Old Dominion University
ODU Digital Commons

Engineering Management & Systems Engineering Theses & Dissertations | Engineering Management & Systems Engineering

Spring 1996

Dynamic Lot Sizing and Scheduling in a Multi-Item Production System

Padmanabhan Soundar
Old Dominion University

Follow this and additional works at: https://digitalcommons.odu.edu/emse_etds

Part of the Industrial Engineering Commons, and the Operational Research Commons

Recommended Citation
Soundar, Padmanabhan. "Dynamic Lot Sizing and Scheduling in a Multi-Item Production System" (1996). Doctor of Philosophy (PhD), dissertation, Engineering Management, Old Dominion University, DOI: 10.25777/jqpq-dc44
https://digitalcommons.odu.edu/emse_etds/126

This Dissertation is brought to you for free and open access by the Engineering Management & Systems Engineering at ODU Digital Commons. It has been accepted for inclusion in Engineering Management & Systems Engineering Theses & Dissertations by an authorized administrator of ODU Digital Commons. For more information, please contact digitalcommons@odu.edu.

للاستشارات المنارة

www.manaraa.com

DYNAMIC LOT SIZING AND SCHEDULING IN A MULTI-ITEM PRODUCTION SYSTEM

by

PADMANABHAN SOUNDAR
B. Tech. July 1989, Indian Institute of Technology
M.S. December 1991, University of Missouri-Columbia

A Dissertation submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

ENGINEERING MANAGEMENT

OLD DOMINION UNIVERSITY
May 1996

Approved by:

Han P. Bao (Director)

Billie M. Reed (Member)

Derya A. Jacobs (Member)

Ravi Mukkamala (Member)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ABSTRACT

DYNAMIC LOT SIZING AND SCHEDULING IN A MULTI-ITEM PRODUCTION SYSTEM

Padmanabhan Soundar
Old Dominion University, 1996
Director: Dr. Han P. Bao

In this research, algorithms are developed to address the problem of dynamic lot sizing and scheduling in a single level (or single operation) production system. This research deviates from previous research in this area in that it does **not** have the kind of assumptions regarding the real world production system that normally were made to reduce the complexity of the problem. Specifically, this research explicitly considers finite capacity, multiple items, known deterministic dynamic demand, sequence dependent setup times and setup costs, setup carryover and variable backlogging. The objective is to simultaneously determine the lot size and the sequence of production runs in each period to minimize the sum of setup, inventory, and backlogging costs.

The research here is motivated by observations of a real world production system that has a highly automated operation with sequence dependent setup times. For problems of this kind, optimal solution algorithms do not yet exist and, therefore, heuristic solution algorithms are of interest. Two distinct approaches are proposed to address the problem. The first is a greedy approach that eliminates setups while potential savings are greater than the increase in inventory or backlogging costs incurred. The second approach solves the much easier single item problem optimally for each item and then adapts the solution to account for capacity constraints. An intelligent modification to the second approach is also tried where a "overload penalty" is used between

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

successive runs of the single product optimization algorithms. A common component of each approach is a dynamic programming algorithm implemented to determine the optimal sequence of production within each period and across the scheduling horizon. The addition of sequence dependent considerations introduces a traveling salesman type problem to the lot sizing and sequencing decisions.

The algorithms have been tested over several combinations of demand and inventory related cost factors. Specifically the following factors at two levels each have been used: problem size, demand type, utilization, setup cost, backlogging cost, and backlogging limit. The test results indicate that, while the performance of the proposed algorithms appear to be affected by all the factors listed above, overall the regeneration algorithm with "overload penalty" outperforms all of the other algorithms at all factor level combinations.

In summary, the contribution of this research has been the development of three new algorithms for dynamic lot sizing and scheduling of multiple items in a single level production system. Through extensive statistical analysis, it has been shown that these algorithms, in particular the regeneration algorithm with "overload penalty", outperform the conventional scheduling techniques such as no lot sizing and economic manufacturing quantity.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

To my parents without whose support and encouragement my pursuit of this degree would not have been possible.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ACKNOWLEDGEMENTS

My thanks are extended to all the faculty and staff of the Department of Engineering Management for their kind support during my course of study. I extend my gratitude to my dissertation committee chair Dr. Han P. Bao for providing guidance and encouragement throughout my research. I also wish to express my thanks to the other members of my dissertation committee: Dr. Billie Reed, Dr. Derya Jacobs and Dr. Ravi Mukkamala, for their help in putting together the final document.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

TABLE OF CONTENTS

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

LIST OF TABLES

Table | Page

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER I

INTRODUCTION

This research deals with generation of production schedules that are used by manufacturing personnel to control the flow of material through a production system. Production schedules contain information on timing, sequencing, and sizing decisions for production lots in a manufacturing environment. Specifically, this thesis applies mathematical and operations research techniques to obtain good solutions to a particular class of real world production scheduling problems called dynamic lot sizing and scheduling problems.

Currently manufacturing resources planning (MRP II) is a very popular approach used by manufacturing enterprises to perform their scheduling operations. However when MRP II is used, it does not always give the best schedule. This is because it does not take into consideration all the factors that influence the operations in the production system. While consideration of more factors improves the quality of schedules generated, it also increases the complexity of the scheduling problem. Scheduling problems are differentiated from one another based on number of demand and production factors that are included in problem formulation.

Scheduling task in a real world production system is a complex endeavor and has been used as an important tool to control production costs. Potential benefits resulting from good schedules have made production scheduling a hot topic for research in the

---

International Journal of Production Research is the journal model used in this dissertation for references, figures and tables.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

area of production and operations management. The nature of scheduling problems and the important role they play in controlling manufacturing costs is discussed next.

Background

For a manufacturing enterprise to succeed, it must keep production costs low and also deliver its customer orders on time. Production costs can be classified into basic production costs and inventory related production costs. Basic production costs include material costs, labor costs, machine setup costs, and overhead costs. Inventory related cost consists of inventory costs and backlogging cost. Inventory cost is the cost of capital tied up in inventory. Backlogging cost is related to the ability of a production system to meet customer due dates. Customer orders feature product requirements, quantity, and due dates. Meeting customer requirements is the ability to deliver products in the quantity ordered at the agreed upon time. When an organization cannot meet its promised due dates, it backlogs the order (assuming the order is not lost). The costs incurred when customer requirements are not met are not easily expressed in monetary terms. They include loss of customer goodwill and loss of sales revenues resulting from a shortage situation. One way of accounting for these intangibles is to levy a penalty, called backlogging cost, when a promised due date is allowed to slip and order is backlogged.

Basic production costs and ability to meet customer orders work as opposing forces, i.e., as the flexibility of the system is increased to meet customer requirements more setups are required, this causes an increase in the basic production costs and a

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

decrease in inventory related production costs. Performing operations in a way that minimizes production costs while meeting customer requirements is a non-trivial endeavor. Scheduling is an important tool available to management to obtain an optimum balance between trying to satisfy customer requirements and increasing production costs. Schedules control setup costs, inventory costs and backlogging costs incurred in a given situation. Since costs like material costs and labor costs are not directly affected by schedules, given two schedules the better one is the one that results in a lower sum of setup, inventory and backlogging costs.

In essence, the scheduling problem reduces to meeting several customer requirements for multiple products by the requested dates while keeping production costs to a minimum. Inputs to the schedule consist of information on demands, production rates, setup time and setup costs, and inventory and backlogging costs. The schedule generates information regarding the timing, size, and sequences of production lots. A real world batch production system where a non trivial scheduling problem exists is described next.

Motivating Case

This research follows observations made by the author at a batch production facility. To protect the proprietary information of the company involved, only a general overview of the production facility is described. However, sufficient detail is provided to grasp the essence of the scheduling problem. The facility consists of two stages, the first being a highly automated coating stage, and the second being a mostly manual packaging

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

stage. The facility serves the needs of about forty customers who place orders for one of the seventy or so finished products. There are about fifteen different coated products, some packaged in several styles to obtain the greater number of finished products. Figure 1 illustrates the structure of the products produced in this facility. All products produced in the facility pass through the first stage before they get to stage 2.

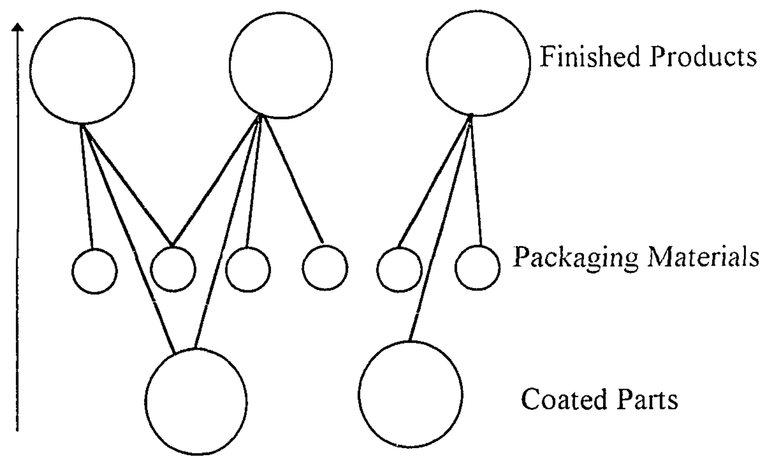

Figure 1: Product Structure Produced in the Facility

Raw material is brought into the first stage on motorized pallets which can accommodate various number of pieces based on the size of the product. A robot arm picks up pieces from the pallet and dips them in a chemical bath. The size of the product and the type of chemicals used determine the speed of the dipping process. Hence, the production rate of the product being coated is a function of its size and chemical coating applied. When production is switched from one product to another, one or more of the following has to be changed:

1. pallet carrying the tubes,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. gripper of the robot arm used to dip the tubes, and/or
3. chemical solution in bath.

These changes result in significant setup times when production is switched from one product to another. Since the changes involved are not always the same, the setup time is dependent on the sequence of production. For example, switching between two products of the same size but different chemicals in the bath would require time for just changing the chemical in the bath. However, when switching between two products of different sizes and bath chemicals the pallet, gripper, and chemical solution have to be changed. Therefore, the second switch would take a longer duration.

Coated products are stored for up to five days, depending on product, before they are moved to the packaging department. Packaging is a team based operation. Packaging requirements, and thus team size, task assignment and production rate are product dependent. Packaging personnel are assigned to teams at the beginning of each shift. The composition and size of teams may vary between shifts, however they are not changed during shifts. The setup required to start packaging is minimal and even this is performed by a single lead operator before the start of each shift for all workstations. Hence, the setup time required for packaging operations is not significant.

Proper coordination of schedules between the two stages is required to meet the twin targets of reducing basic production costs and adequately satisfying customer requirements. However, the scheduling problem in the two stages are not of equal importance. For the following reasons proper scheduling in stage 1 is more important than scheduling in stage 2.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. Stage 1 adds more value to the product than stage 2 and hence must be more closely monitored.
2. Stage 1 is the sole internal source of semi finished products in stage 2, hence it is possible that stage 2 merely mimics the schedule in stage 1.
3. Stage 2 has greater flexibility, in that multiple products can be packaged simultaneously and production rate can be controlled by changing the number of teams allocated to a particular product. The rate of production is constant for a given product in stage 1. This allows stage 2 to quickly adapt to changes in stage 1.
4. Capacity available in stage 2 is greater than stage 1 and this absorbs inequalities in rates of production between stages.

In the above described production system customer requirements are tracked and schedules are generated using a MRP II system. The first stage is certainly the more critical of the two stages and the MRP II system focuses on production in this stage to determine the flow of products through the facility. The modus operandi of MRP II systems is discussed in the next section.

Production Control Using MRP II

MRP II systems are basically an extension of the original material requirements planning (MRP) systems that were first used in the fifties. In addition to MRP, modern MRP II systems help the manufacturing enterprise integrate all of its manufacturing support operations like accounting, quality control, sales, etc. For further description of MRP II functionalities and methodologies readers are referred to Turbide (1995) and

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Wallace (1990) and references therein. MRP II is indispensable in a multi-stage production environment where proper coordination of material flow between stages is required. For this reason, they are used by a large number of manufacturing organizations to perform this task. Salomon (1991) has identified two important types of decisions that MRP II systems support:

1. Given the demand requirements in the final stage and the production information in each stage, MRP II coordinates the production of each item at each stage.
2. Computation of lot sizes at each stage to meet demand requirements and minimize the inventory and setup costs.

MRP II systems use a hierarchical two phase approach to the scheduling problem. Initially, MRP II systems use economic manufacturing quantity (EMQ) calculations to compute the size of the production runs. This is followed by sequencing and timing decisions (the economic lot sizing problem) based on EMQ computations for each product. EMQ calculations are based on the average demand rate per period, production rate per period, setup cost, and inventory holding cost factor for each product. EMQ computations determine the cycle time for each product, the production time required and the cost of the schedule. Cycle time determines the lot size for the product and is essentially the duration of demand that is satisfied by a single lot. Elmaghraby (1978) presents a method to calculate the cycle time for each product. This procedure is repeated below.

The average cost per unit time for each setup when product i is produced in cycles of length τ_i is given by the equation:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$SC_i = \frac{A_i}{\tau_i} + \frac{h_i r_i (1 - \rho_i)\tau_i}{2} \tag{1}$$

the value of τ_i^* the optimal value of τ_i can be obtained from the equation $\frac{dSC_i}{d\tau_i} = 0$

which yields the minimum cost cycle given by

$$\tau_i^* = \sqrt{\frac{2A_i}{h_i r_i (1 - \rho_i)}} \tag{2}$$

and a minimum cost of

$$SC_i^* = \sqrt{2A_i h_i r_i (1 - \rho_i)} \tag{3}$$

From equation (2), optimal lot size is obtained as $Q_i^* = r_i \tau_i^*$, using these lot sizes for each product a economic lot size problem (ELSP) is solved to determine the sequence and the timing of production. EMQ calculations determine the lot sizes that minimize the inventory related production costs represented in equation (1) and this minimum cost value is given in equation (3). However, this method of calculating lot size is based on several assumptions which are not true in the production system that motivated this research. These assumptions oversimplify the problem by ignoring important aspects of the production environment and demand environment. This raises several issues when MRP II generated schedules are used in the motivating case. These issues are addressed next.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Shortcomings of Current Production Control System

Equation (1) which is minimized by the cycle time in equation (2) does not account for two important aspects of the inventory related production cost that exists in the motivating case:

1. sequence dependent nature of the setup costs, and
2. backlogging costs when customer due dates are missed.

Consideration of the first aspect would superimpose a traveling salesman type problem on the EMQ calculations, this issue has been addressed by Taha (1975). However, ignoring the customer due date requirements is a fundamental flaw of the EMQ calculations. This limitation is imposed by equation (1) which artificially imposes a static demand rate over a infinite horizon in what is essentially a dynamic demand environment.

In the specific production system described above the customer orders and forecasts are used to convert date specific orders to average monthly demand values. MRP II then calculates the lot sizes of the products from these static demand values and determines the timing of the production runs without regard to customer requested due dates. The products are shipped to customers as they are produced and the dates requested by customers are not adhered to.

In these cost calculations, backlogging cost aspects are ignored (customer due dates are ignored) while calculating both lot sizes and timing of production runs. In this particular facility, ignoring delivery dates requested by the customers has resulted in perennial backlogging for some orders. Even worse the potential loss to the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

organization due to this loss of customer goodwill is not considered in scheduling decisions. The reason being that coating schedule is based on static demand when the actual demand is dynamic because MRP II systems based on EMQ cycle time computations are incapable of handling dynamic demand.

MRP II systems in facilities that use them contain information that is required to perform the scheduling operation. This fact and other beneficial features of MRP II can be taken advantage of by developing algorithms to work in tandem with MRP II systems. Therefore one popular alternative to address the shortcomings of MRP II systems has been the addition of modules to tackle specific situations. Along this same line, scheduling algorithms have to be incorporated for satisfactory application of MRP II system in the case discussed here. The complexity of the scheduling problems (and therefore complexity of the algorithms used to generate schedules) are influenced by a number of attributes (factors) that exist in the production environment and the demand environment where the schedule is to be applied.

Problem Attributes

The complexity of the scheduling problem increases as more attributes are taken into consideration in the problem formulation. While simplifying assumptions can be made to reduce problem complexity, ignoring one or more of these attributes affects the quality of the solution when it is implemented in a real world production system. The following attributes can be identified in the coating stage of the motivating case and must be properly accounted for:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. *Known deterministic dynamic demand*: The customers place orders in known discrete quantities that can vary with time. This can be linearly transformed into due date in the first stage by subtracting number of days required for packaging and the wait time between the stages.
2. *Sequence Dependency*: The sequencing of production is critical because setup costs are a function of this sequence. Also sequence of production must be determined to calculate feasibility of a schedule in terms of available capacity.
3. *Setup Times*: A known amount of time is used for setup when production is switched from one product to another. During setup, the resource is not available for production and there is a cost incurred for each setup. These durations are typically expressed in hours.
4. *Setup Carryover*: When a product is coated continuously over a period of several days setup is carried over from one day to the next, i.e. a separate setup is not necessarily required for each day of production.
5. *Capacity Constraints*: Available capacity is finite, and this must be considered in schedule development. When a generated schedule requires capacity in excess of what is available then the schedule is infeasible. Capacity is expressed as number of hours of production available in a day.
6. *Multiple Items*: Scheduling decisions must consider the fact that more than one product is competing for limited resources. The algorithm must ensure that only one product is scheduled on the machine at a given time.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

7. *Backlogging Limit*: An upper bound on maximum days allowed for backlogging of orders. In general if customer orders are not met within a certain duration after the due date these orders are lost, i.e. the customer is no longer interested in the order. Therefore it is essential that customer orders are not backlogged for durations greater than that stated in this attribute. A backlogging cost is assessed for each unit of production backlogged for each day between the due date and maximum delay allowed.

All of the above attributes are included in the formulation of the scheduling problem addressed in this research. To facilitate the formulation of the problem and its solution certain conventions are used, these are described next.

Conventions

Before the research problem can be presented conventions required to understand the formulation are listed below.

1. Inventory costs, backlogging costs, and due date specifications are assessed at finite intervals within the scheduling horizon. These intervals differ in magnitude from setup time and capacity availability specifications. To capture this difference two distinct time buckets are used to define the problem. Inventory and backlogging cost computations, and due dates use large time buckets (also called periods). Setup times, and capacity are expressed in small time buckets. For example large time buckets can be days and small time buckets can be hours.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. Scheduling horizon is expressed as number of periods over which demand is to be scheduled.
3. Demand is expressed as hours of capacity required to meet the customer requirements. This is achieved by dividing the actual demand units by production rate. For example, if 800 units of a product is required and its coating rate is 400 per hour, then this demand can be expressed as 2 hours.
4. Setup costs are calculated as linear multiples of setup times and a setup cost factor. This factor can be the labor rate or any other value used to compute setup costs.
5. Inventory cost is calculated as the product of an holding cost and number of units of a product (expressed in hours of demand) carried from one large time bucket to the next. Similarly, backlogging cost is the product of a backlogging cost factor and hours of demand backlogged. Typically, backlogging cost factor is greater than holding cost.
6. The first product scheduled at the start of the scheduling horizon will require setup based on the last product produced on the day before the scheduling horizon. In the absence of this information the following convention is used. There exists a dummy product which is processed before the start of the scheduling horizon and must be reloaded at the end of the scheduling horizon. Without loss of generality, it is assumed that this product is 0 (zero) and that $s_{i0} = s_{0i} = 0 = S_{i0} = S_{0i}$.
7. If idle periods exist in a given schedule then the current setup in the machine is not tampered with during these times. To facilitate modeling a dummy product k is

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

assumed to be produced during the idle periods in the system ∋ $s_{ij} = s_{ik} + s_{kj}$ and $S_{ij} = S_{ik} + S_{kj}$ $\forall$ i,j.

8. Demand information available has been modified to reflect the post coating waiting time and packaging time. This allows production scheduled for day t to be used to satisfy demand on day t.

With this information, the research problem can be formally stated as a mathematical programming problem.

Statement of the Research Problem

The formulation of the research problem as a mixed integer non-linear programming problem is described in this section. The objective is to determine the production schedule that minimizes the sum of inventory costs, backlogging costs and setup costs. Therefore, the objective function (TC) is

$$MIN \sum_{i}\sum_{t}(u_{it}h_i + v_{it}b_i) + \sum_{i}\sum_{j}\sum_{t} S_{ij}y_{ijt} + \sum_{i}\sum_{j}\sum_{t} S_{ij}\, max(\beta_{it-1} + \lambda_{jt} - 1{,}0) \tag{4}$$

Basic production costs are invariant and hence are not included in the objective function. A production schedule is determined to be optimal if equation (4) is optimal for a feasible combination of x_{it}, y_{ijt}, λ_{it} and β_{it}. The feasibility of these variables is determined by their ability to satisfy the following constraints. The first set of constraints model inventory, inventory balance, backlogging, and the relationship between the binary setup variable and production scheduled:

$$u_{it} = max(I_{it}, 0) \qquad \forall\, i, t \tag{5}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$v_{it} = max(-I_{it}, 0) \qquad \forall\, i, t \tag{6}$$

$$I_{it-1} + x_{it} - I_{it} = d_{it} \qquad \forall\, i, t \tag{7}$$

$$I_{it} \geq - \sum_{k=t-\alpha+1}^{t} d_{ik} \qquad \forall\, i, t \geq \alpha \tag{8}$$

$$x_{jt} - M(\sum_{i} y_{ijt} + \lambda_{jt}) \leq 0 \qquad \forall\, j, t \tag{9}$$

Constraints stated in equations (5) and (6) determine whether a positive inventory or a backlogged situation exists at the end of period t. This ensures that proper cost multipliers are assigned to positive and negative inventories. Equation (7) is a typical inventory balance equation. This equation states that the inventory difference between the beginning and end of a period is the difference between the scheduled production in that period and demand in that period. Constraint (8) states that backlogging cannot extend beyond α days or, stated in another way, the amount of production backlogged can at worst be equal to the sum of demand over the past α -1 days. Constraint (9) ensures that if a product is to be produced during a day then it is either produced first in the day or it follows immediately after another product. The next set of constraints handle sequencing issues and avoid more than one setup for a product during a day:

$$\lambda_{it} + \sum_{j} y_{jit} \leq 1 \qquad \forall\, i, t \tag{10}$$

$$\beta_{it} + \sum_{j} y_{ijt} \leq 1 \qquad \forall\, i, t \tag{11}$$

$$\sum_{i}(\sum_{j} y_{ijt} - M\lambda_{it}) \leq 0 \qquad \forall\, t \tag{12}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$\sum_j (\sum_i y_{ijt} - M\beta_{jt}) \leq 0 \qquad \forall\, t \tag{13}$$

$$\beta_{it} \geq \lambda_{it} - \sum_j y_{ijt} \qquad \forall\, i, t \tag{14}$$

$$\sum_i \beta_{it} \leq 1 \qquad \forall\, t \tag{15}$$

$$\sum_i \lambda_{it} \leq 1 \qquad \forall\, t \tag{16}$$

$$\beta_{it} \geq (1 - \sum_j \lambda_{jt}) \times \beta_{it-1} \qquad \forall\, i, t \tag{17}$$

Constraints (10) and (11) avoid cycling of production (multiple lots of the same product) within a period. Constraint (10) ensures that if a product is produced first on a day then it is not sequenced after other products on that day. Similarly, (11) guarantees that if a product is to be produced last in a day then it is not sequenced before other products on that day. Constraints (12) and (13) make sure that on days in which more than one product is produced the sequence is started and ended appropriately. Constraint (12) states that there must exist at least one starting product if a sequence is to exist on a day. Equation (12) along with (10) guarantees that the first product sequenced on that day is at the head of the sequence. Constraint (13) makes sure that if 2 or more products are coated on a day then at least one of them is the last product. Constraint (11) along with (13) ensure that the last product sequenced on a day is at the bottom of the sequence.

Constraint (14) states that if only one product is produced on a day then the last product is the same as the first product. Constraints (15) and (16) ensure that at most one first and one last product are sequenced on a day. Constraint (17) handles situations

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

when there is a whole day in which no production is scheduled, in which case, it ensures that the last product produced is considered as the current product for proper accounting of setup times. The penultimate set of constraints described below is capacity related:

$$\omega_t = C_t - \sum_i x_{it} - \sum_i \sum_j s_{ij} y_{ijt} - \sum_i \sum_j (s_{ij} \times max(\beta_{it-1} + \lambda_{jt} - 1, 0)) + \\ min(\omega_{t-1}, \sum_i \sum_j s_{ij} \times max(\beta_{it-1} + \lambda_{jt} - 1, 0)) \quad \forall\, t \tag{18}$$

Constraint (18) imposes the capacity constraint on final schedule developed and also determines the idle time for each period. Idle time is calculated as the initial capacity from which is subtracted the total production scheduled and setup times consumed. It also accounts for setup carryover and makes sure that any excess capacity in the previous period is used for setup in the current period if so desired. Finally, we are left with constraints that determine feasible ranges for the variables used in formulating the problem and their initial conditions.

$$\omega_t, x_{it}, u_{it}, v_{it} \geq 0 \qquad \forall\, i, t \tag{19}$$

$$M \rightarrow \infty \tag{20}$$

$$\beta_{it}, \lambda_{it}, y_{ijt} \in \{0, 1\} \qquad \forall\, i, j, t \tag{21}$$

$$y_{iit} = 0 \qquad \forall\, i, t \tag{22}$$

$$\omega_0 = 0 \tag{23}$$

$$I_{i0} = I_{iT} = 0 \qquad \forall\, i \tag{24}$$

$$\beta_{i0} = 0 \qquad \forall\, i \tag{25}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$\beta_{00} = 1 \tag{26}$$

Constraints (19), (20) and (21) define the ranges for the variables. Constraint (22) precludes two successive spots in a sequence being taken by the same product. Constraints (23) and (24) initialize the inventory levels and idle capacity at the extremes of the production horizon. Finally, constraints (25) and (26) establish the assumption that a dummy product (product 0) is produced at the beginning of the scheduling horizon.

Objective of the Research

The objective of the research reported in this thesis is to formulate and make a contribution to the solution of a difficult lot sizing problem that exists in the real world. In this study, rigorous algorithms founded in mathematics and operations research are developed to tackle a real world scheduling problem stated in the previous section. Results obtained from this research are expected to lower the cost of schedules generated in manufacturing enterprises when compared to the current MRP II developed schedules. Although the research is motivated by an experience with a single case, this type of scheduling problem is not uncommon in the manufacturing arena.

The problem as formulated in the previous section is a NP complete problem. This is stated without proof since much simpler problems have been established to be NP complete in the literature. Hence, optimal solutions to problems of the size that exist in the real world cannot be determined with reasonable computational effort. For this reason, this research focuses on obtaining good solutions to the class of problems

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

addressed here rather than obtaining optimal solution. This research deviates from previous research in the sense that it does not remove from consideration any of attributes in the production system that increases the complexity of the problem. The robustness of the algorithms developed here under various test conditions is studied. From the results of this study it is possible to obtain insight into the behavior of critical problem attributes.

Summary

In this chapter, an introduction was provided to the problem of dynamic lot sizing and scheduling in a capacitated multi-item production environment. The role of scheduling in controlling production costs was described. The motivating case was introduced and the shortfalls of using MRP II generated schedules in such an environment was discussed. The problem attributes that make scheduling in the motivating case a difficult problem were addressed. A formulation of the scheduling problem was discussed in detail and the objective of this research was stated.

The complexity of the simultaneously determining lot size, timing and sequence of production is a very difficult yet, very relevant problem in production and operations management. Several formulations of the problem have been addressed in the past. Some researchers have attempted to solve problems with several machines in parallel in a single stage and also multi-stage problems. However, most of these attempts involve extensions of the single stage solution approaches. Besides, in a multi-stage system there

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

usually exists a single stage that acts as a bottleneck to the system. In these situations, schedules at the bottleneck controls the schedules adopted by other resources.

Hence, a vast majority of research efforts have focused on the single stage (level) production system. While some have been solved successfully, optimal solutions to a vast majority of the problems have eluded researchers. Some of these formulations and the approach used to solve the problems are discussed in the next chapter.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER II

LITERATURE REVIEW

In this chapter a comprehensive review of past research relevant to the current research problem is presented. The aim is to draw implications from these published results for the solution approach proposed to address the research problem. Research in the area of lot sizing has been on going since before world war I. Hence, a complete review of all literature in this area is beyond the scope of this work. Only the most significant developments in this area and work that is directly relevant to the problem addressed here are reviewed.

This chapter reviews previous research into different formulations and solution approaches to the single level lot sizing problems. Single level manufacturing processes are characterized by single level product structures, in which the product is converted from raw materials to a finished product by one machine. The characteristics of the lot sizing problem of the production facility in the motivating case are similar to single level lot sizing problems. In such environments product demands are assessed from customer orders and market forecast.

Past research in the area of single level lot sizing problems have been characterized based on which of the problem attributes described in the previous chapter have been accounted for. As more of those attributes are included in the problem formulation tackled, both the problem and solution complexity increase. In this chapter previous work is presented in order from the simpler problems to more difficult ones. To facilitate the presentation of previous work, a classification and representation

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

scheme is used. This scheme is adapted to include attributes that are taken into consideration in the scheduling problem addressed here. A framework for classification and representation of single level lot sizing problems is presented next.

Classification and Representation

To classify and represent research in single level lot sizing problems a 6 field notation is used. This classification differs slightly from the one used by Salomon (1991), in that he does not consider problems that allow for backlogging. Two fields considered by Salomon, inventory cost structure and number of machines in parallel, are dropped from the notation used here. Inventory cost structure is used to classify inventory costs as either time dependent or time independent. All literature reviewed here consider time independent inventory costs as this is also the inventory cost structure of the problem addressed here. However, this does not preclude each product from having different inventory cost factors in multi-item problems. Since only a one machine single stage problem is considered here multi machine single stage problems are not relevant to the problem addressed here. On the other hand, backlogging exists in the motivating case and a field is added to classify research according to their backlogging considerations.

The notation used is D/C/B/N/SC/ST. Each field in this notation and the range of possible values is described below.

Type of Demand (D): This field can take two possible values c or d. c means a constant rate and d means deterministic and dynamic.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Capacity Constraints (C): This field is 0 if the problem does not consider capacity constraints, 1 if capacity constraints are considered.

Backlogging (B): This field can take on three possible values 0, i, v. If no backlogging is allowed then this field takes on a value of 0. If backlogging is allowed and unlimited then this field takes on a value of i for infinite, i.e. backlogging for up to the scheduling horizon is allowed. If backlogging is allowed but limited then this field takes on a value of v, which should be less than the scheduling horizon.

Number of Items (N): This number is equal to 1 if single item problem is considered, otherwise N is the number of items greater than 1.

Setup Cost (SC): This field contains letters A, SD, SI, and TD respectively zero setup costs, sequence dependent, sequence independent, and time dependent setup costs.

Setup Times (ST): If setup times are assumed to be zero then this field takes on a value of A, for absent. If setup times are sequence dependent then this field takes a value of SD. If sequence independent setup times are considered this field has a value of SI. Time dependent setup times are represented using TD.

The classification and representation schema is robust and can be used to represent scheduling problems spanning the entire spectrum from the easiest single item problems to the complex problem addressed here. Most early scheduling problems considered one item production systems, and analytical solutions to these problems were easily developed. Several formulations of the one item problem were addressed. Some of these are reviewed next.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

One Item Problems

The easiest class of lot sizing problems is the c/0/0/1/SI/A, i.e. single item uncapacitated problem with no backlogging, sequence independent setup costs and no setup time. A solution to this problem was first proposed by Harris (1913). For an item i the optimal lot size for this problem is given by the equation $Q_i = \sqrt{\frac{2 \times r_i \times A_i}{h_i}}$. This lot size is called the economic order quantity (EOQ) and because of its simplicity is still used in some circles.

d/0/0/1/TD/A is a similar problem to the previous one except that demand is deterministic and changes from period to period. This problem is a slightly more difficult problem than the uniform demand problem. It was first solved optimally by Wagner and Whitin (1958). They proposed a dynamic programming algorithm that can solve problems of any duration. The algorithm optimally determines the periods of zero production and quantity of production in the non-zero periods. The authors used two principles to reduce the solution space searched by their algorithm.

1. Optimal solution would not include a situation where inventory is brought into a period and also non-zero production is scheduled in the period. Because, in the absence of capacity constraints, cost could be reduced by simply adding the inventory to the production in that period. For an arbitrary product i this can also be stated as $I_{it-1}x_{it} = 0$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. For a schedule to be optimal, $x_{it} = 0$ or $\sum_{t'=t}^{k} d_{it'}$ *for some* $k,\ t \le k \le T$ for $\forall$ t and for an arbitrary product i. Any other value for x_{it} implies that the first principle is violated. This limits the search space for the optimal solution to $\frac{T(T+1)}{2}$.

The two criteria outlined above are popularly referred to as the Wagner-Whitin criteria and have been extensively used to improve the quality of solutions for more complex scheduling problems. An alternate solution approach to the same problem was presented by Silver and Meal (1969). Their procedure is founded on EOQ and provides results comparable to Wagner and Whitin with a reduced computational requirements.

EMQ computations presented in chapter I is another class of single item problems, these are represented by the notation c/1/0/1/SI/SI. This approach deviates from EOQ in that it incorporates production rates as well as demand rates and was reported initially by Koepke (1941).

The single item dynamic lot sizing problem with variable backlogging, d/0/v/1/TD/A, was independently solved optimally in the mid 60's by Zangwill (1966, 1969) and Manne and Veinott (1967). The work presented by these authors is reviewed in detail in chapter III as it forms the basis for one of the approaches to solving the research problem proposed here. A simple heuristic solution that is comparable in solution quality to the more computationally intensive optimal algorithms above is presented by Choo and Chan (1990). Their heuristic is based in a "eyeballing" technique that tries to strike a balance between the inventory, backlogging and setup costs.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

In the presence of unlimited capacity single item problems can be successfully applied to multi-item production systems. The unconstrained capacity availability ensures that no conflict arises between schedules for any two or more products. The biggest drawback of the single item problem is that it represents very few real life production systems. On the other hand, the theory developed in solving these problems can be extended and applied to more complicated problems.

Multi-Item Uniform Demand Problems

This class of problems is represented by the notation c/1/0/N/SI/SI, and are popularly known as ELSP. When uniform demand exists and production is cyclical EMQ offers the optimal lot size for a single item problem with uniform demand. However, when this lot size is used in practice it is possible that the resulting schedule is not feasible. Infeasibility arises when the sum of the production times for products is greater than cycle time for one or more of the products in the system. Hsu (1983) proves that determining the lot sizes that are optimal as well as feasible is an NP-hard problem. There are two popular approaches to solving the ELSP problem. One is the common cycle approach first proposed by Hanssmann (1962) and the other is the basic period approach discussed by Bomberger (1963). The problem of determining the feasibility of a given data set is addressed by Davis (1990) and the sequencing issues that arise in ELSP solutions is addressed by Kim and Mabert (1995).

In the common cycle approach the cycle time is assumed to be equal for all products. This cycle time is made long enough to accommodate the production of each

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

item exactly once. The only remaining problem is to determine the optimal value of this common cycle. However, the solution obtained by common cycle approach is not optimal for the original problem. In the basic period approach each item can have different cycle times, the only restriction being that the cycle time be a integer multiple of a basic period. In this approach values of basic period and the integer multipliers of each product have to be computed. Elmaghraby (1978) has reported that the basic period approach offers better solutions than common cycle approaches.

ELSP is widely used in all MRP II software packages. If the demand pattern is close to uniform then its use can be justified. However, in many situations ELSP solutions are used even in the presence of dynamic demand. A case in point is the production facility described in the motivating case. The aim of the current research is to obtain lower cost schedules than that obtained using MRP II in the presence of dynamic demand. However, the effort involved in generating a feasible ELSP schedule for a given set of demand and production data is beyond the scope of this research. Hence, for comparison purposes the lower bound of the ELSP schedule costs were used.

Elmaghraby (1978) and Davis (1990) have shown that the lower bound to the cost of a schedule is obtained by producing in lot size of EMQ for each item. The cost of the schedule can only increase by adjusting the lot size in an attempt to obtain feasibility (either using common cycle or basic period). Since MRP II systems use ELSP solutions it is more than fair that the schedule costs obtained using EMQ calculations are used to represent the cost of MRP II schedules. This is specially significant because, if the proposed methods produce lower cost schedules than one obtained using EMQ

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

calculations it can be claimed that the proposed methods provide a better way to handle the scheduling problem than MRP II. The cost calculations and sequencing issues are addressed in detail in chapter IV.

