AN ABSTRACT OF THE THESIS OF

Sang-Jin Nam for the degree of Master of Science in Industrial and Manufacturing Engineering presented on June 5, 1991.

Title: Grid Search Based Production Switching Heuristic For Aggregate Production Planning

Abstract Approved: Redacted for Privacy

Rasaratnam Logendran

The Production Switching Heuristic (PSH) developed by Mellichamp and Love (1978) has been suggested as a more realistic, practical and intuitively appealing approach to aggregate production planning (APP). In this research, PSH has been modified to present a more sophisticated open grid search procedure for solving the APP problem. The effectiveness of this approach has been demonstrated by determining a better near-optimal solution to the classic paint factory problem using a personal computer based application written in THINK PASCAL. The performance of the modified production switching heuristic is then compared in the context of the paint factory problem with results obtained by other prominent APP models including LDR, PPP, and PSH to conclude that the modified PSH offers a better minimum cost solution than the original PSH model.

Grid Search Based Production Switching Heuristic For Aggregate Production Planning

by

Sang-Jin Nam

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Completed June 5, 1991

Commencement June 1992

APPROVED:

Redacted for Privacy

Dr. Rasaratnam Logendran, Assistant Professor of Industrial and Manufacturing Engineering, in charge of major

Redacted for Privacy

Dr. Tom West, Department Head, Industrial and Manufacturing Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis presented June 5, 1991

Typed by Sang-Jin Nam for Sang-Jin Nam

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Grid Search Based Production Switching Heuristic
for Aggregate Production Planning

INTRODUCTION

1. PROBLEM ENVIRONMENT

Aggregate production planning (APP) involves the simultaneous determination of a company's production, inventory and employment levels over a finite time horizon. Its objective is to minimize the total relevant costs while meeting non-constant, time varying demand, assuming fixed sales and production capacity [Silver 1967]. Since the early 1950's, approaches for APP have varied from simplistic, graphical methods to more sophisticated optimizing, search, parametric, and dynamic methods. These fall into two broad categories - those which guarantee a mathematically optimal solution with respect to the model and those that do not. Within each of these categories are numerous alternative approaches, resulting in an abundance of theoretical solution procedures.

Despite all the approaches available to managers, the impact of APP methods on industry operating practices has been insignificant. Several reasons are cited for the lack of assimilation of aggregate planning techniques into management practice [Mellichamp and Love 1978].

Foremost among these is that the optimal solution models in APP - linear programming (LP) [Hanssmann and Hess 1960], goal programming (GP) [Goodman 1974, Lee and Moore

1974], transportation techniques (TPT) [Bowman 1956], and the linear decision rule (LDR) [Holt et al. 1960] - all incorporate various simplifying assumptions which limit their applicability. As an example, cost functions associated with mathematical programming approaches (LP, GP, TPT) are all required to be linear. If non-linear cost functions are used instead, piecewise linear approximations may be employed to convert them into suitable linear forms. However, the additional complexity required to perform these conversions does not justify wide application of the various models involving non-linear cost functions.

As a second example, the LDR approach, which has become a standard for comparison, utilizes quadratic cost functions for all components of costs. In actual industry situations, however, some costs are non-linear. None of the optimal approaches allow for mixed costs.

Another troublesome simplification involves the way in which demand is treated in the mathematical programming approaches. All these methods incorporate the assumptions that demand forecasts both are accurate and equally weighted over the planning horizon. The result is that the production level for the forthcoming period can be significantly affected by forecasts for future periods even though forecasts for distant periods are less reliable than forecasts for the immediate future [McGarrah 1983].

Near-optimal approaches, including Search decision rule (SDR) [Taubert 1968], Management coefficient model (MCM) [Bowman 1963], Parametric production planning (PPP) [Jones 1967], overcome some of the problems associated with optimal approaches. Complex cost functions which accurately describe actual costs may be embodied in most near-optimal models. An analysis of the impact of forecast errors on strategy development may also be performed by incorporating stochastic demand characteristics in near-optimal models [Mellichamp and Love 1978]. Despite these improvements, however, these models suffer from a limitation that also applies to optimal models. That is, most of these models produces a different set of values for the decision variables - production rate (P_t), work force level (W_t), and inventory level (I_t)- for each period in the planning horizon. This probably is the single most important factor that has contributed to limiting the application of all aggregate planning models. In other words, a majority of APP approaches incorporate continuous decision variables that require frequent adjustments to both production and work force settings to achieve a minimum cost solution.

A large set of decision variable values which frequently adjust the production and workforce level on a planning period by period basis has been observed as being inconsistent with management practices in industry. The Production Switching Heuristic [Mellichamp and Love 1978] was developed to address this inconsistency with the belief, thus, of having more appeal to practicing managers.

This heuristic is based on the observation that managers seem to favor one large change in work force over a series of smaller and more frequent changes over the planning horizon. Thus, as long as demand is being met, i.e., stockouts do not occur too frequently and inventory levels do not increase drastically, managers are often inclined to maintain the same production and work force levels, making minor adjustments when necessary.

Furthermore, a policy that requires frequent hiring and firing of personnel might be impractical because of prior contract agreements, or undesirable due to the potential negative effects on the firm's public image [Nahmias 1989]. If production is confined to a relatively small number of prescribed levels (so that adjustment in production is achieved by given discrete steps), experience of performance and scheduled activities at each level provide good opportunities for controlling costs and minimizing the effects of change [Eilon 1975].

From these and other arguments Mellichamp and Love reasoned that an aggregate production planning methodology which utilizes near-optimal solution techniques to select a small number of decision variable values that are efficient over most levels of demand would have much potential for industry applications. Interestingly, the basic theory of such a method had been previously developed by Orr (1962). It was this method which was dressed up to become the Production Switching Heuristic (PSH) and applied to a limited set of production problems by Mellichamp and Love [Mellichamp and Love 1978].

2. RESEARCH OBJECTIVES

When production operations are carried out at certain predetermined levels (analogous to opening or shutting of a production line), it is not appropriate to treat the level of production as a continuous variable. Mellichamp and Love (1978) described a modified random walk production-inventory heuristic for three production levels which they felt should appeal to managers on the basis of simplicity as well as efficiency. This approach is directed to situations where three-production levels (high, normal, and low) can only be changed in discrete increments or decrements, such as adding or removing a production shift.

In their approach they also described switching algorithms for desirable fixed production levels by analyzing alternative values of various control parameters which provided a set of production, work force, and inventory decisions which were directly related to cost performance over a planning horizon. The problem, therefore, was to find the best set of the control parameters. The Production switching heuristic of Mellichamp and Love, however, limited grid search options in analyzing all sets of control parameters - in effect hindering their approach from determining a better solution.

In this research, the production switching heuristic by Mellichamp and Love (1978) is modified by using a more elaborate grid search method, which exhausts reasonable incremental values over the entire cost surface. This search method widely opens all grid options to evaluate a broader set of alternative parameters than the original PSH approach.

Two different schemes have been proposed as options of the grid search with this alternative approach for evaluating the productivity function used in PSH and then to determine the optimum combination between production and work force sizes. The productivity function, developed in PSH, has been modified in the proposed approach to provide for a better balance between regular work force and overtime rates than that in PSH. Furthermore, it has been demonstrated that the modified productivity function yields better results for reducing overweight regular payroll costs.

To evaluate and validate the modified approach offered, the paint factory problem first described by Modigliani et al. (1955) has been used. Based on the two search schemes, which are labeled as MPSH1, and MPSH2, the paint factory problem is solved using THINK PASCAL software on an IBM PC compatible computer. The results are compared with those obtained from the Parametric Production Planning (PPP), PSH, and Linear Decision Rule (LDR)- (i.e both other near-optimal and optimal solutions) reported in Mellichamp and Love to demonstrate a better performance of the modified PSH.

LITERATURE REVIEW

A review of literature on APP will reveal a number of important issues. These issues are described in the sections that follow. They include: general background about APP and its role in production planning and operation; the common APP strategies used by practitioners and the costs relevant to those strategies; the various problems with various APP approaches; general APP methodology and its classification; and the significant historical highlights of the more notable APP models developed since 1950.

1. BACKGROUND

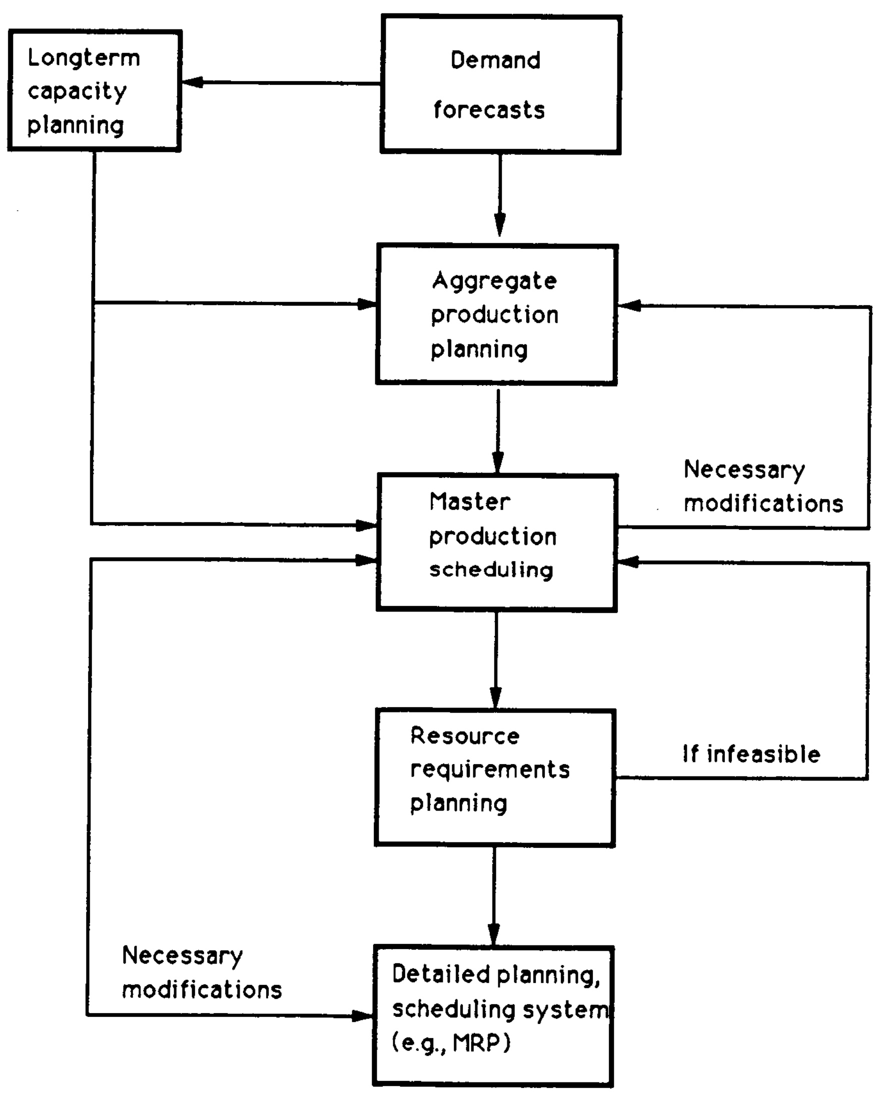
Production planning is concerned with the determination of production, inventory, and work force levels to meet fluctuating demand requirements. Normally, the physical resources of the firm are assumed to be fixed during the planning horizon of interest and the planning effort is oriented toward the best utilization of those resources, given the external demand requirements. A problem usually arises because the times and quantities imposed by the demand requirements seldom coincide with the times and quantities which result in the efficient use of the firm's resources. Whenever the conditions affecting the production process are not stable in time (due to changes in demand, components of costs, or capacity availability), production should be planned on an aggregate level to ensure the most efficient utilization of resources. The time horizon (commonly 6 to 12 months) of this planning activity is dictated by the nature of the dynamic variations such as seasonalities.

Since it is usually impossible to consider every fine detail associated with the production process while maintaining such a long planning horizon, it is mandatory to aggregate the information being processed. This aggregation can take place by consolidating similar items into product groups, different machines into machine centers, different labor skills into labor centers, and individual customers into market regions. The type of aggregation to be performed is suggested by the nature of the planning systems to be used, and the technical as well as managerial characteristics of the production activities. Aggregation forces the use of a consistent set of measurement units. It is common to express aggregate demand in production hours [Hax and Candea 1984].

Once the aggregate plan is generated, constraints are imposed on the detailed production scheduling process which decide the specific quantities to be produced of each individual item. These constraints normally specify production rates or total amounts to be produced per month for a given product family. In addition, crew sizes, levels of machine utilization, and amounts of overtime to be used are determined.

The output of the aggregate production planning process is a master schedule for final assembly/production. Figure 1 shows a schematic diagram of the aggregate planning function and its place in the hierarchy of production planning decisions [Vollmann et al. 1988].

Figure 1. A Schematic Diagram of a Sequential Production Planning Process



2. STRATEGIES AND COSTS IN APP

In general, APP can take either one or a combination of several pure strategies in responding to fluctuating demand [Buffa and Taubert 1972]:

1. Management can change the size of the work force by hiring and laying off, thus allowing changes in the production rate to take place. Excessive use of these practices, however, can create severe labor problems.
2. While maintaining a uniform regular work force, management can vary the production rate by introducing overtime and/or idle time.
3. While maintaining a uniform production rate, management can anticipate future demand by accumulating seasonal inventories. The tradeoff between the cost incurred in changing production rates and holding seasonal inventories is the basic question to be resolved in most practical situations.
4. Management can also resort to planned backlogs whenever customers may accept delays in filling their orders.
5. Additionally, Management may have the opportunity to use subcontracting as a suitable alternative to a part of production.

As with most of the optimization problems considered in production management, the goal of the analysis is to choose the aggregate plan that minimizes cost. It is important to identify and measure those specific costs that are affected by the planning decision [Bedworth and Bailey 1987].

1. Smoothing costs. - Smoothing costs refer to those costs that accrue as a result of changing the production levels from one period to the next. In the aggregate planning context, the most salient smoothing cost is the cost of changing the size of the work force. Increasing the size of the work force requires time and expense to advertise positions, interview prospective employees, and train new hires. Decreasing the size of the work force means that workers must be laid off. Severance pay is one cost of decreasing the size of the work force. Other costs associated with decreasing the work force size which are harder to measure are (1) the costs of a decline in worker morale that may result and (2) the potential for decreasing the size of the labor pool in the future, as workers who are laid off acquire jobs with other firms or in other industries.

2. Holding costs. - Holding costs are the costs that accrue as a result of having capital tied up in inventory. If the firm can decrease its inventory, the money saved could be invested elsewhere with a return that will vary with the specific company. These costs are usually charged against the inventory remaining on hand at the end of the planning period.

3. Shortage costs. - Holding costs are charged against the aggregate inventory as long as it is positive. In some situations it may be necessary to incur shortages, which are represented by a negative level of inventory. Shortages can occur when forecasted demand exceeds the capacity of the production facility or when demands are higher than anticipated. For the purposes of aggregate planning, it is generally assumed that excess demand is backlogged and filled in a future period. In a highly competitive situation, however, it is possible that excess demand is lost and the customer goes elsewhere. This case, which is known as lost sales, is more appropriate in the management of single item and is more common in retailing than in a manufacturing context.

4. Regular time costs. - These costs involve the cost of producing one unit of output during regular working hours. Included in this category are the actual payroll costs of regular employees working on regular time, the direct and indirect costs of materials, and other manufacturing expenses. When all production is carried out on regular time, regular payroll costs become a "sunk cost,' since the number of units produced must equal the number of units demanded over any planning horizon of sufficient length. If there is no overtime or worker idle time, regular payroll costs do not have to be included in the evaluation of different strategies.

5. Overtime and subcontracting costs. - Overtime and subcontracting costs are the costs of production of units not produced on regular time. Overtime refers to production by regular time employees beyond the normal work day, while subcontracting refers to the production of items by an outside supplier.

When planning is done at a relatively high level of the firm, the effects of intangible factors are more pronounced. Any solution to the aggregate planning problem obtained from a cost-based model must be considered carefully in the context of company policy.

3. PROBLEMS IN APP

For the high-volume standardized product system and for the closed job shop system (a shop not open to job order outside the enterprise) the concepts and methods of aggregate planning and scheduling are of particularly great importance. They are important if managers are to obtain the best possible use of facilities within the constraints of policies regarding hiring and layoff, inventories, the use of outside capacity (subcontracting) and internal capacity. Indeed, the process of aggregate planning yields a range of alternative capacity utilizations for management's consideration. In employing the term aggregate planning, we include scheduling, and as used here the term schedule means a production program. The economic significance of aggregate planning and scheduling is by no means minor, for we are confronted with broad, basic questions such as the following: To what extent should inventory be used to absorb these fluctuation in demand that will occur over the next six to 12 months? Why not absorb the fluctuations simply by varying the size of the work force? Hire and fire as demand increases or decreases. Why not maintain a fairly stable work force size and absorb fluctuations through changing production rates by resorting to overtime or shorter hours? Why not maintain a fairly stable work force size and production rate and let subcontractors wrestle with the problems of fluctuating order rates? Should we purposely not meet all demands? In most instances there would not be a single pure strategy that would be applicable but rather a combination of the various strategies. There are costs associated with each strategy, so what we seek is an astute selection of a combination of the alternatives.

If we use inventories to absorb seasonal changes in demand, capital and obsolescence costs as well as the costs associated with storage, insurance, and handling will tend to increase. Besides seasonal factors, the use of inventories to absorb short-term fluctuations will incur increases in the same costs compared to some ideal or minimum inventory level necessary to maintain the production process. When inventories fall below this ideal or minimum level, stock-out costs will increase and all of the costs associated with short runs will increase. Changes in the size of the work force affect the total cost of labor turnover. When new workers are hired, there are costs of selection, training, and lower production effectiveness. The termination of workers may involve unemployment compensation or other termination costs as well as an intangible effect on public relations and public image. If changes in the size of the work force are large, it may mean adding or subtracting an entire shift. The incremental costs involved here are shift premiums, incremental supervision and other overheads. If we absorb fluctuations through changes in the production rate, we will absorb overtime premium costs for increases and probably idle labor costs (higher average labor costs per unit) for decreases. Usually managers try to maintain the same average labor costs by reducing hours worked below normal levels to some extent. Where undertime schedules persist, labor turnover and the costs attendant to it are likely to increase. Many of the costs affected by aggregate planning and scheduling decisions are difficult to measure and are not segregated in accounting records. Some are alternative costs of opportunity, such as interest costs on inventory investment; some cost are not measurable, such as those associated with public relations and public image. However, all of the costs are real and have a bearing on aggregate planning decisions [Groff and Muth 1972].

4. APP METHODOLOGY

Models have played an important role in supporting management decisions in aggregate production planning. Anshen et al. (1958) indicate that models are of great value in helping managers to :

1. Quantify and use the intangibles which are always present in the background of their thinking but which are incorporated only vaguely and sporadically in scheduling decisions.
2. Make routine the comprehensive consideration of all factors relevant to scheduling decisions, thereby inhibiting judgments based on incomplete, obvious, or easily handled criteria.
3. Fit each scheduling decision into its appropriate place in the historical series of decisions and, through the feed back mechanism incorporated in the decision rule, automatically correct for prior forecasting errors.
4. Free themselves from routine decision-making activities, thereby giving them greater freedom and opportunity for dealing with extraordinary situations.

Research literature on APP since 1950 reflects various graphical, mathematical, and heuristic techniques designed to be used to generally implement those specific APP strategies and related cost function. In general, the more adaptable the technique is to all of the strategies listed above, the more robust it is. Furthermore, the more limiting the data assumptions to implement these techniques have been, generally, the more apt the technique is to provide an exact mathematical answer for the APP planner.

At the very broadest level of categorization of the various techniques reported on, two classifications of techniques exist [Silver 1972]. The first classification includes techniques that produce an exact, mathematically optimal solution, while the second includes those that do not. Within this framework, all of the various techniques can be placed and an evolution traced over the years starting with the very simple linear mathematical models and graphical techniques to the present day sophisticated multiple objective goal programming models and search and heuristic approaches. Table 1 shows the classification and a selection of prominent aggregate planning approaches.

The mathematically optimal approaches to APP are by far the greatest in number, and they can cater for a greater number of decision variables than the near-optimal approaches. However, their use in aggregate production planning involves the planner in a dilemma, since although they can obtain optimum results, their ability to model actual problems realistically is still limited.

The near-optimal approaches can more readily handle uncertainty and non-linearities. They can better describe the aggregate planning problem and, when used in conjunction with a computer, they have the potential to consider a wider range of planning variables.

Table 1. A Classification of APP Selected Methods and Models

Classification	Type of Model	Developer(s)
I. Optimal	(a) Linear - Transportation Mathematical L.P	Bowman (1956) Hanssman and Hess (1960)
	(b) Linear Decision Rule	Holt et al. (1960)
	(c) Lot size model	Wagner and whitin (1958) Manne (1958),
	(d) Goal programming	Goodman (1974) Lee and Moore (1974)
II. Near-optimal	(e) Management Coefficient Model	Bowman (1963)
	(f) Search Decision Rule	Taubert (1968)
	(g) Parametric Production Planning	Jones (1967)
	(h) Production Switching Heuristics	Mellichamp and Love (1978)
	(i) Simulation	Silver (1966) Lee and Khumawala (1974) Eilon (1975)

5. HISTORICAL NOTES ON APP

The aggregate production planning problem was conceived in an important series of papers which appeared in the mid 1950's. The first, by Holt, Modigliani, and Simon (1955), discussed the structure of the problem and introduced the quadratic cost approach, while a later study by Holt, Modigliani, and Muth (1956) concentrated on the computational aspects of the model. A complete description of the method which is called the Linear Decision Rule (LDR) and its application to production planning for a paint company is presented in Holt, Modigliani, Muth, and Simon (1960).

Bowman (1956) discussed the use of a transportation model for production planning. The advanced linear programming formulation was due to Manne (1958) who conceived of an innovative approach of incorporating setup cost into a linear program - classified as the Lot Size Model in Table 1. Dzielinski and Gomory (1965) treated computational issues concerning the Lot Size Model. Lasdon and Terjung (1971) considered a number of further computational refinements.

This particular linear programming formulation of the aggregate planning problem is essentially the same as the one developed by Hansmann and Hess (1960). Other linear programming formulations of the production planning problem generally involve multiple products or more complex cost structures [Newson 1975a, 1975b].

More recent work on the aggregate planning problem has focused on aggregation and disaggregation issues [Axsater 1981, Bitran and Hax 1977, and Zoller 1971], the incorporation of learning curves into linear decision rules [Ebert 1976], extensions to allow for multiple products [Bergstrom and Smith 1970], and inclusion of marketing and/or financial variables [Damon and Schramm 1972, and Leitch 1974].

