MULTI-ITEM INVENTORY CONTROL
WITH
JOINT SET-UP COST

BELGIN (İLHAN) VARDAR

MULTI-ITEM INVENTORY CONTROL WITH JOINT SET-UP COST

By

Belgin (İlhan) Vardar

B.S. in I.E., Boğaziçi University, 1978

Submitted to the Faculty of the School of Engineering
in Partial Fulfillment of
the Requirements for the Degree of
Master of Science

in

INDUSTRIAL ENGINEERING

BOĞAZİÇİ UNIVERSITY

1981

APPROVED BY

Çetin Evranuz

Thesis Advisor

Prof.Dr. İbrahim Kavrakoğlu

Dr. Ceyhan Uyar

June 16, 1981

ACKNOWLEDGEMENTS

I would like to express my gratitute to Çetin Evranuz for providing me the opportunity to work on this project, and for his guidance and encouragements throughout the course of this study.

My sincere thanks are due to my sister, Nilgün İlhan, who took the responsibility of all the drawings in this thesis.

Last but not least, I would like to thank Mrs.Nihal Yener for her patience and accuracy in typing this thesis.

BELGİN VARDAR

İstanbul, June 1981

A B S T R A C T

In this study, a multi-item inventory control system for a firm where groups of items are supplied by their own specific factories, was developed. The dependence between items is caused by the set-up cost structure: When several items are ordered on the same order list, a fixed set-up cost, independent of the number of items and quantities, is incurred. In order to use this fixed part of the ordering cost, a joint ordering policy, the (s,c,S), was recommended for the inventory system. As stated by this policy, ordering decisions are taken also for items with inventories below their can-order points (c_i's) when any item with an inventory below its must-order point s is ordered.

A model with a monetary objective function minimizing the sum of ordering and inventory carrying costs and satisfying service level constraints was set-up. An optimization algorithm, which is a combination of mathematical optimization and simulation was used to determine optimal inventory control parameters (must-order points, can-order points, and order-up-to levels).

The solution algorithm was programmed and its implementation for a selected group of items was presented in the study. The cost saving achieved by the dependent policy over a usual independent inventory control system was also demonstrated. Finally, the algorithm's performance under changes in the uncontrollable system parameters was analyzed and a case where changes occur in demand figures was presented at the end of the study.

ÖZET

Bu çalışmada, çok sayıda mamulün belirli fabrikalarda üretilerek bölge depoları aracılığı ile çok sayıda mağazada pazarlandığı bir üretim-dağıtım-satış zinciri incelenmiş ve bölge depolarında mamul stok kontrol sistemlerinin kurulması amaçlanmıştır. Sistemin özelliğinden ötürü, her sipariş açıldığında ısmarlanan mamul sayı ve miktarlarından bağımsız olarak sabit bir sipariş maliyeti oluşmaktadır. Bu tür sipariş yapısı nedeni ile alışılmış stok kontrol politikalarının bu sisteme uygulanabilmesi mümkün görülmemiş ve bölge depolarında aynı fabrikadan karşılanan mamullerin bireysel stok kontrol yerine ortak bir stok politikası ile kontrolu önerilmiştir. Seçilen (s,c,S) stok kontrol politikasına göre, bir mamulün stoku ısmarlanma noktasının (s) altına düştüğünde, stoku ortak ısmarlanabilme noktasının (c) altında tüm mamuller de stok seviyeleri kendi hedef seviyelerine (S) çıkacak şekilde ısmarlanmaktadırlar. Böylece, daha az sayıda sipariş açılarak sabit ısmarlama maliyetinin ekonomik kullanımı mümkün olacaktır.

Sistemin modellenmesinde, toplam ısmarlama ve stokta tutma maliyetlerini enazlayacak bir amaç fonksiyonu oluşturulmuş, stok kopma durumu ise yoksatma maliyetini gerçekçi bir şekilde hesaplamanın zorluğu gözönüne alınarak servis seviyesi olarak kısıtlarda modele dahil edilmiştir. Modelin optimizasyonunda, parametreler arasındaki tüm fonksiyonel ilişkilerin tanımlanamamış olması nedeni ile bilinen matematiksel optimizasyon tekniklerinin kullanımı imkansız olduğu gözlenmiş,böylece matematiksel optimizasyon ve benzetim tekniklerinden yararlanan bir iyileştirme algoritması ile optimal stok kontrol parametrelerine gidilmiştir.

Çalışmanın uygulanmasında ise, algoritmanın bilgisayar programlaması yapılmış ve seçilen bir bölge deposu için model çözümleri verilmiştir. Ayrıca sabit sipariş maliyetinin artması halinde önerilen ortak stok politikasının bağımsız politikalara göre sağladığı kazanç gösterilmiştir. Sistem parametrelerinde bir değişiklik olması halinde algoritmanın çabuk sonuca gidebilme özelliği ise çalışmanın son bölümünde sunulmuştur.

TABLE OF CONTENTS

CHAPTER I
INTRODUCTION

I.I. OVERVIEW OF THE STUDY

The study presented in this thesis deals with one of the problems exposed in the research project entitled "Designing a Stock Control and Distribution System for the Marketing Division of Sümerbank", carried out by the Division of Operational Research of the Marmara Scientific and Industrial Research Institute (MSIRI).

In one of the previous projects carried out by MSIRI, the inclusion of district warehouses in the distribution system of Sümerbank between factories and retail outlets was proposed. After these warehouses were established and started to have an important role in the distribution system, it became necessary to reorganize the stock control and distribution system, to establish the district directorates and to determine their functions.

Motivated by this project, this study deals with a multi-product inventory control system where the items are not treated independently. The dependence is caused by the set-up cost structure. In particular, if KI_i and KI_j are the set-up costs for items i and j respectively under independent replenishments, then the cost of a set-up involving both of the items is less than the sum of set-up costs for two independent replenishments (KI_i+KI_j).

This type of cost structure is particularly appropriate when a group of items are ordered from the same supplier and/or use the same means of transportation; when a replenishment is placed, there is a major fixed cost independent of how many and which items are involved, and a variable cost which depends on the number of items. When several items are included in an order, the unique major fixed cost is shared by all items in that order, and this causes a decrease in the total ordering cost. Same situation exists when a group of items is produced in the same family; a major fixed production cost is required to switch over to the production of the group of items, but only a minor cost is incurred to switch among items within the same group.

A joint ordering policy based on the (s,S) policy as applied to the individual items [10] is proposed for the stock control of this multi-commodity system. In the multi-item (s,c,S) policy as discussed by Balintfy in [2], a third critical inventory control parameter, "can-order point", is added to two inventory control parameters in the individual (s,S) policy. This ordering policy consists of bringing up to its order-up-to level S_i any item i below its can-order point c_i whenever any other item j hits its must-order point s_j.

In this study, a procedure introduced in [22] is used for selecting the control variables (must-order point, can-order point and order-up-to level) of the (s,c,S) system. The procedure which is a combination of mathematical optimization and simulation is computerized. Then this program is run for a real-life data and the results are presented.

In the following sections of this chapter, a general summary of the stock control systems is presented: First, inventory control models are discussed in terms of their general characteristics, and demand, delivery lead time and objective function aspects are analyzed. Next, inventory control policies are examined and several common policies are introduced. Last, some of the studies reported in the literature about multi-item inventory models are summarized.

Chapter II is devoted to problem definition. Existing distribution system and its problems are examined. Then discussing the several alternative policies to eliminate the present problems, an inventory control policy most fitting to the system and with least disadventages is recommended. Mathematical formulation of the problem is given in Chapter III. At the beginning of the chapter, relevant cost terms in the inventory system are determined,then the objective function and the constraints are derived.

Chapter IV deals with the optimization phase and covers the solution algorithm. The updating procedures, some numerical analysis methods used in solving the service-level equations and the steps of the algorithm are all presented in Chapter IV. The adventages of the proposed joint ordering policy over usual independent policies are also shown in this chapter.

Computerization of the optimization algorithm is described in Chapter V. Main structure of the program, explanations about the subroutines and functions, their logic diagrams and the definitions of the variables used in the computer program are all given in Chapter V. Implementation of the model and adventages of the proposed inventory policy over the independent one are demostrated in Chapter VI. The algorithm's performance under changes in the uncontrollable parameters is also analyzed in the same chapter. This thesis ends up with conclusions in Chapter VII. Last chapter contains the criticism of the algorithm and comments on implementation. Theory about the demand characteristics, computations of inventory carrying and item-ordering costs and the list of the computer program are given in appendices.

I.2. INVENTORY CONTROL MODELS

I.2.1. GENERAL CHARACTERISTICS

An inventory problem may be defined as designing an inventory system and of making optimal decisions with respect to that inventory system. Designing an inventory system, first the existing conditions and the characteristics of the present system should be carefully investigated. Then, an operating doctrine which tells us the timing and magnitude of the replenishment decision has to be determined. The chosen operating doctrine may be very different from system to system regarding their characteristics. The existing inventory systems differ in size and complexicity, in the types of items they carry, in the costs associated with operating the system, in the nature of the stochastic processes associated with the system, and in the nature of the information available to decision makers at any given point in time. All these differences can be considered to reflect variations in the structure of the inventory system. Then, all these variations can have an important role on the type of operating policy that should be used in controlling the system.

After a suitable inventory policy is selected, demand properties have to be studied. Since inventories are kept for the purpose of meeting future demands, some information about the nature of demand, based on historical data, market research or executive judgements is essential. There are three possible states of knowledge of demand. First, exact figures of future demand may be available. In such a case, there is no uncertainty on the demand characteristics and it is called a deterministic inventory control system. Secondly, there may be no knowledge about the behaviour of future demand. Neither complete ignorance nor complete certainty is observed in actual situations. A third case, most fitting real-life situations, is one where some knowledge about future demand is obtainable. Most commonly, the demand can be described in probabilistic terms. In practice, probability distribution of demand is not known exactly and parameters characterizing the probability distribution should be determined based on historical data. In such an inventory system, because of the stochastic nature of the demand pattern, there may be times when demands occur and the system is out of stock. An important characteristic of the process generating demands is what happens when a demand occurs and the system is out of stock. Basically, there are two possibilities: Either the demand occurring during the stock-out time is lost, or the customer waits until the inventory system obtains sufficient stock to meet his demand and he is supplied. These are generally referred to as the lost sales case and the backorders case, respectively.

Another important factor in the formulation of inventory control problems is delivery lead time which is defined as the length of time between the placement of an order and the actual addition of that order to the inventory. Delivery lead time is generally not subject to control and consequently is one of the parameters of the inventory systems. Lead time may be zero as in cases of immediate delivery, in which the amount ordered is added to the inventory level as soon as the order is placed. Secondly, lead time may be a constant. After a fixed time interval from the placement of an order, the inventory level is increased by that amount. Last, it may not be constant, since the time to fill the order at the source, the shipping time, and the time required to carry out the paper work, etc. may vary from one order to another. It is seldom possible to predict in advance precisely what the lead time will be, and it will be necessary to assume that a stochastic process generates the lead time. Lead time has generally considerable effects on the solution of probabilistic inventory systems.

The inventory policy and demand properties, with delivery lead time characterize a specific inventory process. Then the criterion for selecting the operating doctrine will have to be formulated. That is called the objective function and it provides a measure of the performance of the inventory control policy. Objective function can be expressed in monetary or non-monetary terms, but in the literature emphasis has been given usually on monetary objective function. The reason is that managers are more concerned with monetary results such as maximization of profit or minimization of cost which both mean more return.

In constructing the monetary objective function, it is often very difficult in practice to determine the stock out cost function. To avoid this problem, an alternative procedure might be to maximize the profit or minimize the cost,each exclusive of the stock out cost, subject to a constraint that the average fraction of the time for which the system is out of stock is not greater than a specified value. Here, instead of specifying the nature of the stock out cost, one instead specifies an upper limit to the average fraction of the time for which the system is out of stock. Alternative criteria for non-monetary objectives may be maximization of the service level to the customers or minimization of the probability of stock outs provided that the capital invested in inventory should not exceed a predetermined value of budget. Either type of objective function may be formulated depending on the characteristics of the system under study.

Studying the characteristics of the inventory process systematically, a model will be set up. When mathematical analysis is used to help develop operating rules and mathematics is applied to the solution of inventory problems while controlling inventory systems, it is necessary to describe mathematically the system under study. Such a description is often referred to as a mathematical model. The procedure is to construct a mathematical model of the system of interest and then to study the properties of the system. However,as it is not possible to represent the real world with complete accuracy, certain approximations and simplifications must be made in constructing a mathematical model. There are many reasons for this. One is that it is essentially impossible to find out what the real world is really like. Another is that a very accurate model of the real world can become impossibly difficult to work with mathematically. Also, accurate models might not be justified economically. Simple approximated ones might yield results which are good enough so that the additional improvement obtained from a more accurate model might not be sufficient to justify its additional cost.

After system is studied, its characteristics are examined, mathematical model is established, and the values of input parameters are calculated, the next step is to determine the values of the decision parameters of the inventory policy which optimize the given objective function.This part of the study is called the optimization stage. Finally, a sensitivity analysis can be carried out to observe to what degree the objective function is influenced by changes in various parameters.

I.2.2. INVENTORY POLICIES

Two fundamental questions to be answered in controlling the inventory of any commodity are (i) when to order and (ii)how much to order. An inventory policy should provide answers to these questions.

There are two possible approaches for deciding about the timing of the ordering. One uses the time and the other uses the inventory level as the decision criterion to answer the question of when to order. These alternatives might be expressed more specifically as follows:

i) placement of an order after every t units of time

ii) placement of an order when the inventory level is equal to or below a given level, say s units.

The amount to be ordered may also be specified in one of two ways:

i) The order quantity is always the same, say q units. An amount of q is ordered whenever one decides to order.

ii) A variable quantity which is the difference between the inventory level and a certain level S is ordered at every replenishment. Here, inventory level is being raised always to a predetermined level.

In the literature, the quantities t, s, q and S are used and defined as the scheduling period, the re-order point, lot-size and order-up-to level respectively.

For immediate delivery, the inventory in-sight is always equal to the inventory on-hand and inventory on-hand will be used deciding for an order. But for a system with a non-zero lead time, inventory position (or inventory in-sight) is defined as the inventory on-hand plus on-order minus back orders. In such a system, inventory position is controlled instead of inventory level.

Most commonly used policies are the (t, S), the (s, Q), and the (s, S) policies. The (t, S) policy, known as the cyclical review system, is characterized by scheduling period and order-up-to level. According to this periodic-review policy, at the end of every interval of t time units, a quantity to bring the inventory position to S is ordered. Re-order point and lot-size characterize the (s,q) policy. Inventory level is reviewed continuously and whenever it is equal to, or below the re-order point, a lot-size q is scheduled for a replenishment. In the (s, S) policy, specified by the two parameters s and S corresponding to "minimum" and "maximum" inventory positions respectively, whenever the inventory position is equal to or below the re-order point s, a quantity that will bring the inventory to S is ordered.

There are several other inventory control policies generated for special purposes by combining these common ones. The (t, s, S) policy is a combination of the periodic review and the continuous review models. According to that policy, an order is placed at the end of the every interval to bring the inventory level up to S, but if inbetween periods , the in-sight inventory declines to or below s, and order to bring the inventory position to S is also scheduled. The (t,s,q) policy differs from the (t, s, S) policy only by the amount of the order; here a lot-size q is scheduled when an ordering decision is taken.

Several other inventory policies have been discussed in the literature. Some of the references are [10], [16] and [25].

I.3. LITERATURE SURVEY

Some of the studies met in the literature about the multi-item inventory control problems are summarized in this section. As the number of items and the degree of complexity of the system increase, the difficulties of the modelling and of the solution procedure also increase. Most of the methods are applicable to the problems which have only a limited number of items.

Luigi Mariani and Bernardo Nicoletti [14] studied a deterministic, continuous time, nonstationary multi-product inventory model and took the minimization of the total cost over a finite time horizon as the criterion of optimality. For both backlogging and nobacklogging cases, they generated a set of difference equations and used a method based on a modified form of the discrete maximum principle for solving the resulting optimization problem. The conditions for the determination of the optimal policy (the number of joint replenishments, the order quantities and the times at which orders should be placed) are given,and the cases for which these conditions are necessary and/or sufficient are discussed. The case in which the costs are quadratic, is dealt within detail and numerical examples are demonstrated.

Joseph L.Balintfy [2] also examined the multi-item inventory problems where through joint ordering of several items total set-up costs may be decreased. His policy operates through the determination of a reorder range within which several items can be ordered. The existence of an optimal reorder range is proven and a computational technique is demonstrated. Then individual and joint ordering policies are compared in this study.

A different inventory policy, (τ, γ), is introduced by Fred Andres and Hamilton Emmons in [1]. In the (τ, γ) policy, a renewal is defined as a time when all products are ordered simultaneously and then, τ is taken as the time between successive renewals and $\gamma=(\gamma_1, .. \gamma_i .., \gamma_n)$ is a vector where γ_i is the number of orders of product i until the next renewal. It is assummed that each of the γ_i orders are equally spaced in time and an amount is ordered so that the stock level of i is zero at the next renewal time.A branch and bound algorithm is presented for finding the optimal policy. In that algorithm, the γ space is searched systematically in such a way that an upper bound on the savings in the total cost for all policies not searched yet is determined. For the two-product case, a special algorithm is developed to give more precise results.

Ho [11] has represented the multi-product inventory system by a Markov process. She developed a model,namely (R, C, Q) model,based on the independent (R, Q) policy and determined the can-order level (C) by balancing the reduced cost of time weighted backorders with the extra carrying cost. Furthermore, she uses a fixed order quantity Q_i for item i rather than an order-up-to level S_i. Her analysis involves certain approximations that allow determination of the stead state probabilities of the associated Markov process. Simulation experiments are also done to compare the joint ordering policy with the independent orderi policy for the case of stuttering Poisson demands, and she show that the joint ordering policy achieves a substantial saving in total cost over the independe ordering policy.

S.K. Goyal examined the multi-item, single supplier systems and gave similar approaches with small differences in [5],[6], [7], [8], [9]. He presented a search procedure for obtaining the optimum packaging frequencies for a number of items which are manufactured jointly but packaged individually after manufacture. His method is equally applicable to those problems where the optimum ordering policy is to be obtained for a number of items from a single supplier. He developed an iterative procedure to obtain the optimum frequencies in a deterministic system and gave lower and upper bounds for packaging frequencies for each item in a sub-algorithm. Evaluating the frequency combinations and fathoming impossible branches, he determined the ordering policy. The effect of the change of the frequency for one item on the total cost function was observed, and a new frequency combination improving the total cost function each time was obtained. He continued adopting the frequency combinations until no more improvement was possible.

CHAPTER II

DEFINITION OF THE INVENTORY PROBLEM

In this chapter, definition of the inventory problem is given and an inventory control policy appropriate for this system is recommended. In the first section, current production-distribution and marketing systems of Sümerbank are described and the flow of goods and information in the overall system is given. Then, the subsystem with which this thesis is concerned is specified and its boundaries are drawn. A new inventory control system is recommended in the second part of the chapter. First, conditions effecting the inventory control are analysed and general characteristics of the new system are determined. Then, alternative inventory control policies suitable for this system are discussed and an appropriate policy is presented.

II.I. DESCRIPTION OF THE GENERAL INVENTORY SYSTEM

In the production-distribution and marketing system of Sümerbank, production is carried out in 23 factories and then marketed over Turkey by the Marketing Organization (the ASM) through four groups of buyers. The operation of the system can be described as follows:

i) Sale estimates prepared by the retail shops and district warehouses are sent to the Marketing Organization. The ASM forecasts the future sales of the whole system by taking these estimates and public sales into consideration.

ii) Yearly protocols are prepared together by the ASM and the factories. Taking the forecasts of the ASM and the available production facilities into consideration, yearly production programs, based on types, are set up which are then approved by the General Directorate of Sümerbank.

iii) Within a year, usually every 3 or 6 months, the ASM gives the factories the desired production volumes for each color and design.

iv) The retail shops send monthly order lists to the district warehouses to which they are affiliated. The district warehouses compile and revise these, and then forward them to the factories.These order lists usually do not cover details such as color and design, but only types.

v) The factories send the goods to the district warehouses and,rarely to the retail shops directly, according to the order lists which they have received. There may be also some direct sales to the public from the factories.

vi) The district warehouses send the goods on the order lists to the retail shops assigned to them, then they are sold to the customer. Other wholesalers and retailers may also be directly supplied by the district warehouses.

Figure II.1. illustrates the main flow of information, goods and money between the General Directorate of Sümerbank, the factories, the ASM, the district directorates and four groups of customers.

The present conditions are such that the direct delivery from factories to the retail shops constitutes a small proportion of the total goods flow. Another fact is that the demands of public customers and wholesalers can be estimated more or less accurately. With these in mind, then the flow of goods from factories to retail shops through district warehouses gains the major importance in the whole distribution system. Consequently, the main product flow which will be analyzed here can be represented schematically as in Figure II.2.

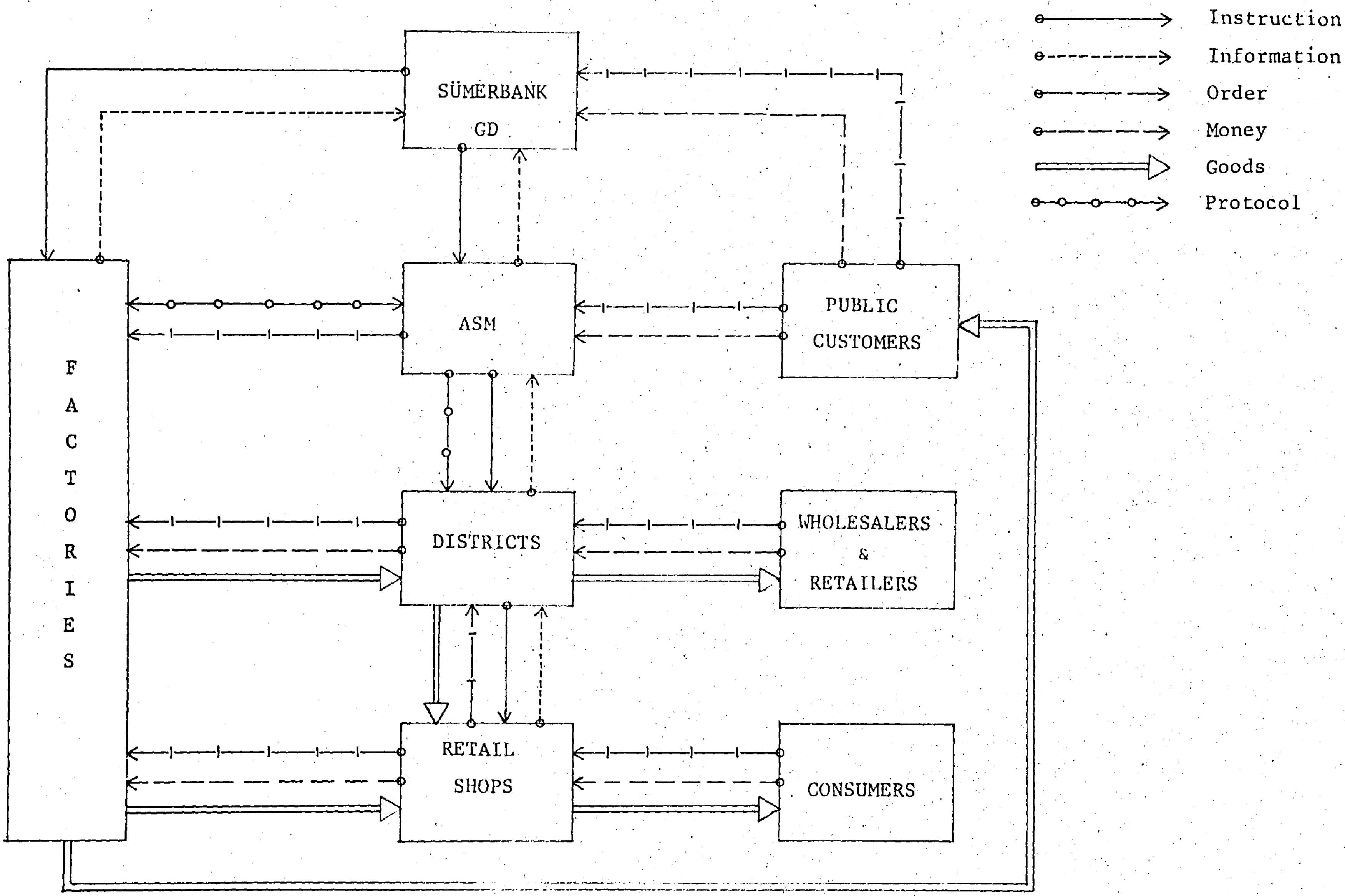


Figure II.1. Information, Orders, Goods, Money Flow Chart of Sümerbank

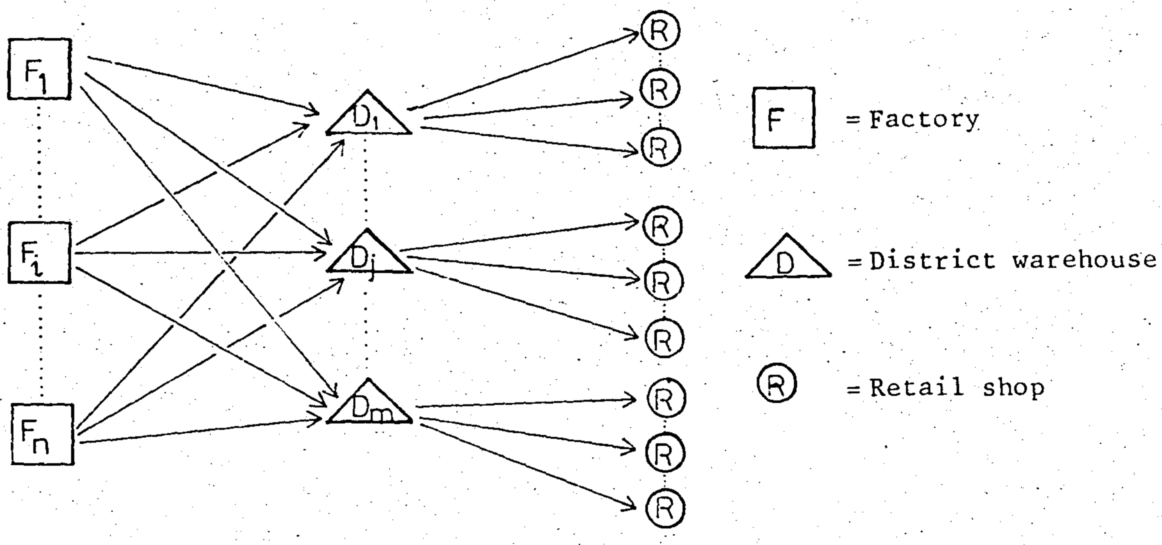


Figure II.2. The Schematic Representation of the Main Product Flow in the Analyzed Distribution System of Sümerbank

District warehouses are very important elements in this system and they act as buffer stocking points between factories and retail shops. By meeting the demands of the retail shops through district warehouses, a more efficient structure with respect to the fixed cost of ordering results. Carrying stocks in district warehouses also make quicker replenishment of retail shops possible. Furthermore, regrouping the goods in the district warehouses and sending in smaller but assorted lots increases the opportunity of quick marketing and thus raises the sales. The deliveries from district warehouses to retail shops, however, have less importance, because smaller amounts and shorter lead times compared to the factory-district warehouse pair are involved. Therefore the main emphasis is given here to the stock control in district warehouses while the retail shops are viewed only as sources of demand for the district warehouses. The model to be set up in this study will deal with the stock control of district warehouses.

I.2. DEVELOPMENT OF A NEW INVENTORY CONTROL SYSTEM

In the system described above, each item is produced in a specific factory and there is accepted no substitution between the products of different factories. Keeping these two assumptions in mind, items which are ordered from one factory and thus can use the same transportation facility will be examined seperately from the others. Dividing the products of each district into groups which are supplied by the same factory and controlling the groups individually, a general stock control system in a specific district warehouse will be set up. So, the whole system is divided into sub-systems for each factory-district pair.

Since the number of productswhich are stocked in a district warehouse is not so small, a sophisticated stock control policy which requires many calculations at every order does not seem very appropriate for the system. Preferably, the recommended inventory policy should be practical and easy to implement .

An overall inventory policy composed of individual policies for each item will not necessiate high stock levels, but, as it increases the total number of replenishments, will not be efficient with respect to the transportation cost. Then to decrease the ordering cost, when the replenishment of an item is dictated by its independent policy, the actual replenishment decision can be taken according to one of the following policies:

i) Do not place the order of that item. Wait until for some other items replenishments are dictated by their individual policies, and then place a joint order.

This policy will lower the total ordering and stock holding costs. However, as stock levels are lower than that prescribed by individual policies, stockouts will occur more frequently.

ii) Place an order for the item. To use the transportation facility efficiently, two possibilities can be considered:

a) Order an amount larger than that determined by the individual policy. In this case total ordering cost will be lowered, but inventory holding cost will increase because more inventories are kept.

b) Do not enlarge the order size for that item. However, include some other items, which don't have to be ordered yet, in the order list. Then the ordering cost is shared by the items on the order list. In the long run, the number of orders and the total ordering cost for the total inventory system will decrease. Inventory levels of all items will be higher, resulting in somewhat increased inventory carrying costs, but providing higher service levels.

Among these policies, the most suitable one with least disadvantages seems the policy given by(ii.b). Through the use of this policy, overall ordering cost is reduced, and instead of unnecessarily raising the stock level of a specific item at a given time, stock levels of several items are raised to some degree. In this way, the increases in inventory levels are more uniformly balanced, and the stockout probabilities are reduced for all items.

When an item has to be replenished, the problem becomes one of deciding on:

i) Which other items should be included in the order list,

and ii) How much should be ordered from these items.

To answer these questions, each item must have another easily applicable control parameter in addition to the ones in individual policies. This critical inventory control parameter is related to the inventory position and is called can-order point. It is used to control the inventory system together with two other parameters which answer the questions of when and how much one must order.Thus, keeping in mind that the (s, S) policy is the optimal one among all individual inventory control policies, the (s, c, S) policy will be the one most suitable to this multi-item inventory system. According to this recommended (s, c, S) policy, the inventory system will be controlled as follows:

i) If all items have inventories above their re-order points (s), no order will be placed as in the individual (s, S) policies.

ii) When the inventory position of an item drops below its re-order point, s, a quantity to bring the inventory position to its order-up-to level S is ordered.
Also, the inventory positions of all other items are reviewed one by one and:
- If its inventory position has not dropped below its can-order point c yet, this item is not included in that order.

- If its inventory position is below its can-order point c, this item is included in the order by a quantity raising its inventory position to its order-up-to level S.

A realization of the (s,c,S) policy for three items is shown in Figure II.3.