The biggest drawback of the ELSP formulation is that it assumes a uniform daily demand rate. In batch manufacturing environment this assumption is rarely true. To overcome this problem researchers included the actual demand patterns requested by customers into the problem formulation. This category of research is reviewed in the next section.

Multi-Item Dynamic Demand Problems

Dynamic demand problems are also referred to as finite scheduling problems. These problems consider demand occurring over a finite scheduling horizon unlike the uniform demand problems that consider a uniform daily demand occurring over a infinite horizon. The original formulations of these problems are represented by the notation d/1/0/N/SI/A. In a comprehensive review Salomon (1991) found that three types of problems have been addressed in the literature.

1. Capacitated Lot Sizing Problem (CLSP): Here multiple items are produced in a period but no setup is allowed to be carried over to the next period.
2. Continuous Setup Lot Sizing Problem (CSLP): This problem allows for setup to be carried from one period to another but restricts production to one item per period.
3. Discrete Lot Sizing Problem (DLSP): This problem is similar to the CSLP problems except that quantity produced in each period is either zero or equal to the full capacity.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Capacitated Lot Sizing Problem (CLSP)

This is by far the most common type of multi-item dynamic demand problem addressed in the literature. Mathematically, CLSP is formulated as follows:

$$MIN \sum_{i=1}^{N} \sum_{t=1}^{T} (A_i w_{it} + h_i I_{it}) \tag{27}$$

subject to

$$I_{it-1} + x_{it} - d_{it} = I_{it} \qquad \forall\, i, t \tag{28}$$

$$\sum_{i=1}^{N} x_{it} \le C_t \qquad \forall\, t \tag{29}$$

$$x_{it} \le (\sum_{k=t}^{T} d_{ik}) w_{it} \qquad \forall\, i, t \tag{30}$$

$$x_{it}, I_{it} \ge 0 \qquad \forall\, i, t \tag{31}$$

$$w_{it} \in \{0, 1\} \qquad \forall\, i, t \tag{32}$$

The objective function, expressed by equation (27), minimizes the sum of setup and inventory costs across all products and all periods. Equation (28) is the same as equation (7) and is the inventory balance equation. Equation (29) introduces capacity constraints and equation (30) limits production in a period to only the future requirements. Equations (31) and (32) place limits on the range of acceptable values for the variables. Chen and Thizy (1987) have proven that CLSP is an NP-Hard problem except when all setup costs are assumed to be zero. Two approaches have been used to obtain good solutions to the problem. These are:

1. mathematical programming based approach, and
2. heuristic "common sense" approach.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The problem reduces to a single item problem if capacity constraint expressed by equation (29) can be relaxed. Elimination of this constraint allows the problem to be solved as a series of inter linked single item problems. A Lagrangean relaxation based approach is used by Thizy and Van Wassenhove (1985) to reduce the problem to a transportation problem which can be solved optimally. However, in order to ensure the feasibility of the final solution they use a sub-gradient optimization technique to calculate the values of the Lagrangean multipliers. While their approach does not guarantee optimality the primary contribution of their work is to suggest a method to obtain good lower bounds to the original problem. A similar relaxation approach that relaxes the demand constraints is presented by Chen and Thizy (1987). An approach that relaxes the integer requirements of the solution is suggested by Maes and Van Wassenhove (1986). The decimal solutions are rounded off using different techniques to attain feasibility. Another approach based on column generation is discussed by Cattrysse et al. (1990). Finally, a procedure that solves the problem for few products at a time is presented by Kirca and Kokten (1994). Subsequent subsets are limited in capacity availability and their inventory carrying capability, and this ensures feasibility of solution over the entire set of products.

Most of "common sense" approaches involve the following features:

1. a method to prioritize products,
2. a rule to allocate limited production capacity, and
3. a feasibility routine.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

A forward pass algorithm is presented by Eisenhut (1975). Production lots for period 1 are based on marginal cost coefficients, determined using equation (33) shown below.

$$U_i(t) = \frac{A_i - h_i \sum_{k=1}^{t} (k-1) d_{ik}}{t^2 \times d_{it}} \tag{33}$$

Equation (33) represents the rate of change in per period costs incurred by including demand in period t into the current lot. At lower values of t large reductions in per period costs can be expected however, as t increases the per period costs decrease at a lesser rate and finally begin to increase. The lot sizes for period 1 are found by increasing t one period for that product which shows the greatest potential savings for each additional unit pulled into the current lot. This is repeated until capacity constraints are violated or no additional cost savings are possible for any product. Then the time axis is re-labeled so that the current period is period 1. The drawback of this approach is that it may assign small lots for production in the earlier period and this leads to infeasibility in the latter periods. A feasibility check provided by Dixon and Silver (1981) for this approach is limited by the fact that it only specifies total production in each period but does not determine the size of lots for individual products for feasibility.

A forward pass algorithm with a backtrack routine for feasibility is discussed by Lambrecht and Venderveken (1979). They use a different equation for calculating the marginal cost coefficients which is represented by the equation shown below

$$U_i(t) = \frac{A_i + h_i \sum_{k=1}^{t-1} (k-1) d_{ik} - h_i \times (t-1)^2 \times d_{it}}{t(t-1) \times d_{it}} \tag{34}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Equation (34) is obtained by comparing the marginal cost of including demand in period t into current production to the cost of including demand in period (t-1). A positive value of $U_i(t)$ indicates that inventory related costs can be reduced by including demand in period t into current production. If at any point in their forward sweep they find that capacity is not sufficient to meet remaining demand they go back to previous periods and increase production lot assignments of earlier periods.

An alternative approach to ensure feasibility is presented by Dixon and Silver (1981). They perform a look ahead computation for cumulative production requirements up to period t (for all t) so that no infeasibility will arise in period t+1. That is, the production in period t must exceed the total amount demand exceeds capacity in all future periods. Their completely unidirectional approach sometimes requires inclusion of production lots that have negative marginal savings coefficients.

Production in the later periods can be straitjacketed by the commitments in earlier periods in the above approaches. To avoid this, an approach that first considers the periods that provide maximum savings is proposed by Dogramaci et al. (1981). Initially the lot sizes are assumed to be equal to demand in that period. This is followed by shifting of production first to reduce costs and then to achieve feasibility. The algorithm is a four step algorithm that will find a feasible solution if one exists.

Comparison of the performance of these heuristics is studied by Maes and Van Wassenhove (1986). The results indicate that the Dogramaci heuristic outperforms the others when tight capacity constraints exist, i.e., capacity utilization of resource is high. When a definite trend in demand can be established then the look ahead approach of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Dixon and Silver out performs the other two. Time variability and lumpiness was found to have little effect on the performance of these heuristics.

Continuous Setup Lot Sizing Problem (CSLP)

Mathematically the CSLP is formulated using the mixed integer program approach:

$$MIN\sum_{i=1}^{N}\sum_{t=1}^{T}(A_i \times \max(0, w_{it} - w_{it-1}) + h_i I_{it} \tag{35}$$

subject to

$$I_{it-1} + x_{it} - d_{it} = I_{it} \qquad \forall\, i, t \tag{36}$$

$$\sum_{i=1}^{N} x_{it} \le C_t \qquad \forall\, t \tag{37}$$

$$x_{it} \le (\sum_{k=t}^{T} d_{ik}) w_{it} \qquad \forall\, i, t \tag{38}$$

$$\sum_{i=1}^{N} w_{it} \le 1 \qquad \forall\, t \tag{39}$$

$$x_{it}, I_{it} \ge 0 \qquad \forall\, i, t \tag{40}$$

$$w_{it} \in \{0, 1\} \qquad \forall\, i, t \tag{41}$$

The objective function in equation (35) reflects the potential cost savings due to setup carryover. This formulation states that when the same product i is produced in period t-1 and t ($w_{it} = 1$ and $w_{it-1} = 1$) then there is no setup cost incurred for production in period t. Equations (36), (37) and (38) are the by now familiar inventory balance, capacity constraint, and lot size limitation equations respectively. Equation (39) limits production to a single item in a period. Equations (40) and (41) control the possible

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

range of values for the variables. CSLP has been proved to be an NP-Hard problem by Florian et al. (1980). However, fairly large problems have been solved with reasonable computational effort. This indicates that CSLP is more tractable than CLSP and hence no heuristic approaches are required to solve the standard CSLP problem. The optimal solution is obtained by a heuristic based approach upon on the relaxation of the capacity constraints and is reported by Karmarkar and Schrage (1985).

Discrete Lot Sizing and Scheduling Problem (DLSP)

The only difference between the DLSP problem and CSLP problem is that, in DLSP the quantity produced in each period is either assumed to be zero or equal to the full production capacity.

$$MIN \sum_{i=1}^{N} \sum_{t=1}^{T} (A_i \times \max(0, w_{it} - w_{it-1}) + h_i I_{it} \tag{42}$$

subject to

$$I_{it-1} + p_i w_{it} - d_{it} = I_{it} \qquad \forall\, i, t \tag{43}$$

$$\sum_{i=1}^{N} w_{it} \le 1 \qquad \forall\, t \tag{44}$$

$$I_{it} \ge 0 \qquad \forall\, i, t \tag{45}$$

$$w_{it} \in \{0, 1\} \qquad \forall\, i, t \tag{46}$$

The objective function (42) is the same as the one used in CSLP formulation. The inventory balance equation (43) is modified to reflect the new restriction on the size of production lots in a period. Equations (44), (45) and (46) are same as (39), (40) and (41). The DLSP formulation does not use the x_{it} decision variable because it considers only all or nothing situations. Fleischmann (1990) presents a branch-and-bound

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

procedure using Lagrangean relaxation for determining the lower bounds and feasible solutions to a given problem.

The original formulations for multi-item lot sizing under dynamic demand were reviewed in this section. Most of the researchers have focused their attention on the CLSP formulation. This is because a vast majority of real world dynamic lot sizing and scheduling environments reflect the attributes included in this model. To improve the applicability of these solution procedures several extensions have been proposed, and they will be reviewed in the next section.

Extensions of Original Formulations of Multi-Item Dynamic Demand Problems

A combination of CLSP and CSLP problem formulations with setup times, /d/1/0//N/A/SI, is considered by Aras and Swanson (1982). Depending on the sequence of production, if a product is produced in an adjacent period, then it is possible to avoid setups and thereby not lose capacity. The authors propose a "common sense" practical approach to solve the problem. The biggest drawback of this formulation is that it does not consider setup costs and only the cost of inventory is reduced and also sequencing is limited to first and last products in a period. A similar formulation which includes setup costs has been solved for small problems Gopalakrishnan et al. (1995). These authors formulate the problem as a mixed integer linear program (MILP), and solve a problem with five products over 12 periods optimally. The applicability of their approach to much larger problems was not discussed.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The CLSP formulation has been extended to include setup times, d/1/0/N/SI/SI, by Trigeiro et al. (1989) and Trigeiro (1989). Consideration of setup times increases the complexity of the problem, for example determining if a given data set has a feasible solution is NP complete. Trigeiro (1989) has proposed an approach similar to the one used Lambrecht and Vanderveken (1979) with setup time considerations. Then a multi-pass algorithm is used to shift production for cost reductions and to attain feasibility. Trigeiro et al. (1989) propose a mathematical approach that uses Lagrange multipliers to calculate the lower bound, in the second stage of the algorithm a feasibility routine is implemented. Work done by Trigeiro (1989) and Trigeiro et al. (1989) show that solution approaches that were developed for zero setup time environment can be successfully modified and used in the presence setup times.

A mathematical approach to solving the DLSP problem with setup times is discussed by Cattrysse et al. (1993). Here the DLSP is formulated as a set partitioning problem and a dual ascent column generation heuristic is used to solve the problem. The heuristic generates both lower and upper bounds for the problems which allows for the measurement of solution quality.

The DLSP problem with sequence dependent setup costs, d/1/0/N/SD/A is studied in Fleischmann (1994). Here, the authors use a traveling salesman formulation of the problem with time windows to determine the lower bounds to the problem. An alternative approach to solve the problem involves the use of various estimating methods to convert the sequence dependent setup costs to sequence independent costs. Six such approaches are tested by Dilts and Ramsing (1989) across a spectrum of problem

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

parameters. The authors found that simple rules to estimate the setup costs are superior to the more complex rules.

This completes the review of literature. The results obtained by previous researchers has a direct bearing on the solution approach proposed for this study. A summary of literature review and implications of this review on the current research are discussed in the following section.

Summary

From the above discussion it can be said that significant advances have been made in accurately solving the simpler problems. However, approaches to obtain good solutions to more complicated problems have not been widely studied. Specifically, none of the researchers have considered all the problem attributes presented in chapter I simultaneously. The assumptions made in the past while proposing solution methods to problems were not always realistic and this hinders the widespread use of finite scheduling methods in the real world. On the other hand this research is focused on a very significant scheduling problem in the real world. Substantial differences exist between the scheduling problem addressed here and those proposed and studied in the past. However, the theory that has been developed by previous researchers can be used to solve the current problem.

The current problem is basically an extension of a combination of the original CLSP and CSLP problem formulations, and can be represented by the notation d/1/v/N/SD/SD with setup carryover. The formulation that comes closest to the one

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

researched here was presented by Gopalakrishnan et al. (1995). However, there are key differences between their formulation and the one addressed here.

1. Variable backlogging is allowed in our formulation, no backlogging is allowed in their formulation.
2. Sequence dependent setup costs and setup times are incorporated here. Only sequence independent setup cost and times are used in their formulation.
3. Their solution procedure restricts the size of the problem that can be addressed. however, even large problems can be solved using the approach presented here.

Although a more difficult problem is addressed here the method proposed here takes advantage of several theoretical foundations laid by previous researchers. The contributions most relevant to the problem addressed here are:

1. research that deals with complexity of different types of dynamic demand problems,
2. optimal solution to the single item dynamic demand problem with backlogging, and
3. solution approaches to the traveling salesman problem to sequence the production.

While the first two contributions are in the area of finite scheduling the third topic is a difficult open problem in the realm of operations research. Contributions of the first category are important because they provide the basis for showing that the problem addressed here is a difficult problem, and justify the use of heuristics to solve this problem is justified. The second and third categories are important because the algorithms developed to solve the research problem are based on these concepts. Hence, in the next chapter the computational complexity of the research problem is discussed

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

followed by a review of theory behind essential concepts that form the crux of the algorithms developed to solve the problem.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER III

THEORETICAL FOUNDATION

The primary aim of this chapter is to discuss the implications of theoretical concepts developed by previous researchers on the solution approach proposed here. Before a solution approach can be proposed for the research problem it is essential that the computational complexity of the problem be addressed. If it can be shown that the problem addressed here belongs to a class of NP-Complete problems then this justifies the use of heuristics to solve the problem. In the first part of this chapter, it will be shown that the research problem addressed here justifiably belongs to the class of NP-Complete problems. This justifies the use of heuristic algorithms to solve the problem. The algorithms proposed for the solution of the research problem are theoretically grounded on certain principles developed by previous research in the area of lot sizing in dynamic demand environments and other related operations research problems. These principles include the optimal solution to the single item problem with backlogging (d/0/v/1/TD/A) and a heuristic solution to the traveling salesman problem.

Computational Complexity of Research Problem

The computational complexity of a combinatorial problem is related to the computational behavior of most efficient algorithms designed to obtain its optimal solution. This behavior is measured by the running time of the algorithm i.e., that is the number of elementary operations such as additions and comparisons required to solve the problem. This running time is related to the size of the problem addressed i.e., the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

number of bytes occupied by the input data. If a problem of size n can be can be solved by an algorithm that is polynomial function of n (example: n, n^2, n^3,..) then the problem can be considered to be well solved. Polynomial algorithms have been developed for a few types of combinatorial optimization problems (Lawler 1976). However, a vast majority of such problems can only be solved by enumerative methods which may require exponential time.

A problem is said to be NP-Complete if it can be proved that no polynomial solution exists for the problem. However, such a proof will not be attempted here. Instead, it is conjectured that the research problem is NP-Complete based on proven results published by others. It has been well established that the single item CLSP is a NP-Complete problem by Florian et al. (1980). This result has been extended to cover multi-item CLSP (Chen and Thizy 1987) and CSLP (Florian et al. 1980). The research problem can be reduced to a combination of CLSP and CSLP problems if no backlogging is assumed ($\alpha = 0$). This means that the reduced problem is at least as hard as CLSP or CSLP. The solution space for the problem is only increased by consideration of backlogging, which does not make the problem any easier. Hence, it can be stated with reasonable certainty that the problem addressed here is NP-Complete and that no polynomial time algorithms exist to determine the optimal solution to the problem.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Single Item Problem with Backlogging

Optimal solutions to this problem was developed in the mid 60's by Zangwill (1966, 1969) and Manne and Veinott (1967). For an arbitrary product i the problem that they addressed can be formulated by the following equations:

$$MIN \sum_{t=1}^{T} (A_i w_{it} + h_i u_{it} + b_i v_{it}) \tag{47}$$

subject to

$$u_{it} = \max(I_{it}, 0) \qquad \forall\, t \tag{48}$$

$$v_{it} = \max(-I_{it}, 0) \qquad \forall\, t \tag{49}$$

$$I_{it-1} + x_{it} - I_{it} = d_{it} \qquad \forall\, t \tag{50}$$

$$I_{it} \geq - \sum_{k=t-\alpha+1}^{t} d_{ik} \qquad \forall\, t \geq \alpha \tag{51}$$

$$x_{it} - M(w_{it}) \leq 0 \qquad \forall\, t \tag{52}$$

$$I_{i0} = I_{iT} = 0 \tag{53}$$

$$u_{it}, v_{it}, x_{it} \geq 0 \qquad \forall\, t \tag{54}$$

$$w_{it} \in \{0, 1\} \qquad \forall\, t \tag{55}$$

The objective function (47) minimizes the sum of setup, inventory and backlogging costs. Equations (48) and (49) ensure that appropriate cost multipliers are used in the objective function. Equation (50) is the familiar inventory balance equation. Equation (51), similar to equation (8), constrains the maximum backlogging allowed. Equation (52) makes certain that a setup cost is assessed every time a production lot is scheduled. Significant differences between this formulation that studied in this research are listed below:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. Single item vs. multi-item.
2. Unconstrained capacity availability vs. finite capacity availability.
3. No setup time vs. significant sequence dependent setup times.
4. No setup carryover vs. setup carryover allowed.
5. Pre-defined setup costs (may be time variant) vs. sequence dependent setup costs.

The scheduling problem with additional attributes considered in this research can be solved by superimposing the additional attributes on the optimal solution algorithm to the formulation presented above. The authors propose a dynamic program based solution procedure to optimally solve the above formulation. To reduce the solution space that must be searched for the optimal solution, a theorem called the regeneration point theorem is used.

For an arbitrary product i, a period t is called a point of regeneration if inventory at the end of the period is zero ($I_{it} = 0$). A schedule is said to have a regeneration point property if;

1. Between any two periods in which production occurs, there exists one and only one regeneration point. This can be formally stated using mathematical terms, for an arbitrary product i; given t<k and $x_{it} > 0$; $x_{it+1} = x_{it+2} = \ldots = x_{ik-1} = 0$; *and* $x_{ik} > 0$ there exists a period t*, $i \leq t^* \leq k -$, which is a regeneration point.
2. Between any two regeneration points there exists only one period with production greater than 0. Formally for an arbitrary product i, given t < k and $I_{it} = I_{ik} = 0$ there exists a period t', $t \leq t' \leq k$ such that

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$x_{it''} = \begin{cases} \sum_{k'=t+1}^{k} d_{k'} & \text{if } t''=t' \\ 0 \text{ otherwise} \end{cases} \tag{56}$$

Equation (56) states that all demand between periods t and k is met by production in period t' and that all other periods in that durations have no production. The importance of the regeneration point property lies in the usefulness of the regeneration point theorem which states that "There is an optimal schedule which has the regeneration point property". Proof of this theorem is available in the original papers. The dynamic programming algorithm proposed by the authors works as follows:

Step 1. Calculate $c_{tk}^{t'}$ the cost of producing a lot in period t' to meet all demand between the regeneration points of t and k

$$c_{tk}^{t'} = A_i + h_i \sum_{t''=t'+1}^{k} (t''-t')d_{it''} + b_i \sum_{t''=t+1}^{t'-1} (t'-t'')d_{it} \qquad \forall\, t, t', k \tag{57}$$

Equation (57) states that the cost of producing a lot in period t' to meet demand for periods (t+1) to k is the sum (1) of setup cost in that period for that product, (2) the inventory cost for producing demand for periods (t'+1) to k in period t' and (3) the cost of backlogging demand in periods (t+1) to (t'-1).

Step 2. Calculate $c_{tk} = \min_{t+1 \le t' \le k} c_{tk}^{t'}$ and save the value of t' that provides the lowest cost.

In this step, the best period to produce in order to meet demand between periods (t+1, k) is identified and the cost of the having regeneration points in periods t and k is computed.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 3. Determine the next regeneration point that minimizes the cost of going from the current regeneration point to regeneration point T. For this the cost of all possible cost combinations from the current regeneration point to T are computed and compared. Formally stated:

$$f_t = \min_{t<k\leq T} (c_{tk} + f_k) \quad t \in (0, T-1) \; and \; f_T = 0 \tag{58}$$

For each regeneration point t equation (58) determines the next regeneration point (k) that minimizes the inventory related production costs. The duration between two successive regeneration points determines the size of the production lot that is produced between the points, and is equal to the total demand in that duration.

Step 4. Forward pass to determine optimal production lots. Given a regeneration point in period 0 then the next best regeneration point can be identified using step 3. Subsequently regeneration points can now be determined by values saved from step 3. This is repeated until the regeneration point in T is reached.

The four step algorithm described above determines the best periods of production to meet a known deterministic dynamic demand in the presence of backlogging. This algorithm is at the crux of some of the methods proposed in chapter IV to solve the research problem addressed here. While the above algorithm can be used to determine the size of the production lots, it is also essential to determine the sequence of production. In this regard, the traveling salesman problem addressed in the next section plays a crucial role.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Traveling Salesman Problem (TSP)

TSP is a network problem to determine the least cost sequence of nodes to visit so that each node in the network is visited exactly once and the trip ends in the starting node. The TSP formulation is widely applied in the real world to determine the sequence of travel between cities for people and places. In this context, the cost to be minimized is the travel cost between cities. In this study, TSP formulation is used to minimize the sum setup costs across the horizon incurred when switching production between products. TSP is a NP-Hard problem. However there are several algorithms that take advantage of the nature of the cost matrix to determine optimal solutions to fairly large problems within reasonable computational effort. Two classes of heuristics have been proposed in the literature to solve the TSP problem (Mathur and Solow 1994):

1. Tour construction heuristics: build a tour (sequence) by including sequentially one node at a time until all nodes are used.
2. Tour improvement heuristics: start with a given tour and attempt to construct better tours with progressively less total cost.

A sequence construction heuristic, also called the cheapest-insertion algorithm, (Mathur and Solow 1994) is used in this research and detailed below. The basic idea is to start with a sequence of products and sequentially create larger sequences by:

1. selecting a product to be included into the sequence, and
2. determining where to insert the selected product.

For every product that is not yet included in the sequence the algorithm calculates the best point of insertion for that product. The product that provides the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

least additional increase in the setup cost of the sequence is selected for insertion at its lowest cost insertion point. The TSP sequencing heuristic is listed below.

Initialize the TSP sequence:

$\phi_1 = i,\ \phi_2 = j \ni S_{ij} + S_{ji} = \underset{i',i''\in\psi}{MIN}\,(S_{i'i''} + S_{i''i'})$

k=2

ψ = all products but i and j

while ($\psi \neq \emptyset$)

{

$$\phi_{k+1} = \phi_1 \tag{59}$$

$$\phi_m = j \ni S_{\phi_{m-1}j} + S_{j\phi_m} = \min_{i\in\psi}\ \min_{l\in 2,k+1}\,(S_{\phi_{l-1}i} + S_{i\phi_l}) \tag{60}$$

k++

remove j from ψ

}

Once the two initial products have been identified there exists two possible insertion points. The third product can either be inserted between ϕ_1 and ϕ_2 or between ϕ_2 and ϕ_1. Similarly, in subsequent insertion point considerations insertions between the last product and first product must be considered. To accommodate this possibility equation (59) creates a dummy position at (k+1) and sets the product at this position equal to one in position 1. Equation (60) identifies both the best product, that has not been inserted into the cycle yet, and the point of insertion of the product. For each item, it calculates the cost of the sub-sequence consisting of its two adjacent nodes and thus

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

identifies the best insertion point for the product if it offers the lowest insertion cost amongst all products.

Once the node with the lowest insertion cost is identified then the tour is updated to reflect this new insertion. The tour is updated by merely incrementing the positional value of the nodes after the new insertion points by one. If the insertion point is between k and (k+1) then the new insertion point is reset to 1. The length of the tour is incremented and the selected product removed from the list of un-inserted nodes. The above procedure is repeated until all products are inserted into the sequence. While, this algorithm does not determine the optimal sequence of production, it provides a good approximation to the optimal solution. The context in which the sequencing issue arises in solving the research problem justifies the use of a heuristic.

Summary

In this chapter, three theoretical concepts that play a large part in the functioning final algorithms have been presented. The complexity of the research problem was compared with other lot sizing problems addressed previously. A dynamic programming approach to optimally solve the single item problem with backlogging was described. Finally a TSP heuristic that plays a large role in sequencing production is detailed.

Research methodology consists of determining the heuristic algorithms that can be used to solve the formulated problem. These heuristic algorithms used to generate schedules for research problem must address the issues of lot sizing and sequencing. No solution procedure has been developed yet to determine lot sizes in the presence of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

capacity constraints. However, the regeneration theorem can be used to optimally determine lot sizes in the absence of capacity constraints. Therefore, the regeneration theorem is incorporated into the heuristics proposed here. The cost of the final schedule is dependent on the sequence of production (due to sequence dependent setup considerations). The TSP algorithm described in this chapter is used to determine the sequence of production of production lots.

Thus far a foundation has been laid to detail the methodology used to successfully address the research problem. The primary purpose of research methodology is to unequivocally layout the sequence of steps that has been used to solve the research problem. Research methodology developed to solve the research problem is complicated and unique. Chapter IV deals with the details of this methodology.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER IV

RESEARCH METHODOLOGY

The research problem addressed here is NP-Complete, which precludes the use of exact approaches to solve problems of the size that exist in the real world. It is for this purpose heuristic approaches are proposed to address the problem. The aim of each heuristic approach is to determine the production lots and sequence of production for each item in each period that minimize the sum of inventory, backlogging and setup costs. For a solution to be accepted, it must be feasible. For a schedule to be feasible, it must satisfy the following two constraints:

1. capacity requirements in each period must be less than or equal to capacity available in each period, i.e., equation (18) of chapter I must be satisfied, and
2. the maximum delay criteria must be satisfied, this ensures that orders are not backlogged more than α days as represented in equation (8) of chapter I.

The original problem formulation has too many variables and constraints to be effectively considered in tandem. To facilitate solving the problem, the strategy adopted here is to fragment the original problem, solve the individual parts, and then combine the parts to obtain a solution to the original formulation. Any solution approach to the research problem must address three issues:

1. determine the size of production lots for each product in each period,
2. sequence the production lots within each period and across the schedule horizon, and
3. convert an infeasible schedule to a feasible schedule without unduly affecting the schedule costs.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Each of the above is a substantial problem on its own merits. However, the situation is further complicated by the inter-relationships between the problems. These are not three separate problems; on the contrary, they are intertwined because:

1. the feasibility of a schedule is linked to the size of the production lot for each item in each period and the setup time required for each item (this is dependent on the sequence of production),
2. the determination of lot sizes depends on capacity considerations (a feasibility requirement), and since setup times are sequence dependent, capacity requirements cannot be accurately determined until the sequencing issue is resolved, and
3. sequencing is a function of non zero production lots scheduled in each period which in turn depends on the capacity availability in each period.

Hence, any approach to solving the research problem must iteratively solve the lot sizing problem, the sequencing problem and also address the feasibility issue. In the past, when a new formulation of a dynamic demand lot sizing problem had been addressed two broad ways have been proposed to handle the problem:

1. develop a "common sense" approach to solve the problem, and
2. use optimal solutions to sub-problems and adapt these solutions to obtain feasible solutions to the current problem.

In this research, one extension of the first approach and two variations of the second approach have been developed resulting in a total of three new algorithms. The three algorithms essentially differ in their approach to addressing the lot sizing issue. The initial lot sizing problem is handled in one of the following ways.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. The "common sense" approach is a lot shifting procedure similar to one used by Dogramaci et al. (1981).
2. The first variation of optimal solutions for sub-problems uses the regeneration theorem to solve the single item lot sizing problem with backlogging optimally for each item.
3. The second variation, of optimal solutions for sub-problems, solves the single item lot sizing problem with backlogging for each item with a "overload penalty" between successive items to reduce infeasibility of the final solution. This is a more intelligent application of this approach than algorithm 2.

The three algorithms differ primarily in the way in which they approach the initial lot sizing problem. Other parts of the algorithms are similar in the three cases, and for this reason the overlapping features of the algorithms will be addressed first. The overlapping features are; sequencing method, capacity requirement calculations, procedure to eliminate infeasibility, and a optimality condition test. The solution to the sequencing problem is founded upon the TSP heuristic discussed in the previous chapter. The details of the sequencing problem and the solution to the problem are described in the next section.

The Sequencing Problem

The sequencing problem consists of determining the optimal sequence for production of lots within each period and across the scheduling horizon. A dynamic programming methodology adopted to solve the sequencing problem is detailed here. A

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

dynamic program is defined by stages, states and policy costs. The cost of a sequence is only dependent on sequence dependent setup costs, and not inventory/backlogging costs. In this case, each day is a stage, and states within a stage are uniquely defined by the first and last products produced on that day. For example, if there are n products scheduled on a day then there are ${}^{n}_{2}P$ possible states in that stage. While the 2 products produced first and last identify the state there are, (n-2) other products that have to be incorporated into the state between the first and last products. Each item incurs a setup cost based on the sequence of production. The policy cost for a state is the minimum cumulative setup cost associated with the state. Furthermore, this cost depends on the sequence of production of the (n-2) products between the first product and last product.

For example, if items a, b, c and d have non zero lots scheduled in a day. Then, there are ${}^{4}_{2}P = 12$ possible states in that stage. Representing each state by the pair of first and last items produced in the stage we get twelve pairs as follows; { (a,b), (a,c), (a,d), (b,a), (b,c), (b,d), (c,a), (c,b), (c,d), (d,a), (d,b), (d,c) }. For the state represented by the pair (a,b), two possible sequences are possible (1) a→c→d→b and (2) a→d→c→b. The cost of sequence 1 is $S_{ac} + S_{cd} + S_{db}$ and the cost of sequence 2 is $S_{ad} + S_{dc} + S_{cb}$. The cost of the state (a,b) is the one that gives the lowest cost of all possible sequences. Hence, for each state in each stage the best sequence between the first and last items must be solved to determine the cost of the state.

This problem is similar to a TSP introduced in chapter III, except that we have our starting and ending nodes pre-defined. A TSP like problem addressed here is called the pseudo TSP. Pseudo TSP, like the original problem, is a NP complete problem

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(Lawler et al. 1985) and optimal solution is difficult to calculate when the number of products becomes large. Typically, very few products are scheduled for production in a day. In such instances, the pseudo TSP can be solved by complete enumeration. n! possible solutions must be compared if complete enumeration is used. However, as n becomes large, n! becomes untractable and a heuristic procedure is used to solve the problem. In this research, the sequencing issues within each state are addressed as follows:

1. when the number of products scheduled for a day is ≤ 7, the TSP problem is solved optimally using complete enumeration, and
2. when the number of products scheduled for a day is > 7, the TSP problem is solved using a tour construction heuristic described in chapter III.

To solve a TSP problem of size n by complete enumeration the cost of all n! possible combinations must be computed and compared. Thus, when 7 products are scheduled in a day the size of the TSP problem is 5, since the first and last products are already defined. This calls for 5! = 120 sequences to be compared and selection of the lowest sequence amongst all possibilities.

The use of a heuristic for larger problems is justified because, in a real life batch production environment, seldom does one encounter more than 7 products scheduled for production in a day. This is true even if the total number of products in the system is much greater than 7. However, sequencing greater number of products may be required in the early stages of problem solving. It is for this purpose that the heuristic approach is mostly used. The sequencing algorithm detailed below is a two phase algorithm. In the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

first phase the cost of all possible states are calculated by solving a pseudo TSP. The second phase invokes a dynamic programming algorithm to calculate the lowest cost sequence across the scheduling horizon.

Phase 1: Cost of States

A state is defined by the first and last products sequenced in a period. These two products are passed to this phase of the algorithm by the dynamic programming procedure described in phase 2. Now the pseudo TSP is reduced to determining o_{kt} for k = 2,. . ., n_{t-1} given o_{1t} and $o_{n_t t}$. The TSP solution procedure used is based on the value of n_t. One of the following procedures is used to determine the sequence and cost of the sequence.

1. If $n_t \in [0, 1]$[1] then $C_{TSP} = 0$
2. If $n_t = 2$ then $C_{TSP} = S_{o_{1t}\, o_{2t}}$
3. If $n_t > 2$ and $n_t < 8$ then $C_{TSP} = \min_k(\Theta_k)$ where k = 1, 2, ..., ${}^{n_t}_{\ 2}P$,

   $\Theta_k = \sum_{i=2}^{n_t} S_{m_{i-1}^k m_i^k}$ where m_i^k is the ith product on the kth tour, also $m_1^k = o_{1t}$ and

   $m_{n_t}^k = o_{n_t t}$ $\forall$ k, t. All possible values of m_i^k are pre-computed. The complete enumeration routine determines all possible sequences of production. The procedure then compares the cost of each of these sequences with others to find a minimum cost sequence.

---

[1] if $n_t = 0$ then dummy product = $o_{1t} = o_{n_t t}$ (indicating a idle period)
if $n_t = 1$ then $o_{1t} = o_{n_t t} = i \ni x_{it} > 0$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4. If $n_t \geq 8$ then a modified tour construction heuristic similar to the one described in chapter 3 is used. Certain modifications are required to accommodate the fact that the initial and the final nodes are pre-determined. The basic idea however remains the same.

Initialize: $l=2$; $\phi_1 = o_{1t}$; $\phi_2 = o_{n_t t}$ (61)

while ($\psi \neq \varnothing$)

{

$$\phi_m = j \ni S_{\phi_{m-1}j} + S_{j\phi_m} = \min_{i \in \psi} \min_{k \in 2,l} (S_{\phi_{k-1}i} + S_{i\phi_k}) \quad (62)$$

l++

remove j from ψ

}

Equation (61) initializes the starting point and ending point of the sequence to the pair products that define the first and last products in period t. Equation (62) inserts a product j at position m such that this insertion causes the lowest possible increase in the sequence. This equation differs from equation (60) in that it does not allow insertion of products after the last product in the period. The procedure is repeated until all products have been positioned. The cost of the sequence is obtained using the equation (63) shown below.

$$C_{TSP} = \sum_{k=2}^{n_t} S_{\phi_{k-1}\phi_k} \quad (63)$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Phase 2: Dynamic Programming Algorithm

This phase of the sequencing algorithm incorporates the TSP procedure described above into a much broader dynamic programming algorithm. In this algorithm, the states within each period that provide the optimal sequence for the entire scheduling horizon are identified.

Initialize: $n_{T+1} = 1$, $n_0 = 1$, $PC_{T+1}^{1} = 0$, $fp_{T+1}^{1} = 0$, $lp_0^1 = 0$, $\theta_{T+1} = 1$

for (t = T to t = 0)

{

$$\sigma_t^k \ni PC_t^k = \min_{m \in [1, \theta_{t+1}]} PC_{t+1}^m + S_{lp_t^k fp_{t+1}^m} + C_t^k \quad \forall\, k \in [1, \theta_t] \qquad (64)$$

}

The initializations provide transition between the current scheduling horizon and adjacent horizons and also add completeness to dynamic programming algorithm at its boundaries. Equation (64) identifies, for each state k in the current stage t, the state in the next stage that gives the lowest cost sequence between the current stage and T. For example, if this state is m then m is the state that minimizes the sum of:

1. the lowest cost of getting from that state m in stage (t+1) to T,
2. the setup cost to switch production between the last product that defines state k and the first product that defines the state m in stage (t+1), and
3. cost of the sequence returned by the pseudo TSP algorithm between the items that define the first and last products of the sequence.