The limitations of the linear and quadratic forms have encouraged management scientists to investigate other models. Some heuristic procedures have been applied to more complex models. Jones (1967), for example, has suggested a heuristic procedure in which the form of the decision rule is hypothesized and the parameters of the rule determined by simulation of the cost model. This procedure enables flexibility of modeling, and promising computational results have been obtained. This method necessitates the prior determination of the mathematical form of the decision rule. In addition, a simulation must be performed in order to determine values for the decision rule parameters.

Vergin (1966) has proposed a simulation approach in order to achieve maximum realism in modeling. The disadvantage of the simulation approach is that it does not offer any specific means for finding an optimal or nearly optimal solution. In an attempt to strike a good balance between realism and solvability, Taubert (1968) has applied several search techniques to higher-than-second-degree models having more than two decision variables. Encouraging computational results were obtained. These methods are limited by the size of the problem, and results are dependent on particular parameter settings of the search technique.

Bowman (1963) advocated a procedure for modeling management decision making with an illustration in the area of production smoothing and work force balancing.

PROBLEM STATEMENT

1. PRODUCTION SWITCHING HEURISTIC (PSH)

Orr (1962) has suggested that certain production-inventory problems can be treated with random walk inventory policies. Based on his work, the approach of Elmaleh and Eilon (1974) assumed that production can only be carried out at discrete levels, which would be the case if certain facilities could be either running or shut. As such, this approach can resemble the decision making process in a wide variety of circumstances more closely than other techniques.

The approach operates by setting control levels on the inventory whereby if the inventory passes a control level then a change in production rate is triggered. The cost parameters used for the determination of the control levels and production rate are purely those directly related to changes in production rate, i.e. a fixed cost per change in production rate and a cost proportional to the magnitude of the change. Their results, based on these parameters, show that solutions can be obtained which are better than those obtained using a simple inventory control model.

The disadvantages of Elmaleh and Eilon's approach are that it cannot easily be applied to a multi-product batch manufacturing system (as it is essentially a single-product or aggregate model) and that it does not accommodate the planning of a variety of products within overall capacity constraints. Furthermore, the method does not implicitly use a forecast of demand. Consequently, since there is no mechanism for producing more during slack periods in anticipation of demand overall production rates can vary dramatically from period to period [O'Grady and Byrne 1986].

Mellichamp and Love (1978) assume that demand can be forecasted accurately and is equally weighted over the planning horizon. This is a simplistic view. They developed an approach which allowed the use of a company's cost structure, which could therefore be made as close as possible to the real system costs. The results obtained showed very small cost penalties as compared with the optimum LDR method [Holt et al. 1960]. Their results have since been criticized [Vergin 1980] on the basis that their cost comparisons were based upon total costs, and if only the controllable costs (variable costs - i.e. overtime, hiring and firing costs and inventory costs) are considered then the cost penalties are much greater.

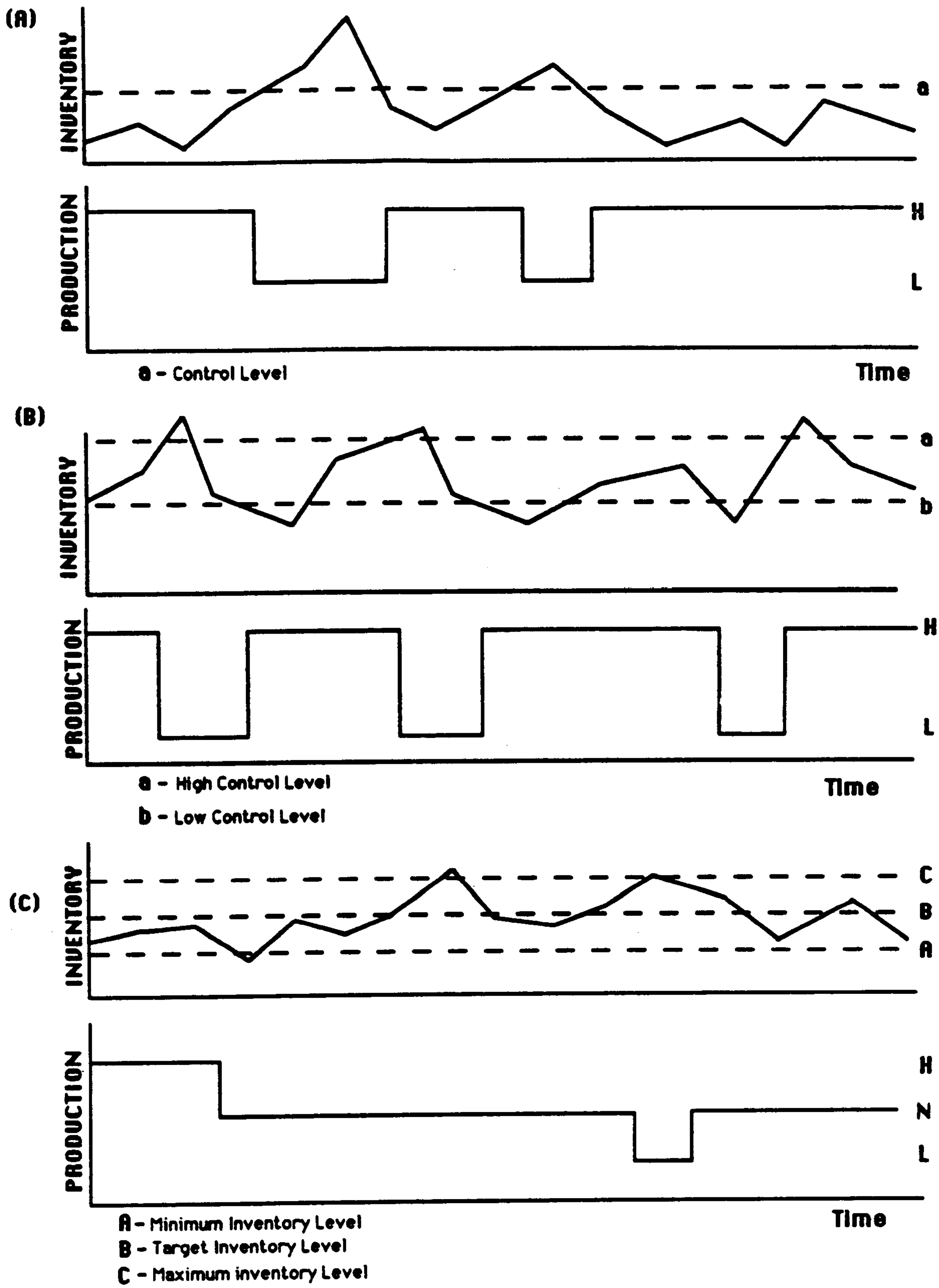
Oliff and Burch (1985) used PSH for Owens-Corning Fiberglas, a manufacturer of glass fiber products, to determine aggregate inventory levels, production and work force levels. Recently, Oliff and Leong (1987) and Oliff, Lewis and Markland (1989) developed a discrete production switching rule for crew-loaded facilities.

2. SWITCHING ALGORITHM

The production switching algorithm accomplishes the assignment of production levels to each planning period using a reasonable control mechanism. This control mechanism is illustrated by the simple two-production-level case, presented in Fig.2-(A) : the levels **H** and **L** represent the high and low production rates at which the system can operate. The inventory level is monitored and when it crosses a control level **a** from below, the production is switched from **H** to **L** and vice versa. A more elaborate control mechanism would involve two control levels **a** and **b** (where **a** > **b**) and switching from **L** to **H** will take place when the inventory level crosses the control limit **a** from below (see Figure. 2-(B)). The rationale for such switching policies is similar to the two-bin or (s,S) inventory control system. The random walk approach to APP proposed by Orr (1962) and adapted by Elmaleh and Eilon (1974), is formulated using three-production-levels as follows: specify three inventory levels, **a** > **b** > **c**, and three production levels, **H** > **N** > **L**, with the operating instructions:

$$P_t = \begin{cases} \mathbf{H} & \text{if } I_{t-1} \text{ passes } \mathbf{c} \text{ from above,} \\ \mathbf{N} & \text{if } I_{t-1} \text{ passes } \mathbf{b}, \\ \mathbf{L} & \text{if } I_{t-1} \text{ passes } \mathbf{a} \text{ from below.} \end{cases} \qquad (1)$$

Figure 2. Switching Mechanisms



Values for **a, b, c** and **H, N, L** are obtained by simulating various combinations of these control parameters over a historical demand series and choosing the set for which costs are minimum.

We propose incorporating F_t, the demand forecast for period t, in the rule for determining P_t as follows. The estimated closing inventory $\boldsymbol{I}_t$ in period t is:

$$\boldsymbol{I}_t = I_{t-1} + P_t - F_t \qquad (2)$$

Since we are attempting to control both production and inventory costs, we can replace $\boldsymbol{I}_t$ by **B** where **B** represents a target inventory level to be determined. Rearranging equation (2) such that the input variables I_{t-1} and F_t are on the left side of the equation and the decision variables P_t and **B** are on the right yields:

$$F_t - I_{t-1} = P_t - \mathbf{B} \qquad (2')$$

The left side of the equation represents the amount of anticipated demand in period t which cannot be met with on-hand inventory, while the right side of the equation reflects the production from period t available to meet demand after satisfying the target inventory requirement. Finally, the rule for P_t is as follows:

$$P_t = \begin{cases} \mathbf{L} & \text{if } F_t - I_{t-1} < \mathbf{L} - \mathbf{C} \\ \mathbf{H} & \text{if } F_t - I_{t-1} > \mathbf{H} - \mathbf{A} \\ \mathbf{N} & \text{otherwise} \end{cases} \qquad (3)$$

Where F_t = forecasted demand for period t, I_{t-1} = ending inventory for period t-1, **A** = Minimum acceptable target inventory, **C** = Maximum acceptable target inventory, and **B** = target inventory level (= (**C** + **A**)/2).

The heuristic suggests that if the net production required after taking into account on-hand and target inventories is less than the low level of production, produce at the low level. If the net production required is greater than the high level of production, produce at the high level. Finally, if required net production is between the low and high levels of production, produce at the normal level (see Figure. 2-(C)).

A general production switching rule that allows both overtime and direct application to discrete operations is presented as following:

For some inventory target **A** < **C** and *n* discrete production levels $R_1 > R_2 > R_3 \ldots\ldots > R_n$, choose the production rate as

$$P_t = \begin{array}{ll} R_1 & \text{if} \quad F_t - I_{t-1} + \mathbf{A} > R_1 \\ R_2 & \text{if } R_1 > F_t - I_{t-1} + \mathbf{A} > R_2 \\ \vdots & \\ R_k & \text{if } R_{k-1} > F_t - I_{t-1} + \mathbf{A} > R_k \\ R_{k+2} & \text{if } R_{k+3} > F_t - I_{t-1} + \mathbf{C} > R_{k+2} \\ \cdot & \\ R_n & \text{if} \quad F_t - I_{t-1} + \mathbf{C} > R_n \\ R_{k+1} & \text{otherwise} \end{array} \tag{4}$$

The rule suggests that if net production required after accounting for on-hand inventory and target inventories exceed the highest production level (R_1), produce at the highest level. If net production required is less than the lowest production level(R_n), produce at the lowest level. The remaining switches are interpreted similarly.

It is straightforward now to include overtime in the model by doubling the number of production levels to 2n, where

$$OT_1 > R_1 > OT_2 > R_2 > \ldots OT_n > R_n,$$

corresponding to n discrete work force levels

$$W_1 > W_2 > \ldots > W_n$$

where

R_i = regular time production rate i

OT_i = cumulative regular and overtime production rate i

W_i = work force level for production rates R_i and OT_i

following the general form of equation (4). The model requires estimation of payroll (regular) cost (PC), overtime costs (OC), hiring (HC) and firing (FC) cost, and inventory costs (IC).

Total costs then are

$$TC = PC + OC + HC + FC + IC. \tag{5}$$

Although the general model could incorporate back orders with minor modifications, a manufacturer seldom would stock out his entire aggregate product line. In the context of a typical aggregate planning problem, the objective is to minimize these total costs.

Generally speaking, the PSH, based on the existing rationale, may be extended to incorporate more than three levels. In other words, more than three different pairs of production and work force sizes may be used to smooth the production over the entire planning horizon. With a greater number of levels, the PSH is expected to perform better since its ability to meet fluctuations in demand increases. On the other hand, as the number of levels increases, the frequency of switching and, therefore, the complexity of the production system also increases. The very advantage of a switching heuristic - less frequent rescheduling of production and work force, is lost gradually as more levels are added. This research is, therefore, directed to situations where the three-production levels (high, normal, and low) can only be changed in discrete increments or decrements, such as adding or removing a production shift.

3. GRID SEARCH PROCEDURE

The problem is to determine values for the control parameters (for the three-production rate) **H**, **N**, **L** and **A**, **B**, **C** which generate a set of production, work force, and inventory decisions (P_t , W_t , and I_t) that will be cost efficient over the planning horizon. Once appropriate values have been determined, planning decisions for each period are made using the above rule.

The procedure used in this research for selecting values for the control parameters of the heuristic is an iterative simulation approach utilizing a historical demand series and includes the following steps:

Step **1**. Obtain input data ;

1. Forecasted demand $\{F_t\}$
2. Initial values: P_0, W_0 , and I_0.
3. Productivity function : $\mathbf{W_t = f(P_t\ ,\ G)}$ where G is the % increase or decrease in the work force required to achieve a high or low level of production.
4. Cost functions : cost component, C_{it} for i=1, 2,...,n.

Step **2**. Specify various values of control parameters that are to be evaluated. Any one of several different search options including various grid and gradient procedures may be used.

Step **3**. Initialize control parameters (**N** and **B**).

Step **4**. Assign the production levels (P_t) for each period using the switching algorithm, then calculate I_t using equation (2), and W_t using productivity function.

Step **5**. Calculate cost component C_{it} and total cost(TC) from equation (5).

Step **6**. Repeat Steps **4** and **5** for all combinations of the control parameters **H**, **N**, **L** and **A**, **B**, **C**.

Step **7**. Select those values for **H**, **N**, **L** and **A**, **B**, **C** for which TC is minimum.

A computer search for the minimum TC was then conducted using an elaborate grid procedure in which the initial values, increments, and ranges were specified for **N**, **E**, **B**, and **D** in this research. Thereafter, using the relationships specified below:

H = **N** + **E**,

L = **N** - **E**,

A = **B** - **D**,

C = **B** + **D**,

all combinations of the control parameters (i.e. grid) were systematically searched to determine a minimum total cost.

For several reasons, the simple grid search procedure used in the PSH cannot precisely determine the location of the best possible solution by the heuristic. First, the normal production level, **N**, is preset to the average demand over the planning horizon. Second, the high production level, **H**, and the low production level, **L**, are required to be equally spaced about the normal production level, **N**. Finally, the low target inventory level, **A**, and the high target inventory level, **C** are again constrained to be equally spaced from the normal target inventory level, **B**.

The heuristic rules in PSH clearly define how the production sizes are to be selected for the periods given in the planning horizon. However, the corresponding decision with regard to the size of the work force to be used for each period in the planning horizon is not explicitly specified by the PSH (Mellichamp and Love) model.

Theoretically, for a given amount of production, P_t, an infinite number of possibilities exist for the corresponding work force size, W_t, using a combination of regular work force and overtime labor. Therefore, the effectiveness of the heuristic is dependent upon the functional relationship between these combinations, because it controls the costs associated with regular payroll, hiring/firing, and overtime.

Mellichamp and Love (1978) presented this issue by introducing a productivity function, with little or no explanation about its operation in determining an appropriate level of regular payroll versus overtime labor. Their productivity function was given as:

$$\mathbf{W_t = \mathit{f}(P_t\ ,\ G)}$$

where G was the percent increase or decrease in the work force required to achieve high or low levels of production.

In order to determine the optimum combination between the production and the work force sizes, the following two schemes are used in this research - each of these can run separately, labeled MPSH1 (Modified PSH1) for scheme 1, and MPSH2 (Modified PSH2) for scheme 2:

$$\text{Scheme 1}\quad W_t = \begin{cases} \mathbf{N}/k & \text{if } P_t = \mathbf{N} \\ (\mathbf{H}/k)*\mathbf{G} & \text{if } P_t = \mathbf{H} \\ (\mathbf{L}/k)*\mathbf{G} & \text{if } P_t = \mathbf{L} \end{cases} \tag{5}$$

$$\text{Scheme 2} \quad W_t = \mathbf{N}/k \qquad \text{if } P_t = \mathbf{N}$$
$$= \mathbf{N}/k + (E/k)*\mathbf{G} \quad \text{if } P_t = \mathbf{H}$$
$$= \mathbf{N}/k - (E/k)*\mathbf{G} \quad \text{if } P_t = \mathbf{L} \qquad (6)$$

where k is the productivity factor such that **N**/k equals the number of workers necessary to produce N units in regular time without incurring any overtime or undertime.

The other factor, **G**, which ranges from 1 to 0 (decreased in steps of 0.1), controls the proportion of hiring (firing) and overtime in adjusting the work force size when production is switched from normal to high (low) levels. For example, when **G** equals one in both of the schemes, the work force sizes for high and low production levels become **H**/k and **L**/k respectively, resulting in no overtime costs but high hiring and firing costs. The procedure searches for the optimal value of **G**.

Table 2 is a sample output that shows the number of the work force that will be needed when the normal production level (N) is 300 units, productivity (k) is 5.4 units per worker and parameter E which determines the high and low production level is 75. In the table N-W_t, H-W_t, and L-W_t represent the size of the work force for a given level of production (Normal, High, or Low).

Examining the values in the table using Scheme 1, it can be noted that there are significantly different values observed in W_t according to the different values of **G**. When these values of W_t are applied to the total cost function, TC, they will cause TC to become very large. This is due to the fact that at low **G** values the resulting size of the work force will always be meaningless (i.e. 0.0 for high and low regular work force size literally means that no workers should be employed and further implies that all production is accomplished using overtime labor which is impossible if no workers are employed).

Table 2. Sample calculations using SCHEME 1, and SCHEME 2.

	Scheme 1			Scheme 2		
G	**N-W_t**	**H-W_t**	**L-W_t**	**N-W_t**	**H-W_t**	**L-W_t**
1.0	55.6	69.4	41.6	55.6	69.4	41.6
0.9	55.6	62.5	37.5	55.6	68.1	43.0
0.8	55.6	55.5	33.3	55.6	66.7	44.4
0.7	55.6	48.6	29.1	55.6	65.3	45.8
0.6	55.6	41.7	25.0	55.6	63.9	47.2
0.5	55.6	34.7	20.8	55.6	62.5	48.6
0.4	55.6	27.8	16.6	55.6	61.1	50.0
0.3	55.6	20.8	12.5	55.6	59.7	51.4
0.2	55.6	13.9	8.3	55.6	58.3	52.8
0.1	55.6	6.9	4.2	55.6	56.9	54.2
0.0	55.6	0.0	0.0	55.6	55.6	55.6

Therefore, only a limited portion of the range of **G** needs to be evaluated (actually around 0.9 the minimum costs will always occur) to determine a minimum cost solution. This fact is very useful in reducing the computer run time required to evaluate the problem using scheme 1.

On the other hand, using scheme 2 seems more reasonable as there are only small incremental changes to the work force size for each change in **G**. Therefore, the whole range needs to be examined to find the minimum cost.

APPLICATION

In order to demonstrate the modified production switching heuristics (MPSH) described and to evaluate its performance relative to other aggregate production planning approaches, the MPSH proposed in this research is applied to the paint factory problem originally described by Holt, Modigliani, and Simon (1955). The paint factory problem has been used in the context of introducing most new or modified APP models proposed by various authors to evaluate whether the newer model can perform as well as the LDR. Competing models are generally judged by evaluating the method which minimizes the total costs using LDR-type quadratic cost functions.

1. LINEAR DECISION RULE (LDR)

LDR is a mathematical model designed to make decisions that set aggregate production rates and work force levels for the upcoming period. The two decision rules involved (one for production rate and one for work force level) are derived from a cost model developed for each individual situation and are optimum for the model. The cost model is the simple sum of the following cost functions:

(1) Regular payroll costs. The cost of regular-time production in period t was assumed to be

$$C_{1t} = \mathbf{c_1} W_t$$

Notice that the cost of regular production was linearly related to the size of work force, as shown in Figure 3-(a).

(2) Hiring and firing costs. The cost of increasing or decreasing the work force in period t was assumed to be

$$C_{2t} = \mathbf{c_2}\ (W_t - W_{t-1})^2$$

The cost of changing the work force was a squared function of the amount of increase or decrease in the work force. This function was an approximation to the costs observed in the paint factory, as shown in Figure 3-(b). The quadratic form was chosen for mathematical convenience, as an approximation.