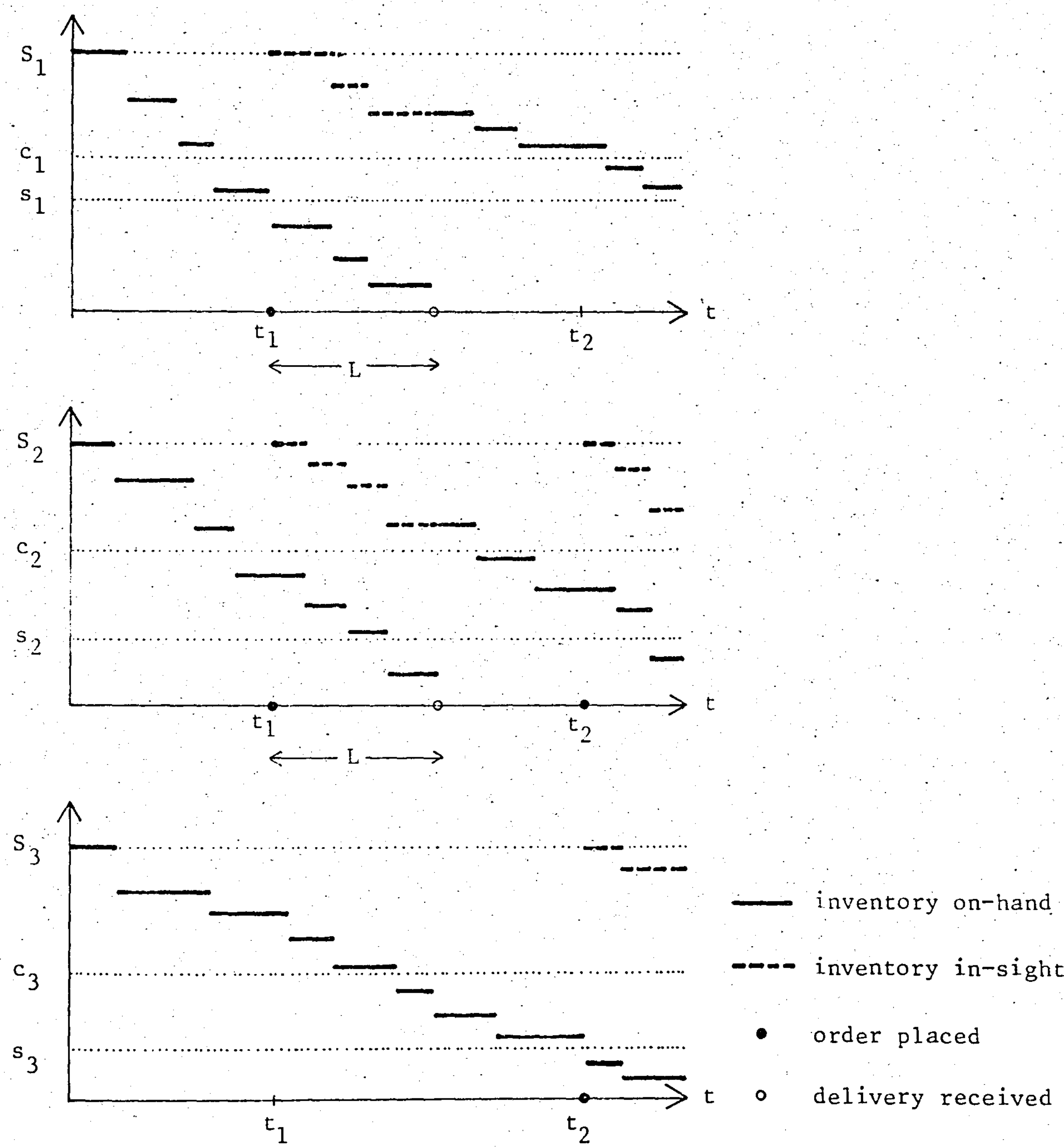


Figure II.3. Realization of the (s,c,S) Policy for Three-Item System

As seen in Figure II.3, as soon as the inventory position of the first item drops below its must-order point s_1 at time t_1, an order to bring its inventory position to S_1 is placed and in addition, inventory positions of the other two items are reviewed. Even though the inventory position of the second item is above its must-order point s_2, a quantity to raise its inventory position up to S_2 is ordered because its inventory position is below its can-order point c_2. But the third item, with an inventory position between its can-order point c_3 and order-up-to level S_3, is not replenished in that joint order. Amounts ordered arrive after L units of time and are added to the respective inventories on-hand. No other replenishment is placed until time t_2 when it becomes necessary to place an order for the third item. First item, whose inventory position is still above c_1, is only reviewed; but the secondone with an inventory position below its can order point c_2 is included in this replenishment to bring its inventory position to its order-up-to level S_2.

In the studied inventory system, lead times of all items ordered by a specific district warehouse from a specific factory are the same because they use the same means of transportation. But lead times are different for each factory-district warehouse pair. This parameter is taken as the average time passed between the placement of an order and the actual addition of that order to the inventory.The main elements of the lead time are the time of communication between district warehouse and factory to place an order, time spent in the factory for the preparation of properly assorted lots; loading and unloading times; and the times passed on the road to transport the goods from the factory to the district warehouse. This last component which is highly longer than the others is really the determining factor of the lead time.

CHAPTER III

FORMULATION OF THE INVENTORY PROBLEM

III.1. INVENTORY COST PARAMETERS

Inventory control models require an objective function to measure system performance and unless it is expressed in non-monetary terms the objective function includes various cost parameters. In real-life problems, the efficiency of any inventory control model largely depends on the accuracy and acceptability of the data used to estimate these cost parameters as well as on the validity of the assumptions made and the techniques used in developing the model. A very sophisticated model developed by using elegant techniques with valid assumptions is not reliable on as long as the cost parameters are not correctly measured or evaluated.

The costs incurred in operating an inventory system play a major role in determining what the operating doctrine should be. The costs which influence the operating doctrine are those which vary as the operating doctrine is changed. Costs that are independent of the applied operating doctrine need not to be included in any analysis where costs are used as an aid in determining the operating doctrine. Fundamentally, there are three catagories of costs which occur in inventory systems and may be important in determining what the operating doctrine should be. These are ordering cost, inventory carrying cost and shortage cost. However, before going into the discussion of these costs, it should be stated that it is quite difficult to represent mathematically all the cost components with near accuracy. Consequently, there is tendency to make some approximations when representing the costs in the mathematical models to be developed.

III.1.1. ORDERING COST

Ordering costs are incurred as a result of placing an order for replenishment. These kinds of costs include all those cost components incurred from the placement of an order to the addition to the inventory such as clerical work, paper and postage costs, labor costs and transportation costs. Ordering cost may be constant per order; or variable depending on the amount ordered, or both.The constant part of the ordering cost is usually called as set-up cost and incurred whenever an order is placed independent of the order content. These costs include fixed order entry costs such as paper, postage,telephone charges etc., as well as parts of transportation, receiving and inspection costs which are independent of the order size. The variable part of the ordering cost enholds the cost terms those depending on the quantity ordered such as the cost of the units, some of the receiving and inspection costs and the quantity-dependent transportation costs.

In the inventory system to be studied, the ordering cost has two components.First one is a set-up cost which is incurred independently from the number of items and the amounts. It contains the cost of placing an order such as communication and clerical work costs and a fixed charge paid to the transportation companies for each distribution. Among these cost terms, the last term is the most important one in the actual system. The second component of the ordering cost is item-ordering cost and incurred when an item is ordered regardless of the amount.This cost consists of the clerical work done to order a specific item, the cost of man-hour to place and receive that order,machine-work to prepare a good lot assorted in color and design, loading and unloading works for each lot.

III.1.2. INVENTORY CARRYING COST

Inventory carrying costs are incurred as a result of holding inventories and increase in direct proportion to increases in inventory on-hand and the time for which inventoried items are held. The following components of the inventory carrying cost are the real out of packet costs:

i) Storage or warehouse rental costs
ii) The cost of operating the warehouse such as light, heat, night watchment etc.
iii) Clerical and administrative costs
iv) Insurance and taxes on inventory held
v) Costs of depreciation, deterioration and obsolence of inventory.

A cost which is frequently the most important cost is not direct out of pocket cost but rather an opportunity cost which would never appear on an accounting statement. This is called the cost of capital and incurred by having capital tied up in inventory rather than having it invested elsewhere. It reflects the lost earning power of capital and is equal to the largest rate of return which the system could obtain from alternative investments.

It is observed that the most significant parts of the inventory carrying cost in the presented system are the cost of capital and the storage and operating costs. Alternative methods for the evaluation of the inventory carrying cost can be found in [17] and how the inventory carrying cost is evaluated in this study will be given later in Appendix III.

III.1.3. SHORTAGE COST

Shortage cost is the cost incurred as a consequence of a stockout, that is when the demand can not be fully and immediately satisfied due to a stock shortage. This cost is composed of:

i) Lost profits: As a result of a stockout, some customers would not want to wait. Some orders and hence the profit that otherwise could be made from these orders will be lost.

ii) Stockout maintenance cost: Even if all demands occurring when the system is out of stock are backordered and no order is lost, cost of loosing the opportunity of using standard and common procedures, cost of extra comminication, cost of additional labor and paperwork will be incurred as a result of a stockout.

iii) Loss of goodwill: The most important component of the cost of a stockout is the somewhat intangible goodwill loss. This may include loss of future sales by that customer or loss of the others' sales affected by him. This cost should also include loss of sales of other items marketed by the same system.

In the system presented, retail shops don't cancel their orders even if stockout takes place, and the portion of demand which is not satisfied directly out of stock is completely backordered. Then no lost profit will occur in this actual system. Still, keeping in mind the difficulties of evaluating the other components of the stockout cost, satisfaction of service levels for all items is imposed in the model instead of costing stockouts.

Mathematical representation of the relevant cost terms and the formulation of the inventory control model are given in the following sections.

III.2. MATHEMATICAL FORMULATION

III.2.1. CONSTRUCTION OF THE OBJECTIVE FUNCTION

The inventory system with all relevant costs is optimized under a monetary objective function. The sum of the ordering and stock holding costs are minimized in objective function and stockout costs are expressed by service levels in constraints while modelling the system. Service criterion for each item is chosen as "not to have probability of running out more than an allowable percent". If

S_i = order-up-to level for item i

c_i = can-order point for item i

s_i = must-order (re-order) point for item i

D_i = expected value of the yearly demand for item i

L = deterministic replenishment lead time

KF = fixed set-up cost per order

KJ_i = ordering cost for item i in the order triggered by another item

KI_i = total ordering cost for item i if the order is triggered by itself ($=KF+KJ_i$)

h_i = inventory carrying cost per year per unit of item i

NIR_i = expected number of independent replenishments of item i in a year

$NJRB_i$ = expected number of joint replenishments triggered by item i in a year

$NJRNB_i$ = expected number of joint replenishments of item i not triggered by i in a year

NTR_i = expected number of total yearly replenishments of item i

n = number of items in a group

expected yearly ordering cost for item i, K(i), will be the sum of all ordering costs in independent and joint replenishments. When an item is included in an order triggered by another one, ordering cost belonging to only that item is charged. But if an item triggers an order, whatever independent or joint replenishment occurs, both components of the ordering cost will be charged. Then, K(i) is expressed by:

$$K(i) = (NIR_i + NJRB_i)\,(KF+KJ_i) + NJRNB_i \cdot KJ_i \tag{III-1}$$

Since demand is assummed not to be unit-sized, when item i triggers an order, its inventory level is at or below its must-order point s_i. Defining

O_i = average value of the inventory level for item i just before ordering when an order is triggers by itself

R_i = average value of the inventory level for item i just before ordering when it is involved in a joint replenishment triggered by some other item

then, average order size will be (S_i-O_i) when the replenishment is triggered by itself and will be (S_i-R_i) when triggered by another one.

From the characteristic of complete backordering in the inventory system under study, the following demand satisfaction equation must hold for each item:

$$D_i = (NIR_i + NJRB_i)\,(S_i-O_i) + NJRNB_i\,(S_i-R_i). \tag{III-2}$$

Dividing (III-2) by NTR_i,

$$\frac{D_i}{NTR_i} = \frac{NIR_i+NJRB_i}{NTR_i}(S_i-O_i) + \frac{NJRNB_i}{NTR_i}(S_i-R_i)\ . \tag{III-3}$$

If set

$$P_i = \frac{NJRNB_i}{NTR_i}\ , \quad Q_i = 1-P_i = \frac{NIR_i+NJRB_i}{NTR_i}\ ,$$

then, P_i represents the probability that a replenishment involving item i is the result of another item hitting its must-order point and Q_i represents the probability that a replenishment involving item i is triggered by itself.

Substituting the P_i and Q_i in (III-3) and using $NTR_i = \frac{NJRNB_i}{P_i}$ and $S_i - R_i = (S_i - O_i) - (R_i - O_i)$, equation for the demand satisfaction takes the form of

$$\frac{P_i \cdot D_i}{NJRNB_i} = (S_i - O_i) - P_i(R_i - O_i).$$

Solving this equation for $NJRNB_i$

$$NJRNB_i = \frac{P_i\ D_i}{S_i - O_i - P_i(R_i - O_i)} \quad . \qquad \text{(III-4)}$$

Using $NTR_i = \frac{NIR_i + NJRB_i}{1 - P_i}$ and following the similar steps, total number of replenishments for item i triggered by itself is expressed by

$$NIR_i + NJRB_i = \frac{(1 - P_i)\ D_i}{S_i - O_i - P_i(R_i - O_i)} \quad . \qquad \text{(III-5)}$$

Substituting (III-4) and (III-5) into (III-1), the total set-up cost for item i is then:

$$K(i) = \frac{(1 - P_i)\ D_i}{S_i - O_i - P_i(R_i - O_i)}\ KI_i + \frac{P_i\ D_i}{S_i - O_i - P_i(R_i - O_i)}\ KJ_i$$

and

$$K(i) = \frac{D_i}{S_i - O_i - P_i(R_i - O_i)}\ (P_i\ KJ_i + (1 - P_i)\ KI_i) \quad . \qquad \text{(III-6)}$$

Let

E_i = expected demand of item i over lead time L. If item i triggers an order and ordered at inventory level of O_i on the average, inventory level drops to (O_i-E_i) after L units of time and amount of (S_i-O_i) is added to the inventory on-hand to raise it up to level of (S_i-E_i). Then the average inventory carried in an independent cycle will be

$$\frac{(S_i-E_i)+(O_i-E_i)}{2} = \frac{S_i+O_i}{2} - E_i \quad .$$

When a joint replenishment occurs at inventory level R_i, after L units of time, inventory level drops to (R_i-E_i) and receiving (S_i-R_i) units, it raises up to (S_i-E_i). Then, the average inventory carried in a joint cycle will be

$$\frac{(S_i-E_i)+(R_i-E_i)}{2} = \frac{S_i+R_i}{2} - E_i \quad .$$

If it is assumed that the average duration of both independent and joint cycles is the same to eliminate complications, expected inventory carrying cost for item i, H(i), is given by:

$$H(i) = h_i \; [(1-P_i) \; (\frac{S_i+O_i}{2} - E_i) + P_i \; (\frac{S_i+R_i}{2}) - E_i)] \quad ,$$

and

$$H(i) = h_i \; [P_i \; (\frac{S_i+R_i}{2}) + (1-P_i) \; (\frac{S_i+O_i}{2}) - E_i \;] \quad . \qquad \text{(III-7)}$$

Then the total expected cost of the system, which is to be minimized, is given by:

$$Z = \sum_{i=1}^{n} (K(i) + H(i))$$

$$= \sum_{i=1}^{n} [\{\frac{D_i}{S_i-O_i-P_i(R_i-O_i)} [P_i.KJ_i + (1-P_i) KI_i]\}$$

$$+ \{h_i [P_i (\frac{S_i+R_i}{2}) + (1-P_i) (\frac{S_i+O_i}{2}) - E_i]\}]$$

$$= \sum_{i=1}^{n} [P_i \{\frac{KJ_i D_i}{S_i-O_i-P_i(R_i-O_i)} + \frac{S_i+R_i}{2} h_i\}$$

$$+ (1-P_i) \frac{KI_i D_i}{S_i-O_i-P_i(R_i-O_i)} + \frac{S_i+O_i}{2} h_i - E_i h_i]. \quad (III-8)$$

where E_i, demand during lead time L has an expectation of D_iL as given in appendix 1.

Setting

$$\xi_i = S_i-O_i \quad (III-9)$$

and

$$\rho_i = R_i-O_i \quad , \quad (III-10)$$

then

$$\frac{S_i+R_i}{2} = \frac{\xi_i+\rho_i}{2} + O_i$$

and

$$\frac{S_i+O_i}{2} = \frac{\xi_i}{2} + O_i \quad .$$

Using the substitutions above, total cost function can be reformulated as follows:

$$Z = \sum_{i=1}^{n} [P_i \{\frac{KJ_i D_i}{\xi_i - P_i \rho_i} + \frac{\xi_i + \rho_i}{2} h_i\} + (1-P_i) \{\frac{KI_i D_i}{\xi_i - P_i \rho_i} + \frac{\xi_i}{2} h_i \}$$

$$+ O_i h_i - D_i L h_i] \qquad \text{(III-11)}$$

Thus, the mathematical model becomes minimizing the total expected cost function subject to the service level functions which will be introduced in the next section.

III.2.2. CONSTRAINTS - SERVICE LEVEL FUNCTIONS -

Rather than explicitly costing backorders and minimizing stockout costs in objective function together with other relevant costs, safety stocks are determined to supply the demand without stockout above predetermined service levels. Determining the inventory policy parameters the total cost, which is composed of the ordering and the stock holding costs, is optimized subject to the constraint of satisfying these service criteria. Then the optimal values of the inventory policy parameters will garantee the system to operate with stockouts not greater than allowed percents as well as minimizing the expected ordering and the stock holding costs.

The demand during lead time L is analyzed in appendix 1 and stated that it approximately follows a normal distribution with mean $\mu_i = D_i L$ and standart deviation $\nu_i = \sqrt{\frac{D_i L}{m_i} (m_i^2 + \sigma_i^2)}$ for each item i. Standardizing the normal approximation, its cumulative density function will be in the form of

$$\Phi(\frac{t-\mu}{\nu}) = \int_{-\infty}^{t} \frac{1}{\sqrt{2\Pi}} e^{-\frac{1}{2} x^2} dx ,$$

where $-\infty < x < +\infty$ is a random variable and $-\infty < \mu < +\infty$ and $\nu > o$ are the mean and the standart deviation respectively.

In order not to have stock outs from placing an order to receiving it, demand incurred during the lead time period must not exceed the inventory available during that duration. If item i has triggered its must-order point and an order is placed for item i at the average stock level of O_i, then the probability of not running out of item i before the order is received is equal to

$$\Phi(\frac{O_i-\mu_i}{\nu_i})$$

where $\mu_i=D_iL$ and $\nu_i = \sqrt{\frac{D_iL}{m_i}(m_i^2+\sigma_i^2)}$.

Define

Event A_k^i = {Not running out of i at the k^{th} order triggered by itself}

Event A^i = {Not running out of i at all orders triggered by itself in a year}.

The total expected number of orders triggered by i was (NIR_i+NJRB_i). Then

$$A^i = A_1^i \cap A_2^i \cap \ldots \cap A_k^i \cap \ldots \cap A^i_{(NIR_i+NJRB_i)}$$

Since A_k^i's are independent events from each other

$$P(A^i) = P(A_1^i) \cdot P(A_2^i) \ldots P(A_k^i) \ldots P(A^i_{(NIR_i+NJRB_i)}) .$$

It has been assumed above that all A_k^i's have the same probability distribution of $\Phi(\frac{O_i-\mu_i}{\nu_i})$. Using this assumption, probability of event A^i can be expressed as:

$$P(A^i) = \{\Phi(\frac{O_i-\mu_i}{\nu_i})\}^{NIR_i+NJRB_i}$$

Similarly, an approximation to the probability of not running out of i when the order has been triggered by some other item but item i is replenished at the average stock level R_i can be

$$\Phi(\frac{R_i-\mu_i}{\nu_i}) \quad .$$

Define

Event B_k^i = {Not running out of i at the k^{th} order not triggered by i but included}

Event B^i = {Not running out of i at all orders in a year when the orders have been triggered by some other items}

The total expected number of orders of item i triggered by others was $NJRNB_i$.

Using $P(B_k^i)=\Phi(\frac{R_i-\mu_i}{\nu_i})$ and following the same steps and discussions above, probability of event B^i will come out as

$$P(B^i) = \Phi(\frac{R_i-\mu_i}{\nu_i})^{NJRNB_i} \quad .$$

The service criterion for item i, which is the probability of not running out in the whole year, may be expressed by:

Event C^i = {Not running out of item i during the whole year in all replenishments either triggered by i or triggered by other i is included}

$$C^i = A^i \cap B^i$$

Since A^i and B^i are independent events

$$P(C^i) = P(A^i) \; . \; P(B^i)$$

and

$$P(C^i) = \{\Phi(\frac{O_i-\mu_i}{\nu_i})\}^{NIR_i+NJRB_i} \quad \{\Phi(\frac{R_i-\mu_i}{\nu_i})\}^{NJRNB_i} \quad .$$

If the maximum allowed probability of running out for item i is determined as Π_i as an operating policy, the stock of i carried in the system must garantee the satisfaction level of demand of item i with a probability greater than $(1-\Pi_i)$.Then the probabilities of not running out from all items in the whole year should be greater or equal to respective service levels without backordering. This statement can be expressed by the followings:

$$P(C^i) \geq (1-\Pi_i) \qquad i=1,\dots,n$$

$$\left\{\Phi\left(\frac{O_i-\mu_i}{\nu_i}\right)\right\}^{NIR_i+NJRB_i} \cdot \left\{\Phi\left(\frac{R_i-\mu_i}{\nu_i}\right)\right\}^{NJRNB_i} \geq (1-\Pi_i) \qquad i=1,\dots,n.$$

First, using the substitutions (III-9) and (III-10) in (III-4) and (III-5), then replacing their final forms in the inequality above, one will get

$$\left\{\Phi\left(\frac{O_i-\mu_i}{\nu_i}\right)\right\}^{\frac{(1-P_i)D_i}{\xi_i-P_i\rho_i}} \cdot \left\{\Phi\left(\frac{R_i-\mu_i}{\nu_i}\right)\right\}^{\frac{P_i D_i}{\xi_i-P_i\rho_i}} \geq (1-\Pi_i) \qquad i=1,\dots,n.$$

Finally, the service level function may be modified as:

$$\left\{\Phi\left(\frac{O_i-\mu_i}{\nu_i}\right)\right\}^{(1-P_i)} \cdot \left\{\Phi\left(\frac{O_i+\rho_i-\mu_i}{\nu_i}\right)\right\}^{P_i} \geq (1-\Pi_i)^{\frac{\xi_i-P_i\rho_i}{D_i}}$$

$$i=1,\dots,n \qquad \text{(III-12)}$$

Then, the selected values of O_i's have to hold (III-12) to satisfy the respective predetermined service criteria and should be as small as possible to minimize the cost of carrying safety stock.

CHAPTER IV
SOLUTION OF THE INVENTORY PROBLEM

IV.I. GENERAL KNOWLEDGE ABOUT THE SOLUTION PROCEDURE

After having determined the relevant cost parameters and completed the mathematical formulation of the problem, next come to the optimization stage. The optimal values of the three critical inventory levels for each item were to be determined. The proposed mathematical model was:

$$\min Z = \sum_{i=1}^{n} [P_i \{ \frac{KJ_i D_i}{\xi_i - P_i \rho_i} + \frac{\xi_i + \rho_i}{2} h_i \} + (1-P_i) \{ \frac{KI_i D_i}{\xi_i - P_i \rho_i} + \frac{\xi_i}{2} h_i \}$$

$$+ O_i h_i - D_i L h_i]$$

s.to

$$\{\Phi(\frac{O_i - \mu_i}{\nu_i})\}^{(1-P_i)} \cdot \{\Phi(\frac{O_i + \rho_i - \mu_i}{\nu_i})\}^{P_i} \geq (1-\Pi_i)^{\frac{\xi_i - P_i \rho_i}{D_i}} \qquad i=1,\ldots,n$$

$$O_i, \rho_i, \xi_i \geq 0 \qquad i=1,\ldots,n$$

$$0 \leq P_i \leq 1 \qquad i=1,\ldots,n.$$

where

P_i = probability that a replenishment involving item i is the result of another item hitting its must-order point

O_i = average value of the inventory level for item i when an order is triggered by itself

ρ_i = difference between the average inventory levels of item i when it is included in an order triggered by itself and by another.

ξ_i = average order size of item i when the order is triggered by itself.

D_i = expected value of the yearly demand for item i

μ_i = expected value of the lead time demand

ν_i = standard deviation of the lead time demand

KJ_i = ordering cost for item i in the order triggered by another item

KI_i = total ordering cost for item i if the order is triggered by itself

h_i = inventory carrying cost per year per unit of item i

Π_i = maximum allowed probability of running out for item i

n = number of items.

The objective is to determine the must-order points, can-order points and the order-up-to levels for all items optimizing (III-11) with respect to s_i's, c_i's, S_i's . But realizating a mathematical optimization on these parameters would require the knowledge of their functional relationships with involved in model, i.e. the P_i's, O_i's, ρ_i's and ξ_i's . Although a closed-form iteration leading to the values of P_i and R_i in the two-item case with unit sized transactions and Poisson arrivals was obtained by J.P.Schaack [21], there has been met no closed-form results in the n-item compound Poisson demand context. Then, it will not be possible to solve this model by a pure mathematical optimization technique and becomes necessary to approach to this problem by a heuristic algorithm.

An algorithm which is a combination of mathematical optimization and simulation and developped by J.P.Schaack and E.A. Silver [22] was selected for the optimization of the model. Since for given values of s_i, c_i, S_i the corresponding P_i and R_i can not be determined by a mathematical procedure as explained above, simulation will have to be accepted as a part of the algorithm. Simulation is used to determine numerical relationships between the different parameters those can not be related to each other functionally. A simultaneous optimization on s_i, c_i, S_i is not realized by the algorithm, but only alternative updating of s_i, c_i and S_i will be available.

The main steps of the algorithm are:

1) Give the alternative values of the control variables,
   i.e. s_i's, c_i's and S_i's .
2) Observe the behaviour of the system through simulation
3) Make appropriate changes in the control variables according to the observations done in step 2 and by means of some mathematical relationships.

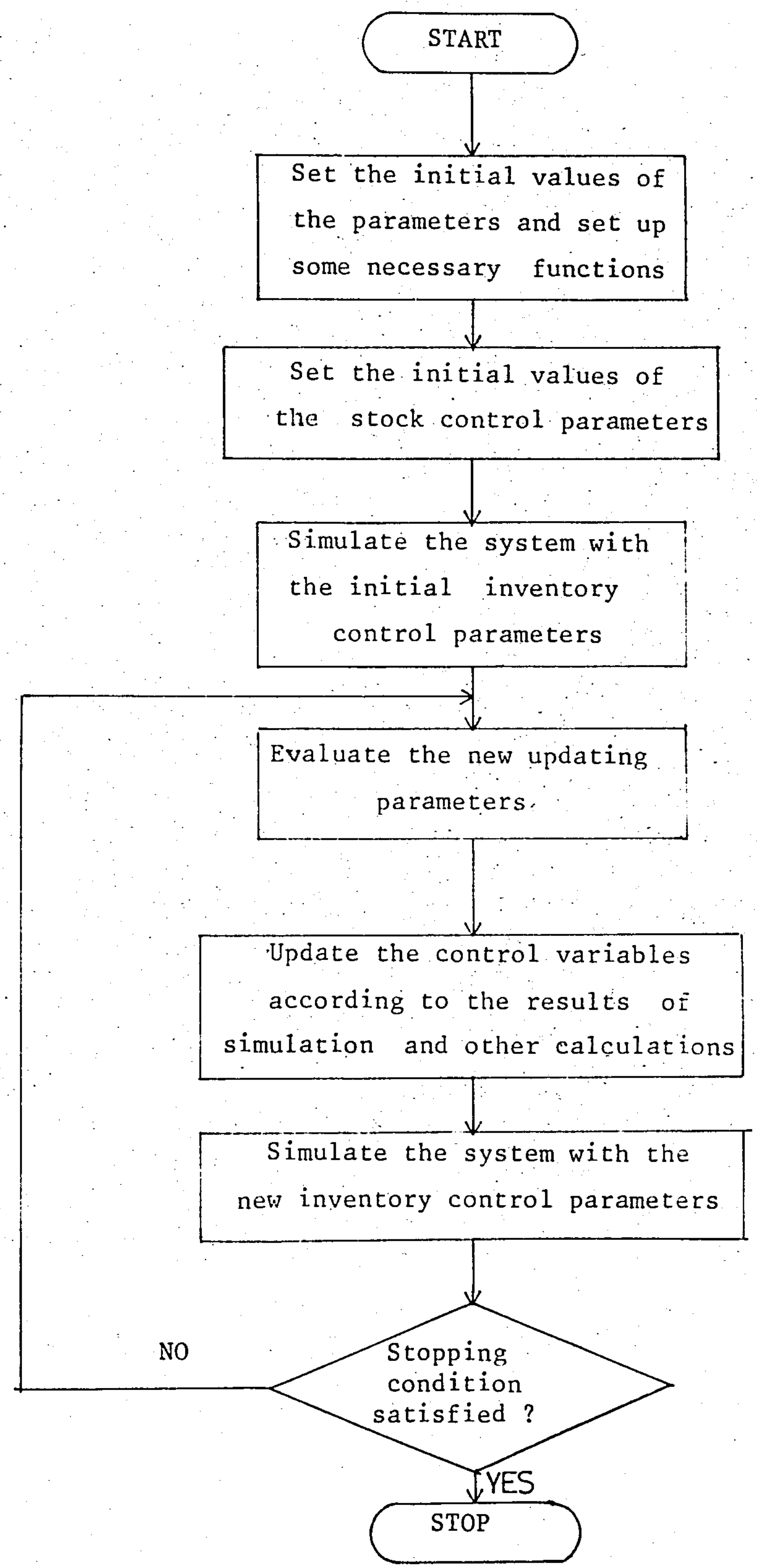


Figure IV.1. Logic Diagram of the Optimization Algorithm

IV.2. UPDATING PROCEDURES FOR THE PARAMETERS

Assume that the state of the system is given. This means that the inventory policy parameters are fixed and the P_i's and R_i's are already observed. In the following sub-sections, it is explained how a new state will be passed realizing seperate optimizations on s_i's, c_i's and S_i's.

IV.2.1. UPDATING MUST-ORDER POINTS, s_i's

Must-order points, s_i's, are not directly involved in the objective function and the constraints, but are represented by the O_i's. Then if one can determine the O_i's and relate them to s_i's somehow, the updating procedure of the must-order points will be completed. In the model, it can be easily observed that each O_i will be pushed up to satisfy the desired service level in the respective constraint and will be pushed down to minimize the stock holding and the ordering costs in the objective function. Then, for predetermined values of P_i, ρ_i and ξ_i, O_i will be chosen as the smallest value of O_i which satisfies the respective service level function (III-12) for each item.

$$\{\Phi(\frac{O_i-\mu_i}{\nu_i})^{(1-P_i)} \cdot \{\Phi(\frac{O_i+\rho_i-\mu_i}{\nu_i})\}^{P_i} \geq (1-\Pi_i)^{\frac{\xi_i-P_i\rho_i}{D_i}}$$

IV.2.1.1. Methods for Solving the Service Level Function (III-12)

Inequality (III-12) has to be solved in terms of O_i for each updating on s_i. But because of its complex form, algebraic solution to this function does not seem possible. Some methods in numerical analysis are examined for this study and two of them are presented below. Then, a procedure based on the last one is developed and used to solve (III-12) in the study.

IV.2.1.1.1. One-Point Iterative Method

If an equation can be rewritten in the form of

$$x = t\,(x) \qquad \text{(IV-1)}$$

then it can be found an x as the limit of the sequence

$$x_1,\ x_2=t(x_1)\ ,\ \ x_3=t(x_2)\ ,\ \ x_4=t(x_3),\ldots.$$

it is clear that if x_1 is selected so that the sequence x_1, x_2, x_3,... does converge to a value r, then r is the root of (IV-1) .

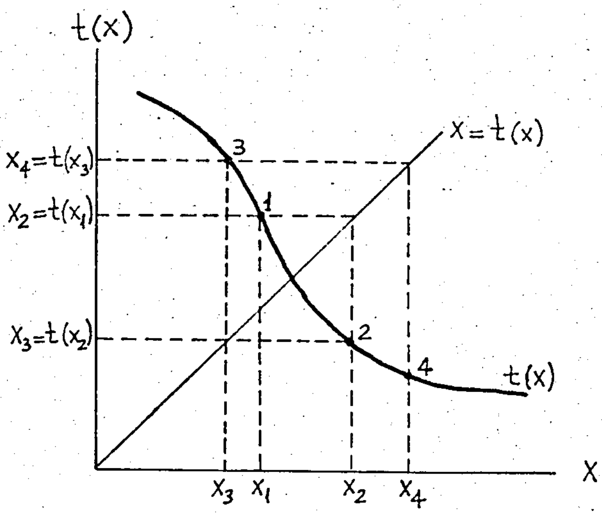


Figure IV.2. An Example of the Diverging Case

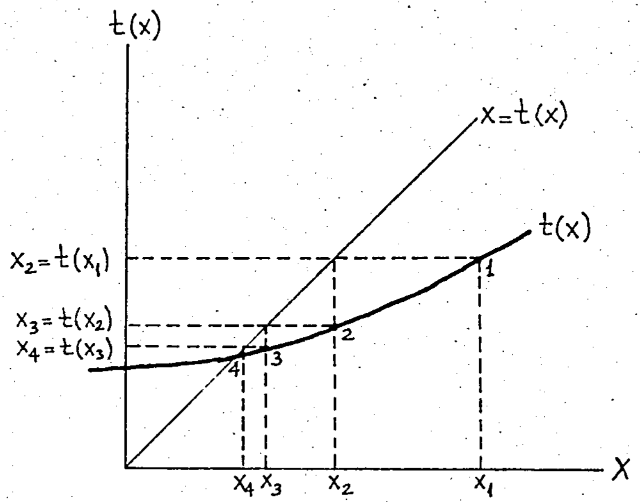


Figure IV.3. An Example of the Converging Case

This method is named as "One-Point Iterative Method" in [18], because we have only single point at each stage different from other methods of finding roots such as the regula falsi or Newton's methods. The same method is referred as "Iterative Procedure" in [24], whereby the answer at any stage is substituted in the given function iteratively to produce the next number in the sequence. In Fig.IV.2. and IV.3.,the diverging and converging cases are illustrated on graphs.