By progressing from the last stage to the first stage the algorithm successively finds the lowest cost between all the intermediate states and the final stage. Once the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

best state in the next stage has been identified, for each state, the problem now reduces to one of retrieving the best sequence. This is achieved as follows.

for (t = 1 to t <= T)

{

$$k \leftarrow \sigma_{t-1}^{k} \tag{65}$$

$$o_{jt} = \xi_t^k \;\; \forall j \tag{66}$$

}

In equation (65) the state k in period 1 that gives the lowest cost sequence between periods 1 and T is identified. Equation (66) then saves the sequence of items that defines state k. The algorithm successively identifies the best sequence in the next stage until stage T is reached. This algorithm is versatile in that it even accommodates intermediate periods during which no production lots are scheduled, using one of the conventions stated in chapter I.

The most important aim of the sequencing algorithm is to reduce the setup cost component of the inventory related production costs. Also, accurate sequencing is required to exactly calculate the capacity requirements for each day. Calculation of these capacity requirements once a sequence has been identified is detailed in the next section.

Calculating Capacity Requirements

With information available on production lot sizes for each product in each day and the best possible way to sequence the production, both across the horizon and within each day the capacity requirements to meet the schedule can be determined. If the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

capacity required is greater than capacity available then this indicates that the schedule is infeasible and modifications are required to make it feasible. The method used to calculate slack capacity and overloading is detailed below:

for (t=1 to t<=T)

{

$q_t = C_t$

$$q_t \leftarrow q_t - \sum_{i=1}^{n_t} x_{it} \tag{67}$$

$$q_t \leftarrow q_t - s_{o_{n_{t-1}t-1}o_{1t}} - \sum_{j=2}^{n_t} s_{o_{j-1t}o_{jt}} \tag{68}$$

}

Initially the slack on each day is set equal to capacity available. Equation (67) then reduces this value by an amount equal to the sum of all the production lots scheduled in that period. Equation (68) uses the sequences generated by the sequencing algorithm to further deduct setup time requirements to meet the schedule. Besides the setup time required for the products produced on that day the equation also takes into account the setup time required between the last product in the previous period and the first product in the current period. However, if the same product is produced on two successive days then it is possible that the same product is sequenced last on the previous day and first on the next day and thus avoid a setup. Once the capacity requirements are computed, it can be determined if the schedule is feasible or infeasible. An infeasible schedule is one in which there exists at least one period in which $q_t < 0$.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

To achieve feasibility either entire lots or parts thereof can be transferred to other periods with positive slack. This shifting of lots may affect both the setup cost and the inventory/backlogging costs of the schedule. Given a multiplicity of lot shifting options an infeasibility elimination procedure must try to achieve feasibility with lowest increase in the total schedule cost. Therefore, it is necessary to calculate the change in the total cost caused by shifting a lot to achieve feasibility. However, the change in setup costs cannot be calculated accurately because setup cost for a lot is dependent on the previous product in the sequence and when production is shifted the sequence changes which in turn changes the cost of the setup. To avoid this infinite looping the setup cost savings can be estimated. A similar problem arises for setup times when lots are shifted from one period to another. Capacity in the target period is checked for its ability to accommodate the lot. In addition, to lot size sufficient capacity must exist to allow for setup. Since sequence is not known an estimate of the setup time must be used. These estimating techniques and a method to accurately calculate the change in inventory/backlogging cost is described in the next section.

Calculating Cost Changes Caused by Production Shifting

First, the problem of estimating the changes in setup costs is addressed. To overcome the problem described in the previous section it is necessary to convert the sequence dependent setup costs and times for a product to sequence independent costs and times. Dilts and Ramsing (1989) describe various ways in which this might be achieved and also report that simple methods prove to be most effective. Especially,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

those that deviate most from the expected value of setup provide the best results. Accordingly, two estimating methods to be tried here.

1. MIN: assigns the minimum of all setup values possible for the product. This technique grossly underestimates the cost and time required for setup. Since the potential setup cost savings are underestimated, this technique tends to reduce the size of the initial lots. However, lower setup time requirements allow for greater mobility of lots in the feasibility elimination routine. $s_i = \min_j s_{ji} \quad \forall\ j \neq i$ and $S_i = \min_j S_{ji} \quad \forall\ j \neq i$.

2. MAX: assigns the maximum of all setup values for the product. This technique grossly over estimates the cost and time required for setups. Since the potential setup cost savings are greater than actual more lots are clubbed together initially producing large lot sizes. However, large setup time requirements affect the possibility of moving lots to other periods to achieve feasibility. $s_i = \max_j s_{ji} \quad \forall\ j \neq i$ and $S_i = \max_j S_{ji} \quad \forall\ j \neq i$.

While the changes in the setup cost can only be estimated the changes in the inventory/backlogging cost can be determined accurately. This change in cost must be computed with respect to the original demand requirements. Before the inventory/backlogging cost for product μ (H_μ) can be calculated inventory/backlogging for μ for each period ($I_{\mu t}$) must be calculated.

$$I_{\mu t} = I_{\mu t-1} + x_{\mu t} - d_{\mu t} \quad \forall\ t \in (1,T)\ and\ I_{\mu 0} = 0 \tag{69}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Equation (69) is obtained directly from constraint (7). This equation merely states that inventory at the end of a period is essentially the sum of inventory at end of previous period and production in the current period from which is subtracted demand in the current period. The inventory/backlogging cost for the product is calculated using equation (70) shown below. This equation is similar to the inventor/backlogging cost component of the objective function described in equation (4).

$$H_{\mu} = \sum_{t'=1}^{T} kh_{\mu} I_{\mu t'} - (1-k) b_{\mu} I_{\mu t'} \quad where\ k = \begin{matrix} 1\,if\ I_{\mu t'} > 0 \\ 0, otherwise \end{matrix} \tag{70}$$

When production is shifted from an overloaded period t to a period with slack capacity tg, the production lots of the affected periods are re-computed as $x'_{\mu t} \leftarrow x_{\mu t} - \eta$ and $x'_{\mu tg} \leftarrow x_{\mu tg} + \eta$. Now the inventory in each period is re-computed using the new production lots as $I'_{\mu t}$ using equation (69). Following which the new inventory/backlogging cost is re-computed as H'_{μ} using equation (70). Finally, the change in inventory/backlogging cost is given by equation (71).

$$\Delta H_{\mu} = H'_{\mu} - H_{\mu} \tag{71}$$

In this section, the methods used to calculate the changes in setup costs and inventory/backlogging costs caused by production shifting was described. These computations are widely used when lot shifting is considered to eliminate infeasibilities. To overcome infeasibilities either the setup for the first product must be moved to the previous period or production must be shifted to another period; this can be done in several ways. All possible options are evaluated before one is selected, the routine that eliminates infeasibility is discussed in the next section.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Eliminating Infeasibilities

Overloading may be detected in a period as a result capacity requirement calculations. These must be eliminated to make the schedule feasible. Sometimes this can be achieved without incurring extra costs or even at some savings however, in most occasions additional costs must be incurred. The production lot selected is the one that contributes most to eliminate overtime at the lowest cost or maximum savings. To achieve feasibility:

(1) setup time for the first product sequenced in that day can be moved to the previous period,

(2) production can be moved backward, or

(3) production can be moved forward.

Options (2) and (3) can be achieved in more than one way. All combinations are tried before selecting the option that provides the move with the lowest cost (maximum savings) per unit of overload eliminated. A brief overview followed by the complete details of the algorithm is presented below.

Overview of Algorithm to Eliminate Infeasibilities

Eliminating infeasibilities is a complex task because of a number of options have to be tried to determine the lowest cost option. A general overview of the approach adopted here is shown below. This approach first attempts to eliminate infeasibilities without additional costs if possible.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Initialize: $\nabla = \infty$

for (t=2 to $t \leq T$)

{

if ($q_t < 0$ & $q_{t-1} \geq |q_t|$ & $s_{o_{n_{t-1}t-1}o_{1t}} \geq |q_t|$) (72)

{

$$q_{t-1} \leftarrow q_{t-1} + q_t \tag{73}$$

$$q_t = 0 \tag{74}$$

}

}

for (t=1 to $t \leq T$)

{

while ($q_t < 0$)

{

$\nabla = \infty$

for ($i \in \Pi_t$)

μ, η, ∇, tg = MIN(production shifting) (75)

if ($\nabla = \infty$) (76)

exit( )

else

{

$$x_{\mu t} \leftarrow x_{\mu t} - \eta \tag{77}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$x_{\mu t g} \leftarrow x_{\mu t g} + \eta \tag{78}$$

re-sequence and calculate capacity

}

}

}

First an attempt is made to move the setup for the first product sequenced on a day to the previous period. Equation (72) ensures that this move is made only if infeasibility in the current period is eliminated. This makes certain that the setup time for the first product in the current period is not accounted for twice when the algorithm enters the while loop. An infeasibility in a period can be eliminated in this step only if two conditions are satisfied:

1. slack is available in the previous period, and this is at least as large as the overloading in the current period, and
2. the setup time required for the first product is at least as large as the overloading in the current period.

Once it is determined that infeasibility can be eliminated by shifting setup time, equation (73) recalculates slack in previous period and (74) eliminates the overloading in the current period. This method achieves feasibility without additional costs and is tried first for all periods with infeasibilities, there is no need to re-sequence after this step. If infeasibilities persist then the best combination of target period, quantity and item to move is determined by executing equation (75) for each item with a production lot in the overloaded period. If there exists a feasible target period in either the forward or

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

backward direction production shifting function will find them, when no feasible target periods exist equation (76) provides escape condition that prevents infinite looping. The size of the lots are reassigned in equations (77) and (78). This may require that the sequence and capacity availabilities to be re-computed. The production shifting function, tries both forward and backward shifts, is at the heart of eliminating infeasibility and it is described in complete detail next.

The Production Shifting Function

This function determines the item and the quantity to shift to the target period in the scheduling horizon that produces the lowest cost per unit of infeasibility eliminated. The routine is implemented for all t with infeasibility and all i.

for ($t' = 1$ to $t' \leq T$ && $t' \neq t$)

{

if ($x_{it'} > 0$ & $q_{t'} \geq x_{it}$) (79)

{

$$z = \frac{\Delta H_i - S_i}{x_{it}} \tag{80}$$

if ($z < \nabla$) (81)

$\nabla = z$; $\mu = i$; $\eta = x_{it}$; $tg = t'$; (82)

}

if ($q_{t'} \geq (x_{it} + s_i)$) (83)

{

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$$z = \frac{\Delta H_i}{x_{it}} \tag{84}$$

if ($z < \nabla$) (85)

$\nabla = z$; $\mu = i$; $\eta = x_{it}$; $tg = t'$; (86)

}

if ($x_{it'} > 0$ & $q_{t'} > 0$) (87)

{

$v = \min(|q_t|, x_{it}, q_{t'})$ (88)

$$z = \frac{\Delta H_i}{v} \tag{89}$$

if ($z < \nabla$) (90)

$\nabla = z$; $\mu = i$; $\eta = v$; $tg = t'$; (91)

}

if ($q_{t'} > s_i$) (92)

{

$v = \min(|q_t|, (q_{t'} - s_i), x_{it})$ (93)

$$z = \frac{S_i + \Delta H_i}{v} \tag{94}$$

if ($z < \nabla$) (95)

$\nabla = z$; $\mu = i$; $\eta = v$; $tg = t'$; (96)

}

}

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The algorithm detailed above shows four different alternatives available for shifting production. Each one is described in detail in the following paragraphs.

1. Move the entire lot to another period where the product is already produced; this eliminates a setup. Equation (79) ensures that this lot shift occurs only if there is sufficient capacity in the target period and that the product has a lot already scheduled in that period. As is illustrated in equation (80), the per unit cost is based on the difference between the change in the inventory cost calculated using equation (71) and the estimated reduction in the setup cost caused by the elimination of the setup in the current period. To encourage the shifting of the whole lot rather just enough to eliminate infeasibility the divisor in equation (80) is lot size for the product in the current period. Two scenarios are possible for relationship between lot size and overload in the period. In the first scenario, lot size is greater than overload in period. In this case, by having a greater divisor the cost of the lot shift appears more attractive in comparison with other alternatives. In the second scenario, lot size is less than overload capacity. In this case, lot size accurately reflects the overload eliminated. Equation (81) ensures that this lot shift occurs only if there is a cost reduction over the best combination so far. Equation (82) replaces the current values of cost, item, quantity and target period with values that provide a lower cost.
2. Move the entire lot to a day on which no production is scheduled for the product. Equation (83) makes sure that there is sufficient capacity in the target period to shift the entire lot and for the estimated setup time. In this shift, there is a setup eliminated and a setup added therefore the effect of the setup cost is nullified, while

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the change in inventory costs is reflected in equation (84). Once again, this equation encourages the movement of an entire lot. Equation (85) ensures that this is the least expensive alternative before reassignments in equation (86) are made.

3. This and the next option might end up breaking the lot to achieve feasibility. Breaking a lot is not an attractive alternative because there is no potential for setup elimination. In this option, an attempt is made to move part of the lot to another period in which the item is produced, this is assured by the if condition in (87). Equation (88) ensures that the quantity moved does not exceed either; the lot size of the item in the current period, what must be moved to achieve feasibility, or slack in the target period. Once again, no setup is added or eliminated and hence the incurred cost is just the change in inventory cost per unit of quantity shifted as stated in (89). Equation (90) ensures that this option is only selected if it is less expensive than other combinations and equation (91) reassigns the variables if this is true.
4. Finally, part of the lot can be moved to another period in which the product is not produced and where the slack is not sufficient to accommodate the entire lot and the setup time estimate. Equation (92) ensures that there is more than enough capacity in the target period to accommodate at least estimated setup time. Since larger lots tend to increase costs equation (93) makes sure that only a minimum of; (1) what is required to be moved to eliminate infeasibility, (2) amount of production the target period can accommodate, and (3) the size of the production lot for that product in that period is moved. The unit cost of the move given in equation (94) accounts for the added setup in the target period and the change in inventory cost. Equation (95)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

makes sure that this option is selected only if this is the lowest cost feasible option available. If this is true variables are updated as shown in equation (96).

A lot can be moved as far back as desired without violating any conditions, however, even if a single unit of production is moved to the next period the algorithm must ensure that the move does not violate the maximum delay constraint. If the maximum delay for backlogging is equal to the scheduling horizon (α = T) then this constraint check can be ignored. Every time production is shifted forward a procedure is executed to determine if it violates the maximum delay constraint; this procedure is described next.

Maximum Delay Criterion Checking Procedure

The validity of a forward move depends not only on the number of periods between the current period and the target period but also on the current inventory for the product in the periods in between. Given t*, the target period, (t*>t) a partial or entire lot of size m can be moved from t to t* only if $\pi = 0$ after the following loop is executed.

$\pi = 0$

for (t' = t to t' = t*-1)

{

$$\text{if}\,(I_{it'} - m < \; - \sum_{t''=t'-\alpha+1}^{t'} d_{it''}) \tag{97}$$

$$\pi = 1 \tag{98}$$

}

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Equation (97) checks if constraint (8) of the formulation presented in chapter I is violated by shifting m units from t to t*. If this violated for any period between the source and the target periods then equation (98) marks this move as an infeasible move. This checking routine is executed before any forward movement is considered and when the criteria is violated the lot shifting not considered.

The feasibility elimination routine does not always produce a feasible solution to an infeasible problem. However, when a feasible schedule can be attained by moving by shifting entire lots or parts thereof this routine ensures that this is attained in the most economical manner. Once a feasible schedule is determined then a simple solution improvement technique can be used to obtain a better solution this is discussed in the next section.

Solution Improvement Technique

For single product dynamic demand problem without capacity constraints the extreme point property of the Wagner-Whitin (Wagner and Whitin 1958) dynamic program algorithm is a necessary condition for optimality. Here, this property is extended for capacitated multi-item dynamic problem that allows for backlogging using the regeneration point theorem discussed in chapter III.

For the situation in which no backlogging is allowed, the extreme point property states that for any product i $I_{it-1}x_{it} = 0 \ \forall$ t for a solution to be optimal. Here two extensions are proposed to the extreme point property theorem. These theorems hold

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

good for a single item uncapacitated problem with backlogging. And with suitable modification can be applied to multi-item capacitated problems.

Theorem 1: There exists an optimal schedule such that $I_{it-1}x_{it} \leq 0$ for all t.

Proof: From the regeneration point theorem we know that given t < k and

$x_{it} > 0; x_{it+1} = x_{it+2} = . . . = x_{ik-1} = 0; \textit{and } x_{ik} > 0$ there exists a period t',

$t \leq t' \leq k-1$ such that t' is a regeneration point. Without loss of generality let us assume that t', $t \leq t' \leq k-1$, is a point of regeneration, i.e. $I_{it'} = 0$. Also the following capacity balancing equation is true $I_{ik-1} = I_{it'} + \sum_{t''=t'+1}^{k-1}(x_{it''} - d_{it''})$. Since we already know that, $\sum_{t''=t'+1}^{k-1} x_{t''} = 0$ and $I_{it'} = 0$, the capacity balancing equation reduces to

$I_{ik-1} = \sum_{t''=t'+1}^{k-1} -d_{it''}$. Therefore, $I_{ik-1} \leq 0$. We also know that $x_{ik} > 0$. Hence,

$\ni I_{k-1}x_k \leq 0$ which gives an optimal solution.

Theorem 2: There exists an optimal schedule such that $I_{it}x_{it} \geq 0$ for all t.

Proof: This theorem will be proved by contradiction. Assume, without loss of generalization, that $I_{it}x_{it} < 0$ is part of an optimal solution. The proof is complete if we can show that a better solution exists where this condition is not true. $I_{it}x_{it} < 0$ is only possible if $I_{it} < 0$ (since $x_{it} \geq 0 \; \forall$ t). The production lot can be increased from x_{it} to $x_{it} + |I_{it}|$ at no extra cost. This results in a reduced cost schedule equal to the cost of backlogging I_{it}. Obviously, this is a better solution than the schedule where $I_{it}x_{it} < 0$,

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

hence the cost of the original schedule was not optimal. Since no restriction was placed on the value of t the above discussion is true for all values of t.

The final solution is tested for the above properties and if there exists a situation such that $I_{it-1}x_{it} > 0$ or $I_{it}x_{it} < 0$ for any item and $q_t > 0$ then production lot is shifted from the closest period to the current period. If more than one item has the above criterion then the one that reduces the cost most is selected. In the next three sections, the three algorithms to solve the research problem will be detailed.

A Lot Shifting Algorithm (LS)

In this approach, the entire problem is considered as a whole and a greedy algorithm is used to reduce costs by eliminating setups. The algorithm works as follows.

Step 1. Assign schedule = demand on each day

$x_{it} = d_{it} \ \forall \ i,t$

Step 2. Use the sequencing routine to sequence production in each day. Calculate available capacities on each day. A positive value indicates slack and negative value indicates overloading.

Step 3. Move entire lots if the costs resulting from increased inventory or backlogging costs is less than the potential setup cost savings. Potential setup cost savings are estimated using one of the methods described previously. The estimated cost reduction is computed from the following equations:

$$E_{tk}^{i} = S_i - h_i \times (t-k) \times x_{it} \quad if\ t > k \tag{99}$$

$$E_{tk}^{i} = S_i - b_i \times (k-t) \times x_{it} \quad if\ t < k \tag{100}$$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Equation (99) determines the estimated cost saving when production is moved backward. Equation (100) estimates the cost savings when production is moved forward. Any cost saving move can only be considered if there is sufficient capacity in the target period. When production is moved forward an additional criteria described in the previous section must be satisfied. The lot shifting part of the algorithm works as follows:

while (.T.)

{

$$if\ (x_{ik} > 0\ and\ \ q_k \geq x_{it}) \tag{101}$$

save j, t', t'' $\ni E^{j}_{t't''} = max(E^{i}_{tk})\ \forall\ i \in [1,N];\ t,k \in [1,T];\ t \neq k;$ (102)

$$\text{if}\ (E^{j}_{t't''} > 0) \tag{103}$$

{

$$x_{jt''} \leftarrow x_{jt''} + x_{jt'} \tag{104}$$

$$x_{jt'} = 0 \tag{105}$$

calculate capacity

}

else

.F.

}

Equation (101) makes sure that the target period has a non zero production lot for the product in question and also capacity availability in the target period. Equation

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

(102) executes a sequence of loops to identify the product and the source and target periods that provide the maximum estimated savings. The "if" condition in equation (103) checks if the potential setup cost savings are greater than increased inventory costs. If maximum savings are less than 0 then no more cost saving moves are possible and the while loop is exited. Equations (104) and (105) reassign the size of the production lots in the source and target periods.

The lot shifting algorithm is similar to the one used by Dogramaci et al. (1981) except that, here both forward as well as backward movement of lots are considered. The salient features of this method are listed below.

1. Only the lot that promises the maximum savings, across all items and all periods, is moved each time.
2. The moves do not create infeasibilities, however infeasibilities may persist due to the nature of the original demand pattern.
3. All potential profitable moves are made before the lot shifting is stopped.

Step 4. Eliminate overloading using algorithm detailed earlier in this chapter.

Step 5. Improve the quality of solution using the methods described earlier.

The five step algorithm produces a low cost feasible schedule using a myopic lot shifting algorithm. By addressing the feasibility issue at every move the algorithm is restrictive in its movements of lots. An alternative approach would be to relax the capacity constraints initially to allow for the unrestricted movements of lots. This is the essence of the next two approaches to solving the problem.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

A Regeneration Algorithm (RA)

This algorithm is founded upon the regeneration point principle developed originally by Manne and Veinott (1967). Two issues have to be addressed before their solution procedure can be used as part of the algorithm proposed here:

1. formulation of the original research problem as a collection of single item problems, and
2. adaptation of the solutions of the single item problems to the constraints of the original problem.

The single item problem is formulated and a solution procedure is provided in chapter III. The single item formulation is targeted at accommodating a time variant setup cost however, setup in the research problem is sequence dependent and time invariant. Hence, the time variant setup costs are replaced by the setup cost estimates while solving the single item problem. All the rest of the information is available in the original formulation.

When the production lots resulting from the single item solutions are combined, the resulting schedule is likely to be infeasible due to capacity overloading. Hence, it is important that the solution be modified at the lowest possible cost to achieve feasibility. Now the five step algorithm used in this approach is described below.

Step 1. Determine the optimal production lots in each period for one product at a time.

Step 2. Sequence the production and calculate capacity requirements.

Step 3. If the solution is feasible then stop further execution, else go to step 4.

Step 4. Use the infeasibility elimination algorithm to determine a feasible schedule.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 5. Improve solution quality using methods described earlier in this chapter.

A potential problem with the above approach is that the lot size allocation may be unwieldy in some periods and negligible in others. This may require that the infeasibility elimination routine undo most of the lot assignments determined in step 1 of the algorithm resulting in the final solution being far from the optimal one generated in step 1. This problem can be overcome if some sense of capacity limitations is introduced in step 1. This is achieved through an intelligent modification of the above approach and is described in the next section.

Regeneration Algorithm with "Overload penalty" (ROP)

The biggest drawback of the approach described in the previous section is that it solves each item completely independent of others as if it were the only product in the system. In this approach, between successive solutions to the single item problem a "overload penalty" is added to the problem formulation. "Overload penalty" is a measure that conveys to the successive formulations of the single item problems information on the capacity usage of the previous items. The idea is to discourage vastly overloading any period in the horizon. Since the lot sizing decisions in the later problems will be affected by those made in the earlier problems, the sequence in which the items are solved affects the final solution. For a successful implementation of this approach, three issues must be addressed:

1. the sequence in which the items are lot sized
2. quantification of the "overload penalty"

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

3. including the “overload penalty” into the single item problems

Sequencing of Items

Here, we have to determine a ranking schema for the items so that the ones with a higher rank can be lot sized first when there are fewer capacity restrictions to consider. Faced with a similar ranking problem Kirca and Kokten (1994) report that a ranking system based on the total cost per unit demand is superior. For each item i, V_i is estimated using the economic order quantity (EOQ) concept. The average cost per unit for an item is computed by dividing the EOQ cost per period by the mean demand per period for that item. Then the item that has the largest average cost per unit is given a higher ranking. V_i is calculated using the equation (106).

$$V_i = \frac{\sqrt{2 \times S_i \times r_i}}{r_i} \qquad \forall\ i \tag{106}$$

Quantification of the "Overload penalty"

The purpose of "overload penalty" is to discourage the single item regeneration algorithm from scheduling production lots in periods during which other items are already scheduled if inclusion of these additional lots is likely violate capacity constraints. This can be achieved by adding the cost of eliminating infeasibility, caused by these additional lots, to the policy costs in those periods in the single item dynamic programming algorithm. This additional cost increases the cost of adding production lots in overloaded periods, and the dynamic programming algorithm is likely to select other alternatives to schedule the lots. However, the exact cost of eliminating infeasibility is difficult to compute. Therefore, it is desirable that a algebraic relationship

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

be established to a easily computed measure for calculating the cost of eliminating infeasibility.

Infeasibility manifests itself as overloading, and cost of eliminating infeasibility is the cost of eliminating overloading. Therefore, there is likely to be a relationship between infeasibility elimination costs and overloading. Using capacity requirements, calculation procedure described earlier overloading in each period can be easily computed. Hence, establishing a algebraic relationship between cost of eliminating infeasibility and overloading in a period is of interest here. Once this relationship is established a "overload penalty" can be computed that is equal in order (linear, quadratic, cubic, etc.) to the cost of eliminating infeasibility. Determining the order of the polynomial relationship between cost of eliminating infeasibility and overloading is discussed in the next few paragraphs.

When overloading occurs in a period, the infeasibility elimination procedure eliminates overloading by shifting production to other periods with positive slack. The procedure moves production to the period which results in least cost increase. When overloading is much greater than capacity available, several lots or parts thereof have to be shifted from the overloaded period. When multiple shifts are required to achieve feasibility, initial lot shifts are made at much lower costs than later lot shifts. This is because as lot shifts are made the best target periods are occupied by initial lots thereby reducing or eliminating slack capacity in those periods. This causes later lots to be moved to less desirable periods, much farther away from current period, to achieve feasibility.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The above argument supports the conjecture that the marginal cost (cost of eliminating one additional unit of overloading) increases as overloading increases. This can be explained with an example, consider two overloaded periods t' and t''. The overloading in these periods are denoted by $q_{t'}$ and $q_{t''}$ such that $q_{t'} > q_{t''}$. This means overloading in period t'' is greater than overloading in period t', since both $q_{t'}$ and $q_{t''}$ are negative. In addition, let us assume that an additional lot of 1 unit (hour) is to be added to production in these periods. It is conjectured here that the cost of eliminating infeasibility caused by additional unit in period t'' is greater than the cost of eliminating infeasibility in caused by additional unit in period t'.

Marginal cost is nothing but the sum of additional setup, inventory and backlogging costs incurred to eliminate 1 unit of infeasibility. Setup costs are a constant irrespective of the size of lot shifted and the target period. Increases in inventory costs and backlogging are proportional to the number of periods between the target period and the current overloaded period. Therefore, marginal cost is proportional to number of periods between the current period and target period. However, from earlier discussion, we know that the target period is likely to be further away as the overloading in a period increases. The above discussion reinforces the conjecture that marginal costs increases as overloading increases. This discussion can be captured by the following inference sequence.

marginal cost $\propto$ *change in inventory/backlogging cost*

change in inventory/backlogging cost $\propto$ *overloading*

therefore, marginal cost $\propto$ *overloading*

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

As already stated marginal cost is nothing but the cost of eliminating one additional unit of infeasibility. One unit of infeasibility is same as one unit of overloading. Therefore, the above inference chain can be extended by an additional link as follows:

$$\frac{\textit{Infeasibility cost}}{\textit{overloading}} \propto \textit{overloading}$$

from which the relationship between infeasibility cost and overloading is readily obtained

$$\textit{Infeasibility cost} \propto (\textit{overloading})^2$$

By simple extension the above relationship is also true for "overload penalty" computations. In conclusion, "overload penalty" is proportional to the square of number of hours of overloading. Assuming a constant of proportionality of 1 equation (107) states the equation that is used to compute the "overload penalty".

$$\textit{overload penalty} = (\textit{overloading})^2 \tag{107}$$

A mathematically complete equation for calculating the "overload penalty" is described below. Let

x_{it} = lot size for item i in period t $\forall$ i $\in \Delta$, and

x_{jt} = is a production lot for product j being considered for inclusion in period t

then

$$\zeta = \begin{cases} (\sum_{i \in \Delta} (x_{it} + s_i) + x_{jt} + s_j - C_t)^2 & \text{if } \sum_{i \in \Delta} (x_{it} + s_i) + x_{jt} + s_j - C_t \geq 0 \\ 0 \ \textit{otherwise} & \end{cases}$$

(108)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Equation (108) calculates the magnitude of the "overload penalty' using the actual sizes of the production lots and estimated setup times. Setup times are estimated because, sequencing issues are not addressed at this point, and therefore it is not possible to calculate the exact time requirements for setups. The "overload penalty" as computed above affects the quality of the optimal solutions to a lesser extent when magnitude of overloading is low. However, the impact is more drastic if overloading is high. Issues involved in incorporating the "overload penalty" into the dynamic programming algorithm used to solve the single item problem with backlogging costs are described next.

Inclusion of the "Overload penalty"

As stated in the previous section, "overload penalty" is included into the policy costs of the dynamic programming algorithm. Equation (57) in chapter III is used to calculate the cost of producing a lot in period t' to meet all demand between two successive regeneration points t and k. This equation is modified as follows to account for the "overload penalty"

$$c_{tk}^{t'} = S_i + \zeta + h_i \sum_{t''=t'+1}^{k} (t''-t')d_{it''} + b_i \sum_{t''=t+1}^{t'-1} (t'-t'')d_{it} \qquad \forall\ t, t', k \qquad (109)$$

The if condition in equation (108) ensures that the modified equation (109) is used only when capacity limit is exceeded. The rest of the algorithm remains the same. At this point, the complete regeneration algorithm with "overload penalty" is presented.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Regeneration Algorithm with "Overload penalty"

The six step algorithm proceeds as follows.

Step 1. Determine the order in which the single item problems are tackled.

Step 2. Determine the production lots in each period for one product at a time by including the overload penalty in each period for each item.

Step 3. Sequence the production and calculate capacity requirements.

Step 4. If the solution is feasible then stop further execution, else go to step 5.

Step 5. Use the infeasibility elimination algorithm to determine a feasible schedule.

Step 6. Improve solution using methods described earlier in this chapter.

While this approach does not guarantee a feasible solution after step 2, it reduces the scope of the infeasibility problem addressed in step 5 at the cost of modifications to the optimal production lot allocations in step 2. It is expected that this additional computational investment made in step 2 will be rewarded with an overall reduction in the cost of schedules generated by this algorithm.

Summary

In this chapter, the approaches used to solve the research problem are described. Before the algorithms were presented, the sequencing, capacity determination, infeasibility elimination, and solution improvement methods which are present in each approach were described in detail. These issues are closely interlocked; their considerations lend richness to the algorithms developed here. This richness

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

distinguishes these algorithms from other techniques that are currently used to solve the problem.

One of the goals of this research was to show that the computationally rich algorithms developed here are superior to current methods used to solve the research problem. One such technique is the EMQ calculations based technique that is used in MRP II , and described in chapter II. Another base case that is used for comparison purposes is the no lot size algorithm. The no lot size approach is a naive approach which essentially boils down to producing all demand requirements in a single lot.

A potential problem with EMQ calculations is that integer multiples of the lot size are not necessarily equal to the to total demand for the product. If this happens it is handled as follows.

1. Only what is required to meet demand during the finite scheduling horizon is produced, this might mean that the final lot is shorter than the EMQ calculations or might be completely eliminated. For example, if the lot size for product A has been calculated to be 8. The total demand for the product is 30. Hence, during the fourth cycle of production the lot size of product A is reduced to 6 so that no excess production occurs. And if some other products in the system require more than 4 cycles product A is not produced in those cycles because its cumulative demand has already been satisfied.
2. Under some circumstances EMQ lot sizes may be rather small. More cycles are required to meet demand in these instances. In these cases, relatively higher proportion of time is spent on setup. For this reason, at the end of the scheduling

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

horizon the quantity of production scheduled is less than cumulative demand for that product. In these situations production lot allocation is continued till all the demand is satisfied.

To provide further understanding of the workings of each algorithm, a sample problem is solved using each of the algorithms. A numerical example is generated and a step by step solution is provided for each algorithm in Appendix A. However, no inferences can be made regarding the performance of the algorithms from these solutions. Experimentation is necessary to evaluate the performance of algorithms. In the next chapter, computational experience with these approaches is presented, and experimentation is performed to determine the validity and robustness of the algorithms.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER V

EXPERIMENTAL PROCEDURE

This chapter focuses on the design and implementation of computational experiments to test the performance of algorithms developed in this research. An experiment consists of solving a series of problem instances that represent a wide variety of factors that may influence the performance of the algorithms. Experimentation is the process of setting goals for the experiments, identifying factors that influence the performance of algorithms, conducting experiments in a controlled environment, and analyzing the results using a statistical tool. The first step in experimentation is the setting of goals for experiments. Goals of an experimentation process are expressed by the hypotheses to be tested. Once the goals of the experiments have been determined, the next step is to identify factors that can potentially influence the results of the experiments. To determine whether a factor significantly affects the outcome of the experiments, more than one instance value, or factor level, of the factor must be experimented with.

Once the factors are identified, and before the algorithms are run over several problem instances, it is necessary to identify the measurement tool used to study the outcome of the experiments. In this research, the outcome is analyzed for the ability of the algorithms to find a feasible and cost efficient schedule. The solution quality of an algorithm is measured by the cost value of the schedule generated by the algorithm. After the models are run on the complete data set, usually some kind of statistical procedure is used to analyze the results of experiments. An introduction to the full

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

factorial analysis of variance statistical procedure used in this study is presented later in this chapter. First the goals of the experimentation process are reviewed.

Experimentation Goals

In this research, three heuristic algorithms have been developed to solve a class of real world manufacturing scheduling problems. Barr et al. (1995) have identified two requirements that must be satisfied when heuristic approaches are evaluated with computational experiments. The experiments must be able to:

1. test the relative effectiveness of algorithms in comparison with one another and with currently available solution techniques, and
2. describe the performance of the algorithms across several factor level combinations.

The goal of relative effectiveness experimentation is to test the quality of solutions obtained by the algorithms proposed in this research with one another and with current methods to solve the problem. The methods are tested against two base cases discussed in chapter IV. The hypothesis to be tested in this regard is presented below.

> Hypothesis 1: On the basis of the total schedule cost, at least one of the lot sizing algorithms proposed here would out perform both the no lot sizing method and the one based on EMQ calculations.

The logic behind this hypothesis is that the more computationally intensive algorithms proposed here consider the more complete problem picture in solving the problem. On the other hand the simpler methods do not consider all the information that directly affect

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the quality of the final solution. Which methods perform better than the other methods proposed here is difficult to predict.

Test of hypothesis 1 is the primary goal of this study. A broad comparison based on typical measure of central tendency, such as means and variance, could be made given a set of problem instances. However, this may not necessarily indicate whether one algorithm is better than another in a specific category of problem instance. Hence, a new hypothesis must be composed to test the latter issue.

The aim of descriptive experiments is to gain understanding about performance of algorithms and the factors that influence their performance. The factors could either influence the performance individually or in combination. The hypothesis to be tested in this regard is presented as follows.

> Hypothesis 2: The performance of the algorithms proposed here will be affected by the combination of factors that determine the inventory related production costs and demand environment.

The reasoning behind hypothesis 2 is that, each method is so information intensive that the quality of the final solution is affected by the multitude of factors involved. The factors that influence inventory related production costs are, inventory holding factor (h_i), setup cost per hour and backlogging cost factor (b_i). Demand environment is determined by the pattern of customer requirements and the utilization of resources in the production system.

A strategy is adopted to test for the two hypotheses stated above. Hypothesis 2 is first tested. If the results indicate that the factors have no influence on the quality of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

solutions then hypothesis 1 can be tested directly. However, even if a semblance of significance is detected then hypothesis 1 must be tested at each factor (or factor combination) that influences the quality of solution. The above strategy ensures that the quality of ANOVA results obtained by test of hypothesis 2 is not affected by extraneous factors that may affect the outcome of the algorithms. A complete discussion of these factors is presented in the next section.