(3) Overtime costs. The overtime cost was expressed as zero cost up to 100 percent utilization of the work force and then as a linear cost for overtime production beyond 100 percent (Figure 3-(c)). Through the use of a quadratic function, this overtime cost was approximated as follows:

$$C_{3t} = \mathbf{c_3}\ (P_t - \mathbf{c_4}W_t)^2 + \mathbf{c_5}P_t - \mathbf{c_6}W_t$$

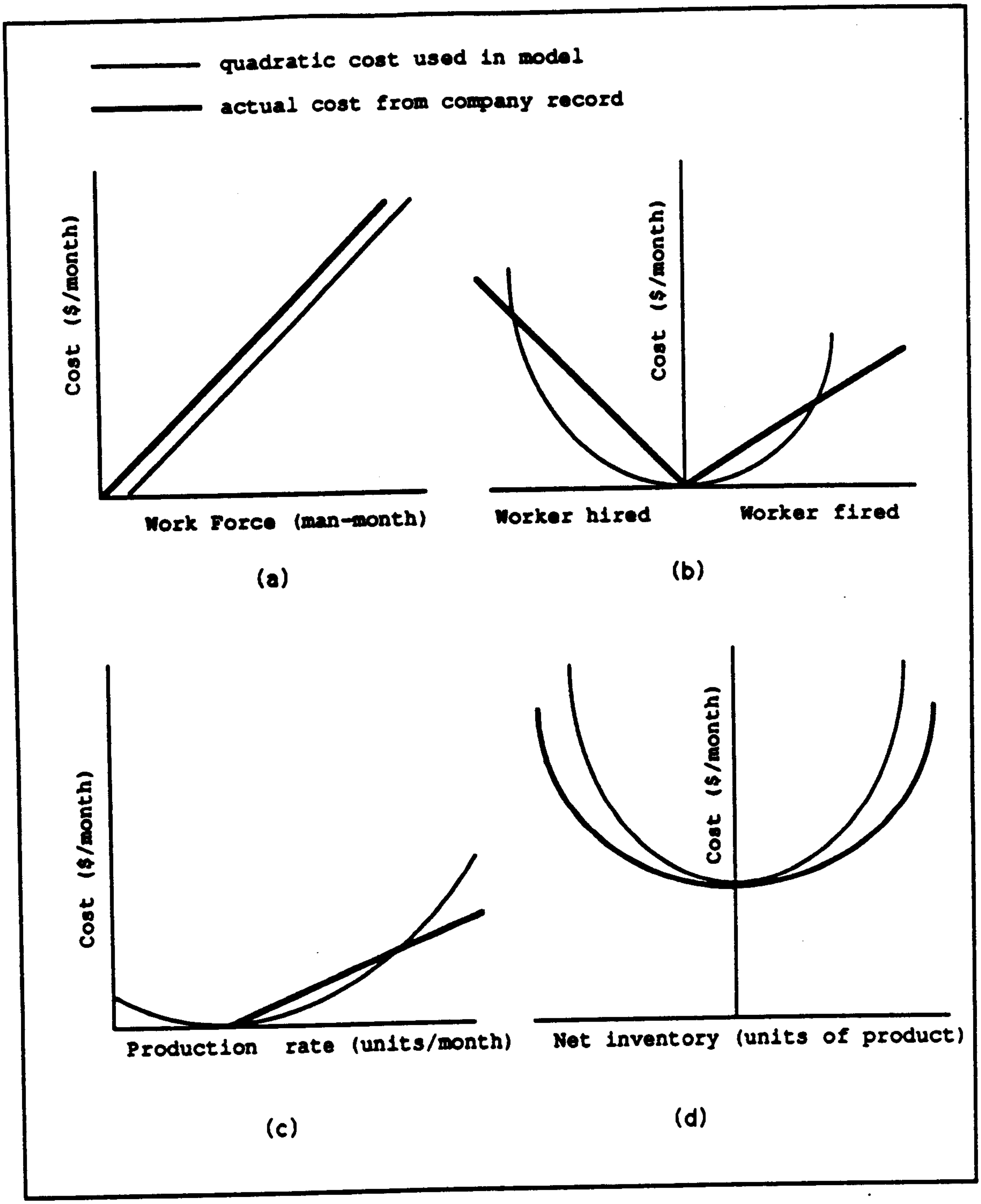
(4) Cost of inventories and back orders. In the LDR formulation, back orders were treated as negative inventory. The following quadratic inventory/back order cost function was used (figure 3-(d)).

$$C_{3t} = \mathbf{c_4}\ (I_t - \mathbf{c_8})^2$$

The resulting two decision rules are simple expressions that make it possible to compute the required decisions (production and work force level) for the upcoming period in five to ten minutes, given the ending aggregate work force and inventory levels and sales forecasts for the planning horizon [Buffa and Taubert 1967].

The mere proposal of the LDR probably would not have achieved so much attention had it not been for the fact that its authors had the wisdom and foresight to carry their research through to an actual extensive application in a paint factory. It was in a sense an ideal piece of academic-management research because it involved the derivation and development of a theory as well as its application. At any rate, the LDR rapidly achieved the status of the "standard for comparison" for aggregate production planning models because it proposed optimum rules, given the model, and because of the extensive reporting of the paint factory data and the LDR approximating cost structure as the basis for comparison [Buffa and Taubert 1972].

Figure 3. Costs for Linear Decision Rule



2. COST ESTIMATION

Regular payroll costs that include fixed or allocated portions are easily accessible via the firm's accounting system. From a modeling standpoint, however, we are interested in marginal or variable costs - costs that often are difficult to obtain. Vergin (1980) noted that much of the APP literature is plagued by this problem; models are developed or compared utilizing costs that are constant, regardless of the methodology involved. The PSH utilizes only direct payroll costs and fringe payments that vary directly with hours worked. Overtime costs are estimated directly as a multiple of the regular hourly rate. An upper bound on these hours normally is established by the union.

Hiring and firing costs, at best, are difficult to quantify. These costs result directly from training time incurred for new or bumped personnel. Losses in efficiency are experienced following a layoff or a hire. In most firms, union seniority rules often force experienced workers into new production areas requiring various levels of retraining. Expected values are required for layoff periods, bumping costs, new (versus retrained) workers involved per hire and per layoff, and related training costs.

The inventory costs are approximated using the point estimate of Holt, Modigliani, Muth and Simon (1960) rather than interval estimate. Holt et al.'s economic justification is based on the assumption that individual lot sizes and safety stocks can be aggregated to determine an overall target inventory that reflects the relevant costs of setups and inventory. Quadratic, linear, or general cost functions are formulated to penalize deviations from this point estimate. It is assumed that setup costs increase as inventory drops below the target and lots become smaller and more frequent. Handling and obsolescence costs increase as inventory moves above the desired point.

3. THE PAINT FACTORY

The cost relationships used for the paint factory were:

$C_{1t} = 340*W_t$ (Regular Payroll)

$C_{2t} = 64.3*(W_t - W_{t-1})^2$ (Hiring and Layoffs)

$C_{3t} = 0.2*(P_t - 5.67*W_t)^2 + 51.2*P_t - 281*W_t$ (overtime)

$C_{4t} = 0.0825*(I_t - 320)^2$ (Inventory)

and the objective is to minimize total costs

$$TC = C_{1t} + C_{2t} + C_{3t} + C_{4t}$$

where,

P_t = production level during the period t

W_t = work force size during the period t

I_t = ending inventory for period t

t = 1,2, . . ., 12

beginning inventory (I_0) = 263

beginning work force (W_0) = 81.

forecasted demands are:

Table 3. Forecasted Demand

t	F(t)	t	F(t)
1	430	7	292
2	447	8	458
3	440	9	400
4	316	10	350
5	397	11	284
6	375	12	400

Especially, 5.67, the coefficient in the overtime cost component, is the average worker productivity which is introduced as k. The overtime cost component yields negative overtime costs for certain values of P_t and W_t. Whenever this occurred in the calculations, C_{3t} was set to zero. In the paint factory problem, when back orders occur they show up in the results as minus inventory.

4. PROGRAM SUMMARY

The program seeks to minimize total production costs by looking at the best combinations of inventory, work force and overtime costs. Two different schemes are used in determining this. A fundamental assumption made is that productivity per worker is a constant. Starting with the forecasts, an initial production rate (N) is arbitrarily determined by looking at the demand over the time horizon and picking an initial value for N that falls somewhere between the high and low demand value. An estimation of the target inventory level (B) is made by looking at the current level of inventory, I_0, then picking an inventory level small enough to include the initial inventory level in a grid search (in the paint factory problem 240 was selected). With initial values for N and B the grid search can be iterated by fixed increments to determine the best values of N and B to minimize the cost function.

The level of production needs to be assigned as either High, Normal or Low by using the appropriate switching algorithms. The differences between Normal and High production rates and inventory levels are labeled E and D, respectively. These differences are also increased by fixed increments after each iteration.

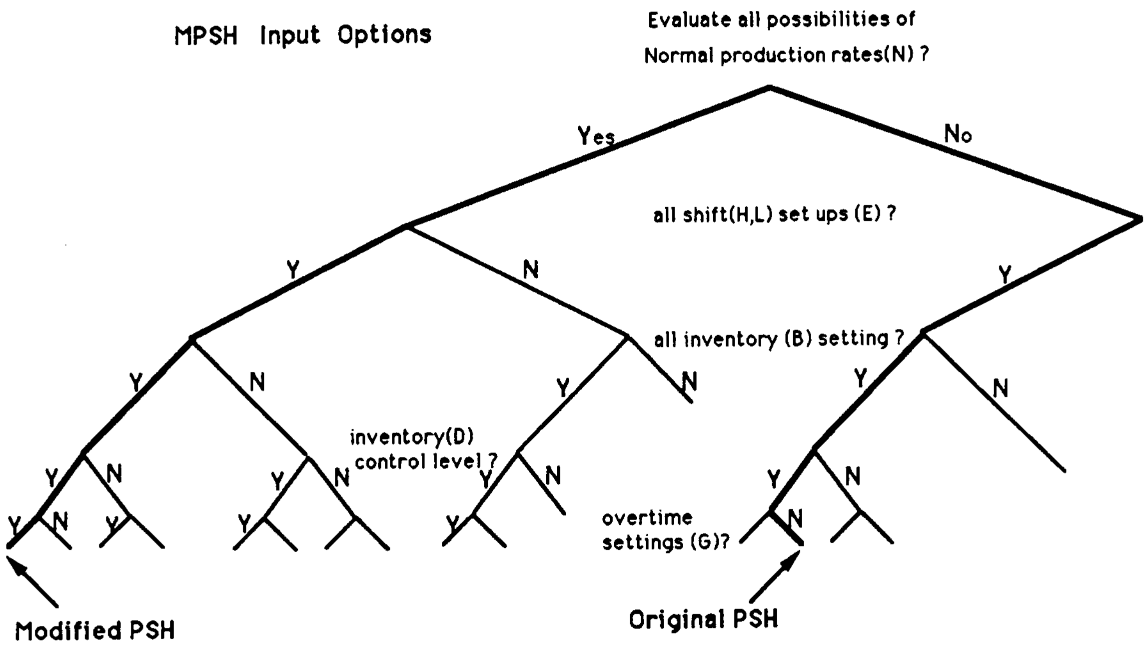
At each iteration, only one parameter is exhausted according to the FOR LOOP in a computer program. The search method is carried out in three steps to reduce cpu time and systematically searches over the entire cost surface for the minimum cost point. First, using a relatively large incremental value "20" for the parameter **N**, it rapidly determines the zone for the lowest cost point. It then uses a medium incremental value "5" to search a smaller surface around the previous point for a better solution. Finally, the small incremental value "1" is used to determine the global minimum point - here "global" means not mathematical optimum but the best answer for the entire set of incremental values. Clearly, the smaller incremental value yields the better solution.

Given the demand forecasts for the next twelve months in the paint factory problem, the PSH determines the best control parameters available. These parameters determine shift settings, overtime levels, production levels, and inventory levels that minimize aggregate costs.

The MPSH is interactive in nature. The firm exercises the option to view the total set of regular and overtime production settings and inventory target levels or any number thereof. Aggregate plans can be determined based on restricted production rate shift settings, with or without overtime, with various forecast series and with varying ranges for inventory targets. Figure 4 illustrates the decision tree and interactive nature of the MPSH's. It shows the MPSH options as compared with the original PSH, including the incremental values of each parameter, the range varied for the parameter, and the number of arrays (combinations) for each iteration. Starting conditions are provided, aggregate costs are then calculated for each planning component and explicitly given. A flow chart of the MPSH is provided in Appendix 1. The program listing is provided in Appendix 2. The program was run using a MacII personal computer in THINK PASCAL.

Figure 4. Input Options and Grid Search

MPSH Input Options



Grid Search for options

Control Parameter	Incremental Value	Range	Array
N Normal Production Level	20, 5, 1	80, 40, 10	5+9+11
E High(H) Production Level Low(L) Production Level	5	90	19
B Target inventory Level	10	80	9
D Maximum Acceptable Inventory Level (C) Minimum Acceptable Inventory Level(A)	5	80	17
G Overtime and Regular Production	0.1	1	11
	Maximum # of Evaluations		799,425

RESULTS AND ANALYSIS

1. RESULTS

A computer search routine based on the seven step procedure described previously was conducted to determine the best values to use for control parameters with the MPSH. With MPSH1 the computer run time was 52 minutes. With MPSH2, the run time was 92 minutes. In both cases the program was in THINK PASCAL and a MacII, personal computer was utilized to perform the computer runs. Table 4 below shows the results of that search routine.

Table 4. Results For MPSH Parameters

PARAMETER	MPSH1	MPSH2
H	450	447
N	360	362
L	270	277
E	90	85
A	300	290
B	300	290
C	300	290
D	0	0

With these sets of parameters, MPSH1 and MPSH2 were used to determine a production plan. Tables 5, and 6 show the computer results of production plans including production and work force levels, and overtime rates for

Table 5. Production planning for MPSH1

	PRODUCTION (P_t)		WORK FORCE (W_t)	OVERTIME	
period	**level**	**gallon**	**people**	**people**	**gallon**
1	H	450	71.43	7.9	45
2	H	450	71.43	7.9	45
3	H	450	71.43	7.9	45
4	N	360	63.49	0.0	0
5	N	360	63.49	0.0	0
6	N	360	63.49	0.0	0
7	N	360	63.49	0.0	0
8	N	360	63.49	0.0	0
9	N	360	63.49	0.0	0
10	N	360	63.49	0.0	0
11	N	360	63.49	0.0	0
12	N	360	63.49	0.0	0

Table 6. Production planning for MPSH2

	PRODUCTION (P_t)		WORK FORCE (W_t)	OVERTIME	
period	**level**	**gallon**	**people**	**people**	**gallon**
1	H	447	72.83	6.0	34
2	H	447	72.83	6.0	34
3	H	447	72.83	6.0	34
4	N	362	63.84	0.0	0
5	N	362	63.84	0.0	0
6	N	362	63.84	0.0	0
7	N	362	63.84	0.0	0
8	N	362	63.84	0.0	0
9	N	362	63.84	0.0	0
10	N	362	63.84	0.0	0
11	N	362	63.84	0.0	0
12	N	362	63.84	0.0	0

the problem planning horizon. These results show that only two production levels (high and normal), without a low production level, are chased to minimize the total cost over the planning horizon.

From these results, it is easy to determine the appropriate values for G which provide the best combination of production and work force levels.

G for MPSH1 is 0.9 ((450 / k)* G = 71.43), and

G for MPSH2 is 0.6 ((362 / k) + (85 / k)* G = 72.83),

where k (the productivity rate) is equal to 5.67. The overtime production required at the high production level for each production planning period can then be calculated, simply, as follows:

overtime used for MPSH1 : 450 / 5.67 - 71.43 = 7.9 people

overtime used for MPSH2 : 447 / 5.67 - 72.83 = 6 people

These calculations yield overtime rates of 11% and 8.2% for MPSH1 and MPSH2, respectively. It is important to note that only high and low production levels can have overtime production associated with them (see equation 4, and 5).

In the overtime component, C_{3t},

$$C_{3t} = 0.2(P_t - 5.67W_t)^2 + 51.2P_t - 281W_t$$

the coefficient value, 5.67 represents the average worker's productivity rate. The linear portion of the overtime component, $P_t - 5.67W_t$, denotes the required overtime production at the average worker productivity level in order to achieve the production rate, P_t, at a work force level, W_t. Clearly, overtime production can only occur in period t if $P_t - 5.67W_t > 0$. Since the low production level is not used in the production schedule produced by MPSH1 and MPSH2, $P_t - 5.67W_t$ is evaluated as 45, and 34, respectively. Both of these values are > 0, indicating that, in the case of the high production level required overtime labor will be used.

Table 7 gives the total costs and cost components (i.e. payroll, hiring, firing, overtime and inventory costs) for MPSH1 and MPSH2, respectively. The overall total cost was $295,178 with MPSH1 and $294,979 with MPSH2. This is reasonable since MPSH2 requires a more elaborate search and thus should produce a better cost solution. It should also be noted that MPSH1 yields a solution with less regular labor costs and more overtime costs than does MPSH2. This is further reflected in the hiring and firing costs in that, since MPSH2 chases a higher regular payroll and less overtime combination, it produces hiring and firing costs that are somewhat less than MPSH1 which chases an opposite combination of regular payroll and overtime. Inventory costs will be addressed later in the comparative analysis section.

Table 7. Cost Component and Total Cost for Paint Factory

COST COMPONENT	MPSH1	MPSH2
Regular payroll	$ 267,142	$ 269,661
Overtime	15,437	13,259
Hiring/firing	9,941	9,484
Inventory	2,658	2,538
Total cost	295,178	294,979

2. COMPARISON WITH OTHER APPROACHES

Tables 8, 9, and 10 provide the period by period production plans resulting from using LDR, PPP, PSH with the paint factory problem, respectively, when perfect forecasts (no errors between forecasted and actual sales) are available to the firm. These values have been well reported on in numerous APP articles. These particular values are taken from the article by Mellichamp and Love (1978). At this point, it should be recognized that LDR gives an optimal solution to the paint factory problem against which all other approaches should be compared. The PPP and PSH results are both near optimal. PSH uses fixed increments or decrements based on the production level selected, high, low or normal. The results for the period by period production plans using MPSH1 and MPSH2 are shown in table 10. These results were determined much like those in the original PSH model except that they use the best control parameters as determined and given previously in table 2.

Table 8: Linear Decision Rule (LDR) Optimal Aggregate Plan

period	Forecast (gallons)	Production (gallons)	Work force (people)	Inventory (gallons)
0	-	-	81.00	263.00
1	430	467.72	78.63	292.72
2	447	441.32	75.32	289.08
3	440	414.88	72.24	263.92
4	316	379.83	69.55	328.75
5	397	375.28	67.21	309.03
6	375	367.09	66.29	301.12
7	292	358.51	65.66	369.64
8	458	380.57	65.87	295.21
9	400	376.80	66.49	270.01
10	350	366.70	67.68	283.71
11	284	366.59	69.67	365.30
12	400	405.95	72.62	366.24

Table 9: Parametric Production Planning (PPP) Aggregate Plan

period	Forecast (gallons)	Production (gallons)	Work force (people)	Inventory (gallons)
0	-	-	81.00	263.00
1	430	461.26	78.56	286.26
2	447	440.50	75.37	281.76
3	440	417.11	72.44	258.88
4	316	380.38	69.82	324.25
5	397	379.80	67.98	309.06
6	375	371.34	66.74	305.39
7	292	360.91	66.13	376.30
8	458	390.73	66.53	312.03
9	400	385.71	67.25	297.74
10	350	372.14	68.39	314.88
11	284	367.31	70.17	397.19
12	400	408.72	72.93	400.91

Table 10: PSH - Mellichamp and Love's Aggregate Plan

period	Forecast (gallons)	Production (gallons)	Work force (people)	Inventory (gallons)
0	-	-	81.00	263.00
1	430	452.42	70.82	285.42
2	447	452.42	70.82	290.83
3	440	382.42	67.45	233.25
4	316	382.42	67.45	299.67
5	397	382.42	67.45	285.09
6	375	382.42	67.45	292.50
7	292	382.42	67.45	382.92
8	458	382.42	67.45	307.34
9	400	382.42	67.45	289.75
10	350	382.42	67.45	322.17
11	284	312.42	64.07	350.59
12	400	382.42	67.45	333.00

Table 11. Modified PSH aggregate plans

		MPSH1			MPSH2		
Period	Forecast	Production	Work force	Inventory	Production	Work force	Inventory
0	-		81.0	263	-	81.0	263
1	430	450	71.43	283	447	72.83	280
2	447	450	71.43	286	447	72.83	280
3	440	450	71.43	296	447	72.83	287
4	316	360	63.49	340	362	63.84	333
5	397	360	63.49	303	362	63.84	298
6	375	360	63.49	288	362	63.84	285
7	292	360	63.49	356	362	63.84	355
8	458	360	63.49	258	362	63.84	259
9	400	360	63.49	218	362	63.84	221
10	350	360	63.49	228	362	63.84	233
11	284	360	63.49	304	362	63.84	311
12	400	360	63.49	264	362	63.84	273

The values obtained using all methods are discussed in more detail later, however one should note several obvious differences in the production, work force and inventory columns corresponding to each period in the planning horizon.

If total production is considered between the various techniques, PPP yields the largest annual production with 4735 gallons. Next is LDR with 4,700. PSH yields 4,659 while MPSH1 & 2 yields 4,590 and 4,599, respectively. This production can be compared against the total annual demand of 4,589 gallons to show that MPSH1 & 2 yield the least difference between the amount of total annual production and forecasted demand.

In the paint factory problem, the inventory cost function is represented by the inventory costs accrued from the difference between the end of the month inventory and a target inventory value of 320 (i.e. $C_{4t} = 0.0825*(I_t - 320)^2$). This means that the penalty costs of holding inventory will be greatly reduced even when I_t is large, due to multiplying by a very small constant (.0825).

Another way to view this would be that with regard to the total inventory costs where the holding costs c_I is some amount in the formulation:

$$IC = \mathbf{c_I} * (P_t - F_t + I_{t-1}).$$

The best policy would be to chase production (that's why LDR production is greater than the other approaches) since doing so serves to minimize the total costs for holding inventory.

As stated previously, the work force level changes at every period with both the optimal LDR approach and the PPP near-optimal approach. PSH was developed in part to reduce these numerous production changes. The results presented in table 10 show four changes with respect to work force, to include the requirement for low level production in period 11. With the MPSH1 & 2 approaches, these changes are further reduced to just two production changes over the planning horizon. This should be even more appealing to practitioners. Finally the tables show that when ending inventory balances are not restricted to some level, MPSH1 & 2 yield the least amount of ending inventory.

Table 12 below shows a comparison for all approaches with respect to their total variable costs. Note that as in previous comparisons of APP approaches the regular payroll costs are considered in analyzing the performance of the approaches.

Table 12. Cost Comparison Results

Cost component	MPSH1	MPSH2	PSH	PPP	LDR
Regular payroll	$267,142	$269,661	$276,338	$285,141	$282,642
Overtime	15,437	13,295	13,200	7,810	8,518
Hiring/layoff	9,941	9,484	8,863	3,229	3,514
Inventory	2,658	2,538	1,494	1,865	1,362
Total cost	295,178	294,979	299,895	298,045	$296,036
Adjusted cost	$301,294*	$300,555	$301,873		

* $301,294 = 295,178 + 340*(102 gallons)/(5.67 gallons/man month).

Vergin (1980) has pointed out that these payroll costs should be treated as essentially fixed or the various models should require some similar ending conditions if any comparative analysis is to be made. The total cost values of $295,178 for MPSH1, and $294,979 for MPSH2 are excellent in comparison to PSH and the other models. However, noting Vergin's notes, no direct comparison of the models can be made since the figures in table 12 include neither only "relevant" costs nor similar ending conditions.

To develop similar ending conditions (i.e. inventories) it is known that MPSH1, and MPSH2 resulted in an ending inventory difference from the optimal LDR balance of 102 gallons (366-264) and 93 gallons (366-273), respectively. In order to make MPSH1 and MPSH2 costs comparable we must consider the cost without overtime to make their ending inventories equal to that in the optimal LDR. The regular payroll cost associated with producing 102 and 93 gallons is determined from the cost function C_{12} = 340 * W_t . This means that, at most, we must incur $6116 (= 102*340/5.67) and $5576 (= 93*340/5.67), respectively. These amounts are added to the total costs in table 12 to arrive at the adjusted MPSH1, MPSH2 cost figures.

Now comparing these results based on similar ending conditions we find that the modified PSH's - both MPSH1 and MPSH2, perform better than PSH by $579 and $1,318, respectively. The total cost value of $300,555 ($301,294) obtained with MPSH2 (MPSH1) is only 1.52 (1.77) percent greater than the optimum value of $296,036 generated by LDR. This coupled with the less frequent production changes should make both MPSH models more appealing to practitioners than the PSH approach.

