j > \ell_i, \]

we have

\[ t_{\ell_k} \sum_{i=k+2}^{p} \frac{a|G_i|}{t_{\ell_i}} \leq \sum_{i=k+2}^{p} \frac{a|G_i|}{\beta^{\ell_i - \ell_k - 1}}. \]

Because \( t \in R_p^2 \), we have \( t_i/t_j \geq \beta^{i-j-1} \), for all \( i, j, j \geq i \). Therefore,

\[ t_{\ell_k} \sum_{i=k+2}^{p} \frac{a|G_i|}{t_{\ell_i}} \leq \frac{1}{\beta^{\vert G_{k+1} \vert - 1}} \left( \frac{a|G_{k+2}|}{\beta^{\vert G_{k+2} \vert}} + \frac{a|G_{k+3}|}{\beta^{\vert G_{k+2} \vert + \vert G_{k+3} \vert}} + \cdots + \frac{a|G_p|}{\beta^{\vert G_{k+2} \vert + \cdots + \vert G_p \vert}} \right) \]

\[ \leq \frac{c}{\beta^{\vert G_{k+1} \vert - 1}} \left( 1 + \frac{1}{\beta^{\vert G_{k+2} \vert}} + \cdots + \frac{1}{\beta^{\vert G_{k+2} \vert + \cdots + \vert G_p \vert}} \right) \]

\[ \leq \frac{c}{\beta^{\vert G_{k+1} \vert - 1}} \frac{1}{1 - 1/\beta} \]

\[ = \frac{c}{\beta^{\vert G_{k+1} \vert - 1}} \frac{\beta}{\beta - 1}. \]

There are two cases to consider:

Case 1: \( G_k = \{\ell_k\}, j = \ell_k \).

Using Lemma 5.4.3, we have

\[ t_{\ell_k} \frac{a|G_{k+1}|}{t_{\ell_{k+1}}} \leq \begin{cases} \frac{a_1}{b}, & \text{if } |G_{k+1}| = 1, \\ \frac{a_2}{\beta}, & \text{if } |G_{k+1}| = 2, \end{cases} \]

Therefore,

\[ f_{t_{\ell_k},k}^b(t, T) = e(a_1) + \frac{tt_k}{2} \sum_{i=k+1}^{p} \frac{a|G_i|}{t_{\ell_i}}. \]
Case 2: $G_k = \{\ell_k - 1, \ell_k\}$.

Observe that $1 < t_{\ell_k}/t_{\ell_k-1} \leq b$ and $t_{\ell_k+1}/t_{\ell_k} > \beta/b \geq b$. We have two subcases:

Case 2.1: $j = \ell_k$.

\[
\begin{align*}
    f^b_{\ell_k,k}(t, T) &= e(a_2) + \frac{t_{\ell_k}}{2} \sum_{i=k+1}^{p} \frac{a_{|G_i|}}{t_i} \\
    &\leq \begin{cases} 
        \frac{1}{2} \left( a_2 + \frac{1}{a_2} + \frac{a_1}{b} + \frac{c}{\beta - 1} \right), & \text{if } |G_{k+1}| = 1, \\
        \frac{1}{2} \left( a_2 + \frac{1}{a_2} + \frac{a_2}{\beta} + \frac{c}{\beta - 1} \right), & \text{if } |G_{k+1}| = 2,
    \end{cases}
\end{align*}
\]

Case 2.2: $j = \ell_k - 1$.

Because $1 \leq t_{\ell_k}/t_{\ell_k-1} \leq b$ and $0 < a_2 \leq \sqrt{b}$, by the inequality in Lemma 5.4.4, we have $e \left( \frac{t_{\ell_k}/t_{\ell_k-1}}{a_2} \right) \leq e \left( \frac{b}{a_2} \right)$. Because $t \in R^n_\beta$, we have

\[
    t_{k+1} \sum_{i=k+2}^{p} \frac{a_{|G_i|}}{t_i} \leq t_{k+1} \sum_{i=k+2}^{p} \frac{a_{|G_i|}}{t_i} \leq \frac{c}{\beta} \frac{\beta}{\beta - 1}.
\]

Therefore

\[
\begin{align*}
    f^b_{\ell_k-1,k}(t, T) &= e \left( \frac{a_2}{t_{\ell_k-1}} \right) + \frac{t_{\ell_k-1}}{2} \sum_{i=k+1}^{p} \frac{a_{|G_i|}}{t_i} \\
    &\leq \begin{cases} 
        \frac{1}{2} \left( \frac{b}{a_2} + \frac{a_2}{b} + \frac{a_1}{\beta} + \frac{c}{\beta - 1} \right), & \text{if } |G_{k+1}| = 1, \\
        \frac{1}{2} \left( \frac{b}{a_2} + \frac{a_2}{b} + \frac{a_2}{\beta^2} + \frac{c}{\beta (\beta - 1)} \right), & \text{if } |G_{k+1}| = 2.
    \end{cases}
\end{align*}
\]

This completes our proof.
Chapter 5. Grouping Policies

Corollary 5.4.6 (Upper Bound on $W(n, t, G(b))$)

Suppose $G(b)$ is the grouping generated by heuristic $H_b$, and let

$$W_\beta \triangleq \min_{a_1, a_2, b} \left\{ \max_{i=1, \ldots, b} g_i(a_1, a_2, b) \middle| 0 < a_1, a_2 \leq \sqrt{b}, \ 1 < b \leq \sqrt{b} \right\}.$$  

then

$$W(n, t, G(b)) \leq W_\beta.$$  

Theorem 5.4.7 (Upper Bound on Performance Ratio)

The following is an upper bound on the average cost of worst case performance ratio of grouping policies:

$$r^* \leq \inf_{\beta \in N \setminus \{1\}} \rho_\beta W_\beta \leq 1.4480$$

Proof. For any $\beta \in N \setminus \{1\}$, by definition of $r^*$ in Theorem 5.4.1, we have

$$r_\beta^* = \sup_{n \in N} \sup_{(K, h) \in B^\beta} r(n, K, h, C^n) \leq \sup_{n \in N} \sup_{i \in R^\beta_+} \sup_{x \in R^i_+} \min_{G \in \mathcal{C}} \inf_{T \in R^{|G|}_+} C_{GP}(n, K, G, T) \inf_{T \in R^{|G|}_+} LB(n, K, h) \leq \sup_{n \in N} \sup_{i \in R^\beta_+} \sup_{x \in R^i_+} W(n, t, G(b)) \leq W_\beta.$$  

Therefore, by Theorem 5.4.1, we have

$$r^* \leq \inf_{\beta \in N \setminus \{1\}} \rho_\beta W_\beta.$$
Table 5.1: Values of $\rho_\beta$, $W_\beta$ and $r^*$

<table>
<thead>
<tr>
<th>$\beta$</th>
<th>$\rho_\beta = \frac{1}{\ln \beta} - \frac{1}{\sqrt{\beta}}$</th>
<th>$W_\beta$</th>
<th>$r^* \leq \rho_\beta W_\beta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>$\frac{1}{\sqrt{2\ln 2}}$ = 1.02014</td>
<td>$\leq 1.6453$</td>
<td>$\leq 1.6785$</td>
</tr>
<tr>
<td>3</td>
<td>$\frac{1}{\sqrt{3\ln 3}}$ = 1.05105</td>
<td>$\leq 1.4413$</td>
<td>$\leq 1.5149$</td>
</tr>
<tr>
<td>4</td>
<td>$\frac{1}{\sqrt{4\ln 4}}$ = 1.08202</td>
<td>$\leq 1.3540$</td>
<td>$\leq 1.4651$</td>
</tr>
<tr>
<td>5</td>
<td>$\frac{1}{\sqrt{5\ln 5}}$ = 1.11148</td>
<td>$\leq 1.3028$</td>
<td>$\leq 1.4480$</td>
</tr>
<tr>
<td>6</td>
<td>$\frac{1}{\sqrt{6\ln 6}}$ = 1.13924</td>
<td>$\leq 1.2730$</td>
<td>$\leq 1.4503$</td>
</tr>
<tr>
<td>7</td>
<td>$\frac{1}{\sqrt{7\ln 7}}$ = 1.16541</td>
<td>$\leq 1.2625$</td>
<td>$\leq 1.4714$</td>
</tr>
<tr>
<td>8</td>
<td>$\frac{1}{\sqrt{8\ln 8}}$ = 1.19016</td>
<td>$\leq 1.2529$</td>
<td>$\leq 1.4912$</td>
</tr>
<tr>
<td>9</td>
<td>$\frac{1}{\sqrt{9\ln 9}}$ = 1.21365</td>
<td>$\leq 1.2571$</td>
<td>$\leq 1.5257$</td>
</tr>
</tbody>
</table>