To use one-point iterative method for the solution of (III-12), the service level function can be rewritten in the form (IV-1). Then:

$$O_i = t(O_i)$$

$$\{\Phi(\frac{O_i - \mu_i}{\nu_i})\} \geq \left[\frac{(1-\Pi_i)^{\frac{\xi_i - P_i \rho_i}{D_i}}}{\{\Phi(\frac{O_i + \rho_i - \mu_i}{\nu_i})\}^{P_i}} \right]^{\frac{1}{1-P_i}}$$

t is well known that for the case of continuous variable, O_i will be the solution f equality minimizing the cost of carrying safety stocks. Multiplying both ides by the inverse of normal cumulative density function Φ^{-1} and making the ecessary simplifications, (III-12) will turn into the form (IV-1) as:

$$O_i = \{\Phi^{-1} [(1-\Pi_i)^{\frac{\xi_i - P_i \rho_i}{D_i(1-P_i)}} / \{\Phi(\frac{O_i + \rho_i - \mu_i}{\nu_i})\}^{\frac{P_i}{1-P_i}}]\} \nu_i + \mu_i$$

tarting with an initial value of O_i and iterating $O_{in} = t(O_{i,n-1})$ we may come to he solution in a few iterations, but the convergency must be garanteed before sing this method. $\Phi(x)$ is often included in a computer library or can be rogrammed by using one of the integration methods. $\Phi^{-1}(y)$ can also be rogrammed by solving $y=\Phi(x)$ for x by means of the Newton method [20] or by olynomial approximation [3].

V.2.1.1.2. The Regula Falsi (Linear Interpolation)

ne of the oldest methods for computing the real roots of a numerical equation s the method of 'Regula Falsi' or 'Falso Position' [18]. It requires a nowledge of the approximate location of the root and the values of the function t two points, $y_1=f(x_1)$ and $y_2=f(x_2)$, where $x_1<r<x_2$, r being a root of $f(x)=0$. f x_1 and x_2 are close enough to r, the function f(x) is continuous in $x_1<x<x_2$. ince the root lies between x_1 and x_2, the function $y=f(x)$ must cross the -axis between x_1 and x_2. Then $y_1=f(x_1)$ and $y_2=f(x_2)$ must have opposite signs.

ince any portion of a smooth curve is practically straight for a short distance, t may be assumed that the change in f(x) is proportional to the change in x ver a short interval, as in the case of linear interpolation from logarithmic

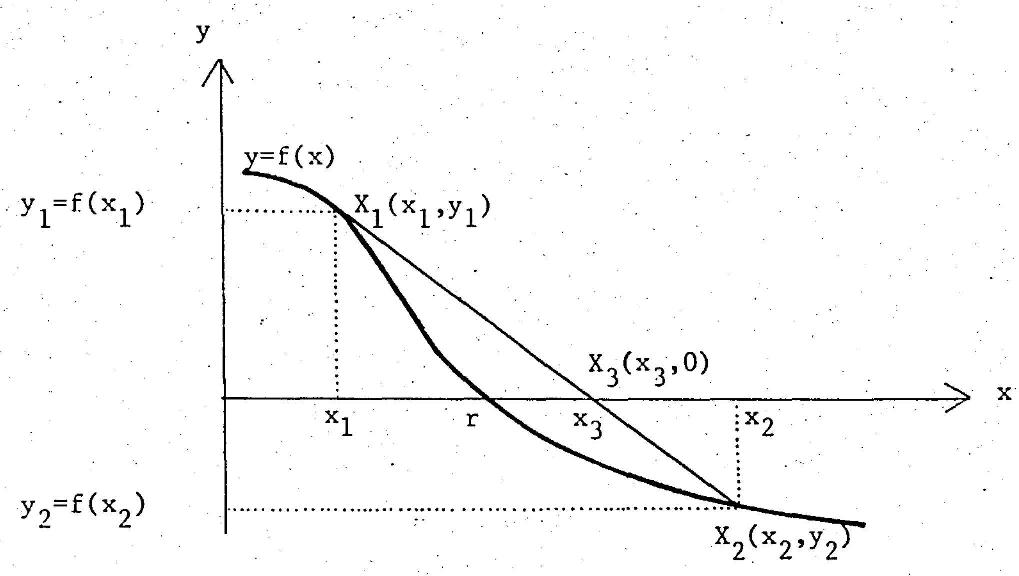


Figure IV.4. The Regula Falsi Method

nd trigonometric tables. The method of falso position is based on this principle nd called also as linear interpolation in [18] and [19] .

o derive a formula for computing the root, the arc X_1X_2 is replaced by a nique straight line which passes through the points (x_1, f(x_1)) and (x_2,f(x_2)). ecalling some coordinate geometry, the equation of this straight line may be ritten as

$$f(x) - f(x_1) = \frac{f(x_2) - f(x_1)}{x_2 - x_1}(x - x_1) \quad . \qquad \text{(IV-2)}$$

ne intersection of the straight line and the x-axis, which is X_3, will have ne abscissa

$$x_3 = \frac{x_1y_2 - x_2y_1}{y_2 - y_1} \qquad \text{(IV-3)}$$

is value of x is not however the true value of the root, because the function f(x) is not a perfect straight line between the points X_1 and X_2. It is merely a closer approximation than x_1 and x_2 to the root. The process may then be repeated using the arch X_1X_3. In the practical application of linear interpolation, short tables showing the corresponding values of x and f(x) and also by means of (IV-3), corrections to be applied to the previously obtained approximate values may be prepared.

IV.2.1.1.3. Procedure Developed in the Study

In this study, the Regula Falsi method is adopted somewhat and used to solve the service level function (III-12). Without using the subscripts and setting the right side of (III-12) to Δ, solution of this inequality will be nothing but the intersection of two functions

$$f^1(x) = \{\Phi(\frac{x-\mu}{\nu})\}^{(1-P)} \quad \{\Phi(\frac{x+\rho-\mu}{\nu})\}^{P}$$

and

$$f^2(x) = \Delta$$

where

$$\Delta = (1-\pi)^{\frac{\xi-P\rho}{D_i}}$$

In the adopted Regula Falsi method, instead of looking for the root of a function, i.e. its intersection with function y=0, the intersection of two functions f^1 and f^2 will be searched.

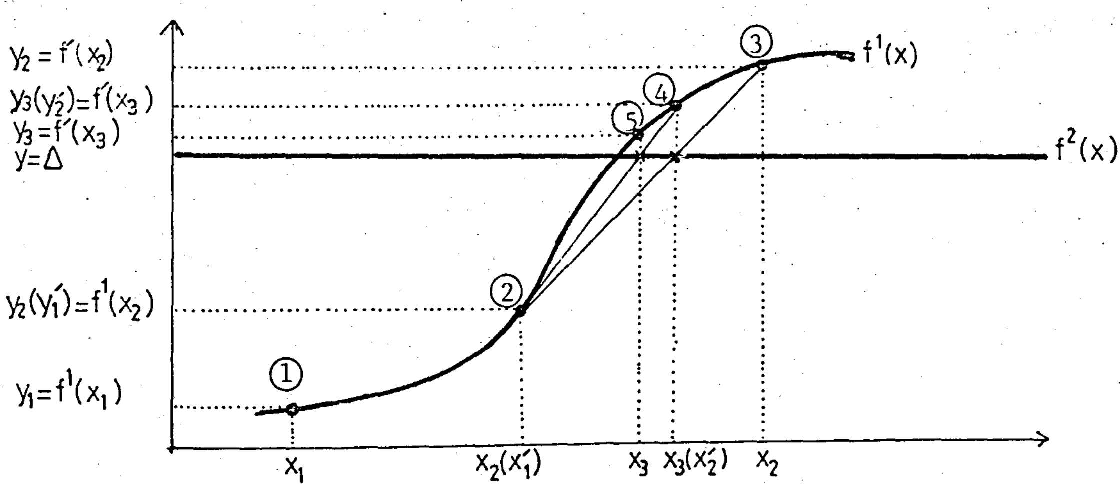


Figure IV.5. Graphical Representation of the Procedure Developed to solve (III.12)

The equation of the straight line between points (x_1,y_1) and (x_2,y_2) is still in form (IV-2). The point, where the straight line and f^2 have the same value, can be obtained setting these two functions in equal. Then, substituting Δ in place of y in (IV-2), the abscissa of the next point will be

$$x_3 = \frac{x_1y_2-y_1x_2-(x_1-x_2)}{y_2-y_1}$$

This updating procedure will continue until you are close enough to the true value.

It is obvious that the function f^1 is monoton, non-decreasing and non-negative because both terms are the cumulative density functions for normal distribution. Then the following algorithm is developed using the Regula Falsi method and these characteristics of f^1.

ITERATIVE ALGORITHM

1) Calculate Δ and choose a positive step size of Δx

2) Choose a starting point X_1 and compute $y_1=f^1(x_1)$, $\Delta y_1=y_1-\Delta$
   If $\Delta y_1>0$ set $\Delta x=-\Delta x$

3) Compute the next point X_2:
   $x_2=x_1+\Delta x$
   $y_2=f^1(x_2)$
   $\Delta y_2=y_2-\Delta$

4) Control the interval (x_1-x_2) for the searched point:
   If $(\Delta y_1 * \Delta y_2)>0$, intersection point is not between these two points.

   Replace point X_2 by X_1:

   $x_1=x_2$

   $y_1=y_2$

   $\Delta y_1=\Delta y_2$

   Go to 3

   Otherwise, continue

The desired point lies between X_1 and X_2 . Select the new point.

$$x_3 = \frac{x_1y_2-y_1x_2-\Delta(x_1-x_2)}{y_2-y_1}$$

$y_3 = f^1(x_3)$

$\Delta y_3 = y_3 - \Delta$

If its accuracy is accepted, stop. X_3 is the desired solution. Otherwise, continue

Reduce the selected interval.

If $(\Delta y_1 * \Delta y_3) > 0$, intersection point is between X_2 and X_3 .

Replace point X_3 by X_1:

$x_1 = x_3$

$y_1 = y_3$

$\Delta y_1 = \Delta y_3$

Go to 5

Otherwise, intersection point is between X_1 and X_3 .

Replace point X_3 by X_2 :

$x_2 = x_3$

$y_2 = y_3$

$\Delta y_2 = \Delta y_3$

Go to 5.

algorithm may not seem to be able to find the solution in a short time, in the whole optimization algorithm, an approximate value of 0 is ready by previous iteration. Then, using this value as the starting point, it will possible to use this search algorithm with an higher efficiency.

IV.2.1.2. Method Used for Computing the cdf Function Φ: Simpsons $\frac{1}{3}$ Rule Formula

Evaluating $f^1(x)$ in the previous section, Φ, the cumulative density function of normal distribution, has to be known. In order to calculate this function methods of numerical integration are examined and it has met lots of methods in this area such as rectangular quadrature rule, the trapezoidal rule,Gregory's formula, Simpson's rule, Weddle's Rule etc. [18], [24]. And among these methods, Simpson's Rule with the simplicity and fairly high accuracy is selected to be used in the evaluation of the function Φ.

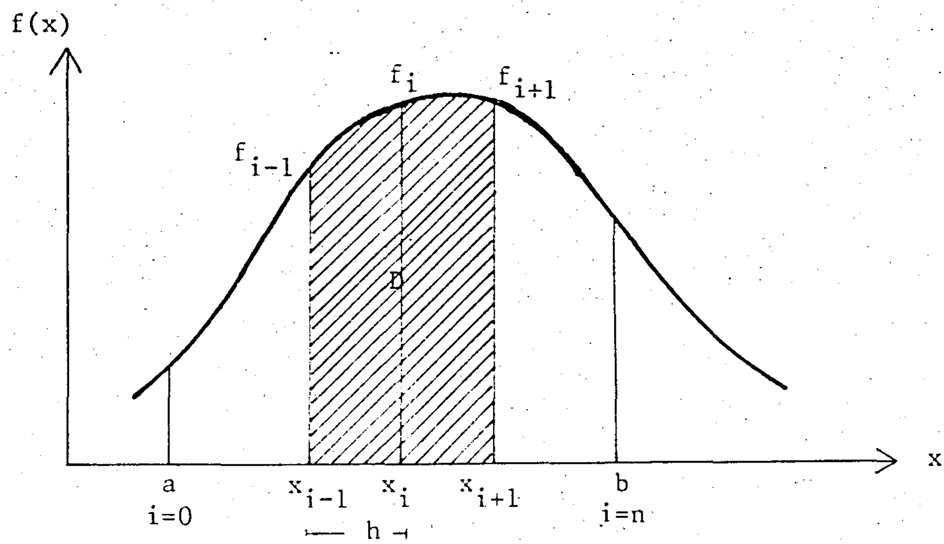


Figure IV.6.

Given the pivotal values of the function f(x) which is to be integrated:

$$\ldots, f_{i-2}, f_{i-1}, f_i, f_{i+1}, f_{i+2}, \ldots$$

evenly spaced by h, the area under the integrand between x_{i-1} and x_{i+1} can be approximated by the area under the parabola passing through points (x_{i-1}, f_{i-1}), (x_i, f_i) and (x_{i+1}, f_{i+1}). Then that parabola will have the formula of $f(x)=Ax^2+Bx+C$ with parameters

$$A = \frac{f_{i+1}-2f_i+f_{i-1}}{2\,h^2}, \quad B = \frac{f_{i+1}-f_{i-1}}{2\,h}, \quad C=f_i .$$

The area under this parabola between x_{i-1} and x_{i+1} is given

$$D = \int_{x_{i-1}}^{x_{i+1}} f(x)dx = \frac{h}{3}(f_{i+1} + 4f_i + f_{i-1}) .$$

This result is known as Simpson's $\frac{1}{3}$ rule formula for the area under two strips of width h.

In general, $\int_a^b f(x)dx$ represents the area from a to b under the curve $y=f(x)$. But if $f(x)$ is not a quadratic function, the graph will not usually be a parabola. However, if we split the interval (a, b) into an even number, n, of parts, then f(x) may be accurately represented by different parabolas for each pair of consecutive intervals. Applying the Simpsons Rule $\frac{n}{2}$ times, which means the graph of the given function is replaced by $\frac{n}{2}$ arcs of parabolas, yields the so-called General Simpson's Rule:

$$\int_a^b f(x)dx \doteq \frac{h}{3}(f_o + 4f_1 + 2f_2 + 4f_3 + \ldots + 2f_{n-2} + 4f_{n-1} + f_n)$$

This formula can also be written in the general form:

$$\simeq \frac{h}{3} \sum_{i=0}^{n} cf_i \quad \text{where} \quad c = 1,4,2,\ldots,2,4,1 .$$

Eventhough Simpson's Rule can be used for definite integrals, from the peculiarities of the probability density function for normal distribution, $\psi(x)$, this method is used to evaluate $\phi(z) = \int_{-\infty}^{z} \psi(x)dx$.

IV.2.1.3. Updating Procedure

If demand size for each item has mean m_i and standard deviation σ_i, then the difference between the must-order point and the average inventory level as soon as the inventory drops below s_i has an expected value of $\alpha_i = \frac{1}{2}\left(\frac{m_i^2+\sigma_i^2}{m_i}\right)$. Proof of this statement is given in Appendix II. Also in section IV.2.1.1., the minimum values of O_i's which must be carried to satisfy the desired service levels were determined solving (III-12). Then, the must-order point will be higher than the average value of the inventory when the order is placed, by the expected difference between s_i and O_i. For the determined value of O_i, the new s_i in the next iteration will be updated as

$$s_i = O_i + \frac{1}{2}\left(\frac{m_i^2+\sigma_i^2}{m_i}\right) \qquad i=1,\ldots,n \;.$$

IV.2.2. UPDATING CAN-ORDER POINTS, c_i's

IV.2.2.1. Derivative of the Objective Function with Respect to c_i

The c_i does not appear in the objective function and the constraints, but eventhough the functional relationships between c_i and P_i, ρ_i are not clear, we know that R_i depends on ρ_i by (III-10) and P_i also depends on c_i somehow. Then, ignoring the other dependencies, differentiation of (III-11) with respect to c_i gives us:

$$\frac{\partial Z}{\partial c_i} = \frac{\partial P_i}{\partial c_i}\left[\frac{KJ_i\, D_i}{\xi_i - P_i\rho_i} + \frac{\xi_i+\rho_i}{2}\, h_i\right] + P_i\left[-\left(\frac{-\partial P_i\rho_i}{\partial c_i}\right)\frac{KJ_i\, D_i}{(\xi_i-P_i\rho_i)^2} + \frac{\partial\rho_i}{\partial c_i}\,\frac{h_i}{2}\right]$$

$$+\left(-\frac{\partial P_i}{\partial c_i}\right)\left[\frac{KI_i\, D_i}{\xi_i-P_i\rho_i} + \frac{\xi_i}{2}\, h_i\right] + (1-P_i)\left[-\left(\frac{-\partial P_i\rho_i}{\partial c_i}\right)\frac{KI_i\, D_i}{(\xi_i-P_i\rho_i)^2}\right]$$

The derivative function may be simplified as:

$$= \left(\frac{-KF\ D_i}{\xi_i - P_i\rho_i}\right)\frac{\partial P_i}{\partial c_i} + \frac{h_i}{2}\left(\rho_i\frac{\partial P_i}{\partial c_i} + P_i\frac{\partial \rho_i}{\partial c_i}\right) + \frac{D_i}{(\xi_i - P_i\rho_i)^2}(P_i KJ_i + (1-P_i)KI_i)\frac{\partial (P_i\rho_i)}{c_i}$$

By chain rule, $\dfrac{\partial (x\ y)}{\partial \Theta} = x\dfrac{\partial y}{\partial \Theta} + y\dfrac{\partial x}{\partial \Theta}$.

Then , $\rho_i\dfrac{\partial P_i}{\partial c_i} + P_i\dfrac{\partial \rho_i}{\partial c_i} = \dfrac{\partial (P_i\rho_i)}{\partial c_i}$

and by (IV-6), $D_i(P_i KJ_i + (1-P_i)KI_i) = \dfrac{h_i}{2}(\xi_i - P_i\rho_i)^2$ ..

Substituting these values into the derivative function, it will be

$$\frac{\partial Z}{\partial c_i} = -\frac{KF\ D_i}{\xi_i - P_i\rho_i}\frac{\partial P_i}{\partial c_i} + \frac{h_i}{2}\frac{\partial (P_i\rho_i)}{\partial c_i} + \frac{(h_i/2)(\xi_i - P_i\rho_i)^2}{(\xi_i - P_i\rho_i)^2}\frac{\partial (P_i\rho_i)}{\partial c_i} .$$

It will finally be reduced to

$$\frac{\partial Z}{\partial c_i} = h_i\frac{\partial (P_i\rho_i)}{\partial c_i} - \frac{KF\ D_i}{\xi_i - P_i\rho_i}\frac{\partial P_i}{\partial c_i} \qquad i=1,\ldots,n. \qquad \text{(IV-4)}$$

The optimal value of c_i must satisfy

$$\frac{\partial Z}{\partial c_i} = 0 \qquad i=1,\ldots,n.$$

IV.2.2.2. Iterative Algorithm to Update c_i's

Since the functional relationships between P_i, ρ_i and c_i are not known, it can not be obtained a closed-form solution for the optimal c_i. Instead, performing small changes on c_i until the derivative approaches zero, it will be tried to catch the optimal values.

By the definition of the derivative [12], if $\Delta\Theta$ is small enough, the following approximation can be done:

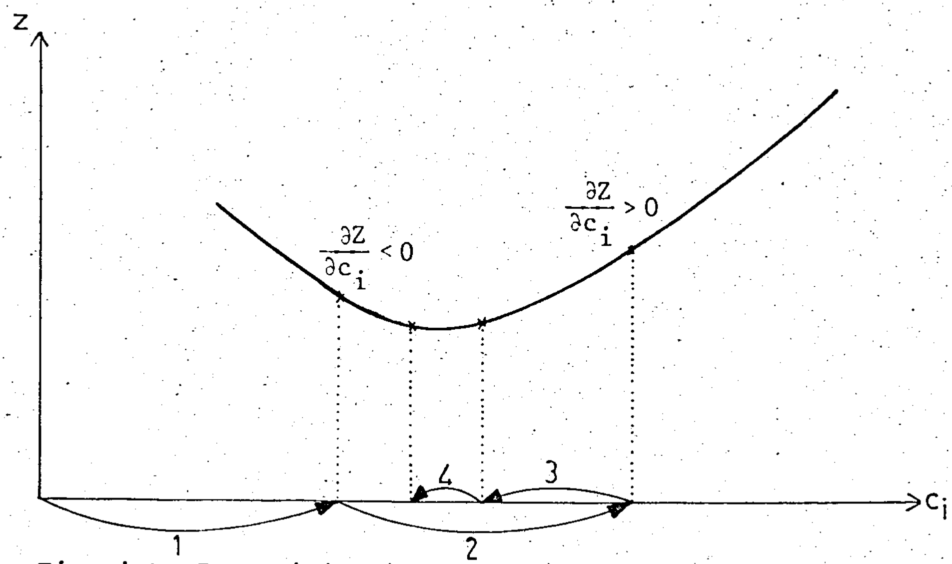
$$\frac{df}{d\Theta} \approx \frac{\Delta f}{\Delta\Theta} = \frac{f(\Theta+\Delta\Theta) - f(\Theta)}{\Delta\Theta}$$

If Δc_i is chosen small enough, the same approximation can be accepted for:

$$\frac{\partial P_i}{\partial c_i} \approx \frac{\Delta P_i}{\Delta c_i} \quad \text{and} \quad \frac{\partial(P_i\rho_i)}{\partial c_i} \approx \frac{\Delta(P_i\rho_i)}{\Delta c_i} .$$

When we know the values P_{io} and ρ_{io} for the value c_{io} and similarly P_{i1} and ρ_{i1} for the value c_{i1}, if $(c_{i1}-c_{io})$ is small enough, (IV-4) can be approximated by

$$\frac{\partial Z}{\partial c_i} \approx h_i \frac{P_{i1}\rho_{i1} - P_{io}\rho_{io}}{c_{i1} - c_{io}} - \frac{KF\ D_i}{\xi_i - P_{io}\rho_{io}} \frac{P_{i1} - \rho_{io}}{c_{i1} - c_{io}} . \qquad \text{(IV-5)}$$



Figuri IV.7. Optimization of c_i

If $\frac{\partial Z}{\partial c_i} > 0$, it means that the objective function increases as c_i increases, then c_i must be reduced.

If $\frac{\partial Z}{\partial c_i} < 0$, the objective function decreases as c_i increases, then c_i must be increased.

Then the value of c_i must be corrected according to the sign of $\frac{\partial Z}{\partial c_i}$. The term γ_i is defined as the difference between c_i and O_i, and it is used to define c_i when O_i is known. At each iteration, the γ_i is also corrected by another parameter $STEP_i$ which is updated according to the increase or decrease in Z_i: In order to converge to the optimal c_i, the $STEP_i$ is cut by two each time as soon as Z_i increases after having first decreased. The main steps of the updating procedure are

i) if $\frac{\partial Z}{\partial c_i} > 0$ Reduce γ_i by $STEP_i$; $\gamma_i = \gamma_i - STEP_i$

if $\frac{\partial Z}{\partial c_i} < 0$ Increase γ_i by $STEP_i$; $\gamma_i = \gamma_i + STEP_i$

and

$c_i = O_i + \gamma_i$

ii) At the end of the each iteration, if you are in the optimal interval, reduce the change on γ_i and get closer to the optimum.

If $Z_i \geq Z_{i_{previous}}$ Divide $STEP_i$ by 2.

If $Z_i < Z_{i_{previous}}$ Continue with the previous $STEP_i$.

While updating c_i's in the algorithm, if the value of $STEP_i$ is too large, the approximations made above

$$\frac{\partial P_i}{\partial c_i} \simeq \frac{\Delta P_i}{\Delta c_i} \quad \text{and} \quad \frac{\partial (P_i \rho_i)}{\partial c_i} \simeq \frac{\Delta (P_i \rho_i)}{\Delta c_i}$$

are no longer valid. But if $STEP_i$ is too small, the algorithm will require too many iterations to reach to the optimal region. Taking both into account, $EOQ_i/10$ is not a bad assumption as the starting value of $STEP_i$.

IV.2.3. UPDATING ORDER-UP-TO LEVELS, S_i's

If O_i is known in any iteration, S_i and ξ_i hold the same information by $\xi_i = S_i - O_i$. Then they can be used interchangeably in the optimization of the total cost function. Differentiating the objective function (III-11) with respect to ξ_i gives us

$$\frac{\partial Z}{\partial \xi_i} = P_i \left(\frac{-KJ_i \; D_i}{(\xi_i - P_i \rho_i)^2} + \frac{h_i}{2} \right) + (1-P_i) \left(\frac{-KI_i \; D_i}{(\xi_i - P_i \rho_i)^2} + \frac{h_i}{2} \right)$$

where the changes in P_i and ρ_i due to changes in ξ_i, i.e. $\frac{\partial P_i}{\partial \xi_i}$ and $\frac{\partial \rho_i}{\partial \xi_i}$ are neglected.

Setting the result equal to zero and solving for ξ_i leads to

$$-\frac{D_i}{(\xi_i-P_i\rho_i)^2}(P_iKJ_i+(1-P_i)KI_i)+\frac{h_i}{2}=0$$

$$h_i\xi_i^2 \; -2h_iP_i\rho_i\xi_i + h_iP_i^2\rho_i^2 -2D_i(P_iKJ_i + (1-P_i)KI_i) = 0$$

$$\xi_i = \frac{h_iP_i\rho_i \pm \sqrt{(h_iP_i\rho_i)^2-h_i[h_iP_i^2\rho_i^2 -2D_i(P_iKJ_i + (1-P_i)KI_i)]}}{h_i}$$

$$\xi_i = P_i\rho_i \pm\sqrt{\frac{2\,D_i\,(P_iKJ_i + (1-P_i)KI_i)}{h_i}}\,. \qquad \text{(IV-6)}$$

Positive second-order derivative guarantees the global minimum:

$$\frac{\partial Z}{\partial\xi_i^2} = \frac{2D_i}{(\xi_i-P_i\rho_i)^3}(P_iKJ_i + (1-P_i)KI_i) \geq 0$$

where $\xi_i = S_i-O_i$, $\rho_i=R_i-O_i$, $O_i\leq R_i\leq S_i$ $\rho_i \leq \xi_i$,

$\rho_i \leq \xi_i$, $0 \leq P_i \leq 1$ $\rightarrow$ $(\xi_i-P_i\rho_i) \geq 0$.

The same result, $(\xi_i-P_i\rho_i)\geq 0$, eliminates also the negative sign in the formula (IV-6). When P_i and ρ_i are known, the optimal value of ξ_i will be then

$$\xi_i^* = P_i\rho_i + \sqrt{\frac{2D_i}{h_i}(P_iKJ_i + (1-P_i)KI_i)} \qquad \text{(IV-7)}$$

If $P_i=0$, it means that only independent replenishments occur and the ordering cost is KI_i at each time. The same formula will be relevant for each independent item and ξ_i will be nothing but the economic order quantity in [10].

$$\text{If } P_i=0, \quad \xi_i = \sqrt{\frac{2\, D_i\, KI_i}{h_i}} = EOQ_i \; .$$

When the optimal O_i is determined, S_i will be updated by the following:

$$S_i = O_i + \xi_i^* \; .$$

IV.3. STEPS OF THE OPTIMIZATION ALGORITHM

The algorithm used for the optimization of the problem is an iterative one. In the algorithm, the parameters are updated at each iteration according to the rules described in the previous section. It is continued until an acceptable result is obtained. The algorithm proceeds as follows:

1- Select $STEP_i$ for each item.

2- Start with the initial values of the parameters assuming independent system. i.e. $P_{io}=0$:

i) $s_i = O_{io} + \alpha_i$

where $\alpha_i = \frac{1}{2} \left(\frac{m_i^2 + \sigma_i^2}{m_i} \right)$

and O_{io} is the safety stock of item i in an independent system guaranteeing the desired service level. Therefore O_{io} will be the solution of (III-12) with $P_i=0$:

$$\{\Phi(\frac{O_{io}-\mu_i}{\gamma_i})\} \geq (1-\Pi_i)^{\frac{\xi_i^*}{D_i}}$$

ii) $c_i = O_{io} + \gamma_i$

where γ_i=STEP$_i$ as explained in (IV.2.2.2.)

O_{io} as in (2-i) .

iii) $S_i = O_{io} + \xi_i$

where ξ_i is calculated by (IV-7) with P_i=0:

$$\xi_i = \sqrt{\frac{2\ D_i\ KI_i}{h_i}} = EOQ_i$$

O_{io} as in (2-i) .

3- Simulate the system with the initial values of S_i's, c_i's and s_i's and determine $NJRB_i$, $NJRNB_i$, NTR_i, R_i, O_i .
Then calculate

$$P_i = \frac{NJRB_i + NJRNB_i}{NTR_i}, \quad o_i = R_i - O_i$$

4- Calculate Z_{io}, the objective function values by (III-11)

and set $Z_o = \sum_i Z_{io}$

---

* ξ_i is equal to EOQ_i for P_i=0. Then O_{io} will be the solution of

$$\{\Phi(\frac{O_{io}-\mu_i}{\gamma_i})\} (1-\Pi_i)^{\frac{EOQ_i}{D_i}}$$

5- Determine the new values for updating the parameters

i) Determine new O_i solving (III-12) with the new values of P_i and ρ_i

ii) Evaluate $\frac{\partial Z}{\partial c_i}$ by (IV-5)

$$\text{if} \quad \frac{\partial Z}{\partial c_i} < 0 \qquad \gamma_i = \gamma_i + STEP_i$$

$$\text{if} \quad \frac{\partial Z}{\partial c_i} > 0 \qquad \gamma_i = \gamma_i - STEP_i$$

iii) Determine new ξ_i solving (IV-6) with the new values of P_i and ρ_i .

6- Update the parameters with the new values determined in step (5).

$$s_i = O_i + \alpha_i$$

$$c_i = O_i + \gamma_i$$

$$S_i = O_i + \xi_i$$

7- Simulate the system with the new values of s_i's, c_i's and S_i's in order to find the corresponding values of the P_i's and ρ_i's necessary for the next updating.

8- Calculate the new objective function values Z_i's by (III-11)

If $Z_i \geq Z_{io}$ Cut the $STEP_i$ by 2 $STEP_i = STEP_i/2$

If $Z_i < Z_{io}$ Continue

9- If stopping condition is achieved, stop
Otherwise set $Z_{io} = Z_i$ and go to 5.

IV.4. COMPARISON OF THE PROPOSED JOINT ORDERING POLICY WITH INDEPENDENT POLICIES AND THE STOPPING RULE

One important practical point to know before introducing an (s,c,S) policy in a particular context is whether or not the savings accomplished by the change offset the cost of implementation. A necessary step then will be the examination of the maximum possible saving realized by a joint ordering policy over the usual continuous review independent inventory control system. The comparison of the joint and independent policies shows us what is the best position which can be achieved by a joint policy. The difference between the most possible improvement and the improvement which has been reached will give a criterion to decide where we can stop, i.e. the stopping condition for the algorithm.

For an inventory system with n items and dependent ordering cost, the worst policy will be to control this system by n independent policies. Among the independent policies, the best thing will be continuous review (s, S) policy . Then Z_I, the maximum cost which may be occurred controlling the system, will be the sum of all costs for a system allowing no joint replenishment, i.e. $P_i=0$. It may be calculated by

$$Z_I = \sum_i [\frac{D_i}{EOQ_i} KI_i + (\frac{EOQ_i}{2} + O_i^* + D_i L)h_i] \qquad \text{(IV-8)}$$

where O_i^* is the safety stock in the independent system.