3. ANALYSIS OF THE RESULTS

This section addresses the analysis of the sensitivity of the various control parameters with regard to the total cost function. To conduct this analysis, one parameter was retained as a constant value while the others were varied according to the grid search routine. This amounted to evaluating various combinations of the control parameters at fixed values of N, E, B, D, and G, respectively, to determine for each fixed value the local minimum associated with that value. Tables 13 - 17, in Appendix 3, show the minimum total cost solution with parameter combination which determined the point of local minimum for MPSH1. In the same manner, the parameters for MPSH2 were investigated. Tables 18 - 22, in Appendix 4, show the results of the local minimum total cost determined with associated parameters for various fixed values of each parameter (N,E,B,D, and G). Based on these tables, two composite graphs (figure 10 for MPSH1 and figure 11 for MPSH2), showing how the total cost is affected by each of the parameters, are shown in Appendix 5. Additionally, a comparison of the results with MPSH1 and MPSH2 has been performed by plotting each parameter (N, E, B, D, and G) as shown in figures 5-9. For four of the parameters (N, E, B, and,D), it can be readily observed that the variation of costs obtained using MPSH1 and MPSH2 are very similar. It was previously stated that in MPSH1 at certain values of G, the work force size becomes meaningless (i.e. the costs vary widely - see figure 9).

Figure 5. Total Costs on Parameter N

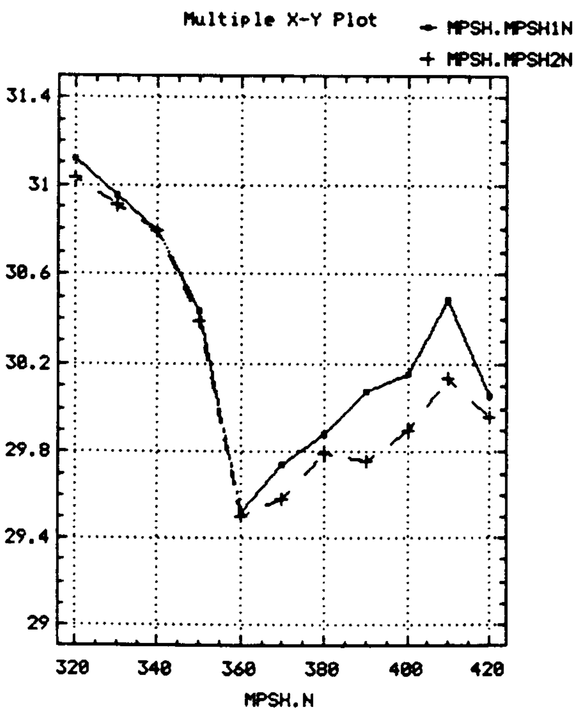


Figure 6. Total Costs on Parameter E

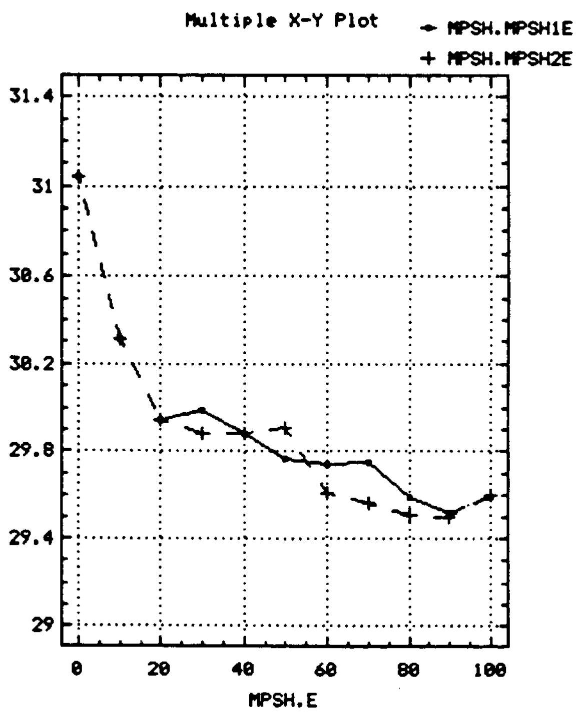


Figure 7. Total Costs on Parameter B

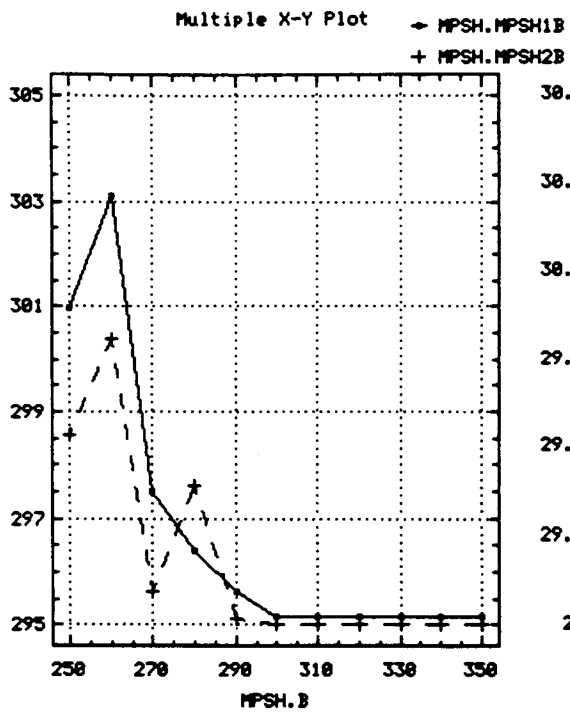


Figure 8. Total Costs on Parameter D

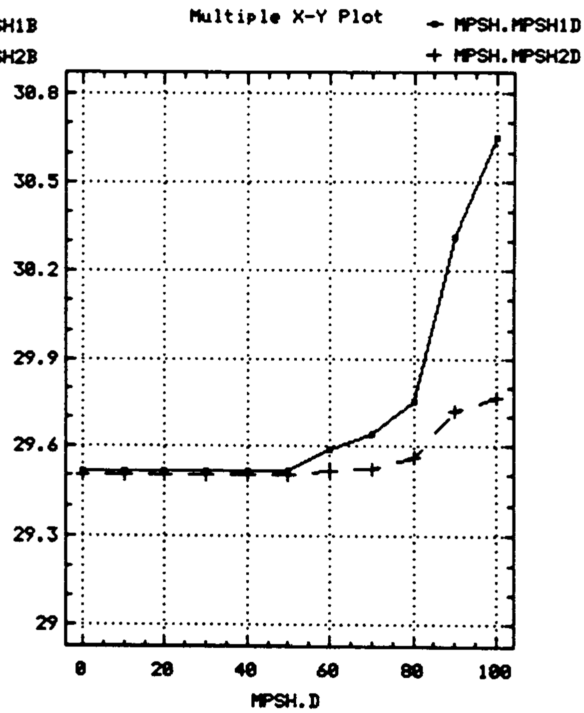
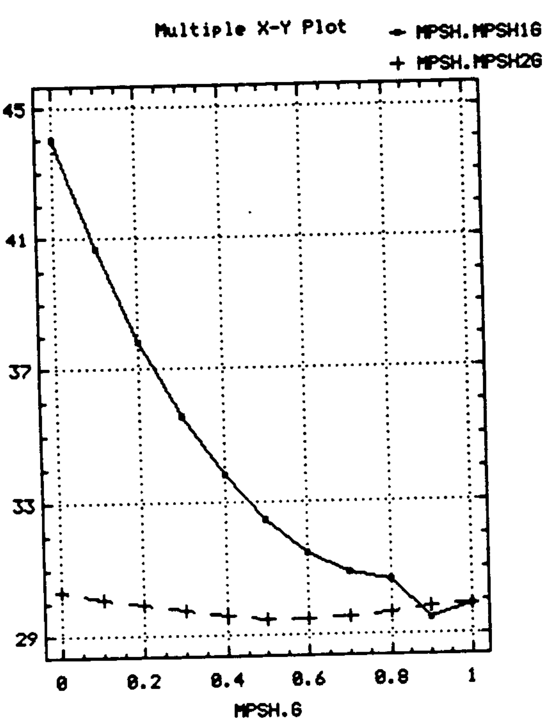


Figure 9. Total Costs on Parameter G



Appropriately, then, G, at values less than 0.8, shows a wide variation in total costs compared to the other parameters as shown in figure 10. It is followed in minimum total cost sensitivity by parameters N and E. On the other hand the minimum total cost is relatively insensitive to various values of parameters B and D.

These results are reasonable since parameters N, E, and G are directly related to the total cost function while parameters B and D are related to the production switching algorithm. N and E are used in an evaluation of P_t, the production level. G is used in an evaluation of W_t, the work force. Both of these values (P_t and W_t) are used to determine production and labor costs as well as inventory costs. B and D on the other hand simply determine whether or not high, normal or low production rates will be used during a given planning period.

As can be seen in figure 11 the minimum total cost in MPSH2 is most sensitive to various values of parameters N and E followed. In MPSH2 the productivity function is more rational than in MPSH1. Accordingly, each level of G has a small but meaningful impact on the determination of W_t. This accounts for G being significant in MPSH1 where meaningless work force levels can dramatically effect the cost function but less significant in MPSH2 where it only has a small impact at each level of production.

The minimum total cost again appears as being relatively insensitive to various fixed values of parameters B and D. This result is reasonable as both MPSH1 and MPSH2 treat these parameters in the same manner with regard to the switching algorithm.

Eilon (1974) and Mellichamp and Love (1978) point out several disadvantages of the PSH. These can be summarized as the tendency of PSH to peg to either high or low production rates when confronted with seasonal or nonstationary demand patterns. This in effect causes PSH to create excessive inventory surpluses or shortages. With the MPSH1 and MPSH2, in the paint factory problem, the models did not produce the low production level in period 11 like PSH. Instead, the models chased only a high and normal production level. This can be accounted for the more sophisticated grid search employed in this research which tends to less closely track between production and demand as compared to the PSH model.

A final point should be made with regard to the computer run times experienced in this research. It is important to point out that PSH, PPP, and LDR results were obtained using a UNIVAC 1110 computer or its equivalent with run times, in the case of PSH, being in the vicinity of 30 seconds. In this research, a MacII, Personal Computer was used with a run time of 52 minutes with MPSH1 and 92 minutes with MPSH2. In evaluating any claim of efficiency of MPSH1 or MPSH2 over PSH, the models should be run on an equivalent computer and the run time results compared. This may be a consideration worth further future research.

Within the context of cost, however, PC time might be argued to be substantially cheaper for managers than time-sharing or other mini or main frame arrangements. The power of personal computers is a strong plus for practitioners in adopting either the MPSH1 or MPSH2 model. The run time difference, between the models is, obviously, the difference between searching over the entire range of G, in MPSH2, and only searching in the range of G = 0.9, in MPSH1. The savings is approximately 40 minutes of personal computer time for a total cost solution that is within 1.52% of the optimal LDR solution with MPSH2 and 1.77% with MPSH1. Both models outperform the PSH model in total cost.

4. SUMMARY AND CONCLUSION

The production switching approach described in this research offers several clear advantages over other approaches for handling the aggregate production planning (APP) problem. The principal advantage is that it produces production, work force, and inventory decisions which require a minimum amount of period to period adjustment - a characteristic that should be appealling to the natural inclinations of practicing managers in industry.

In this research, the PSH has been modified with an improved search method, which exhaustively searches over the entire cost surface. These modifications are accomplished using two schemes which are labeled MPSH1 and MPSH2 for convenience. The computational requirements of either scheme are quite large, however, the search procedure is relatively straightforward. The run times for the series of programs developed on a personal computer are time-consuming - to solve the paint factory problem, MPSH1 took approximately 0.86 hours, as opposed to nearly 1.55 hours by the MPSH2, running on MACII PC computer using the THINK PASCAL language.

Modified PSH, applied to the paint factory problem, however, has shown that it can improve the total cost performance of the original PSH (Mellichamp and Love, 1978). The modified production switching approach also offers benefits in simplicity and flexibility for a minimal sacrifice in cost - around two percent of optimal. This feature may appeal to decision makers in industries who are not pursuing an optimal scheduling policy.

Analysis of the model showed the minimum cost function to be sensitive to values of the control parameters directly related to the cost function versus insensitive to those only indirectly related. These results were reasonable and they demonstrated the importance of allowing these parameters to be determined by an open search versus specified as fixed based on average demand as was the case in the PSH model.

The modified approaches developed in this research offer practitioners a model with a series of programs which can be run on a personal computer with relative ease. This is an advantage over other published works which offer switching heuristics but do not do so in the context of a pre-packaged program that might be adapted for use by practitioners in industry.

6. FUTURE RESEARCH

As noted previously, one of the directions for future research is to run the modified PSH on a similar computer to that used in presenting LDR, PPP, and PSH to help establish some run time efficiency comparisons in the context of the paint factory or other common APP problem reported in published literature. One might expect the modified approaches to take longer to run but how much longer would be the question of interest. Based upon these findings a comparative cost benefit analyses may be performed.

Although MPSH1 and MPSH2 provide decisions on production and work force levels for the entire planning horizon (January through December) in an industrial setting, such decisions can be improved as and when better forecasts become available. For instance, the same analysis performed in January can be repeated in February with forecasts for February through December and January of next year. This is generally referred to in literature as a rolling horizon [Buffa and Taubert 1972] approach to solving the APP problem. As MPSH1 and MPSH2 rely on a fairly elaborate exhaustive search technique, given a family of cost functions (i.e., regular payroll, overtime, firing and hiring, and inventory), future research should be focused on determining a set of favorable starting values for the control parameters (N, E, B, D, and G) that can lead to identifying the local optimum more efficiently than performing a search over the entire cost surface when a new set of forecasted demand becomes available. This would first require performing sensitivity analyses to determine the variation in total cost for changes in each control parameter. From this analysis, it is hoped, a set of decision rules relating each parameter and the forecasted demand may be established and used in the determination of favorable starting values.

Finally, further future research may be directed to follow the lead of other APP investigations that point out that hiring and firing costs generally do not take into account the effects of the skill of the workers being laid off or hired. This investigation may be described as attempting to refine the MPSH models to account for learning curve effects and would entail developing some sophistication in the general cost function representing the cost of hiring or firing workers.

BIBLIOGRAPHY

Anshen, M., Holt, C. C., Modigliani, F., Muth, F. J., and Simon, H. A., 1958, Mathematics for production scheduling. Harvard Business Review, **3,** 51.

Axsater, S., 1981, Aggregation of product data for hierarchical production planning. Oper. Res., **29,** 744.

Bedworth, D. D., and Bailey, J. E, 1982, Integrated Production Control Systems: Management, Analysis, Design 2/E. (Wiley, New York)

Bergstrom, G. L., and Smith, B. E., 1970, Multi-item production planning- An extension of the HMMS rule. Mgmt. Sci., **16,** B614.

Bitran, G. R., and Hax, A. C., 1977, On the design of Hierarchical Production Planning systems. Deci. Sci., **8,** 28.

Bowman, E. H., 1956, Production scheduling by the transportation method of linear programming. Operations Research, **4**, 100.

Bowman, E. H., 1963, Consistency and optimality in managerial decision making. Mgmt. Sci., **9,** 310.

Buffa, E. S., and Taubert, W. H., 1972, Production-Inventory Systems: Planning and control (Irwin, Homewood, Ill).

Damon, W. W. and Schramm, R., 1972, A simultaneous decision model for production,marketing and finance. Mgmt. Sci., **19,** 161.

Dzielinski, B. P. and Gomory, R. E., 1965, Optimal programming of lot sizes, inventory and labor allocation. Mgmt. Sci., **11,** 874.

Ebert, R. J., 1976, Aggregate planning with learning curve productivity. Mgmt. Sci., **23,** 171.

Eilon, S., 1975, Five approaches to aggregate production planning.AIIE Trans., **7,** 118.

Elmaleh, J., and Eilon, S., 1974, A new approach to production switching. Int. J. Prod. Res., **12,** 673.

Goodman, D. A., 1974, A goal programming approach to aggregate planning of production and work force. Mgmt. Sci., **20,** 1569.

Groff, G. K., and Muth, J. F., 1972, Operations management: Analysis for Decisions. (Irwin, Homewood, Ill.).

Hanssmann, F., and Hess, S. W., 1960, A Linear Programming approach to production and employment scheduling. Management Technology, **1**, 46.

Hax, A. C., and Candea, D., 1984, Production and Inventory Management (Englewood Cliffs, NJ:Prentice Hall).

Holt, C. C., Modigliani, F., and Simon, H. A., 1955, A linear decision rule for production and employment scheduling. Mgmt. Sci., **2**, 1.

Holt, C. C., Modigliani, F., and Muth, J. F., 1956, Derivation of a linear decision rule for production and employment. Mgmt. Sci., **2**, 159.

Holt, C. C., Modigliani, F., Muth, J. F., and Simon, H. A., 1960, Planning Production Inventories and Work Force (Englewood Cliffs, N.J.:Prentice Hall).

Jones, C. H., 1967, Parametric Production Planning. Mgmt. Sci., **13**, 843.

Lasdon, L. S., and Terjung, R. C., 1971, An efficient algorithm for multi-item scheduling. Oper. Res., **19**, 946.

Lee, S. M., and Moore, L. J., 1974, A practical approach to production scheduling. Prod.& Inventory Mgmt., **15**, 79.

Lee, W, B., and Khumawala, B. M., 1974, Simulation testing of aggregate production planning models in an implementation methodology. Mgmt. Sci., **20**, 903.

Leitch, R. A., 1974, Marketing strategy and the optimal production schedule. Mgmt. Sci., **21**, 302.

Manne, A. S., 1958, Programming of economic lot sizes. Mgmt. Sci., **4**, 115.

McGarrah, R. E., 1983, Production and Logistics Management (Wiley, New York).

Mellichamp, J. M., and Love, R. M., 1978, Production switching Heuristics for the aggregate planning problem. Mgmt. Sci., **24**, 1242.

Nahmias, S., 1989, Production and operations analysis (Irwin, Boston, MA).

Newson, E. P., 1975a, Multi-Item lot size scheduling by heuristic Part1:with fixed resources. Mgmt. Sci., **21**, 1186.

Newson, E. P., 1975b, Multi-Item lot size scheduling by heuristic Part2:with variable resources. Mgmt. Sci., **21**, 1194.

Oliff, M. D., and Bruch, E. E., 1985, Multi-product production scheduling at Owens-corning Fiberglas. Interfaces, **15**, 25.

Oliff, M. D., and Leong, G. K., 1987, A discrete production switching rule for aggregate planning. Deci. Sci., **18**, 582.

Oliff, M. D., Lewis, H. S., and Markland, R. E., 1989, Aggregate planning in crew-loaded production environments. Comput. & Oper. Res., **16**, 13.

O'Grady, P. J., and Byrne, M. D., 1986, A combined switching algorithm and linear decision rule approach to production planning. Int. J. Prod. Res., **23**, 285.

Orr, D., 1962, A random walk production-inventory policy: rationale and implementation. Mgmt. Sci., **9**, 108

Silver, E. A., 1967, A tutorial on production smoothing and work force balancing. Oper. Res., **15**, 985.

Silver, E. A., 1972, Medium range aggregate production planning: state of the art. Prod. & Inventory Mgmt., **13**, 15.

Taubert, W. H., 1968, A search decision rule for the aggregate scheduling problem. Mgmt. Sci., **14**, 343.

Vergin, R. C., 1966, Production scheduling under seasonal demand. JIE, **17**, 260.

Vergin, R. C., 1980, On "A new look at production switching heuristics for the aggregate planning problem". Mgmt. Sci., **26**, 1185.

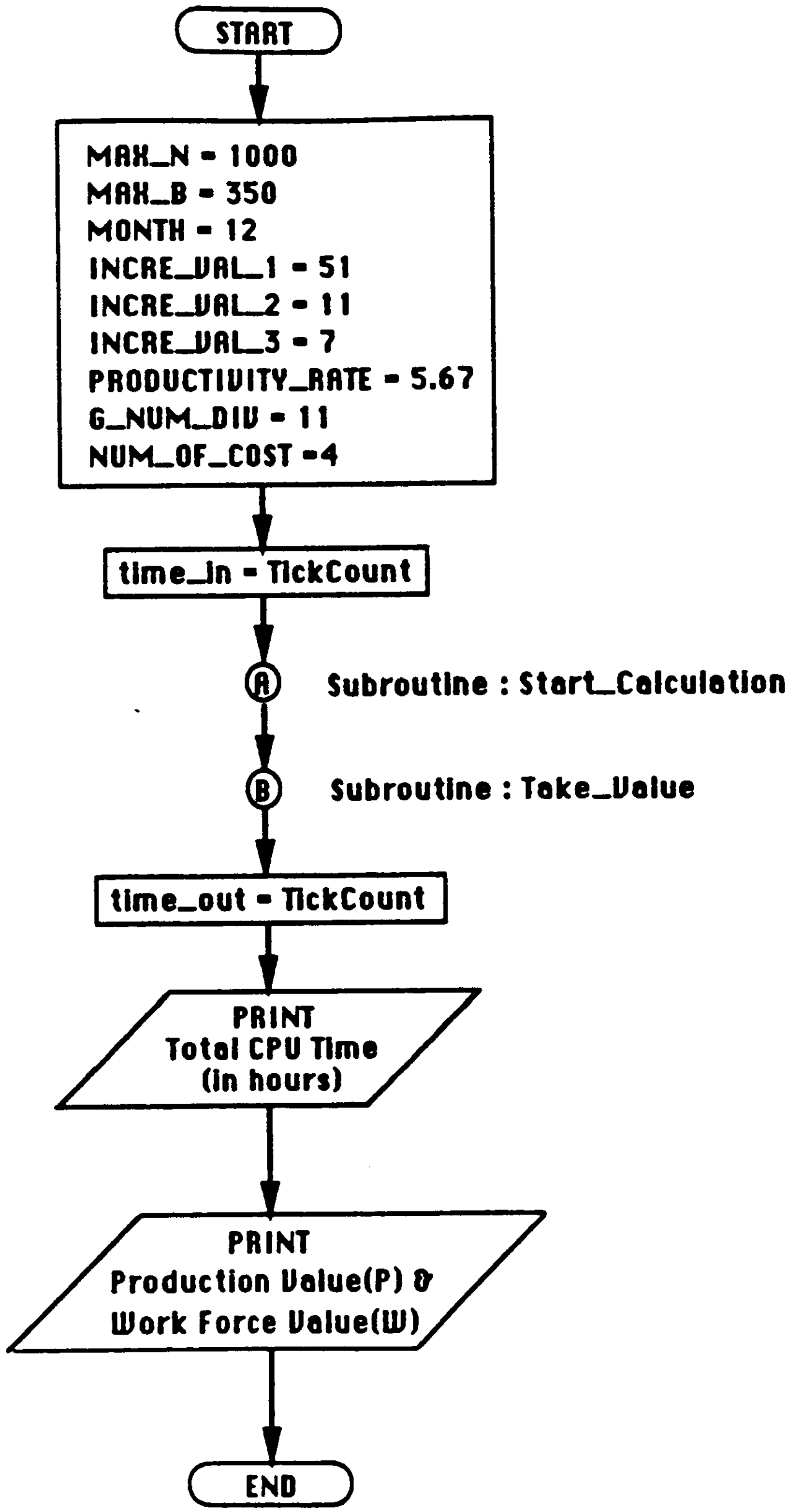
Vollmann, T. E., Berry, W. L., Whybark, D. C., 1988, Manufacturing planning and control systems. (Irwin, Homewood, Illinois).