Experimental Factors

In the past, as a result of research in the area of lot sizing in a dynamic demand environment, certain factors have been identified as having an effect on the performance of heuristics (Dilts and Ramsing 1989, Maes and Van Wassenhove 1986). These experimental factors can be divided into one of the following three categories:

1. nature of demand,
2. inventory related production costs, and
3. solution approach.

Nature of Demand

The factors in this category model the demand environment in which the production system is trying to meet customer requirements. Four factors determine the nature of the demand.

1. Size of the problem: This is determined by the number of products produced in the system and the duration of the schedule horizon. When there are very few products

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

and scheduling horizon is short, the performance of more complex algorithms deteriorates in comparison with optimal solutions (Dogramaci et al. 1981). This is because even the smallest changes in schedules causes a relatively large change in the cost of the schedule. For this reason, it is necessary to test the influence of problem size on the performance of algorithms. Here two sizes are considered: small and large. Small problem instances are those that have 6 items and a scheduling horizon of 10 days. Large problem instances have 15 products and a scheduling horizon of 30 days. Large problems are similar to the size of the problem faced in the motivating case.

2. Type of demand: EMQ lot size calculations are based on average daily demand. Hence, it is likely that the quality of its solutions are sensitive to the deviations of actual daily demands from average values. For this reason, two types of demand patterns are to be tested. Smooth demand patterns are those in which every product has almost similar demand in each period. Lumpy demand is the case when all the demand for the product over the scheduling horizon occurs in a few periods.
3. Capacity requirements: The ability of algorithms to find feasible solution is affected by the utilization of the constrained resource. This factor measures the capacity requirements of the demand for all items in each period. Loose capacity and tight capacity are to be tried. Loose capacity is when the demand in a period is equal to 50% of capacity available and tight capacity is when demand in each period is 75% of capacity available. This is acceptable as these utilization values do not take into account the time required for setup. Setup times, when switching from one product

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

to another, are sequence dependent and vary between 4% and 25% of the capacity available in each period. These values are similar to one encountered in the motivating case.

4. Backlogging limit: This factor controls the maximum delay allowed in meeting customer due dates and is the same as the α value used in the problem formulation. The tightness of the backlogging constraint potentially controls the ability of the algorithms to find a feasible solution. Here this value is to be varied over two levels, 10% and 50%, of the scheduling horizon. For small problems, this works out to values of 1 and 5 periods and for large problems it is 3 and 15 periods.

Inventory Related Production Cost Factors

The factors included in this category are setup cost factor, holding cost factor, and backlogging cost factor. It has been reported in the literature (Dogramaci et al. 1981, Dilts and Ramsing 1989, among others) that more than the actual values the ratios amongst these various cost factors are more important. Hence, in generating the data sets for this study the holding cost factor is assumed to be one and the setup and backlogging cost factors are varied as follows.

1. Backlogging cost factors: Hsieh et al. (1992) conducted an experimental study in which they found the ratio between backlogging cost factor and inventory cost factor to be significant. The two factor levels used in their study was 1.2 and 2. To show a greater importance for due date performance, the two levels of backlogging to inventory cost ratios studied here are 2 and 5.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. Setup cost factors: Setup times are generated randomly and are in the range of 1 to 6 hours. In this study, the setup costs are also sequence dependent for each product and are computed from setup time requirements for that sequence. Here, two factors are used as multipliers to convert setup time into setup cost. They are 2 and 8. This means that setup cost/holding cost factor could be as low as 2 or as high as 48 (this occurs when the setup time between two products is 6 hours and a setup cost factor of 8 is used).

Solution Approach

Given a set of demand and setup data this category of factors determine the method used to solve the problem. The solution approach is determined by the algorithm used to solve the problem and the setup estimating technique used. The factor levels for these factors are detailed below.

1. Algorithm: this factor is varied over 5 levels, which include the three methods proposed here and the two base cases. These are, lot shifting algorithm (LS), a regeneration algorithm (RA), regeneration algorithm with "overload penalty" (ROP), economic manufacturing quantity (EMQ), and no lot sizing method (NL).
2. Setup estimator: this factor is varied over two levels namely MIN and MAX. MIN uses the minimum of all possible setup for the product and MAX uses maximum possible setup for the product.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

To study the independent influence of experimental factors in the three categories listed above, it is essential to eliminate other external factors that can influence the results. To control the experimental conditions the following assumptions are made.

1. The holding cost factors are equal for all products. Holding cost factor for an item is proportional to space required to store one hour of production for that item and cost of producing one hour of that item. Typically, in a manufacturing environment larger products are more expensive to produce, and also their production rate is lower than smaller products. Hence, on balance space requirements and cost of production tend to be similar for all items. This justifies the assumption of equal holding cost factors for all items. However, in some production systems the above argument does not hold. In such cases, holding cost factor differences cannot be ignored and an additional factor must be included in the experimentation.
2. Backlogging cost factors are equal for all items. This cost is the penalty levied for late delivery of items. This penalty is usually used to indicate a preference for which products to be backlogged. Here, it is assumed that all orders are of equal importance, therefore backlogging cost factor is equal for all products.
3. Maximum delay (backlogging limit) is equal for all products. This factor controls the duration of backlogging allowed. Backlogging requirements are dependent on the demand pattern for a product and utilization of the system resource. Variations amongst products would affect factors already included in the study . For this reason different delay values for different items is not studied.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

4. Capacity availability is maintained constant. In a typical system, the day to day capacity availability remains constant, unless affected by breakdowns and other downtimes. It is assumed that the influence of these factors on capacity availability is minimal. In the motivating case three shifts are run on the coating machine per day. To reflect this situation capacity availability is held constant at 24 hours per day throughout the study.

Each of the factors and their levels included in this study has been determined after careful consideration of the prevailing values in the motivating case and from published literature. Table 1 summarizes the factors to be used in the experiments. For each factor name, it identifies a factor symbol used represent the factor in the experiments, the number of factor levels for each factor and the instance value of each factor level. Once the factors and factor levels have been identified the next step is to determine the performance measure and the methodology adopted to generate the data sets for the experiments.

Factor Name	Factor Symbol	Number of Levels	Factor Level Values
Algorithm	A	5	a1 = LS; a2 = RA; a3 = ROP; a4 = EMQ; a5 = NL
Setup estimator	B	2	b1 = MAX; b2 = MIN
Size	C	2	c1 = 6 products 10 periods (small) c2 = 15 products 30 periods (large)
Demand type	D	2	d1 = smooth; d2 = lumpy
Capacity utilization	E	2	e1 = 50% (low); e2 = 75% (high)
Setup cost factor	F	2	f1 = 2 (low); f2 = 8 (high)
Backlog cost factor	G	2	g1 = 2 (low); g2 = 5 (high)
Backlogging limit	H	2	h1 = 10% (low); h2 = 50% (high)

Table 1. Factors and factor levels used in experiments

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Performance Measure and Data Sets

While the most important performance yardstick is the cost of schedules generated, it is also important to compare the algorithms in terms of their ability to find feasible solution to a given problem set. Hence, the first performance measure merely counts the number of feasible solutions determined by the algorithm under consideration. Regarding the conventional methods, because of the fact that they neglect backlogging limit, the likelihood of an infeasible solution is quite high. Nevertheless in this research, they are given the benefit of doubt, and an assumption is made to consider all solutions as far as conventional methods are concerned. For this reason, feasibility analysis is only performed on the algorithms developed in this research. There are two aspects to the feasibility of a schedule.

1. Capacity feasibility: The ability of an algorithm to generate a schedule whose capacity requirements are less than or equal to capacity available on each day. Capacity infeasibilities are indicated by a negative slack value at the end of infeasibility elimination routine in any of the periods.
2. Backlogging limit feasibility: The ability of algorithms to find a schedule in which customer requirements are satisfied within the maximum delay duration from the due date. Each algorithm considers this constraint in the lot shifting procedure and also in the capacity infeasibility elimination procedure. Therefore, this requirement is embedded into every schedule that is generated by the algorithms. For this reason maximum delay feasibility need not be checked.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Therefore, a solution is termed feasible only if the final solution determined by the algorithm does not violate capacity requirements in any period of the scheduling horizon. The second performance measure is based on the total schedule costs which is the sum of inventory, setup, and backlogging costs. However, the absolute cost depends on the demand data and setup data input into the algorithm. To eliminate the effect of data, a comparative performance measure is used. Ideally, the performance measure would be based on the optimal value for a problem instance but, in the absence of such information, the lowest feasible cost generated amongst all combinations of algorithms and setup estimator is used. The performance measure is calculated as follows:

1. each data set is run with all solution approaches,
2. the lowest cost amongst them is determined, and
3. this lowest cost is then divided into the cost determined by each solution approach.

For each data set the performance measure is calculated using equation (110):

$$PM_{ab} = \frac{TC_{ab}}{\min\limits_{\forall\, a,b}(TC_{ab})} \qquad a = 1,2,3,4,5 \text{ and } b = 1,2 \tag{110}$$

where PM_{ab} is the performance measure for algorithm a using setup estimator b, and TC_{ab} is the total cost of the schedule generated by that combination. The advantage of using a relative performance measure is that it eliminates the inherent variances in the cost between data sets for a given treatment, i.e. the performance of a solution approach is more likely to be stable for a certain combination of demand and inventory cost factors.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The data sets, demand, and setup values, are generated randomly to fit the profile of the demand treatment being considered. The following procedures are adopted for generating smooth and lumpy demand patterns of desired capacity utilization. In any event, negative values are not allowed for any product on any day. For smooth demand pattern, initially set d_{it} = rand(1,4) for all i,t, where the function rand(m, n) generates a random number between m and n, both inclusive. If capacity requirement is not what is desired then, in the period in which requirements are not satisfied, do the following:

1. randomly select a product and adjust (increase or decrease) its demand by 1 unit,
2. if requirements satisfied then EXIT else repeat 1.

In the case of lumpy demand pattern, initially set d_{it} = rand(1,8) for all i,t. Now for each product set $d_{it} = 0$ for 0.6*T randomly selected days. Ensuring capacity utilization is handled using different approaches based on whether there is a shortfall or excess. If capacity requirement is less than what is desired then:

1. randomly select a product with non-zero demand and increase its demand by 1 unit,
2. if requirements satisfied then EXIT else repeat.

If capacity requirement is greater than what is desired then:

1. randomly select a product and reduce its demand by 1 unit,
2. if requirements satisfied then EXIT else repeat 1.

Using the procedure described above, it is possible to generate multiple sets of data that meet the demand pattern requirements. The number of data sets that should be generated for each demand pattern depends on the statistical procedure used to analyze the results and confidence desired in the results. These issues are discussed in the

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

"Analysis of Variance" section later in this chapter. At this point, it suffices to state that 10 data sets are generated for each demand pattern.

A complete listing of the demand data and the setup data used for experimentation in this study is listed in Appendix B. Essentially, eight sets of demand data are generated. The eight demand data sets are determined by the combination of problem size, demand type and utilization factor levels. Ten different problem instances are generated for each demand data set. A complete listing of these factor level combinations is provided in Table 2. There are only two sets of setup time matrices used; one for the small problem size and another for the large problem size. With demand data sets, setup time matrices and factor levels identified, computational experiments can be carried out.

Number	Size	Demand Type	Utilization
1	6x10	smooth	50%
2	6x10	smooth	75%
3	6x10	lumpy	50%
4	6x10	lumpy	75%
5	15x30	smooth	50%
6	15x30	smooth	75%
7	15x30	lumpy	50%
8	15x30	lumpy	75%

Table 2. Demand data sets

Computational Experience

Each of the eight demand data sets are solved at 4 possible combinations of inventory cost factors and 2 possible values of the backlogging limit factor. Table 3

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

provides a complete listing of these 8 factor combinations. In turn each of these 8 factor combinations are solved using 10 possible combinations of solution approach listed in Table 4. Hence, there are a total of 8x8x10 factor level combinations, called treatments, to be tested. With 10 repetitions in each treatment there are a total of 640x10=6400 problem instances. The algorithms are implemented using the C programming language on a IBM® Powerstation-220™ workstation, using the RISC System/6000™ architecture, running a UNIX™ operating system.

Number	Setup Cost Factor	Backlogging Cost Factor	Maximum Delay
1	2	2	10%
2	2	2	50%
3	2	5	10%
4	2	5	50%
5	8	2	10%
6	8	2	50%
7	8	5	10%
8	8	5	50%

Table 3. Complete list of inventory costs and maximum delay factors

Solution Approach	Algorithm	Setup Cost Estimator
1	LS	MAX
2	LS	MIN
3	RA	MAX
4	RA	MIN
5	ROP	MAX
6	ROP	MIN
7	EMQ	MAX
8	EMQ	MIN
9	NL	MAX
10	NL	MIN

Table 4. Solution approaches

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

All the 6400 problem instances were run, and the following observations were made regarding the computational requirements for the algorithms.

1. All problems were solved to conclusion within reasonable CPU time and memory space requirements.
2. The LS algorithm, when used to solve large smooth problems at low utilization level, requires the most number of computations and hence takes the most CPU time to find a feasible solution.
3. The ROP takes considerably less computational effort than the RA algorithm for a given problem. This is because the initial single item problem overloads capacity to a greater extent in the RA algorithm and it takes longer for the infeasibility elimination algorithm to eliminate infeasibilities in this case.

The solution obtained by each algorithm is shown in Appendix C. This appendix lists the actual cost of the schedule generated by each algorithm for each of the ten replications. The keyword marker "INF" indicates that no feasible solution was found to the problem. The computational requirements for the algorithms are reasonable, even for large problems, and for this reason no further analysis is performed with respect to the CPU time and memory requirements for the algorithms.

Output of the experimental runs are then prepared for statistical analysis. This is a two step process:

1. all infeasible results of the algorithms proposed here are removed from further analysis because their costs do not reflect the true costs of the schedule, and

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. the performance measure of each problem instance is computed using methods previously described.

This resulted in the loss of 1070 data points from the full factorial ANOVA design proposed in the next section. This loss of data results in unbalanced ANOVA design with empty cells. While a balanced design is preferable there are alternative statistical techniques available for unbalanced ANOVA. The general linear model (GLM) procedure of the SAS™ statistics module was used for analysis. The software was run on a IBM 3090 mainframe computer running the CMS operating system. The use of statistical output to make inferences about the performance of the algorithms and test the hypotheses is addressed in the next chapter. In the next section, a brief overview of the ANOVA procedure is presented.

Analysis of Variance

Analysis of variance (ANOVA) is a popular and robust statistical procedure for isolating the sources of variability in a set of measurements. When two or more factors are to be investigated simultaneously, as in this study, a multi-factor ANOVA must be used. In multi-factor ANOVA, a treatment corresponds to a combination of factor levels, for example in the study proposed here there are a total of 8 factors with 5 algorithms and each of the remaining factors over two levels so there are a total of $5 \times 2^7 = 640$ treatments.

Use of an ANOVA model is based on the scope of the interpretation of results. Three types of ANOVA models can be used in a multi-factor study.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

1. Model I ANOVA: also called fixed effects ANOVA, it is used to study factors that are of intrinsic interest, i.e. the results of the study are applicable only to the factor levels used in the study.
2. Model II ANOVA: also called random ANOVA, it is used in studies where all the factor levels are a representation of a wider population and interest is in the larger population.
3. Model III ANOVA: also called mixed model ANOVA, are used in studies where some factors are intrinsic and others represent a wider population.

The statistical methods used to evaluate a model are dependent on the type of model used. Hence, it is necessary that the correct model be identified for this study. In order to test the hypotheses stated earlier a fixed effects ANOVA model will suffice, besides the underlying statistics of the fixed effects ANOVA model are more robust than either the random ANOVA or mixed model ANOVA (Neter et al. 1990). Specifically, the results of mixed model ANOVA and random ANOVA are sensitive to departures from the following requirements for the ANOVA model:

1. the probability distribution associated with each treatment is normal distribution,
2. each probability distribution has the same variance (standard deviation), and
3. the observations for each treatment are random observations from the corresponding probability distribution and are independent of the observations at other treatments.

On the other hand, the fixed effect model is robust to deviations from normal distribution and unequal variances between treatments. However, its is sensitive to non-independence of observations between treatments. Fortunately, this problem can be

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

eliminated by randomization. Suitably, the data used in this study are generated by random number generators. For the above reasons, a fixed effects model is to be used in this study. Therefore the data does not have to be checked for deviations from the three assumptions listed above.

Like other statistical procedures ANOVA models are subject to Type I and Type II errors. In ANOVA, Type I error represents the risk of determining a factor effect to be significant when actually it is not. Type II error represents the risk of determining a factor effect to be insignificant when in reality it is. The protection against both Type I and Type II errors in ANOVA is controlled by the number of repetitions (sample size) at each treatment. The sample size has to be large enough to detect important differences with high probability. However, if the sample size is too large then unimportant differences between treatments become important with high probability (Neter et al. 1990). Therefore, determining sample size is an integral part of designing an analysis of variance study. Four factors influence the selection of a sample size (Bratcher et al. 1970):

1. number of factor levels,
2. the smallest difference the experimenter would like to detect (expressed by the number of standard deviations),
3. risk of type 1 error, and
4. risk of type 2 error.

The primary goal of this experiment is to determine if the differences between the performance measure means of the solution approaches are significant (tested by

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

hypothesis 1). Therefore, the number of factor levels is 10 (5 algorithms with 2 setup estimation techniques each). Type I error level (risk of asserting that a difference exist when the true difference is 0) should be maintained low. This greatly increases the confidence in the results when statistical results indicate that there is significant difference between the means. Accordingly, the risk of type 1 error is fixed at 0.05 (confidence interval = 0.95). Bratcher et al. (1970) have determined sample sizes for several combinations of the above factors. For type 1 error of 0.05 and number of factor levels = 10. Table 5 shows the sample sizes for some values of type II error and smallest difference to be detected.

	Smallest difference to be detected (number of std. dev.)		
Type 2 error	1	2	3
0.30	27	6	4
0.20	33	9	5
0.10	41	11	6
0.05	48	13	7

Table 5. Some sample sizes for type 1 error = 0.05 and number of levels = 10 (Bratcher et al. 1970)

When very close means are to be analyzed (number of standard deviations = 1), sample size is very sensitive to type 2 error level. However, at larger deviations the sensitivity of sample size to type 2 error level is dramatically less. A preliminary perusal of the output obtained (listed in Appendix C) indicates that there is substantial difference between the means of performance measure. Hence, the differences between means for significance can be set at a relatively high value. For example, to detect means that are

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

at least 2.5 standard deviations apart with 10 repetitions per treatment, risk of type 2 error can be limited to 0.05.

In general, larger sample sizes are required to detect smaller deviations and place a tighter control of the error levels. In this study, it is far more critical to control type 1 error than type 2 error. Increasing the risk of saying differences are insignificant when they actually are not (type 2 error), merely increases strength of the claim when algorithms proposed here significantly out perform currently used techniques. From the above discussion, it can be determined that the number of standard deviations and type 2 error level are not too critical to this study. Fixing the differences between means to be detected at 2 and risk of type II error at 0.15, in addition type 1 error at 0.05 and number of levels at 10, a sample size of 10 is obtained.

Therefore, experiments have been performed at 10 repetitions for each treatment, this number is typical for similar studies (Dilts and Ramsing 1989). When the number of repetitions is equal across all treatments then the experimental data is said to be balanced otherwise the it is unbalanced. The presence of infeasibilities converts a balanced analysis to an unbalanced analysis. When unbalanced ANOVA is used alternative statistical procedures are available to determine the significance of factors.

ANOVA results indicate the main factors and/or interaction factors that are significant. Further analysis is required to determine if one solution approach is significantly better than others. For example, as a result of ANOVA if we decide that the algorithm factor has a significant effect, then we would still be interested in determining which algorithm provides the better results amongst all the algorithms studied. For this

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

purpose the Tukey method of multiple comparison described by Neter et al. (1990) is used. This method holds the confidence in the tests at the same level as the type I error level selected for the model.

Summary

In this chapter the experimental procedure used to analyze the proposed algorithms were laid out. The goals of the experiments were set and the hypotheses to be tested were presented. Factors that are to be varied and their factor levels were discussed. In addition, the factors that were to be held constant were also detailed. The procedure adopted to generate random data sets was presented. Initial observations from computational experimentation were described. The appropriateness of ANOVA statistical procedure and a discussion of selection of a sample size was also presented.

Once the experiments have been performed and SAS™ software applied to the output of the experiments the results are ready for analysis. Analysis of results is necessary to determine which algorithm finds the most number of feasible solutions and to test for hypotheses. These are discussed in the next chapter.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER VI

ANALYSIS OF RESULTS

This chapter presents the results of the feasibility study and analysis of the experimental results with ANOVA. First, the capacity requirements of the final solution are examined. If the requirements are less than the capacity available then we a have feasible solution and the final cost can be accepted. However, even if capacity requirements are violated for one period then that schedule is infeasible and the final cost cannot be accepted as the true cost of that schedule. Following the feasibility study, ANOVA is performed on the feasible solutions to test for hypothesis 1 and hypothesis 2. The test of hypothesis 1 is dependent on the results of hypothesis 2, if the interactions are not significant then the factor level means of the algorithms can be compared. However, if some or all of the factor interactions are significant then treatment level means have to be compared. Therefore, the test of hypothesis 2 is performed first. Before any of the hypothesis are tested the feasibility results of the algorithms are analyzed.

Feasibility Results

The feasibility requirements are only tested for the three algorithms proposed here because, for the two bases cases feasibility is not required for the final cost to be accepted. The ability of an algorithm to provide a feasible solution is critical, because it affects the applicability of the solution approach in a real world production environment. However, in a complex problem, like the one addressed here, it is difficult to determine if

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

the demand and inventory cost related factors make the problem inherently infeasible. Therefore, in some problem instances, especially at high utilization levels, the ability of an algorithm to find feasible solution might be under estimated. The ability of the algorithms to provide a feasible solution is examined below.

The feasibility results for LS, RA and ROP are presented in Tables 6, 7, and 8 respectively. The factor symbols used in this table are same as the ones outlined in Table 1. Each cell in the tables indicates the number of feasible solutions that were found for the treatment represented by that cell, out of 10 problems that were tested for that treatment. First the feasibility results of the LS algorithm are analyzed.

	b1				b2			
	c1		c2		c1		c2	
	d1	d2	d1	d2	d1	d2	d1	d2
e1, f1, g1, h1	10	10	10	10	10	10	10	10
e1, f1, g1, h2	10	10	10	10	10	10	10	10
e1, f1, g2, h1	10	10	10	10	10	10	10	10
e1, f1, g2, h2	10	10	10	10	10	10	10	10
e1, f2, g1, h1	10	10	10	10	10	10	10	10
e1, f2, g1, h2	10	10	10	10	10	10	10	10
e1, f2, g2, h1	10	10	10	10	10	10	10	10
e1, f2, g2, h2	10	10	10	10	10	10	10	10
e2, f1, g1, h1	0	8	0	0	0	9	0	0
e2, f1, g1, h2	0	10	0	0	0	10	0	0
e2, f1, g2, h1	0	8	0	0	0	9	0	0
e2, f1, g2, h2	0	10	0	0	0	10	0	0
e2, f2, g1, h1	0	7	0	0	0	10	0	0
e2, f2, g1, h2	0	10	0	0	0	10	0	0
e2, f2, g2, h1	0	8	0	0	0	9	0	0
e2, f2, g2, h2	0	10	0	0	0	10	0	0

Note: factor symbols are defined in Table 1

Table 6. Feasibility results for the LS algorithm

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

From Table 6 it can bee seen that of the 128 treatments solved with the LS algorithm, on 48 occasions it did not find even a single feasible solution to the 10 problems at that treatment. On the whole the LS algorithm did not find feasible solution to 490 out of the 1280 problems presented to it. When high utilization problems alone are considered the algorithm found solutions for only 150 out of the 640 (about 23%) problems presented to it. Furthermore, the algorithm did not find a single feasible solution out of the 320 high utilization uniform demand problems presented to it.

	b1				b2			
	c1		c2		c1		c2	
	d1	d2	d1	d2	d1	d2	d1	d2
e1, f1, g1, h1	9	10	10	10	10	10	10	10
e1, f1, g1, h2	10	10	10	10	10	10	10	10
e1, f1, g2, h1	9	10	3	10	10	10	10	10
e1, f1, g2, h2	10	10	10	10	10	10	10	10
e1, f2, g1, h1	1	10	3	10	10	10	10	10
e1, f2, g1, h2	10	10	10	10	10	10	10	10
e1, f2, g2, h1	6	8	0	9	10	10	10	10
e1, f2, g2, h2	10	10	10	10	10	10	10	10
e2, f1, g1, h1	0	3	0	0	0	10	0	0
e2, f1, g1, h2	10	9	5	2	0	10	0	0
e2, f1, g2, h1	0	3	0	0	0	10	0	0
e2, f1, g2, h2	10	9	1	4	0	10	0	0
e2, f2, g1, h1	0	1	0	0	1	5	0	0
e2, f2, g1, h2	10	9	10	10	10	10	0	0
e2, f2, g2, h1	0	0	0	0	3	4	0	0
e2, f2, g2, h2	10	10	10	9	7	10	0	0

Note: Factor symbols are defined in Table 1

Table 7 Feasibility results for the RA algorithm

The reason for these disappointing results can be found in the way the algorithm operates. The LS algorithm is initiated by assuming a schedule equal to the demand

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

matrix. When this initial schedule is sequenced and capacity requirements calculated capacity requirements are greater than capacity availability in every period. This does not leave the algorithm any room to maneuver, as the capacity availability is checked before the lots are shifted. This situation is further aggravated in uniform demand environment where there are more products in each period of the demand matrix, causing increased setup time requirements.

	b1				b2			
	c1		c2		c1		c2	
	d1	d2	d1	d2	d1	d2	d1	d2
e1, f1, g1, h1	10	10	10	10	10	10	10	10
e1, f1, g1, h2	10	10	10	10	10	10	10	10
e1, f1, g2, h1	10	10	10	10	10	10	10	10
e1, f1, g2, h2	10	10	10	10	10	10	10	10
e1, f2, g1, h1	10	10	10	10	10	10	10	10
e1, f2, g1, h2	10	10	10	10	10	10	10	10
e1, f2, g2, h1	10	10	10	10	10	10	10	10
e1, f2, g2, h2	10	10	10	10	10	10	10	10
e2, f1, g1, h1	9	9	0	1	10	10	10	9
e2, f1, g1, h2	10	10	10	10	10	10	10	10
e2, f1, g2, h1	10	8	0	0	10	10	10	10
e2, f1, g2, h2	10	10	10	10	10	10	10	10
e2, f2, g1, h1	1	3	0	0	9	10	10	6
e2, f2, g1, h2	10	10	10	10	10	10	10	10
e2, f2, g2, h1	1	4	0	0	10	9	10	7
e2, f2, g2, h2	10	10	10	10	10	10	10	10

Note: Factor symbols are defined in Table 1

Table 8. Feasibility results for the ROP algorithm

The RA algorithm ended up with infeasible solution for 457 out of the 1280 problems it was used on. Similar numbers for the ROP algorithm is 124 out of 1280. The RA algorithm by ignoring the capacity requirements initially is able to find more

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

feasible solutions to the uniform demand high utilization problems, 87 out of 320 compared with 0 out of 320 for the LS algorithm. However, for this very same reason it fails to find a feasible solution for 42 out of the 640 low utilization problems, whereas the LS algorithm was able to find feasible solutions for all of these problems. This is caused by the unrestrained overloading of certain periods by the unconstrained single item optimization routine solved initially. The overloading is so great that the infeasibility elimination routine cannot find feasible periods to move the excess capacity. From the ROP feasibility results, it can be seen that placing restraints on the initial single item routine does increase the chance of finding a feasible solution. However, for some treatment combinations none of the methods proposed here are able to find a feasible solution. When all the problem instances are considered, there are only 10 out of 640 (1.6%) for which the ROP problem with the MIN setup cost estimator did not find a feasible solution. This is a very low percentage considering the fact that it is not known if these problems have a feasible solution at all.

Analysis of feasibility results indicates that ROP algorithm is superior to RA and LS algorithms for finding feasible solutions. However, it might very well be that the other two methods out perform the ROP algorithm in instances when they do find feasible solutions. Besides, it is yet to be determined if these algorithms are in fact better than the base case methods namely, EMQ and NL. Before such comparisons can be attempted, it is necessary to determine if the performance of the algorithms is affected by the interaction of the algorithms with nature of demand and inventory cost factors. This

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

requires a test of hypothesis 2 stated in chapter V. The testing methodology and the results are described in the next section.

Test of Hypothesis 2

As per the strategy presented in chapter V, hypothesis 2 is first tested. The results of the unbalanced ANOVA are first examined for the importance of the interactions between main effects. Testing methodology for hypothesis 1 is dependent on the results of test for hypothesis 2. If interactions are unimportant then the same ANOVA model can be used to test for hypothesis 1. On the other hand, if the interactions are important then, the ANOVA model has to be modified before hypothesis 1 is tested. Hence, at this stage of the statistical analysis we are merely interested in the presence of interaction effects between the factors. The test for hypothesis 2 can be formally stated as

H_o : Interaction effects are significant

H_a : Interaction effects are not significant

The relevant results of the SAS output for ANOVA model are presented in Table 9. The F value for the interaction effects is given by the equation

$F^* = \dfrac{Mean\ Square\ Interactions}{Mean\ Square\ Error}$. For $\alpha = 0.05$ the decision rule becomes

If $F^* > F(0.95, 539, 4779)$ conclude H_o

If $F^* \leq F(0.95, 539, 4779)$ conclude H_a

$F^* = 62.79$ and $F \cong 1$, accept H_o. Therefore, with a confidence of 95% ($\alpha = 0.05$) the results indicate that the interactions are important.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Source	DF	Sum of Squares	Mean Square	F Value
A*B*C*D*E*F*G*H	539	75266.64	139.64	62.79
Model	550	177445.45	322.63	145.07
Error	4779	10628.40	2.22	
Corrected Total	5329	188073.86		

Note: Factor levels are defined in table 1

Table 9. ANOVA results for test of hypothesis 2

However, the reliability of the ANOVA results in this particular instance is questionable. This is caused by the unbalanced nature of the input data. Especially, presence of empty cells (treatments at which at which no data are available) leaves the software to make assumptions of the data that can be undesirable (Neter et al 1990). However, the ANOVA results reinforce the influence of factors observed in the feasibility analysis. Therefore, it seems reasonable to assume that interactions are significant. This is a safer conclusion to arrive at than to decide that interactions are not important, as supported by the following analysis.

The first option is to test further (test for hypothesis 1) on the assumption that interactions are important. Then the means of the solution approaches would have to be compared at each one of the 64 treatments. During this analysis, some treatments that do not give a feasible solutions can be dropped from consideration resulting in a more balanced design. It is possible that the interactions were not actually important and this option was selected. In this worst case scenario the maximum damage done is the time invested in performing the more detailed analysis.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The second option is to continue on the assumption that interactions are not important. This makes test of hypothesis 1 much easier as only one more ANOVA run need to be carried out. But, the penalty for having made the wrong choice in this case is substantial. For example, based on all means it can be determined that ROP is superior to other algorithms. This does not mean that ROP out performs all other algorithms under every treatment. It is possible that the LS algorithm out performs the ROP in a low utilization uniform demand environment or alternatively it is possible that there exists no significant difference between the performance of the algorithms at certain treatments. The possibility of arriving at a result similar to the one just described would be completely missed if option 2 is selected.

As indicated by the above discussion, the more conservative option 1 is a better choice. Accordingly, hypothesis 1 is performed over all 64 treatments. This process is described in the next section.

Test of Hypothesis 1

Since significant interactions are present between the factors a single comparison across all factor level combinations would not represent the actual reality about the performance of the algorithms. This can only be found out if the treatment means are compared using the Tukey procedure.

However, this drastically increases the number of Tukey tests to be performed. There are 4 demand factor levels and 2 inventory cost factor levels yielding 64 (2^4x2^2) treatments. Hence, 64 single factor ANOVA models are run and each time the 10

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

solution approaches, listed in Table 4, are compared to determine the best solution for a given treatment. It is possible that for a given treatment one or more of the algorithms developed here provide infeasible solutions. In these cases, the comparison of means must be based on unequal sample sizes. Fortunately, the Tukey procedure used for comparison of means can accommodate unequal sample sizes. In fact when the sample sizes are unequal the results obtained from Tukey tests are more conservative (Neter et al. 1990). This allows for the comparison of those solution approaches that find feasible solutions for at least 1 of the 10 problems presented to it.

The complete ranking of solution approaches by performance at each ANOVA run is presented in Appendix D. For each of the 64 treatments, this appendix lists the mean performance measure for each solution approach, the number of feasible solutions found and which solution approaches are significantly different from others. The solution approach(es) that provides significantly lower performance measure than others at each ANOVA run is presented in Table 10. The solution approach number presented here are the same those used in Table 4. This table lists, for each treatment, the solution approach(es) that provide significantly lower performance measures than others. When more that one solution approach is listed in a treatment it is listed in the ascending order performance measures, i.e., the solution approach listed first is better than other listed solution approaches but not significantly better. At these treatments, we cannot say that one approach is better than others listed for that treatment. Formally hypothesis 1 can be stated as:

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

H_o: A solution approach developed here produces significantly lower schedule costs than base cases

H_a: Otherwise

Hypothesis II is tested after each ANOVA run, if the solution approach that provides the significantly lower performance measure is a combination of an algorithm developed here and a setup estimator then accept H_o. Any other result substantiates the acceptance of H_a.

From Table 10 it is seen that solution approach 6 (ROP with MIN setup estimator) provides the lowest cost feasible schedule at each treatment. However, at some treatments its cost do not substantially deviate from some other methods proposed here. Also, at all treatments but four, solution approach 6 provides significantly lower performance measure ratio than any of the base cases. Thus, the hypothesis H_o is valid for 60 out of the 64 treatments. The four treatments that do not support H_o have the following factor combinations:

treatment 1: c2, d1, e2, f2, g1, h1;

treatment 2: c2, d1, e2, f2, g1, h2;

treatment 3: c2, d2, e2, f2, g1, h1;

treatment 4: c2, d2, e2, f2, g2, h1.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

	c1				c2			
	d1		d2		d1		d2	
	e1	e2	e1	e2	e1	e2	e1	e2
f1, g1, h1	6	6	6,5,2,4,1,3	6,2,1,4,5	6	6	6,4,2,1	6
f1, g1, h2	6	6	6,5,2,4,1,3	6,2,1,4,3,5	6	6	6,4,2,1	6
f1, g2, h1	6,2,4	6,5	6,2,4,5,1,3	6,2,1,4,5	6,4	6	6,4,2,1,3,5	6
f1, g2, h2	6,2,4	6	6,2,4,5,1,3	6,2,4,1,3,5	6,4,2	6	6,4,2,1,3,5	6,5
f2, g1, h1	6	6	6,5,1,4,2,3	6,4,5,2,1	6,4	6,8	6,4,1,2	6,8,7
f2, g1, h2	6	6	6,5,4,1,2,3	6,2,5,4,1,3	6,4	6,8	6,4,5	6
f2, g2, h1	6	6	6,5,4,1,2,3	6,5,4,2,1	6	6	6,4,2	6,7,8
f2, g2, h2	6	6	6,4,1,2	6,2,1,4,5,3	6	6	6,4	6

Note: Solution approaches are defined in Table 5.4, factor symbols are defined in Table 5.1

Table 10. Best solution approach for each demand and inventory cost combination

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The four treatments that substantiate H_a are high utilization large problems with high setup cost factors. At these treatments, solution approach 6 performs better than any of the current approaches though not significantly better. The following analysis is based on the cost values listed in Appendix C and performance measure comparisons in Appendix D for each of the above four treatments. The reason for narrowing performance gap can be explained as follows:

1. Treatment 1 and treatment 2: These are long schedule horizon problems with uniform demand type. The average daily demand values used to calculate EMQ lot sizes is a close approximation of the actual demand values. When EMQ is used in these circumstances, the scheduling cycles tends to balance out with the demand pattern (Elmagrabhy 1978). This is especially true when EMQ is used with MIN setup estimator (approach 8) as this tends to produce smaller lots resulting in shorter cycles. Further, at these particular treatments the penalty for backlogging is low, and this reduces the cost for deviating from customer requirements. The corrections for capacity constraints using the "overload penalty" has a significant effect in these demand patterns. This is especially true for the items that have lower average cost per unit as defined in chapter IV (these tend to be scheduled later when most the earlier items have cornered substantial amounts of the available capacity). For this reason, optimal solutions for the single item problems are drastically affected by the "overload penalty". Furthermore, due to the tight capacity considerations the infeasibility elimination part of the heuristic can find feasible solutions only with substantial increase in costs incurred. This explains the narrowed gap in the quality

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

of solutions provided by approaches 6 and 8. However, it must be emphasized that the schedules generated by approach 6 are feasible whereas those generated by approach 8 are not necessarily so.