The answer to the question of what a joint policy can do the best is given in the following.For a joint policy, the total expected cost function (III-11) can be rewritten as the sum of the expected ordering cost in terms of the number of orderings (III-1) and the expected inventory carrying cost. After some substitutions and simplifications:

$$Z = \sum_i [(NIR_i + NJRB_i)(KF + KJ_i) + NJRNB_i\, KJ_i$$

$$+ (\xi_i + P_i \rho_i) \frac{1}{2} h_i + O_i\, h_i - D_i\, L\, h_i]$$

One lower boundary of the total cost function will be obtained setting $NIR_i=0$, i.e. no independent replenishments will occur. Therefore

$$Z \geq \sum_i [NJRB_i(KF+KJ_i) + NJRNB_i\ KJ_i + (\xi_i+P_i\rho_i)\frac{1}{2}h_i+O_ih_i-D_i\ L\ h_i]$$

or

$$Z \geq \sum_i [NJRB_i\ KF+(NJRB_i+NJRNB_i)KJ_i+(\xi_i+P_i\rho_i)\frac{1}{2}h_i+O_ih_i-D_i\ L\ h_i] \quad (IV-9)$$

The lowest cost in this last inequality represents the cost of a system operating without any independent replenishments, in which case all replenishments occur jointly and $(NJRB_i+NJRNB_i)$ represents the total number of replenishments for item i per annum. Hence, by substituting (III-9) and (III-10) into (III-4) and (III-5) then summing up two:

$$NJRB_i + NJRNB_i = \frac{D_i}{\xi_i-P_i\rho_i}$$

and where $KI_i \geq KJ_i$ and $0 \leq P_i \leq 1$ for all i

$$\sqrt{\frac{2D_i(P_iKJ_i+(1-P_i)KI_i)}{h_i}} \leq \sqrt{\frac{2D_i\ KI_i}{h_i}} = EOQ_i$$

It can be seen from (IV-6) that $\xi_i-P_i\rho_i \leq EOQ_i$. Therefore a lower bound for the total number of replenishments where no independent replenishment occurs is

$$NJRB_i + NJRNB_i \geq \frac{D_i}{EOQ_i}. \quad (IV-10)$$

On the other hand $\sum_{i=1}^{n} NJRB_i$ represents how many times an order is triggered in a year, that means the total number of replenishments for the system during the whole year. The minimum number of replenishments for the whole system has to be at least equal to the number of replenishments of the fastest moving item.

$$\sum_i NJRB_i \geq \max_i (NJRB_i + NJRNB_i) = \max_i \left(\frac{D_i}{\xi_i - P_i \rho_i}\right) \geq \max_i \left(\frac{D_i}{EOQ_i}\right). \qquad \text{(IV-11)}$$

Then, substituting the lower bounds of the number of replenishments for each item and for the whole system into (IV-9), the smallest value of the lower bound on total cost function becomes

$$Z \geq \max_i \left(\frac{D_i}{EOQ_i}\right) KF + \sum_i \left[\frac{D_i}{EOQ_i} KJ_i + (\xi_i + P_i \rho_i)\frac{1}{2} h_i + O_i h_i - D_i\ L\ h_i\right] .$$

First two terms are the total ordering cost for the system and the others represent the inventory carrying cost. Then the expression

$$\left(\frac{\xi_i\ P_i \rho_i}{2} + O_i - D_i\ L\right)$$

represents the average inventory of item i under a joint replenishment policy. The joint ordering policy gives an overall saving in the total cost decreasing the total number of orderings thus the ordering cost, but it makes necessary to carry more inventory than carried in the independent policy ordering items more frequently. Then the average inventory level carried under an independent system gives a lower bound for the inventory carried in the dependent system. Finally, a lower bound for Z in a joint policy can be expressed using the approximations above:

$$Z_L = \max_i \left(\frac{D_i}{EOQ_i}\right) KF + \sum_i \left[\frac{D_i}{EOQ_i} KJ_i + \left(\frac{EOQ_i}{2}\ O_i^* - D_i\ L\right) h_i\right] \qquad \text{(IV-12)}$$

Thus the maximum possible saving which can be achieved by a joint policy is

$$MPS = \frac{Z_I - Z_L}{Z_I}$$

where Z_I is the total cost of an independent continuous review system and Z_L is the lower bound on the total cost of a dependent continuous review system.

At any instant, if Z is the total cost for a joint policy calculated by (III-11), the saving achieved by that over the independent policy is

$$\text{Ach S} = \frac{Z_I - Z}{Z_I} .$$

Then, if Z* is the best Z found up to now by the algorithm, the actual saving achieved by the algorithm will be

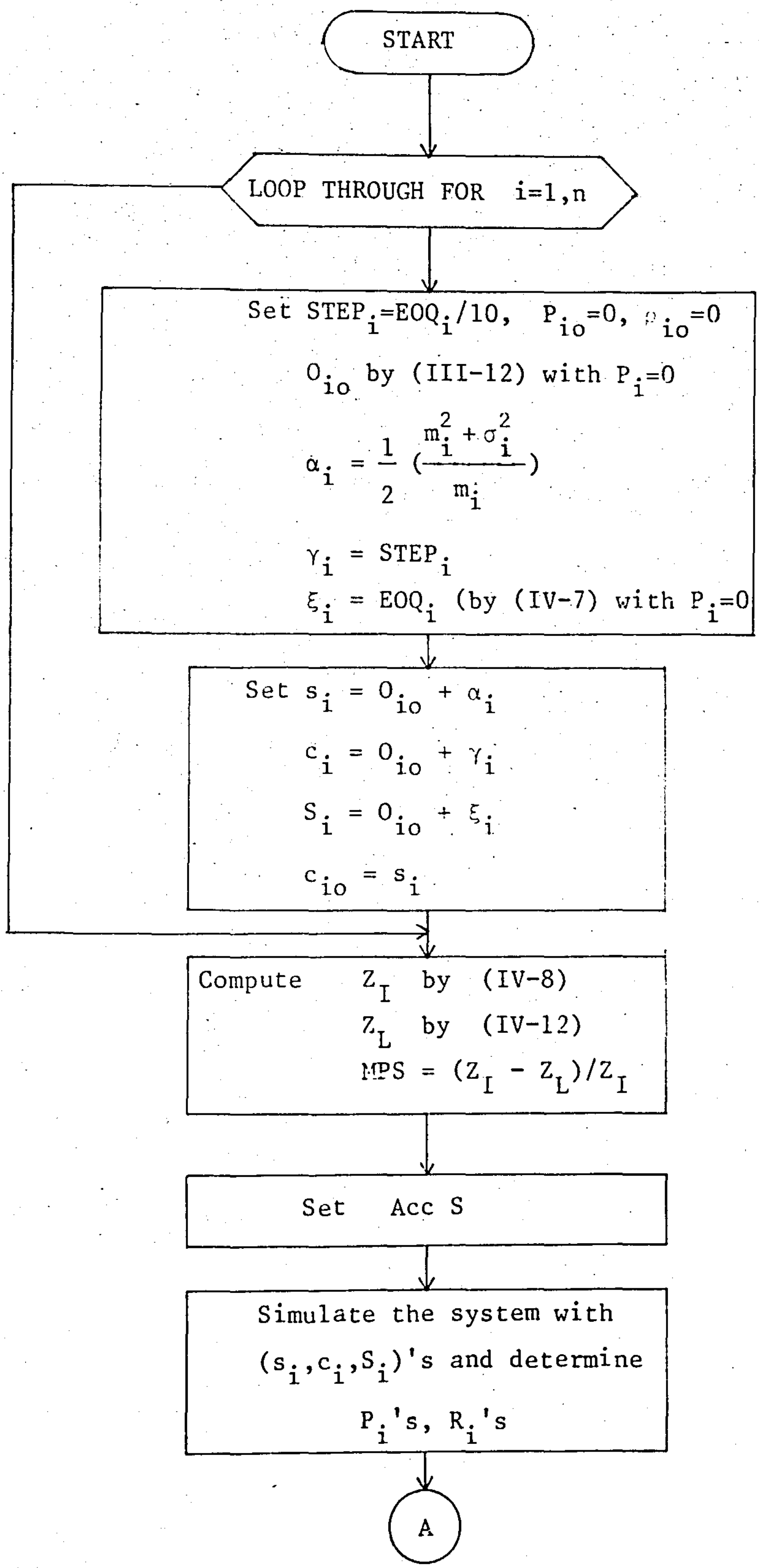
$$\text{AS} = \frac{Z_I - Z^*}{Z_I} .$$

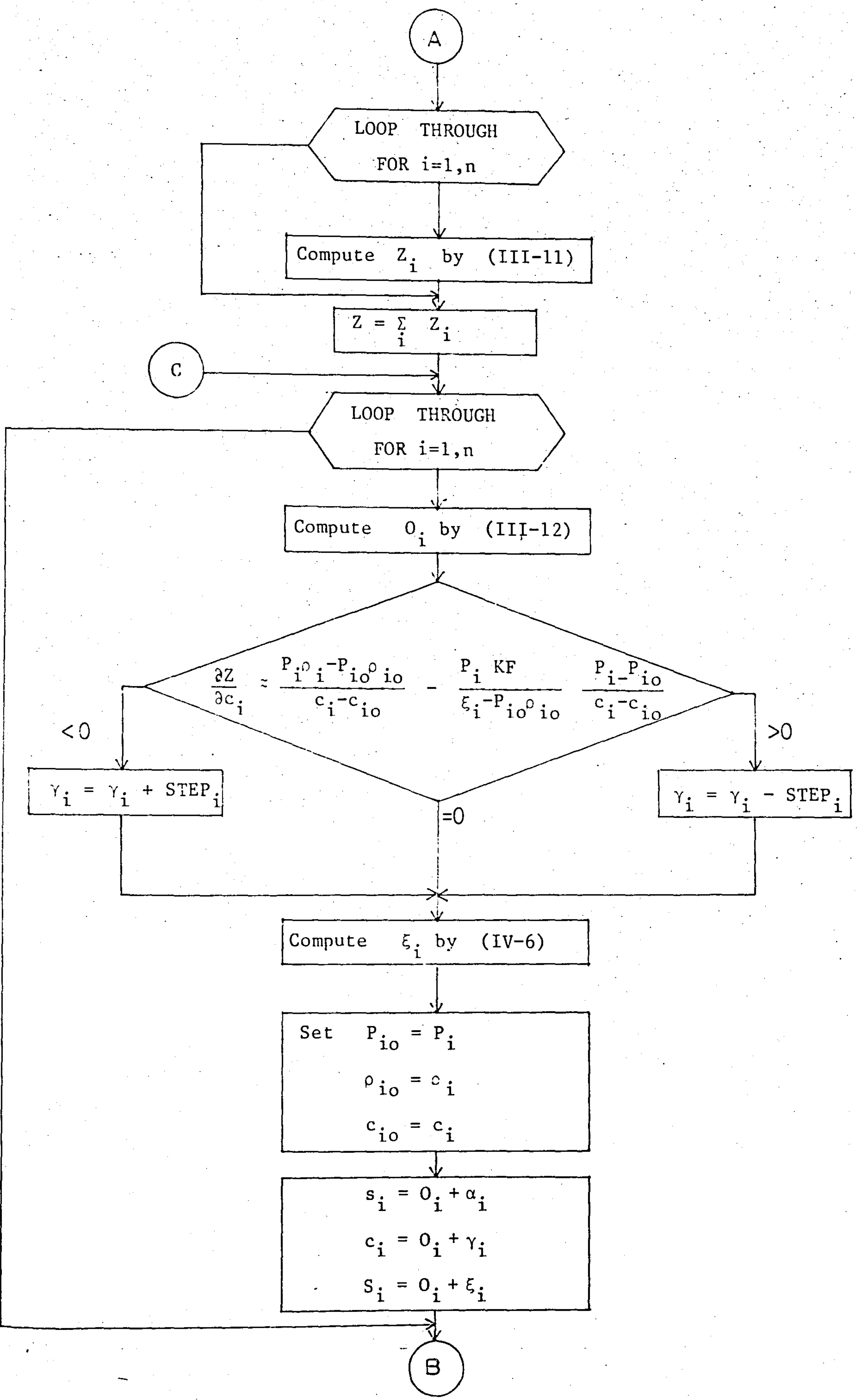
Comparing the actual saving achieved by the algorithm up to now (Ach S) and the maximum possible saving of the algorithm (MPS), one may put a lower bound on the saving of the algorithm (Acc S). Then the stopping condition will be

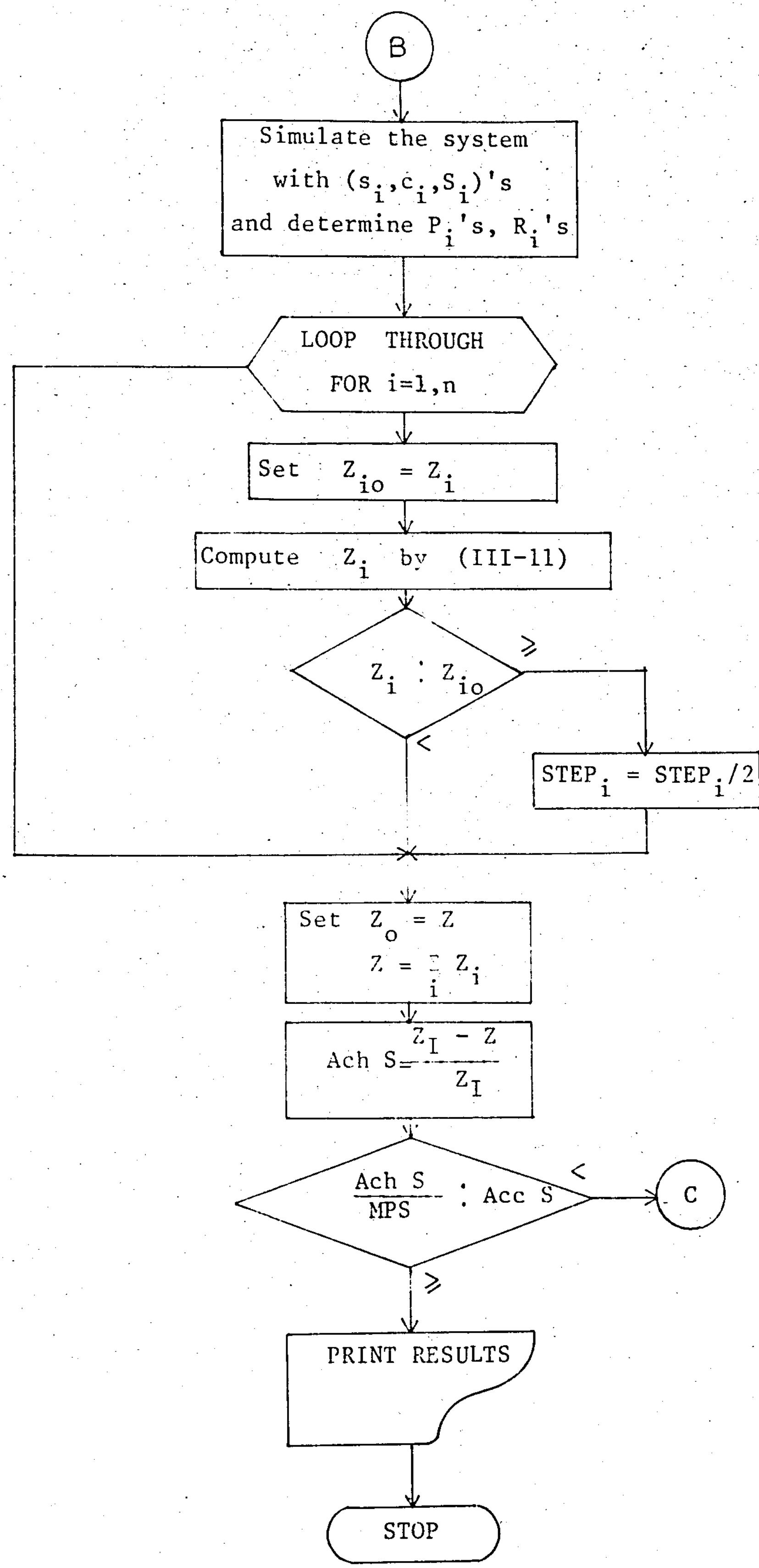
$$\frac{\text{Ach S}}{\text{MPS}} > \text{Acc S} .$$

When this target is reached and improvements are two small, the algorithm will not be run anymore. The best value of Z reached until then will be the optimal cost and the inventory control policy defined by the control parameters of that iteration will be accepted as optimum.

IV.5. MACRO FLOW-CHART OF THE OPTIMIZATION ALGORITHM




A
LOOP THROUGH
FOR i=1,n
Compute Z_i by (III-11)
$Z = \sum_i Z_i$
C
LOOP THROUGH
FOR i=1,n
Compute O_i by (III-12)
$\frac{\partial Z}{\partial c_i} \simeq \frac{P_i \rho_i - P_{io}\rho_{io}}{c_i - c_{io}} - \frac{P_i KF}{\xi_i - P_{io}\rho_{io}} \frac{P_i - P_{io}}{c_i - c_{io}}$
<0
>0
=0
$\gamma_i = \gamma_i + STEP_i$
$\gamma_i = \gamma_i - STEP_i$
Compute ξ_i by (IV-6)
Set $P_{io} = P_i$
$\rho_{io} = \rho_i$
$c_{io} = c_i$
$s_i = O_i + \alpha_i$
$c_i = O_i + \gamma_i$
$S_i = O_i + \xi_i$
B


B
Simulate the system with (s_i, c_i, S_i)'s and determine P_i's, R_i's
LOOP THROUGH FOR i=1,n
Set Z_io = Z_i
Compute Z_i by (III-11)
Z_i : Z_io
≥
<
STEP_i = STEP_i/2
Set Z_o = Z
Z = Σ_i Z_i
Ach S = (Z_I - Z) / Z_I
Ach S / MPS : Acc S
<
C
≥
PRINT RESULTS
STOP

CHAPTER V

COMPUTER PROGRAMMING

V.I. INTRODUCTION

Since the algorithm described earlier in the thesis contains a simulation stage, computer usage becomes necessary in determining the optimal values of the decision variables. A highly large computer program with 2 subroutines and 6 functions is written to computerize the model. The connections of the main program with the subroutines and functions may be structurally shown as in Figure V.1.

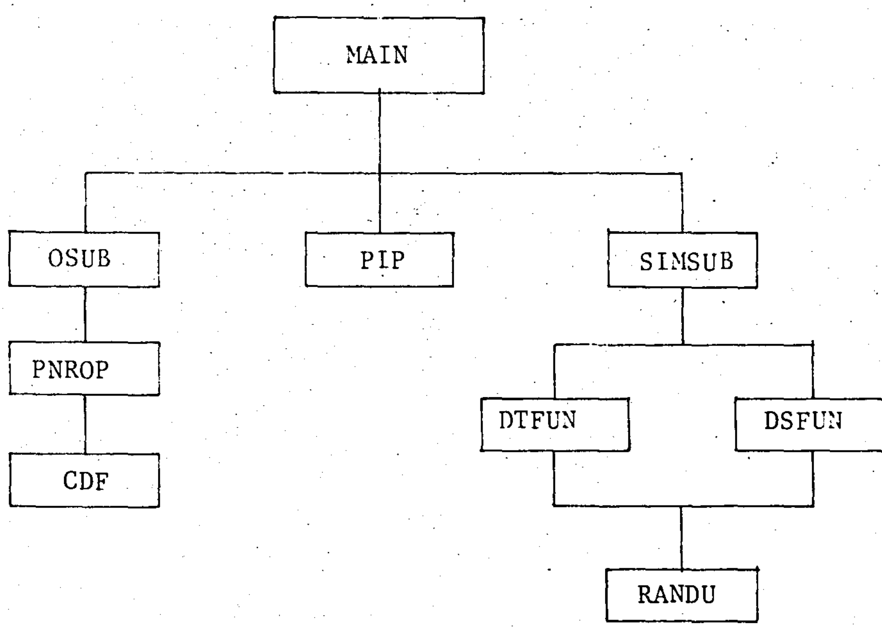


Figure V.1. Program Structure

Main steps of the algorithm are realized in the main program. First, the entrance of the input data and the computation of the necessary parameters occur. Then subroutine OSUB is called to solve the stockout function and the results, O_i's, are used in the updating procedure of the decision variables. With the new values of the inventory control parameters, subroutine SIMSUB is called by the main program. It realizes the simulation stage of the algorithm and makes it possible to observe the behaviour of the inventory system through simulation. At the end of the each cycle in the main program, the inventory control and update parameters are corrected according to the results of the present system.

Finally, if the achieved point is acceptable, the last values of the decision variables are accepted as the best solution and the program ends. But if it is not so good, it is passed through the same cycle in the main program with the new values of the parameters to achieve a better solution.

V.2. SUBROUTINES AND FUNCTIONS

V.2.1. SUBROUTINE OSUB

Subroutine OSUB determines the value of the parameter O_i in the algorithm. It solves the service level function (III-12) in terms of O_i . A new iterative procedure developed in the study and explained in (IV.2.1.1.3.) is used in the subroutine OSUB. After the interval in which the root lies is reached, new point x_3 is generated using linear interpolation. Some checking points are included in the subroutine and an upper limit, NITMAX, for the number of iterations is given. If the desired accuracy is not reached by the maximum number of iterations, last value is accepted as the solution. The logic diagram of subroutine OSUB is shown in Fig. V.2.

V.2.2. FUNCTION PNROP

Function PNROP calculates the yearly probability of not stocking out with current values of the parameters for each item. It computes the left-hand side of (III-12) and returns this value to the subroutine OSUB which will give the solution of (III-12) . Figure V.3. illustrates the logic flow in function PNROP.

V.2.3. FUNCTION CDF

Computing the yearly probability of not stocking out, service level function is the product of two special functions which are both cumulative density functions of normal distribution. Then, function PNROP calls another function CDF to evaluate the value of cumulative density function Φ. CDF makes numerical integration using Simpson's $\frac{1}{3}$ rule as explained in (IV.2.1.2.):

$$\int_a^b f(x)\,dx \approx \frac{h}{3} \sum_{i=0}^{n} c\, f_i \quad \text{where} \quad c=1,4,2,\ldots,2,4,1$$

where f(x) is taken as $(1/2\Pi)*\exp(-x^2/2)$ for normal distribution.

The results obtained in function CDF are compared to the values given in the standard tables and for n=10, it gives quite correct results up to four decimal compared to the tabulated values. The logic diagram of function CDF is given in Figure V.5.

V.2.4. SUBROUTINE SIMSUB

Subroutine SIMSUB process the simulation stage. In this subroutine, inventory system is observed taking the operating decisions with the current values of the inventory control parameters and the new values of some parameters (P_i and σ_i) are returned to the main program. It uses an event base simulation technique: The simulation time, CLOCK, passes from the present event to the closest event. In the system, there are two types of events which change the position of the inventories. First, there will be additions to the inventories if an order is received. Secondly, arrival of a demand decreases the inventory and a review will be done. As a result, another order may be placed containing some items according to their inventory positions. Choosing the smallest of the all demand times, time of the earliest demand, NEXTDT, is determined. In the same way, time of the earliest order receipt, NEXTOT, will be computed. Then the next event to which system will jump will be the one with earlier time.

If the system jumps to a time at which an order is received, amounts received are added to the respective inventory **levels** and these orders are erased from the list of order on-road. But if a demand occurs as the next event,demand is supplied by the respective inventory. After the updatings in the inventory position and the inventory level are done, the new inventory position is controlled whether it drops below the critical level. If placement of an order decided, all the other items are reviewed according to the joint ordering licy and the possible orders are included jointly in that order for the items th the inventory positions below their can-order points.

broutine SIMSUB calls two functions, DTFUN and DSFUN, in order to generate a w demand for an item. Functions DTFUN and DSFUN are used to determine the mand time and demand size respectively.

2.5. FUNCTION DTFUN

nce the distribution of the arrival times of demand for an item is taken as isson, it will have a probability mass function in the form of, [4]

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

ere $\lambda \geq 0$ is the rate of arrivals and $t \geq 0$.

en, the distribution of time between successive demands is exponentially stributed and has probability mass function and cumulative mass function as ven in Appendix 1:

$$f(t) = \lambda e^{-\lambda t}, \qquad 0 < t < \infty$$

$$F(t) = \int_0^t e^{-\lambda x}\, dx = 1-e^{-\lambda t} \quad .$$

lving F(t) for t and setting r = F(t)

$$e^{-\lambda t} = 1-F(t)$$

$$-\lambda t = Ln(1-r)$$

$$t = -\frac{1}{\lambda}\ Ln(1-r) \ . \qquad \text{(V-1)}$$

generate exponentially distributed random variables, a random number r, tween 0 and 1 is selected and t is evaluated by (V-1).

nction DTFUN is called to determine when the next demand will occur after the st one and it generates an exponentially distributed interdemand time for an em. The logic diagram of function DTFUN is given in Figure V.6.

2.6. FUNCTION DSFUN

ter the time of the next demand is generated by function DTFUN, function FUN is called to generate the size of that demand. Since the amount demanded each time is assumed to be non-negative normally distributed, the demand ze will follow a normal distribution truncated to the left of 0. For certain rameter values, if the probability of negative demand size is negligible $(x<0) \sim 0)$ a normal process generator can be used for that purpose.

x is normally distributed with mean μ and variance σ^2, the cumulative stribution function of x is given by

$$F(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\Pi}\,\sigma} e^{-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2} dt .$$

was shown in [23] that generator for that distribution will be

$$x = \mu + \frac{r-0.5}{|r-0.5|}\,\sigma\,\left(\nu - \frac{2.515517 + 0.802853\nu + 0.010328\nu^2}{1 + 1.432788\nu + 0.189269\nu^2 + 0.001308\nu^3}\right)$$

ere $\nu = \sqrt{-2 \text{ Ln } 0.5(1-|1-2r|)}$ and r is a random number between 0 and 1.

in the study, an easier and quicker approximation is used to generate a rmal random variable. This process generator depends on central limit theorem]. If y is a random variable with finite mean μ and variance σ^2, from the tral limit theorem:

$$\sum_{i=1}^{N} y_i$$

an approximate normal distribution with mean $N\mu$ and variance $N\sigma^2$ for ge N. Since r_i, a random variable between 0 and 1, is uniformly distributed on the interval (0,1),

$$E\ (r_i) = \frac{1}{2}$$

$$Var\ (r_i) = \frac{1}{12} \quad .$$

Therefore for large N,

$$\sum_{i=1}^{N} r_i$$

has an approximate normal distribution with mean N/2 and variance N/12. This immediately leads to a process generator for the standard normal random variable, Z, and is given by:

$$Z = \frac{\sum_{i=1}^{N} r_i - (N/2)}{\sqrt{N/12}}$$

To generate a normal random variable x with mean μ and variance σ^2, we need only the relationship between the general normal and standard normal random variables as

$$Z = \frac{x - \mu}{\sigma} \quad .$$

Using the last two equations, the process generator for a normal variable with mean μ and variance σ^2 will be:

$$x = \mu + \sigma \quad \frac{\sum_{i=1}^{N} r_i - (N/2)}{\sqrt{N/12}} \quad . \qquad \text{(V-2)}$$

In order to simplify the formula, if N is selected 12 as in the program, the generator becomes as

$$x = \mu + \sigma \left(\sum_{i=1}^{12} r_i - 6\right) .$$

DSFUN, normal process generator function, is summarized in Figure V.7.

V.2.7. FUNCTION RANDU

Function RANDU is called by functions DTFUN and DSFUN whenever the demand of an item is to be determined. It generates uniformly distributed random numbers for generating two random events, the demand time and the demand size. The multiplicative congruential method is used in RANDU to generate each random number. This uniform random number generator is written for a binary computer with 36 bits per word. All we need to do is supply the first feed number IR and make sure it is five digit odd. The routine takes over and generates the new integer value of parameter JR to use for next entry into function. This gives the opportunity of generating any number of independent random numbers at each time.

V.2.8. FUNCTION PIP

Function PIP is used to determine the minimum possible number of orders for independent inventory system. Its function is to determine the smallest integer number equal to or greater than a floating point number. Then this result is used to compute the total cost for the system where items are controlled independently.

V.3. DEFINITION OF THE VARIABLES IN THE COMPUTER PROGRAM

The indicators used in the program are listed as:

NI : Card reader unit number

NO : Printer unit number

MPS : Maximum possible saving by the model

ACCS : Acceptable percent of the saving by the model over the independent one

ACHS : Achieved percent of the saving at each iteration

NITMAX : Maximum number of iterations for solving service level function

NIT : Number of iterations passed for solving service level function

DX : Step size in the iterative method for solving service level function

ACCN, ACCO : Acceptable accuracies in vertical and horizontal axis respectively in service level function

NINT : An even number indicating the number of intervals in the Simpson's integration method

DV : Interval lenght in the Simpson's integration method

DURSIM : Duration of the simulation run

CLOCK : Time indicator in the simulation run

FLAGO : Order indicator (1 if an order has to be placed, 0 otherwise)

FLAGJO : Joint order indicator (1 if the order is placed jointly)

FLAGOR : Order receipt indicator (1 if an order is received at the present time, 0 otherwise)

NORRAN : Number of samples in normal distribution generator

IR : Initial number for uniformly distributed random number generator

JR : Integer number used at each generation to generate uniformly distributed random number.

Function defined in the program are:

TRCFUN(,...,): computes the value of the objective function

EFUN(,...,) : gives the optimal value of the parameter E(I) by IV.7.