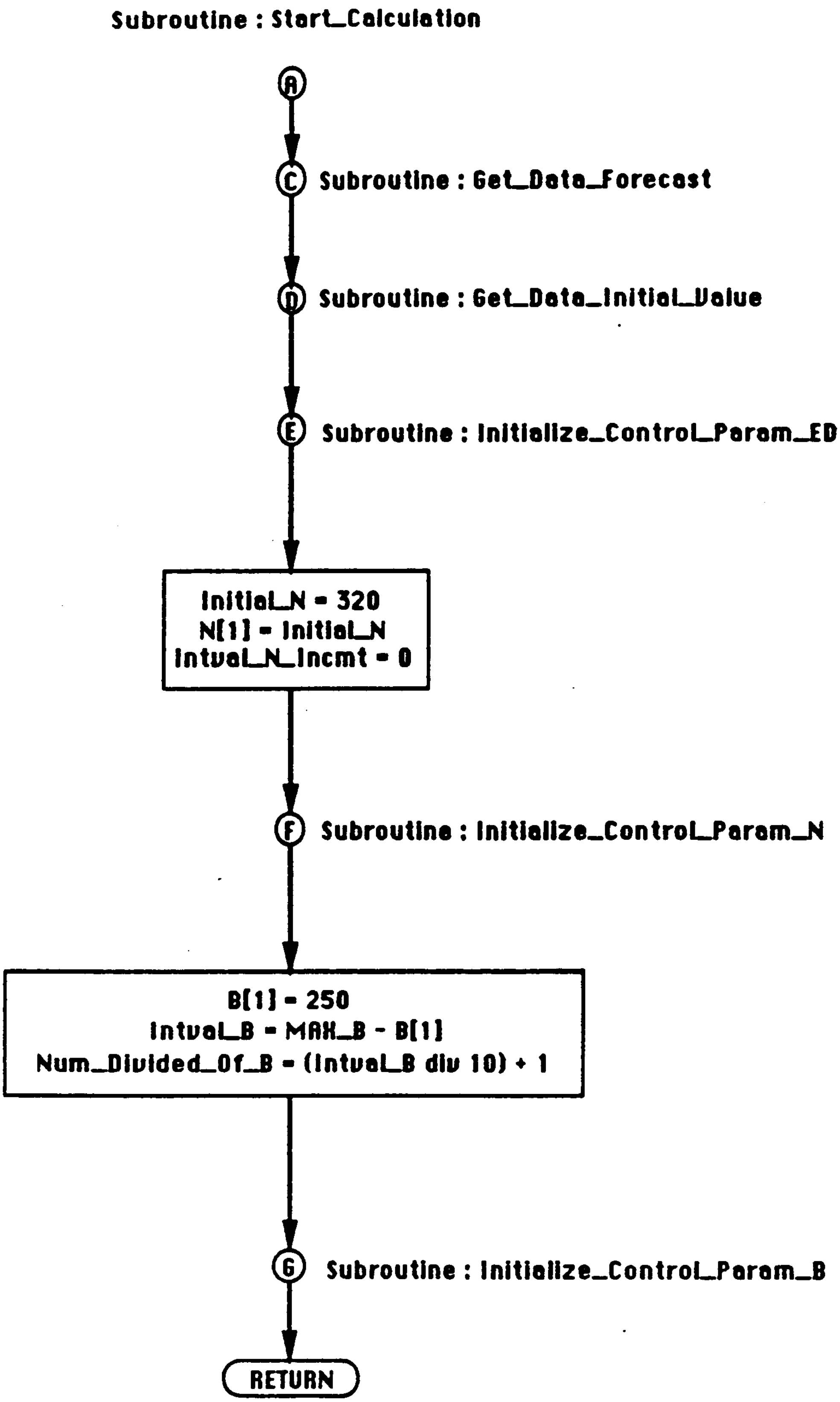
Wagner, H. M., and Whitin, T. M., 1958, Dynamic version of the economic lot size model. Mgmt. Sci., **5**, 89.

Zoller, K., 1971, Optimal disaggregation of aggregate production plans. Mgmt. Sci., **17**, B533.

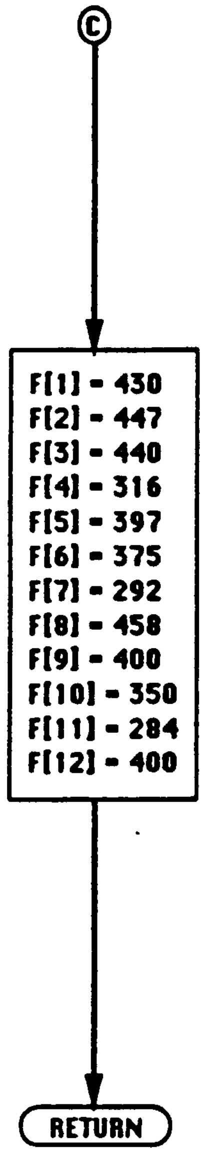
APPENDICES

APPENDIX 1: Program Flow Chart

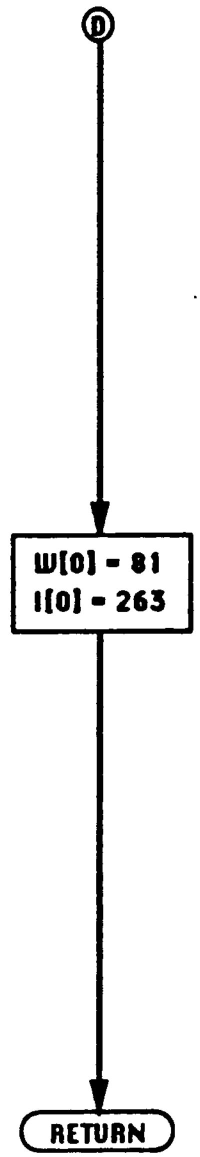



Subroutine : Start_Calculation
A
C
Subroutine : Get_Data_Forecast
D
Subroutine : Get_Data_Initial_Value
E
Subroutine : Initialize_Control_Param_ED
Initial_N = 320
N[1] = Initial_N
Intval_N_Incmt = 0
F
Subroutine : Initialize_Control_Param_N
B[1] = 250
Intval_B = MAX_B - B[1]
Num_Divided_Of_B = (Intval_B div 10) + 1
G
Subroutine : Initialize_Control_Param_B
RETURN