2. Treatment 3 and treatment 4: The reason for the statistical indifference at these treatments lies in calculation of performance measures. Once again, performance measure is the ratio of the minimum cost schedule found for a problem divided into the cost of schedule determined by the approach in question. However, these performance measures are skewed when none of the approaches proposed here is able to find a feasible solution. At treatment 3 there are 4 such problem instances and at treatment 4 there are 3 such instances. For these problem instances, the performance measure for either approach 7 or 8 equals 1 because they provide the lowest cost schedule amongst the base cases (remember, only feasible solutions amongst the algorithms proposed here are considered for statistical comparison). Therefore, at these treatments the repetitions used for Tukey procedure is unbalanced, with ten repetitions for the base cases and 6 repetitions at treatment 3 and 7 repetitions at treatment 4 for approach 6. For solution approaches 7 and 8, this leads to 4 performance measures of "1" at treatment 3 and 3 such values in treatment 4. These values reduce the statistical difference between the means of the approaches. However, when approach 6 is able to find feasible solutions, these solutions out perform those found by approaches 7 and 8, by at least, a factor of 2. For this reason, when only the problem instances for which approach 6 found a

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

feasible solution are compared the difference between the procedures becomes significant.

From the analysis thus far, it can be seen that solution approach 6 is superior to the other approaches tested here. In an experiment of the size performed here, several inferences can be drawn about the general performance of the approaches that do not directly relate to either of the hypotheses. These inferences are not statistically tested but are based on rough cut analysis of the experimental and statistical results. These general trends in the performance of the solution approaches are discussed below.

Other Results

1. The above comparisons are made without testing for the feasibility of EMQ and NL algorithms. Since feasibility requirements merely increase the cost of schedules, this arrangement gives a fair shake to the traditional algorithms.
2. The LS, RA, and ROP algorithms perform better with MIN setup estimator than with the MAX setup estimator. This indicates that a conservative approach in estimating the potential setup cost savings is preferable. Also, the MIN estimator under estimates the setup time requirements and this encourages greater mobility when the lots are shifted around to achieve feasibility.
3. In small problem sizes with lumpy demand environment LS and RA based approaches provide good solutions that are mostly statistically indifferent from ROP based approaches. This is true in both low utilization and high utilization rates. However, in large problem environments with lumpy demand there is a significant

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

difference in performance under low utilization and high utilization rates. In such problems, when the LS and RA based approaches are able to find feasible solutions they tend to be as good as ROP approach. But, in large high utilization environments they are less likely to find feasible solutions.

4. Between two problems with the same capacity utilization the one with the greater number of products is more difficult to solve. This is due to the non-negligible setup times required for switching between products. This difference is greater in uniform demand environments, where greater number of setups are generally required.

Summary

In this chapter, the results of the experiments have been analyzed. The ability of the algorithms proposed to provide feasible solutions was analyzed. Results indicate that a "overload penalty" between successive solutions to the single item lot sizing problem is a far superior approach compared to other algorithms tested here. Especially, when the ROP algorithm is used with MIN setup estimator (solution approach 6), it fails to find a feasible solution to only 1.6% of the 640 problems solved using the approach.

An ANOVA test for the significance of interactions between the factors indicated that interactions are significant. The Tukey procedure was used to compare the performance of the approaches at each treatment defined by these interactions. Once again, approach 6 performed better than other approaches tested at all treatments. In 28 of the 64 treatments approach 6 was significantly better than other approaches and in 60

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

out of the 64 treatments the approaches proposed in this research performed significantly better than the base cases.

So far the methodology, experimentation and analysis of the results of the experimentation have been described. In the next chapter, conclusions that can be drawn from this research study are presented, the significant contributions are highlighted and directions for future research are discussed.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHAPTER VII

CONCLUSIONS, CONTRIBUTIONS AND FUTURE RESEARCH

This research was a significant undertaking in the area of scheduling in a dynamic demand environment. This study has explored the use of information that already exists in a manufacturing enterprise to develop better, less expensive, schedules. The implications of this study, in terms of major conclusions, contributions, and directions for future research are discussed in this chapter. Definite conclusions can be drawn from the vast array of problems which were addressed here. This research was built upon results obtained by previous researchers and has further extended the knowledge in the area of dynamic lot sizing and scheduling. Because of the complexity of the problem, research in the area of lot sizing and scheduling has been on going for several decades. Potential cost savings ensure that it is likely to be on going for several decades into the future. In the next section, conclusions that can be drawn from the analysis of results in chapter VI are discussed.

Conclusions

The problem of determining schedules, including lot sizes, sequence and timing, in a dynamic demand environment in the presence of sequence dependent setups, finite capacity, setup carryover and variable backlogging has been successfully addressed in this research. Three different heuristics, each used with two different setup cost estimators, for solving the problem have been proposed, studied and evaluated. These heuristics are compared with two base cases, NL and EMQ. In all, ten solution

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

approaches have been extensively tested and their performance evaluated over a wide variety of problem instances duplicating real world problem conditions. Results show that the regeneration algorithm with "overload penalty" (ROP) using minimum setup estimating procedure (MIN) is most adept in finding feasible solutions.

Statistical analysis of the schedule costs generated by the algorithms indicates that the demand environment and the inventory related costs in the production system significantly affect the performance of the solution approaches. However, further analysis shows that for a majority of factor combinations the ROP heuristic used in combination with MIN setup estimator significantly out performs all other solution approaches tested. In treatments at which the solution approach is not significantly better than other approaches it performs at least as well as any other solution approach.

Each of the 10 solution approaches tried here is tested over the same set of 640 problem instances. Table 11 shows:

1. the number of times each solution approach found the minimum cost feasible schedule over all the approaches tested,
2. the worst case performance of the algorithm, expressed using the same performance measure used in the statistical analysis, and
3. the average performance measure for the algorithm.

The table clearly indicates that solution approach 6 (combination of ROP and MIN) is superior to all others. On the average the cost of schedules generated by EMQ is at least 5.8 times and those generated by no lot size method is 9.5 times the cost of schedules generated using solution approach 6. Finally, this research shows that with intelligent

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

application of operations research techniques good solutions can be obtained to even the most challenging large scheduling problems.

Solution Approach	Number of Minimum Cost Solutions	Worst Case Performance	Average Performance
1	3	4.85	2.07
2	3	4.21	1.90
3	0	8.57	2.66
4	1	4.21	1.85
5	2	11.4.3	2.54
6	623	1.21	1.00
7	4	19.44	5.82
8	9	47.89	7.80
9	0	40.92	9.55
10	0	40.92	9.55

Note: Solution approaches are defined in Table 4

Table 11. Relative performance of solution approaches

Contributions

In this study, in addition to development, implementation and testing of heuristic solutions to a complex dynamic lot sizing and scheduling problem, several other contributions have been made to the literature in the area of production scheduling. These contributions are listed below.

1. Formulation of the Problem: In this study, a new formulation is presented for a real world scheduling problem. This extends the formulation introduced by Gopalakrisknan et al (1995). This formulation can potentially be used by other operations research practitioners to determine alternate approaches to solve the problem.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

2. Solution Improvement Techniques: Two extensions to the Wagner-Whitin (Wagner and Whitin 1958) extreme point property have been proposed and proved. These extensions allow the application of the property when backlogging is allowed. In this study, they have been incorporated into each of the algorithms proposed here and are applied to improve the quality of the solutions determined by the algorithms.
3. Optimal Sequencing Technique: In this research, a new procedure, a combination of dynamic programming and a heuristic solution to the TSP problem, has been found to determine the optimal sequence for production once the production lots within each period is determined. This technique provides significant cost reductions when used in a sequence dependent setup environment. At a practical level, this method provides an optimal solution to most real world sequencing problems. The sequencing technique can be used in other sequence dependent setup environments. For example, it can be used as part of algorithms that assume sequence dependent setup costs, zero setup times and no backlogging, a problem studied by Fleischmann (1994).
4. "Overload penalty": In this study "overload penalty" plays a crucial role in generating not only feasible schedules but also superior quality schedules. "Overload penalty" is based on the marginal cost of eliminating infeasibility. This simple yet powerful concept can be transformed into other multi-item environments. This allows the solution approaches to take advantage of optimal solutions to simpler problems. This elegant approach can be used in place of the mathematically daunting Lagrangean relaxation approach.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

5. Problem Size: This study is different from previous studies in that it makes very few assumptions about the nature of demand and inventory related costs in the real world production system. By solving several variations of large problems (15 products 30 periods), this research has shown that it is not necessary to make assumptions that debilitate the quality of results when attempting to develop solutions large real world scheduling problems.

Directions for Future Research

The problem of dynamic lot sizing and scheduling is NP hard hence, it is unlikely that optimal solutions for complex problems in this area can be found within reasonable computational efforts. For this reason, heuristic approaches have to be developed to solve scheduling problems in real world production systems. In this research, a set of approaches have been proposed to generate schedules in the presence of several complicating factors. However, there exists a possibility for improving the quality of solutions. The potential savings in production costs resulting from efficient schedules justifies further research in this area. Research work here can be extended in the following directions:

1. Setup estimator: It has been reported that MIN and MAX provide the best estimates of setup time and cost (Dilts and Ramsing 1989). In this research, both setup time and setup costs are estimated using only one of these estimators at a time. However, it may be possible to obtain better schedules by using the MAX estimator to estimate setup costs and MIN estimator to estimate setup time. The reasoning being that

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

MAX cost estimator encourages elimination of greater number of lots to reduce setups and the MIN time estimator will increase the mobility of the lots when infeasibility eliminating routine tries to shift production to eliminate infeasibility.

2. "Overload penalty": In this study, only one type of "overload penalty" is used. The penalty for exceeding capacity is proportional to the square of the difference between required capacity and available capacity. The function used to determine the "overload penalty" affects the schedules generated. When the "overload penalty" is linear, it may not have a significant effect on the overloading of periods resulting in greater number of infeasible solutions. On the other hand, a higher degree polynomial may negatively affect the quality of the solution in low capacity utilization environments. More work is needed to determine the exact relationship between the "overload penalty" and capacity overloading.

Summary

A comprehensive study has been performed on a class of real world multi-item dynamic lot sizing and scheduling problem. The problem formulation accounts for all possible significant factors that affect the quality of schedules. These are: known dynamic demand, multiple items, capacity constraint, sequence dependent setup cost and setup time, finite variable backlogging, and setup carryover. A review of current research showed that solutions, to the scheduling problem of the complexity attempted here, has not been attempted yet. Yet the benefit accruing from the solution methodology is very real. It was statistically proven that the regeneration algorithm with

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

"overload penalty" is the most effective solution approach to this problem. On an average the cost of schedules generated by EMQ computations was at least 5.8 times the cost of feasible schedules generated by the regeneration algorithm with "overload penalty".

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

NOMENCLATURE

A_i	setup cost per production lot of i, independent of sequence (\$/setup)
b_i	penalty cost for backlogging per unit per period for i (\$/hour/day)
C_t	capacity available in a period t (hour)
C_{TSP}	cost of the solution provided by the TSP algorithm (\$)
$c_{tk}^{t'}$	cost of producing a lot in period t' to meet all demand between the regeneration points of t and k (\$)
d_{it}	demand for product i in day t (hour)
E_{tk}^{i}	estimated savings from moving product i from t to k (\$)
f_t	minimum cost incurred between regeneration points t and T (\$)
fp_t^k	product that will be produced first by state k on day t
h_i	inventory holding cost per unit per period for i (\$/hour/day)
H_i	inventory/backlogging cost for i (\$)
i	item index (i = 1, 2, ... N)
I_{it}	inventory of item i at the end of period t (hour)
j	item index (j = 1, 2, ... N)
l	number of products currently included in the TSP sequence
lp_t^k	product that will be produced last by state k on day t
N	number of products in the system
n_t	number of products scheduled for production on day t
o_{kt}	product sequenced kth on day t

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

p_i production rate for i per period (unit/period) where $p_i > r_i$

PC_t^k lowest possible setup cost from state k in period t to T ($)

q_t slack capacity available in period t, a negative value indicates overloading (hour)

Q_i lot size for i resulting from EOQ and EMQ computations (unit/cycle)

r_i demand rate for i per period (unit/period) ; $r_i = \frac{\sum_{t=1}^{T} d_{it}}{T}$

s_{ij} setup time to switch from product i to product j (hour)

s_i estimated setup time (hour)

S_{ij} cost to switch from product i to product j ($)

S_i estimated setup cost ($)

SC_i average cost per unit time per setup ($/period/setup)

t period index (t = 1, 2, ..., T)

tg target period to move production to achieve feasibility

TC the total cost of a feasible schedule ($)

TSP_t^k cost returned by the TSP algorithm for state k on day t ($)

$U_i(t)$ marginal cost coefficient for i if demand in period t is produced in period 1 ($/time^2/unit)

V_i the average cost per unit for i estimated using EOQ formula ($/unit)

w_{it} 1 if i is produced in period t
0 otherwise

x_{it} production lot scheduled for i in period t (hour)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

y_{ijt} 1 if product j is produced immediately after product i in period t
0 otherwise

α maximum number of periods of backlogging allowed (day)

β_{it} 1 if product i is produced last in period t
0 otherwise

Δ set of items for which lot sizes have already been determined

ΔH_{μ} change in inventory/backlogging cost associated with product μ ($)

ϕ_i the product occupying the ith position in the TSP sequence

γ_i^k cost of including product i in the current TSP sequence at position k ($)

$\gamma_i^k = (S_{\phi_{k-1}i} + S_{i\phi_k})$ where k = 2, . . ., l+1

η quantity to be moved to remove infeasibility (hour)

λ_{it} 1 if product i is produced first in period t
0 otherwise

μ product selected for moving to remove infeasibility

Π_t set of items that have production lots > 0 in period t, i.e. $i \in \Pi_t$ *iff* $x_{it} > 0$

θ_k cost of kth sequence when solving TSP by complete enumeration;

k = 1, 2, ..., ${}^{n_t}_{2}P$ ($)

ρ_i utilization ratio for i = $\dfrac{r_i}{p_i}$

σ_t^k state in t+1 that gives the lowest cost solution for state k in period t to T ($)

τ_i cycle time for item i (period)

ω_t idle capacity in period t (hour)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

ξ_t^k sequence returned by the TSP algorithm for state k on day t

ψ set of nodes not yet included in TSP sequence

ζ overload penalty (hours^2)

∇ cost of moving 1 unit of the selected product to target period (\$/hour)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

REFERENCES

ARAS, O. A., and SWANSON, L. A., 1982, A lot sizing and sequencing algorithm for dynamic demands upon a single facility. *Journal of Operations Management*, **2** (3), 177-185.

BARR, R. S., BRUCE G. L., KELLY J. P., RESENDE M. G. C., and STEWART, W. R., 1995, Designing and reporting on computational experiments with heuristic methods. *Journal of Heuristics*, **1** (1), 1-27.

BOMBERGER, E., 1966, A dynamic programming approach to a lot size scheduling problem. *Management Science*, **12** (11), 778-784.

BRATCHER, T. L., MORAN, M. A., ZIMMER, W. J., 1970, Tables of sample sizes in the analysis of variance. *Journal of Quality Technology*, **2** (3), 156-164.

CATTRYSSE, D., MAES, J., and VAN WASSENHOVE, L. N., 1990, Set partitioning and column generation heuristics for capacitated dynamic lot sizing. *European Journal of Operational Research*, **46** (1), 38-47.

CATTYRSSE, D., SALOMON, M., KUIK, R., and VAN WASSENHOVE. L. N., 1993, A dual ascent and column generation heuristic for discrete lot sizing and scheduling problem with setup times. *Management Science*, **39** (4), 477-486.

CHEN, W. H., and THIZY, J. M., 1987, Analysis of relaxations for the multi-item capacitated lot sizing problem. *Technical Report 86-1-3* (School of Management and Department of Industrial Engineering, NY: State University at Buffalo).

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

CHOO, E. U. and CHAN, G. H., 1990, Two-way eyeballing heuristics in dynamic lot-sizing with backlogging. *Computers and Operations Research*, **17** (4), 359-363.

DAVIS, S. G., 1990, Scheduling economic lot size production runs. *Management Science*, **36** (8), 985-998.

DILTS, D. M. and RAMSING, K. D., 1989, Joint lot sizing and scheduling of multiple items with sequence dependent setup costs. *Decision Sciences*, **20** (1), 120-133.

DIXON, P. S. and SILVER, E. A., 1981, A heuristic solution procedure for the multi-item, single-level, limited capacity, lot-sizing problem. *Journal of Operations Management*, **2** (1), 23-39.

DOGRAMACI, A., PANAYIOTOPOULOS, J. C., and ADAM, N. R., 1981, The dynamic lot-sizing problem for multiple items under limited capacity. *AIIE Transactions*, **13** (4), 294-303.

EISENHUT, P. S., 1975, A dynamic lot sizing algorithm with capacity constraints. *AIIE Transactions*, 7 (2), 170-176.

ELMAGHRABY, S. E., 1978, The economic lot scheduling problem (ELSP): review and extensions. *Management Science*, **24** (6), 587-598.

FLEISCHMANN, B., 1990, The discrete lot-sizing and scheduling problem. *European Journal of Operational Research*, **44** (3), 337-348.

FLEISCHMANN, B., 1994, The discrete lot-sizing and scheduling problem with sequence dependent setup costs. *European Journal of Operational Research*, **75** (2) 395-404.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

FLORIAN, M., LENSTRA, J. K., and RINNOOY KAN, A. H. G., 1980, Deterministic production planning algorithms and complexity. *Management Science*, **26** (7), 669-679.

GOPALAKRISHNAN, M., MILLER, D. M., and SCHMIDT, C. P., 1995, A framework for modeling setup carryover in the capacitated lot sizing problem. *International Journal of Production Research*, **33** (7), 1973-1988.

HANSSMANN, F., 1962, *Operations Research in Production and Inventory* (New York, NY: John Wiley and Sons)

HARRIS, F. W., 1913, How many parts to make at once. *Factory, The Magazine of Management*, **10** (2), 135-136.

HSIEH, H., LAM, K. F., and CHOO, E. U., 1992, Comparative study of dynamic lot sizing heuristics with backlogging. *Computers and Operations Research*, **19** (5), 393-407.

HSU, W., 1983, On the general feasibility test of scheduling lot sizes for several products on one machine. *Management Science*, **29** (1), 93-105.

KARMARKAR, U. S. and SCHRAGE, L., 1985, The deterministic dynamic product cycling problem. *Operations Research*, **33** (2), 326-345.

KIM, D. and MABERT, V. A., 1995, Integrative versus separate cycle scheduling heuristics for capacitated discrete lot sizing and sequencing problems. *International Journal of Production Research*, **33** (7), 2007-2021.

KIRCA, O. and KOKTEN, M., 1994, A new heuristic approach for the multi-item dynamic lot sizing problem. *European Journal of Operational Research*, **75** (2), 322-341.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

KOEPKE, C. A., 1941, *Plant Production Control* (New York, NY: John Wiley and Sons Inc.).

LAMBRECHT, M. R. and VANDERVEKEN, H., 1979, Heuristic procedures for the single operation, multi-item loading problem. *AIIE Transactions*, **11** (4), 319-326.

LAWLER, E. L., 1976, *Combinatorial Optimization: Networks and Matroids* (New York, NY: Holt, Rinehart and Winston).

LAWLER, E. L., LENSTRA, J. K., RINNOOY KAN, A. H. G., and SHMOYS, D. B., 1985, *The Traveling Salesman Problem* (New York NY: John Wiley and Sons).

LITTLE, J. D. C., MURTY, K. G., SWEENEY, D. W., and KAREL, C., 1963, An algorithm for the traveling salesman problem. *Operations Research*, **11** (6), 972-989.

MAES, J. and VAN WASSENHOVE, L. N., 1986, Multi item single level capacitated dynamic lot sizing heuristics: a computational comparison (part 1: static case). *IIE Transactions*, **18** (2), 114-123.

MANNE, A.S., and VEINOTT, A. F., 1967, Optimal plant size with arbitrary increasing time of paths demand. *Investments for Capacity Expansion: Size, Location and Time Phasing*, A. S. Manne (ed) (Cambridge, MA: The M.I.T. Press), pp. 178-190.

MATHUR, K and SOLOW, D., 1994, *Management science: The Art of Decision Making* (Englewood Cliffs, NJ: Prentice Hall).

NETER, J., WASSERMAN, W., and KUTNER, M. H., 1990, *Applied Linear Statistical Models*, 3rd edn (Homewood, IL: IRWIN)

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

RITCHIE, E., TSADO, A. K., 1986, A review of lot-sizing techniques for deterministic time-varying demand. *Production and Inventory Management Journal*, **27** (3), 65-79.

SALOMON, M., 1991, *Deterministic Lot Sizing Models for Production Planning* (New York, NY: Springer-Verlag).

SILVER, E. A., and MEAL, H. C., 1969, A simple modification of the EOQ for the case of a varying demand rate. *Production and Inventory Management Journal*, **10** (4), 52-65.

TAHA, H. A., 1975, *Integer Programming: Theory, Applications and Computations* (Academic Press).

THIZY, J. M., and VAN WASSENHOVE, L. N., 1985, Lagrangean relaxation for the multi-item capacitated lot-sizing problem: a heuristic implementation. *IIE Transactions*, **17** (4), 308-313.

TRIGEIRO, W. W., THOMAS, L. J., and MCCLAIN, J. O., 1989, Capacitated lot sizing with setup times. *Management Science*, **35** (3), 353-366.

TRIGEIRO, W. W., 1989, A simple heuristic for lot sizing with setup times. *Decision Sciences*, **20** (2), 294-303.

TURBIDE, D. A., 1995, MRP II: still number one. *Industrial Engineering Solutions*, **27** (7), 28-31.

WAGNER, H. M., and WHITIN, T. M., 1958, Dynamic version of the economic lot size model. *Management Science*, **5** (1), 89-96.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

WALLACE, T. F., 1990, *MRP II: Making It Happen: The Implementer's Quick Guide to Success with Manufacturing Resource Planning*, (Essex Junction, VT: O. Wight Ltd.).

ZANGWILL, W. I., 1966, A deterministic multi-period production scheduling model with backlogging. *Management Science*, **13** (1), 105-119.

ZANGWILL, W. I., 1969, A backlogging model and a multi-echelon model of a dynamic economic lot size production system-a network approach. *Management Science*, **15** (9), 506-527.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX A

NUMERICAL EXAMPLE

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Consider the 4 product 5 period problem with demand matrix given in Table A.1 and setup time matrix in Table A.2. Demand as well as setup time is expressed in hours of production.

	Period				
Product	1	2	3	4	5
A	3	0	0	2	0
B	0	6	0	1	4
C	2	0	3	0	0
D	0	0	5	0	2

Table A.1 Demand Matrix

	To Product			
From	A	B	C	D
A	0	1	1	2
B	2	0	1	2
C	1	3	0	1
D	1	1	2	0

Table A.2 Setup Matrix

In addition capacity is 8 hours per day $C_t = 8 \ \forall \ t$; Setup cost factor = 3, i.e. $S_{ij} = 3 * s_{ij}$; backlogging limit $\alpha = 2$; $h_i = 1 \ \forall \ i$; $b_i = 2 \ \forall \ i$. The problem will be solved using the minimum setup estimator (MIN) to convert sequence dependent setup to sequence independent setup.

No Lot Sizing

In this method all the requirements over the scheduling horizon is produced in one lot. Lot size for items are obtained by adding up their demand over the entire

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

horizon. For example the lot size for product A is 3+2 = 5. Similarly, lot size for other items would be 11, 5 and 7 for items B, C and D respectively. The sequencing problem here is simply to determine the sequence that provides the lowest sum of setup costs. From inspection of the setup time matrix the optimal sequence A→B→C→D is determined, with a total setup time of 3 and a setup cost of 9 (no setup time is levied for product A as per convention defined in chapter I). Taking capacity considerations into account, production is scheduled as shown in Table A.3. Table A.4 indicates the number of units in inventory/backlogged at the end of the period.

	Period				
Product	1	2	3	4	5
A	5	0	0	0	0
B	2	8	1	0	0
C	0	0	5	0	0
D	0	0	0	7	0

Table A.3 No Lot Size Schedule

	Period				
Product	1	2	3	4	5
A	2	2	2	0	0
B	2	4	5	4	0
C	-2	-2	0	0	0
D	0	0	-5	2	0

Table A.4 No Lot Size Inventory/Backlogging

The total inventory cost is obtained by adding up all the positive values in Table A.4 and multiplying it by holding cost factor ($h_i = 1$). Similarly, backlogging cost is obtained by adding up all the negative values in Table A.4 and multiplying by

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

backlogging cost factor ($b_i = 2$). Therefore, the inventory/backlog cost is 23*1 + 9*2 = 41. There are only three setups performed with total setup time of 3 hours. Using setup cost per hour factor of 3 we get a total setup cost of 9. Therefore the total cost of the schedule is 9 + 41 = 50.

EMQ Calculations

The MIN setup estimator is used to estimate the setup time s_A = min (2, 1, 1) = 1; similarly the setup time estimates for other products can be obtained; $s_B = 1$; $s_C = 1$ and $s_D = 1$. Using the setup cost factor (f=3) the estimates of setup costs can be calculated $S_A = 3*1 = 3$; similarly $S_B = 3$; $S_C = 3$; $S_D = 3$. The average demand values per period (r_i) is calculated next: $r_A = \dfrac{3+2}{5} = 1$; similarly $r_B = 2.2$; $r_C = 1$; $r_D = 1.4$. $p_i = 8$ for all i. Therefore, $\rho_A = 1/8$; $\rho_B = 11/40$; $\rho_C = 1/8$; $\rho_D = 7/40$. Now we can calculate the cycle time for each product using $\tau_i^* = \sqrt{\dfrac{2S_i}{h_i r_i (1-\rho_i)}}$. This yields

$\tau_A^* = \sqrt{\dfrac{2 \times 3}{1 \times 1 \times (1 - \frac{1}{8})}} = 2.619$; similarly $\tau_B^* = 1.94$; $_C^* = 2.619$;

$\tau_D^* = 2.279$ and the lot sizes are calculated using $r_i \tau_i^*$; for product A lot size = 1x2.619 = 2.619 which is rounded off to 3; similarly the lot sizes for other products can be calculated as 4, 3, and 3 respectively for items B, C, and D.

For products A and C only 2 lots are required to meet their cumulative demand. Whereas, products B and D need 3 lots. Since the setup time to switch from D to A is

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

only 1 hour the optimal sequence remains the same. Making the required changes as stated in chapter IV we get the schedule shown in Table A.5 and Inventory/Backlogging situation shown in Table A.6.

	Period				
Product	1	2	3	4	5
A	3	0	2	0	0
B	4	0	4	0	3
C	0	3	0	2	0
D	0	3	0	3	3

Table A.5 EMQ Schedule

	Period				
Product	1	2	3	4	5
A	0	0	2	0	0
B	4	-2	2	1	0
C	-2	1	-2	0	0
D	0	3	-2	1	0

Table A.6 EMQ Inventory/Backlogging

The complete sequence of production is A→B→C→D→A→B→C→D→B→D. This results in a total setup cost of 30. The total inventory/backlogging cost is $14*1 + 8*2 = 30$, yielding a total cost of $30 + 31 = 61$.

A Lot Shifting Algorithm (LS)

The sequence independent setup costs and times are once again calculated using MIN estimator as $S_A = 3$; $S_B = 3$; $S_C = 3$; $S_D = 3$ and $s_A = 1$; $s_B = 1$; $s_C = 1$; $s_D = 1$

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 1 The algorithm initially sets schedule = demand = $\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 6 & 0 & 1 & 4 \\ 2 & 0 & 3 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}$. In this matrix each column represents a day and each row represents a product. For example according to the above schedule the production lot for product B on day 4 is 1.

Step 2 In this step the schedule matrix is sequenced. Here there are 5 stages corresponding to each day in the horizon. Number of states in each stage is equal to the factorial of number non zero lots produced in that state. Stages 1, 3, 4 and 5 have 2 states each and stage 2 has 1 state. For example the 2 possible states for stage 1 are (A,C) and (C,A). Two dummy stages (stage 6 and stage 0) represent the transition from this scheduling horizon to the adjacent horizons.

The dynamic programming algorithm starts by first calculating the optimal setup cost sequence within each state. When there are several non-zero products in a period then a TSP algorithm is used to calculate the best sequence. However, in this situation this is a trivial problem as the maximum number of products produced on any day is 2. In this case the cost of sequence within each state is merely equal to the setup cost between the first product and the last product in the state. In stage 2 there is only one non zero lot (product B), in which case the cost of the state is zero.

The dynamic programming algorithm starts in stage 5 and goes back up to stage 0. For each state in a stage the algorithm determines the state in the next stage that produces the lowest cost sequence from that stage to stage 5 (equation 64). These calculations are shown in table A.7. The state in the next stage that produces the lowest

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

cost to stage 5 is shown in *italics*. Costs shown in the table are sum of cost of sequence within the state, cost of switching from the last product in the state to the first product in the state selected in next stage, and cost of going from state selected in next stage to stage 5. For example the cost of going from state (C,D) in stage 3 to stage 5 when the state in the next stage is (A,B) is the sum of:

1. cost of sequence (C,D) = 3,
2. cost of changing from product D in stage 3 to A in stage 4 = 3, and
3. cost from state (A,B) in stage 4 to stage 5 = 9,

resulting in a total cost of 15.

Stage	State	State in next Stage	Cost
6	(0)	-	0
5	(B,D)	(0)	6
	(D,B)	(0)	3
4	(A,B)	*(B,D)*	*9*
		(D,B)	12
	(B,A)	*(B,D)*	*15*
		(D,B)	*15*
3	(C,D)	*(A,B)*	*15*
		(B,A)	21
	(D,C)	*(A,B)*	*18*
		(B,A)	30
2	(B)	*(C,D)*	*18*
		(D,C)	24
1	(A,C)	(B)	30
	(C,A)	*(B)*	*24*
0	(0)	(A,C)	30
		(C,A)	*24*

Table A.7 Cost of states

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The algorithm then traces the optimal sequence starting from stage 0, at each stage selecting the state in the next stage that provides the lowest cost to stage 6 (equations 65 and 66). In relation to Table A.7 the optimal sequence is determined as follows:

- at stage 0 select state (C,A) in stage 1 as it has a lower cost than state (A,C)
- at stage 1 the only option in stage 2 is state B
- at stage 2 the best state in the next stage is state (C,D)
- for state (C,D) in stage 3 the best state in stage 4 is (A,B)
- for state (A,B) in stage 4 the best state in stage 5 is (B,D)

The sequence of production in each stage and across the scheduling horizon can be represented using the matrix $\begin{pmatrix} C & B & C & A & B \\ A & & D & B & D \end{pmatrix}$. In this matrix the columns represent each stage (period) and the rows represent the sequence of production starting from the first row. Capacity availability/overloading for each period is calculated next. For example consider period 3. In this period capacity is required to switch production from product B in period 2 to product C, produce 3 units of product C, then switch to product D and produce 5 units of product D. The total capacity required in this period then is $1 + 3 + 1 + 5 = 10$. Since the capacity available in each period is 8 we have an overload of 2 units. Overload is represented with a negative sign in the capacity availability matrix. Similar calculations for the other periods yields a capacity matrix of (2, 1, -2, 3, 0) for the schedule and sequence matrix listed above.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Step 3 In this step entire lots that can be moved to save setup costs are considered. Taking capacity availabilities into account entire lots that can be considered for moving are product A from period 4 to period 1 and product B from period 4 to period 2. The first move costs 6 units to carry additional inventory from period 1 to period 4 and saves a setup for product A estimated to be 3 units ($S_A = 3$). Therefore the move costs more than it saves and is not selected. Whereas the second move costs 2 and potentially saves 3 units, so this move is made. The schedule is recomputed, and the new schedule is

$$\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 7 & 0 & 0 & 4 \\ 2 & 0 & 3 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}.$$ The sequence of production and capacity availabilities are recalculated as before, the sequence is $\begin{pmatrix} C & B & C & A & B \\ A & & D & & D \end{pmatrix}$ and capacities matrix is (2, 0, -2, 5, -1). No more lots can be moved around to save setup.

Step 4 Overloading can be eliminated in period 5 by moving setup item B to period 4 (equation 72 is satisfied). The sequences and lot sizes remain the same but the new capacity matrix is (2, 0, -2, 4, 0) (using equations 73 and 74). Now the overloading in period 3 has to be eliminated.

Both items C and D have to be considered. First the algorithm considers item C. The estimated setup time is 1 hours ($s_C = 1$). Two options are available.

1. Shift some production to period 1 (equation 87 is satisfied). The number of units to be shifted is determined by equation (88), in this case v = min(2, 3, 2). The cost of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

this move per unit overload eliminated is calculated using equation (89). the change in the inventory/backlogging cost is calculated using equations (69, 70 and 71). Plugging in all the numbers into these equations we get ΔH = 4; $z = \frac{\Delta H}{v} = \frac{4}{2} = 2$.

2. Another alternative is to move production into period 4. Equation (83) is satisfied. The entire lot for product C can be moved (3 units). Since, production is shifted forward the feasibility of this move has to be evaluated. Equation (98) is satisfied (t=3, t* = 4, m=3). Therefore, the move is a feasible move. Once again the cost per unit shifted is calculated using equation (84). Plugging in all the numbers, we get $z = \frac{\Delta H}{x_{it}} = \frac{6}{3} = 2$.

The above procedure is repeated for product D. Equation (92) is satisfied by both periods 1 and 4. Let us first consider shifting production to period 1. In this case v can be calculated form equation (93), v = min(2, 1,5) = 1. The cost per unit of infeasibility eliminated is calculated from equation (94), $z = \frac{3+2}{1} = 5$ Now let us consider shifting production to period 4. Once again the feasibility of the move is evaluated. Using equation (93) the size of the lot to be shifted is calculated v = min(2, 3, 5) = 2. The cost of this move per unit of overload eliminated is $z = \frac{\Delta H + S_C}{v} = \frac{4+3}{2} = 3.5$. Obviously the lowest cost option is to move two units of product C from period 3 to period 1 or 3 units from period 3 to period 4. Shifting 2 units to period 1 is selected.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The new schedule is $\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 7 & 0 & 0 & 4 \\ 4 & 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}$ the sequence remains the same and is

$\begin{pmatrix} C & B & C & A & B \\ A & & D & & D \end{pmatrix}$ the new capacities are (0, 0, 0, 4, 0). Hence the infeasibility has been eliminated.

Step 5. The quality of the solution cannot be further enhanced as no conditions required for optimality have been violated.

Now the inventory/backlogging matrix for the above schedule is calculated

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

The total setup cost is calculated from the over all sequence of production C→A→B→C→D→A→B→D. The total setup time for this sequence is 8, and the total cost is 24 (8*3). Once again the total inventory/backlogging cost is 6 (from inventory/backlogging matrix). Therefore, total cost of the schedule is 24+6 = 30.

A Regeneration Algorithm (RA)

Once again $S_A = 3$; $S_B = 3$; $S_C = 3$; $S_D = 3$ and $s_A = 1$; $s_B = 1$; $s_C = 1$; $s_D = 1$.

Step 1. Solve the single item problem for each item using the regeneration point technique described in chapter III. The algorithm first considers item A with a demand pattern (3 0 0 2 0). The best period for production between any two regeneration points

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

is calculated using equation (57). For example given a regeneration point at period 0, Table A.8 shows the cost of production in each period for the next regeneration point. The values in Table A.8 are equal to $c_{0k}^{t'}$.

For example, consider the first regeneration point in period 0 and the next regeneration point in period 3. There are three possible periods (1, 2, 3) to produce the lot to meet the cumulative demand in this period (cumulative demand = 3). If lot is produced in period 1, then inventory at the end of this period becomes 0 (production = 3 units and demand = 3 units). This makes period 1 a regeneration point. This is not a feasible solution because it violates that the rule that the next regeneration point following period 0 is period 3. Next consider production in period 2. The setup cost for the period is 3 and the backlogging cost is 6. Therefore the total cost of production in period 2 is 9. Similarly the total cost of production in period 3 is 12 + 3 = 15.