In Table 5.1, we list the values of $\rho_\beta$ and the upper bounds on $W_\beta$ and $r^*$ for $\beta = 2, \ldots, 9$. From this table we get the best estimate from $\beta = 5$. More specifically, let $\beta = 5$, $\rho_\beta = 1.111478$, $a_1 = 0.7675$, $a_2 = 1.2243$, $b = 2.2361$. We have $g_1(a_1, a_2, b) = 1.3028$, $g_2(a_1, a_2, b) = 1.1768$, $g_3(a_1, a_2, b) = 1.2881$, $g_4(a_1, a_2, b) = 1.1622$, $g_5(a_1, a_2, b) = 1.2829$, $g_6(a_1, a_2, b) = 1.2153$. Therefore, $W_\beta \leq 1.3028$, and $\rho_\beta W_\beta \leq 1.4480$. □

5.5 Lower Bound on the worst Case Performance Ratio and Bad Examples

In the previous section, we have established an upper bound 1.4480 on the worst case performance ratio $r^*$ of grouping heuristics. In this section, we give bad examples for lower bounds on $r^*$ and $r^*(n)$. We also get the exact value of $r^*(2)$. To begin with, we introduce a sufficient condition to check that the singleton grouping heuristic, where each group contains exactly one product, is a best grouping:
Lemma 5.5.1 (Sufficient Conditions for Optimal Singleton Grouping)

Suppose that \((K, h) \in \left(\mathcal{M}^n, R^n_+\right)\), and

\[
\frac{K_{i+1} - K_i}{2\sqrt{K_{i+1}K_i}} \geq \sqrt{\frac{h_{i+1}}{h_i}}, \quad \text{for } i = 1, 2, \ldots, n - 1. \tag{5.5}
\]

Let \(G_s \triangleq \{\{1\}, \{2\}, \ldots, \{n\}\}\) be the singleton grouping. Then \(G_s\) is an optimal grouping, i.e.,

\[
C_{GP}(n, K, h, G_s) = C_{GP}(n, K, h, C^n), \quad \text{and}
\]

\[
(K, h) \in \Omega^n.
\]

Proof. We leave it as an exercise for the readers. \qed

It is not difficult to show that the following Lemma holds:

Lemma 5.5.2 (Monotonicity of \(r^*(n)\))

\[
r^*(n) \leq r^*(n + 1) \leq r^*, \quad \forall n \in N \setminus \{1\},
\]

and \(r^* = \lim_{n \to \infty} r^*(n)\).

Proof. Another exercise for the readers. \qed

For \(n = 2\), we have a closed form for \(r^*(2)\).

Lemma 5.5.3 (Value of \(r^*(2)\))

\[
r^*(2) = \frac{7 + 3\sqrt{3}}{11} = 1.108741.
\]

Proof. By using differentiation and a few inequalities, we can show the Lemma holds. We omit the details and give an example which achieves the worst case performance ratio:

\[
(K, h) \in (\mathcal{M}^n, R^n_+), \quad K_0 = 0,
\]
\[ K_1 = 1, \quad K_2 = 2/\sqrt{3} + 1, \]
\[ h_1 = 1, \quad h_2 = 2/\sqrt{3} - 1, \]
then

\[
C_{GP}(2, K, h, C^2) = 2 \left( 1 + 1/\sqrt{3} \right),
\]
\[
LB(n, K, h) = 2 \left( 1 + 1 / \sqrt{4/3 - 2/\sqrt{3}} \right),
\]
\[
r^*(2) = \left( 7 + 3\sqrt{3} \right) / 11.
\]

This completes the proof. \(\square\)

For \(n > 2\), we do not have a closed form expression for \(r^*(n)\), but we have upper and lower bounds on \(r^*(n)\). It is obvious that the upper bound 1.4480 on \(r^*\) is also an upper bound on \(r^*(n)\). The following example, which is a modification of an example appearing in Zheng [39] (1987), and independently found by the author, produces a lower bound on \(r^*(n)\).

**Example.** Let \((K, h) \in (\mathcal{M}^n, R^n_+), K_0 = 0,\)
\[
K_i = 3^i, \quad h_i = 3^{-i}, \quad \text{for } i = 1, 2, \ldots, n - 1,
\]
\[
K'_n = 2 \cdot 3^{n-1}, \quad h'_n = 1/(8 \cdot 3^{n-1}).
\]

Then
\[
\frac{K_{i+1} - K_i}{2 \sqrt{K_{i+1} K_i}} = \frac{3^{i+1} - 3^i}{2 \sqrt{3^{i+1} 3^i}} = \frac{1}{\sqrt{3}} = \frac{h_{i+1}}{h_i}, \quad \text{for } i = 1, 2, \ldots, n - 2,
\]
\[
\frac{K_n - K_{n-1}}{2 \sqrt{K_n K_{n-1}}} = \frac{1}{2 \sqrt{2}} = \frac{h_n}{h_{n-1}}.
\]

By Lemma 5.5.1,
\[
C_{GP}(n, K, h, C^n) = C_{GP}(n, K, h, G_3) = 2 \sum_{i=1}^{n} \sqrt{K_i h_i} = 2 [(n - 1) + 1/2],
\]
and \((K, h) \in \Omega^n\). By Corollary 5.2.2, we have

\[
LB(n, K, h) = 2 \sum_{i=1}^{n} \sqrt{(K_i - K_{i-1})} h_i = 2 \left[ 1 + (n - 2) \sqrt{2/3} + \sqrt{1/8} \right].
\]

Let

\[
r_1(n) \triangleq \frac{n - 1/2}{n \sqrt{2/3} + \left( 1 + \sqrt{2/4} - 2 \sqrt{2/3} \right)},
\]

then

\[
r^* (n) \geq r_1(n), \quad \forall n \in N \setminus \{1\}.
\]

That is, \(r_1(n)\) is a lower Bound on \(r^*(n)\). Let \(n \to \infty\), we get \(r^* \geq \sqrt{3/2} = 1.2247\), which is a lower bound on the performance ratio of grouping policies.

Note that the lower bound given in this example is not tight, for \(n = 2\): \(r_1(2) = 6 / \left( 4 + \sqrt{2} \right) = 1.10819 < 1.108741 = r^*(2)\).

Note that Zheng’s example [39] (1987) with \(K_0 = 0, K_i = 3^i, h_i = 3^{-i}\), for \(i = 1, 2, \ldots, n\), produces a lower bound

\[
r_2(n) \triangleq \frac{n}{1 + (n - 1) \sqrt{2/3}}.
\]

For all \(n \geq 2\), we have \(r_1(n) > r_2(n)\), i.e., the lower bound on \(r^*(n)\) given by the example above is always strictly better than that given by Zheng’s example for any finite \(n\). However, \(r_1(n)\) and \(r_2(n)\) have the same limit \(\sqrt{2/3}\), when \(n\) goes to infinity.

In summary, the bounds on \(r^*\) we have established herein are:

\[
\sqrt{2/3} = 1.2247 \leq r^* \leq 1.4480.
\]

5.6 Conclusions

Grouping policies have been widely used in real world for the simplicity of their implementation. If the setup cost function is separable, i.e., there is no saving on joint
replenishment, the grouping policy consisting of the EOQ solution for each product is the optimal and outperforms power-of-two policies. But when setup costs are not separable, power-of-two policies can have the average cost lower than grouping policies by 20%, as shown in the example above. For the inventory model with network structure, some heuristics, such as integer-multiple lot size policies and integer-split lot size policies in Chapter 3, can be arbitrarily bad. Therefore, it is of interest to determine whether grouping policies can be arbitrarily bad or not. As the submodular setup joint cost is fairly general, and allows the derivation of very tight bounds on the optimal cost, we limit ourselves to this case.