Subroutine : Get_Data_Forecast

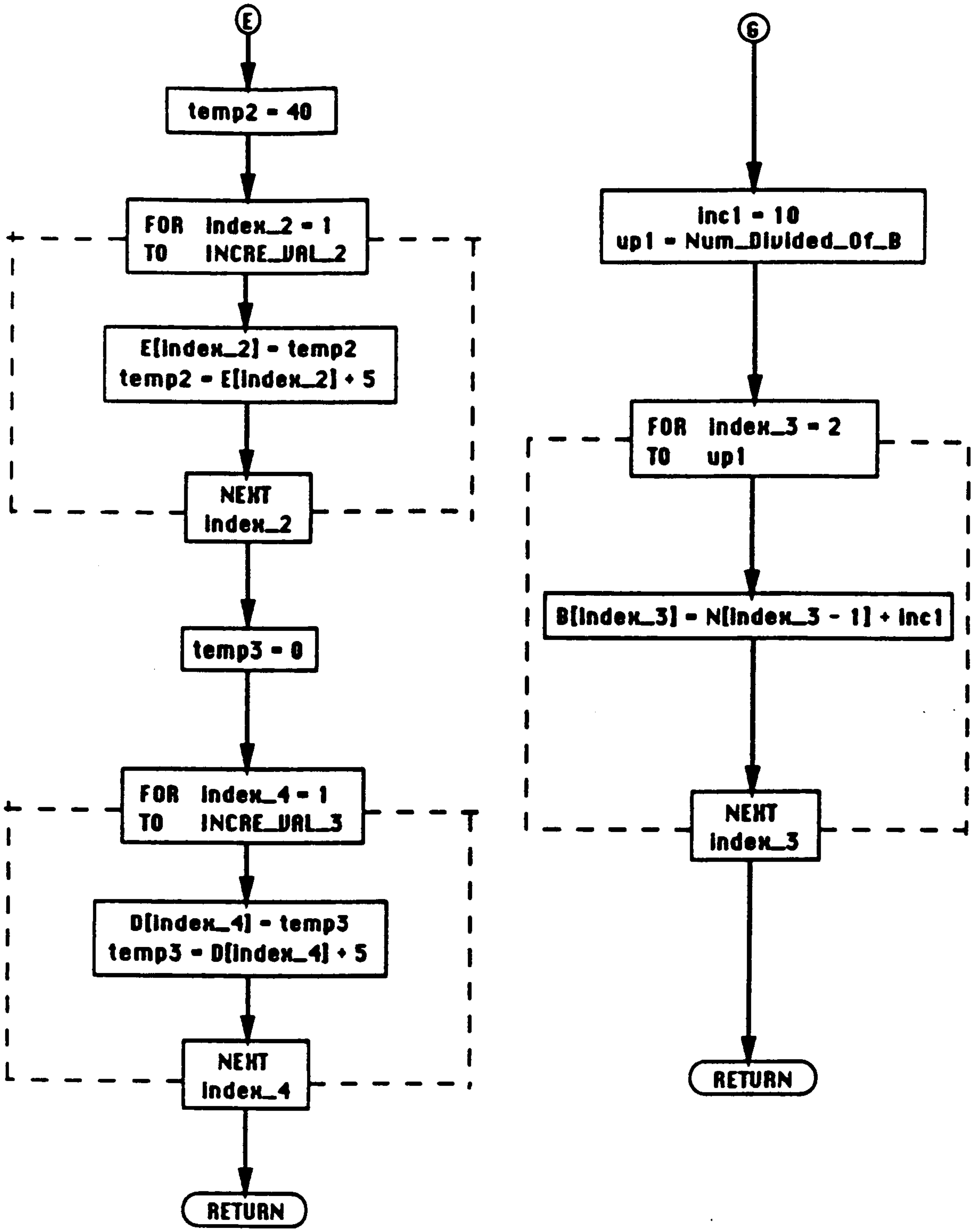


Subroutine : Get_Data_Initial_Value

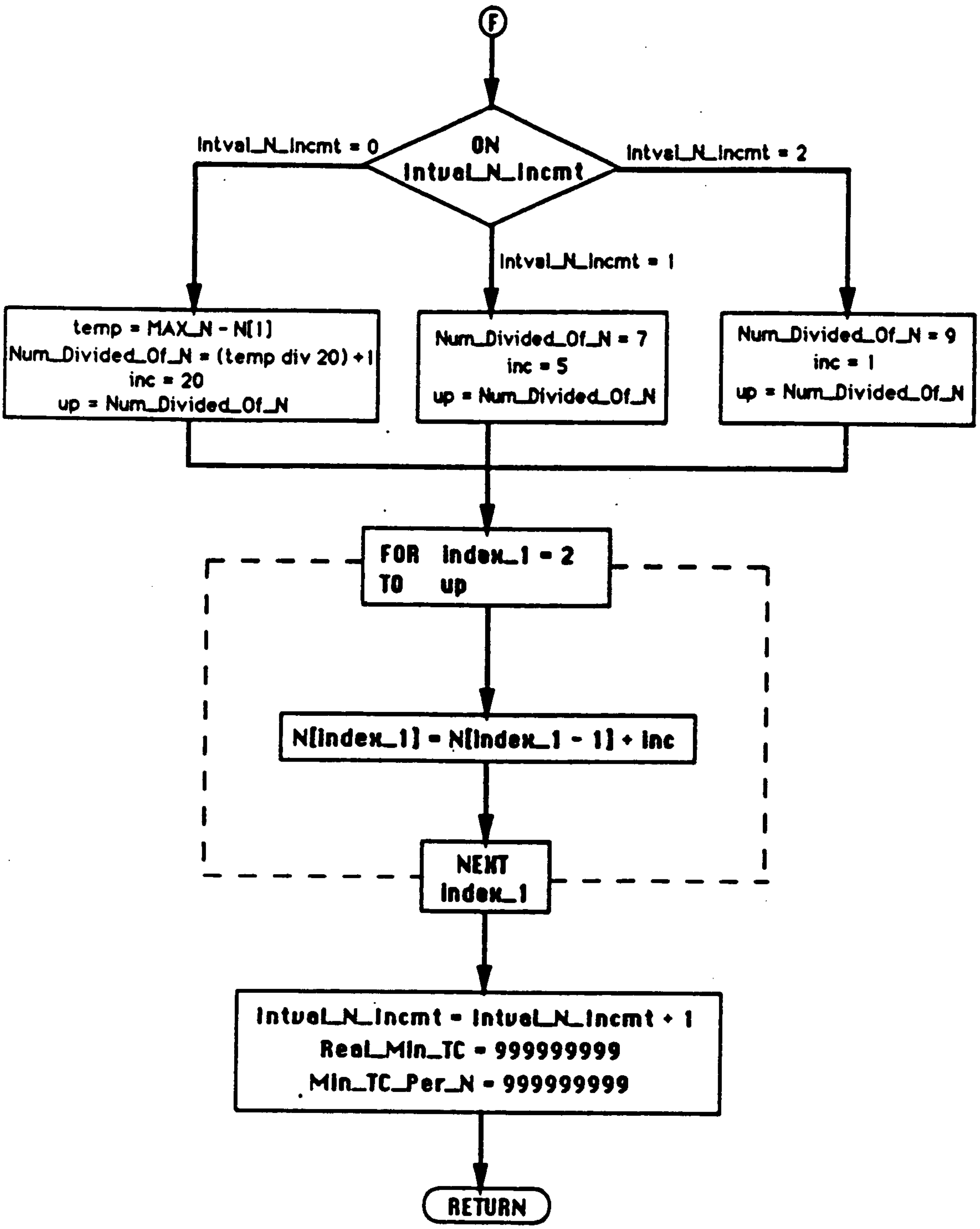


Subroutine : Initialize_Control_Param_ED

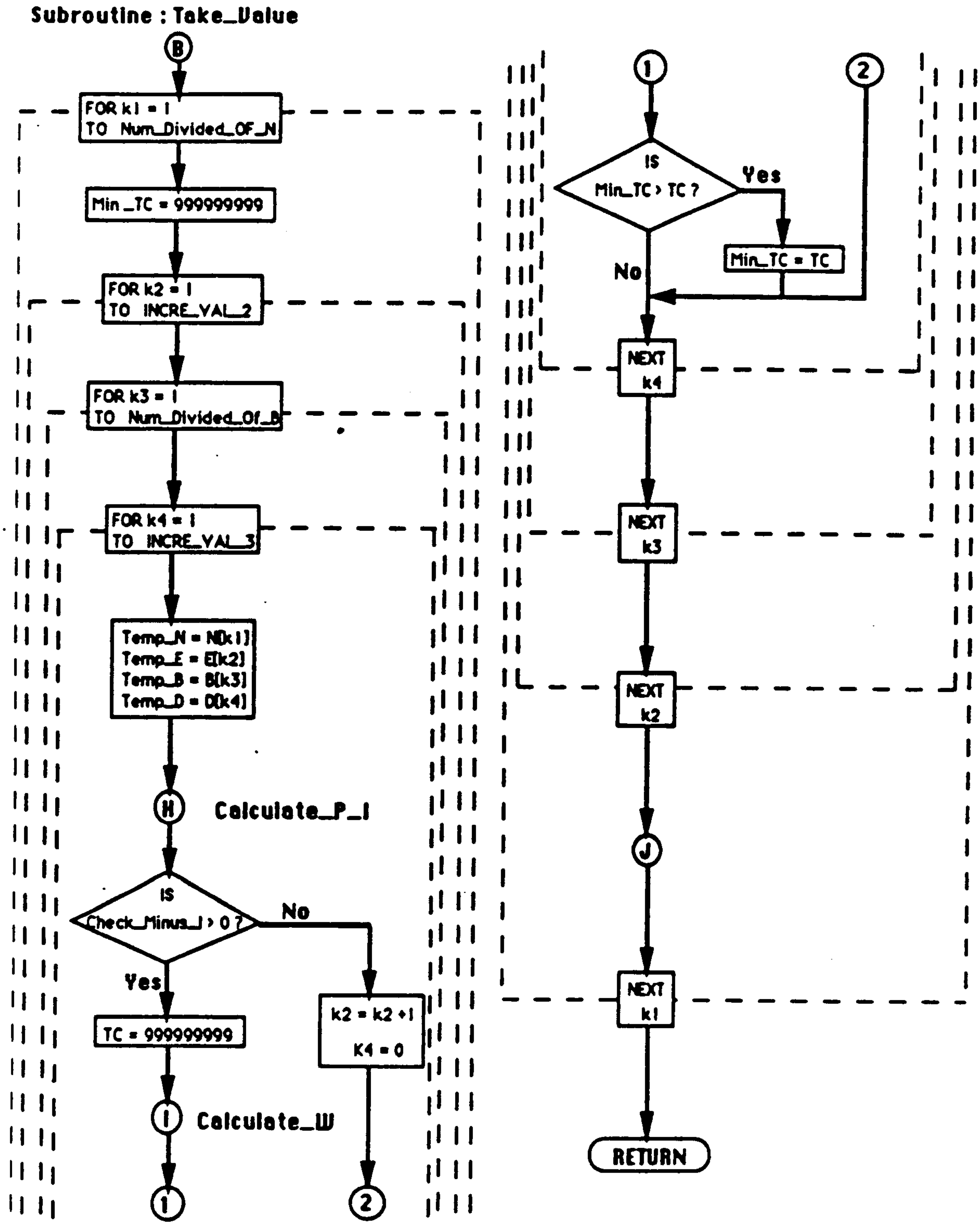
Subroutine : Initialize_Control_Param_B



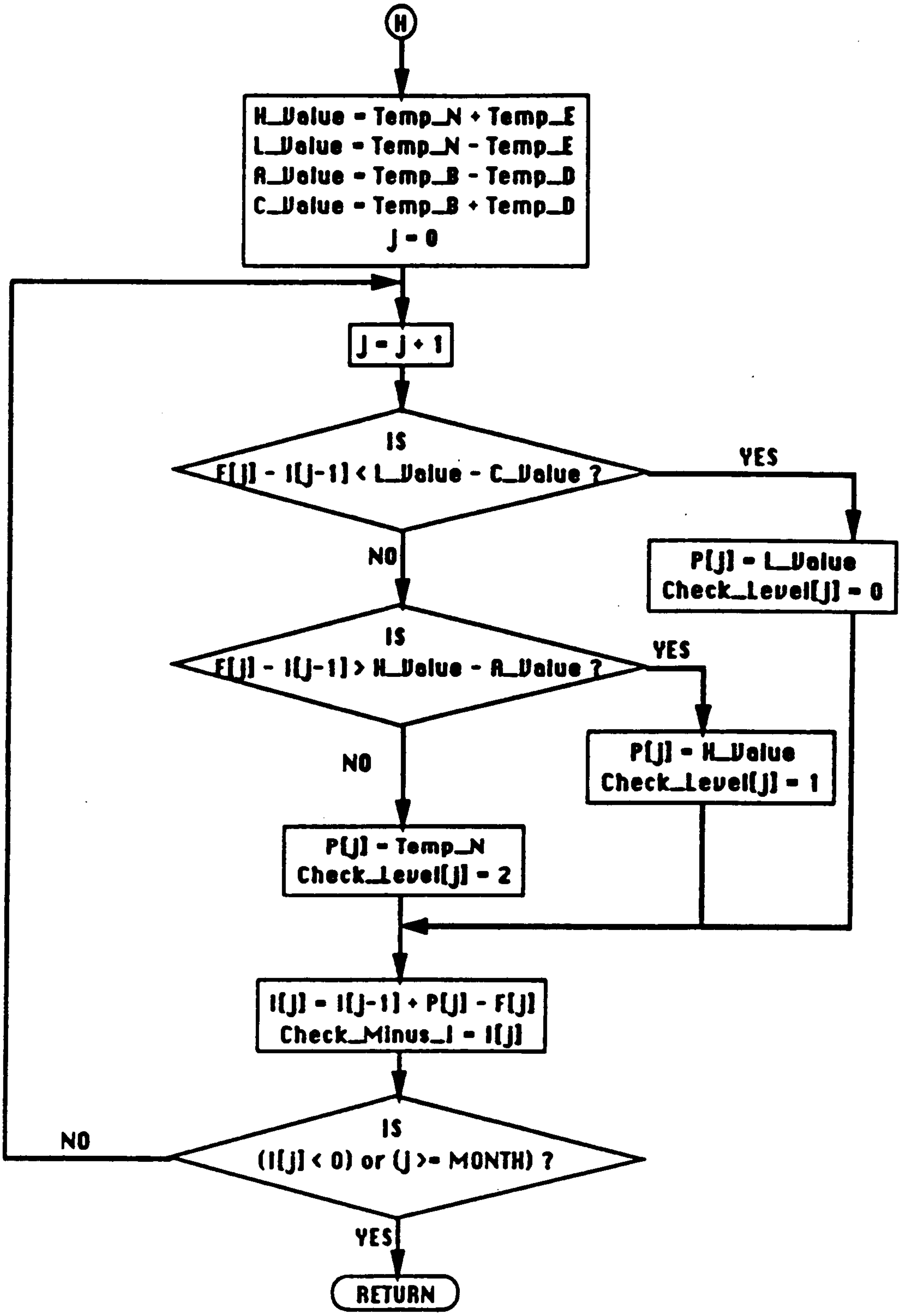
Subroutine : Initialize_Control_Param_N



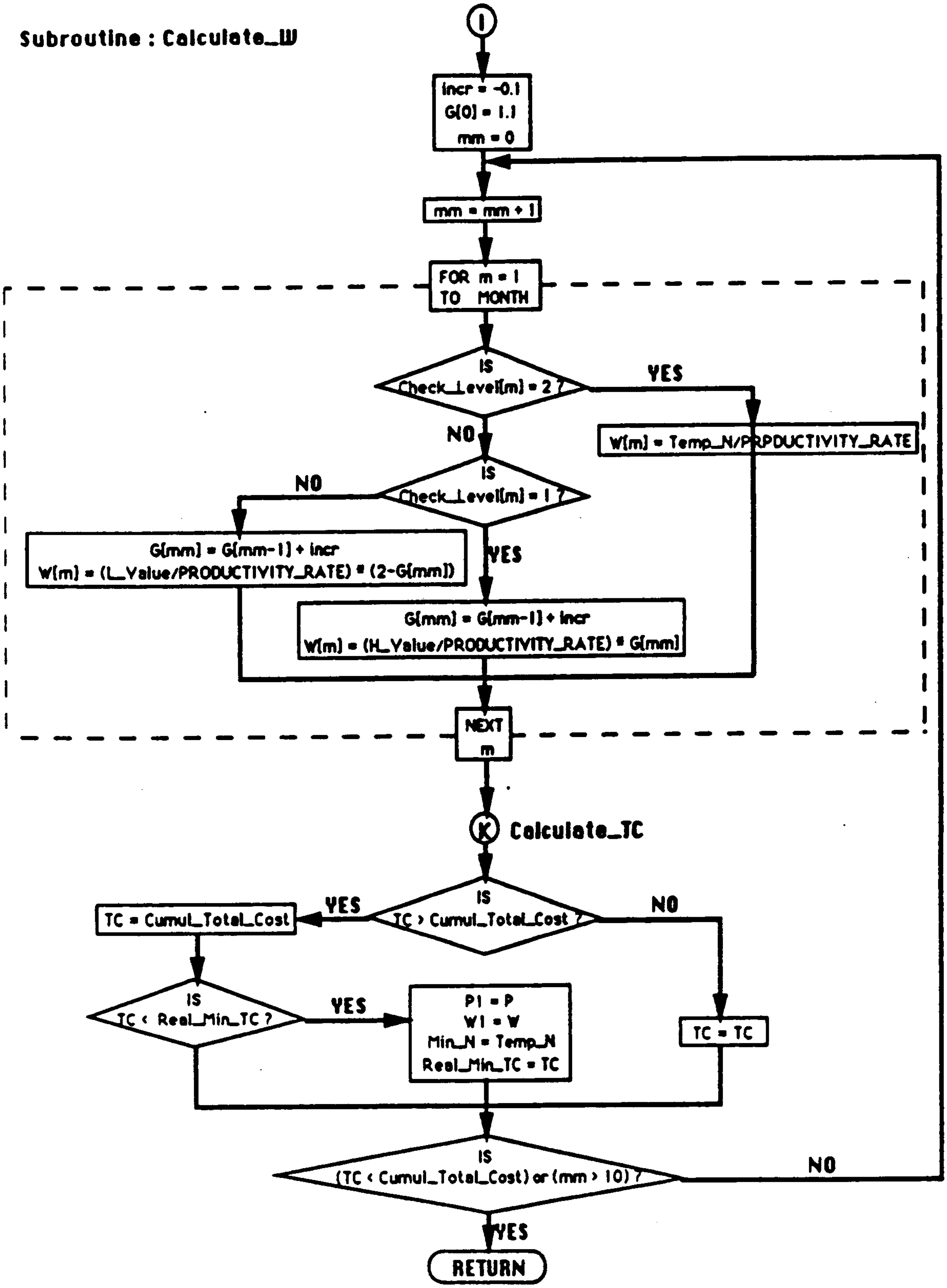
F
ON
Intval_N_Incmt
Intval_N_Incmt = 0
Intval_N_Incmt = 1
Intval_N_Incmt = 2
temp = MAX_N - N[1]
Num_Divided_Of_N = (temp div 20) + 1
inc = 20
up = Num_Divided_Of_N
Num_Divided_Of_N = 7
inc = 5
up = Num_Divided_Of_N
Num_Divided_Of_N = 9
inc = 1
up = Num_Divided_Of_N
FOR index_1 = 2
TO up
N[index_1] = N[index_1 - 1] + inc
NEXT
index_1
Intval_N_Incmt = Intval_N_Incmt + 1
Real_Min_TC = 999999999
Min_TC_Per_N = 999999999
RETURN

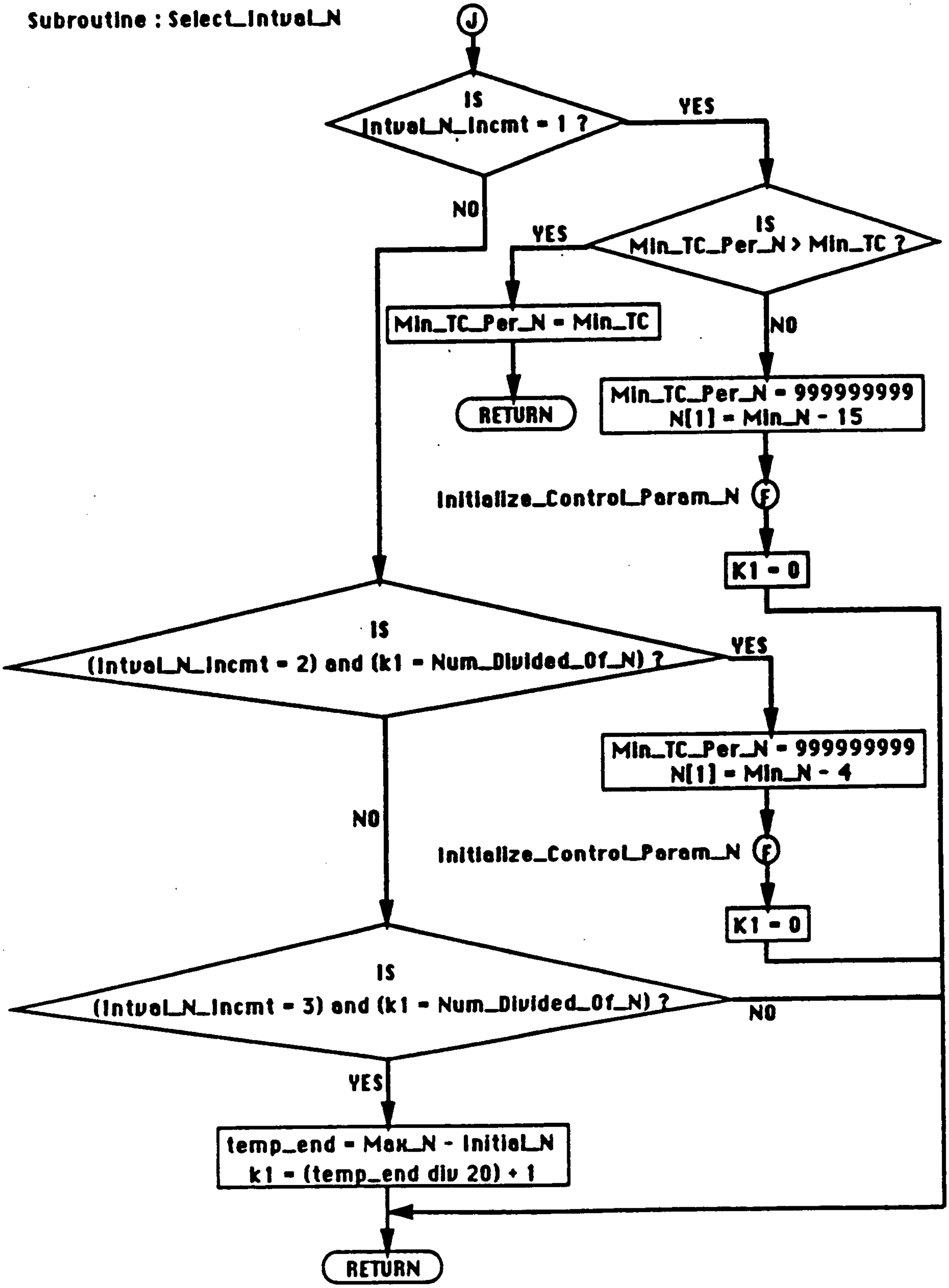

Subroutine : Take_Value
B
FOR k1 = 1 TO Num_Divided_OF_N
Min_TC = 999999999
FOR k2 = 1 TO INCRE_VAL_2
FOR k3 = 1 TO Num_Divided_Of_B
FOR k4 = 1 TO INCRE_VAL_3
Temp_N = N[k1]
Temp_E = E[k2]
Temp_B = B[k3]
Temp_D = D[k4]
H
Calculate_P_I
IS Check_Minus_I > 0 ?
Yes
No
TC = 999999999
k2 = k2 +1
K4 = 0
I
Calculate_W
1
2
1
2
IS Min_TC > TC ?
Yes
No
Min_TC = TC
NEXT k4
NEXT k3
NEXT k2
J
NEXT k1
RETURN

Subroutine : Calculate_P_I

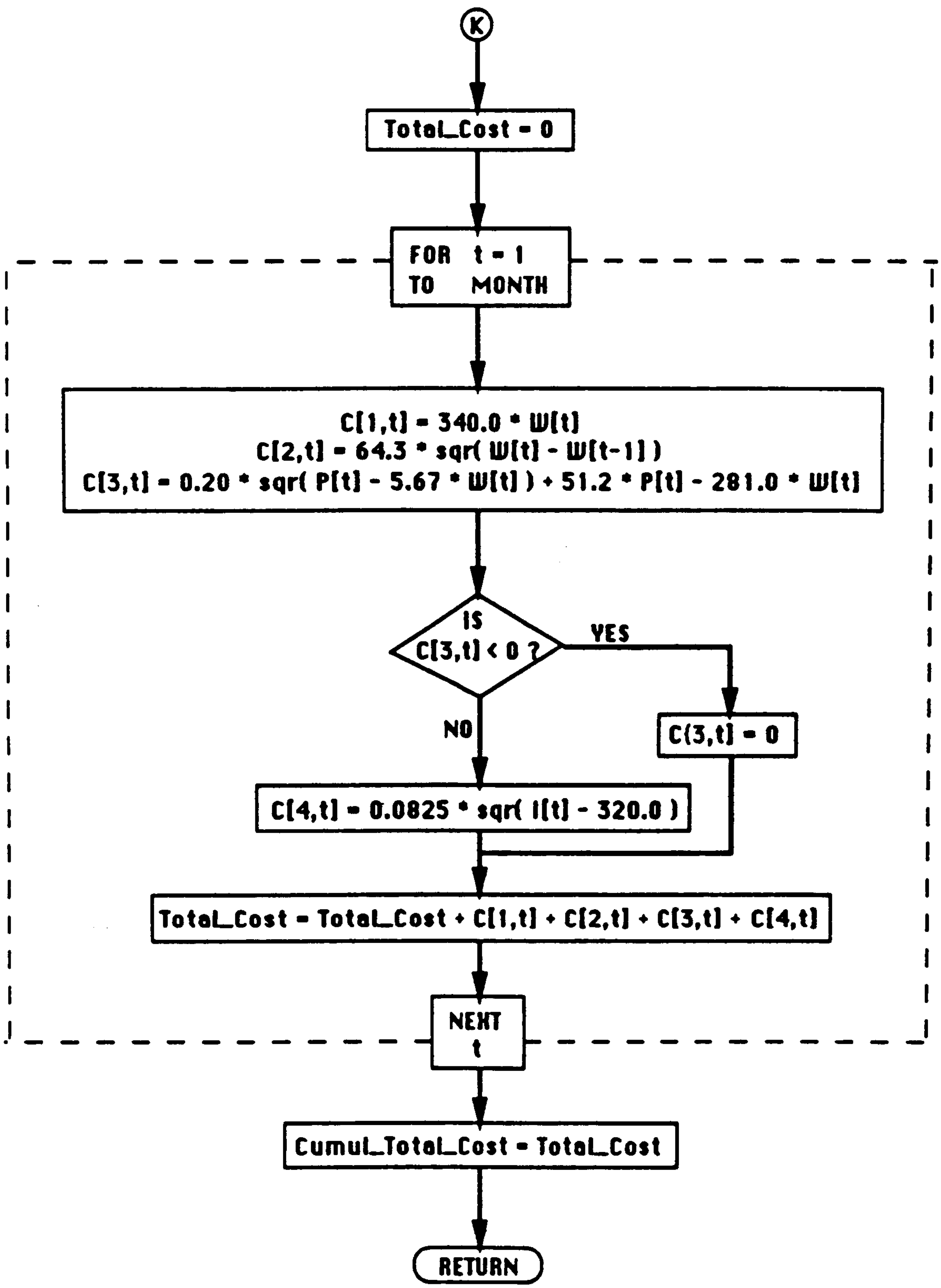


Subroutine : Calculate_W




Subroutine : Select_Intval_N
J
IS
Intval_N_Incmt = 1 ?
YES
NO
IS
Min_TC_Per_N > Min_TC ?
YES
Min_TC_Per_N = Min_TC
RETURN
NO
Min_TC_Per_N = 999999999
N[1] = Min_N - 15
Initialize_Control_Param_N
F
K1 = 0
IS
(Intval_N_Incmt = 2) and (k1 = Num_Divided_Of_N) ?
YES
Min_TC_Per_N = 999999999
N[1] = Min_N - 4
NO
Initialize_Control_Param_N
F
K1 = 0
IS
(Intval_N_Incmt = 3) and (k1 = Num_Divided_Of_N) ?
NO
YES
temp_end = Max_N - Initial_N
k1 = (temp_end div 20) + 1
RETURN

Subroutine : Calculate_TC



APPENDIX 2: Program Listing

MPSH1

```
{ Operating system and compiler used : Think Pascal.                       }
{ INPUT    : Normal Production Level Value(N) , Target Inventory Level Value(B) }
{            and Forecast values(F).                                        }
{ OUTPUT : Production values(P) and Work force values(W).                  }

Program MPSH1;

 Const
  MAX_N = 1000;               { A MAXIMUM VALUE OF A NORMAL PRODUCTION LEVEL VALUE. }
  MAX_B = 350;                { A MAXIMUM VALUE OF A TARGET INVENTORY LEVEL VALUE. }
  MONTH = 12;
  INCRE_VAL_1 = 51;         { A MAXIMUM NUMBER OF 'N' AND 'B' VALUES. }
  INCRE_VAL_2 = 11;            { NUMBERS OF 'E' VALUES. }
  INCRE_VAL_3 = 7;             { NUMBERS OF 'D' VALUES. }
  PRODUCTIVITY_RATE = 5.67;
  G_NUM_DIV = 11;             { NUMBERS OF COEFFICIENT OF 'G'. }
  NUM_OF_COST = 4;            { NUMBERS OF COSTS.}

 Type
  Array_1 = Array[1..MONTH] Of integer;
  Array_2 = Array[0..MONTH] Of real;
  Array_3 = Array[1..INCRE_VAL_1] Of integer;
  Array_4 = Array[1..INCRE_VAL_2] Of integer;
  Array_5 = Array[0..G_NUM_DIV] Of real;
  Array_6 = Array[1..NUM_OF_COST, 1..MONTH] Of real;

 Var
  F, P, P1, Check_Level, Pri: Array_1;
  W, W1, I: Array_2;
  N, B: Array_3;
  E, D: Array_4;
  G: Array_5;
  C: Array_6;
  time_in, time_out: longint;
  Total_Cost, Cumul_Total_Cost, TC, Min_TC, Real_Min_TC, Min_TC_Per_N, Check_Minus_I: real;
  Intval_B, Intval_N_Incmt, Num_Divided_Of_N, Num_Divided_Of_B: integer;
  H_Value, L_Value, A_Value, C_Value: integer;
  Initial_N, Min_N, Temp_N, Temp_E, Temp_B, Temp_D: integer;

{-----------------------------------------------------------------------------}
{ Procedure name : Get_Data_Forecast                   }
{ Purpose : To take Forecast values.                   }
{ Input variables: None.                                                      }
{ Output variables: None.                                                     }
{ Globals: F.                                }

 Procedure Get_Data_Forecast;
 Begin
  F[1] := 420;
  F[2] := 379;
  F[3] := 403;
  F[4] := 371;
  F[5] := 388;
```

MPSH1

```
  F[6] := 368;
  F[7] := 433;
  F[8] := 324;
  F[9] := 314;
  F[10] := 422;
  F[11] := 336;
  F[12] := 379;
 End; { procedure Get_Data_Forecast }

{-----------------------------------------------------------------------------}
{ Procedure name : Get_Data_Initial_Value                          }
{ Purpose : To get the initial value of  Work Force and Inventory.          }
{ Input variables: None.                                                     }
{ Output variables: None.                                                    }
{ Globals: F.                                                  }

 Procedure Get_Data_Initial_Value;
 Begin
  W[0] := 81;
  I[0] := 263;
 End;{ procedure Get_Data_Initial_Value  }

{-----------------------------------------------------------------------------}
{ Procedure name : Initialize_Control_Param_ED                          }
{ Purpose : To get the initial value of  High and Low Production Level(E) and  }
{            Maximum and Minimum Acceptable Inventory Level(B)              .}
{ Input variables: None.                                                      }
{ Output variables: None.                                                     }
{ Globals: E, D, INCRE_VAL_2, and INCRE_VAL_3.                          }

 Procedure Initialize_Control_Param_ED;
  Var
   temp2, temp3, index_2, index_4: integer;
 Begin
  temp2 := 40;
  For index_2 := 1 To INCRE_VAL_2 Do
   Begin
    E[index_2] := temp2;            { TO ASSIGN 'E' VALUES. }
    temp2 := E[index_2] + 5;
   End; {for index_2 := 1 to INCRE_VAL_2 do}
  temp3 := 0;
  For index_4 := 1 To INCRE_VAL_3 Do
   Begin
    D[index_4] := temp3;            { TO ASSIGN 'D' VALUES. }
    temp3 := D[index_4] + 5;
   End;
 End;   { procedure Initialize_Control_Param_ED  }

{-----------------------------------------------------------------------------}
{ Procedure name : Initialize_Control_Param_N                          }
{ Purpose : To get the initial value of  Normal Production Level(E).          }
{ Input variables: None.                                                      }
{ Output variables: None.                                                     }
```

MPSH1

```
{ Globals: Intval_N_Incmt, N, Num_Divided_Of_N, Real_Min_TC and Min_TC_Per_N.        }

 Procedure  Initialize_Control_Param_N;
  Var
   index_1, inc, up, temp: integer;
 Begin
  Case Intval_N_Incmt Of
   0:                          { WHEN THE INCREMENTAL VALUE IS 20. }
    Begin
     temp := MAX_N - N[1];
     Num_Divided_Of_N := (temp Div 20) + 1;   { TO CALCULATE NUMBERS OF 'N' VALUES.}
     inc := 20;
     up := Num_Divided_Of_N;
    End;  {CASE 0}
   1:                          { WHEN THE INCREMENTAL VALUE IS 5. }
    Begin
     Num_Divided_Of_N := 7;
     inc := 5;
     up := Num_Divided_Of_N;
    End;  {CASE 1}
   2:                          { WHEN THE INCREMENTAL VALUE IS 5. }
    Begin
     Num_Divided_Of_N := 9;
     inc := 1;
     up := Num_Divided_Of_N;
    End;  {CASE 2}
  End;  {case Intval_N_Incmt of}
  For index_1 := 2 To up Do
   N[index_1] := N[index_1 - 1] + inc;          { TO ASSIGN 'N' VALUES. }
  Intval_N_Incmt := Intval_N_Incmt + 1;        { TO INCREMENT THE INTERVAL OF 'N' VALUES. }
  Real_Min_TC  := 999999999;               { TO  MAKE A MAXIMUM NUMBER. }
  Min_TC_Per_N  := 999999999;
 End;  {procedure Initialize_Control_Param_N}

{---------------------------------------------------------------------------}
{ Procedure name : Initialize_Control_Param_B                          }
{ Purpose : To get the initial value of Target Inventory Level(B).             }
{ Input variables: None.                                                   }
{ Output variables: None.                                                  }
{ Globals: Num_Divided_Of_B and B                                }

 Procedure  Initialize_Control_Param_B;
  Var
   inc1, up1, index_3: integer;
 Begin
  inc1  := 10;
  up1  := Num_Divided_Of_B;
  For index_3 := 2 To up1 Do
   B[index_3]  := B[index_3 - 1] + inc1;      { TO ASSIGN 'B' VALUES. }
 End;  {procedure Initialize_Control_Param_B}

{---------------------------------------------------------------------------}
{ Procedure name : Start_Calculation                              }
```

MPSH1

```
{ Purpose : To take input data and arrange input data.                    }
{ Input variables: None.                                                   }
{ Output variables: None.                                                  }
{ Globals: Initial_N, N, Intval_N_Incmt, B, Intval_B,Num_Divided_Of_B.     }

 Procedure Start_Calculation;
 Begin
  Get_Data_Forecast;                { CALL PROCEDURE Get_Data_Forecast. }
  Get_Data_Initial_Value;           { CALL PROCEDURE Get_Data_Initial_Value. }
  Initialize_Control_Param_ED;      { CALL PROCEDURE Initialize_Control_Param_ED. }
  Initial_N := 340;                 { TO INITIALIZE A NORMAL VALUE. }
  N[1] := Initial_N;                { TO ASSIGN A FIRST NORMAL VALUE. }
  Intval_N_Incmt := 0;              { TO INITIALIZE A INCREMENT OF INTERVAL OF A NORMAL VALUE. }
  Initialize_Control_Param_N;       { CALL PROCEDURE Initialize_Control_Param_N. }
  B[1] := 250;                      { TO INITIALIZE A DESIRED INVENTORY VALUE. }
  Intval_B := MAX_B - B[1];
  Num_Divided_Of_B := (Intval_B Div 10) + 1;    { NUMBERS OF INTERVAL OF A DESIRED INVENTORY
      VALUE. }
  Initialize_Control_Param_B;       { CALL PROCEDURE Initialize_Control_Param_B. }
 End;  {procedure Start_Calculation}

{-----------------------------------------------------------------------------}
{ Procedure name : Calculate_P_I                                       }
{ Purpose : To calculate Production Values and Inventory Values.       }
{ Input variables: None.                                               }
{ Output variables: None.                                              }
{ Globals: H_Value, L_Value, A_Value, C_Value, Temp_N, Temp_E, Temp_B, Temp_D   }
{           F, I, P, Check_Level, Check_Minus_I                        }

 Procedure Calculate_P_I;
  Var
   j: integer;
 Begin
  H_Value := Temp_N + Temp_E;       { TO ASSIGN THE HIGH PRODUCTION LEVEL VALUE. }
  L_Value := Temp_N - Temp_E;       { TO ASSIGN THE LOW PRODUCTION LEVEL VALUE. }
  A_Value := Temp_B - Temp_D;       { TO ASSIGN THE MINIMUM ACCEPTABLE INVENTORY LEVEL. }
  C_Value := Temp_B + Temp_D;       { TO ASSIGN THE MAXIMUM ACCEPTABLE INVENTORY LEVEL. }
  j := 0;
  Repeat
   j := j + 1;
   If (F[j] - I[j - 1] < L_Value - C_Value) Then     { THIS IS THE CONDITION OF THE LOW PRODUCTION
      LEVEL. }
    Begin
     P[j] := L_Value;               { THE PRODUCTION VALUE IS THE LOW PRODUCTION LEVEL. }
     Check_Level[j] := 0;
    End {if (F[j] - I[j - 1] < L_Value - C_Value) then}
   Else If (F[j] - I[j - 1] > H_Value - A_Value) Then  { THIS IS THE CONDITION OF THE HIGH PRODUCTION
      LEVEL. }
    Begin
     P[j] := H_Value;               { THE PRODUCTION VALUE IS THE HIGH PRODUCTION LEVEL. }
     Check_Level[j] := 1;
    End   {else if (F[j] - I[j - 1] > H_Value - A_Value) then}
   Else
```