DERFUN(,...,): takes the derivative of the objective function with respect to can-order point

F(,...,) : defines the probability density function for normal distribution

Main inputs to the program are defined as:

N : Number of items in the group

K1 : Fixed set-up cost per order

K2(I) : Variable part of the ordering cost for item I per order

LTIME : Replenishment lead time

D(I) : Expected yearly demand for item I

MTRAN(I), SDTRAN(I): Expected value and standard deviation of the transaction size for item I, respectively

H(I) : Unit inventory carrying cost per year for item I

APRO(I) : Maximum allowed probability of running out for item I (the service criterion)

K3(I) : Total ordering cost for item I for an independent replenishment

DRATE(I) : Parameter of the demand time distribution for item I

MDLT(I), SDDLT(I): Expected value and standard deviation of the lead-time demand for item I with normally distributed demand

EOQ(I) : Economic order quantity for item I

The variables used in the program as:

OUL(I) : Order-up-to level for item I (Decision variable S_i)

COL(I),COLO(I): Can-order points for item I at the last two iterations (Decision variables c_i, c_{io})

OP(I)	:	Must-order point for item I (Decision variable s_i)
SAFES(I)	:	Safety stock of item I in independent system
TRCI	:	Total relevant cost for independent system
TRC, TRCO	:	Values of the objective function at the last two iterations
MINTRC	:	Lower bound on the objective function
P(I), PO(I)	:	Probabilities of joint ordering for item I in the last two iterations, respectively.
E(I)	:	Parameter used to update order-up-to levels (ξ_i in the model)
Y(I)	:	Parameter used to update can-order points (γ_i in the model)
DELTA(I)	:	Change in the Y(I) at each iteration
U	:	Derivative of the objective function with respect to can-order point
A(I)	:	Parameter used to update must-order points (α_i in the model)
R(I), RSIM(I)	:	Computed and simulated values of the order level when a joint ordering occurs
O(I), OSIM(I)	:	Computed and simulated values of the order level when order is triggered by item I
G(I), GO(I)	:	Differences between RSIM(I) and OSIM(I) at the previous and present iterations respectively
Z1, Z2	:	Two values between -4 and +4
F1, F2	:	Values of the cumulative density function of normal distribution for Z1, Z2, respectively
INV(I)	:	Inventory level for item I
INVPOS(I)	:	Inventory position for item I
DTIME(I)	:	Next demand time for item I
DSIZE(I)	:	Next demand size for item I
NEXTDT	:	Time of the next demand for the system
ORTIME(I)	:	Next order receipt time for item I
OSIZE(I)	:	Next order size for item I

NEXTOT : Time of the next order receipt for the system

J1 : Index of the item which has a demand arrival

J2 : Index of the item which has an order receipt

NORI : Number of items included in an order

OMAT(K,J) : Order Matrix

For J=1; Return time of the K^{th} order

For J=2; Number of items ordered in the K^{th} order

For J=3,2 * OMAT (K,2)+2; Indices of the items included in the K^{th} order and amounts ordered are stored in pairs

RCI(I) : Relevant cost for item I in independent system

RC(I), RCO(I): Objective function values for item I at the last two iterations

FLAGRC(I) : Indicator for optimal region of COL(I) (1 if RC(I) increased after having first decreased, 0 otherwise)

.4. FLOW CHARTS

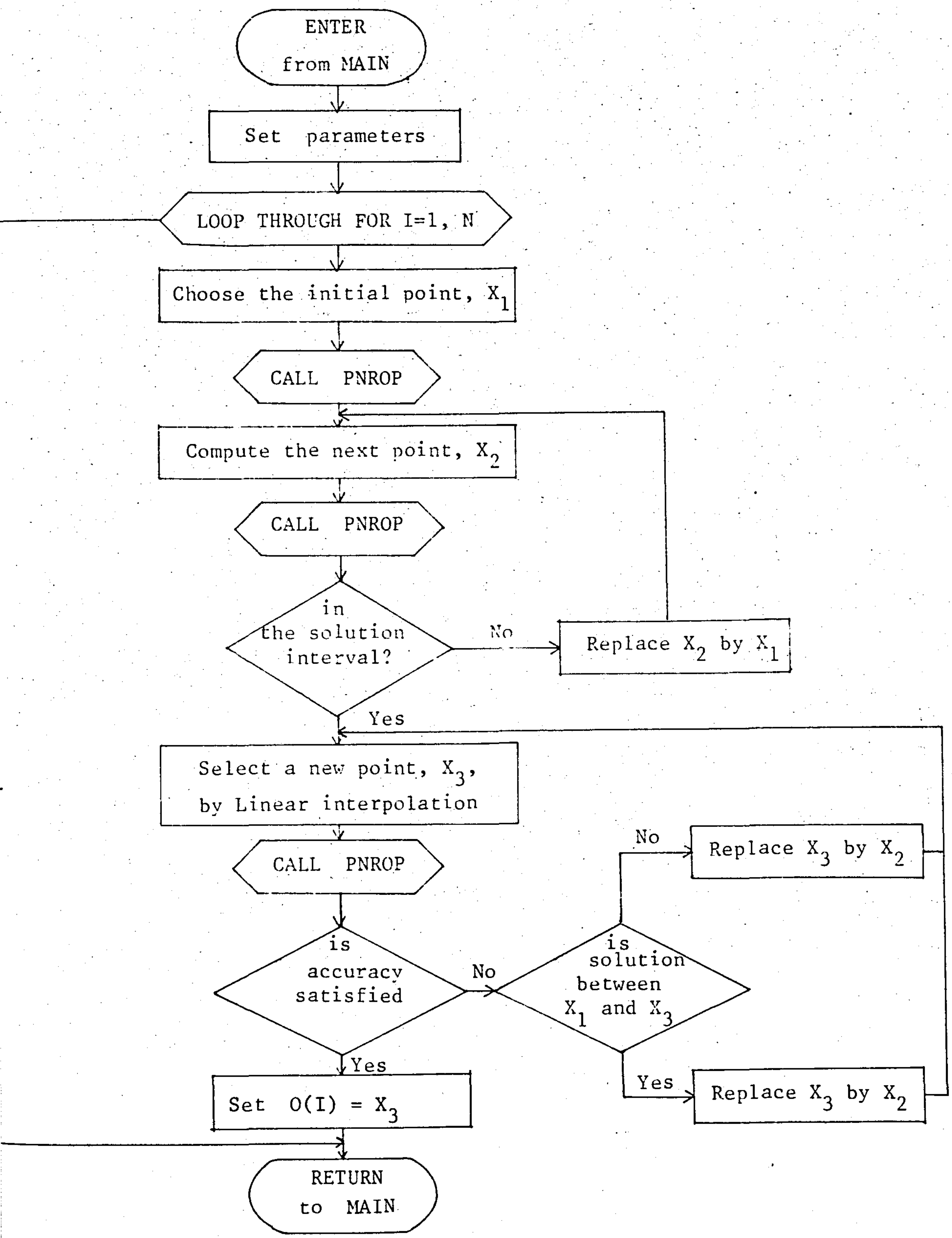


Figure V.2. Macro Flow-Chart of Subroutine OSUB

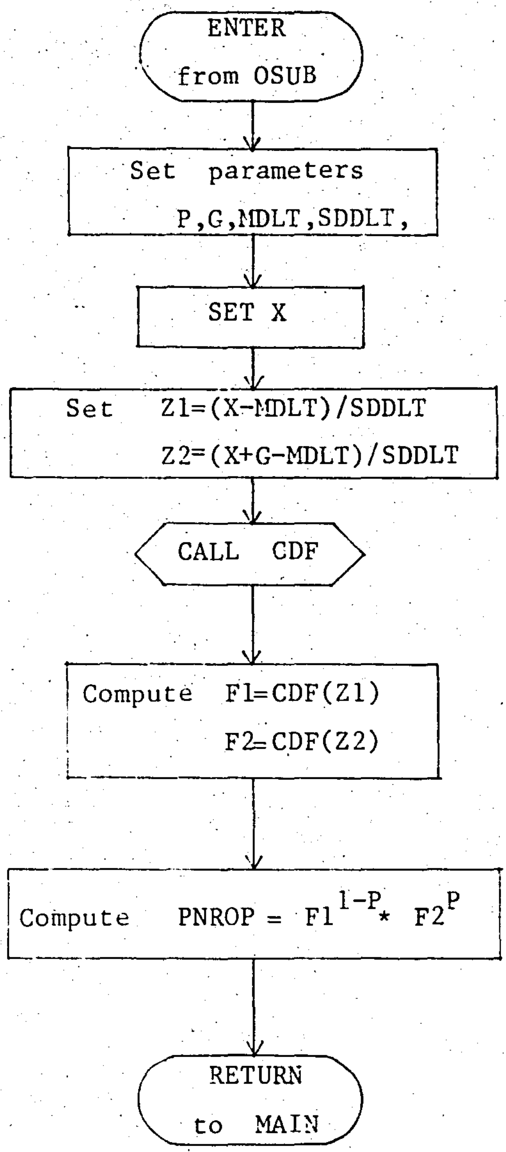


Figure V.3. Macro Flow-Chart of Function PNROP

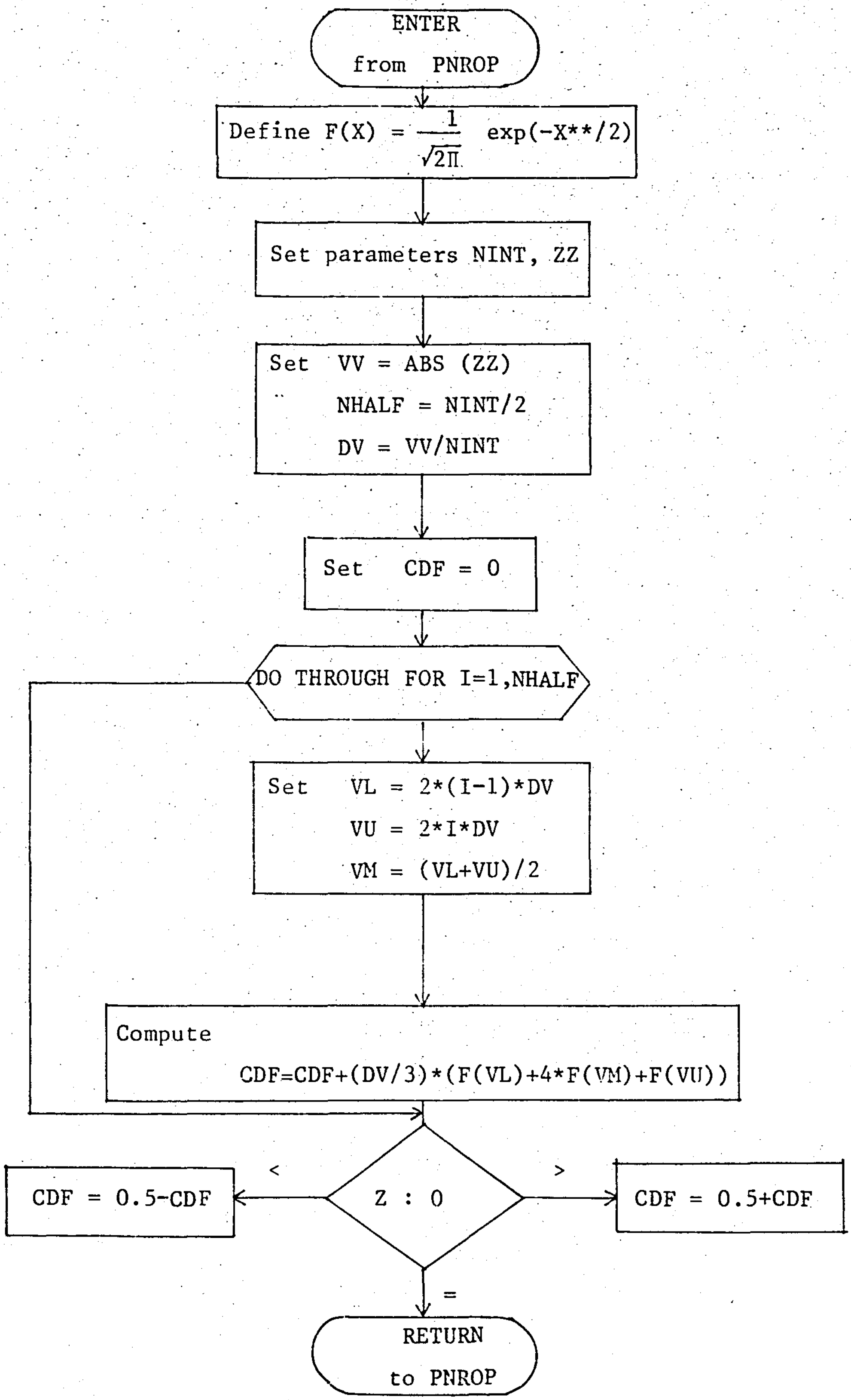
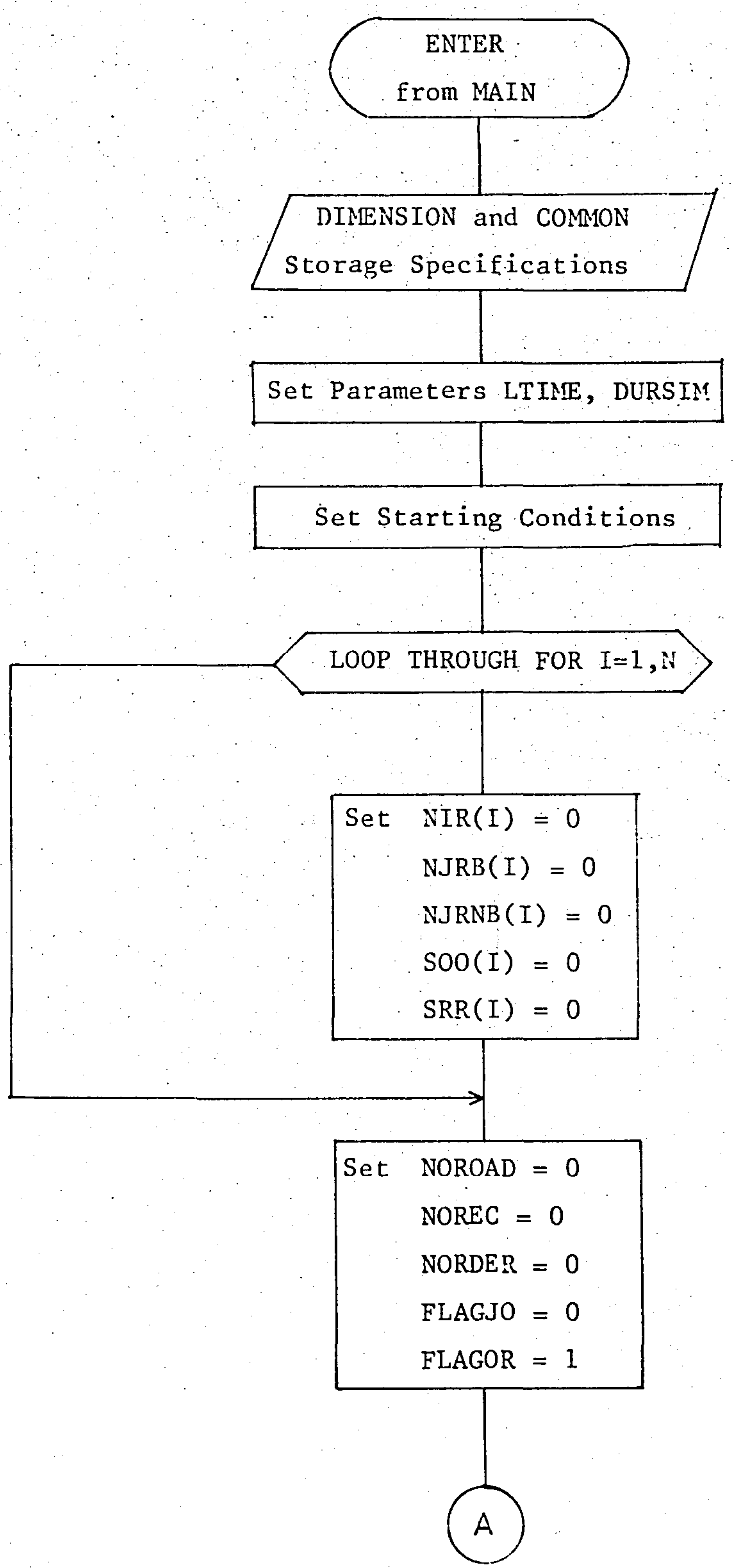
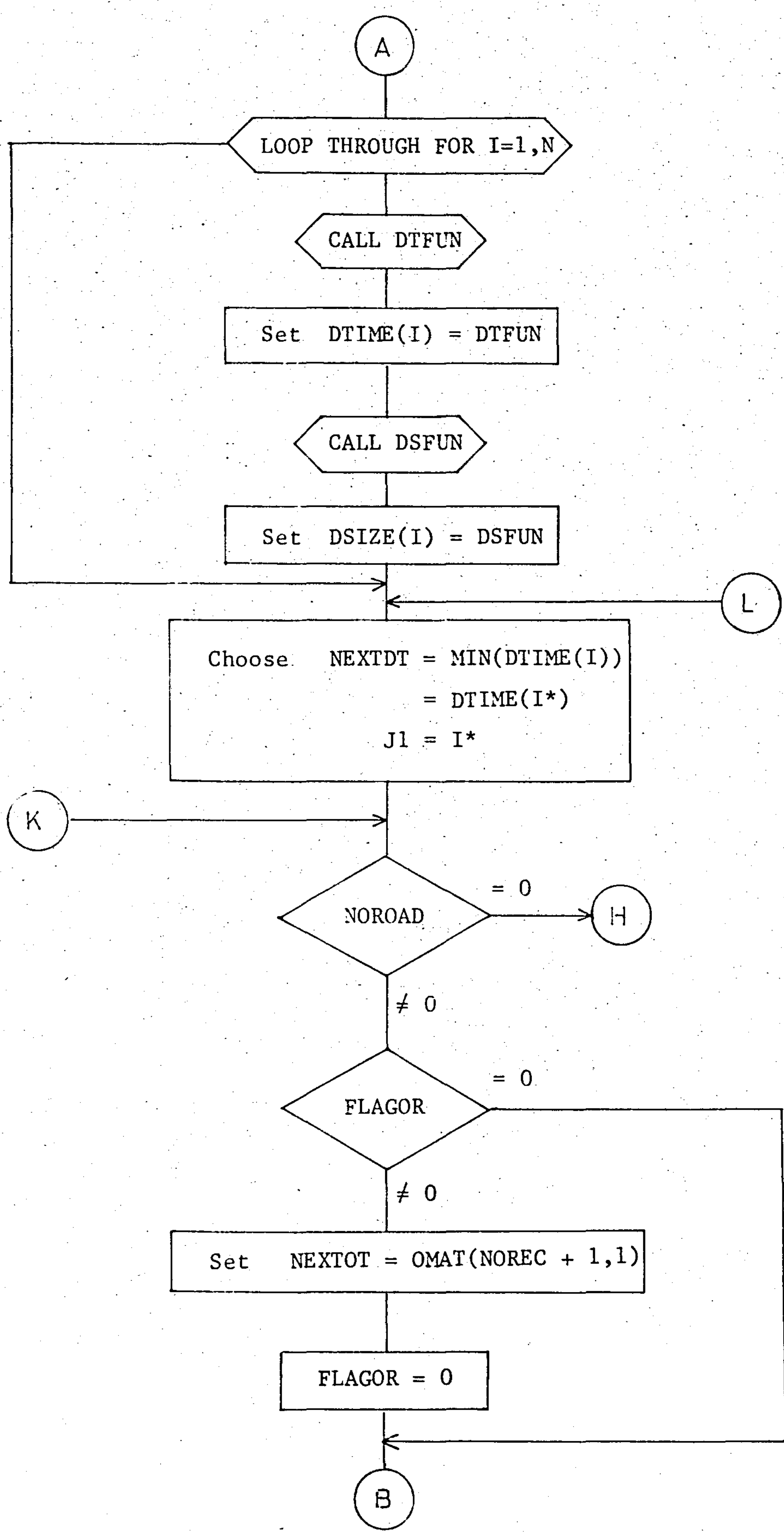
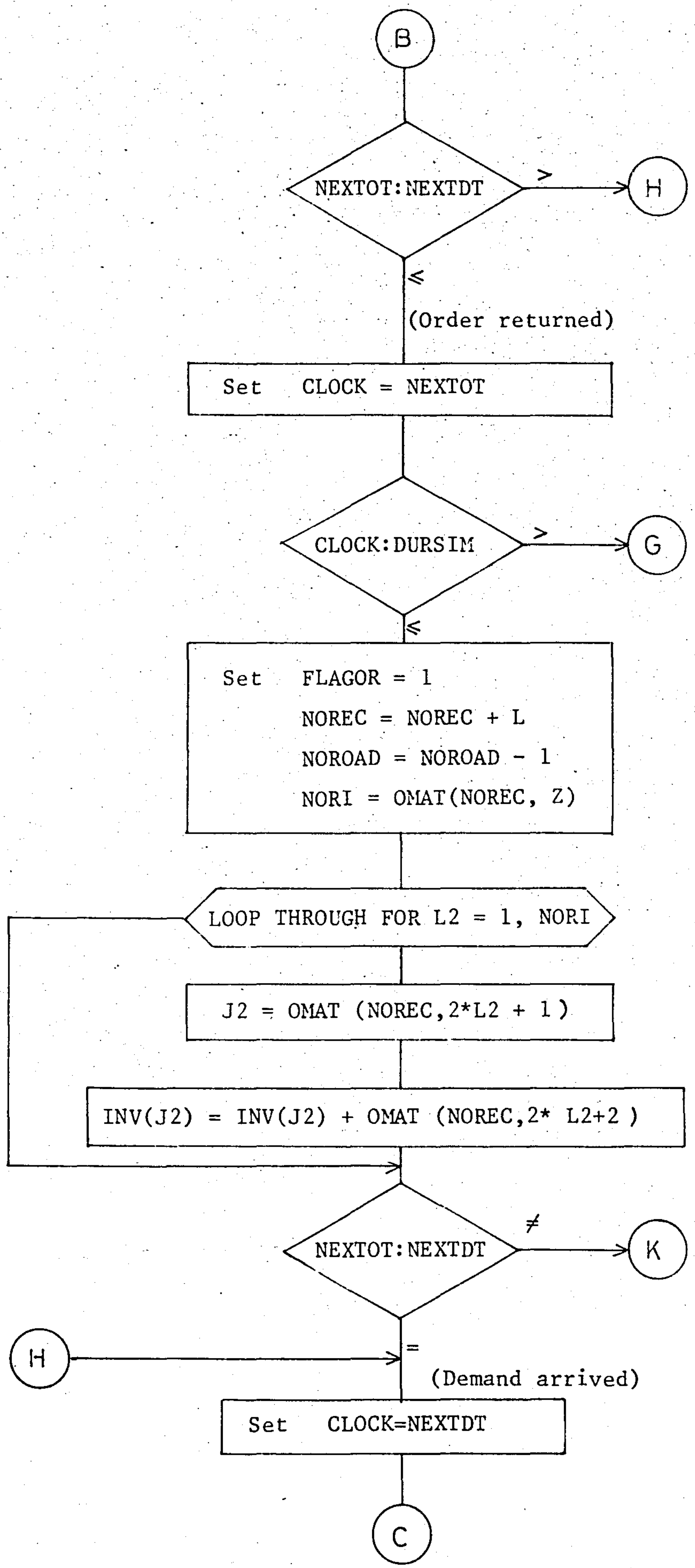
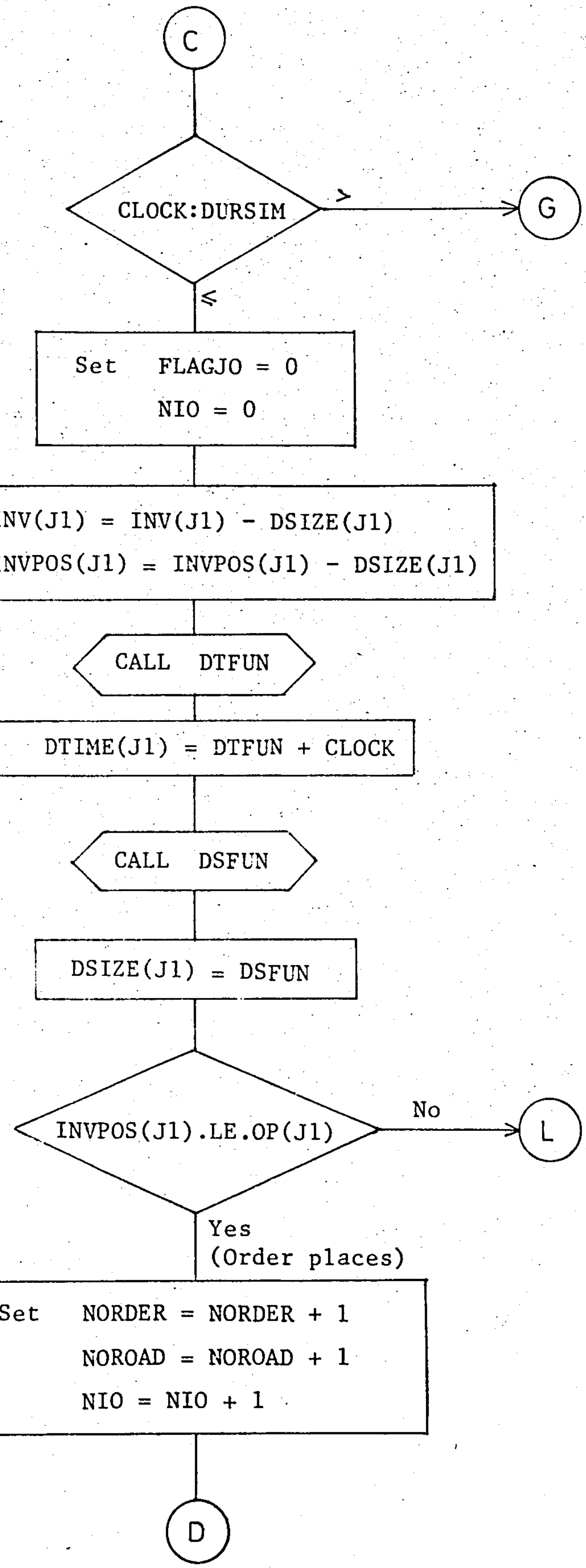


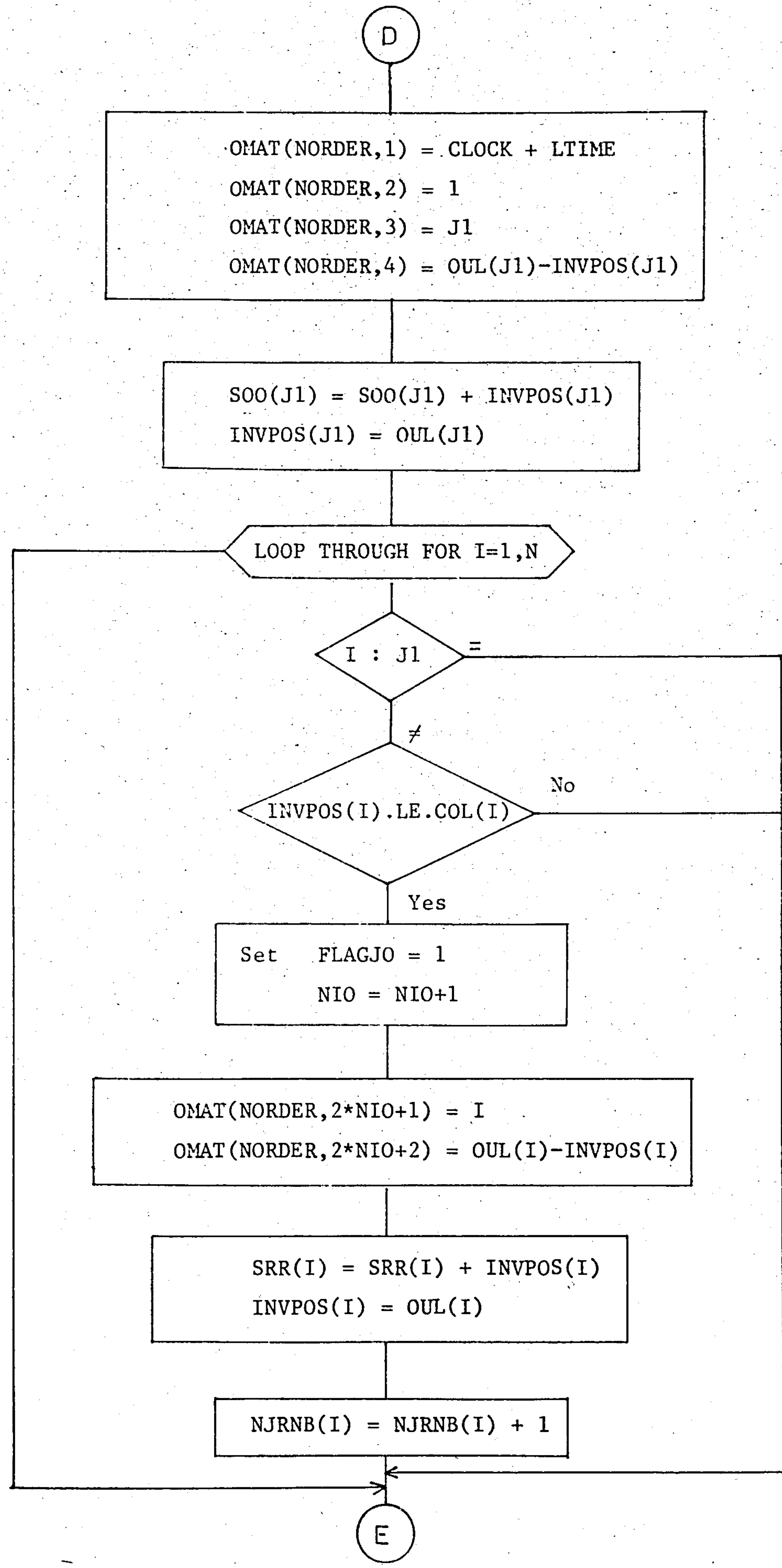
Figure V.4. Macro Flow-Chart of Function CDF

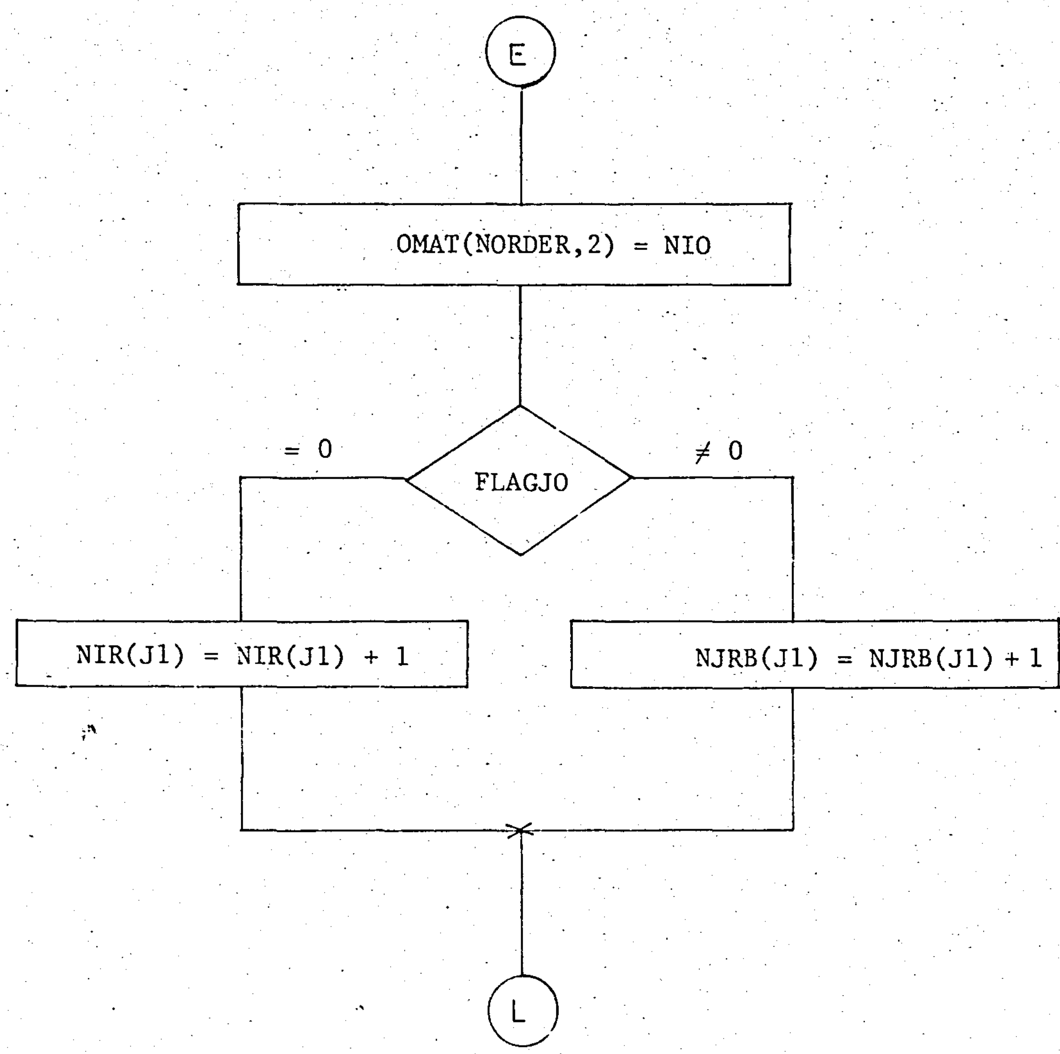

ENTER
from MAIN
DIMENSION and COMMON
Storage Specifications
Set Parameters LTIME, DURSIM
Set Starting Conditions
LOOP THROUGH FOR I=1,N
Set NIR(I) = 0
NJRB(I) = 0
NJRNB(I) = 0
SOO(I) = 0
SRR(I) = 0
Set NOROAD = 0
NOREC = 0
NORDER = 0
FLAGJO = 0
FLAGOR = 1
A


A
LOOP THROUGH FOR I=1,N
CALL DTFUN
Set DTIME(I) = DTFUN
CALL DSFUN
Set DSIZE(I) = DSFUN
L
Choose NEXTDT = MIN(DTIME(I))
= DTIME(I*)
J1 = I*
K
NOROAD
= 0
H
≠ 0
FLAGOR
= 0
≠ 0
Set NEXTOT = OMAT(NOREC + 1,1)
FLAGOR = 0
B


B
NEXTOT:NEXTDT
>
H
≤
(Order returned)
Set CLOCK = NEXTOT
CLOCK:DURSIM
>
G
≤
Set FLAGOR = 1
NOREC = NOREC + L
NOROAD = NOROAD - 1
NORI = OMAT(NOREC, Z)
LOOP THROUGH FOR L2 = 1, NORI
J2 = OMAT (NOREC,2*L2 + 1)
INV(J2) = INV(J2) + OMAT (NOREC,2* L2+2)
NEXTOT:NEXTDT
≠
K
H
=
(Demand arrived)
Set CLOCK=NEXTDT
C


C
CLOCK:DURSIM
>
G
≤
Set FLAGJO = 0
NIO = 0
INV(J1) = INV(J1) - DSIZE(J1)
INVPOS(J1) = INVPOS(J1) - DSIZE(J1)
CALL DTFUN
DTIME(J1) = DTFUN + CLOCK
CALL DSFUN
DSIZE(J1) = DSFUN
INVPOS(J1).LE.OP(J1)
No
L
Yes
(Order places)
Set NORDER = NORDER + 1
NOROAD = NOROAD + 1
NIO = NIO + 1
D