In order to find the worst case performance ratio of grouping policies, we first introduce three reduction lemmas which reduce the space of problem instances from the submodular setup joint cost function to the 'maximum' submodular set function, which has only $O(n)$ variables. The minimization problem is then on a set of consecutive grouping policies, and can be solved by an $O(n^2)$ shortest path algorithm. This problem is much simpler than the original one defined on submodular joint setup cost functions. However, as a max-min problem, it is still too difficult to find the exact value of $r^* = \sup_n r^*(n)$ because:

1. The objective function of this problem is neither quasi-concave nor quasi-convex in variables $x$ and $t$. It has many local minima and maxima. Therefore, the classic convex analysis methods does not apply.

2. This problem is first to minimize on a set of consecutive grouping policies, then to maximize with respect to variables $x$ and $t$. The problem cannot be solved by exchanging the max and min operators.

3. This problem also has many constraints, which further complicate the situation.

Therefore, we use a different approach to estimate an upper bound on $r^*$. We further reduce the search space from set $\Omega^n$ to a subset of it — $B^\beta$, the set of 'root-of-$\beta$ path'
instances. Although the worst case performance ratio $r^*_p$ of grouping heuristics over problem instances in $B^p$ need not equal $r^*$, it yields an upper bound on $r^*$: $r^* \leq \omega_p r^*_p$. We construct a grouping heuristic $H_b$. An upper bound $W_p$ on the performance ratio for the heuristic $H_b$ is also an upper bound on $r^*_p$: $r^*_p \leq W_p$ for $\beta = 2, 3, \ldots$. Finally, we use nonlinear programming to get upper bounds on $W_p$ and therefore on $r^*$: $r^* \leq 1.448$. This means that the average cost given by the best grouping policy is within 44.8% of the optimal.

This result can be easy to extend to the case with backlogging by establishing an equivalence between the backlogging and non-backlogging cases.

5.7 Appendix to Chapter 5

5.7.1 The Proofs of Lemmas in the Text

Lemma 5.2.4 (Performance Ratio $R_p(\beta_0)$ for Power-of-$\beta$ Policies).

For any fixed base period $\beta_0$, let

$$t_i^* \triangleq \beta_0 \beta^\left[\log_\beta \left(\frac{t_i}{\beta_0}\right)\right], \quad \forall i \in N,$$

where $t = (t_1, t_2, \cdots, t_n)$ is the optimal solution to $(JR)$. Then $t^* = (t_1^*, t_2^*, \cdots, t_n^*)$ is an optimal power-of-$\beta$ policy for $(JR)$ with $\beta_0$ fixed. The worst case performance ratio $R_\beta(\beta_0)$ of power-of-$\beta$ policies satisfies

$$R_\beta(\beta_0) \triangleq \frac{C_\beta(n, K, h, \beta_0)}{LB(n, K, h)} \leq \epsilon\left(\sqrt{\beta}\right).$$

Proof. The proof of the optimality of power-of-$\beta$ policies is the same as the proof for power-of-two policies, please refer to Queyranne [24] (1985) and Zheng [39] (1987). We do not repeat it here. Though the proof of the worst case performance ratio is also the same, we need the estimate value in the next lemma and therefore give a brief description in the following.
From the definition of $t^*$, we have

$$\log_\beta \frac{t_i^*}{\beta_0} = \left\lfloor \log_\beta \left( \frac{t_i \sqrt{\beta}}{\beta_0} \right) \right\rfloor,$$

$$\frac{t_i^*}{\beta_0} \leq \frac{t_i \sqrt{\beta}}{\beta_0} < \frac{t_i^* \beta}{\beta_0},$$

$$\frac{1}{\sqrt{\beta}} \leq \frac{t_i}{t_i^*} < \sqrt{\beta},$$

i.e.,

$$t_i \in \left[ \frac{1}{\sqrt{\beta}}, \sqrt{\beta} \right].$$

Let

$$t^*(\ell) \triangleq \beta_0 \beta \left\lfloor \log_\beta \left( \frac{\alpha_i \sqrt{\beta}}{\beta_0} \right) \right\rfloor,$$

for $\ell = 1, 2, \ldots, q$,

then for all $i \in S_\ell \setminus S_{\ell-1}$, because $t_i = t(\ell)$, we have $t_i^* = t^*(\ell)$ and $t(\ell)/t^*(\ell) \in \left[ \frac{1}{\sqrt{\beta}}, \sqrt{\beta} \right]$. Therefore,

$$C_\beta(n, K, h, \beta_0) \leq \sum_{\ell=1}^{q} \left\{ \frac{K(S_\ell) - K(S_{\ell-1})}{t^*(\ell)} + h(S_\ell \setminus S_{\ell-1}) t^*(\ell) \right\},$$

$$= \sum_{\ell=1}^{q} \sqrt{[K(S_\ell) - K(S_{\ell-1})]} h(S_\ell \setminus S_{\ell-1}) \left\{ \frac{t(\ell)}{t^*(\ell)} + \frac{t^*(\ell)}{t(\ell)} \right\},$$

$$= 2 \sum_{\ell=1}^{q} \sqrt{[K(S_\ell) - K(S_{\ell-1})]} h(S_\ell \setminus S_{\ell-1}) e \left( \frac{t(\ell)}{t^*(\ell)} \right),$$

$$\leq LB(n, K, h) e \left( \sqrt{\beta} \right).$$

This completes the proof. \(\square\)

**Lemma 5.2.5 (Performance Ratio $R_\beta^*$ of Power-of-$\beta$ Policies).**

Let $C_\beta(n, K, h, \beta_0) \triangleq \inf_{\beta_0 > 0} C_\beta(n, K, h, \beta_0)$ be the optimal average cost of power-of-$\beta$ policies for instance $(K, h)$ of $n$-product. The worst case performance ratio $R_\beta^*$ of power-of-$\beta$ policies satisfies

$$R_\beta^* \triangleq \inf_{\beta_0 > 0} R_\beta(\beta_0) = \frac{C_\beta(n, K, h)}{LB(n, K, h)} \leq \rho_\beta,$$
where \( \rho_\beta \triangleq \frac{1}{\ln \beta} \frac{\beta - 1}{\sqrt{\beta}} \) is the upper bound on worst case performance ratio of grouping policies.

Proof. Though the proof is exactly as in Roundy [28] (1985), we need the value of \( \rho_\beta \) and thus repeat it below. Let

\[
m_\ell \triangleq \lfloor \log_\beta t(\ell) \rfloor, \text{ for } \ell = 1, 2, \cdots, q,
\]

then \( t(\ell) \in [\beta^{m_\ell}, \beta^{m_\ell+1}) \). Let

\[
Q_\ell \triangleq \frac{t(\ell)}{\beta^{m_\ell+1/2}}, \text{ for } \ell = 1, 2, \cdots, q.
\]

then \( Q_\ell \in \left[1/\sqrt{\beta}, \sqrt{\beta}\right) \). Let

\[
g_\ell(\beta_0) \triangleq \frac{t(\ell) \sqrt{\beta}}{\beta_0} = \frac{\beta^{m_\ell+1} Q_\ell}{\beta_0}, \text{ for } \ell = 1, 2, \cdots, q,
\]

we have

1. if \( \beta_0 \in \left[1/\sqrt{\beta}, Q_\ell\right] \), then \( \lfloor \log_\beta g_\ell(\beta_0) \rfloor = m_\ell + 1 \).

2. if \( \beta_0 \in (Q_\ell, \sqrt{\beta}) \), then \( \lfloor \log_\beta g_\ell(\beta_0) \rfloor = m_\ell \).