MPSH1

```
    Begin                          { OTHERWISE, THE PRODUCTION VALUE GETS THE NORMAL LEVEL. }
     P[j] := Temp_N;
     Check_Level[j] := 2;
    End;    {else}
   I[j] := I[j - 1] + P[j] - F[j];        { TO CALCULATE THE INVENTORY VALUE. }
   Check_Minus_I := I[j];                 { TO CHECK THE INVENTORY VALUE IF NEGATIVE OR NOT. }
  Until (I[j] < 0) Or (j >= MONTH);    { UNTIL THE INVENTORY VALUE IS NEGATIVE OR OVER THE 12
      MONTHS. }
 End;  { procedure Calculate_P_I }

{--------------------------------------------------------------------------------}
{ Procedure name : Calculate_TC                                   }
{ Purpose : To calculate Total Costs.                               }
{ Input variables: None.                                                          }
{ Output variables: None.                                                         }
{ Globals: Total_Cost, C, W, P, I, and Cumul_Total_Cost                     }

 Procedure Calculate_TC;
  Var
   t: integer;
 Begin
  Total_Cost := 0;                  { TO INITIALIZE THE TOTAL COST. }
  For t := 1 To MONTH Do
   Begin
    C[1, t] := 340.0 * W[t];                                { REGULAR PAYROLL. }
    C[2, t] := 64.3 * sqr(W[t] - W[t - 1]);                        { HIRING AND LAYOFF. }
    C[3, t] := 0.20 * sqr(P[t] - 5.67 * W[t]) + 51.2 * P[t] - 281.0 * W[t]; { OVER TIME. }
    If C[3, t] < 0 Then              { IF THE OVER TIME IS NEGATIVE THEN THE OVER TIME IS ZERO. }
     C[3, t] := 0;
    C[4, t] := 0.0825 * sqr(I[t] - 320.0);                  { INVENTORY COST. }
    Total_Cost := Total_Cost + C[1, t] + C[2, t] + C[3, t] + C[4, t];{ TO CUMULATE TOTAL COSTS. }
   End; { for t := 1 to MONTH do }
  Cumul_Total_Cost := Total_Cost;
 End;  {procedure Calculate_TC}

{--------------------------------------------------------------------------------}
{ Procedure name : Calculate_W                                    }
{ Purpose : To calculate Value of Work Force.                         }
{ Input variables: None.                                                          }
{ Output variables: None.                                                         }
{ Globals: G, Check_Level, W, Temp_N, H_Value, L_Value, TC,Cumul_Total_Cost         }
{       Real_Min_TC, P, P1, W, W1, and Min_N                           }

 Procedure Calculate_W;
  Var
   m, mm: integer;
   incr: real;
 Begin
  incr := -0.1;          { TO DECREASE THE OVERTIME AND REGULAR PRODUCTION VALUE(G). }
  G[0] := 1.1;           { TO INITIALIZE THE 'G' VALUE. }
  mm := 0;               { THE NUMBER OF 'G' VALUE IS FROM ZERO TO ELEVEN. }
  Repeat
   mm := mm + 1;
```

MPSH1

```
  For m := 1 To MONTH Do
   Begin
    If Check_Level[m] = 2 Then        { IF THE PRODUCTION VALUE IS A NORMAL VALUE... }
      Begin
       W[m] := Temp_N / PRODUCTIVITY_RATE;       { TO CALCULATE THE VALUE OF WORK FORCE. }
      End    {if Check_Level[m] = 2 then}
    Else If Check_Level[m] = 1 Then    {IF THE PRODUCTION VALUE IS A HIGH LEVEL... }
      Begin
       G[mm] := G[mm - 1] + incr;
       W[m] := (Temp_N / PRODUCTIVITY_RATE) + ((H_Value - Temp_N) / PRODUCTIVITY_RATE) *
      G[mm];
                                     { TO CALCULATE THE VALUE OF WORK FORCE. }
      End    {else if Check_Level [m]= 1 then}
    Else                          { IF THE PRODUCTION VALUE IS A LOW LEVEL... }
      Begin
       G[mm] := G[mm - 1] + incr;
       W[m] := (Temp_N / PRODUCTIVITY_RATE) - ((Temp_N - L_Value) / PRODUCTIVITY_RATE) *
      G[mm];
                                     { TO CALCULATE THE VALUE OF WORK FORCE. }
      End;
   End;   {for m := 1 to MONTH do}

  Calculate_TC;                  { CALL PROCEDURE Calculate_TC. }

  If (TC > Cumul_Total_Cost) Then        { IF NEW TOTAL COST IS SMALLER THEN OLD TOTAL COST... }
   Begin
    TC := Cumul_Total_Cost;
    If TC < Real_Min_TC Then       { IF NEW TOTAL COST IS SMALLER THEN THE REAL MINIMUM TOTAL
     COST... }
     Begin
      P1 := P;                { GET PRODUCTION VALUES TO WRITE THE OUTPUT. }
      W1 := W;                { GET VALUES OF WORK FORCE TO WRITE THE OUTPUT. }
      Min_N := Temp_N;             { ASSIGN THE MINIMUM 'N' VALUE WHEN WE GET THE REAL MINIMUM
     TOTAL COST. }
      Real_Min_TC := TC            { TO ASSIGN THE REAL MINIMUM TOTAL COST. }
     End;
   End
  Else
   TC := TC;
 Until (TC < Cumul_Total_Cost) Or (mm > 10);{ UNTIL NEW TOTAL COST IS BIGGER THEN THE OLD ONE OR
     OVER 11 OF 'G'}
End;  {Calculate_W }

{-------------------------------------------------------------------------------}
{ Procedure name : Select_Intval_N                                          }
{ Purpose : To decide the interval of 'N' value.                              }
{ Input variables: k_First.                                                      }
{ Output variables: k_First.                                                     }
{ Globals:  Num_Divided_Of_N, Min_TC  , N, Intval_N_Incmt, Min_TC_Per_N, and Initial_N   }
 Procedure Select_Intval_N (Var k_First: integer);
  Var
   temp_end: integer;
 Begin
```

MPSH1

```
If Intval_N_Incmt = 1 Then          { IF THE INTERVAL OF 'N' VALUE IS 20 THEN... }
 Begin
  If Min_TC_Per_N > Min_TC Then     { UNTIL GET THE MINIMUM TOTAL COST, WHEN THE INTERVAL IS
    20. }
   Begin
    Min_TC_Per_N := Min_TC
   End
  Else                    { WHEN GET THE MINIMUM TOTAL COST, WHEN THE INTERVAL IS 20. }
   Begin
    Min_TC_Per_N := 999999999; { TO ASSIGN THE NEW TOTAL COST PER EACH INTERVAL TO THE
    LARGEST NUMBER. }
    N[1] := Min_N - 15;          { TO DECIDE THE FIRST NUMBER OF 'N' VALUE, WHEN THE INTERVAL
    IS 5. }
    Initialize_Control_Param_N;     { CALL PROCEDURE Initialize_Control_Param_N }
    k_First := 0;              { TO RESET THE 'D' VALUE FROM ZERO. }
   End;
 End {if Intval_N_Incmt = 1 then}
Else If (Intval_N_Incmt = 2) And (k_First = Num_Divided_Of_N) Then   {THE END OF LOOPING WHEN
    THE INTERVAL IS 5. }
 Begin
  Min_TC_Per_N := 999999999;
  N[1] := Min_N - 4;            { TO DECIDE THE FIRST NUMBER OF 'N' VALUE, WHEN THE INTERVAL IS
    1. }
  Initialize_Control_Param_N;      { CALL PROCEDURE Initialize_Control_Param_N }
  k_First := 0;
 End    {else if Intval_N_Incmt = 2 then}
Else If (Intval_N_Incmt = 3) And (k_First = Num_Divided_Of_N) Then   {THE END OF LOOPING WHEN
    THE INTERVAL IS 1. }
 Begin
  temp_end := MAX_N - Initial_N;
  k_First := (temp_end Div 20) + 1; { TO ASSIGN THE LOOPING PARAMETER OF 'N' VALUE TO FINISH
    THE TOTAL LOOPING. }
 End;   {else if (Intval_N_Incmt = 3) and (k_First = Num_Divided_Of_N) then}
End;   {procedure Take_Value}

{--------------------------------------------------------------------------}
{ Procedure name : Take_Value                                   }
{ Purpose : To arrange Values of 'N', 'E', 'B', and 'D'.                  }
{ Input variables: None.                                                 }
{ Output variables: None.                                                }
{ Globals:  Num_Divided_Of_N, Min_TC  , INCRE_VAL_2, Num_Divided_Of_B, INCRE_VAL_3,   }
{          Temp_N, Temp_E, Temp_B, Temp_D, N, E, B, D, Check_Minus_I, and TC          }
 Procedure Take_Value;
  Var
   k1, k2, k3, k4: integer;
 Begin
  For k1 := 1 To Num_Divided_Of_N Do      { LOOPING FOR NUMBERS OF INTERVALS OF 'N' VALUES. }
   Begin
    Min_TC := 999999999;          { TO ASSIGN THE TEMPORARY MINIMUM TOTAL COST TO THE
      LARGEST NUMBER. }
    For k2 := 1 To INCRE_VAL_2 Do        { LOOPING FOR NUMBERS OF INERVALS OF 'E' VALUES. }
     Begin
      For k3 := 1 To Num_Divided_Of_B Do   { LOOPING FOR NUMBERS OF INTERVALS OF 'B' VALUES. }
```

MPSH1

```
        Begin
         For k4 := 1 To INCRE_VAL_3 Do     { LOOPING FOR NUMBERS OF INTERVALS OF 'D' VALUES. }
          Begin
           Temp_N := N[k1];          { TO ASSIGN THE TEMPORARY 'N' VALUE. }
           Temp_E := E[k2];
           Temp_B := B[k3];
           Temp_D := D[k4];
           Calculate_P_I;            { CALL PROCEDURE Calculate_P_I }
           If Check_Minus_I > 0 Then      { IF INVENTORY VALUE IS POSITIVE THEN ... }
            Begin
             TC := 999999999;        { TO ASSIGN THE NEW TOTAL COST TO THE LARGEST NUMBER. }
             Calculate_W;            { CALL PROCEDURE Calculate_W }
             If Min_TC > TC Then       { IF THE NEW TOTAL COST IS SMALLER THEN THE OLD ONE ... }
              Begin
               Min_TC := TC;         { THE OLD ONE GETS THE NEW TOTAL COST. }
              End
            End  {if Check_Minus_I > 0 then}
           Else                      { IF INVENTORY VALUE IS NOT POSITIVE THEN ... }
            Begin
             k2 := k2 + 1;           { TO JUMP NEXT 'E' VALUE. }
             k4 := 0;                { TO RESET THE 'D' VALUE FROM ZERO. }
            End; {else}
          End;  {  for k4:=1 to INCRE_VAL_2 do  }
        End; {for k3 := 1 to Num_Divided_Of_B + 1 do   }
      End;  { for k2:=1 to INCRE_VAL_2  do  }
     Select_Intval_N(k1);
    End;   { for k1 := 1 to Num_Divided_Of_N do  }
  End;    {procedure Take_Value}

{---------------------------------------------------------------------------}
{ Procedure name : Formating_Output                              }
{ Purpose : To write the output.                          }
{ Input variables: None.                                                    }
{ Output variables: None.                                                   }
{ Globals: Pri, Real_Min_TC, F, P, P1, and W1                    }
  Procedure Formating_Output;
   Var
    ii, iii: integer;
  Begin
   For iii := 1 To 12 Do
    Begin
     Pri[iii] := iii;
    End;
   writeln('Minmum ToTal Cost is : ', Real_Min_TC : 12 : 8);
   writeln;
   writeln;
   writeln('______________________________________________________________');
   writeln('   Period          Forecast          Production          Workforce   ');
   writeln('______________________________________________________________');
   For ii := 1 To 12 Do
    Begin
     write(Pri[ii] : 6);            { TO WRITE NUMBERS OF MONTH. }
     write(F[ii] : 15);               { TO WRITE FORECAST PRODUCTION VALUES. }
```

MPSH1

```
    write(P1[ii] : 15);              { TO WRITE PRODUCTION VALUES. }
    writeln(W1[ii] : 20 : 8);          { TO WRITE VALUES OF WORK FORCE. }
   End;
 End;   { procedure Formating_Output}

Begin      {  MAIN  }
 showtext;
 time_in := TickCount;               { TO ESTIMATE A START CPU TIME. }
 Start_Calculation;                 { CALL PROCEDURE Start_Calculation. }
 Take_Value;                       { CALL PROCEDURE Take_Value. }
 time_out := TickCount;              { TO ESTIMATE A END CPU TIME. }
 writeln('Total CPU Time (in hours) : ', abs(time_out - time_in) / 216000 : 20 : 10); { TO WRITE A CPU
       TIME. }
 Formating_Output;                   { CALL PROCEDURE Formating_Output. }
End.    {  MAIN  }
```

MPSH2

```
{ Operating system and compiler used : Think Pascal.                          }
{ INPUT    : Normal Production Level Value(N) , Target Inventory Level Value(B) }
{            and Forecast values(F).                                           }
{ OUTPUT : Production values(P) and Work force values(W).                      }

Program MPSH2;

 Const
  MAX_N = 1000;              { A MAXIMUM VALUE OF A NORMAL PRODUCTION LEVEL VALUE. }
  MAX_B = 350;               { A MAXIMUM VALUE OF A TARGET INVENTORY LEVEL VALUE. }
  MONTH = 12;
  INCRE_VAL_1 = 51;         { A MAXIMUM NUMBER OF 'N' AND 'B' VALUES. }
  INCRE_VAL_2 = 11;            { NUMBERS OF 'E'  VALUES. }
  INCRE_VAL_3 = 7;             { NUMBERS OF 'D'  VALUES. }
  PRODUCTIVITY_RATE = 5.67;
  G_NUM_DIV = 11;              { NUMBERS OF COEFFICIENT OF 'G'. }
  NUM_OF_COST = 4;             { NUMBERS OF COSTS.}

 Type
  Array_1 = Array[1..MONTH] Of integer;
  Array_2 = Array[0..MONTH] Of real;
  Array_3 = Array[1..INCRE_VAL_1] Of integer;
  Array_4 = Array[1..INCRE_VAL_2] Of integer;
  Array_5 = Array[0..G_NUM_DIV] Of real;
  Array_6 = Array[1..NUM_OF_COST, 1..MONTH] Of real;

 Var
  F, P, P1, Check_Level, Pri: Array_1;
  W, W1, I: Array_2;
  N, B: Array_3;
  E, D: Array_4;
  G: Array_5;
  C: Array_6;
  time_in, time_out: longint;
  Total_Cost, Cumul_Total_Cost, TC, Min_TC, Real_Min_TC, Min_TC_Per_N, Check_Minus_I: real;
  Intval_B, Intval_N_Incmt, Num_Divided_Of_N, Num_Divided_Of_B: integer;
  H_Value, L_Value, A_Value, C_Value: integer;
  Initial_N, Min_N, Temp_N, Temp_E, Temp_B, Temp_D: integer;

{----------------------------------------------------------------------------}
{ Procedure name : Get_Data_Forecast                      }
{ Purpose : To take Forecast values.                      }
{ Input variables: None.                                                      }
{ Output variables: None.                                                     }
{ Globals: F.                                   }

 Procedure Get_Data_Forecast;
 Begin
  F[1] := 430;
  F[2] := 447;
  F[3] := 440;
  F[4] := 316;
  F[5] := 397;
```

MPSH2

```
 F[6] := 375;
 F[7] := 292;
 F[8] := 458;
 F[9] := 400;
 F[10] := 350;
 F[11] := 284;
 F[12] := 400;
End; { procedure Get_Data_Forecast }

{-------------------------------------------------------------------------}
{ Procedure name : Get_Data_Initial_Value                    }
{ Purpose : To get the initial value of  Work Force and Inventory.      }
{ Input variables: None.                                                }
{ Output variables: None.                                               }
{ Globals: F.                                           }

 Procedure Get_Data_Initial_Value;
 Begin
  W[0] := 81;
  I[0] := 263;
 End;{ procedure Get_Data_Initial_Value }

{-------------------------------------------------------------------------}
{ Procedure name : Initialize_Control_Param_ED                     }
{ Purpose : To get the initial value of High and Low Production Level(E) and   }
{            Maximum and Minimum Acceptable Inventory Level(B)          .}
{ Input variables: None.                                                }
{ Output variables: None.                                               }
{ Globals: E, D, INCRE_VAL_2, and INCRE_VAL_3.                      }

 Procedure Initialize_Control_Param_ED;
  Var
   temp2, temp3, index_2, index_4: integer;
 Begin
  temp2 := 40;
  For index_2 := 1 To INCRE_VAL_2 Do
   Begin
    E[index_2] := temp2;          { TO ASSIGN 'E' VALUES. }
    temp2 := E[index_2] + 5;
   End; {for index_2 := 1 to INCRE_VAL_2 do}
  temp3 := 0;
  For index_4 := 1 To INCRE_VAL_3 Do
   Begin
    D[index_4] := temp3;          { TO ASSIGN 'D' VALUES. }
    temp3 := D[index_4] + 5;
   End;
 End;   { procedure Initialize_Control_Param_ED }

{-------------------------------------------------------------------------}
{ Procedure name : Initialize_Control_Param_N                   }
{ Purpose : To get the initial value of Normal Production Level(E).     }
{ Input variables: None.                                                }
{ Output variables: None.                                               }
```

MPSH2

```
{ Globals: Intval_N_Incmt, N, Num_Divided_Of_N, Real_Min_TC and Min_TC_Per_N.        }

 Procedure  Initialize_Control_Param_N;
  Var
   index_1, inc, up, temp: integer;
 Begin
  Case Intval_N_Incmt Of
   0:                          { WHEN THE INCREMENTAL VALUE IS 20. }
    Begin
     temp := MAX_N - N[1];
     Num_Divided_Of_N := (temp Div 20) + 1;  { TO CALCULATE NUMBERS OF 'N' VALUES.}
     inc := 20;
     up := Num_Divided_Of_N;
    End;  {CASE 0}
   1:                          { WHEN THE INCREMENTAL VALUE IS 5. }
    Begin
     Num_Divided_Of_N := 7;
     inc := 5;
     up := Num_Divided_Of_N;
    End;  {CASE 1}
   2:                          { WHEN THE INCREMENTAL VALUE IS 5. }
    Begin
     Num_Divided_Of_N := 9;
     inc := 1;
     up := Num_Divided_Of_N;
    End;  {CASE 2}
  End;  {case Intval_N_Incmt of}
  For index_1 := 2 To up Do
   N[index_1] := N[index_1 - 1] + inc;          { TO ASSIGN 'N' VALUES. }
  Intval_N_Incmt := Intval_N_Incmt + 1;         { TO INCREMENT THE INTERVAL OF 'N' VALUES. }
  Real_Min_TC  := 999999999;              { TO  MAKE A MAXIMUM NUMBER. }
  Min_TC_Per_N  := 999999999;
 End;  {procedure Initialize_Control_Param_N}

{----------------------------------------------------------------------------}
{ Procedure name : Initialize_Control_Param_B                          }
{ Purpose : To get the initial value of Target Inventory Level(B).              }
{ Input variables: None.                                                       }
{ Output variables: None.                                                      }
{ Globals: Num_Divided_Of_B and B                              }

 Procedure  Initialize_Control_Param_B;
  Var
   inc1, up1, index_3: integer;
 Begin
  inc1  := 10;
  up1  := Num_Divided_Of_B;
  For index_3 := 2 To up1 Do
   B[index_3]  := B[index_3 - 1]  + inc1;     { TO ASSIGN 'B' VALUES. }
 End;  {procedure Initialize_Control_Param_B}

{----------------------------------------------------------------------------}
{ Procedure name : Start_Calculation                           }
```

MPSH2

```
{ Purpose : To take input data and arrange input data.                    }
{ Input variables: None.                                                   }
{ Output variables: None.                                                  }
{ Globals: Initial_N, N, Intval_N_Incmt, B, Intval_B,Num_Divided_Of_B.     }

 Procedure Start_Calculation;
 Begin
  Get_Data_Forecast;               { CALL PROCEDURE Get_Data_Forecast. }
  Get_Data_Initial_Value;          { CALL PROCEDURE Get_Data_Initial_Value. }
  Initialize_Control_Param_ED;     { CALL PROCEDURE Initialize_Control_Param_ED. }
  Initial_N := 360;                { TO INITIALIZE A NORMAL VALUE. }
  N[1] := Initial_N;               { TO ASSIGN A FIRST NORMAL VALUE. }
  Intval_N_Incmt := 0;             { TO INITIALIZE A INCREMENT OF INTERVAL OF A NORMAL VALUE. }
  Initialize_Control_Param_N;      { CALL PROCEDURE Initialize_Control_Param_N. }
  B[1] := 250;                     { TO INITIALIZE A DESIRED INVENTORY VALUE. }
  Intval_B := MAX_B - B[1];
  Num_Divided_Of_B := (Intval_B Div 10) + 1;   { NUMBERS OF INTERVAL OF A DESIRED INVENTORY
      VALUE. }
  Initialize_Control_Param_B;      { CALL PROCEDURE Initialize_Control_Param_B. }
 End;  {procedure Start_Calculation}

{---------------------------------------------------------------------------}
{ Procedure name : Calculate_P_I                                           }
{ Purpose : To calculate Production Values and Inventory Values.           }
{ Input variables: None.                                                   }
{ Output variables: None.                                                  }
{ Globals: H_Value, L_Value, A_Value, C_Value, Temp_N, Temp_E, Temp_B, Temp_D }
{          F, I, P, Check_Level, Check_Minus_I                             }

 Procedure Calculate_P_I;
  Var
   j: integer;
 Begin
  H_Value := Temp_N + Temp_E;      { TO ASSIGN THE HIGH PRODUCTION LEVEL VALUE. }
  L_Value := Temp_N - Temp_E;      { TO ASSIGN THE LOW PRODUCTION LEVEL VALUE. }
  A_Value := Temp_B - Temp_D;      { TO ASSIGN THE MINIMUM ACCEPTABLE INVENTORY LEVEL. }
  C_Value := Temp_B + Temp_D;      { TO ASSIGN THE MAXIMUM ACCEPTABLE INVENTORY LEVEL. }
  j := 0;
  Repeat
   j := j + 1;
   If (F[j] - I[j - 1] < L_Value - C_Value) Then     { THIS IS THE CONDITION OF THE LOW PRODUCTION
      LEVEL. }
    Begin
     P[j] := L_Value;              { THE PRODUCTION VALUE IS THE LOW PRODUCTION LEVEL. }
     Check_Level[j] := 0;
    End {if (F[j] - I[j - 1] < L_Value - C_Value) then}
   Else If (F[j] - I[j - 1] > H_Value - A_Value) Then  { THIS IS THE CONDITION OF THE HIGH PRODUCTION
      LEVEL. }
    Begin
     P[j] := H_Value;              { THE PRODUCTION VALUE IS THE HIGH PRODUCTION LEVEL. }
     Check_Level[j] := 1;
    End   {else if (F[j] - I[j - 1] > H_Value - A_Value) then}
   Else
```