D
OMAT(NORDER,1) = CLOCK + LTIME
OMAT(NORDER,2) = 1
OMAT(NORDER,3) = J1
OMAT(NORDER,4) = OUL(J1)-INVPOS(J1)
SOO(J1) = SOO(J1) + INVPOS(J1)
INVPOS(J1) = OUL(J1)
LOOP THROUGH FOR I=1,N
I : J1
=
≠
INVPOS(I).LE.COL(I)
No
Yes
Set FLAGJO = 1
NIO = NIO+1
OMAT(NORDER,2*NIO+1) = I
OMAT(NORDER,2*NIO+2) = OUL(I)-INVPOS(I)
SRR(I) = SRR(I) + INVPOS(I)
INVPOS(I) = OUL(I)
NJRNB(I) = NJRNB(I) + 1
E


E
OMAT(NORDER,2) = NIO
= 0
FLAGJO
≠ 0
NIR(J1) = NIR(J1) + 1
NJRB(J1) = NJRB(J1) + 1
L

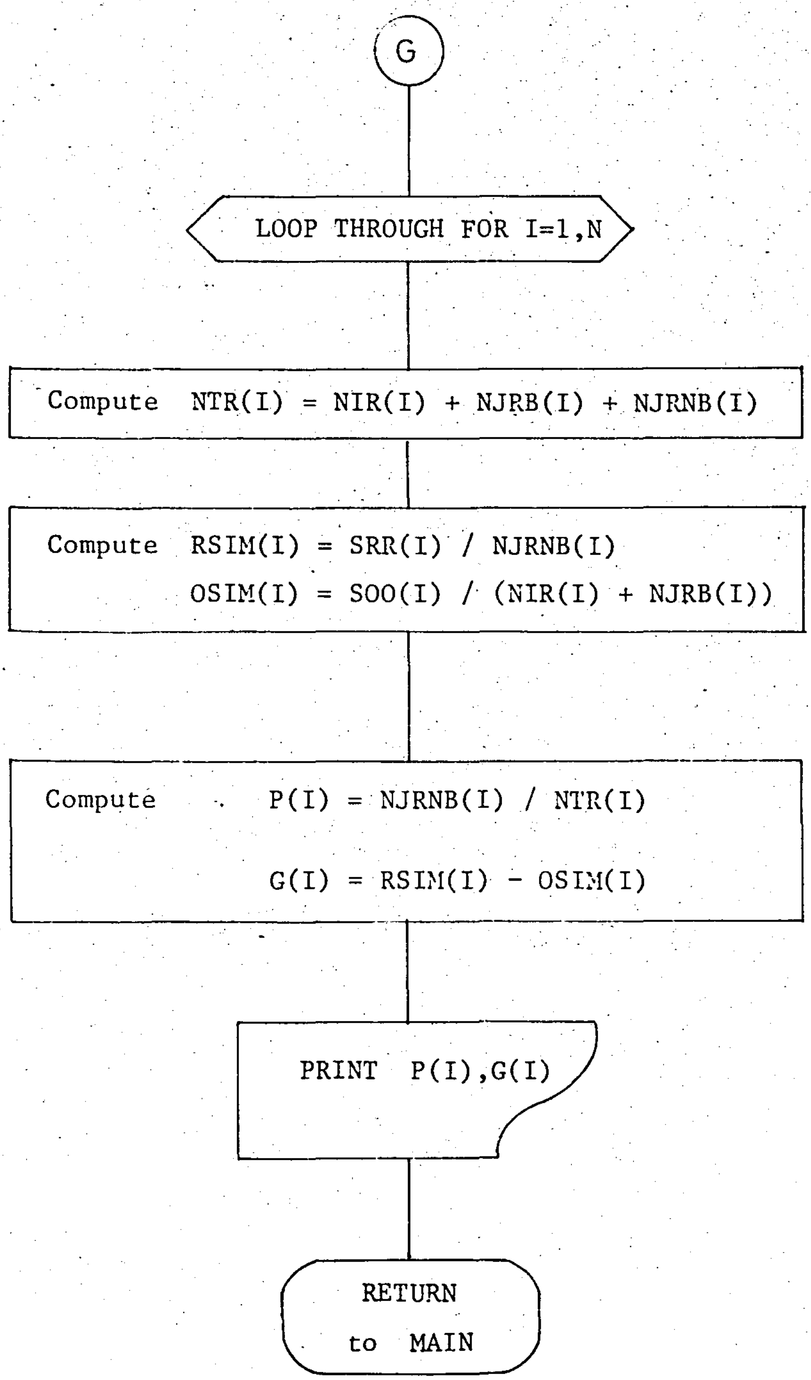


Figure V.5. Flow-Chart of Subroutine SIMSUB

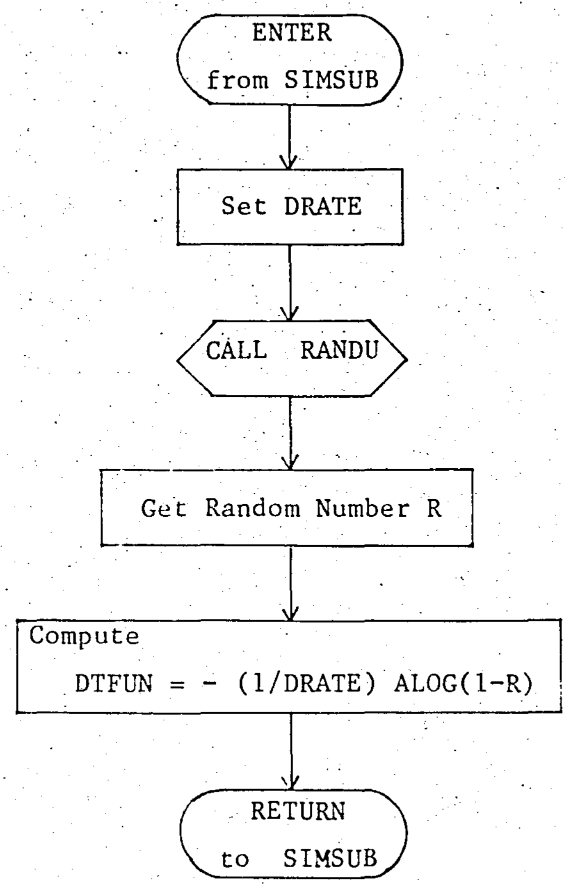


Figure V.6. Macro Flow-Chart of Function DTFUN

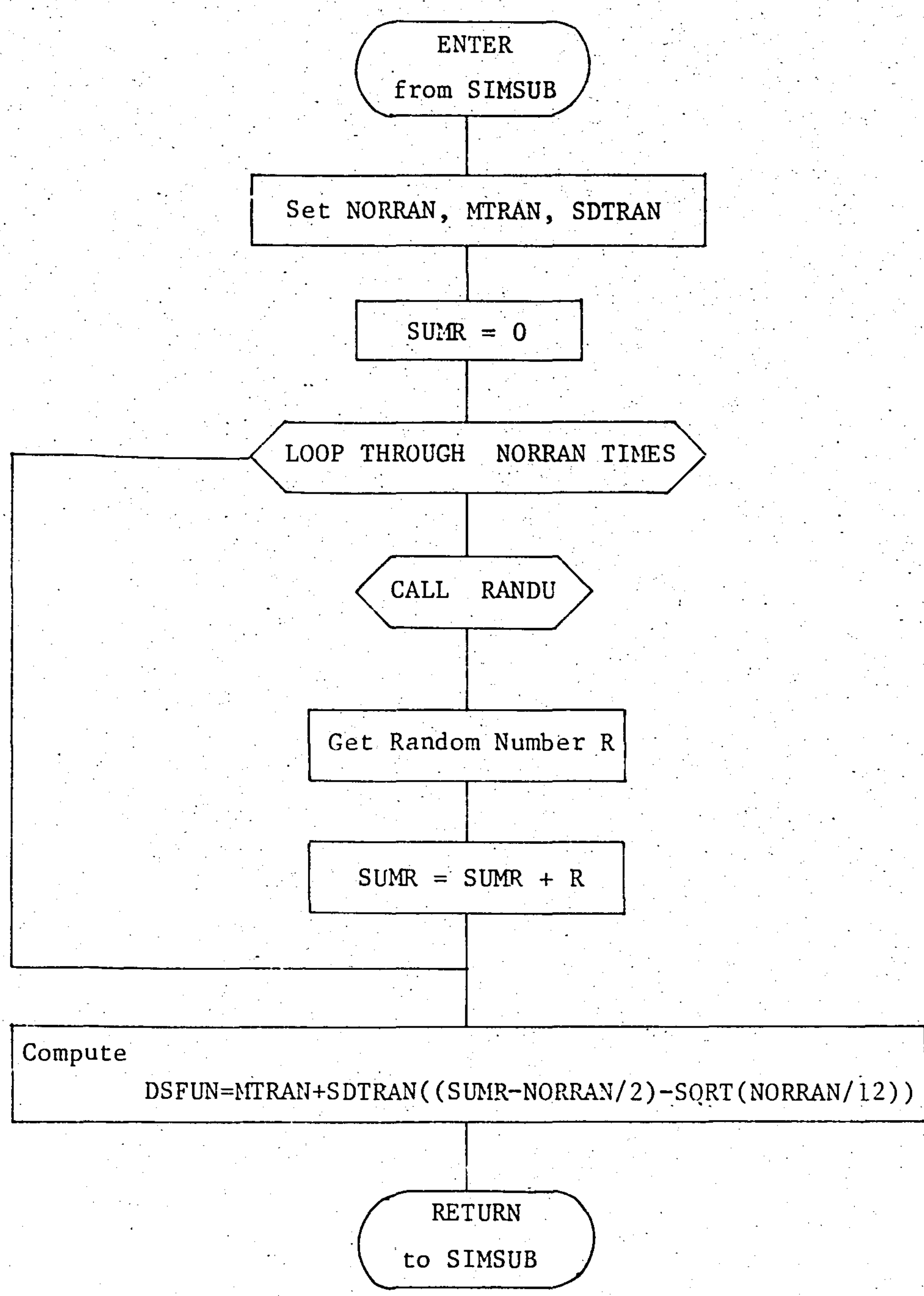


Figure V.7. Macro Flow Chart of Function DSFUN

CHAPTER VI
IMPLEMENTATION OF THE STUDY

VI.I. IMPLEMENTATION OF THE MODEL

A mathematical model for controlling the inventories in the district warehouses of Sümerbank was developed and the solution algorithm to determine the optimal inventory control parameters has been computerized. As explained in Chapter II, the model will be run seperately for items of each factory-district warehouse pair. The complete inventory control system for a specific district warehouse will then be set up combining the stock control systems of all groups of items delivered by that district.

Application of the model for a selected group of items which are produced in a specific factory and delivered by a specific district warehouse will be presented in the following section.

VI.1.1. SAMPLE PROBLEM

Eskişehir Factory and İstanbul District Warehouse are selected as sample factory and district warehouse respectively. The model will be run for items produced in Eskişehir Factory and delivered by İstanbul District Warehouse.

First evaluation of the parameters for 1980 will be given, then the results will be presented. Selected items produced in Eskişehir Factory and sold through İstanbul District Warehouse and their price in 1980 are given in Table VI.1.

TABLE VI.1. ITEMS USED IN THE EXAMPLE

Factory and Type No:	Description	Price (TL/mt)
ESK.205	Printed Cloth (Basma)	54
ESK.214	Film Printed Satin (Film Saten Emprime)	100
ESK.283	Kreton Emprime	115
ESK.290	Jet.Poplin Emprime	100
ESK.293	Jet.Raised Emprime (Jet.Divitin Emprime)	120
ESK.722	Satin Curtaining (Saten Döşemelik)	125

i) Evaluation of Ordering Cost:

A) Fixed Set-up Cost, KF: It is taken as the amount paid to transportation companies for a truck moving between Eskişehir and İstanbul. KF is recently 20,000 TL per order.

B) Item-Ordering Cost, KJ_i: Method used for computing this cost term and results for items in Table VI.1 are given in Appendix IV.

ii) Evaluation of Inventory Carrying Cost, h_i:
Inventory carrying cost was examined in Appendix III and it has been found out that unit cost is 24.3 % of the inventory value per year. Unit inventory costs per year for all items which are computed multiplying their unit prices with 24.3 % are given in Table VI.3.

iii) Delivery Lead Time, L:
Lead time for all items is taken equally and it is the whole time passed from placing an order to receiving it into the inventory. This time is approximately 15 days between Eskişehir and İstanbul.

iv) Evaluation of Yearly Demand, D_i:
Sales forecasts for all items within the whole system were done in [26] and the figures are presented in Table VI.3 of the same reference. "Exponential Smoothing Method" which considers trade and seasonal effects on demand is used to determine the sales figures. Mathematical background, computer programs and other information about the forecasting method can

be met in [27].

Since 24.52 % of the production of Eskişehir Factory is sold through İstanbul District Warehouse on the average from the statistical analysis in [26], demands of items by this district will be approximately 24.52% of the total sales forecasts. Resulting figures are given in Table VI.2.

TABLE VI.2. DEMAND FIGURES

Type No:	Total Sales Forecast of Eskişehir in 1980 (mt)	D_i Sales Forecast of İstanbul District in 1980 (mt)
205	4,943,738	1,212,205
214	600,000	147,120
283	2,039,678	500,130
290	3,380,348	828,860
293	3,766,918	923,648
722	1,419,625	348,092

v) Evaluation of Demand Size Parameters, m_i and σ_i:
There does not exist exact data in the analyzed system to compute these parameters. As an approximation, means of demand sizes for all items are expressed as 1 % of their yearly demand figures and standard deviations are computed as a proportion by means of [27]. These figures are listed in Table VI.3. taking the percents as 15 %, 25 %, 15 %, 20 %, 10 %, 10%.

vi) Allowed Probability of Running Out, Π_i:
It is a service criterion and depends on management policy. For the sample run, maximum allowed probabilities of stocking out are taken as in Table VI.3.

The complete data set for the sample run is tabulated in Table VI.3.

TABLE VI.3. SET OF DATA USED IN THE EXAMPLE

N = 6

L = 0.04 yr

KF= 20,000 TL/order

Item	D_i (mt/yr)	m_i (mt)	σ_i (mt)	KJ_i (TL/order)	KI_i (=KF+KJ_i) (TL/order)	h_i (TL/mt/yr)	Π_i
1	1,212,205	12,000	1,800	1,258	21,258	13.12	0.10
2	147,120	1,500	375	3,957	23,957	24.30	0.05
3	500,130	5,000	750	3,957	23,957	27.90	0.15
4	828,860	8,000	1,600	4,924	24,924	24.30	0.05
5	923,648	9,000	900	3,957	23,957	29.16	0.10
6	348,092	3,500	350	3,957	23,957	30.38	0.20

VI.1.2. EVALUATION OF THE RESULTS

Computer program was run for the data set in Table VI.3. It takes 2.59 minutes of CPU time on a Univac 1106 machine for six-item inventory system for 10 iterations.

Additional parameters which are evaluated from the input data and used through the program are listed in Table VI.4.

TABLE VI.4. COMPUTED PARAMETERS

Item	μ_i $(=D_i\ L)$	ν_i $(=\frac{D_i\ L}{m_i}(m_i^2+\sigma_i^2))$	α_i $(=\frac{m_i^2+\sigma_i^2}{2\ m_i})$	EOQ_i $(=\sqrt{\frac{2\ D_i\ KI_i}{h_i}})$	$STEP_i$ $(=EOQ_i/10)$
1	48,488	24,392	6,135	62,675	6,268
2	5,885	3,062	797	17,032	1,703
3	20,005	10,113	2,556	29,281	2,928
4	33,154	16,609	4,160	41,235	4,124
5	36,946	18,326	4,545	38,958	3,896
6	13,924	7,016	1,768	23,430	2,343

The starting conditions - Inventory control parameters and cost figures for the independent inventory control system - and the minimum cost for the dependent system are given in Table VI.5. If no joint replenishment is allowed in the inventory system, total independent cost comes out to be 9,562,604 TL. When no independent order is placed in the dependent inventory system, lower bound on the total dependent cost is 7,968,085 TL. The maximum possible saving which the dependent system can realize without any constraint on the service levels will then be 16.67 %.

Iterative algorithm gives a total cost of 8,532,800 at the end of 9 iterations realizing 10.77 % saving over the independent policy. Solutions for several iterations and final results are tabulated in Table VI.6. Simulated values of O_i ($OSIM_i$) are also given in the same table to compare the computed ones in the algorithm. Figure VI.1 shows the pattern of the total system cost through iterations of the solution algorithm.

TABLE VI.5. SOLUTION FOR THE INDEPENDENT SYSTEM

Item	P_i	ξ_i	O_i	s_i	S_i	$H_{I,i}$	$K_{I,i}$	$Z_{I,i}$
1	0	62,675	110,620	116,754	173,294	1,226,311	411,151	1,637,462
2	0	17,032	13,593	14,390	30,625	394,256	206,938	601,194
3	0	29,281	43,739	46,295	73,020	1,072,567	409,198	1,481,765
4	0	41,235	79,681	83,840	120,915	1,631,583	501,000	2,132,583
5	0	38,958	84,913	89,457	123,870	1,966,716	568,000	2,534,716
6	0	23,430	29,166	30,933	52,596	818,972	355,912	1,174,884
						H_I=7,110,405	K_I=2,452,199	Z_I=9,562,604

Total independent cost, Z_I : 9,562,604

Minimum dependent cost, Z_L : 7,968,085

Maximum possible saving, MPS : 16.67 %

TABLE VI.6. RESULTS

Iteration	Item	O_i	γ_i	$\frac{\partial Z}{\partial C_i}$	ξ_i	s_i	c_i	S_i	$Osim_i$	H_i	K_i	Z_i
1	1	110619	12535	–	62675	116754	123154	173294	111369	1239791	342347	1582138
	2	13355	3407	–	14947	14152	16302	28302	13467	383882	108493	492375
	3	43707	5856	–	29098	46263	49563	72805	43820	1087943	318280	1406223
	4	79680	8247	–	41235	83840	87927	120914	80454	1642683	450993	2093676
	5	84912	7792	–	38958	89457	92703	123869	86791	1975012	532117	2507129
	6	29034	4686	–	22584	30801	33719	51617	29677	820425	274755	1095180
										H=7149735	K=2026985	Z=9176720
2	1	109552	18803	–	58301	115687	128354	167853	110513	1211630	293371	1505001
	2	12729	5110	–	12580	13525	17838	25309	12924	356412	96332	452744
	3	43003	8784	–	26487	45559	51787	69490	42901	1058910	278364	1337275
	4	79211	12370	–	39601	83370	91581	118811	78647	1639596	385719	2025315
	5	84621	11687	–	38000	89166	96308	122621	86363	1972913	502901	2475814
	6	28588	7029	–	20871	30355	35617	49459	28633	802734	240923	1043657
										H=7042196	K=1797610	Z=8839806
3	1	108884	25070	–	53427	115018	133953	162310	109383	1204517	256422	1460938
	2	12037	6812	–	11751	12833	18849	23788	11981	348263	109423	457686
	3	42352	11712	–	24785	44908	54064	67136	42217	1041768	278248	1320016
	4	78291	16494	–	37212	82451	94784	115502	78932	1618672	376374	1995046
	5	84228	15583	–	37221	88772	99810	121448	85211	1966744	473984	2440728
	6	28065	9372	–	19648	29832	37437	47712	28385	805197	181418	986615
										H=6985162	K=1675869	Z=8661031

Iteration	Item	O_i	γ_i	$\frac{\partial Z}{\partial C_i}$	ξ_i	s_i	c_i	S_i	$Osim_i$	H_i	K_i	Z_i
6	1	108548	31338	-	53310	114682	139885	161857	109631	1221775	229007	1450782
	2	13398	5110	-	13842	14194	18507	27239	13421	381468	107198	488666
	3	41106	14640	-	23418	43661	55747	64523	41114	1009972	297001	1306972
	4	75700	20617	+	37817	79860	96317	113517	76197	1585651	381211	1966861
	5	81759	27270	-	35193	86303	109029	116951	81061	2034082	361333	2395416
	6	27281	7029	+	18989	29048	34309	46269	27807	726875	298340	1025215
										H=6959823	K=1674090	Z=8633913
9	1	108263	37605	-	55642	114397	145868	163905	108803	1254211	218715	1472926
	2	12234	3406	+	12044	13031	15648	24278	12255	316066	155471	471538
	3	41479	16104	-	25119	44035	57583	66598	41675	1074877	229705	1304582
	4	73727	24741	+	38656	77887	98468	112384	74480	1562742	360208	1922950
	5	82438	21427	+	36346	86982	103864	118783	82592	1978224	391882	2370105
	6	27515	10543	-	19202	29282	38058	46716	28168	790330	200370	990700
										H=6976450	K=1556351	Z=8532801
												AchS=.1077

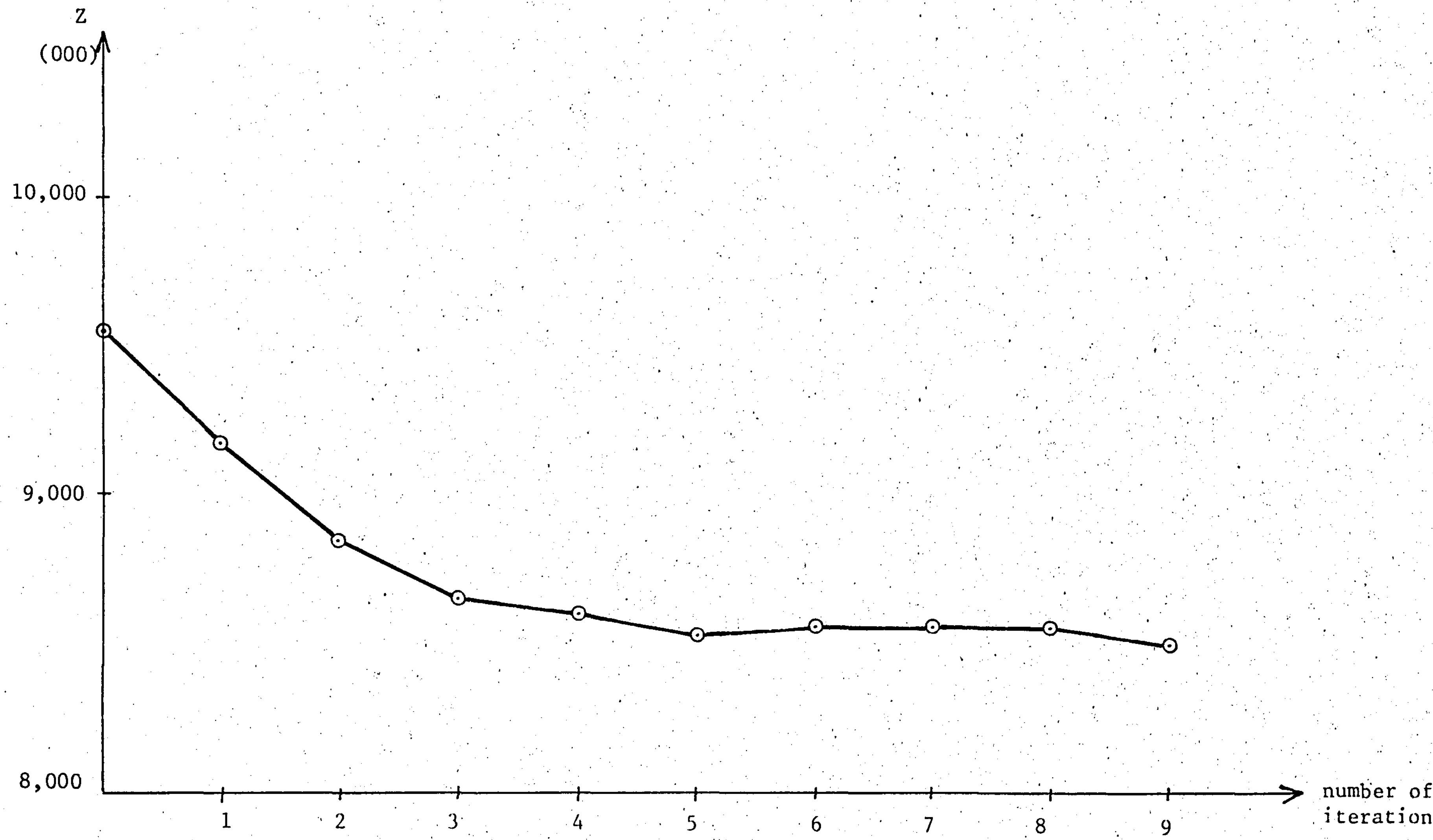


Figure VI.1. Total Cost Through Iterations of the Solution Algorithm

VI.2. ADVENTAGES OF THE RECOMMENDED (s,c,S) POLICY OVER THE INDEPENDENT (s,S) POLICY

The objective in controlling the inventories by a common policy in a multi-item inventory system is to use the fixed set-up cost of ordering in an economical way. When an increase occurs in the fixed set-up cost, if items are controlled by their own independent policies, the system will tend to place less number of orders in order not to increase the total ordering cost in the same percent.At the end, lower re-order points but higher order-up-to levels for all items will be observes. Placing orders rarely but in larger quantities results in carrying higher stocks for each item. Finally, the total cost will be highly influenced by the change in the ordering cost term. But if a dependent inventory policy is used, increase in the set-up cost will tend to have higher can-order points for items in order to decrease the number of independent replenishments. Joint ordering probabilities will be higher in the new system than in the old dependent system. For the new parameters, the resulting dependent policy will order from each item with an higher frequency but by smaller quantities. In short, total cost will be less effected by the change in the set-up cost in the dependent system than it would be in the case of independent system.

For alternative values of the set-up cost, total costs for independent and dependent systems and also percent increases are listed in Table VI.7. Percent increases in total independent and dependent costs vs. percent increase in set-up cost are plotted in Figure VI.2. As seen in the figure, recommended inventory policy has obsolute adventage over the dependent one.

Maximum possible saving, actual saving achieved by the algorithm in at most 10 iterations and achieved percent of maximum possible saving are given in Table VI.8 for several numerical examples. Then, actual achieved saving vs. maximum possible saving is plotted in Figure VI.3.

TABLE VI.7. TOTAL INDEPENDENT AND DEPENDENT COSTS FOR DIFFERENT SET-UP COSTS

Alternative values of KF	% increase in KF	Z_I	% increase in Z_I	Z	% increase in Z
10,000	-	8,556,451	-	8,119,953	-
15,000	50	9,088,598	6.22	8,301,490	2.24
20,000	100	9,562,604	11.76	8,532,801	5.08
25,000	150	9,994,679	16.81	8,761,161	7.90
30,000	200	10,394,630	21.48	8,879,601	9.36
40,000	300	11,122,017	29.98	9,386,254	15.59
50,000	400	11,777,219	37.64	9,644,928	18.78

TABLE VI.8.

Maximum Possible Saving (%)	Actual Saving achieved by the algorithm (%)	Percent Reached of MPS
12.3	5.1	41.5
14.8	8.7	58.8
16.7	10.8	64.7
18.1	12.3	68.0
19.3	14.6	75.6
21.0	15.6	74.3
22.4	18.1	80.8

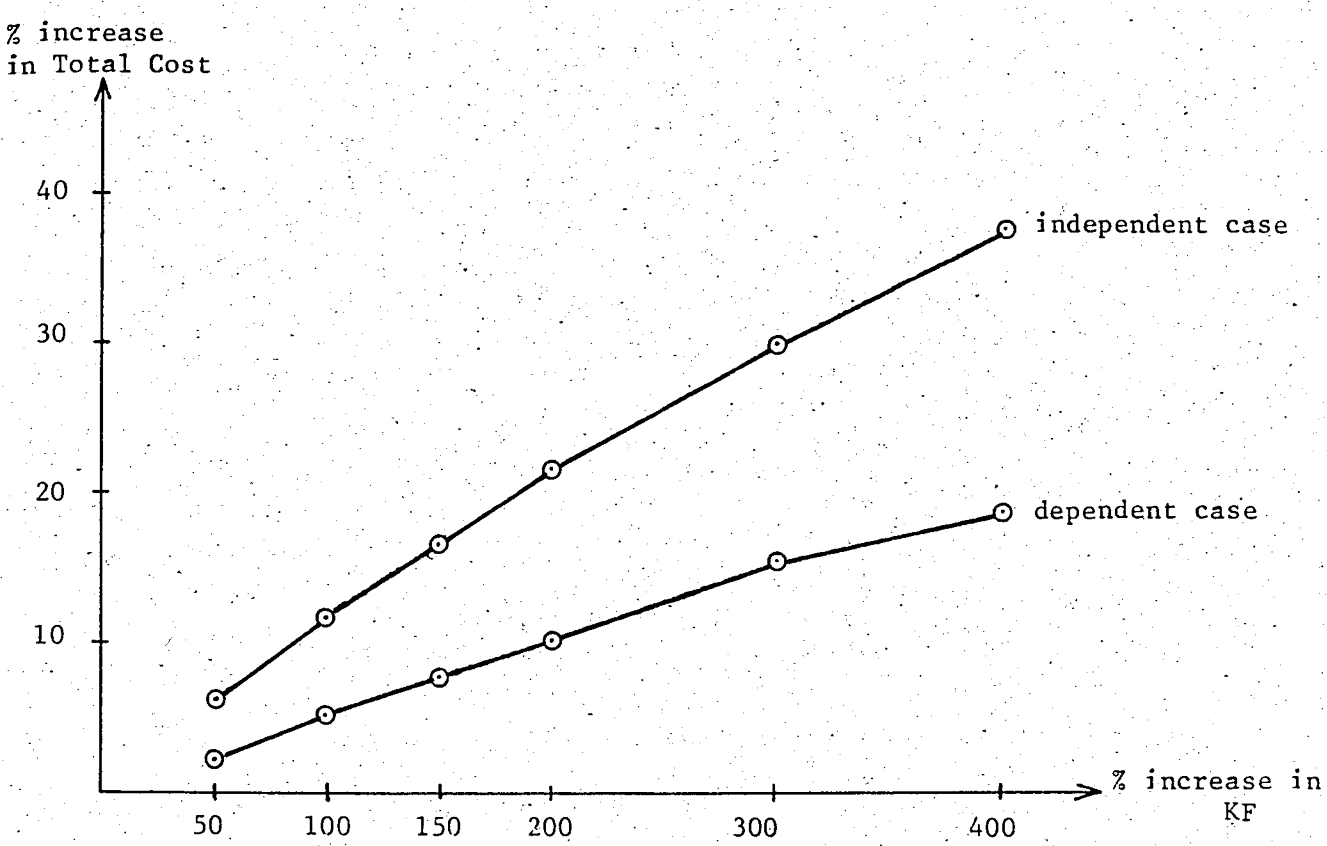


Figure VI.2. Adventages of the Dependent Policy Over the Independent Policy

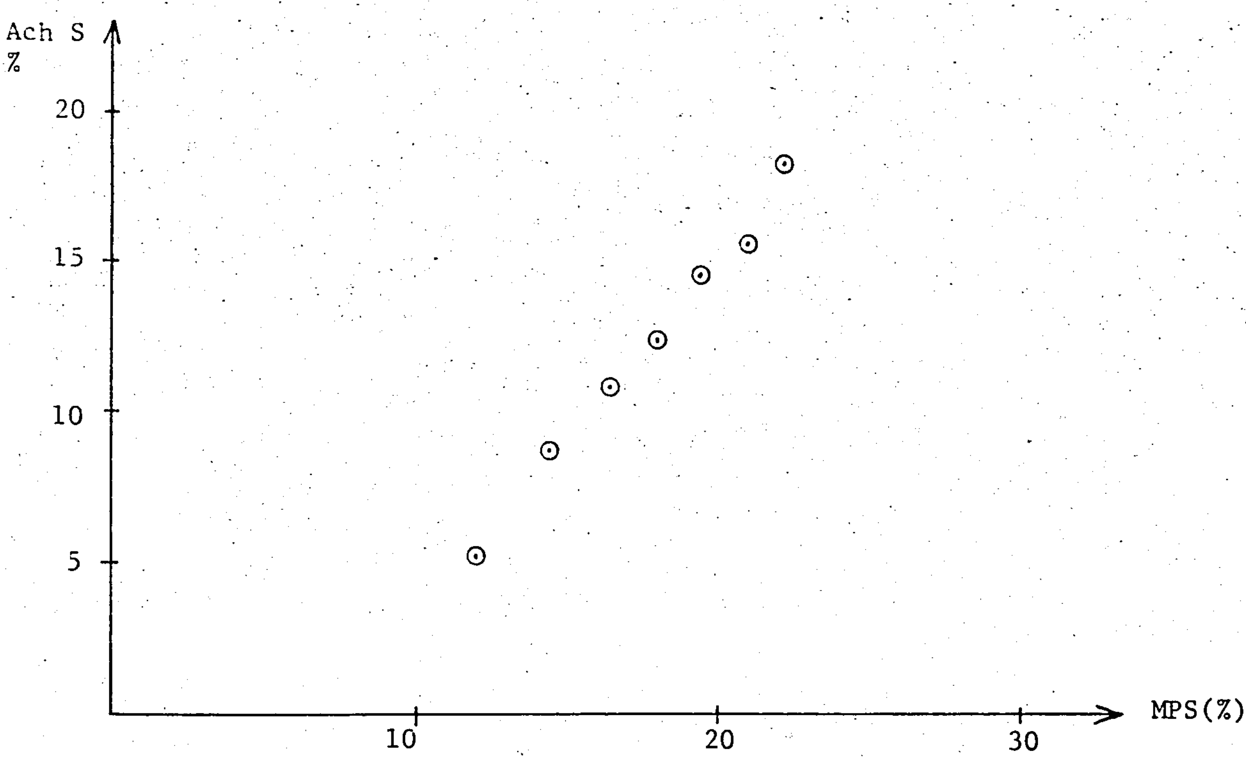


Figure VI.3. Actual Achieved Saving v.s. Maximum Possible Saving

VI.3. THE ALGORITHM'S PERFORMANCE UNDER CHANGES IN THE SYSTEM PARAMETERS

An interesting feature of the algorithm is its adaptability to changes in the values of the parameters. In a real situation, the demand for certain items will change with time. Decreases or increases in the demand parameters may take place. Secondly, change on the lead time may occur. Thirdly, new pricing practices instituted by the supplier will yield a change in the inventory carrying cost. Also fluctuations of the ordering cost might be encountered. Last, managers may propose new service levels for items.