Therefore,

\[
t^*(\ell) = \begin{cases} 
\beta_0 \beta^{m_\ell+1}, & \text{if } \beta_0 \in \left[1/\sqrt{\beta}, Q_\ell\right], \\
\beta_0 \beta^{m_\ell}, & \text{if } \beta_0 \in (Q_\ell, \sqrt{\beta}), 
\end{cases}
\]

or

\[
\frac{t^*(\ell)}{t(\ell)} = \begin{cases} 
\frac{\beta_0 \sqrt{\beta}}{Q_\ell}, & \text{if } \beta_0 \in \left[1/\sqrt{\beta}, Q_\ell\right], \\
\frac{\beta_0}{\sqrt{\beta} Q_\ell}, & \text{if } \beta_0 \in (Q_\ell, \sqrt{\beta}), 
\end{cases}
\]

Because \( \int_{1/\sqrt{\beta}}^{\sqrt{\beta}} \frac{d \beta_0}{\beta_0 \ln \beta} = 1 \), and

\[
\int_{1/\sqrt{\beta}}^{\sqrt{\beta}} e^{\left(\frac{t^*(\ell)}{t(\ell)}\right) \frac{d \beta_0}{\beta_0 \ln \beta}}
\]
\[
\begin{align*}
&= \int_{1/{\sqrt{\beta}}}^{\sqrt{\beta}} e\left(\frac{\beta \sqrt{\beta}}{Q_\ell}\right) \frac{d \beta_0}{\beta_0 \ln \beta} + \int_{Q_\ell}^{\sqrt{\beta}} e\left(\frac{\beta_0}{\sqrt{\beta} Q_\ell}\right) \frac{d \beta_0}{\beta_0 \ln \beta} \\
&= \int_{1/{\sqrt{\beta}}}^{\sqrt{\beta}} e(y) \frac{dy}{y \ln \beta} + \int_{1/{\sqrt{\beta}}}^{1/Q_\ell} e(y) \frac{dy}{y \ln \beta} \\
&= \int_{1/{\sqrt{\beta}}}^{\sqrt{\beta}} e(\beta_0) \frac{d \beta_0}{\beta_0 \ln \beta} \\
&= \frac{1}{\ln \beta} \int_{1/{\sqrt{\beta}}}^{\sqrt{\beta}} \left(\frac{1}{\beta_0^2} + 1\right) d \beta_0 \\
&= \frac{1}{\ln \beta} \frac{\beta - 1}{\sqrt{\beta}},
\end{align*}
\]
we have
\[
C_\beta(n, K, h) \leq \frac{1}{\ln \beta} \frac{\beta - 1}{\sqrt{\beta}}.
\]

This completes the proof. \(\square\)

Lemma 5.3.1 (First Reduction).

\[
r^*(n) = \sup_{(K, h) \in (\mathcal{M}^n, \mathcal{R}_n^\Phi)} r(n, K, h, D^n).
\]

Proof. By theorem 5.2.1, for any \(K \in S^n\), there exist \(q\) distinct values \(t(1) < t(2) < \cdots < t(q)\), and a nested path \((S_1, S_2, \ldots, S_q)\) in \(N\) with \(S_\ell \setminus S_{\ell-1} = \{i \in N \mid t_i = t(\ell)\}\),
\[ t(\ell) = \sqrt{\frac{K(S_\ell) - K(S_{\ell-1})}{h(S_\ell \setminus S_{\ell-1})}}, \text{ for } \ell = 1, 2, \ldots, p, \text{ such that} \]
\[ LB(n, K, h) = 2 \sum_{\ell=1}^{q} \sqrt{[K(S_\ell) - K(S_{\ell-1})] h(S_\ell \setminus S_{\ell-1})}, \]
where \( S_0 = \emptyset \). Now define a new set function \( K' : 2^N \to R_+ \) such that for all \( S \subseteq N \),
\[ K'(S) \overset{\Delta}{=} \min \{ K(S_\ell) \mid S \subseteq S_\ell, \text{ for } \ell = 1, 2, \ldots, q \} \]

It is obvious that \( K'(S) \) is nondecreasing, that is, if \( S \subseteq T \), then \( K'(S) \leq K'(T) \).

Because \( K(S) \) is nondecreasing, \( K'(S) \geq K(S) \), for all \( S \subseteq N \). Therefore, we have
\[ C_{GP}(n, K', h, D^n) \geq C_{GP}(n, K, h, D^n). \]

It is easy to verify that all the conditions of theorem 5.2.1 hold for problem instance \((K', h)\). Therefore, by \( K'(S_\ell) = K(S_\ell) \), for \( \ell = 1, 2, \ldots, q \), we have
\[ LB(n, K', h) = 2 \sum_{\ell=1}^{q} \sqrt{[K'(S_\ell) - K'(S_{\ell-1})] h(S_\ell \setminus S_{\ell-1})} = LB(n, K, h). \]

Observe that \( \emptyset = S_0 \subseteq S_1 \subseteq S_2 \subseteq \cdots \subseteq S_q = N \) and \( 0 = K(S_0) < K(S_1) \leq K(S_2) \leq \cdots \leq K(S_q) = K(N) \). If \( i \in S_j \setminus S_{j-1} \), then, since \( K \) is non-decreasing,
\[ K' \{i\} = \min \{ K(S_\ell) \mid \{i\} \subseteq S_\ell, \text{ for } \ell = 1, 2, \ldots, q \} \]
\[ = \min_{\ell \in \{j, j+1, \ldots, q\}} K(S_\ell) = K(S_j). \quad (5.6) \]

For any \( S \subseteq N \), let \( j = \min \{ \ell \mid S \subseteq S_\ell \} \), then there exists \( i \in S \cap (S_j \setminus S_{j-1}) \). Therefore, by definition and equation \((5.6)\), \( K'(S) = K(S_j) = K' \{i\} \). However, \( \{i\} \in S \) implies that \( K' \{i\} \leq \max_{k \in S} K' \{k\} \leq K'(S) \). Therefore, we have \( K'(S) = \max_{k \in S} K' \{k\} \), for all \( S \subseteq N \).

Now let \( K'_1 \overset{\Delta}{=} K'(\{i\}) \), and we reorder the index set \( N \) such that \( 0 < K'_1 \leq K'_2 \leq \cdots \leq K'_n \), we have \( K' \in M^n \).

Therefore, for each \( K \in S^n \), there exists a \( K' \in M^n \) such that
\[ r(n, K', h, D^n) = \frac{C_{GP}(n, K', h, D^n)}{LB(n, K', h)} \]
Chapter 5. Grouping Policies

\[ \frac{C_{GP}(n, K', h, \mathcal{D}^n)}{LB(n, K, h)} = r(n, K, h, \mathcal{D}^n), \]

and we get

\[ \sup_{(K, h) \in (\mathcal{M}^n, \mathcal{R}_n^3)} r(n, K, h, \mathcal{D}^n) \geq \sup_{(K, h) \in (\mathcal{S}^n, \mathcal{R}_n^3)} r(n, K, h, \mathcal{D}^n). \]

However, \( \mathcal{M}^n \subset \mathcal{S}^n \) implies the reverse inequality. Hence, we have

\[ \sup_{(K, h) \in (\mathcal{M}^n, \mathcal{R}_n^3)} r(n, K, h, \mathcal{D}^n) = \sup_{(K, h) \in (\mathcal{S}^n, \mathcal{R}_n^3)} r(n, K, h, \mathcal{D}^n). \]

This completes the proof. \( \square \)

Lemma 5.3.2 (Consecutive Grouping of Three Products).

If \( \xi_3 \geq \xi_2 > 0, \eta_1, \eta_2, \eta_3 > 0 \), then

\[ \sqrt{\xi_2\eta_2} + \sqrt{\xi_3(\eta_1 + \eta_3)} > \min \left\{ \sqrt{\xi_2(\eta_1 + \eta_2)} + \sqrt{\xi_3\eta_3}, \sqrt{\xi_3(\eta_1 + \eta_2 + \eta_3)} \right\}. \]

Proof. By contradiction, if

\[ \sqrt{\xi_2\eta_2} + \sqrt{\xi_3(\eta_1 + \eta_3)} \leq \min \left\{ \sqrt{\xi_2(\eta_1 + \eta_2)} + \sqrt{\xi_3\eta_3}, \sqrt{\xi_3(\eta_1 + \eta_2 + \eta_3)} \right\}, \]

then we have

\[ \sqrt{\xi_2\eta_2} + \sqrt{\xi_3(\eta_1 + \eta_3)} \leq \sqrt{\xi_2(\eta_1 + \eta_2)} + \sqrt{\xi_3\eta_3}, \quad (5.7) \]

and

\[ \sqrt{\xi_2\eta_2} + \sqrt{\xi_3(\eta_1 + \eta_3)} \leq \sqrt{\xi_3(\eta_1 + \eta_2 + \eta_3)}. \quad (5.8) \]

From (5.7), we have

\[ \xi_2\eta_2 + \xi_3(\eta_1 + \eta_3) + 2\sqrt{\xi_3\xi_2\eta_2(\eta_1 + \eta_3)} \leq \xi_2(\eta_1 + \eta_2) + \xi_3\eta_3 + 2\sqrt{\xi_3\xi_2\eta_3(\eta_1 + \eta_2)}, \]
and therefore,

\[(\xi_3 - \xi_2)\eta_1 \leq 2\sqrt{\xi_3\xi_2} \left(\sqrt{\eta_3(\eta_1 + \eta_2)} - \sqrt{\eta_2(\eta_1 + \eta_3)}\right)\].