	Period of Production				
Next Regeneration Period	1	2	3	4	5
1	3	-[1]	-	-	-
2	x[2]	9	-	-	-
3	x	9	15	-	-
4	9	13	17	x	-
5	x	x	x	x	x

1 - indicates periods that are not within regeneration points 0 and the next regeneration point
2 x indicates infeasible periods as producing in these periods violates regeneration point theorem

Table A.8 Cost coefficients for regeneration in period 0

Similar cost coefficients are generated for all other possible regeneration points. The period that provides the lowest cost period of production between two successive regeneration points is saved. Table A.9 shows the lowest cost between two regeneration

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

points and period of production to achieve this low cost. The cost coefficients shown in Table A.9 are c_{tk} for all combinations of t and k.

	Next regeneration point									
	lowest cost coefficient					period of production				
regeneration at point i	1	2	3	4	5	1	2	3	4	5
0	3	9	9	9	x	1	2	2	1	x
1	-[1]	0	x[2]	7	x	-	2	x	2	x
2	-	-	0	5	x	-	-	3	3	x
3	-	-	-	3	7	-	-	-	4	5
4	-	-	-	-	0	-	-	-	-	5

1 - indicates periods that are not within regeneration points considered
2 x indicates infeasible periods as producing in these periods violates regeneration point theorem

Table A.9 Cost coefficients and period of production between all regeneration points

For example consider first regeneration in period 2. Three possibilities exist for the next regeneration period namely, 3, 4 and 5. Let us consider each one in sequence. First consider period 3. Cumulative demand in this period is 0. Therefore, the total cost of production is 0 and the only period in which this 0 production can be scheduled is 3.

Next, consider the second regeneration period 4. The cumulative demand now is 2 (sum of demand in periods 3 and 4). Production of these 2 units can either be scheduled in period 3 or 4. First consider period 3. The total production cost is 5 (3 for setup and 2 for inventory). Now consider production in period 4. The inventory at the end of period 3 in this case is 0. Therefore, production cannot be scheduled in period 4. So the only option is producing 2 units in period 3 at a cost 5, as shown in Table A.9.

Now consider the second regeneration point in period 5. Once again the cumulative demand is 2. If production is scheduled in period 3 then inventory at end of

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

period 4 is 0. Hence, this is not a feasible option. Similarly if production is scheduled in periods 4 or 5 inventory at the end of period 3 is 0, which makes production in these periods infeasible. Therefore, there is no feasible solution possible for a regeneration point in period 2 and the next one in period 5.

Once the lowest costs between two regeneration points have been determined the task is to use dynamic programming to select the best possible pairs of regeneration periods in each period. This is achieved by recursion starting in period 5. The recursion works as follows:

$$f_5 = 0$$

$$f_4 = \min_{4<k\leq 5}(c_{4k} + f_k) = c_{45} + f_5 = 0$$

$$f_3 = \min_{3<k\leq 5}(c_{3k} + f_k) = c_{34} + f_4 = 3$$

$$f_2 = \min_{2<k\leq 5}(c_{2k} + f_k) = c_{23} + f_3 =$$

$$f_1 = \min_{1<k\leq 5}(c_{1k} + f_k) = c_{12} + f_2 = 3$$

$$f_0 = \min_{0<k\leq 5}(c_{0k} + f_k) = c_{01} + f_1 = 6$$

where f_t is the lowest cost of going from that stage t to the regeneration point in period 5. f_t is calculated recursively starting from period 5 using equation (58). For example, consider stage 3. Given a regeneration period in 3 then there are two options to get to stage 5. The first option is to directly have the second regeneration point in period 5 with total cost of 7 (c_{35} +f_5). The second option is to have the second regeneration point

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

in period 4 and then select the best way to go from period 4 to period 5. The total cost of this option is 3 ($c_{34} + f_4$). Therefore the second option is selected.

Similarly the best path to period 5 is calculated for other periods and is shown above. The optimal regeneration points are selected by starting from period 0 and successively determining the next lowest cost regeneration point. From period 0 the next lowest cost regeneration period is 1, similarly from 1 the next lowest cost regeneration period is period 2 and so on. The sequence of regeneration points is the optimal sequence.

Therefore the best policy will be to have regeneration points at 0, 1, 2, 3, 4 and 5 i.e. just producing enough in each period to meet demand in that period. Here the lots are scheduled in periods 1 and 4, and the lot sizes are 3 and 2 respectively. The total cost of the schedule is 6. Similarly, the optimal schedules are generated for all products. At the expense of being succinct only the final optimal schedules are presented for the other items. For item B there are two optimal schedules (0, 7, 0, 0, 4) or (0, 6, 0, 0,5). For item C the optimal schedule is (2, 0, 3, 0, 0) and for item D the optimal schedule is (0, 0, 5, 0, 2). Hence at the end of the first step the following schedule can be obtained

$$\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 7 & 0 & 0 & 4 \\ 2 & 0 & 3 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}.$$

<u>Step 2</u>. The above schedule is exactly same as the one generated at the end of the lot shifting procedure of the LS algorithm. Therefore the infeasibility elimination procedure is similar. Hence only the final solution is provided here. The schedule is

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

$\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 7 & 0 & 0 & 4 \\ 4 & 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}$ and the sequence is $\begin{pmatrix} C & B & C & A & B \\ A & & D & & D \end{pmatrix}$ once again this approach yields the final schedule cost of 30.

Regeneration Algorithm with "Overload Penalty" (ROP)

Still $S_A = 3$; $S_B = 3$; $S_C = 3$; $S_D = 3$ and $s_A = 1$; $s_B = 1$; $s_C = 1$; $s_D = 1$. Also we know $r_A = 1$; $r_B = 2.2$; $r_C = 1$; $r_D = 1.4$.

Step 1. First the sequence in which the regeneration principle is used to generate optimal solution is determined. The sequence is based on the average cost values (V_i) determined for each product using equation (106). For product A we have

$A = \dfrac{\sqrt{2 \times S_A \times r_A}}{r_A} = \dfrac{\sqrt{2 \times 3 \times 1}}{1} = 2.45$ similarly $V_B = 1.65$; $V_C = 2.45$; and $V_D = 2.07$. The average cost for items C and A are equal. In such cases the algorithm selects the product whose average cost value was first generated. Therefore, the order of scheduling is A, C, D, B.

Step 2. This algorithm is similar to RA algorithm until the 'if condition' in equation (108) is satisfied. Since the "overload penalty" does not come into play for the first two products the final results for these products determined earlier are merely repeated here. The schedule for A is (3, 0, 0, 2, 0) and for C is (2, 0, 3, 0, 0). Next item D is considered for scheduling. Now the 'if condition' in equation (108) is satisfied and "overload penalty" comes into play.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

First the capacity requirements in each period is calculated using estimated setup times. For example in period 1 for A and C together we need total production time of 5 (3+2), the total estimated setup time is 2. Therefore total capacity required for A and C in period 1 is 7 (5+2). Similar capacity requirements are calculated in other periods and the capacity requirements in each period is (7, 0, 4, 3, 0). From these values the capacity available in each period can be computed by subtracting capacity requirements from capacity available in each period (8). The capacity availability matrix is (1, 8, 4, 5, 8).

With information on capacity availabilities the optimal regeneration periods can be calculated as before. However, this time around some of the cost coefficients will be modified by incorporation of "overload penalty". The demand sequence for product D is (0, 0, 5, 0, 2). Once again the cost of production in each period for a given pair of regeneration points is determined. However, now the $c_{tk}^{t'}$ values are computed using equation (109) to account for "overload penalty". Table A.10 shows the cost coefficients with the first regeneration point in period 0.

For example consider the next regeneration in period 5. One option is to produce in period 1. Cumulative demand is 7 (5+2). Total capacity required to produce a lot of size 7 is 8 (including an estimated setup time of 1 hour). However, only 1 hour of capacity is available in this period. "Overload penalty" is calculated using equation (109) as 49 $(8-1)^2$. Now the cost coefficient in that period is the sum of "overload penalty", setup and inventory cost to carry 5 units from period 1 to period 3. "Overload penalty" is 49, estimated setup cost is 3, and inventory cost is 10 (5x2). The cost

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

coefficient is 70 (49+3+18). Production in other periods is not feasible because inventory in period 1 is 0 with these options.

	Period of Production				
Next Regeneration Period	1	2	3	4	5
1	0	-[1]	-	-	-
2	x[2]	x	-	-	-
3	38	x	x	-	-
4	x	x	x	x	-
5	70	x	x	x	x

1 - indicates periods that are not within regeneration points 0 and the next regeneration point
2 x indicates infeasible periods as producing in these periods violates regeneration point theorem

Table A.10 Cost coefficient for regeneration in period 0 with "overload penalty"

Similar cost values are generated for all other possible regeneration points. The period that provides the lowest cost period of production between two successive regeneration points is saved. Table A.11 shows the lowest cost between two regeneration points and period of production to achieve this low cost. The values in Table A.11 is equal to c_{tk} for all combinations of t and k. These computations are similar to one used in RA except for the incorporation of the "overload penalty".

	Next regeneration point									
	lowest cost coefficient					period of production				
regeneration at point i	1	2	3	4	5	1	2	3	4	5
0	0	x	38	x	70	1	x	1	x	1
1	-[1]	0	8	x	14	-	2	2	x	2
2	-	-	7	14	23	-	-	3	4	3
3	-	-	-	0	5	-	-	-	4	4
4	-	-	-	-	3	-	-	-	-	5

1 - indicates periods that are not within regeneration points considered
2 x indicates infeasible periods as producing in these periods violates regeneration point theorem

Table A.11 Cost coefficients and period of production between all regeneration points

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

The recursion would work as follows:

$f_5 = 0$

$f_4 = \min_{4<k\leq 5}(c_{4k} + f_k) = c_{45} + f_5 = 3$

$f_3 = \min_{3<k\leq 5}(c_{3k} + f_k) = c_{34} + f_4 =$

$f_2 = \min_{2<k\leq 5}(c_{2k} + f_k) = c_{23} + f_3 = 10$

$f_1 = \min_{1<k\leq 5}(c_{1k} + f_k) = c_{12} + f_2 = 10$

$f_0 = \min_{0<k\leq 5}(c_{0k} + f_k) = c_{01} + f_1 = 10$

The optimal solution is to have regeneration points at (0, 1, 2, 3, 4, 5). Once again the solution is to produce only in periods in which demand exists i.e., produce 5 units in period 3 and 2 units in period 5. The schedule is (0, 0, 5, 0, 2). Now item B is scheduled, taking into account the schedules of previous three items. Once again the regeneration points are determined using procedure described above. The schedule for item B is (0, 6, 0, 0, 5).

Step 3. The combined schedule of the four items are $\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 6 & 0 & 0 & 5 \\ 2 & 0 & 3 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}$. The optimal sequence is $\begin{pmatrix} C & B & C & A & D \\ A & & D & & B \end{pmatrix}$ with capacities (2, 1, -2, 5, -2).

Step 4. The overloading in period 5 can be overcome by moving the setup for the switch between product A in period 4 and product D in period 5 to period 4. The new

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

capacities are (2, 1, -2, 3, 0). Overloading in period 3 can be eliminated using methods described earlier. Here only the final solution is provided. The final schedule is

$$\begin{pmatrix} 3 & 0 & 0 & 2 & 0 \\ 0 & 7 & 0 & 0 & 4 \\ 4 & 0 & 1 & 0 & 0 \\ 0 & 0 & 5 & 0 & 2 \end{pmatrix}$$ with sequence shown as $\begin{pmatrix} C & B & C & A & D \\ A & & D & & B \end{pmatrix}$, and the total cost is 30.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX B

DEMAND AND SETUP DATA SETS

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.1 Demand data set 1: Small problem size, smooth demand, low utilization

					Period					
Product	1	2	3	4	5	6	7	8	9	10
1	2	2	1	4	2	1	1	2	3	1
2	3	3	3	1	3	1	4	2	2	2
3	1	2	2	2	1	2	2	2	3	1
4	1	1	2	3	1	3	1	1	1	3
5	2	2	2	1	3	3	2	3	1	2
6	3	2	2	1	2	2	2	2	2	3
1	2	2	2	2	2	4	4	2	2	2
2	2	2	1	2	2	3	2	1	2	2
3	2	2	2	2	3	1	1	2	2	2
4	2	2	2	3	2	2	1	1	3	2
5	2	2	2	2	1	1	2	3	2	2
6	2	2	3	1	2	1	2	3	1	2
1	4	1	1	2	2	3	2	3	2	2
2	1	1	2	1	2	4	3	2	2	3
3	2	2	2	2	2	1	2	2	2	1
4	1	2	3	1	3	1	2	2	2	3
5	2	3	1	5	2	1	2	2	3	2
6	2	3	3	1	1	2	1	1	1	1
1	2	2	3	1	1	1	1	2	3	1
2	1	3	3	2	2	2	3	1	2	3
3	2	1	1	2	2	2	1	2	3	4
4	4	3	1	3	2	2	2	2	1	1
5	2	2	2	1	3	1	2	3	1	2
6	1	1	2	3	2	4	3	2	2	1
1	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2
3	2	2	2	2	2	2	2	2	2	2
4	2	2	2	2	2	2	2	2	2	2
5	2	2	2	2	2	2	2	2	2	2
6	2	2	2	2	2	2	2	2	2	2
1	1	2	1	2	1	3	2	1	2	3
2	2	1	2	4	2	2	2	2	3	1
3	2	2	3	1	2	1	2	2	1	2
4	3	3	2	2	1	1	1	2	1	1
5	2	1	2	1	4	2	2	2	3	2
6	2	3	2	2	2	3	3	3	2	3
1	2	1	1	1	3	2	2	3	2	2
2	2	2	3	3	1	3	2	2	2	2
3	5	2	1	2	2	1	1	2	2	1
4	1	3	3	3	2	1	2	1	1	2
5	1	3	1	2	2	3	3	2	2	3
6	1	1	3	1	2	2	2	2	3	2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.1 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10
1	4	3	3	2	3	2	2	1	3	2
2	2	2	1	2	1	2	2	3	1	2
3	2	2	2	1	2	2	2	1	3	2
4	1	1	2	3	1	2	3	3	2	1
5	2	2	2	2	2	1	2	1	1	3
6	1	2	2	2	3	3	1	3	2	2
1	3	3	2	2	2	2	1	2	2	2
2	1	2	2	2	3	3	2	2	2	2
3	2	3	3	4	2	2	2	3	2	2
4	3	1	1	2	1	1	3	2	2	2
5	1	2	3	1	2	2	2	1	2	2
6	2	1	1	1	2	2	2	2	2	2
1	3	1	2	2	2	3	1	1	1	3
2	2	3	2	2	1	1	2	1	3	1
3	1	2	2	3	3	1	2	1	3	2
4	3	3	1	1	1	3	2	2	3	2
5	1	1	3	3	2	2	4	4	1	2
6	2	2	2	1	3	2	1	3	1	2

Table B.2 Demand data set 2: Small problem size, smooth demand, high utilization

Product	Period 1	2	3	4	5	6	7	8	9	10
1	3	3	3	3	4	5	3	3	2	4
2	4	2	4	3	1	2	4	3	4	2
3	1	3	4	4	4	4	3	3	3	4
4	3	4	1	2	3	3	3	3	2	2
5	4	4	3	3	2	2	3	3	3	4
6	3	2	3	3	4	2	2	3	4	2
1	2	3	2	4	2	4	3	2	4	3
2	3	3	2	4	2	3	4	3	4	2
3	4	3	3	3	5	2	4	4	2	2
4	2	4	4	2	3	2	3	4	3	3
5	4	2	4	3	3	3	2	3	3	5
6	3	3	3	2	3	4	2	2	2	3
1	4	4	4	2	2	1	4	4	2	4
2	2	3	4	2	2	4	2	3	4	2
3	2	3	2	3	2	4	3	4	4	4
4	2	3	3	5	4	4	3	3	2	3
5	3	2	3	3	4	2	3	3	2	3
6	5	3	2	3	4	3	3	1	4	2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.2 (Continued)

	Period									
Product	1	2	3	4	5	6	7	8	9	10
1	3	4	2	4	2	4	2	3	2	4
2	4	3	4	2	4	2	3	4	2	2
3	2	3	3	3	4	4	3	2	4	3
4	2	4	2	5	2	4	4	4	3	3
5	3	2	3	2	3	2	3	3	4	3
6	4	2	4	2	3	2	3	2	3	3
1	3	4	4	4	4	3	3	2	2	4
2	4	3	4	2	1	2	4	2	2	4
3	2	2	2	4	3	3	3	3	4	2
4	2	4	4	2	2	4	4	3	4	1
5	4	2	2	2	4	4	2	5	4	4
6	3	3	2	4	4	2	2	3	2	3
1	4	5	4	2	3	2	2	2	4	2
2	2	3	3	4	3	3	3	4	4	4
3	3	2	3	4	2	4	3	3	2	4
4	4	4	2	3	3	3	4	3	3	1
5	3	2	3	2	3	4	4	3	3	3
6	2	2	3	3	4	2	2	3	2	4
1	3	3	3	3	3	3	3	3	3	3
2	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3
4	3	3	3	3	3	3	3	3	3	3
5	3	3	3	3	3	3	3	3	3	3
6	3	3	3	3	3	3	3	3	3	3
1	4	2	2	4	2	2	4	5	4	1
2	2	3	3	1	3	2	2	2	3	4
3	4	3	4	3	4	4	3	3	2	3
4	3	3	2	4	3	4	4	4	3	3
5	2	3	4	3	4	2	2	2	4	3
6	3	4	3	3	2	4	3	2	2	4
1	4	4	2	3	2	3	3	3	4	4
2	2	3	2	2	3	4	4	2	3	4
3	3	2	3	3	4	2	2	3	2	3
4	3	3	4	3	4	3	2	2	4	2
5	3	4	3	3	3	3	4	4	3	1
6	3	2	4	4	2	3	3	4	2	4
1	2	4	3	3	3	3	3	2	4	2
2	2	3	4	2	3	5	3	4	4	3
3	2	2	2	4	3	4	4	3	2	2
4	3	4	2	3	2	2	2	4	3	3
5	4	3	4	3	3	2	3	3	3	4
6	5	2	3	3	4	2	3	2	2	4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.3 Demand data set 3: Small problem size, lumpy demand, low utilization

	Period									
Product	1	2	3	4	5	6	7	8	9	10
1	5	6	0	0	0	2	0	0	0	4
2	4	0	0	4	0	2	0	0	0	2
3	0	0	0	0	0	4	5	12	0	4
4	3	6	0	0	0	3	0	0	0	2
5	0	0	0	4	6	1	0	0	12	0
6	0	0	12	4	6	0	7	0	0	0
1	0	2	0	0	0	0	2	0	12	6
2	0	4	5	5	6	0	0	0	0	0
3	0	3	3	0	4	0	0	0	0	6
4	8	0	4	4	0	0	0	12	0	0
5	4	0	0	0	2	3	3	0	0	0
6	0	3	0	3	0	9	7	0	0	0
1	0	4	12	0	0	0	4	0	3	0
2	6	0	0	12	0	4	0	3	2	0
3	0	0	0	0	0	4	4	0	2	2
4	0	0	0	0	0	4	4	0	3	3
5	0	5	0	0	12	0	0	0	2	4
6	6	3	0	0	0	0	0	9	0	3
1	0	4	6	0	0	0	0	4	3	0
2	6	0	6	7	2	0	0	0	0	0
3	0	0	0	5	0	12	4	0	2	0
4	6	0	0	0	4	0	0	4	4	0
5	0	4	0	0	3	0	8	0	0	12
6	0	4	0	0	3	0	0	4	3	0
1	0	8	4	6	6	0	0	0	0	0
2	0	4	0	0	0	12	0	7	3	0
3	3	0	4	3	0	0	0	0	0	8
4	0	0	4	0	6	0	9	0	1	0
5	5	0	0	3	0	0	3	0	4	0
6	4	0	0	0	0	0	0	5	4	4
1	4	3	0	0	0	0	0	0	5	3
2	4	0	0	0	0	1	12	6	0	0
3	4	6	0	0	0	4	0	0	0	5
4	0	3	0	6	6	0	0	0	4	0
5	0	0	12	0	6	4	0	0	3	0
6	0	0	0	6	0	3	0	6	0	4
1	3	4	0	0	0	0	0	4	0	2
2	2	0	0	4	6	7	0	0	0	0
3	0	0	7	5	6	5	0	0	0	0
4	4	5	0	0	0	0	0	2	0	2
5	3	0	5	0	0	0	12	0	12	4
6	0	3	0	3	0	0	0	6	0	4

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.3 (Continued)

	Period									
Product	1	2	3	4	5	6	7	8	9	10
1	0	12	0	0	2	0	3	5	0	0
2	0	0	3	0	0	4	2	0	0	2
3	0	0	5	0	3	4	0	0	0	4
4	0	0	2	0	2	0	1	0	0	4
5	12	0	2	12	5	0	2	0	0	2
6	0	0	0	0	0	4	4	7	12	0
1	3	0	0	0	0	0	5	4	0	4
2	0	2	4	0	4	12	0	0	0	0
3	0	10	2	0	0	0	0	5	5	0
4	3	0	0	12	0	0	4	0	4	8
5	3	0	0	0	4	0	0	3	3	0
6	3	0	6	0	4	0	3	0	0	0
1	0	0	2	0	0	6	1	0	0	5
2	0	0	2	0	8	0	3	0	7	0
3	0	0	0	12	0	0	1	5	5	0
4	12	0	4	0	4	0	4	0	0	0
5	0	12	0	0	0	0	1	3	0	7
6	0	0	4	0	0	6	2	4	0	0

Table B.4 Demand data set 4: Small problem size, lumpy demand, high utilization

	Period									
Product	1	2	3	4	5	6	7	8	9	10
1	8	7	0	0	0	3	0	0	0	6
2	6	0	0	6	0	4	0	0	0	3
3	0	0	0	0	0	5	11	18	0	6
4	4	11	0	0	0	4	0	0	0	3
5	0	0	0	5	7	2	0	0	18	0
6	0	0	18	7	11	0	7	0	0	0
1	0	0	6	8	0	0	5	4	0	0
2	9	0	0	0	7	0	0	2	0	5
3	0	10	12	0	11	0	0	4	0	0
4	9	0	0	0	0	0	6	0	4	7
5	0	0	0	10	0	0	7	8	5	0
6	0	8	0	0	0	18	0	0	9	6
1	9	3	0	0	0	0	0	0	5	9
2	0	8	13	0	3	0	0	0	6	0
3	0	4	5	0	4	0	0	0	7	0
4	0	0	0	18	5	8	0	10	0	0
5	9	0	0	0	6	0	12	8	0	0
6	0	3	0	0	0	10	6	0	0	9

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.4 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10
1	7	0	0	0	0	3	0	0	5	7
2	0	9	18	0	0	0	0	5	0	5
3	8	0	0	0	8	0	18	0	6	0
4	0	9	0	0	0	4	0	3	3	0
5	0	0	0	0	10	6	0	10	0	6
6	3	0	0	18	0	5	0	0	4	0
1	0	0	0	0	18	0	7	5	0	3
2	9	18	0	0	0	0	0	3	3	3
3	0	0	6	0	0	0	7	4	0	7
4	0	0	6	0	0	8	4	6	0	0
5	0	0	0	12	0	10	0	0	8	5
6	9	0	6	6	0	0	0	0	7	0
1	6	3	0	8	0	0	0	0	0	5
2	7	3	0	0	0	10	0	8	0	0
3	0	6	4	0	4	0	0	0	0	3
4	0	0	8	0	5	0	0	10	0	4
5	0	6	6	0	5	0	0	0	18	6
6	5	0	0	10	4	8	18	0	0	0
1	0	0	0	0	7	6	5	0	9	0
2	0	2	0	4	0	0	0	18	9	0
3	0	7	13	5	0	0	7	0	0	0
4	0	6	0	0	4	6	0	0	0	13
5	0	0	0	0	7	6	6	0	0	5
6	18	3	5	9	0	0	0	0	0	0
1	6	0	0	0	18	3	0	0	0	5
2	0	6	18	0	0	5	7	0	0	0
3	3	5	0	6	0	0	0	0	0	4
4	0	7	0	5	0	3	0	0	0	6
5	2	0	0	0	0	7	6	18	0	3
6	7	0	0	7	0	0	5	0	18	0
1	0	4	6	4	0	0	18	0	0	0
2	12	3	0	0	0	0	0	7	6	0
3	0	5	0	4	18	9	0	0	0	0
4	0	6	0	3	0	0	0	6	4	0
5	0	0	0	3	0	9	0	0	8	18
6	6	0	12	4	0	0	0	5	0	0
1	9	4	0	7	0	0	3	0	0	0
2	0	0	6	0	8	0	7	0	18	0
3	0	0	8	5	10	0	8	0	0	0
4	9	7	0	2	0	0	0	6	0	0
5	0	3	4	0	0	18	0	0	0	12
6	0	4	0	4	0	0	0	12	0	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.5 Demand data set 5: Large problem size, smooth demand, low utilization

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	1	0	2	0	1	1	0	1	2	2	1	2	1	2	0	1	1	1	1	2	2	1	0	1	0	2	1	1	1
2	1	0	2	1	0	0	1	0	0	0	2	1	0	0	0	1	0	1	0	2	0	0	0	0	0	0	0	0	1	0
3	0	0	0	0	0	0	0	2	1	0	0	0	1	2	1	1	1	2	2	0	0	1	0	1	0	1	2	0	1	0
4	1	1	0	0	1	2	0	1	0	0	0	0	2	0	1	2	1	2	1	1	0	1	1	0	1	1	1	1	1	1
5	1	0	2	0	2	0	1	1	1	1	2	0	0	1	1	1	1	0	1	1	1	0	0	0	0	1	0	2	0	0
6	1	1	0	1	1	1	2	0	1	0	0	1	2	2	0	0	1	0	0	0	2	0	1	1	2	2	0	2	2	0
7	0	2	1	2	0	2	1	1	1	2	2	2	1	2	1	1	2	1	0	1	0	1	1	1	2	1	2	0	0	1
8	1	2	2	2	0	0	2	0	2	1	2	1	0	2	0	0	2	0	1	0	2	2	0	2	1	0	2	0	0	2
9	1	0	1	0	2	2	0	0	1	1	0	2	0	0	2	0	0	2	2	0	1	1	1	1	1	0	0	1	0	2
10	0	2	1	2	0	0	0	2	1	1	0	0	1	1	1	1	0	0	0	1	1	0	2	1	1	1	1	2	0	2
11	1	0	2	0	1	2	0	1	1	1	2	1	1	0	1	1	0	0	1	1	1	1	1	0	0	0	1	0	1	0
12	2	0	0	0	2	0	1	2	0	1	0	1	1	0	1	1	0	1	2	1	0	0	1	0	0	2	0	1	1	2
13	1	2	0	0	0	0	1	2	0	0	0	1	0	0	0	0	0	1	0	1	0	0	1	1	0	2	0	0	0	0
14	0	1	0	1	2	1	1	0	0	1	0	1	0	0	0	1	2	0	1	1	1	1	0	2	2	1	0	0	2	0
15	1	0	1	1	1	1	1	0	2	1	0	0	1	1	1	2	1	1	0	1	1	2	2	2	1	0	1	2	2	1
1	1	0	0	1	0	1	1	0	1	2	0	1	0	0	2	1	1	0	0	2	2	0	1	1	0	0	1	0	0	0
2	1	1	0	1	2	0	1	1	0	2	1	2	0	0	0	2	0	1	1	0	0	0	1	0	1	2	0	1	0	0
3	1	0	1	0	1	1	0	1	0	0	1	0	0	2	0	0	1	0	2	0	1	2	0	1	2	1	0	0	2	1
4	0	1	0	2	2	0	2	1	2	1	1	0	2	2	2	0	2	1	0	0	0	2	2	1	0	0	1	2	1	0
5	2	0	1	0	1	1	2	1	0	0	0	0	1	0	1	0	1	0	1	2	0	0	0	2	0	1	0	0	2	0
6	2	0	0	1	0	2	1	0	1	1	2	0	0	0	0	0	0	1	1	2	0	0	1	1	1	0	1	0	0	2
7	0	1	1	0	0	1	1	1	1	0	2	0	1	1	1	1	0	0	2	0	2	1	0	0	1	1	0	1	0	1
8	0	2	1	2	1	0	1	1	0	0	1	2	0	0	1	2	1	1	2	0	2	2	1	1	0	1	0	0	0	2
9	0	2	2	1	0	0	1	1	2	1	0	0	1	0	0	1	1	1	1	2	1	0	2	1	2	0	1	1	2	0
10	1	1	2	1	0	2	1	1	0	0	0	1	2	1	1	2	0	1	1	0	1	1	0	2	1	2	2	2	1	2
11	0	0	0	1	0	1	0	1	1	0	1	2	1	0	0	0	1	0	0	1	1	1	0	0	1	0	0	2	0	1
12	0	1	0	0	0	1	0	1	1	0	2	0	0	1	0	0	1	1	1	2	0	0	2	0	1	0	1	2	1	0
13	1	2	2	2	2	1	0	1	2	2	1	2	2	1	1	1	1	1	0	0	2	1	0	0	0	2	1	0	1	0
14	1	1	1	0	2	1	1	1	0	1	0	1	0	2	2	2	1	2	0	0	0	1	1	0	2	1	2	0	1	2
15	2	0	1	0	1	0	0	0	1	2	0	1	2	2	1	0	1	2	0	1	0	1	1	2	0	1	2	1	1	1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.5 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	2	2	2	0	1	1	1	2	0	2	0	0	2	2	2	2	2	1	1	1	0	2	1	2	1	2	0	0	1
2	1	1	1	1	1	1	1	2	2	2	1	1	2	0	0	0	1	0	1	0	1	2	1	1	1	0	1	0	0	0
3	0	1	2	1	0	1	2	2	1	0	0	0	2	1	1	0	0	1	2	2	0	0	2	0	0	2	1	2	0	0
4	1	1	1	0	0	1	1	0	1	1	2	0	0	0	0	0	0	1	0	1	0	2	1	1	1	0	0	1	1	0
5	1	1	0	0	0	0	1	0	0	1	1	0	0	2	1	1	1	0	1	1	1	0	1	1	2	1	0	0	1	1
6	1	1	0	0	2	1	1	1	1	1	2	0	0	0	1	0	1	2	0	1	0	0	1	0	1	1	0	1	1	0
7	1	0	2	2	2	1	2	0	0	0	1	1	1	0	1	2	1	0	0	0	2	1	1	0	1	1	2	0	0	1
8	1	0	1	0	0	1	0	2	0	2	1	1	1	0	0	2	2	0	1	1	1	2	0	1	0	1	2	0	1	1
9	1	2	1	0	0	1	0	1	2	1	0	2	2	1	1	1	0	0	1	0	0	0	0	0	2	2	0	2	1	0
10	0	0	0	1	0	0	0	0	0	1	0	2	1	1	0	1	1	1	2	1	1	1	2	1	1	1	0	1	2	1
11	1	1	1	1	1	2	2	2	1	0	0	2	0	1	2	0	0	0	1	2	0	0	0	0	0	2	1	0	2	2
12	1	0	0	0	2	0	0	0	1	0	0	0	0	2	0	1	1	0	0	0	1	0	1	0	1	0	1	0	0	0
13	1	1	0	2	2	2	0	0	0	2	1	0	1	1	1	1	1	1	1	0	1	2	0	2	0	0	1	1	1	2
14	1	0	0	1	2	0	1	1	1	1	0	1	1	0	2	1	0	2	1	1	2	0	0	2	0	0	0	2	1	2
15	1	1	1	1	0	0	0	0	0	0	1	2	1	1	0	0	1	2	0	1	1	2	0	2	0	0	1	2	1	1
1	0	1	0	0	0	0	2	0	1	0	1	1	0	2	1	0	1	1	0	0	2	0	1	0	0	1	1	0	2	0
2	1	2	1	2	1	2	1	0	0	1	0	1	1	1	0	1	0	0	1	2	0	1	0	1	2	1	1	1	0	0
3	0	1	1	0	0	1	0	1	0	0	0	0	0	2	1	1	2	2	2	1	0	2	0	2	0	1	0	0	0	0
4	2	2	1	0	0	0	2	1	2	0	0	2	1	0	0	1	2	0	0	1	2	0	2	2	0	0	0	2	1	1
5	0	2	2	0	0	1	1	2	1	0	0	2	1	0	1	1	1	2	0	1	0	2	0	0	2	2	1	1	0	1
6	0	0	1	2	0	1	0	2	0	1	0	0	2	0	1	1	1	0	0	0	1	0	1	2	1	0	1	0	0	1
7	1	1	2	2	1	2	0	0	0	1	2	0	1	0	0	0	0	2	1	1	0	0	1	0	1	0	0	0	1	2
8	1	0	1	1	1	0	1	0	1	2	1	0	0	2	2	1	0	0	2	1	0	1	1	0	1	1	2	1	0	0
9	1	0	1	2	2	2	1	1	1	1	1	1	0	1	1	2	0	1	0	2	1	0	1	1	0	1	0	1	2	1
10	0	0	0	0	2	1	0	0	1	2	1	1	1	1	0	0	0	0	1	0	1	0	2	2	0	2	0	1	0	2
11	1	2	1	0	2	1	1	0	1	1	2	2	0	1	0	1	2	0	1	0	1	1	0	0	1	0	2	0	1	1
12	1	0	0	1	0	0	1	1	0	0	2	0	1	0	2	1	2	2	1	1	1	2	1	1	1	2	0	2	1	1
13	1	0	0	1	1	1	1	1	1	1	2	1	0	2	1	1	1	1	2	0	1	0	0	0	2	0	2	1	2	1
14	2	0	0	0	0	0	1	2	1	1	0	1	2	0	0	0	0	1	1	1	2	2	1	1	0	0	2	2	1	0
15	1	1	1	1	2	0	0	1	2	1	0	0	2	0	2	1	0	0	0	1	0	1	1	0	1	1	0	0	1	1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.5 (Continued)