When one of the cases is observed, the new inventory control parameters may be achieved solving the model from the beginning. But the respective changes can be introduced in the process of the solution and starting from the independent system will be hence avoided.

The algorithm performance over the change in demand figures is examined in the following. When demand parameters in Table VI.2 are changed as:

$$D'_1 = 0.90\ D_1$$

$$D'_2 = D_2$$

$$D'_3 = 0.95\ D_3$$

$$D'_4 = 0.90\ D_4$$

$$D'_5 = 0.95\ D_5$$

$$D'_6 = 0.90\ D_6$$

the solution given in Table VI.8. is found by the algorithm at the end of 9 iterations.

TABLE VI.8. SOLUTION FOR NEW DEMAND FIGURES

Item	D_i'	s_i	c_i	S_i	Z_i
1	1,090,985	99,087	137,546	143,127	1,333,710
2	147,120	12,186	18,202	24,766	466,896
3	475,124	43,498	51,644	66,182	1,318,267
4	745,974	77,443	86,975	110,214	1,871,714
5	877,466	83,259	100,548	113,636	2,291,355
6	313,283	25,315	35,773	43,210	914,775
					Z = 8,196,716

Curve 1 of Figure VI.4. shows the behaviour of the solution for the initial demand parameters (D_1, D_2, D_3, D_4, D_5, D_6).Curve 2 in the same figure corresponds to the system with the new demand figures (D_1', D_2', D_3', D_4', D_5', D_6'). As a third case, give the solution of the first system at the end of the fifth iteration (P_{i4}, P_{i5}, ρ_{i4}, ρ_{i5}, s_{i5}, c_{i4}, c_{i5}, S_{i5}, Z_{i5}, Z_5) as the initial condition to the second system. Instead of starting from the independent system, initiate from that point. Give the new parameters into the algorithm and continue through the algorithm. Curve 3 of Figure VI.4 shows the solution pattern for the third case.

These results show clearly that if changes in the parameters are encountered, it is not necessary to start from the beginning. An iteration based on their new values along with the ex-optimal values of the other parameters leads quickly to a new nearly optimal solution. In short, the algorithm is adaptable to changes in the values of the uncontrollable parameters. This property of the algorithm is very useful in reaching a new solution in a short time, when a change occurs in the system parameters.

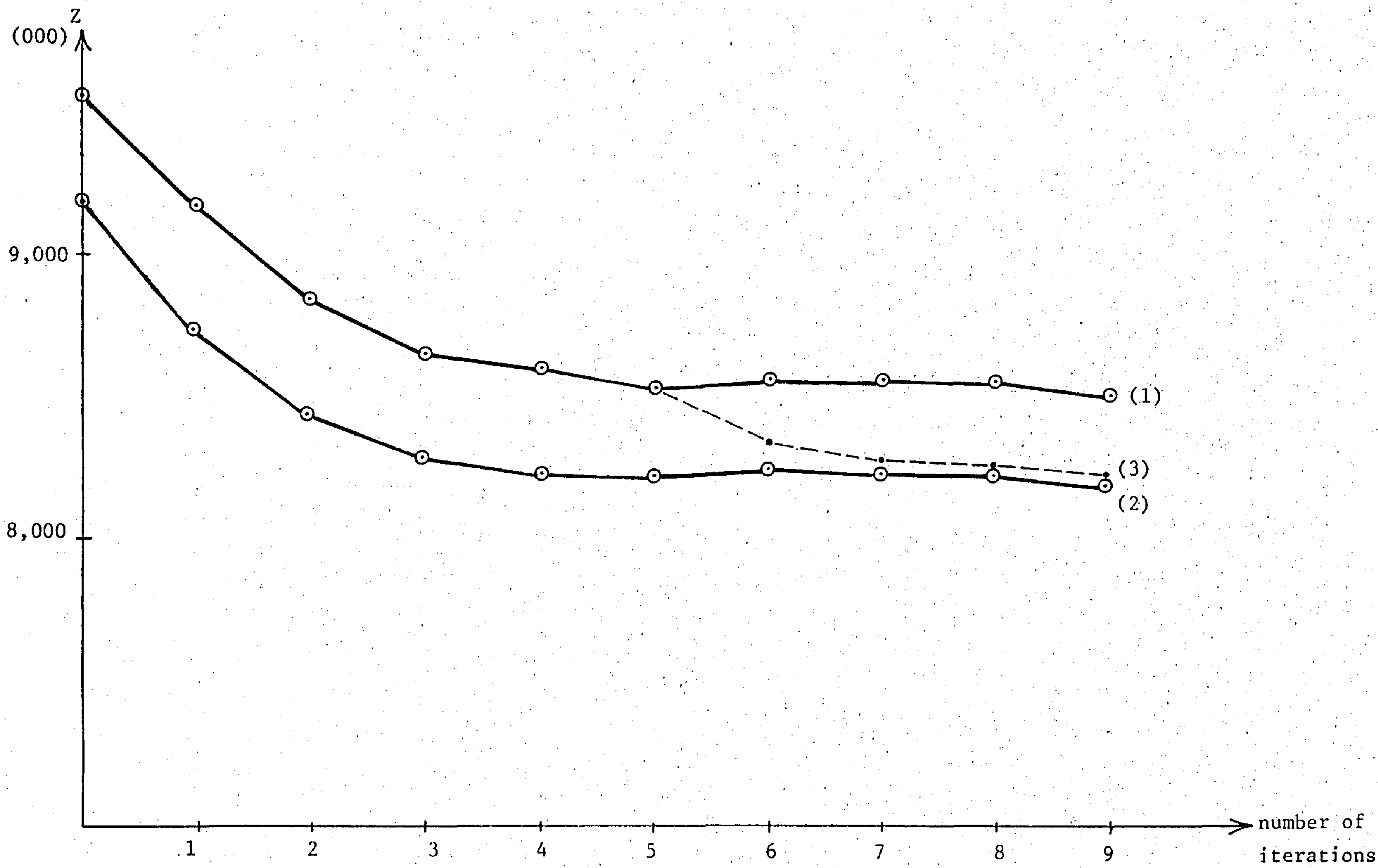


Figure VI.4. Algorithm's Performance under Changes in Parameter Values

CHAPTER VII
CONCLUSIONS

The objective of this study was to develop an inventory control system for warehouses where several groups of items, each of which is supplied by a given supplier, are stored. Consequently there is a fixed set-up cost for each replenishment from a given supplier. In order to use this fixed part of the ordering cost, the (s,c,S) policy is proposed to be used in controlling the inventories in the warehouses instead of using independent ordering policies for items.

In modelling the system, total inventory carrying cost and ordering cost are taken to be minimized in the objective function. The effects of the stock out cost are considered by the service level constraints, because of the difficulties of computing this cost term.

Implementation of the recommended joint ordering policy for a six-item case has been realized and it has been shown that taking the ordering decisions by the (s,c,S) policy instead of controlling each item by its independent (s,S)policy saves 929,804 TL. Dependent system achieves a reduction of 10.77 % in the total system cost over the independent system.When an increase in the cost parameters takes place, its effect on the total system cost will be less in the dependent system than it would be in the independent systems.

The iterative algorithm used for finding the values of the inventory control parameters of the (s,c,S) policy results in considerable improvements over the independent replenishment policies in a few iterations. The best that the algorithm can do may be observed in Figure VI.3 for several numerical examples. For the sample run, the algorithm achieves 64.61 % of the maximum saving achieved by the dependent policy without any service level criterion.

But the algorithm has weaknesses on two fronts. First, since the functional relationships between the variables ξ_i, O_i, R_i and P_i in the model can not be defined, simulation is required as a part of the algorithm in determining the values of P_i and ρ_i for the alternative values of s_i, c_i, S_i. Thus computer usage is required in the solution of the model.Secondly, updating of can-order points is not a complete optimization. As it can be seen from the results, the algorithm provides a fast stabilization of the S_i's. Their values approach the optimal values within a range of 10 percent after 2 or 3 iterations. In the remaining iterations, it tries to reach the optimal values of the c_i's by dichotomy method.

If the remnant stocks (R_i's) and joint ordering probabilities (P_i's) can be expressed as functions of s_i, c_i, S_i, simulation will not be required as a part of the algorithm. Also, if updating procedure for determining the can-order points is improved optimum will be reached in a shorter time.

Another criticism of the optimization algorithm concerns the updating of s_i. If the simulated values of O_i differ from their computed values too much, α_i given in Appendix 2 may be used as only an approximation in updating s_i, but the total cost can be evaluated using the simulated values of O_i's.

A very useful property of the algorithm is its adaptability to the changes in the uncontrollable system parameters such as demand rates, ordering and holding costs, service criteria etc. When a change occurs in one of these parameters, it is possible to reach the optimum continuing from the ex-optimal values of the other parameters using this property of the algorithm.

In order to set up an overall inventory control system in a district warehouse of the whole inventory system, items coming from the same supplier will be handled in the same set. The model will be run separetely for each group and optimal values of the three inventory control parameters will be determined for each item. Then the total inventory policy for a district warehouse will be described combining the separate inventory policies of all groups of items.

In the implementation of the study, if inventories are controlled on cards, three critical inventory parameters are kept for each item and items coming from the same factory are put in the same file. When the inventory of an item drops below its must-order point, only items in this file will be searched for their can-order points. Thus, the review of items will not be a very time-consuming task and can easily be done manually.

BIBLIOGRAPHY

1. Andres, F., H.Emmons, "A Multiproduct Inventory System with Interactive Set-up Costs", Management Science, Vol.21, No.9, May 1975, pp.1055-1063.

2. Balintfy, J.L., "On a Basic Class of Multi-item Inventory Problems", Management Science, Vol.10, No.2, January 1964, pp.287-297.

3. Brown, R.G., Decision Rules for Inventory Management, New York:Holt, Rinehart and Winston, 1967.

4. Çınlar, E., Introduction to Stochastic Processes, Prentice Hall, Inc., Englewood Cliffs, New Jersey 1975.

5. Goyal, S.K., "A Method for Improving Joint Replenishment Systems with a Known Frequency of Replenishment Orders", Int.J.Prod.Res., Vol.11, No.2, 1973, pp.195-200.

6. Goyal, S.K., "Analysis of Joint Replenishment Inventory Systems with Resource Restriction", Operational Research Quarterly, Vol.26, No.11, 1973, pp.197-203.

7. Goyal, S.K., "Determination of Economic Packaging Frequency For Items Jointly Replenished", Management Science, Vol.20, No.2, October 1973, pp.232-235.

8. Goyal, S.K., "Determination of Optimum Packaging Frequency of Items Jointly Replenished", Management Science, Vol.21, No.4, December 1974, pp.436-443.

9. Goyal, S.K., "Economic Packaging Frequency of Perishable Jointly Replenished Items", Operational Research Quarterly, Vol.28, No.1, 1977, pp.215-219.

10. Hadley, G., T.M.Whitin, Analysis of Inventory Systems, Prentice-Hall, Inc., Englewood Cliffs, N.J. 1963,

11. HO, Chung-Mei, "A Multi-Product Joint Ordering Model with Dependent Set-up Cost", Management Science, Vol.20, No.7, March 1971, pp.1081-1091.

12. Hocking, G.H., Calculus, Hold, Rinehart and Winston, Inc. 1970.

13. Karlin, S., "Applications of Renewal Theory to the Study of Inventory Policies" in Studies in the Mathematical Theory of Inventory and Production, Stanford University Press, 1958.

14. Mariani, L., B.Nicoletti, "Optimization of Deterministic, Multiproduct Inventory Model with Joint Replenishment", Management Science, Vol.20,No.3, November 1973, pp.349-362.

15. Meyer, P.L, Introductory Probability and Statistical Applications, Addison-Wesley, 1972.

16. Naddor, E., Inventory Systems, John Wiley and Sons, 1966.

17. Oral, M., M.S.Salvador, A. Reisman, and B.V.Dean, Industrial Inventory Control, Gorbon and Breach, New York, N.Y., 1972.

18. Phillips, G.M., P.J.Taylor, Theory and Applications of Numerical Analysis, Academic Press, 1973.

19. Salvadori, M.G., M.L.Baron, Numerical Methods in Engineering, Prentice-Hall, 1961.

20. Scarborough, J.B., Numerical Mathematical Analysis, The Johns Hopking Press, 1961.

21. Schaack, J.P., "An Approach to Selecting the Control Variables of an (S,c,s) Joint Ordering Strategy", Department of Management Sciences, University of Waterloo, MA SC Dissertation, September 1970.

22. Schaack, J.P., E.A.Silver, "A Procedure Involving Simulation for Selecting the Control Variables of an (S.c,s) Joint Ordering Strategy", Infor,Vol.10, No.1, February 1972, pp.154-170.

23. Schmidt, J.W., R.E.Taylor, Simulation and Analysis of Industrial Systems, Richard D.Irwin, Inc., 1970.

24. Stanton, R.G., Numerical Methods for Science and Engineering, Prentice-Hall, 1961.

25. Starr, M.K., D.W.Miller, Inventory Control: Theory and Practice, Prentice-Hall, 1962.

26. Sümerbank Genel Üretim Planlaması ve Mamul Seçimi Projesi, Son Rapor, MAE-YB, proje kod no: 01-20-02-78-01, Haziran, 1979.

27. Sümerbank Fabrika Düzeyinde Araştırma Projesi, "Eskişehir Basma Sanayii Müessesesi", Son Rapor, MAE-YB, proje kod no: 01-20-03-78-01, Temmuz, 1979.

APPENDIX I

EXAMINATION OF DEMAND CHARACTERISTICS

I.I. THEORY ABOUT THE DEMAND DISTRIBUTION

Definition 1: Let Ω be a sample space and w be any realization on it. A stochastic process $N=\{N_t;\ t\geq 0\}$ defined on Ω will be an "arrival process" such that for any $w\in\Omega$, the mapping $t\rightarrow N_t(w)$ is non-decreasing, increases by jumps only, is right continuous, and has $N_0(w)=0$.

Definition 2: An arrival process $N=\{N_t;\ t\geq 0\}$ is called a "Poisson Process" provided that:

i) For almost all $w\in\Omega$, each jump of $t\rightarrow N_t(w)$ is of unit magnitute.

ii) For any $t,s\geq 0$, $N_{t+s} - N_t$ is independent of $\{N_u;\ u\leq t\}$.

iii) For any $t,s\geq 0$, the distribution of $N_{t+s} - N_t$ is independent of t, but is dependent on s.

Lemma 1 : for all $t\geq 0$,

$$P\{N_t=0\}=e^{-\lambda t} \quad \text{for some constant } \lambda\geq 0.$$

Lemma 2 : We have

$$\lim_{t\rightarrow 0} \frac{1}{t} P\{N_t\geq 2\}=0.$$

Lemma 3 : We have

$$\lim \frac{1}{t} P\{N_t=1\}=\lambda \quad \text{where } \lambda \text{ is the constant appearing in Lemma 1.}$$

Theorem 1 : If $\{N_t;\ t\geq 0\}$ is a Poisson process, then for any $t\geq 0$,

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

for some constant $\lambda\geq 0$.

Expected value and variance of Poisson process with parameter λ are given by

$$E[N_t] = \lambda t \quad \text{and} \quad Var(N_t)=\lambda t \ .$$

Proposition 1: If $T_1, T_2, \ldots$ are the successive times of jumps, for any $n \geq 0$,

$$P\{T_{n+1} - T_n \leq t / T_o, \ldots, T_n\} = 1 - e^{-\lambda t}, \qquad t \geq 0 \ .$$

In other words, the interarrival times $T_1, T_2-T_1, T_3-T_2, \ldots$ are independent and identically distributed random variables with the common distribution being

$$1 - e^{-\lambda t}, \quad t \geq 0 \ .$$

The distribution above is called the "exponential distribution" with parameter λ. The expected value and the variance of interarrival times in a Poisson Process are

$$E[T_{n+1} - T_n] = \frac{1}{\lambda} \quad \text{and} \quad Var\,(T_{n+1} - T_n) = \frac{1}{\lambda^2} \ .$$

Definition 3: If the restriction of unit jump size is removed from the definition of Poisson process and allowed jumps of any size, the stochastic process $Z=\{Z_t;\ t \geq 0\}$ is said to be a "compound Poisson" process. Then, the definition of compound Poisson process will be such that:

i) for almost all $w \in \Omega$, the function $t \to Z_t(w)$ has only finitely many jumps in any finite interval.

ii) and iii) for $N=\{N_t;\ t \geq 0\}$ in definition 2 are the same for $Z=\{Z_t;\ t \geq 0\}$.

A possible realization of compound Poisson process is shown in Fig. I.1.

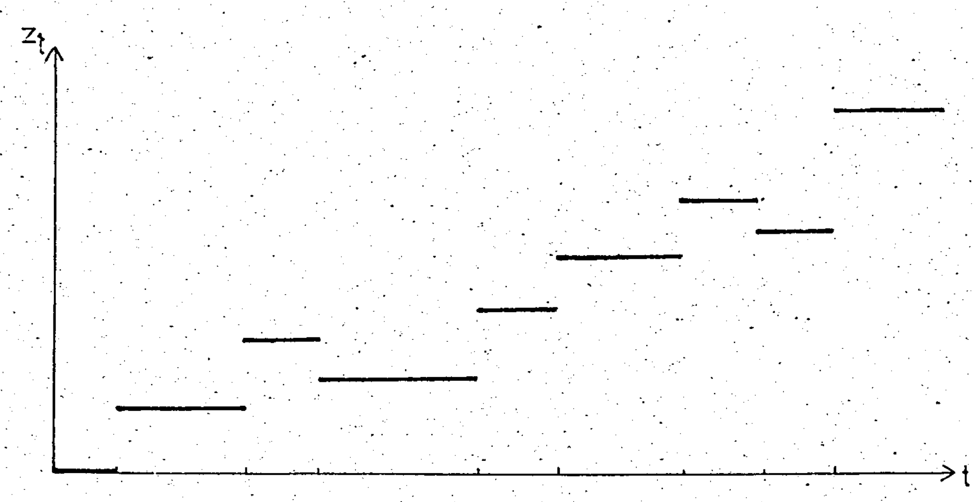


Figure I.1. A Possible Realization of Compound Poisson Process

If T_1, T_2,...are the arrival times in Poisson process N_t and if Y_1, Y_2,... are independent and identically distributed random variables which are also independent of the T_n, then the process Z obtained by summing up all the Y_j for which $T_j \leq t$ to make up Z_t is a compound Poisson process.

$$Z_t = \sum_{n=1}^{N_t} Y_j^n , \quad P\{N_t = k\} = \frac{e^{-\lambda t}(\lambda t)^k}{k!} \qquad k=0,1,\ldots \text{ for some constant } \lambda \geq 0$$

where Y_j is independent and identically distributed random variable with mean m and standard deviation σ.

Characteristic parameters of the compound Poisson process may be evaluated as follows:

i) $E[Z_t]$

If the number of jumps N_t of Z in (0,t] is n, then Z_t is the sum of n independent and identically distributed random variables. Hence, if $E[Y_j]=m$ and the rate of jumps is λ, then

$$E[Z_t/N_t] = E[Y_1+Y_2+\ldots+Y_{N_t} / N_t] = N_t.E[Y_j] = m N_t$$

$$E[Z_t] = E[E[Z_t / N_t]] = m\, E[N_t]$$

$$E[Z_t] = m\lambda t$$

ii) $Var(Z_t)$

Variance of the Z_t may also be derived as follows:

Definition of the variance was

$$Var(X) = E[(X - E[X])^2]$$

$$= E[X^2 - 2\, X\, E[X] + (E[X])^2]$$

$$= E[X^2] - (E[X])^2 \ .$$

Same formula is also valid for Z_t :

$$Var(Z_t) = E[Z_t^2] - (E[Z_t])^2 \ .$$

Substituting

$E[N_t] = \lambda t$, $Var(N_t) = \lambda t$, $E[N_t^2] = \lambda t + \lambda t^2$

$E[Z_t/N_t] = m\, N_t$

$$Var(Z_t/N_t) = Var(Y_1 + .. + Y_{N_t} / N_t) = N_t\ Var(Y_j) = N_t \cdot \sigma^2$$

and using some facts in probability theory

$$Var(Z_t) = E[E[Z_t^2/N_t]] - (E[Z_t])^2$$

$$= E[[Var(Z_t/N_t) + (E[Z_t/N_t])^2] - (E[Z_t])^2$$

$$= E\ [\ N_t\ \sigma^2 + N_t^2\ m^2] - m^2\lambda^2 t^2$$

$$= \sigma^2\ E[N_t] + m^2\ E[N_t^2] - m^2\lambda^2 t^2$$

$$= \sigma^2\lambda t + m^2(\lambda t + \lambda^2 t^2) - m^2\lambda^2 t^2$$

then

$$Var(Z_t) = \lambda t(m^2 + \sigma^2).$$

I.2. LEAD TIME DEMAND

In the system presented, expected value of yearly demand Z_t is taken as D, and the same parameters m and σ^2 are used for the mean and the variance of the distribution of jump size respectively. Then analysing $E[Z_t]$, rate of arrival of demand will come out as $\lambda=D/m$.

By Proposition 1, interarrival time $\{S_j\}$ of poisson distribution follows exponential distribution. Then, for arrival rate $\lambda=D/m$,

$$E[S_j] = m/D \qquad Var(S_j) = m^2/D^2.$$

Proposition 2: For a continuous review inventory system with constant replenishment lead times, intervals of time between successive demand points and quantities demanded at each point form independent sequences of independent, identically distributed random variables with distribution function A(.) and B(.) respectively. If the number of arrivals during lead time L is large, demand during lead time can be approximated as [13]:

$$D(L) \sim N\left(\frac{\mu_1 L}{\mu_2}\, ,\ \frac{\sigma_1^2\ L}{\mu_2} + \frac{\mu_1^2\ \sigma_2^2\ L}{\mu_2^3}\right)$$

where the subscripts 1 and 2 are used for demand size and interarrival time, and the parameters μ and σ are used for means and standard deviations of the distributions respectively.

Substituting the parameters used in the study as:

$$\mu_1 = m \qquad \sigma_1 = \sigma \qquad L = L$$

$$\mu_2 = \frac{m}{D} \qquad \sigma_2 = \frac{m}{D}$$

distribution of demand during lead time will come out as

$$D(L) \sim N\left(\frac{m\ L}{m/D}\, ,\ \frac{\sigma^2 L}{m/D} + \frac{m^2(\frac{m^2}{D^2})\ L}{(m/D)^3}\right)$$

and

$$\sim N\left(DL,\ \frac{DL}{m}\ (\sigma^2 + m^2)\right).$$

APPENDIX II

PROOF OF $\alpha = \frac{1}{2} \left(\frac{m^2 + \sigma^2}{m} \right)$

STATEMENT:

Z_t is a compound Poisson process with rate λ and jump sizes $\{Y_i\}$ are independent, identical non-negative random variables with mean m and variance σ^2. When the Z_t raises above a specified level Δ at the first time, the difference between the Z_t and Δ has expected value of $\frac{1}{2}(\frac{m^2+\sigma^2}{m})$.

For fixed $\Delta > 0$, define

$$u_\Delta = \inf\{t \geq 0;\ Z_t > \Delta\}$$

Find $f(\Delta) = E[Z_{u_\Delta} - \Delta]$

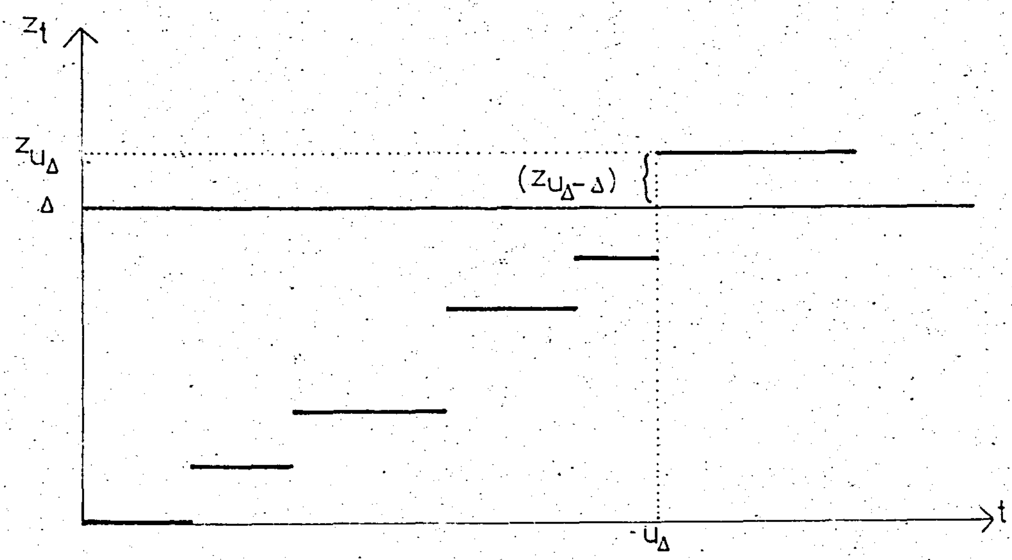


Figure II.1.

HEORY:

efinition 1: Let $\{W_i\}$ be a sequence of i.i.d. and non-negative random variables. hen, the stochastic process $S=\{S_n;\ n\geq 0\}$ with state space R_+ defined by $S_o=0$, $_{n+1}=S_n+W_{n+1}$ will be called a "renewal process".

efinition 2: Let ψ be a distribution function on R_+, and let f be a non-negative unction defined on R_+ which is bounded over any finite interval [0,t]. Then he function $\psi * f$ defined by

$$\psi * f(t) = \int_{[0,t]} \psi(ds)\ f(t-s), \qquad t \geq 0$$

is called the "convolution" of ψ and f.

Definition 3: $S=(S_n, n\varepsilon N)$ is a renewal process and F is the distribution of the interarrival times. For any m, the distribution of $S_{n+m} - S_n$ is the m-fold convolution F^m of F with itself. Along with the independence of the increments, this implies that

$$P\{S_{n+m} - S_n \leq t/S_o, \ldots, S_n\} = F^m(t), \qquad t \geq 0.$$

Definition 4: If N_t is the number of renewals in the interval [0,t],

$$P\{N_t=k\} = P\{S_{k-1} \leq t\ ,\ S_k>t\} = P\{S_{k-1} \leq t\} - P\{S_k \leq t\}$$

$$= F^{k-1}(t) - F^k(t)$$

Definition 5: The expected number of renewals in [0,t] will be

$$E[N_t] = \sum_{n=0}^{\infty} E[I_{[0,t]}(S_n)]$$

$$= \sum_{n=0}^{\infty} P\{S_n \leq t\} = \sum_{n=0}^{\infty} F^n(t) = 1+F+F^2+F^3 + \ldots.$$

Then the function $R(t)=E[N_t]=1+F+F^2+F^3 +\ldots$ is called the "renewal function" corresponding to the distribution F.

Definition 6: A renewal process S is said to be "recurrent" if $W_n<+\infty$ almost surely for every n; otherwise S is called "transient". $F(\infty)=\lim_{t\to\infty} F(t)$ will be equal to 1 and less than 1 for recurrent and transient processes respectively.

Definition 7: A renewal process S is said to be "periodic" with period σ if the random variables $W_1, W_2, \ldots$ take values in a discrete set $\{0, \sigma, 2\sigma, \ldots\}$ and σ is the largest such number. Otherwise, if there is no such $\sigma>0$, S is said to be "aperiodic".

Definition 8: $f(t)=g(t) + \int_{[0,t]} F(ds)\, f(t-s)$ is called a "renewal equation" where F is a distribution on R_+, f and g are functions bounded over finite intervals and $g(t) = \int_t^\infty (y-t)\, df(y)$.

Theorem 1: The renewal equation $f=g+F*f$ has one and only one solution; it is

$$f=R*g$$

where $R = \sum^\infty F^n$ is the renewal function corresponding to F.

Theorem 2 (Key Renewal Theorem):

i) If $F(\infty)<1$, then

$$\lim_{t\to\infty} f(t) = \lim_{t\to\infty} R*g(t) = R(\infty) \cdot g(\infty)$$

provided that $g(\infty) = \lim_{t\to\infty} g(t)$ exists.

ii) If $g \in D$, $F(\infty)=1$, and F is not arithmetic, then

$$\lim_{t\to\infty} f(t) = \lim_{t\to\infty} R*g(t) = \frac{1}{m} \int_0^\infty g(y)\, dy \,.$$

PROOF:

By the statement of the problem, $f(\Delta)$ can be rewritten in the form

$$f(\Delta) = E[Z_{u_\Delta} - \Delta]$$

$$= \int_\Delta^\infty (y-\Delta)\, dF(y) + \int_{[0,\Delta]} f(\Delta-y)\, dF(y) \qquad \Delta \geq 0.$$

Hence, if we define

$$g(\Delta) = \int_{\Delta}^{\infty} (y-\Delta)\ dF(y)$$

then, we have

$$f(\Delta) = g(\Delta) + \int_{[0,\Delta]} f(\Delta-y)\ dF(y).$$

By definition 8, $f(\Delta)$ will be a renewal equation satisfying

$$f = g + F * f\ .$$

By theorem 1, $f(\Delta)$ has a unique solution and is

$$f(\Delta) = \int_{[0,\Delta]} R(du).\ g(\Delta-u)$$

where $R(u) = \sum_{n=0}^{\infty} F^{n}(u)$ by definition 5.

By Key Reneway Theorem,

$$\lim_{\Delta\to\infty} f(\Delta) = \frac{1}{m} \int_{0}^{\infty} g(u)du$$

Substituting the equivalent of $g(u)$ into the equation above,

$$\lim_{\Delta\to\infty} f(\Delta) = \frac{1}{m} \int_{0}^{\infty} [\int_{u}^{\infty} (y-u)\ dF(y)]\ du.$$

Changing the order of integration,

$$= \frac{1}{m} \int_0^\infty dF(y) \left[\int_0^y (y-u)du \right]$$

$$= \frac{1}{m} \int_0^\infty \frac{y^2}{2} dF(y)$$

$$= \frac{1}{m} E[Y^2] = \frac{1}{2m} (Var(Y_i) + E[Y_i]^2)$$

and, finally

$$\lim_{\Delta\to\infty} f(\Delta) = \frac{\sigma^2+m^2}{2m} .$$

Therefore, for $\Delta \gg 0$

$$E[Z_{u_\Delta} - \Delta] \simeq \frac{1}{2} \frac{\sigma^2+m^2}{m} .$$

APPENDIX III

COMPUTATION OF INVENTORY CARRYING COST

Computation of inventory carrying cost was done as follows:

1. Value of the average inventory carried in İstanbul District Warehouse throughout the year, $\overline{I}$.

   Total yearly stock carried in 1979 : 9,972,000,000 TL

   $\overline{I}$ = 9,972,000,000 /12 = 831,000,000 TL

2. Labor costs, L.

   Total number of personnels worked in the warehouse : 25

   Average salary : 18,000 TL/month

   Total labor cost : 18,000 x 25 x 12 = 5,400,000 TL/year

3. Storage cost, S.

Storage cost which contains electricity, maintenance, cleaning supplies and other expences is estimated as 2,5 % of average inventory carried per year.