From (5.8), we have

\[\xi_2\eta_2 + \xi_3(\eta_1 + \eta_3) + 2\sqrt{\xi_3\xi_2\eta_2(\eta_1 + \eta_3)} \leq \xi_3(\eta_1 + \eta_2 + \eta_3),\]

and therefore,

\[2\sqrt{\xi_3\xi_2(\eta_1 + \eta_3)} \leq (\xi_3 - \xi_2)\sqrt{\eta_2}\]

Combining these two inequalities and noticing the conditions of the Lemma, we have

\[
\begin{align*}
2\eta_1\sqrt{\xi_3\xi_2(\eta_1 + \eta_3)} & \leq 2\sqrt{\xi_3\xi_2} \left(\sqrt{\eta_3(\eta_1 + \eta_2)} - \sqrt{\eta_2(\eta_1 + \eta_3)}\right)\sqrt{\eta_2} \\
\eta_1\sqrt{\eta_1 + \eta_3} & \leq \sqrt{\eta_3\eta_2(\eta_1 + \eta_2)} - \eta_2\sqrt{\eta_1 + \eta_3} \\
(\eta_1 + \eta_2)\sqrt{\eta_1 + \eta_3} & \leq \sqrt{\eta_3\eta_2(\eta_1 + \eta_2)} \\
\sqrt{(\eta_1 + \eta_2)(\eta_1 + \eta_3)} & \leq \sqrt{\eta_3\eta_2} \\
\eta_1(\eta_1 + \eta_2 + \eta_3) & \leq 0
\end{align*}
\]

This contradiction completes our proof. \(\square\)

Lemma 5.3.3 (Consecutive Grouping Heuristics are Optimal).

If \((K, h) \in (M^n, R^n_+)\), then \(C_{GP}(n, K, h, C^n) = C_{GP}(n, K, h, D^n)\).

Proof. Because of \(C^n \subset D^n\), we have

\(C_{GP}(n, K, h, C^n) \geq C_{GP}(n, K, h, D^n)\).

For the converse inequality, we show that, if \((K, h) \in (M^n, R^n_+)\), and partition \(G = (G_1, G_2, \cdots, G_p) \in D^n \setminus C^n\), then

\[C_{GP}(n, K, h, G) > C_{GP}(n, K, h, D^n)\] (5.9)
We prove (5.9) above by induction on $n$:

Let $\xi_3 = K_3$, $\xi_2 = K_2$, $\eta_1 = h_1$, $\eta_2 = h_2$, $\eta_3 = h_3$. Then inequality (5.9) follows from lemma 5.3.2 for the case of $n = 3$. Next, suppose inequality (5.9) holds for $n$ in general.

For any instance $(K, h) \in (\mathcal{M}^n, R^n)$, let $G = (G_1, G_2, \cdots, G_p)$ be a partition of $N_1 = \{1, 2, \cdots, n, n+1\}$ such that $G \notin \mathcal{C}^n$. To show that inequality (5.9) also holds, we consider two cases:

**Case 1:** If there is at least one group in $G$ containing two consecutive indices $j, j+1$, then we define $(K', h') \in (\mathcal{M}^n, R_n)$ by combining $j$ and $j+1$ together, i.e., let

\[
K'_i = K_i, \quad h'_i = h_i, \quad \text{for } i = 1, 2, \cdots, j-1;
\]
\[
K'_j = K_{j+1}, \quad h'_j = h_j + h_{j+1};
\]
\[
K'_i = K_{i+1}, \quad h'_i = h_{i+1}, \quad \text{for } i = j+1, j+2, \cdots, n.
\]

We also define partition $G' \overset{\Delta}{=} (G'_1, G'_2, \cdots, G'_p)$ of $N$ by letting, for $i = 1, 2, \cdots, p$,

\[
G'_i \overset{\Delta}{=} \{ m : m \in G_i, m \leq j \} \cup \{ m : m + 1 \in G_i, m > j + 1 \}.
\]

Every grouping of $n$-product for $(K', h')$ induces a grouping of $n+1$-product for $(K, h)$ with the same cost. Therefore,

\[
C_{GP}(n+1, K, h, D^{n+1}) \leq C_{GP}(n, K', h', D^n).
\]

It is obvious that

\[
C_{GP}(n+1, K, h, G) = C_{GP}(n, K', h', G').
\]

Note that $G \notin \mathcal{C}^n$ implies $G' \notin \mathcal{C}^n$, and $(K', h') \in (\mathcal{M}^n, R^n)$, by the induction we have:

\[
C_{GP}(n, K', h', G') > C_{GP}(n, K', h', D^n).
\]

The last three inequalities imply inequality (5.9) for $n + 1$. 

Case 2: If there is no group in \( G \) which contains consecutive indices, let group \( G_p \) contain product \( n + 1 \). There are two subcases to consider:

**Case 2.1:** Group \( G_p \) contains only one product, i.e., \( G_p = \{ n + 1 \} \). Since partition \( G \notin C^{n+1} \), partition \( G' \triangleq (G_1, \cdots, G_{p-1}) \notin C^n \). Let \((K', h')\) denote the restriction of \((K, h)\) to production set \( N = \{1, 2, \cdots, n\} \). By induction, we have \( C_{GP}(n+1, K, h, G) > C_{GP}(n, K', h', G') \). Note that \( C_{GP}(n+1, K, h, G) = 2\sqrt{K_{n+1}h_{n+1}} + C_{GP}(n, K', h', G') \), and \( C_{GP}(n, K', h', D^n) + 2\sqrt{K_{n+1}h_{n+1}} \geq C_{GP}(n + 1, K, h, D^{n+1}) \).

We get that inequality (5.9) holds for \( n + 1 \).

**Case 2.2:** Group \( G_p \) contains more than one product, but does not contain product \( n \). (Otherwise, Case 1 applies.) Let \( G'_p = G_p \setminus \{ n + 1 \} \): we have \( G'_p \neq \emptyset \), and \( n \notin G'_p \). W.l.o.g, suppose \( n \in G_{p-1} \). To apply Lemma 5.3.2, let \( \xi_3 \triangleq K_{n+1} = K(G_p), \xi_2 \triangleq K_n = K(G_{p-1}) = K(G_{p-1} \cup G'_p), \eta_1 \triangleq h(G'_p) > 0, \eta_2 \triangleq h(G_{p-1}) > 0, \eta_3 \triangleq h(G_p) > 0 \). It is easy to verify that the conditions of Lemma 5.3.2 hold. Therefore,

\[
\sqrt{K_n h(G_{p-1})} + \sqrt{K_{n+1} h(G_p)} > \min \left\{ \frac{\sqrt{K_n h(G'_p \cup G_{p-1})}}{\sqrt{K_{n+1} h(G'_p \cup G_{p-1})}} \right\}.
\]

Let

\[
\begin{align*}
G' &= (G_1, G_2, \cdots, G_{p-2}, G_{p-1} \cup G'_p, \{ n + 1 \}), \\
G'' &= (G_1, G_2, \cdots, G_{p-2}, G_{p-1} \cup G_p).
\end{align*}
\]

Then we have

\[
C_{GP}(n+1, K, h, G) > \min \left\{ C_{GP}(n+1, K, h, G'), C_{GP}(n+1, K, h, G'') \right\}.
\]

As \( G' \) fits Case 2.1, we have

\[
C_{GP}(n+1, K, h, G') > C_{GP}(n+1, K, h, D^n).
\]

As \( G'' \) fits Case 1, we have

\[
C_{GP}(n+1, K, h, G'') > C_{GP}(n+1, K, h, D^n).
\]
The last three inequalities imply inequality (5.9) for \( n+1 \), and this completes the proof.

\[ \square \]

Lemma 5.3.5 (Third Reduction).