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	1	2	1	1	0	2	1	1	0	0	0	0	0	0	0	1	2	1	0	0	1	1	2	1	1	0	0	2	2
2	1	0	0	2	1	1	1	0	1	1	2	1	2	0	2	2	1	0	0	0	1	2	2	1	0	1	0	0	1	1
3	2	0	1	0	1	0	0	1	1	0	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	2	0	1
4	0	0	0	0	1	1	0	0	1	1	1	0	0	2	0	1	0	0	2	1	2	0	1	2	0	0	0	0	1	0
5	2	2	0	0	0	1	1	1	1	0	1	1	0	0	1	0	0	2	1	0	0	1	0	1	0	0	2	2	2	0
6	0	0	0	1	1	2	1	0	1	1	0	1	2	1	2	1	2	0	1	1	0	1	0	0	1	0	0	1	1	2
7	0	0	2	1	1	1	0	0	0	1	1	1	2	0	0	1	2	1	1	1	1	2	0	1	2	1	2	1	0	1
8	1	1	0	1	0	2	2	2	1	2	1	1	0	1	1	0	1	2	0	0	1	1	1	2	0	1	1	2	0	1
9	1	0	1	0	2	0	2	1	0	1	0	2	0	2	2	1	1	1	1	1	1	1	1	0	0	0	0	0	0	1
10	1	1	0	1	2	1	2	0	1	2	1	1	1	2	1	0	0	0	1	0	2	0	2	0	0	1	1	0	1	0
11	1	1	1	1	0	0	0	1	1	0	1	0	0	1	1	1	0	0	0	0	1	1	1	1	2	0	1	2	1	0
12	0	2	2	1	0	2	0	1	0	1	1	0	0	0	0	2	0	0	1	2	0	0	0	0	0	0	2	1	1	0
13	1	2	1	2	2	0	1	2	1	1	1	0	2	0	0	0	1	1	2	1	0	0	0	1	2	2	2	0	1	0
14	1	1	1	1	0	1	0	1	0	0	1	1	2	1	1	1	1	0	0	2	1	1	1	0	1	2	1	1	0	2
15	0	1	1	0	0	0	0	1	2	1	1	2	1	1	1	1	2	2	1	2	1	0	1	0	2	2	0	0	1	1
1	1	0	1	0	1	1	0	1	1	1	0	1	1	2	1	1	1	0	1	0	0	0	0	0	1	0	2	0	0	0
2	0	1	1	0	1	0	0	1	1	2	2	1	1	0	2	0	1	0	1	2	0	2	0	2	0	1	0	1	0	0
3	1	1	2	1	0	2	1	0	0	0	0	1	1	0	1	1	1	2	2	1	0	1	1	0	0	0	2	2	1	1
4	2	0	1	1	0	1	2	1	1	0	1	1	0	2	1	1	1	1	0	2	2	1	0	0	0	0	0	0	1	0
5	0	0	0	0	1	2	0	1	1	0	0	0	0	1	0	0	0	0	0	0	2	0	0	1	2	0	0	0	2	2
6	1	0	1	1	0	0	0	1	0	0	0	0	1	2	0	2	2	1	2	0	0	1	2	0	0	1	1	1	2	1
7	0	2	0	1	0	0	1	0	1	2	0	1	0	0	0	2	1	1	0	1	0	1	2	2	0	0	1	1	0	1
8	1	0	0	0	1	2	0	1	1	2	2	1	2	2	2	0	1	2	1	2	2	0	2	2	1	0	1	2	1	0
9	0	2	1	1	1	0	2	2	0	1	1	0	1	0	2	0	1	1	0	1	0	0	0	0	1	2	0	0	0	1
10	1	1	0	1	2	1	0	0	2	0	2	1	1	2	0	0	2	1	0	0	2	2	1	1	1	2	1	2	2	1
11	2	1	1	2	2	1	2	2	0	1	0	1	1	0	0	0	0	0	1	1	1	0	1	1	2	0	0	0	0	0
12	0	1	2	1	1	0	1	1	1	1	0	2	2	0	2	1	1	0	0	1	1	2	0	0	2	2	1	1	0	1
13	1	1	1	2	1	0	1	0	1	0	2	0	0	1	0	0	0	1	2	0	1	1	1	1	2	2	1	0	1	2
14	0	1	0	0	1	1	1	0	0	2	2	0	0	0	1	2	0	2	1	0	0	0	1	1	0	1	1	2	1	1
15	2	1	1	1	0	1	1	1	2	0	0	2	1	0	0	2	0	0	1	1	1	1	1	1	0	1	1	0	1	1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.5 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	0	0	0	1	0	0	0	2	2	1	1	2	1	0	1	0	0	1	1	0	0	0	1	0	1	2	2	1	0
2	1	0	2	0	2	0	0	1	1	0	1	2	1	0	1	1	0	1	0	0	2	1	2	0	1	2	1	1	1	0
3	2	1	1	1	1	2	0	0	1	1	0	0	1	1	2	0	1	1	2	0	0	1	2	0	2	1	0	1	1	1
4	2	1	0	2	1	1	1	2	1	1	1	0	1	1	1	0	2	1	1	0	0	0	2	1	1	1	1	0	0	2
5	2	1	2	1	0	1	1	1	0	1	0	0	1	1	0	1	1	0	0	2	2	1	0	0	0	0	1	1	0	0
6	2	0	1	2	1	2	2	1	1	2	1	1	1	0	1	2	2	1	0	0	1	0	0	1	0	0	0	2	1	0
7	2	0	0	0	1	1	0	1	0	0	1	1	2	2	1	2	1	0	1	1	0	2	1	1	2	0	0	1	1	0
8	0	2	2	1	0	1	2	1	2	2	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	1
9	0	2	0	0	1	0	0	0	0	0	0	1	0	0	1	1	1	0	1	0	2	0	1	2	1	1	1	1	1	1
10	1	0	0	0	2	0	0	1	1	0	0	1	0	2	1	1	1	0	0	1	0	2	1	2	0	1	0	1	1	1
11	0	0	1	0	0	1	2	0	1	1	2	2	1	2	1	0	2	2	1	1	1	2	1	1	0	0	0	0	2	1
12	0	1	0	0	1	1	0	0	1	1	0	1	2	1	2	1	0	2	1	2	1	0	0	0	1	2	1	0	0	1
13	0	2	1	2	0	0	2	0	1	0	1	0	0	0	0	0	0	2	0	2	1	1	2	0	1	2	1	2	2	1
14	0	0	0	2	0	1	0	2	0	0	2	0	0	1	0	1	0	1	2	1	0	2	0	0	2	1	0	0	0	2
15	0	2	2	1	1	1	2	2	0	1	2	1	0	0	1	1	1	0	2	1	1	0	0	2	1	0	2	0	1	1
1	0	0	0	2	2	0	2	2	0	0	2	2	1	1	1	0	0	1	1	1	1	0	1	0	0	0	1	1	0	1
2	0	0	1	1	1	1	0	0	1	0	0	1	1	1	1	2	0	2	0	0	0	0	1	1	2	1	1	1	1	1
3	0	2	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1	1	1	1	0	2	0	0	2	0	2	1	0	2
4	2	1	1	2	1	1	1	1	0	1	0	0	1	2	2	1	1	0	0	1	0	1	1	1	0	1	0	1	1	1
5	1	1	0	1	0	1	1	0	1	1	2	1	0	2	1	1	0	0	2	0	0	0	0	1	1	1	0	0	2	1
6	1	1	1	0	1	1	0	0	0	1	1	0	1	1	0	1	1	0	0	2	1	1	1	1	1	0	0	2	0	1
7	1	0	1	0	0	1	0	1	0	2	0	0	1	0	1	0	2	2	0	0	1	1	1	0	1	1	0	0	1	1
8	1	2	2	0	1	1	2	1	0	0	1	2	0	2	1	0	0	0	1	2	1	0	1	0	0	2	2	2	0	0
9	0	1	0	2	1	1	1	1	0	2	0	2	1	1	0	1	1	1	0	0	2	0	0	1	0	0	2	0	1	0
10	1	0	1	0	2	0	2	1	1	0	0	1	0	1	1	1	0	1	1	2	1	1	1	2	1	0	1	2	0	0
11	0	2	0	1	0	1	0	1	2	0	2	1	2	0	1	0	0	0	0	0	2	0	0	0	1	0	1	1	0	1
12	1	0	2	0	0	2	1	1	1	1	2	0	0	0	1	1	1	1	2	1	1	2	1	2	1	1	1	1	2	1
13	2	0	1	1	0	0	0	1	2	2	0	1	0	0	1	2	2	2	1	1	1	2	2	0	1	1	1	0	1	1
14	0	1	1	1	2	1	1	1	2	2	2	1	1	0	0	1	2	1	1	0	1	2	2	1	1	2	0	0	1	1
15	2	1	1	1	1	1	0	1	1	0	0	0	2	1	1	1	1	0	2	1	0	0	0	2	0	2	0	0	2	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.5 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	2	0	2	2	1	0	1	1	1	1	0	0	0	0	1	2	0	0	1	0	0	0	0	1	0	2	1	0	0
2	1	0	1	0	0	1	0	1	0	0	1	0	0	0	1	1	0	1	0	1	1	0	1	1	1	1	2	1	2	0
3	2	0	0	0	2	0	1	1	1	1	1	0	1	2	1	0	1	0	2	1	0	0	0	1	0	2	1	0	1	0
4	1	1	0	0	0	1	1	0	0	1	2	1	2	1	1	2	0	2	1	1	2	2	1	1	0	2	1	2	0	2
5	0	0	2	0	1	0	2	0	2	0	1	1	2	2	0	1	1	0	1	1	1	1	1	0	2	0	1	2	0	2
6	0	0	2	2	1	1	0	1	0	1	2	1	1	2	1	1	0	2	1	0	0	1	0	0	0	0	0	2	1	0
7	1	1	1	0	0	1	1	0	0	0	1	1	1	0	2	0	1	2	0	1	1	2	2	1	1	2	1	0	2	2
8	1	0	0	0	1	1	0	2	1	2	0	2	1	0	0	1	2	0	2	1	2	0	0	2	2	1	1	0	0	0
9	1	1	0	2	0	0	2	1	0	0	0	0	2	0	1	2	0	1	2	0	0	1	1	2	0	0	1	1	1	2
10	1	2	0	0	1	2	0	1	0	1	0	1	1	1	0	2	2	1	1	2	0	2	1	1	2	1	0	0	0	2
11	0	0	2	0	0	0	1	2	2	1	0	2	1	1	1	0	1	1	1	2	2	1	0	1	0	0	0	1	0	1
12	2	1	1	2	1	0	1	1	2	1	0	1	0	1	0	0	0	1	0	1	0	0	1	1	2	0	0	0	1	0
13	1	2	0	1	0	1	0	1	2	2	2	1	0	0	1	0	0	0	0	0	1	0	1	0	0	1	1	0	2	0
14	0	2	1	1	1	2	2	0	0	0	0	1	0	1	2	1	1	1	0	0	0	1	1	1	0	2	1	1	0	1
15	0	0	2	2	2	1	1	0	1	1	1	0	0	1	1	0	1	0	1	0	2	1	2	0	1	0	0	1	2	0
1	2	2	2	1	2	1	0	0	0	1	1	2	0	1	1	0	2	1	1	0	2	0	0	1	0	2	2	2	1	1
2	1	0	1	1	1	1	1	1	2	0	0	0	1	2	1	0	1	0	1	0	0	1	1	2	1	1	1	1	2	0
3	0	0	0	2	0	1	0	2	1	2	0	1	1	0	1	2	0	0	1	0	0	1	0	1	0	2	2	1	1	1
4	0	0	0	0	1	0	2	2	2	0	2	2	1	1	2	0	1	0	0	1	2	1	2	0	2	0	0	1	0	0
5	0	2	1	0	1	1	0	2	0	1	2	1	1	2	1	0	1	0	0	2	0	2	0	0	1	0	0	0	0	1
6	1	1	1	2	0	1	0	0	0	2	1	0	0	0	0	1	0	1	1	1	0	1	0	0	1	0	2	1	0	1
7	1	1	0	0	1	1	1	0	1	2	1	1	1	0	0	0	0	1	1	2	1	1	1	0	1	0	0	1	0	1
8	1	2	0	0	1	0	1	2	1	1	0	1	2	1	1	2	1	2	2	1	1	2	1	0	1	2	1	0	2	0
9	2	2	1	0	1	0	2	0	1	0	1	0	1	0	0	1	1	2	0	1	0	0	0	2	0	1	1	0	0	0
10	0	0	1	2	0	1	1	0	0	0	0	0	1	0	1	1	2	0	1	0	0	1	1	1	0	1	0	2	1	1
11	0	0	1	1	1	1	0	0	0	1	1	1	0	0	0	2	0	2	1	1	0	0	2	0	1	0	1	1	2	1
12	2	0	1	1	0	2	0	0	2	1	0	0	2	1	1	1	2	0	1	1	1	0	1	0	2	0	1	0	0	0
13	1	0	0	0	2	0	2	1	1	0	2	2	1	0	0	0	0	2	2	0	2	1	1	2	0	0	0	0	1	2
14	0	1	1	1	1	1	0	0	1	0	0	1	0	2	1	1	1	1	0	1	2	1	1	1	0	1	0	1	1	1
15	1	1	2	1	0	1	2	2	0	1	1	0	0	2	2	1	0	0	0	1	1	0	1	2	2	2	1	1	1	2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.6 Demand data set 6: Large problem size, smooth demand, high utilization

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	0	1	1	1	2	2	2	1	0	2	1	0	0	1	0	2	1	2	1	2	1	0	1	1	1	1	1	2	2
2	1	0	2	0	2	1	2	2	2	2	2	0	2	1	0	1	1	2	1	1	1	2	1	1	2	1	2	0	2	2
3	1	1	1	1	1	2	0	1	1	1	2	2	0	2	1	0	1	0	1	1	1	1	1	2	1	1	1	2	1	0
4	1	1	1	0	1	0	1	1	2	1	0	2	0	2	2	1	1	2	2	2	1	1	0	2	1	1	1	1	0	1
5	1	0	1	2	2	0	2	1	0	2	1	1	2	2	1	1	2	1	1	1	1	0	1	2	0	2	1	1	1	1
6	1	1	1	2	1	1	1	2	2	2	1	1	1	1	2	1	0	2	0	1	1	1	1	2	2	2	1	1	2	2
7	1	2	1	1	1	1	1	0	1	0	1	0	2	0	1	1	1	0	0	1	2	2	2	1	1	1	1	2	1	1
8	1	1	0	2	0	1	1	2	1	1	2	1	2	2	2	1	0	1	1	1	2	2	1	1	2	2	1	2	1	2
9	2	1	1	1	2	1	1	0	1	0	0	2	2	1	2	1	2	1	2	1	1	0	2	0	1	1	2	1	1	1
10	2	2	1	2	0	2	1	2	2	2	2	1	1	2	1	2	2	2	2	1	1	2	2	1	2	1	0	1	2	1
11	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2	1	2	1	2	2	0	1	1	2	1	0	1
12	0	2	2	1	1	1	1	1	1	1	2	1	2	1	1	1	2	1	2	1	1	0	2	2	2	1	1	2	1	1
13	2	2	1	1	2	2	2	1	1	2	2	2	1	1	2	2	2	2	1	1	0	1	1	1	1	1	1	2	1	2
14	1	1	2	0	1	0	1	1	0	1	0	1	2	1	0	2	0	0	1	1	1	1	1	1	0	1	2	0	1	1
15	2	2	1	2	1	2	1	1	2	2	0	2	0	1	1	2	0	1	1	2	2	2	1	1	1	1	1	1	2	0
1	2	1	2	1	1	1	2	2	1	1	1	1	2	1	2	1	2	2	1	2	2	1	1	0	2	1	2	1	0	0
2	1	1	1	1	2	1	2	1	0	2	2	2	1	1	2	1	1	2	2	2	2	1	0	1	1	0	1	0	2	1
3	1	0	1	1	2	1	0	1	1	2	1	2	1	1	1	2	0	1	1	1	2	0	1	1	0	1	1	1	2	2
4	1	1	1	1	1	1	1	2	2	0	1	0	2	2	0	2	0	2	1	0	2	2	2	2	2	2	0	2	1	0
5	1	2	0	1	2	1	2	0	2	0	2	1	0	2	1	1	2	0	1	1	1	2	2	1	2	1	0	1	2	2
6	1	2	1	2	1	1	2	2	2	1	1	0	1	1	2	1	1	1	1	1	0	0	1	1	1	2	2	0	2	0
7	2	1	2	2	2	1	2	1	2	2	1	1	2	1	1	1	2	1	1	1	1	0	1	2	1	2	0	1	2	2
8	1	1	0	2	1	2	1	1	0	1	0	1	0	0	1	2	2	1	1	1	1	2	1	2	1	0	2	2	1	1
9	0	1	2	1	1	1	1	1	2	1	2	2	1	1	2	1	1	1	2	2	0	2	2	2	1	2	1	2	0	0
10	2	1	2	1	1	1	0	1	1	1	0	1	2	2	2	2	2	1	1	2	2	1	1	1	1	0	1	2	1	1
11	2	2	2	1	2	1	1	2	0	1	2	2	1	1	2	1	1	2	1	2	1	2	2	1	2	2	1	0	0	2
12	2	2	1	0	1	2	1	2	1	1	2	2	2	1	1	0	1	1	2	0	0	2	1	1	1	1	2	2	1	2
13	1	1	1	1	0	1	0	1	1	2	2	2	2	1	0	1	1	1	1	1	2	1	2	1	1	1	2	1	1	2
14	1	1	1	1	1	2	2	0	2	1	0	0	1	1	0	1	1	2	1	2	0	1	0	2	2	1	2	2	2	2
15	0	1	1	2	0	1	1	1	1	2	1	1	0	2	1	1	1	0	1	0	2	1	1	0	0	2	1	1	1	1

Table B.6 (Continued)

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	1	2	1	1	2	1	1	2	1	1	1	1	2	1	1	2	1	1	0	1	2	0	2	1	1	0	2	0	2
2	0	1	1	1	1	1	0	2	1	1	1	1	1	1	1	0	0	1	1	2	1	1	1	1	1	2	1	2	1	2
3	2	1	1	2	1	1	1	1	1	1	1	1	1	2	1	1	2	2	0	0	0	1	1	1	2	1	2	1	2	2
4	2	1	2	1	2	2	1	0	2	2	1	2	1	0	1	1	2	0	1	1	1	2	1	1	2	2	2	1	2	0
5	2	1	2	1	0	0	2	1	1	0	0	2	2	1	1	1	2	1	1	2	1	1	2	1	2	1	1	0	2	1
6	1	0	1	2	2	1	2	1	1	0	1	1	2	1	0	1	1	1	2	1	2	0	1	2	0	1	1	2	1	2
7	1	1	2	1	0	2	1	0	1	2	1	1	1	2	1	1	1	2	0	1	2	1	1	2	2	0	2	2	1	1
8	1	2	0	1	1	1	1	2	2	0	2	1	2	1	1	2	1	1	1	1	2	1	2	2	1	2	1	2	1	1
9	1	0	1	0	2	1	2	1	1	1	2	2	2	2	2	2	1	2	2	1	2	1	1	0	1	2	2	1	1	2
10	2	2	1	1	0	0	1	0	1	2	2	2	1	1	2	1	1	2	1	1	1	1	1	1	1	1	1	1	1	0
11	1	1	2	1	2	2	0	2	1	1	2	2	0	2	2	0	1	0	1	2	1	2	1	0	2	1	1	0	1	0
12	1	2	1	1	1	2	1	2	1	2	1	1	0	0	0	2	1	2	2	2	0	2	2	1	0	1	1	2	1	1
13	0	1	1	2	1	1	2	1	1	2	2	0	1	2	2	2	0	1	1	2	2	1	2	2	1	2	1	0	1	2
14	1	2	0	1	2	0	1	2	1	2	0	0	1	0	1	1	1	0	2	1	1	2	1	1	1	1	1	1	1	0
15	2	2	1	2	2	2	2	2	1	1	1	1	2	1	2	2	2	2	2	1	1	0	1	1	1	0	1	1	2	2
1	1	1	2	0	1	2	2	2	1	2	1	1	2	2	0	1	1	2	2	2	2	1	2	1	1	2	1	1	2	0
2	0	0	1	2	1	1	2	0	2	1	1	0	0	1	1	1	0	2	2	2	1	1	1	0	0	0	2	1	1	2
3	2	2	2	1	1	2	1	1	0	1	2	2	0	1	2	2	1	2	1	2	1	1	0	2	1	2	1	1	2	1
4	1	1	1	1	2	1	1	0	2	2	1	1	2	1	1	2	2	0	1	0	0	0	2	1	1	2	2	2	2	2
5	2	1	2	0	1	2	1	1	1	2	1	0	1	1	2	1	0	1	2	0	2	0	1	1	1	0	2	2	1	2
6	1	2	2	2	2	2	1	1	2	1	2	2	1	2	0	1	1	2	1	1	0	2	1	2	1	1	1	1	1	1
7	2	2	1	1	0	1	1	2	0	0	0	2	1	1	1	1	1	2	2	1	2	1	1	1	1	2	2	2	1	1
8	1	2	1	1	1	2	0	2	1	2	2	2	0	0	2	2	2	1	2	1	1	2	2	1	2	1	1	1	1	2
9	2	0	1	2	2	0	1	2	1	1	2	1	1	1	2	1	0	1	1	1	1	2	1	2	1	2	2	2	1	1
10	1	0	1	1	2	1	0	2	0	1	1	0	0	0	2	2	1	1	1	1	2	1	1	1	1	0	0	1	0	1
11	1	1	0	1	0	0	2	1	1	2	2	1	2	2	1	0	2	2	1	1	1	0	1	1	2	2	1	1	2	0
12	1	2	1	1	2	2	1	1	2	1	0	1	2	1	2	1	2	0	0	2	1	1	1	1	2	0	2	1	0	1
13	1	1	1	2	2	1	2	1	1	0	1	2	2	2	1	1	2	1	1	2	2	2	1	1	2	2	1	1	0	1
14	0	2	1	1	1	1	2	2	2	1	1	1	2	1	0	1	2	1	1	2	0	2	1	1	1	1	0	0	2	1
15	2	1	1	2	0	0	1	0	2	1	1	2	2	2	1	1	1	0	0	0	2	2	2	2	1	1	0	1	2	2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.6 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1	0	0	2	2	2	1	2	1	0	0	1	2	2	2	2	1	1	2	1	1	1	1	0	1	0	1	1	1	2
2	1	1	2	0	1	0	1	2	1	1	1	0	2	1	0	1	2	0	1	2	0	1	1	2	1	2	1	1	0	1
3	1	1	1	1	1	1	1	1	2	1	1	2	1	2	1	1	2	1	1	2	2	1	2	1	1	2	1	2	2	1
4	1	1	2	2	0	0	2	0	0	1	2	1	1	2	1	2	1	2	1	1	1	2	1	1	1	1	2	1	1	2
5	1	2	1	1	0	2	1	2	0	1	1	1	2	0	2	1	1	2	1	1	1	2	1	1	2	1	1	2	1	2
6	1	2	2	0	2	2	1	2	1	2	1	1	2	0	2	1	1	1	1	2	1	0	1	1	1	2	1	2	2	1
7	0	2	1	1	1	0	1	1	1	2	1	2	0	2	2	1	2	2	0	0	2	1	1	2	0	1	1	1	1	2
8	1	0	1	2	1	1	1	1	1	1	0	1	0	2	1	2	1	0	2	1	2	1	1	1	1	0	2	1	2	2
9	2	2	1	2	1	2	2	1	2	2	1	1	0	1	0	2	1	1	2	1	0	1	2	2	1	0	1	0	1	2
10	2	1	2	1	2	2	1	1	1	2	2	0	2	1	1	1	1	1	1	1	2	0	1	0	2	2	1	2	2	1
11	1	2	0	0	2	0	1	1	2	1	2	2	2	2	2	1	1	1	1	1	1	1	1	2	1	2	1	1	1	2
12	1	2	1	1	1	1	2	1	1	2	2	2	1	1	1	2	0	1	1	1	1	1	1	1	1	1	1	1	1	0
13	2	0	1	1	2	2	1	1	1	1	1	1	1	0	1	0	1	2	1	2	2	2	1	2	1	2	2	1	1	0
14	1	1	1	2	0	2	1	1	2	1	1	1	0	2	1	0	2	2	2	1	1	2	1	1	2	0	1	1	1	0
15	2	1	2	2	2	1	1	1	2	0	2	2	2	0	1	1	1	1	1	1	1	2	2	1	2	2	1	1	1	0
1	1	1	1	2	2	2	1	1	1	2	1	1	1	2	2	0	1	2	1	1	1	0	2	0	2	0	0	1	2	1
2	2	1	1	1	1	0	1	1	2	1	1	0	0	2	1	1	2	1	2	1	1	2	1	2	1	1	0	1	2	1
3	1	1	1	0	1	2	1	2	0	1	2	1	2	2	0	1	2	1	2	2	1	2	0	2	0	2	1	1	2	2
4	0	2	1	2	2	1	1	1	0	0	1	1	1	2	1	2	1	1	1	1	0	1	2	2	2	1	2	1	1	1
5	2	1	1	1	0	1	1	1	1	0	2	2	0	0	2	1	2	2	2	2	2	1	2	1	1	2	2	2	2	2
6	2	1	2	1	2	2	2	1	1	1	2	2	2	1	1	1	1	0	1	0	2	0	0	1	1	1	2	1	1	0
7	0	1	2	1	1	1	1	2	0	2	2	2	0	2	1	1	2	1	2	1	2	2	2	1	0	0	2	1	1	2
8	0	1	1	2	2	2	2	0	2	0	1	1	1	1	1	1	1	1	1	2	1	0	1	2	1	2	1	1	0	1
9	1	1	2	1	1	2	1	0	1	2	0	1	1	1	1	2	1	1	2	1	0	1	1	1	2	1	1	1	2	2
10	1	2	1	0	0	0	2	2	1	2	2	1	2	1	2	1	1	2	1	1	2	2	1	1	1	2	2	1	0	1
11	2	1	0	0	1	1	0	2	2	1	1	1	2	0	1	2	1	1	0	1	1	2	0	1	1	1	1	1	0	1
12	2	1	2	2	2	2	2	2	1	2	0	2	2	1	1	1	0	2	0	1	1	1	1	0	1	0	1	1	0	1
13	1	1	0	1	2	0	1	1	2	0	1	1	2	1	2	1	1	0	0	2	2	1	2	1	2	2	1	1	2	0
14	2	2	2	2	0	1	1	1	2	2	1	1	1	1	0	1	1	2	1	2	1	1	2	2	1	2	1	2	1	2
15	1	1	1	2	1	1	1	1	2	2	1	1	1	1	2	2	1	1	2	0	1	2	1	1	2	1	1	2	2	1

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.6 (Continued)

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	2	2	2	2	2	0	1	1	1	1	2	1	1	1	2	2	1	0	0	1	2	0	2	2	1	0	1	2	2	2
2	2	1	1	2	1	2	1	2	2	2	0	2	2	1	1	2	2	2	1	1	1	2	1	2	2	2	0	2	2	1
3	1	0	2	1	0	2	1	0	2	1	2	1	0	2	1	1	1	2	1	2	1	1	1	2	0	0	0	0	2	2
4	1	1	1	2	1	0	2	1	1	1	0	1	1	1	2	0	1	1	2	1	2	2	0	2	2	1	1	1	1	1
5	1	2	1	0	1	2	1	1	0	1	1	1	2	2	1	2	1	1	2	1	1	2	2	1	2	2	1	1	0	1
6	1	0	2	1	2	1	1	2	2	0	2	1	1	1	2	1	1	2	1	1	1	1	1	1	1	1	1	1	2	1
7	2	2	0	1	1	0	2	1	1	1	1	2	1	1	1	2	1	1	1	1	1	1	2	1	1	1	2	1	2	2
8	2	2	1	1	0	1	1	1	1	1	0	2	2	1	2	2	1	1	2	2	1	1	1	1	2	1	0	2	2	1
9	2	1	1	2	1	2	0	2	1	1	1	0	2	0	1	0	1	2	1	1	1	0	1	0	0	0	2	1	1	1
10	0	2	1	1	2	2	2	1	1	1	1	0	2	1	1	1	2	0	1	1	1	2	2	2	1	2	2	1	1	1
11	0	0	1	0	1	2	2	1	2	2	1	0	1	0	0	2	1	1	2	1	1	1	1	1	1	1	1	2	1	1
12	1	2	1	1	1	1	1	1	1	2	2	1	1	2	2	2	1	2	1	2	1	2	0	1	1	2	1	1	0	0
13	2	1	1	2	1	0	1	1	1	2	2	2	0	2	0	1	2	2	1	2	1	2	2	1	1	2	2	0	0	2
14	1	2	2	1	2	2	1	2	1	1	1	2	1	2	2	0	0	0	1	1	1	0	0	0	2	1	2	2	1	1
15	0	0	1	1	2	1	1	1	1	1	2	2	1	1	0	0	2	1	1	0	2	1	2	1	1	2	2	1	1	1
1	1	1	1	1	1	1	1	2	2	2	1	1	2	1	2	2	1	1	1	0	2	1	1	2	1	2	1	1	1	1
2	1	0	1	1	0	2	1	1	1	1	2	2	0	1	2	2	1	1	1	0	0	2	1	1	1	2	1	1	2	2
3	1	0	1	2	2	2	1	1	1	1	2	1	1	1	2	1	0	2	1	1	1	0	1	2	1	2	2	1	1	1
4	1	2	1	2	1	2	1	2	2	1	1	1	2	2	0	1	2	1	0	2	1	2	1	1	2	1	2	2	1	2
5	1	1	2	2	1	1	1	1	1	1	2	2	0	1	2	2	2	0	1	2	2	1	2	1	0	1	2	0	2	2
6	1	1	1	1	2	2	2	2	1	1	1	2	0	2	1	1	2	2	2	1	0	1	0	2	0	1	2	2	1	1
7	1	1	2	0	2	1	0	1	2	1	2	0	1	0	2	1	1	2	1	2	2	0	1	0	2	1	1	1	2	1
8	1	1	1	1	1	0	0	0	1	1	2	0	2	1	0	1	1	1	1	0	1	2	1	1	2	1	1	2	1	1
9	1	0	1	2	2	1	2	2	1	1	0	1	1	1	0	1	1	2	1	0	0	0	1	0	2	0	2	1	1	0
10	1	1	1	1	2	1	2	2	0	1	1	2	2	2	2	1	0	0	2	2	1	2	0	1	1	0	1	2	1	1
11	2	2	1	0	0	0	2	2	1	2	1	1	2	0	2	1	2	0	2	2	1	2	2	2	1	2	0	1	1	1
12	1	2	1	0	2	2	2	0	1	0	0	1	0	1	0	1	1	2	1	2	2	2	2	2	1	1	1	1	0	1
13	1	2	1	1	1	1	1	1	1	2	0	2	1	1	0	1	2	0	2	0	2	2	2	1	2	1	0	1	1	1
14	2	2	2	2	1	1	1	0	1	2	1	1	2	2	2	2	1	2	2	2	1	0	1	1	0	2	1	0	1	1
15	2	2	1	2	0	1	1	1	2	1	2	1	2	2	1	0	1	2	0	2	2	1	2	1	2	1	1	2	2	2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.6 (Continued)

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	1	2	2	0	1	1	1	0	1	1	2	0	1	0	1	1	0	1	1	2	0	1	1	1	2	2	1	0	1
2	0	2	1	1	1	1	1	2	2	1	0	1	0	0	2	1	1	1	1	1	1	2	1	1	1	0	1	1	2	1
3	1	2	1	1	0	1	2	1	0	2	1	1	2	2	2	2	0	1	1	2	0	0	2	2	1	2	1	1	2	1
4	1	1	1	1	1	1	1	0	1	0	1	0	0	2	1	1	2	2	1	0	1	2	1	0	2	2	0	2	1	1
5	1	2	1	2	1	2	1	1	2	1	2	1	2	0	1	1	2	1	1	2	2	1	2	1	2	2	2	1	1	0
6	2	1	1	2	1	2	1	1	1	1	2	2	1	1	2	1	2	1	2	1	1	1	1	2	2	0	1	1	0	2
7	2	0	2	0	1	1	1	1	2	0	0	1	2	0	0	2	1	2	1	1	2	1	2	1	1	1	1	2	1	2
8	2	1	2	1	2	0	1	2	0	1	1	1	1	1	1	2	0	2	1	0	1	2	1	2	2	1	2	1	2	1
9	1	1	1	1	2	1	2	1	2	1	2	2	2	2	2	1	1	1	1	1	2	2	0	1	1	1	2	2	0	2
10	1	2	2	1	0	2	1	1	1	2	2	1	2	2	2	1	1	1	1	2	1	1	0	1	1	1	2	0	1	1
11	2	1	0	1	2	2	2	1	1	2	1	1	0	2	1	0	2	1	1	1	1	1	2	0	1	2	2	1	0	1
12	2	0	1	2	2	1	1	2	1	2	1	2	2	2	1	0	2	1	2	2	1	1	2	1	1	1	0	2	2	2
13	1	1	2	1	1	0	2	1	2	1	2	1	2	1	1	1	1	1	2	1	1	1	1	2	1	0	0	1	2	0
14	1	2	1	1	2	2	1	1	1	1	1	1	1	2	2	2	0	2	2	1	1	1	2	2	0	2	2	1	2	1
15	1	1	0	1	2	1	0	2	2	2	1	1	1	0	0	2	2	1	0	2	1	2	0	1	1	1	0	1	2	2
1	0	1	1	1	0	2	0	1	1	2	1	2	0	1	1	2	1	1	1	1	0	1	2	1	2	1	2	0	1	1
2	1	0	1	1	2	0	1	1	1	1	1	1	1	2	2	1	2	2	1	2	0	1	0	2	1	0	1	2	1	2
3	2	1	1	2	1	2	1	1	2	1	2	1	2	2	2	2	1	1	1	1	1	2	2	1	0	2	2	1	1	1
4	0	2	1	1	1	1	1	2	1	0	1	2	1	1	1	1	1	0	1	1	1	2	2	1	1	2	2	1	1	2
5	2	1	1	2	1	1	2	1	2	2	2	1	1	2	2	2	1	0	1	0	2	2	2	1	2	1	0	2	1	0
6	2	2	1	0	1	1	2	1	1	1	1	0	1	0	0	0	1	1	1	2	2	1	0	1	2	1	1	1	1	2
7	1	0	2	2	1	1	0	1	1	1	2	2	2	1	1	0	1	2	1	2	1	0	1	1	1	2	0	1	2	1
8	1	1	0	2	1	0	2	2	0	2	1	2	2	1	2	2	0	1	2	2	2	1	0	2	1	1	0	2	1	0
9	2	1	2	1	2	0	1	1	2	1	0	1	2	2	0	1	2	1	1	1	0	1	2	1	2	1	1	1	1	2
10	1	1	2	1	2	2	2	0	2	0	1	2	1	1	2	1	2	2	2	2	1	1	2	1	2	1	2	1	1	2
11	1	2	0	1	1	2	1	0	2	1	1	0	0	1	2	1	0	2	1	0	1	1	0	1	1	0	2	1	2	1
12	1	2	1	2	2	2	1	2	0	1	2	1	1	1	0	1	2	0	1	1	2	1	0	1	1	2	2	0	0	0
13	2	1	1	0	1	1	1	2	1	2	1	1	1	1	1	2	2	2	2	1	2	1	1	2	1	2	2	1	1	0
14	1	2	2	2	1	2	1	1	1	2	2	1	1	1	1	0	1	1	0	2	2	2	2	1	1	0	0	2	2	2
15	1	1	2	0	1	1	2	2	1	1	0	1	2	1	1	2	1	2	2	0	1	1	2	1	0	2	1	2	2	2

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.7 Demand data set 7: Large problem size, lumpy demand, low utilization