$S = \overline{I} \times 2,5\ \%$

$= 831,000,000 \times 2,5\ \%$

$= 20,775,000$ TL/year

4. Cost of Capital, C.

Since Sümerbank is a public firm, it is not possible investing money in any other area. This component of the inventory carrying cost is taken as the interest rate charged by Secretary of Finance due to delay in paying income tax. Then, this figure is taken as 18 % per annum.

$C = \overline{I} \times 18\ \%$

$= 831,000,000 \times 18\ \%$

$= 149,580,000$ TL/year

5. Insurance cost, I.

Inventory carried in the warehouse is insurred by 3.5 % per annum. Then

$$I = \overline{I} \times 3,5\ \%$$

$$= 831,000,000 \times 3,5\ \%$$

$$= 29,085,000\ \text{TL/year}$$

6. Inventory turnover, TR.

$$TR = \text{Sales for } 1979/\overline{I}$$

$$= 2,114,894,312\ /\ 831,000,000$$

$$= 2.55\ \%$$

7. Inventory Carrying Cost, H.

$$H = \frac{L/TR + S + C + I}{\overline{I}}$$

$$= \frac{5,400,000/2.55+20,775,000+149,580,000+29,085,000}{831,000,000}$$

$$= .243$$

Therefore, inventory carrying cost will be 24.3 % of inventory value per annum.

APPENDIX IV

COMPUTATION OF ITEM - ORDERING COST

Computation of item-ordering cost (KJ_i) will be given in Appendix IV. Item-ordering cost is a cost constant per order per item and has two important components:

1. First component is the part of production cost, charged in the factory,in order to supply orders from each item in time and to have opportunity of preparing a good lot assorted in colour and design. In order to have this opportunity, there must be enough production from each item in factories for orders of district warehouses. This leads to machine set-up cost in factories.

Machine set-up cost for starting production of each item will be computed and then item set-up cost will be divided among orders of districts according to their sales percentages.

Machine set-up cost takes place when a type of item starts to be produced.Method used for computing machine set-up cost can be found in [27]. Computations for 1979 prices are given below:

Machine set-up cost consists of two factors:

i) Damage cost, DC : Damages of models used in weaving machines occur with a certain probability during production and this cost is taken as the expected expense for models. Computation of damage cost was evaluated for each machine as follows:

$$DC = MC \times N \times P$$

where MC = Unit model cost

N = Average number of models

P = Probability of damage of models in this machine .

Type of machine	Cost of model* (TL)	Average number of models	Probability** of damage	DC (TL)
Roller	15680	3	0.45/6	3528
Rotation	5880	6	0.5/6	2940
Film	9800	9	0.35/6	5145

*It is computed for 1979 with 40% increase compared to 1978 costs.

**Probability of damage for each machine is given for 6 design changes.Then probability of damage for one design change is computed dividing the given probabilities by 6.

ii) Lost production cost, LPC: It is the cost of machine hours spent as set-up time for changing design and colour. This opportunity cost is computed as the extra profit which would be done if set-up time were used for production. The formula used for computation of lost profit cost for each machine is:

$$LPC = AP \times P \times (DT-CT) \times 60$$

where
AP = Average profit done per meter of production on the machine.
P = Amount of production per minute
DT = Set-up time for changing design (in hours)
CT = Set-up time for changing colour (in hours)

Type of Machine	AP*(TL)	P (m)	DT (hr)	CT (hr)	LPC (TL)
Roller	3.77	17.5	2.8	1.1	6750
Rotation	14.1	27.5	2.8	1.2	37224
Film	16.93	12.5	2.8	1.5	16507

*Average profit is computed as the weighted mean of profits gained for unit productions of items manufactured in the machine.

Total set-up cost for each machine will be the sum of damage cost and the lost production cost:

$$TSC = DC + LPC$$

Type of machine	DC (TL)	LPC (TL)	TSC (TL)
Roller	3528	6750	10258
Rotation	2940	37224	40164
Film	5145	16507	21652

Distribution of machine set-up cost among items analyzed in the study was done as follows:

Item, type no:	Machines used for its production	Average production set-up cost* for the item (TL)
205	Roller	10258
214	Rotation, Film	30908
283	Rotation, Film	30908
290	Rotation	40164
293	Rotation, Film	30908
722	Rotation, Film	30908

*Production set-up cost for each item is computed as the average of set-up costs of machines which are used in production of item.

Since 24.52 % of production of Eskişehir Factory is send to Istanbul District, production set-up costs for the items are taken as 24.52 % of the total production set-up costs in the factory. Each time when an item is ordered by Istanbul District, half of its production set-up cost was evaluated belonging to this order lot analyzing the economical production lot sizes for items in [27].

Then cost charged for each order of an item when the item is demanded from the factory will be computed as:

OC = .2452 x 1/2 x Average Production Set-up Cost for the item

Computed values for this cost is given below:

Item, type no:	OC (TL)
205	1090
214	3789
283	3789
290	4756
293	3789
722	3789

2. Second component of item-ordering cost is the cost of loading and unloading. It is paid 168 TL for loading and unloading of each lot in the recent system. Then this component of item-ordering cost (LULC) was taken as 168 for all items.

Finally, item-ordering cost (KJ_i) for each order of item i between Eskişehir Factory-Istanbul District Warehouse pair is computed as:

$$KJ_i = OC_i + LULC$$

Item no:	Type no:	OC_i	LULC	KJ_i ($=OC_i$+LULC)
1	205	1090	168	1258
2	214	3789	168	3957
3	283	3789	168	3957
4	290	4756	168	4924
5	293	3789	168	3957
6	722	3789	168	3957

APPENDIX V

LIST OF COMPUTER PROGRAM

```
@RUN,E  BELGIN,111-15-216,THESIS,5,100      .BELGIN  (ILHAN)  VARDAR.
CC
CC
CC     THIS  PROGRAM  DETERMINES  THE  OPTIMAL  VALUES  OF  THE  STOC
CC     CONTROL  PARAMETERS  ( OP,COL,OUL )  FOR  THE  (R,C,S) JOINT
CC     ORDERING  INVENTORY  POLICY
CC     IT  USES  AN  HEURISTIC  ALGORITHM , WHICH  IS  A  COMBINATION
CC     OF  MATHEMATICAL  OPTIMIZATION  AND  SIMULATION , FOR OPTIMIZA
CC
CC
       PARAMETER     N=6
       INTEGER       OUL,COL,OP,COLD
       REAL          LTIME,MDLT,MNREP,MPS,MINTRC,MTRAN
       DIMENSION     K2(N),K3(N),H(N),SAFES(N)
       DIMENSION     EOQ(N),DELTA(N),A(N),Y(N),COLD(N),PO(N),GO(N)
       DIMENSION     RC(N),RCO(N),RCI(N),FLAGRC(N)
       COMMON /CS1/  D(N),APRO(N),O(N),E(N)
       COMMON /CS2/  P(N),G(N)
       COMMON /CS3/  MDLT(N),SDDLT(N)
       COMMON /CS4/  OUL(N),COL(N),OP(N)
       COMMON /CS5/  MTRAN(N),SDTRAN(N),DRATE(N),LTIME,DURSIM,IR
       COMMON /CS6/  NORRAN
       COMMON /CS7/  OSIM(N),RSIM(N)
       COMMON        NO
       DATA          NI,NO / 5,6 /
       DATA          DURSIM,IR,NORRAN / 3.,13557,12/
       DATA          ACCS / 0.80 /
CC
CC     FUNCTION  DEFINITIONS
CC
       TRCFUN(PP,KK3,DD,EE,GG,HH,KK2,OO,MMDLT)=
      +(1-PP)*((KK3*DD/(EE-PP*GG))+(EE*HH/2))+PP*((KK2*DD/(EE-PP*GG))
      +(EE+GG)*HH/2) +OO*HH-MMDLT*HH
       EFUN(PP,GG,DD,HH,KK3,KK2)=
      +          PP*GG+SQRT((2*DD/HH)*((1-PP)*KK3+PP*KK2))
       DERFUN(P1,G1,PO,GO,C1,CO,HH,DD,KK1,EE)=
      +          (P1*G1-PO*GO)*HH/(C1-CO)-(DD*KK1)*(P1-PO)/((EE-P1*G1)*
      +          (C1-CO))
       NCYCLE=0
CC
CC     DATA  READING
CC
       READ(NI,101)  K1,LTIME
       WRITE(NO,1119) K1
       DO  201  I=1,N
       READ(NI,103) D(I),MTRAN(I),SDTRAN(I),H(I),APRO(I),K2(I)
       WRITE(NO,103)D(I),MTRAN(I),SDTRAN(I),H(I),APRO(I),K2(I)
   201 CONTINUE
CC
```

```
CC      EVALUATION  OF  THE  NECESSARY  SYSTEM  PARAMETERS
CC
        DO  301  I=1,N
        K3(I)=K1+K2(I)
        MDLT(I)=D(I)*LTIME
        SDDLT(I)=SQRT((D(I)*LTIME)    *(MTRAN(I)**2+SDTRAN(I)**2)/MTRAN(
        EOQ(I)=SQRT(2*D(I)*K3(I)/H(I))
        A(I)=0.5*(MTRAN(I)**2+SDTRAN(I)**2)/MTRAN(I)
        DRATE(I)=D(I)/MTRAN(I)
        WRITE(NO,165)   K3(I),MDLT(I),SDDLT(I),EOQ(I),A(I),DRATE(I)
  301   CONTINUE
CC
CC      INITIAL  VALUES  FOR  THE  UPDATING  PARAMETERS
CC
        DO  401  I=1,N
        P(I)=0
        PO(I)=0
        GO(I)=0
        G(I)=0
        E(I)=EOQ(I)
        Y(I)=DELTA(I)
        DELTA(I)=EOQ(I)/10
        O(I)=EOQ(I)
  401   CONTINUE
CC
CC      COMPUTATIONS  FOR  INDEPENDENT  INVENTORY  CONTROL  SYSTEM
CC
        CALL          OSUB
        DO  417  I=1,N
        WRITE(NO,225)   E(I),DELTA(I),Y(I),O(I)
        SAFES(I)=O(I)
        OUL(I)=O(I)+E(I)
        COL(I)=O(I)+Y(I)
        OP(I)=O(I)+A(I)
        COLO(I)=OP(I)
  417   CONTINUE
        TRCI=0
        SSUM=0
        MNREP=0
        DO  402  I=1,N
        MNREP=AMAX1(MNREP,D(I)/EOQ(I))
        SSUM=SSUM+(D(I)*K2(I)/EOQ(I))+((EOQ(I)/2)+SAFES(I)-MDLT(I)).H(
        RCI(I)=   K3(I)*D(I)/E(I)+E(I)*H(I)/2+O(I)*H(I)-MDLT(I)*H(I)
        TRCI=TRCI+RCI(I)
        WRITE(NO,195) E(I),U,Y(I),OUL(I),COL(I),OP(I),RCI(I)
  402   CONTINUE
        MNREP=PIP(MNREP)
        MINTRC=K1*MNREP+SSUM
        MPS=(TRCI-MINTRC)/TRCI
        WRITE(NO,563)TRCI,MINTRC,MPS
```

```
      CALL        SIMSUB
      TRC=0
      DO  405  I=1,N
      G(I)=RSIM(I)-O(I)
      IF  ( G(I).LT.0 )  G(I)=0
      WRITE(NO,4847) G(I)
      RC(I)=  TRCFUN(P(I),K3(I),D(I),E(I),G(I),H(I),K2(I),O(I),MDL
      TRC=TRC+RC(I)
  405 CONTINUE
  502 CONTINUE
CC
CC    A  NEW  ITERATION  IN  THE  OPTIMIZATION  ALGORITHM  STARTS
CC
      NCYCLE=NCYCLE+1
      WRITE(NO,518) NCYCLE
      IF ( NCYCLE . GT . 10 )  GO  TO  515
CC
CC    THE  NEW  UPDATING  PARAMETERS
CC
      CALL        OSUB
      DO  407  I=1,N
      IF  ( NCYCLE.EQ.1  )  GO TO  562
      U=DERFUN(P(I),G(I),PO(I),GO(I),COL(I),COLO(I),H(I),D(I),K1,E
      IF( U.GT.0 )  Y(I)=Y(I)-DELTA(I)
      IF( U.LT.0 )  Y(I)=Y(I)+DELTA(I)
      IF  (  Y(I).LE.0  )  Y(I)=A(I)
      GO  TO  572
  562 Y(I)=Y(I)+DELTA(I)
  572 CONTINUE
      E(I)=EFUN(P(I),G(I),D(I),H(I),K3(I),K2(I))
      PO(I)=P(I)
      GO(I)=G(I)
      COLO(I)=COL(I)
CC
CC    THE  NEW  STOCK  CONTROL  PARAMETERS
CC
      OUL(I)=O(I)+E(I)
      COL(I)=O(I)+Y(I)
      OP(I)=O(I)+A(I)
      WRITE(NO,185) E(I),U,Y(I),OUL(I),COL(I),OP(I),RC(I)
  407 CONTINUE
      CALL        SIMSUB
      TRCO=TRC
      TRC=0
      DO  579  I=1,N
      G(I)=RSIM(I)-O(I)
      IF  ( G(I).LT.0 )  G(I)=0
      WRITE(NO,4847) G(I)
      RCO(I)=RC(I)
      RC(I)=  TRCFUN(P(I),K3(I),D(I),E(I),G(I),H(I),K2(I),O(I),MDL
```

```
      TRC=TRC+RC(I)
      WRITE(NO,547) I,RCO(I), RC(I),DELTA(I)
  579 CONTINUE
      DO  409  I=1,N
      IF  ( RC(I).LT.RCO(I) )  GO  TO  409
      DELTA(I)=DELTA(I)/2
      WRITE(NO,548) DELTA(I)
  409 CONTINUE
      WRITE(NO,195) MINTRC,TRCO,TRC,(DELTA(I),I=1,N)
      ACHS=(TRCI-TRC)/TRCI
      WRITE(NO,189) ACHS,MPS,ACCS
CC
CC    AN  ITERATION  IN  THE  OPTIMIZATION  ALGORITHM  ENDS  UP
CC
      IF  (  (ACHS/MPS).LT.ACCS)  GO  TO  502
  515 CONTINUE
      DO  411  I=1,N
        WRITE(NO,185) E(I),U,Y(I),OUL(I),COL(I),OP(I),RC(I)
  411 CONTINUE
      WRITE(NO,195) TRC
CC
CC    INPUT  AND  OUTPUT  FORMAT  SPECIFICATIONS  IN  PROGRAM  ' MAI
CC
  101 FORMAT(I10,F10.5)
  103 FORMAT(F10.0,2F10.2,F5.2,F5.4,I10)
  165 FORMAT(/I10,10X,5F10.1///)
  185 FORMAT(//'E=',F10.1,5X,'U=',F10.5,5X,'Y=',F10.1,5X,'OUL=',I10,5
     +COL=',I10,5X,'OP=',I10,5X,'RC=',F10.1/)
  189 FORMAT('    ACHS =',F10.7,'MPS =',F10.7,5X,'ACCS =',F10.7/)
  195 FORMAT( ' MINTRC=',F15.1,5X,'TRCO=',F15.1,5X,'TRC=',F15.1/10(F
     -,5X))
  225 FORMAT(10F10.1)
  518 FORMAT(5X, ' ITERATION  NO = ' ,I5/ )
  547 FORMAT(/5X,'I=',I5,5X,'RCO=',F10.2,5X,'RC=',F10.2,5X,'DELTA',F8
  548 FORMAT(5X,'NEW DELTA =',F8.2)
  563 FORMAT(5X,'TRCI=',F15.1,5X,'MINTRC',F25.1,5X,'MPS=',F10.7)
 1119 FORMAT(5X,'K1= ',I15/)
 4847 FORMAT(5X,'G',F10.2)
      STOP
      END

ANK 602 DBANK 101 COMMON

.OSUB
0/81-14:31(,0)
CC
CC
CC    SUBROUTINE  '' OSUB ''  SOLVES  THE  SERVICE  LEVEL  FUNCTION
CC    USING  '' REGULA  FALSI ''  METHOD  AND  DETERMINES  THE  O(I)
CC
CC
```

```
      SUBROUTINE   OSUB
      PARAMETER    N=6
      REAL         MDLT
      COMMON /CS1/ D(N),APRO(N),O(N),E(N)
      COMMON /CS2/ P(N),G(N)
      COMMON /CS3/ MDLT(N),SDDLT(N)
      COMMON       NO
      DATA          NITMAX,ACCN,ACCO / 30,0.000001,0.001  /
      DO  89  I=1,N
      DX=O(I)/10.
      C2=(1-APRO(I))**((E(I)-P(I)*G(I))/D(I))
      Y=C2
      WRITE(NO,209)  I,Y
      NIT=1
      X1=O(I)
      Y1   =   PNROF(X1,I)
      B1=Y1-C2
      IF  (  B1.GT.0. )  DX=-DX
      X2=X1+DX
   80 CONTINUE
      Y2   =   PNROF(X2,I)
      B2=Y2-C2
      IF  (  ( B1*B2 ).GT.0. )   GO  TO  85
   81 X3=((X2-X1)/(Y2-Y1))*(Y-Y1)+X1
      Y3   =   PNROF(X3,I)
      B3=Y3-C2
      IF  (  NIT.GE.NITMAX  )   GO  TO  87
      NIT=NIT+1
      IF  ( ABS(B3).LE.ACCN  )   GO  TO  888
      IF  (  ( B1*B3 ).GT.0. )   GO  TO  83
      IF  (  ( ABS(  (  X3-X1  ) / X3  ) ).LE.ACCO  )   GO  TO  888
      X2=X3
      Y2=Y3
      B2=B3
      GO  TO  81
   83 CONTINUE
      IF  (  ( ABS(  (  X3-X2  ) / X3  ) ).LE.ACCO  )   GO  TO  888
      X1=X3
      Y1=Y3
      B1=B3
      GO  TO  81
   85 CONTINUE
      X1=X2
      Y1=Y2
      B1=B2
      X2=X1+DX
      NIT=NIT+1
      IF  (  NIT.LT.NITMAX  )   GO  TO  80
      WRITE(NO,205)  I
   87 CONTINUE
```

```
      WRITE(NO,207)  I,NITMAX
      X3=X2
  888 O(I)=X3
      WRITE(NO,135)  I,O(I),NIT
   89 CONTINUE
  135 FORMAT ( ' I=',I5,' O=',F10.1,'NIT=',I5/////)
  205 FORMAT( //' SOLUTION  INTERVAL  FOR  O(',I3,')  HAS  NOT  B
     +REACHED  YET'//)
  207 FORMAT(// ' ACCURACY  OF  O(',I3,')  IS  NOT  SATISFIED  IN
     +'ITERATIONS  AND  LAST  VALUE  IS  TAKEN  AS  OPTIMUM')
  209 FORMAT( ' I=',I5,5X,'C2=',F10.6)
      RETURN
      END

9 IBANK 143 DBANK 49 COMMON
  .PNROF
05/30/81-14:31(,0)
CC
CC
CC    FUNCTION  '' PNROF ''  COMPUTES  THE  SERVICE  LEVEL  FUNCT
CC
CC
      FUNCTION      PNROF(XX,I)
      PARAMETER     N=6
      REAL    MDLT
      COMMON /CS2/ P(N),G(N)
      COMMON /CS3/ MDLT(N),SDDLT(N)
      COMMON        NO
      Z1=( XX-MDLT(I) ) / SDDLT(I)
      F1=CDF ( Z1 )
      Z2=( XX+G(I)-MDLT(I) ) / SDDLT(I)
      F2=CDF ( Z2 )
      PNROF=F1**(1-P(I))*F2**P(I)
      RETURN
      END

 IBANK 20 DBANK 25 COMMON
.CDF
05/30/81-14:31(,0)
CC
CC
CC    SUBROUTINE  '' CDF ''  DOES  NUMERICAL  INTEGRATION
CC    USING  '' SIMPSON'S  RULE ''
CC    IT  COMPUTES  NOT  RUNNING  OUT  PROBABILITIES
CC
CC
      FUNCTION  CDF(ZZ)
      COMMON        NO
      DATA NINT /10/
      F(X)=.39894228*EXP(-(X**2)/2)
```

```
      VV=ABS(ZZ)
      CDF=.5
      IF  (VV.EQ.0)  RETURN
      IF  (VV.GT.4)  GO  TO  901
      NHALF=NINT/2
      DV=VV/NINT
      CDF=0.
      DO  91  I=1,NHALF
      VL=2*(I-1)*DV
      VU=2*I*DV
      VM=(VL+VU)/2
      CDF=CDF+(DV/3)*(F(VL)+4.*F(VM)+F(VU))
   91 CONTINUE
  901 CONTINUE
      IF  ( ZZ.GT.0 )  CDF=.5+CDF
      IF  ( ZZ.LT.0 )  CDF=.5-CDF
      RETURN
      END

BANK 42 DBANK 1 COMMON
 .PIP
30/81-14:31(,0)
   CC
   CC
   CC
   CC     FUNCTION  '' PIP ''  DETERMINES  THE  MINIMUM  NUMBER  OF  ORI
   CC     IN  A  DEPENDENT  SYSTEM
   CC
   CC
      FUNCTION  PIP(P)
      IP=P
      PIP=IP
      IF((P-PIP).LE.0.1E-8) RETURN
      PIP=PIP+1
      RETURN
      END

ANK 7 DBANK
      .SIMSUB
30/81-14:31(,0)
   CC
   CC
   CC
   CC     SUBROUTINE  '' SIMSUB ''  SIMULATES  THE  INVENTORY  SYSTEM
   CC     WITH  CURRENT  INVENTORY  CONTROL  PARAMETERS  AND
   CC     IT  DETERMINES  P(I)'S  AND  G(I)'S
   CC
   CC
      SUBROUTINE  SIMSUB
      PARAMETER   N=6
```

```
      REAL           MDLT
      REAL           INV,NEXTDT,NEXTOT,LTIME,MTRAN,INVPOS
      INTEGER        OUL,COL,OP,FLAGO,FLAGJO,FLAGOR
      COMMON /CS2/ P(N),G(N)
      COMMON /CS3/ MDLT(N),SDDLT(N)
      COMMON /CS4/ OUL(N),COL(N),OP(N)
      COMMON /CS5/ MTRAN(N),SDTRAN(N),DRATE(N),LTIME,DURSIM,IR
      COMMON /CS6/ NORRAN
      COMMON /CS7/ OSIM(N),RSIM(N)
      COMMON         NO
      DIMENSION      NIR(N),NJRB(N),NJRNB(N),NTR(N)
      DIMENSION      INV(N),DTIME(N),ORTIME(N),OSIZE(N),IWMORT(N),
     +               SOO(N),SRR(N),INVPOS(N),DSIZE(N),OMAT(300,25)
CC
CC    INITIALIZATION OF VARIABLES AND SETTING STARTING CONDITI
CC
      DO 21 I=1,N
      NIR(I)=0
      NJRB(I)=0
      NJRNB(I)=0
      NTR(I)=0
      SOO(I)=0
      SRR(I)=0
      ORTIME(I)=10.
      OSIZE(I)=0.
      INV(I)=OUL(I)-MDLT(I)
      INVPOS(I)=OUL(I)
   21 CONTINUE
CC
CC    SETTING THE INITIAL SYSTEM INDICATORS
CC
      DO 14 KK=1,300
      DO 14 J=1,25
      OMAT(KK,J)=0
   14 CONTINUE
      NOROAD=0
      NOREC=0
      NORDER=0
      NEXTOT=100
      FLAGO=0
      FLAGJO=0
      FLAGOR=1
CC
CC    GENEATING THE DEMAND TIMES AND SIZES FOR ALL ITEMS
CC
      DO 23 I=1,N
      DTIME(I)=DTFUN(DRATE(I),IR)
   23 CONTINUE
      DO 24 I=1,N
   26 CONTINUE
```

```
      DSIZE(I)=DSFUN ( MTRAN(I),SDTRAN(I),IR )
      IF  (  DSIZE(I).LE.0  )  GO  TO  26
   24 CONTINUE
   25 CONTINUE
CC
CC    DETERMINING  THE  NEXT  DEMAND  TIME ( NEXTDT )AS  THE  MINI
CC    OF  ALL  DEMAND  ARRIVAL  TIMES  IN  THE  INVENTORY  SYSTEM
CC
      NEXTDT=DTIME(1)
      J1=1
      DO  27  I=2,N
      IF  (  NEXTDT.LT.DTIME(I)  )  GO  TO  27
      NEXTDT=DTIME(I)
      J1=I
   27 CONTINUE
   28 CONTINUE
CC
CC    DETERMINING  THE  NEXT  ORDER  RECEIPT  TIME ( NEXTOT )
CC    AS MINIMUM OF ALL ORDER  RETURN  TIMES IN THE INVENTORY  SYS
CC
      IF  (  NOROAD.NE.0  )  GO  TO  34
      NEXTOT=100
      GO  TO  35
   34 IF  (  FLAGOR.EQ.0  )  GO  TO  39
      NEXTOT=OMAT(NOREC+1,1)
   33 CONTINUE
      FLAGOR=0
   39 CONTINUE
CC
CC    DETERMINING  THE  NEXT  EVENT  TIME  AS  THE  MINIMUM  OF
CC    TWO  EVENT  TIMES  AND  SETTING THE SIMULATION  TIME TO THAT
CC
      IF  (  NEXTOT.GT.NEXTDT  )  GO  TO  35
   41 CONTINUE
CC
CC    NEXT  EVENT  IS  AN  ORDER  RETURN
CC    RECEIVE  THAT  ORDER  AND  ADD  THE  RECEIVED  AMOUNTS
CC    TO  THE  RESPECTIVE  INVENTORIES
CC
      CLOCK=NEXTOT
      IF  (  CLOCK.GT.DURSIM  )  GO  TO  44
      FLAGOR=1
      NOREC=NOREC+1
      NOROAD=NOROAD-1
      NORI=OMAT(NOREC,2)
      DO  42  L2=1,NORI
      J2=OMAT(NOREC,2*L2+1)
      INV(J2)=INV(J2)+OMAT(NOREC,2*L2+2)
   42 CONTINUE
      IF  (  NEXTOT.EQ.NEXTDT  )  GO  TO  35
```

```
      GO TO 28
   35 CONTINUE
CC
CC    NEXT EVENT IS A DEMAND ARRIVAL
CC    SUPPLY THE DEMAND AND DECREASE THE RESPECTIVE INVENT
CC    BY AMOUNTS DEMANDED
CC
      CLOCK=NEXTDT
      IF ( CLOCK.GT.DURSIM ) GO TO 44
      NIO=0
      FLAGO=0
      FLAGJO=0
      INV(J1)=INV(J1)-DSIZE(J1)
      INVPOS(J1)=INVPOS(J1)-DSIZE(J1)
      DTIME(J1)=DTFUN(DRATE(J1),IR)+CLOCK
   31 CONTINUE
      DSIZE(J1)=DSFUN(MTRAN(J1),SDTRAN(J1),IR)
      IF ( DSIZE(J1).LE.0 ) GO TO 31
CC
CC    REWIEW OF INVENTORY POSITIONS FOR CRITICAL LEVELS IS
CC
      IF ( INVPOS(J1).GT.OP(J1) ) GO TO 25
CC
CC    PLACEMENT OF AN ORDER
CC
      FLAGO=1
      NORDER=NORDER+1
      NIO=NIO+1
      NOROAD=NOROAD+1
      ORTIME(J1)=CLOCK+LTIME
      OSIZE(J1)=OUL(J1)-INVPOS(J1)
      SOO(J1)=SOO(J1)+INVPOS(J1)
      OMAT(NORDER,1)=CLOCK+LTIME
      OMAT(NORDER,2)=1
      OMAT(NORDER,3)=J1
      OMAT(NORDER,4)=OUL(J1)-INVPOS(J1)
      INVPOS(J1)=OUL(J1)
      NIR(J1)=NIR(J1)+1
   37 CONTINUE
      DO 38 I=1,N
      IF (I.EQ.J1.OR.INVPOS(I).GT.COL(I) ) GO TO 38
      FLAGJO=1
      NIO=NIO+1
      ORTIME(I)=CLOCK+LTIME
      OSIZE(I)=OUL(I)-INVPOS(I)
      SRR(I)=SRR(I)+INVPOS(I)
      OMAT(NORDER,2*NIO+1)=I
      OMAT(NORDER,2*NIO+2)=OUL(I)-INVPOS(I)
      INVPOS(I)=OUL(I)
      NJRNB(I)=NJRNB(I)+1
```

```
   38 CONTINUE
      OMAT(NORDER,2)=N10
      IF  (  FLAGJC.NE.1  )  GO  TO  25
      NIR(J1)=NIR(J1)-1
      NJRB(J1)=NJRB(J1)+1
      GO  TO  25
   44 CONTINUE
CC
CC    SIMULATION  PERIOD  ENDS  UP
CC    CALCULATION  OF  THE  NECESSARY  PARAMETERS
CC
      DO  48  I=1,N
      NTR(I)=NIR(I)+NJRB(I)+NJRNB(I)
      WRITE(NO,345)I,NIR(I),NJRB(I),NJRNB(I),NTR(I)
  345 FORMAT(5X,'I=',I5,5X,'NIR=',I5,5X,'NJRB=',I5,5X,'NJRNBI=',I5,I(
      RSIM(I)=SRR(I)/NJRNB(I)
      OSIM(I)=SOO(I)/(NIR(I)+NJRB(I))
      P(I)=  FLOAT(NJRNB(I)) /NTR(I)
   53 CONTINUE
      WRITE(NO,335) I,RSIM(I),OSIM(I),P(I),G(I)
  335 FORMAT(5X,'I=',I3,5X,'RSIM=',F10.2,5X,'OSIM=',F10.2,5X,'P=',F1(
     +5X,'G=',F10.2)
   48 CONTINUE
      RETURN
      END

ANK 7717 DBANK 77 COMMON
    .DTFUN
0/81-14:32(,0)
CC
CC
CC    FUNCTION  '' DTFUN ''  GENERATES  EXPONENTIONALLY  DISTRIBUTED
CC    RANDOM  VARIABLES  AS  '' TIME  TILL  NEXT  DEMAND ''  FOR  IT
CC
CC
      FUNCTION      DTFUN(DDRATE,IR)
      R=RANDU(IR,JR)
      DTFUN=-(1/DDRATE)*ALOG(R)
      RETURN
      END

NK 14 DBANK
    .RANDU
0/81-14:32(,0)
CC
CC
CC    FUNCTION  '' RANDU ''  GENERATES  UNIFORMLY  DISTRIBUTED  RAND
CC    NUMBERS  BETWEEN  (0-1)
CC
CC
```

```
      FUNCTION      RANDU(IR,JR)
      JR=IR*03125
      IF  (  JR.LT.0  )  JR=JR+2*(2**34-1)+2
      FJR=JR
      RANDU=FJR/2.0**20
      RANDU=RANDU/2.0**15
      IR=JR
      RETURN
      END

 18 DBANK
   .DSFUN
31-14:32(,0)
:C
:C
:C    FUNCTION  '' DSFUN ''  GENERATES  NORMALLY  DISTRIBUTED  RANDOM
:C    VARIABLES  AS  '' DEMAND  SIZE ''  FOR  AN  ITEM
:C
:C
      FUNCTION      DSFUN(DMEAN,STDV,IR)
      COMMON /C56/ NORRAN
      SUMR=0.0
      DO  701  IN=1,NORRAN
      SUMR=SUMR+ RANDU(IR,JR)
  701 CONTINUE
      DSFUN=DMEAN+STDV*((SUMR-NORRAN/2.)/SQRT(NORRAN/12.))
      RETURN
      END

 19 DBANK  1  COMMON
```