The worst case performance ratio of consecutive grouping heuristics for all problem instances \((K, h) \in \Omega^n\) is \( r^*(n) \), i.e.,

\[
r^*(n) = \sup_{(K,h) \in \Omega^n} r(n, K, h, \mathcal{C}^n) = \sup_{(K,h) \in \Omega^n} \frac{\min_{G \subseteq \mathcal{C}^n} 2 \sum_{k=1}^p \sqrt{K_{tk} h_{G_k}}}{2 \sum_{i=1}^n \sqrt{(K_i - K_{i-1}) h_i}},
\]

where

\[
G_k = \{\ell_{k-1} + 1, \ldots, \ell_k\} \quad \text{for } p = 1, 2, \ldots, p.
\]

Proof. The inequality \( \sup_{(K,h) \in \Omega^n} r(n, K, h, \mathcal{C}^n) \leq r^*(n) \) follows from \( \Omega^n \subseteq (\mathcal{M}^n, R^n_+) \).

For the converse inequality, suppose \((K, h) \in (\mathcal{M}^n, R^n_+)\). By Theorem 5.2.1, there exists a nested path \((S_1, S_2, \ldots, S_q)\) such that

\[
LB(n, K, h) = 2 \sum_{\ell=1}^q \sqrt{[K(S_\ell) - K(S_{\ell-1})] h(S_\ell \setminus S_{\ell-1})}
\]

and

\[
t(\ell) = \sqrt{[K(S_\ell) - K(S_{\ell-1})] h(S_\ell \setminus S_{\ell-1})}, \quad \text{with } t(1) < t(2) < \cdots < t(q).
\]

Define a \( q \)-product problem instance \((q, K', h')\) by

\[
\begin{align*}
K'_\ell &\triangleq K(S_\ell) \quad \text{for } \ell = 1, 2, \ldots, q, \\
K'(S) &\triangleq \max_{\ell \in S} K'_\ell \quad \forall \ S \subseteq \{1, 2, \ldots, q\}, \\
h'_\ell &\triangleq h(S_\ell \setminus S_{\ell-1}).
\end{align*}
\]
It is obvious that $(K', h') \in \Omega^t$.

By lemma 5.3.2, we have

\[ LB(q, K', h') = 2 \sum_{\ell=1}^{q} \sqrt{[K'_{\ell} - K'_{\ell-1}]} h'_{\ell} = LB(n, K, h). \]

Because each grouping policy in $C^q$ with problem instance $(K', h')$ corresponds to a grouping in $C^n$ with problem instance $(K, h)$ having the same average cost, we have

\[ C_{GP}(q, K', h', C^q) \geq C_{GP}(n, K, h, C^n). \]

Therefore, $r(q, K', h', C^q) \geq r(n, K, h, C^n)$, that is, for any $(K, h) \in (M^n, R^n)$, we have

\[ \sup_{q \leq n} \sup_{(K'', h'')} r(q, K'', h'', C^q) \geq r(n, K, h, C^n), \]

which implies the requisite inverse inequality, and completes the proof.

**Theorem 5.4.1 (Upper Bound on Performance Ratio of Grouping Policies).**

Let $r_\beta^* \triangleq \sup_{n \in N} \sup_{(K, h) \in R^n} r(n, K, h, C^n)$ be the performance ratio of consecutive grouping policy for the 'root-of-\( \beta \) paths' instances, then the worst case performance ratio of grouping policies $r^*$ satisfies:

\[ r^* \leq \inf_{\beta \in N \setminus \{1\}} \rho_\beta r_\beta^*. \]

**Proof.** By Lemma 5.2.5, for any $(K, h) \in (M^n, R^n)$, there exist $\beta_0 \in \left[1, \sqrt{\beta}, \sqrt{\beta} \right]$ and a power-of-$\beta$ policy associated with a nested path $(S_1, S_2, \cdots, S_q)$ with base period $\beta_0$, such that

\[ C_\beta(n, K, h, \beta_0) \leq \rho_\beta LB(n, K, h), \]

where

\[ C_\beta(n, K, h, \beta_0) = \sum_{\ell=1}^{q} \left\{ \frac{K(S_\ell) - K(S_{\ell-1})}{t^*(\ell)} + h(S_\ell \setminus S_{\ell-1}) t^*(\ell) \right\}, \]

\[ LB(n, K, h) = 2 \sum_{\ell=1}^{q} \sqrt{[K(S_\ell) - K(S_{\ell-1})]} h(S_\ell \setminus S_{\ell-1}). \]
for $\ell = 1, 2, \ldots, q$:

$$t(\ell) = \sqrt{\frac{K(S_\ell) - K(S_{\ell-1})}{h(S_\ell \setminus S_{\ell-1})}} \in \left[1 / \sqrt{\beta}, \sqrt{\beta}\right] t^*(\ell),$$

$$t^*(\ell) = \beta_0 \beta^{n_\ell}, \text{ for } n_\ell \in \mathbb{Z}, \text{ with } n_1 \leq n_2 \leq \ldots \leq n_q,$$

$$t^*_i = t^*(\ell), \text{ if } i \in S_\ell \setminus S_{\ell-1},$$

Note that the consecutive inequalities between $\{n_1, n_2, \ldots, n_q\}$ are not strict. In the following we define a problem instance $(K', h') \in \mathcal{B}_\beta^j$ by putting all products with the same replenishment time $t^*_i$ into a same group. Suppose $\{n_1, n_2, \ldots, n_q\}$ take $j$ different values, i.e., let $(r_1, r_2, \ldots, r_j)$ satisfy

$$n_{r_{k-1}+1} = \ldots = n_{r_k}, \quad \text{ for } k = 1, 2, \ldots, j,$$

$$n_{r_k} < n_{r_{k+1}}, \quad \text{ for } k = 1, 2, \ldots, j - 1,$$

where $r_0 = 0, r_j = q$. We recursively define $S'_k$ by letting

$$S'_0 \triangleq \emptyset,$$

$$S'_k \setminus S'_{k-1} \triangleq S_{r_k} \setminus S_{r_{k-1}+1} \text{ for } k = 1, 2, \ldots, j.$$

Partition $\{S'_1, S'_2, \ldots, S'_j\}$ has the following properties:

1. $\{S'_1, S'_2, \ldots, S'_j\}$ forms a nested path.

2. $t^*_i = t^*(r_k) = \beta_0 \beta^{n_{r_k}}, \text{ if } i \in S'_k \setminus S'_{k-1}.$

3. Note also that

$$S'_k \setminus S'_{k-1} = \bigcup_{\ell = r_{k-1}+1}^{r_k} (S_\ell \setminus S_{\ell-1}),$$

$$K(S'_k) - K(S'_{k-1}) = \sum_{\ell = r_{k-1}+1}^{r_k} [K(S_\ell) - K(S_{\ell-1})]$$

$$h(S'_k \setminus S'_{k-1}) = \sum_{\ell = r_{k-1}+1}^{r_k} h(S_\ell \setminus S_{\ell-1})$$
4. \( t^\#(k) \triangleq \sqrt{\frac{K(S'_k) - K(S'_{k-1})}{h(S'_k \setminus S'_{k-1})}} \in \left[ \frac{1}{\sqrt{\beta}}, \sqrt{\beta} \right] t^\ast(r_k), \) for \( k = 1, 2, \cdots, j. \)

This follows from property 3 and \( t(\ell) \in \left[ \frac{1}{\sqrt{\beta}}, \sqrt{\beta} \right] t^\ast(r_k), \) for \( \ell = r_{k-1} + 1, \cdots, r_k, \) we have this property.

5. \( C_\beta(n, K, h, \beta_0) \geq 2 \sum_{k=1}^{j} \sqrt{[K(S'_k) - K(S'_{k-1})] h(S'_k \setminus S'_{k-1})}. \)

Indeed, by properties 2 and 4, we have:

\[
C_\beta(n, K, h, \beta_0) = \sum_{k=1}^{j} \left\{ \frac{K(S'_k) - K(S'_{k-1})}{t^\ast(r_k)} + h(S'_k \setminus S'_{k-1}) t^\ast(r_k) \right\} \\
= \sum_{k=1}^{j} \sqrt{[K(S'_k) - K(S'_{k-1})] h(S'_k \setminus S'_{k-1})} \left\{ \frac{t^\#(k)}{t^\ast(r_k)} + \frac{t^\ast(r_k)}{t^\#(k)} \right\} \\
\geq 2 \sum_{k=1}^{j} \sqrt{[K(S'_k) - K(S'_{k-1})] h(S'_k \setminus S'_{k-1})}.
\]

Now we define a \( j \)-product instance \( (K', h') \) of the joint replenishment problem by letting \( K'_k \triangleq K(S'_k), K'(S) \triangleq \max_{k \in S} K'_k, \) for all \( S \subseteq N, \) and \( h'_k = h(S'_k \setminus S'_{k-1}), \) for \( k = 1, 2, \cdots, j. \) Instance \( (K', h') \) has following properties.