MPSH2

```
    Begin                              { OTHERWISE, THE PRODUCTION VALUE GETS THE NORMAL LEVEL }
     P[j] := Temp_N;
     Check_Level[j] := 2;
    End;   {else}
   I[j] := I[j - 1] + P[j] - F[j];          { TO CALCULATE THE INVENTORY VALUE. }
    Check_Minus_I := I[j];                  { TO CHECK THE INVENTORY VALUE IF NEGATIVE OR NOT. }
  Until (I[j] < 0) Or (j >= MONTH);       { UNTIL THE INVENTORY VALUE IS NEGATIVE OR OVER THE 12
       MONTHS. }
 End;  { procedure Calculate_P_I }

{--------------------------------------------------------------------------------}
{ Procedure name : Calculate_TC                                   }
{ Purpose : To calculate Total Costs.                               }
{ Input variables: None.                                                          }
{ Output variables: None.                                                         }
{ Globals: Total_Cost, C, W, P, I, and Cumul_Total_Cost                      }

 Procedure Calculate_TC;
  Var
   t: integer;
 Begin
  Total_Cost := 0;                     { TO INITIALIZE THE TOTAL COST. }
  For t := 1 To MONTH Do
   Begin
    C[1, t] := 340.0 * W[t];                                  { REGULAR PAYROLL }
    C[2, t] := 64.3 * sqr(W[t] - W[t - 1]);                          { HIRING AND LAYOFF. }
    C[3, t] := 0.20 * sqr(P[t] - 5.67 * W[t]) + 51.2 * P[t] - 281.0 * W[t]; { OVER TIME. }
    If C[3, t] < 0 Then              { IF THE OVER TIME IS NEGATIVE THEN THE OVER TIME IS ZERO. }
      C[3, t] := 0;
    C[4, t] := 0.0825 * sqr(I[t] - 320.0);                     { INVENTORY COST. }
    Total_Cost := Total_Cost + C[1, t] + C[2, t] + C[3, t] + C[4, t];{ TO CUMULATE TOTAL COSTS. }
   End; { for t := 1 to MONTH do }
  Cumul_Total_Cost := Total_Cost;
 End;  {procedure Calculate_TC}

{--------------------------------------------------------------------------------}
{ Procedure name : Calculate_W                                        }
{ Purpose : To calculate Value of Work Force.                               }
{ Input variables: None.                                                          }
{ Output variables: None.                                                         }
{ Globals: G, Check_Level, W, Temp_N, H_Value, L_Value, TC,Cumul_Total_Cost        }
{        Real_Min_TC, P, P1, W, W1, and Min_N                          }

 Procedure Calculate_W;
  Var  .
   m, mm: integer;
   incr: real;
 Begin
  incr := -0.1;         { TO DECREASE THE OVERTIME AND REGULAR PRODUCTION VALUE(G). }
  G[0] := 1.1;          { TO INITIALIZE THE 'G' VALUE. }
  mm := 0;              { THE NUMBER OF 'G' VALUE IS FROM ZERO TO ELEVEN. }
  Repeat
   mm := mm + 1;
```

MPSH2

```
  For m := 1 To MONTH Do
   Begin
    If Check_Level[m] = 2 Then        { IF THE PRODUCTION VALUE IS A NORMAL VALUE... }
     Begin
      W[m] := Temp_N / PRODUCTIVITY_RATE;        { TO CALCULATE THE VALUE OF WORK FORCE. }
     End    {If Check_Level[m] = 2 then}
    Else If Check_Level[m] = 1 Then    {IF THE PRODUCTION VALUE IS A HIGH LEVEL... }
     Begin
      G[mm] := G[mm - 1] + incr;
      W[m] := (H_Value / PRODUCTIVITY_RATE) * G[mm];  { TO CALCULATE THE VALUE OF WORK FORCE.
    }
     End    {else if Check_Level [m]= 1 then}
    Else                    { IF THE PRODUCTION VALUE IS A LOW LEVEL... }
     Begin
      G[mm] := G[mm - 1] + incr;
      W[m] := (L_Value / PRODUCTIVITY_RATE) + (L_Value / PRODUCTIVITY_RATE) * (1 - G[mm]);
     End;
   End;    {for m := 1 to MONTH do}

  Calculate_TC;                { CALL PROCEDURE Calculate_TC. }

  If (TC > Cumul_Total_Cost) Then      { IF NEW TOTAL COST IS SMALLER THEN OLD TOTAL COST... }
   Begin
    TC := Cumul_Total_Cost;
    If TC < Real_Min_TC Then      { IF NEW TOTAL COST IS SMALLER THEN THE REAL MINIMUM TOTAL
     COST... }
     Begin
      P1 := P;              { GET PRODUCTION VALUES TO WRITE THE OUTPUT. }
      W1 := W;              { GET VALUES OF WORK FORCE TO WRITE THE OUTPUT. }
      Min_N := Temp_N;          { ASSIGN THE MINIMUM 'N' VALUE WHEN WE GET THE REAL MINIMUM
    TOTAL COST. }
      Real_Min_TC := TC          { TO ASSIGN THE REAL MINIMUM TOTAL COST. }
     End;
   End
  Else
   TC := TC;
 Until (TC < Cumul_Total_Cost) Or (mm > 10);{ UNTIL NEW TOTAL COST IS BIGGER THEN THE OLD ONE OR
    OVER 11 OF 'G'}
End;  {Calculate_W }

{-----------------------------------------------------------------------------}
{ Procedure name : Select_Intval_N                       }
{ Purpose : To decide the interval of 'N' value.              }
{ Input variables: k_First.                            }
{ Output variables: k_First.                           }
{ Globals:  Num_Divided_Of_N, Min_TC , N, Intval_N_Incmt, Min_TC_Per_N, and Initial_N  }

 Procedure Select_Intval_N (Var k_First: integer);
  Var
   temp_end: integer;
 Begin
  If Intval_N_Incmt = 1 Then          { IF THE INTERVAL OF 'N' VALUE IS 20 THEN... }
   Begin
```

MPSH2

```
    If Min_TC_Per_N > Min_TC Then      { UNTIL GET THE MINIMUM TOTAL COST, WHEN THE INTERVAL IS
      20. }
     Begin
      Min_TC_Per_N := Min_TC
     End
    Else                        { WHEN GET THE MINIMUM TOTAL COST, WHEN THE INTERVAL IS 20. }
     Begin
      Min_TC_Per_N := 999999999; { TO ASSIGN THE NEW TOTAL COST PER EACH INTERVAL TO THE
      LARGEST NUMBER }
      N[1] := Min_N - 15;         { TO DECIDE THE FIRST NUMBER OF 'N' VALUE, WHEN THE INTERVAL
      IS 5. }
      Initialize_Control_Param_N;   { CALL PROCEDURE Initialize_Control_Param_N }
      k_First := 0;              { TO RESET THE 'D' VALUE FROM ZERO. }
     End;
   End {If Intval_N_Incmt = 1 then}
  Else If (Intval_N_Incmt = 2) And (k_First = Num_Divided_Of_N) Then   {THE END OF LOOPING WHEN
      THE INTERVAL IS 5. }
   Begin
    Min_TC_Per_N := 999999999;
    N[1] := Min_N - 4;             { TO DECIDE THE FIRST NUMBER OF 'N' VALUE, WHEN THE INTERVAL IS
      1. }
    Initialize_Control_Param_N;      { CALL PROCEDURE Initialize_Control_Param_N }
    k_First := 0;
   End   {else if Intval_N_Incmt = 2 then}
  Else If (Intval_N_Incmt = 3) And (k_First = Num_Divided_Of_N) Then   {THE END OF LOOPING WHEN
      THE INTERVAL IS 1. }
   Begin
    temp_end := MAX_N - Initial_N;
    k_First := (temp_end Div 20) + 1; { TO ASSIGN THE LOOPING PARAMETER OF 'N' VALUE TO FINISH
      THE TOTAL LOOPING. }
   End;  {else if (Intval_N_Incmt = 3) and (k_First = Num_Divided_Of_N) then}
 End;  {procedure Take_Value}

{-------------------------------------------------------------------------------}
{ Procedure name : Take_Value                                  }
{ Purpose : To arrange Values of 'N', 'E', 'B', and 'D'.                    }
{ Input variables: None.                                                     }
{ Output variables: None.                                                    }
{ Globals:  Num_Divided_Of_N, Min_TC , INCRE_VAL_2, Num_Divided_Of_B, INCRE_VAL_3,  }
{           Temp_N, Temp_E, Temp_B, Temp_D, N, E, B, D, Check_Minus_I, and TC        }

 Procedure Take_Value;
  Var
   k1, k2, k3, k4: integer;
 Begin
  For k1 := 1 To Num_Divided_Of_N Do     { LOOPING FOR NUMBERS OF INTERVALS OF 'N' VALUES. }
   Begin
    Min_TC := 999999999;           { TO ASSIGN THE TEMPORARY MINIMUM TOTAL COST TO THE
      LARGEST NUMBER }
    For k2 := 1 To INCRE_VAL_2 Do      { LOOPING FOR NUMBERS OF INERVALS OF 'E' VALUES. }
     Begin
      For k3 := 1 To Num_Divided_Of_B Do  { LOOPING FOR NUMBERS OF INTERVALS OF 'B' VALUES. }
       Begin
```

MPSH2

```
        For k4 := 1 To INCRE_VAL_3 Do     { LOOPING FOR NUMBERS OF INTERVALS OF 'D' VALUES. }
         Begin
          Temp_N := N[k1];          { TO ASSIGN THE TEMPORARY 'N' VALUE. }
          Temp_E := E[k2];
          Temp_B := B[k3];
          Temp_D := D[k4];
          Calculate_P_I;            { CALL PROCEDURE Calculate_P_I }
          If Check_Minus_I > 0 Then     { IF INVENTORY VALUE IS POSITIVE THEN ... }
           Begin
            TC := 999999999;        { TO ASSIGN THE NEW TOTAL COST TO THE LARGEST NUMBER. }
            Calculate_W;            { CALL PROCEDURE Calculate_W }
            If Min_TC > TC Then       { IF THE NEW TOTAL COST IS SMALLER THEN THE OLD ONE ... }
             Begin
              Min_TC := TC;         { THE OLD ONE GETS THE NEW TOTAL COST. }
             End
           End  {If Check_Minus_I > 0 then}
          Else                    { IF INVENTORY VALUE IS NOT POSITIVE THEN ... }
           Begin
            k2 := k2 + 1;           { TO JUMP NEXT 'E' VALUE. }
            k4 := 0;               { TO RESET THE 'D' VALUE FROM ZERO. }
           End; {else}
         End;  {  for k4:=1 to INCRE_VAL_2 do  }
       End; {for k3 := 1 to Num_Divided_Of_B + 1 do   }
     End;  { for k2:=1 to INCRE_VAL_2   do  }
    Select_Intval_N(k1);
   End;   { for k1 := 1 to Num_Divided_Of_N do  }
 End;   {procedure Take_Value}

{--------------------------------------------------------------------------}
{ Procedure name : Formating_Output                              }
{ Purpose : To write the output.                          }
{ Input variables: None.                                                   }
{ Output variables: None.                                                 }
{ Globals: Pri, Real_Min_TC, F, P, P1, and W1                      }

 Procedure  Formating_Output;
  Var
   ii, iii: integer;
 Begin
  For iii := 1 To 12 Do
   Begin
    Pri[iii]  :=  iii;
   End;
  writeln('Minmum ToTal Cost is : ', Real_Min_TC : 12 : 8);
  writeln;
  writeln;
  writeln('_________________________________________________________________');
  writeln('    Period              Forecast             Production          Workforce    ');
  writeln('_________________________________________________________________');
  For ii := 1 To 12 Do
   Begin
    write(Pri[ii]  :  6);            { TO WRITE NUMBERS OF MONTH. }
    write(F[ii]  :  15);               { TO WRITE FORECAST PRODUCTION VALUES. }
```

MPSH2

```
    write(P1[ii] : 15);                 { TO WRITE PRODUCTION VALUES. }
    writeln(W1[ii] : 20 : 8);               { TO WRITE VALUES OF WORK FORCE. }
   End;
 End;    { procedure Formating_Output}

Begin      {  MAIN  }
 showtext;
 time_in := TickCount;                 { TO ESTIMATE A START CPU TIME. }
 Start_Calculation;                  { CALL PROCEDURE Start_Calculation. }
 Take_Value;                       { CALL PROCEDURE Take_Value. }
 time_out := TickCount;                { TO ESTIMATE A END CPU TIME. }
 writeln('Total CPU Time (in hours) : ', abs(time_out - time_in) / 216000 : 20 : 10); { TO WRITE A CPU
      TIME. }
 Formating_Output;                   { CALL PROCEDURE Formating_Output. }
End.     {  MAIN  }
```

APPENDIX 3 : MPSH1 Control Parameters Sensitivity Results

Table 13. Total costs with various N for MPSH1

N	TC	E	B	D	G
320	$311,188.4	75	340	5	0.9
330	309,516.5	65	340	0	0.9
340	307,977.9	75	310	0	0.9
350	304,358.0	65	310	0	0.9
360	295,178.5	90	300	0	0.9
370	297,412.1	60	280	5	1.0
380	298,852.0	40	310	25	1.0
390	300,785.4	55	280	0	0.9
400	301,535.5	55	300	0	0.9
410	304,894.2	55	280	5	0.9
420	300,532.3	50	260	25	1.0

Table 14. Total costs with various E for MPSH1

E	TC	N	B	D	G
0	$310,410.2	395	250	0	1.0
10	303,147.6	395	250	5	1.0
20	299,445.5	400	270	5	1.0
30	299,903.5	390	260	0	1.0
40	298,852.1	380	310	25	1.0
50	297,655.2	375	280	20	1.0
60	297,421.1	370	250	0	0.8
70	297,488.3	370	290	10	0.9
80	295,885.4	365	290	10	0.8
90	295,178.5	360	300	0	0.9
100	295,943.1	360	330	0	0.9

Table 15. Total costs with various B for MPSH1

B	TC	N	E	D	G
250	$300,979.7	420	40	0	0.8
260	303,116.9	360	65	0	0.9
270	297,512.2	365	70	0	0.9
280	296,421.7	365	75	0	0.9
290	295,627.5	360	85	5	0.9
300	295,178.5	360	90	0	0.9
310	295,178.5	360	90	5	0.9
320	295,178.5	360	90	15	0.9
330	295,178.5	360	90	25	0.9
340	295,178.5	360	90	35	0.9
350	295,178.5	360	90	45	0.9

Table 16. Total costs with various D for MPSH1

D	TC	N	E	B	G
0	$295,178.5	360	90	300	0.9
10	295,178.5	360	90	310	0.9
20	295,178.5	360	90	320	0.9
30	295,178.5	360	90	330	0.9
40	295,178.5	360	90	340	0.9
50	295,178.5	360	90	350	0.9
60	295,885.4	365	80	350	0.9
70	296,412.7	365	75	350	0.9
80	297,512.2	365	70	350	0.9
90	303,116.2	360	65	350	0.8
100	306,472.1	365	50	350	0.8

Table 17. Total costs with various G for MPSH1

G	TC	N	E	B	D
0.0	$440,263.5	400	90	250	30
0.1	406,955.9	400	90	250	30
0.2	378,261.2	400	90	250	30
0.3	355,835.1	400	90	250	30
0.4	338,290.9	400	90	250	30
0.5	324,590.7	400	90	250	30
0.6	314,734.7	400	90	250	30
0.7	308,722.8	400	90	250	30
0.8	306,555.0	400	90	250	30
0.9	295,178.5	360	90	300	0
1.0	298,852.1	360	40	310	25

APPENDIX 4 : MPSH2 Control Parameters Sensitivity Results

Table 18. Total costs with various N for MPSH2

N	TC	E	B	D	G
320	$310,333.8	75	340	5	0.8
330	309,074.7	65	340	0	0.9
340	307,993.6	70	290	0	0.5
350	303,903.2	90	280	0	0.3
360	295,012.4	90	300	0	0.6
370	295,815.9	65	280	10	0.7
380	297,917.7	65	300	20	0.6
390	297,599.3	40	280	0	0.6
400	299,026.5	45	280	0	0.6
410	301,373.2	40	300	10	0.7
420	299,600.6	50	260	25	0.8

Table 19. Total costs with various E for MPSH2

E	TC	N	B	D	G
0	$310,424.5	390	350	30	1.0
10	303,147.6	395	250	5	1.0
20	299,445.5	400	270	5	1.0
30	298,811.3	385	250	0	0.6
40	298,789.1	380	310	25	0.9
50	299,103.3	380	330	25	0.8
60	296,128.0	370	280	5	0.7
70	295,654.5	365	270	0	0.7
80	295,613.4	365	290	0	0.6
90	295,012.4	360	300	0	0.6
100	295,973.0	360	330	0	0.5

Table 20. Total costs with various B for MPSH2

B	TC	N	E	D	G
250	$298,553.0	425	45	0	0.7
260	300,654.5	380	45	0	0.3
270	295,654.5	365	70	0	0.6
280	297,599.3	390	40	0	0.5
290	295,134.5	360	85	5	0.5
300	295,012.4	360	90	0	0.6
310	295,012.4	360	90	5	0.6
320	295,012.4	360	90	15	0.6
330	295,012.4	360	90	25	0.6
340	295,012.4	360	90	40	0.6
350	295,012.4	360	90	50	0.6

Table 21. Total costs with various D for MPSH2

D	TC	N	E	B	G
0	$295,012.4	360	90	300	0.6
10	295,012.4	360	90	310	0.6
20	295,012.4	360	90	320	0.6
30	295,012.4	360	90	330	0.6
40	295,012.4	360	90	340	0.6
50	295,012.4	360	90	350	0.6
60	295,163.4	365	80	350	0.6
70	295,214.0	365	75	350	0.6
80	295,654.5	365	70	350	0.6
90	297,173.8	375	55	350	0.8
100	297,674.2	375	90	350	1.0

Table 22. Total costs with various G for MPSH2

G	TC	N	E	B	D
0.0	$303,723.2	380	65	290	10
0.1	301,403.8	380	70	300	5
0.2	299,752.9	380	70	300	5
0.3	297,746.5	360	90	300	0
0.4	296,089.9	360	90	300	0
0.5	295,178.6	360	90	300	0
0.6	295,012.4	360	90	300	0
0.7	295,486.6	360	85	290	0
0.8	296,486.4	360	85	290	0
0.9	298,159.7	360	85	290	0
1.0	298,852.1	380	40	310	25

APPENDIX 5 : Total Cost on Various Control Parameters

Figure 10. Total costs on five parameters for MPSH1

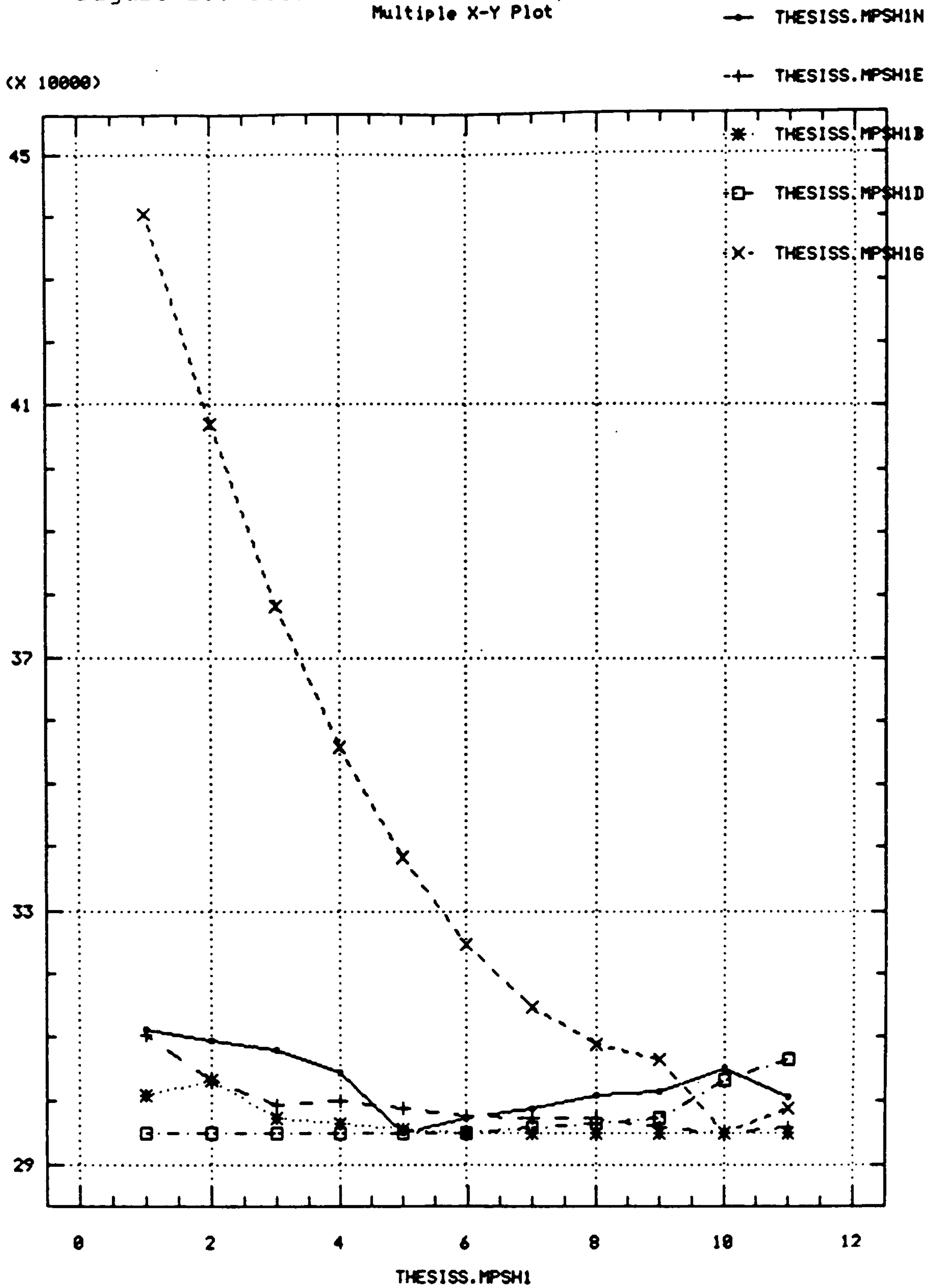


Figure 11. Total costs on five parameters for MPSH2