1. \( (K', h') \in (M^j, R^2_+). \)

2. \( (K', h') \in B^j_{\beta}. \)

By property 4 of partition \( \{S'_1, S'_2, \cdots, S'_j\}, \) we have, for \( k = 1, 2, \cdots, j: \)

\[
t'(k) \triangleq \sqrt{\frac{K'_k - K'_{k-1}}{h'_k}} = t^\#(k) \in \left[ \frac{1}{\sqrt{\beta}}, \sqrt{\beta} \right] \beta_0 \beta^{n_{r_k}},
\]

where \( K'_0 = 0, \) all \( n_{r_k} \in \mathbb{Z} \) and \( n_{r_1} < n_{r_2} < \cdots < n_{r_j}. \)

3. \( LB(j, K', h') \leq \rho_\beta LB(n, K, h). \)

Indeed, by Corollary 5.2.2, property 5 for partition \( \{S'_1, S'_2, \cdots, S'_j\} \) and Lemma 5.2.5:

\[
LB(j, K', h') = 2 \sum_{k=1}^{j} \sqrt{(K'_k - K'_{k-1}) h'_k}
\]
\[
2 \sum_{k=1}^{j} \sqrt{[K(S_k^t) - K(S_k^t - 1)] h(S_k^t \setminus S_k^t - 1)} \\
\leq C_\beta(n, K, h, \beta_0) \\
\leq \rho_\beta LB(n, K, h).
\]

4. \(C_{GP}(n, K, h, C^n) \leq C_{GP}(j, K', h', C^n)\).

Because to each grouping for \(j\)-product instance \((K', h')\) corresponds a grouping for \(n\)-product instance \((K, h)\) with the same average cost.

Based on these properties, for any \((K, h) \in \mathcal{M}^n, R^n_+\), there exist \(j < n\) and \((K', h') \in B_\beta^j\) such that

\[
r(j, K', h', C^n) = \frac{C_{GP}(j, K', h', C^n)}{LB(j, K', h')} \\
\geq \frac{C_{GP}(n, K, h, C^n)}{\rho_\beta LB(n, K, h)} \\
= \frac{1}{\rho_\beta} r(n, K, h, C^n)
\]

Therefore, for any \((K, h) \in \mathcal{M}^n, R^n_+\),

\[
\max_{j \leq n} r(j, B_\beta^n, C^n) \geq \frac{1}{\rho_\beta} r(n, K, h, C^n) \\
\max_{j \leq n} r(j, B_\beta^n, C^n) \geq \frac{1}{\rho_\beta} \sup_{(K, h) \in \mathcal{M}^n, R^n_+} r(n, K, h, C^n) \\
\rho_\beta r_\beta^* \geq r^*.
\]

This completes the proof. \(\square\)

**Lemma 5.4.4**

If \(1 \leq y \leq b\) and \(0 < a_2 \leq \sqrt{b}\), then we have

\[
e\left(\frac{y}{a_2}\right) \leq e\left(\frac{b}{a_2}\right).
\]
Proof. Considering $e\left(\frac{y}{a_2}\right)$ as a function of $y$, we know that $e\left(\frac{y}{a_2}\right)$ achieves its minimum at $y = a_2$. If $y \geq a_2$, $e\left(\frac{y}{a_2}\right)$ increases as $y$ increases. If $y \leq a_2$, $e\left(\frac{y}{a_2}\right)$ increases as $y$ decreases. We have the following two cases:

**Case 1**: $0 < a_2 \leq 1$.

As $y \geq 1 \geq a_2$, $e\left(\frac{y}{a_2}\right)$ achieves its maximum at $y = b$. Therefore $e(y/a_2) \leq e(b/a_2)$.

**Case 2**: $a_2 > 1$.

There are two subcases:

**Case 2.1**: $a_2 \leq y \leq b$.

$e\left(\frac{y}{a_2}\right)$ achieves its maximum at $y = b$. Therefore $e(y/a_2) \leq e(b/a_2)$.

**Case 2.2**: $1 \leq y < a_2$.

$e\left(\frac{y}{a_2}\right)$ achieves its maximum at $y = 1$. Therefore $e(y/a_2) \leq e(1/a_2) = e(a_2) \leq e(b/a_2)$.

The last inequality holds because $1 \leq a_2 \leq b/a_2$.

That is, $e(y/a_2) \leq e(b/a_2)$ holds for all cases. □

### 5.7.2 Notation

The followings are the collection of notation arranged in lexicographic order.

$$\sum_{i=k+1}^{k} a_k \triangleq 0$$, for any $a$, by convention.

$$B_{\beta}^n \triangleq \left\{ (K, h) \in \Omega^n \left| m_1(\beta_0) < m_2(\beta_0) < \ldots < m_n(\beta_0), \right. \begin{array}{l} \text{for some } \beta_0 \in \left[\frac{1}{\sqrt{\beta}}, \sqrt{\beta}\right) \end{array} \right. \right\}$$, is the set of 'root-of-\(\beta\) paths'.

Note that $m_i(\beta_0) \triangleq \log_{\beta} \left( \sqrt[2]{\frac{K_i - K_{i-1}}{h_i}} \frac{\sqrt{\beta}}{\beta_0} \right)$, $\forall i$.

$C_{GP}(n, K, h, C^n) \triangleq \min_{G \in C^n} C_{GP}(n, K, h, G)$ is the optimal average cost of consecutive grouping heuristics.
Chapter 5. Grouping Policies

\[ C_{GP}(n, K, h, D^n) \triangleq \min_{G \in D^n} C_{GP}(n, K, h, G) \] is the optimal average cost of grouping policies for \( n \)-product instance \((K, h)\).

\[ C_{GP}(n, K, h, G) \triangleq \inf_{T \in R^{G}} C_{GP}(n, K, h, G, T) \] is the optimal average cost of grouping policy \( G \).

\[ C_{GP}(n, K, h, G, T) \triangleq \sum_{i=1}^{G} \left\{ \frac{K(G_i)}{T_i} + h(G_i)T_i \right\}. \]

\[ C^n \triangleq \text{the set of all consecutive partitions on } N. \]

\[ C_{\beta}(n, K, h, \beta_0) \triangleq \min_{\beta_0 > 0} \left\{ \sum_{i=1}^{n} \left[ \frac{K(U_i) - K(U_{i-1})}{t_{\alpha_i}} + h_{\alpha_i}t_{\alpha_i} \right], \text{ s.t. } t_i = \beta^m \beta_0, m_i \in Z, \forall i \in N \right\} \]

where \( \alpha \) satisfies (5.1a)

and the \( U_i \)'s are defined by (5.1b).

\[ C_{\beta}(n, K, h) \triangleq \min_{\beta_0 > 0} C_{\beta}(n, K, h, \beta_0) \] is the optimal average cost of power-of-\( \beta \) policies for \( n \)-product instance \((K, h)\).

\[ D^n \triangleq \text{the set of all partitions on } N. \]

\[ e(x) \triangleq \frac{1}{2} \left( x + \frac{1}{x} \right), \text{ for } x \in R_+, \text{ is the "effectiveness function"}. \]

\[ f_{jk}(t, T) \triangleq e \left( \frac{t_j}{T_k} \right) + \frac{1}{2} \sum_{i=k+1}^{G} \frac{t_j}{T_i}, \forall j \in G_k, k = 1, 2, \cdots, |G|, \]

\[ G \triangleq \{ G_1, G_2, \cdots, G_p \} \] is a grouping of \( N. \)

\[ |G| = p \] is the cardinality of set \( G. \)

\[ g(G_i) \triangleq 2\sqrt{K(G_i)h(G_i)} \] is the optimal average cost of group \( G_i \).

\[ h(S) \triangleq \sum_{i \in S} h_i \] is the holding cost rate for any set \( S \subseteq N. \)
Chapter 5. Grouping Policies

\[ \text{LB}(n, K, h) \triangleq \min_{t > 0} \sum_{i=1}^{n} \left[ \frac{K(U_{i}) - K(U_{i-1})}{t_{a_{i}}} + h_{a_{i}} t_{a_{i}} \right] \]

is the lower bound on all feasible policies. Note that \( \alpha \) satisfies (5.1a) and the \( U_{i}'s \) are defined by (5.1b).

\[ \mathcal{M}^{n} \triangleq \left\{ K : 2^{N} \rightarrow R^{+} \mid \begin{array}{l}
0 < K_{1} \leq K_{2} \leq \cdots \leq K_{n}, \text{(non-decreasing)} \\
K(S) = \max_{i \in S} K_{i}, \forall S \subseteq N, \text{(maximum)}
\end{array} \right\} \]

denotes the set of 'maximum' submodular set functions on \( N \). Note that \( K_{i} \triangleq K(\{i\}) \forall i \in N \).

\[ N \triangleq \{1, 2, \cdots, n\} \text{ is the set of products.} \]

\[ N \triangleq \{1, 2, 3, \cdots\} \text{ is the set of natural numbers.} \]

\[ r(n, K, h, C^{n}) \triangleq \frac{C_{GP}(n, K, h, C^{n})}{\text{LB}(n, K, h)} \]

is the optimal performance ratio of optimal consecutive grouping policies for \( n \)-product instance \((K, h)\).

\[ r(n, K, h, D^{n}) \triangleq \frac{C_{GP}(n, K, h, D^{n})}{\text{LB}(n, K, h)} \]

is the optimal performance ratio of optimal grouping policies for \( n \)-product instance \((K, h)\).

\[ r(n, K, h, G) \triangleq \frac{C_{GP}(n, K, h, G)}{\text{LB}(n, K, h)} \]

is the performance ratio of grouping policy \( G \) for \( n \)-product instance \((K, h)\).

\[ r(n, K, h, G, T) \triangleq \frac{C_{GP}(n, K, h, G, T)}{\text{LB}(n, K, h)} \]

is the performance ratio ratio of grouping policy \( G \) with replenishment period vector \( T \) for \( n \)-product instance \((K, h)\).

\[ r^{*} \triangleq \sup_{n \in N} r^{*}(n) \text{ is the worst case performance ratio of grouping heuristics.} \]

\[ r^{*}(n) \triangleq \sup_{(K, h) \in (S^{n}, R^{n})} r(n, K, h, D^{n}) \text{ is the worst case performance ratio of the grouping heuristics for } n \text{-product.} \]

\[ r^{*}_{\beta} \triangleq \sup_{n \in N} \sup_{(K, h) \in B^{n}_{\beta}} r(n, K, h, C^{n}) \text{ is the worst case performance ratio of consecutive grouping heuristics over problem instances } (K, h) \in B^{n}_{\beta} \text{ — 'root-of-\beta paths'.} \]
\( R_\beta \) is the worst case performance ratio of power-of-\( \beta \) policies with base period \( \beta_0 \) fixed.

\( R_\beta^* \) is the worst case performance ratio of power-of-\( \beta \) policies with variable base period \( \beta_0 \).

\( R_+ \triangleq (0, \infty) \) is the set of positive real numbers.

\( R_+^n \triangleq \bigtimes_{i=1}^n R_+ \) is the set of \( n \)-dimensional positive real numbers.

\( R_\beta^n \triangleq \{ t \in R_+^n \mid t_1 < t_2 < \cdots < t_n \} \) is the 'monotone cone' in \( R_+^n \).

\( m_i(\beta_0) \triangleq \left \lfloor \log_\beta \left( \frac{t_i \sqrt{\beta}}{\beta_0} \right) \right \rfloor, \forall i. \)

\( S^n \triangleq \{ K : 2^N \to R_+ \mid K(\emptyset) = 0; K(S) \geq 0, \text{ and } K(N) > 0 \) (non-negativity); 

\( K(S) \leq K(T), \text{ if } S \subseteq T \) (non-decreasing); 

\( K(S \cap T) + K(S \cup T) \leq K(S) + K(T), \forall S, T \subseteq N \) (submodularity) \} \)

\( (S_1, S_2, \cdots, S_q) \) is the \textbf{Nested path} of \( N \) satisfying \( \emptyset \subset S_1 \subset S_2 \subset \cdots \subset S_q = N. \)

\( t = (t_1, t_2, \cdots, t_n) \) is the replenishment time vector of a power-of-\( \beta \) policy.

\( t^*_i \triangleq \beta_0 \beta \left \lfloor \log_\beta \left( \frac{t_i \sqrt{\beta}}{\beta_0} \right) \right \rfloor, \forall i \in N, \) is an optimal power-of-\( \beta \) policy to \( (JR) \) with \( \beta_0 \) fixed.

Note that \( t = (t_1, t_2, \cdots, t_n) \) is the optimal solution to \( (RJR) \).

\( T \triangleq (T_1, T_2, \cdots, T_p) \) is the replenishment period vector of grouping \( G \).
Chapter 5. Grouping Policies

\( T_i^* \triangleq \sqrt{K(G_i)/h(G_i)} \) is the optimal replenishment period of group \( G_i \).

\( T^* = (T_1^*, T_2^*, \ldots, T_p^*) \) is the optimal replenishment period vector of grouping \( G \).

\[
W(n, t, G) \triangleq \inf_{T \in R^{|G|}} \max_{j \in G} f_{jk}(t, T).
\]

\[
W_\beta \triangleq \min_{a_1, a_2, b} \left\{ \max_{i=1, \ldots, 6} g_i(a_1, a_2, b) \left| 0 < a_1, a_2 \leq \sqrt{b}, \ 1 < b \leq \sqrt{\beta} \right. \right\}.
\]

\( Z \triangleq \{0, \pm 1, \pm 2, \ldots\} \) is the set of integers.

\[
\rho_\beta \triangleq \frac{1}{\ln \beta} - \frac{1}{\sqrt{\beta}} \] is the upper bound on worst case performance ratio of power-of-\( \beta \) policies.

\( \Omega^n \triangleq \left\{ (K, h) \in (M^n, R^n_+) \left| \sqrt{\frac{K_1}{h_1}} < \sqrt{\frac{K_2 - K_1}{h_2}} < \cdots < \sqrt{\frac{K_n - K_{n-1}}{h_n}} \right. \right\} \) denotes the set of 'monotone path' instances. Note the strict inequalities in the definition.

### 5.7.3 Performance Ratios of Power-of-\( \beta \) Policies

<table>
<thead>
<tr>
<th>( \beta )</th>
<th>Fixed Base Period</th>
<th>Variable Base Period</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>( 1.06066 )</td>
<td>( \sqrt{2 \ln 2} ) = 1.02014</td>
</tr>
<tr>
<td>3</td>
<td>1.15470</td>
<td>( \sqrt{3 \ln 3} ) = 1.05105</td>
</tr>
<tr>
<td>4</td>
<td>1.25000</td>
<td>( \frac{4 \ln 4}{4} ) = 1.08202</td>
</tr>
<tr>
<td>5</td>
<td>1.34164</td>
<td>( \sqrt{5 \ln 5} ) = 1.11148</td>
</tr>
<tr>
<td>6</td>
<td>1.42887</td>
<td>( \sqrt{6 \ln 6} ) = 1.13924</td>
</tr>
<tr>
<td>7</td>
<td>1.51186</td>
<td>( \sqrt{7 \ln 7} ) = 1.16541</td>
</tr>
<tr>
<td>8</td>
<td>1.59099</td>
<td>( \sqrt{8 \ln 8} ) = 1.19016</td>
</tr>
<tr>
<td>9</td>
<td>1.66667</td>
<td>( \sqrt{9 \ln 9} ) = 1.21365</td>
</tr>
<tr>
<td>10</td>
<td>1.73925</td>
<td>( \sqrt{10 \ln 10} ) = 1.23602</td>
</tr>
</tbody>
</table>
Bibliography


Chapter 2. Optimal Inventory Policies


[12] R. Hassin and N. Megiddo. “Exact Computation of Optimal Inventory Policies over an Unbounded Horizon”. Working paper, Tel Aviv University, Israel, School of Mathematical Sciences, Aviv University, Tel Aviv 69978, Israel, October 1989.


Chapter 2. Optimal Inventory Policies