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	1	0	0	7	0	4	0	0	4	5	0	5	0	0	0	0	0	0	0	0	3	0	1	0	0	0	3	0	0
2	0	0	0	0	0	2	0	3	0	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	3	0	0	3
3	0	0	0	1	4	1	3	0	5	4	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	1
4	0	0	0	3	0	0	0	4	0	0	0	5	0	0	0	0	0	0	0	0	0	0	9	0	6	0	1	0	0	0
5	0	0	1	0	0	6	0	0	0	2	0	0	0	0	0	0	1	0	0	0	4	0	0	0	0	0	0	1	0	0
6	7	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	4	4	1	0	0	0	0	0	0	5	0	0	0
7	0	5	2	5	0	0	0	0	1	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	4	0	4	0	0
8	0	0	0	0	0	0	0	3	0	0	0	0	0	7	0	0	1	0	0	0	1	3	0	3	0	0	0	0	0	2
9	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	2	0	5	2	0	1	5	0	0	0	2	0
10	0	0	1	2	0	0	0	0	0	0	0	0	5	0	0	3	0	1	2	0	0	0	0	0	0	0	0	4	4	0
11	0	0	0	0	0	2	1	0	0	1	0	0	0	0	3	0	0	0	0	6	0	0	0	0	0	0	0	0	3	4
12	0	6	0	1	0	0	0	0	0	0	0	6	0	5	0	3	0	0	4	0	0	0	0	4	0	0	2	0	0	0
13	0	0	1	0	0	0	0	0	2	0	0	0	0	0	7	2	0	0	0	0	2	0	3	3	0	5	0	0	0	2
14	0	0	3	0	1	1	0	2	0	0	0	0	0	0	0	0	1	0	0	5	0	4	0	0	0	0	0	0	3	0
15	5	0	0	0	0	0	4	0	0	1	5	1	0	0	2	4	0	0	0	0	0	0	0	0	1	0	1	0	0	0
1	2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	3	0	3	0	0	0	0	0	0	0	6	0	0
2	3	1	0	0	0	0	0	5	0	0	4	2	0	0	3	0	0	4	0	0	0	2	0	3	0	0	0	0	0	4
3	0	0	1	3	0	0	5	0	0	0	0	0	0	0	0	0	5	0	0	0	4	0	0	3	0	0	3	0	0	2
4	0	0	0	0	3	0	0	1	0	0	0	0	0	1	0	0	0	0	5	0	0	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	5	0	6	0	0	0	0	1	0	3	0	0	0	0	1	0	3	6	0	0	0	0	0	0
6	1	0	3	4	0	0	0	0	0	1	0	5	0	0	7	0	0	0	0	8	0	1	2	0	0	4	0	0	2	0
7	0	2	0	0	0	0	0	0	0	0	0	0	7	0	0	0	2	0	7	0	0	1	0	0	5	0	0	0	5	0
8	0	0	0	0	3	0	2	0	0	5	0	0	0	0	0	0	1	0	0	0	0	2	2	0	4	0	7	2	0	0
9	0	0	6	0	1	0	0	0	0	1	0	2	0	0	0	0	3	4	0	0	0	0	0	0	0	0	0	0	0	2
10	2	0	0	0	0	0	0	0	0	0	2	0	0	0	0	5	0	0	0	0	3	0	0	0	2	0	1	0	5	0
11	0	2	0	0	0	7	0	5	4	0	6	0	0	4	0	0	0	0	0	0	0	0	0	0	1	4	0	0	0	0
12	0	4	0	0	1	0	0	0	0	1	0	0	0	0	0	4	1	0	0	0	0	1	2	0	0	3	0	0	0	0
13	2	3	2	0	0	5	0	0	0	0	0	0	1	3	2	0	0	0	0	1	0	1	0	0	0	1	0	4	0	0
14	2	0	0	0	0	0	0	0	2	0	0	3	0	3	0	0	0	1	0	0	4	3	0	0	0	0	0	0	0	4
15	0	0	0	5	3	0	0	1	0	4	0	0	4	0	0	0	0	0	0	0	0	0	3	0	0	0	1	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.7 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	6	0	4	0	0	0	4	2	0	0	0	0	4	2	0
2	0	4	0	4	0	0	0	1	2	0	0	0	0	0	0	0	0	0	2	0	0	0	0	1	1	0	0	0	0	1
3	0	0	0	0	0	0	1	4	0	7	0	2	0	0	7	0	0	0	1	0	0	0	0	0	1	0	1	4	0	0
4	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	3	0	0	0	0	4	3	0	0	4	1	0	0	0	1
5	4	0	1	0	3	0	0	0	5	0	0	0	0	0	0	0	0	5	0	4	0	1	0	0	0	0	0	0	0	0
6	0	0	5	0	2	0	0	0	1	0	0	0	0	0	0	0	5	3	3	2	4	0	4	0	0	4	0	0	0	1
7	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	1	0	3	0	0	0	0	0	1	1	0	0	0	4
8	3	0	0	0	3	0	1	0	2	0	0	0	0	7	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0
9	0	0	0	4	0	0	2	0	0	0	0	2	3	0	2	0	0	0	3	0	0	0	0	1	3	1	4	0	0	0
10	1	5	0	3	2	0	2	0	1	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	10	0
11	0	0	6	0	0	4	0	2	0	0	3	0	0	1	0	0	0	0	0	0	0	0	2	2	0	0	1	2	0	0
12	3	2	0	0	0	0	0	0	0	0	0	5	5	4	0	0	6	0	0	0	4	3	0	4	0	3	0	0	0	0
13	1	0	0	0	0	1	0	2	1	0	0	0	0	0	0	3	0	0	0	6	0	1	0	4	0	0	0	1	0	0
14	0	0	0	0	0	0	0	0	0	0	8	1	0	0	3	0	0	0	0	0	0	0	0	0	2	2	3	0	0	0
15	0	1	0	1	2	3	6	0	0	5	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	3	0	0	5
1	6	0	9	0	0	0	0	0	0	4	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	4	0	0	5	0
2	0	4	0	0	0	0	4	6	0	0	0	0	0	3	2	0	0	0	0	0	3	0	3	0	0	0	0	0	0	0
3	0	0	0	3	0	0	4	1	0	0	6	0	2	0	0	0	0	0	0	0	0	1	0	4	0	0	0	5	0	4
4	0	0	0	0	0	0	1	0	0	0	3	2	0	0	0	0	0	0	1	0	0	0	0	0	3	0	0	0	0	5
5	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	1	6	1	3	2	0	0	0	0	0
6	0	0	0	2	0	0	0	0	3	5	0	0	0	0	0	0	0	4	0	0	1	0	4	0	0	0	0	0	0	0
7	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	3	4	5	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	5	4	2	3	0	1	1	2	0	0	0	3
9	0	0	0	0	0	0	2	2	0	3	2	0	0	5	0	0	5	0	0	0	0	2	0	0	0	0	0	0	3	0
10	0	3	0	2	6	6	0	1	0	0	0	0	0	0	0	5	3	0	0	0	0	0	0	0	0	3	0	0	0	0
11	0	0	0	2	0	1	0	0	0	0	1	4	0	2	3	0	0	0	0	2	2	0	4	0	0	0	0	0	0	0
12	0	5	0	0	3	0	1	2	2	0	0	3	2	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0
13	6	0	0	0	0	0	0	0	0	0	0	0	3	0	1	0	4	0	4	0	0	0	0	4	6	0	5	0	0	0
14	0	0	3	3	0	0	0	0	4	0	0	0	4	2	0	0	0	2	0	0	1	0	0	0	0	0	3	2	0	0
15	0	0	0	0	0	5	0	0	0	0	0	3	1	0	0	0	0	5	2	6	2	0	0	0	0	0	0	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.7 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	0	0	4	0	0	0	3	0	0	0	0	0	0	2	1	0	0	0	0	7	2	0	0	0	0	0	0	0	4
2	3	4	0	0	0	0	0	0	0	5	0	0	0	0	0	2	1	0	0	5	0	0	1	1	0	0	0	0	0	0
3	1	0	1	0	0	0	0	0	0	0	4	0	5	0	5	0	0	0	0	0	0	0	1	0	0	0	3	0	0	6
4	1	0	0	4	0	0	0	0	2	1	0	3	4	0	0	2	0	0	0	0	0	0	0	0	1	0	0	6	3	0
5	1	0	0	0	2	5	0	0	0	0	2	0	0	0	0	2	0	0	0	0	1	0	0	0	0	0	2	0	0	0
6	4	0	2	0	0	0	0	0	0	0	0	2	0	0	0	0	0	4	0	0	0	5	0	6	5	2	3	0	0	0
7	1	0	2	1	6	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	0	0	0	0	1	0	0	5	0	3	3	0	0	0	0	2	0	0	0	1	0	0	0	2	0	0	0	2
9	0	0	0	0	0	0	5	0	5	0	0	0	0	4	1	0	0	0	0	0	2	0	0	0	1	0	1	1	0	0
10	0	0	0	1	0	0	0	0	0	0	0	2	0	0	1	0	6	1	0	7	0	0	0	0	0	0	0	0	1	0
11	0	2	4	0	4	5	0	0	3	0	0	0	0	4	0	2	0	0	5	0	0	0	0	5	0	7	0	0	4	0
12	1	6	0	0	0	0	0	0	0	0	1	0	0	4	0	0	0	0	0	0	0	2	2	0	0	0	0	5	0	0
13	0	0	0	2	0	0	1	2	0	1	5	1	0	0	0	0	0	0	0	0	0	2	3	0	0	0	3	0	4	0
14	0	0	0	0	0	1	0	7	2	0	0	0	0	0	0	3	5	5	0	0	2	0	3	0	4	0	0	0	0	0
15	0	0	3	0	0	1	4	0	0	0	0	0	0	0	3	0	0	0	7	0	0	0	2	0	1	0	0	0	0	0
1	0	0	4	0	0	0	5	0	0	0	0	0	0	0	0	1	1	0	0	1	0	5	0	0	0	0	0	0	0	2
2	2	0	0	3	3	0	0	0	0	0	3	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	1	0	3	0
3	0	0	4	0	0	0	5	0	0	0	0	0	0	3	0	0	0	0	2	0	0	4	0	0	0	4	4	3	0	0
4	0	0	0	0	0	0	0	0	0	0	1	4	4	0	1	0	4	0	0	0	0	0	0	1	0	1	0	0	3	0
5	5	0	3	3	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	5	0	1	0	2	0	0
6	0	0	0	0	0	2	1	0	6	5	0	4	0	0	1	0	0	0	0	0	2	3	0	0	0	0	0	0	0	0
7	0	2	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	2	0	4	0	0	0	0
8	0	0	0	0	0	5	0	0	0	3	0	0	1	0	3	1	0	0	3	0	0	0	0	0	0	0	0	1	0	4
9	0	0	0	0	0	0	0	4	0	1	0	0	4	3	1	0	0	0	0	1	1	0	4	4	0	0	0	0	2	1
10	0	6	0	4	2	0	1	0	0	0	0	0	0	0	2	1	2	0	0	0	0	0	0	0	0	1	3	0	0	0
11	0	0	0	0	6	0	0	0	0	0	5	0	0	0	0	0	0	0	1	3	0	0	3	0	5	0	0	0	4	0
12	0	0	1	0	0	3	0	0	2	0	0	0	0	0	0	6	1	0	0	2	4	0	2	0	0	0	4	0	0	0
13	3	0	0	0	0	0	0	2	0	0	0	4	0	0	4	3	0	0	0	0	5	0	0	0	4	0	0	6	0	5
14	2	0	0	0	1	2	0	3	0	0	0	0	0	0	0	0	0	5	0	0	0	0	3	0	0	1	0	0	0	0
15	0	4	0	2	0	0	0	0	4	3	3	0	3	6	0	0	0	3	0	5	0	0	0	0	3	0	0	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.7 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	5	0	0	0	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	1	0	0	0	6
2	0	0	1	0	1	0	4	4	0	0	0	0	3	0	0	0	3	0	0	0	0	4	0	0	0	0	0	0	0	0
3	0	0	0	0	0	3	1	8	0	2	0	0	0	2	0	0	0	0	5	0	0	0	2	0	0	5	0	0	0	2
4	3	0	4	0	2	3	0	0	2	0	0	0	0	0	0	0	3	0	0	2	0	0	0	3	3	1	0	0	0	1
5	0	0	0	1	6	1	0	0	5	0	0	0	7	0	0	1	0	2	0	0	0	4	0	0	0	0	0	6	0	0
6	2	3	0	7	0	0	0	0	0	0	0	1	0	0	0	2	0	0	1	2	0	0	0	1	0	3	1	0	0	0
7	0	4	0	3	0	5	0	0	0	4	0	0	2	0	0	0	0	0	0	0	2	0	0	0	3	0	2	1	0	0
8	0	0	0	1	3	0	0	0	2	0	0	2	0	0	6	0	0	0	1	0	0	0	0	2	0	0	0	0	2	0
9	1	0	0	0	0	0	5	0	0	3	2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	4	0
10	0	0	0	0	0	0	0	0	0	0	6	2	0	5	0	0	0	1	1	0	4	0	8	2	0	1	0	1	0	0
11	0	0	0	0	0	0	0	0	0	0	3	1	0	0	0	0	4	0	0	2	0	0	0	0	2	0	0	0	3	3
12	0	0	0	0	0	0	0	0	0	0	0	2	0	3	0	3	0	0	0	0	6	0	2	4	0	0	3	2	0	0
13	2	0	1	0	0	0	1	0	0	2	1	0	0	0	5	0	1	4	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	3	0	0	0	0	0	1	0	0	4	0	0	0	4	0	0	4	0	0	0	0	0	1	0	6	2	3	0
15	4	0	3	0	0	0	0	0	0	1	0	0	0	2	0	2	1	5	0	6	0	2	0	0	3	0	0	0	0	0
1	3	0	0	0	0	0	4	4	0	0	1	0	0	0	0	3	3	0	0	0	0	4	0	6	0	0	0	0	0	0
2	2	5	0	0	0	3	2	3	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	2	5	0	0	0	0
3	0	0	0	0	0	1	0	0	3	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	5	3	2	0	0
4	0	0	0	4	1	3	0	0	0	0	0	0	0	2	0	0	1	0	1	0	0	0	0	0	1	0	2	3	0	0
5	0	0	3	3	0	3	0	3	0	0	0	0	3	1	0	0	5	0	0	0	0	0	0	0	0	0	0	0	2	0
6	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	7	5	4	0	0	0	0	0	0	0
7	4	0	1	2	1	0	0	0	0	0	0	2	0	0	1	3	0	0	6	6	1	1	0	0	0	0	0	0	0	0
8	0	0	0	1	5	0	0	0	3	3	0	0	0	0	0	3	0	0	0	0	0	0	0	0	2	0	5	1	0	4
9	0	1	0	0	0	0	0	0	2	0	4	0	0	0	0	0	0	2	1	0	0	0	0	1	6	0	0	0	3	0
10	3	0	3	0	0	0	0	0	0	1	4	4	9	0	0	0	0	0	0	0	0	1	4	0	1	0	0	0	7	0
11	0	0	0	0	0	0	0	0	0	4	0	0	0	2	3	0	0	0	0	1	0	0	0	2	0	0	0	6	0	8
12	0	0	0	0	0	2	0	0	3	1	0	6	0	0	8	3	1	0	0	0	0	1	0	0	0	2	0	0	0	0
13	0	4	0	0	0	0	2	0	0	3	0	0	0	0	0	0	0	5	2	0	0	0	0	0	0	0	2	0	0	0
14	0	0	5	2	5	0	4	2	0	0	0	0	0	0	0	0	0	0	2	0	2	0	2	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	3	0	0	2	0	0	0	4	0	5	2	0	2	3	0	0	0	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.7 (Continued)

Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
														Period																
1	0	0	3	0	0	0	0	7	0	1	0	1	0	3	2	3	5	0	1	0	0	0	0	1	1	0	0	0	0	2
2	0	0	0	0	0	0	0	0	0	1	1	2	0	0	0	0	0	0	7	0	2	0	0	1	0	0	2	8	0	0
3	0	0	0	2	0	0	2	0	0	0	0	0	0	1	0	0	0	0	1	0	0	4	3	0	0	0	0	0	4	0
4	0	0	0	2	2	1	1	0	0	0	0	0	0	2	0	2	0	0	1	0	0	0	2	0	0	0	0	0	0	3
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	7	0	0	0	0	3	0	0	2	6	0	0	0	0
6	4	9	0	0	0	0	2	0	0	1	4	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	7
7	0	2	1	0	0	3	0	0	0	0	0	0	4	1	4	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0
8	0	0	0	0	0	8	0	0	7	4	6	3	2	0	0	0	0	0	0	4	0	0	1	0	4	0	3	0	0	0
9	2	0	0	3	2	0	6	0	0	0	0	2	0	0	0	0	0	0	0	0	1	2	0	0	2	0	0	3	0	0
10	5	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	2	0	0	5	0	0	0	0	4	0	0	2	0
11	0	1	0	0	3	0	0	0	0	5	0	0	0	4	4	0	0	0	0	0	0	2	0	0	0	2	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	5	1	0	4	1	2	4	0	0	1	0	0	0
13	0	0	3	0	0	0	0	0	2	0	0	0	0	0	0	1	0	4	1	0	0	0	0	2	0	0	6	0	0	0
14	1	0	0	5	5	0	0	0	0	0	1	0	2	0	0	0	0	0	0	4	0	0	0	0	3	0	0	1	0	0
15	0	0	5	0	0	0	0	5	3	0	0	0	4	0	0	6	0	1	0	4	0	0	4	0	0	0	0	0	0	0
1	0	0	0	0	2	2	7	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	2	4	0	0	0	0
2	0	0	3	4	0	4	0	3	3	4	0	0	0	0	0	0	1	0	0	0	3	0	0	0	0	0	0	0	0	3
3	2	0	3	0	0	0	0	0	0	0	2	0	1	0	2	0	0	0	0	5	0	0	0	0	0	0	2	0	0	0
4	2	1	0	0	0	0	3	0	0	6	0	0	3	0	1	0	0	3	2	0	0	2	0	0	0	0	0	0	0	0
5	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	2	0	0	5	0	0	4	6	0
6	0	5	0	2	5	0	1	0	0	0	0	0	0	2	0	3	0	0	0	0	3	3	4	2	0	0	0	0	0	0
7	2	0	1	2	0	0	0	1	0	0	5	4	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
8	6	0	0	0	0	0	0	0	2	0	0	0	0	0	0	2	4	0	0	0	0	0	0	0	0	5	0	1	1	0
9	0	0	0	0	0	1	1	0	4	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	6	0	0	6
10	0	0	0	0	0	0	0	0	0	0	0	0	2	0	5	0	4	4	2	0	0	0	5	0	0	0	0	0	0	0
11	0	3	0	0	0	1	0	0	0	0	0	2	1	0	0	0	0	0	0	6	0	0	2	0	0	1	0	7	2	3
12	0	0	0	0	0	4	0	4	0	0	4	6	5	0	0	1	0	0	6	0	0	5	0	7	0	0	0	0	0	0
13	0	0	2	0	0	0	0	0	3	0	1	0	0	0	4	0	3	5	0	0	4	0	0	0	4	0	0	0	0	0
14	0	0	0	4	3	0	0	4	0	0	0	0	0	4	0	0	0	0	1	0	1	0	0	0	1	0	0	0	3	0
15	0	3	0	0	2	0	0	0	0	2	0	0	0	5	0	3	0	0	1	0	0	0	0	2	0	2	4	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.8 Demand data set 8: Large problem size, lumpy demand, high utilization

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	4	0	0	10	0	7	0	3	5	5	0	6	0	0	0	0	0	0	0	0	4	0	3	0	0	0	5	0	1
2	0	0	0	0	0	1	0	4	0	0	2	0	0	0	0	0	0	8	0	0	0	0	0	0	1	0	4	0	0	1
3	0	0	0	2	6	5	5	0	2	3	1	2	2	0	0	0	0	0	0	0	0	0	0	0	0	5	1	0	0	4
4	2	0	0	6	0	0	0	5	0	1	2	8	3	0	0	0	0	0	0	0	0	0	11	0	8	0	2	0	0	0
5	0	0	1	0	0	6	0	0	0	1	0	0	0	0	0	0	2	0	0	3	5	0	0	0	0	0	1	4	0	0
6	7	0	0	0	0	0	0	1	5	0	0	0	0	0	0	0	0	6	3	1	0	0	0	0	0	0	5	0	4	0
7	0	6	4	5	0	0	0	0	4	0	0	0	0	0	0	0	6	0	5	0	0	0	0	0	1	5	0	4	0	1
8	0	0	6	0	0	2	0	0	0	0	2	1	0	8	0	0	0	0	0	0	2	4	0	4	0	0	0	0	0	5
9	0	0	3	0	0	0	0	0	1	0	2	0	0	0	0	0	4	0	3	0	7	2	0	3	7	0	0	0	1	0
10	2	0	1	4	0	0	0	0	0	0	0	0	5	0	0	4	0	2	5	0	0	1	0	0	0	0	0	2	3	0
11	0	0	0	0	0	0	1	3	0	5	2	0	1	0	8	0	0	2	0	7	0	0	0	0	0	0	0	0	3	5
12	0	5	1	1	0	0	1	0	0	0	0	4	0	10	0	6	0	0	2	0	0	0	0	6	0	0	5	3	0	0
13	0	0	2	0	0	0	0	0	3	0	0	0	1	0	7	3	0	0	0	2	4	0	7	2	0	5	0	0	0	1
14	0	3	0	0	2	4	0	5	0	1	0	2	0	0	0	0	6	0	0	5	0	7	0	0	0	0	0	0	7	0
15	7	0	0	0	0	0	4	0	0	2	2	1	0	0	3	5	0	0	0	0	0	0	0	0	1	3	0	0	0	0
1	2	0	0	4	0	0	0	4	0	0	0	0	6	0	0	4	0	0	0	0	0	2	3	0	2	3	4	7	0	0
2	0	0	0	0	0	0	0	5	0	1	0	0	0	2	1	0	0	0	6	0	0	3	3	0	0	0	0	0	1	0
3	0	0	0	0	0	0	0	0	0	0	1	0	5	2	0	0	1	5	0	7	8	0	5	5	0	0	0	0	0	0
4	0	4	0	0	0	3	7	0	7	0	0	0	0	0	0	2	0	0	0	0	0	3	0	1	6	0	0	4	0	18
5	0	0	0	0	0	1	0	0	0	6	0	6	0	0	6	0	5	0	12	0	6	0	0	0	0	0	3	1	0	0
6	1	0	0	1	0	0	0	0	0	6	6	0	0	3	0	4	0	2	0	0	0	1	1	0	0	0	0	0	3	0
7	0	0	0	0	5	0	0	0	0	0	0	0	5	0	0	1	3	0	0	0	0	0	0	7	0	1	0	6	4	0
8	0	0	8	0	0	0	4	0	8	0	1	0	0	4	0	0	1	0	0	4	0	5	0	0	0	0	5	0	0	0
9	0	4	3	0	0	0	2	0	1	0	0	0	0	0	0	0	1	0	0	0	2	0	2	4	0	0	4	0	0	0
10	5	0	4	6	4	0	5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	2	0
11	2	0	0	0	0	7	0	6	0	0	3	0	0	5	4	0	2	8	0	0	0	0	0	1	0	0	0	0	0	0
12	0	5	0	0	5	7	0	0	0	0	0	0	2	0	1	1	3	1	0	0	2	0	0	0	4	0	0	0	0	0
13	2	0	0	5	0	0	0	3	2	5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	0	0	6	0
14	6	5	0	2	0	0	0	0	0	0	0	6	0	0	4	6	0	0	0	0	0	0	4	0	0	4	2	0	0	0
15	0	0	3	0	4	0	0	0	0	0	7	5	0	1	2	0	2	2	0	7	0	0	0	0	6	4	0	0	2	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.8 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	4	2	0	0	0	3	0	0	0	0	0	0	0	0	0	1	5	0	0	0	0	0	1	0	0	0	0	8	0	0
2	0	0	2	0	0	0	0	6	0	4	0	0	0	0	0	6	1	8	0	0	0	4	5	0	0	0	0	0	0	0
3	0	0	0	0	2	0	0	0	0	0	0	1	1	2	0	0	0	0	0	4	0	1	0	0	0	3	0	0	0	0
4	0	5	3	0	2	0	7	0	0	0	0	0	3	0	0	4	4	4	0	0	0	0	0	6	0	0	0	0	0	7
5	1	0	0	0	3	0	0	0	2	5	0	3	1	0	0	0	2	0	0	0	0	0	0	0	7	3	3	0	0	0
6	7	4	1	0	0	7	2	0	3	6	0	2	0	0	7	0	0	0	0	0	6	4	0	4	0	0	0	0	0	0
7	0	0	0	1	0	0	0	0	2	3	8	0	8	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0
8	0	4	3	0	0	2	0	0	0	0	0	0	2	5	0	2	0	0	0	7	0	0	3	0	0	6	0	0	0	0
9	0	0	6	1	0	4	0	0	3	0	10	3	0	0	0	0	0	0	0	0	0	0	1	4	0	0	6	3	0	4
10	0	2	0	2	0	1	0	6	0	0	0	0	1	5	0	0	0	0	3	0	0	3	0	0	0	0	5	0	7	0
11	0	0	0	0	0	1	1	0	3	0	0	0	0	0	0	0	6	0	8	7	0	0	0	0	0	0	0	7	3	0
12	5	1	0	7	6	0	0	0	0	0	0	6	0	0	0	0	0	6	3	0	0	0	7	2	4	0	2	0	0	0
13	0	0	0	4	0	0	4	0	0	0	0	3	2	6	8	5	0	0	4	0	4	0	0	0	5	4	0	0	0	0
14	1	0	0	0	0	0	0	6	5	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	2	2	2	0	3	7
15	0	0	3	3	5	0	4	0	0	0	0	0	0	0	3	0	0	0	0	0	0	6	0	1	0	0	0	0	5	0
1	5	0	0	0	0	0	0	0	0	0	5	4	0	0	0	7	4	0	5	4	0	0	0	0	2	2	2	0	0	7
2	0	0	10	2	3	0	0	0	1	4	0	0	0	0	2	0	0	3	0	0	0	0	0	0	3	0	3	0	0	0
3	0	0	8	1	1	0	0	6	1	8	0	0	0	0	6	0	0	2	0	0	0	0	0	5	4	0	0	0	0	5
4	3	3	0	0	7	0	0	0	1	0	0	5	0	0	6	0	2	5	0	0	0	0	6	0	0	1	0	0	4	4
5	0	0	0	0	4	5	0	0	0	0	2	0	4	0	0	0	0	0	6	0	0	1	0	0	0	0	4	0	4	0
6	0	0	0	6	0	0	5	0	0	0	0	0	0	6	0	0	0	0	0	0	0	2	2	3	3	0	3	0	3	2
7	0	2	0	4	0	0	0	0	3	0	0	1	2	7	0	0	0	0	0	0	0	5	0	0	0	0	4	3	0	0
8	0	5	0	0	0	0	0	3	5	0	0	0	0	0	1	0	0	0	2	6	0	2	0	0	0	2	1	0	0	0
9	0	0	0	0	3	9	3	6	5	1	0	3	0	0	0	0	0	4	1	0	0	0	4	1	0	1	0	0	0	0
10	0	1	0	0	0	4	0	0	2	0	0	0	5	5	0	1	0	4	0	0	7	2	0	0	0	0	0	7	0	0
11	0	1	0	0	0	0	7	1	0	0	0	0	6	0	0	4	0	0	0	4	4	6	0	0	2	0	0	1	5	0
12	3	4	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	2	4	2	1	4	0	0
13	0	0	0	4	0	0	1	2	0	0	2	2	0	0	0	0	4	0	4	2	0	0	5	1	0	6	0	0	0	0
14	0	2	0	0	0	0	2	0	0	0	4	3	0	0	3	1	2	0	0	2	0	0	0	4	0	4	0	0	2	0
15	7	0	0	0	0	0	0	0	0	5	5	0	1	0	0	4	6	0	0	0	7	0	0	2	0	0	0	3	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.8 (Continued)

Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
	Period																													
1	8	0	0	0	2	0	0	0	0	0	5	0	0	0	0	0	4	0	7	0	0	0	0	0	4	0	8	0	5	2
2	0	0	0	2	0	0	3	5	0	0	1	1	0	0	0	0	1	2	0	0	0	0	0	0	6	0	0	0	6	0
3	6	1	2	0	0	0	5	0	6	7	0	6	0	7	0	0	0	0	0	3	6	0	0	0	0	0	0	8	0	0
4	0	4	0	0	0	0	0	0	4	0	0	0	1	0	0	4	3	0	0	0	8	2	0	0	0	6	0	0	0	0
5	0	0	3	0	3	6	2	0	0	0	6	0	0	0	6	0	0	0	0	0	0	0	3	0	0	0	1	0	0	4
6	0	0	7	0	1	0	0	0	0	0	0	4	0	1	0	1	4	0	0	0	0	0	0	0	0	0	0	4	2	0
7	0	4	0	0	0	5	1	0	0	0	2	0	1	0	0	0	5	0	0	0	4	0	0	0	0	5	0	6	0	0
8	0	5	3	0	0	0	0	2	0	0	0	2	3	0	5	0	0	5	0	0	0	0	4	3	3	0	0	0	0	0
9	3	2	0	4	0	7	0	0	0	4	0	5	0	3	5	0	0	4	0	0	0	0	0	0	0	0	0	0	0	1
10	0	2	0	5	0	0	0	5	8	0	2	0	0	0	0	3	0	0	3	2	0	0	0	4	0	0	2	0	0	0
11	0	0	0	0	4	0	0	6	0	0	1	0	0	4	2	0	0	5	0	0	0	7	0	0	0	0	0	0	0	6
12	0	0	3	4	0	0	0	0	0	0	0	0	4	3	0	0	0	0	1	5	0	0	6	2	3	5	0	0	0	0
13	0	0	0	0	0	0	7	0	0	0	0	0	0	0	0	2	0	0	1	4	0	7	1	6	2	0	7	0	5	0
14	1	0	0	1	3	0	0	0	0	1	0	0	3	0	0	8	1	0	0	3	0	0	0	2	0	0	0	0	0	5
15	0	0	0	2	5	0	0	0	0	6	1	0	6	0	0	0	0	2	6	1	0	2	4	1	0	2	0	0	0	0
1	6	0	0	0	0	0	0	1	5	0	0	0	0	7	0	0	0	0	3	0	7	0	3	0	0	0	0	2	0	3
2	4	0	4	0	0	5	0	1	0	8	0	2	0	0	0	0	5	6	0	0	0	0	3	0	0	0	0	0	5	0
3	0	0	0	0	0	3	0	3	0	3	0	3	0	2	0	0	7	0	3	0	0	2	0	3	0	5	5	0	0	0
4	0	2	3	0	0	0	3	5	0	0	0	0	0	0	0	0	0	0	6	5	0	0	0	2	4	8	6	0	4	7
5	2	1	0	0	0	0	0	0	3	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0
6	0	0	0	0	4	0	1	0	0	3	4	3	1	0	2	0	0	0	0	2	0	0	0	0	6	0	0	0	6	3
7	0	0	0	0	0	0	5	0	5	0	1	0	0	0	6	7	0	0	4	0	0	3	0	0	0	0	0	1	0	0
8	0	2	6	2	6	0	0	0	0	1	4	6	7	0	0	0	0	0	0	0	0	0	1	0	3	0	1	5	0	0
9	0	0	4	0	0	2	0	5	0	0	0	3	0	0	0	1	3	0	0	0	1	3	0	3	3	0	0	6	0	1
10	0	5	1	1	0	6	6	0	0	0	0	0	0	0	1	0	0	2	0	2	4	3	0	0	0	0	0	0	3	3
11	0	0	0	0	5	0	3	0	2	3	8	0	7	4	2	3	0	0	0	3	0	0	0	3	0	0	0	0	0	0
12	0	6	0	4	0	0	0	3	0	0	0	1	0	0	0	0	0	0	2	3	0	2	7	0	0	0	5	0	0	0
13	6	0	0	2	0	2	0	0	3	0	0	0	0	5	0	0	0	5	0	0	0	3	0	0	0	1	0	0	0	1
14	0	0	0	4	3	0	0	0	0	0	0	0	0	0	5	0	0	3	0	3	4	2	3	0	0	4	0	4	0	0
15	0	2	0	5	0	0	0	0	0	0	0	0	2	0	2	7	3	2	0	0	2	0	1	7	0	0	0	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.8 (Continued)

Product	Period 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	4	0	3	0	0	0	2	0	0	4	0	4	0	0	0	0	0	0	0	0	0	2	0	1	5	5	0	0	6
2	0	0	1	3	4	0	0	0	0	0	0	0	6	0	1	0	1	0	2	0	3	0	0	0	1	0	0	0	0	0
3	0	0	0	0	0	0	6	0	9	0	0	4	0	0	0	0	0	7	8	2	0	6	0	0	5	0	0	0	0	7
4	0	6	0	0	0	0	2	0	0	0	0	3	0	1	0	0	0	0	1	0	0	7	0	0	0	0	0	4	4	0
5	0	0	3	0	2	3	0	0	0	0	2	0	0	5	0	0	4	0	0	0	0	0	0	0	3	3	0	0	0	0
6	0	0	0	0	0	0	6	0	0	5	2	5	0	0	0	2	0	0	4	0	0	1	0	0	5	0	4	0	0	0
7	0	0	0	4	2	0	3	0	0	0	2	0	1	4	0	0	0	0	0	0	0	4	0	3	0	1	0	2	6	0
8	6	0	0	5	1	5	0	5	0	4	0	0	0	0	0	0	6	3	0	6	0	0	0	0	0	0	2	0	0	0
9	2	0	0	0	0	0	0	0	0	0	4	0	0	3	0	0	0	0	0	1	0	0	0	3	1	5	2	0	0	0
10	4	0	5	0	0	0	1	8	0	0	2	5	0	0	7	0	0	1	0	0	0	0	2	0	0	0	2	5	0	0
11	1	0	5	0	5	0	0	0	0	0	0	0	0	0	0	8	7	0	0	0	6	0	0	3	0	0	3	0	2	2
12	0	0	0	3	4	0	0	3	0	0	0	0	0	0	6	2	0	2	0	7	0	0	6	5	0	0	0	0	0	0
13	0	8	1	0	0	1	0	0	0	6	0	0	7	0	0	0	0	0	0	2	9	0	0	2	2	1	0	3	3	0
14	1	0	3	0	0	5	0	0	9	0	0	0	0	3	0	6	0	5	0	0	0	0	8	0	0	3	0	0	3	0
15	4	0	0	0	0	4	0	0	0	3	2	1	0	2	4	0	0	0	3	0	0	0	0	2	0	0	0	4	0	3
1	6	0	0	0	0	0	0	0	0	1	0	1	3	0	0	0	7	0	0	0	6	0	0	0	1	0	0	0	4	0
2	0	0	0	0	0	6	0	2	10	7	5	0	0	1	4	1	0	4	0	0	1	0	0	0	0	0	0	0	0	6
3	0	0	5	4	5	5	5	0	0	0	3	0	5	0	0	0	0	0	2	0	0	0	0	2	6	0	4	0	0	0
4	2	0	0	1	0	0	0	0	7	0	0	0	3	4	0	0	3	0	0	0	0	4	0	4	0	3	0	3	6	0
5	8	0	1	0	10	0	0	1	0	0	3	0	0	0	0	0	0	4	0	0	1	0	1	0	0	0	0	2	4	0
6	0	0	0	0	0	0	0	0	0	0	0	7	2	0	2	0	0	0	0	0	0	1	0	0	0	4	4	2	0	0
7	2	1	0	0	0	0	0	0	1	0	0	6	0	4	6	0	0	0	0	0	0	4	5	0	0	0	3	0	0	3
8	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	5	0	6	2	0	4	0	4	7	4	6	0	1	0	0
9	0	0	5	0	0	2	0	3	0	0	0	4	0	5	1	4	0	0	0	0	0	0	0	4	0	0	0	0	0	0
10	0	8	1	0	0	0	0	1	0	0	0	0	0	0	3	0	0	4	0	3	0	7	0	0	0	0	1	4	0	0
11	0	2	0	0	0	5	5	5	0	0	3	0	0	2	0	0	0	0	0	7	0	0	0	1	2	0	0	0	4	6
12	0	0	5	3	0	0	0	0	0	0	4	0	1	0	0	0	3	0	7	0	0	2	6	0	0	5	0	1	0	3
13	0	0	0	0	0	0	4	6	0	2	0	0	2	0	2	1	0	0	3	6	6	0	0	0	1	0	0	0	0	0
14	0	0	1	5	0	0	0	0	0	8	0	0	0	0	0	7	5	0	3	0	0	0	0	0	4	0	0	0	0	0
15	0	7	0	5	3	0	4	0	0	0	0	0	2	0	0	0	0	0	1	2	0	0	2	0	0	0	6	5	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.8 (Continued)

	Period																													
Product	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0	3	0	0	2	0	0	4	0	0	1	0	0	0	0	5	7	0	0	0	0	4	0	4	0	0	3	0	0	0
2	0	4	0	0	5	0	0	0	0	0	0	5	0	0	5	0	7	3	0	2	0	0	0	0	2	0	0	4	0	0
3	0	1	0	2	3	0	0	0	0	0	6	0	0	0	0	0	1	4	0	0	0	0	0	0	0	1	0	0	2	2
4	0	1	3	6	0	0	0	0	0	0	0	0	0	0	2	1	0	2	0	0	1	4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	5	0	0	3	0	6	0	0	0	6	0	0	0	0	0	0	0	1	5	0	3	2	0	3
6	0	0	7	0	8	0	0	0	0	1	6	0	0	0	0	0	0	0	5	0	7	0	0	5	0	0	0	3	0	7
7	0	0	0	4	0	7	0	0	1	5	0	0	0	3	0	0	3	0	7	0	4	0	0	0	0	0	5	0	0	0
8	0	0	2	1	0	4	6	0	0	0	5	0	0	0	0	0	0	0	6	2	0	5	0	0	0	7	0	0	3	0
9	0	0	0	0	0	0	6	4	1	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	3	3	0	1	3
10	0	0	0	0	0	0	1	0	8	5	0	0	2	0	0	0	0	1	0	2	0	0	6	6	7	0	0	0	2	0
11	18	0	2	1	0	0	0	4	0	4	0	0	5	8	0	0	0	3	0	0	0	0	0	0	4	4	0	4	0	3
12	0	6	0	0	0	0	0	6	0	0	0	5	6	5	0	1	0	5	0	6	0	0	9	0	0	0	4	0	6	0
13	0	3	0	0	0	2	0	0	8	0	0	2	0	0	4	0	0	0	0	3	1	0	3	0	0	0	0	1	0	0
14	0	0	0	0	0	5	0	0	0	0	0	0	5	0	1	2	0	0	0	0	5	0	0	0	0	2	0	4	4	0
15	0	0	4	4	0	0	0	0	0	0	0	0	0	0	6	3	0	0	0	3	0	5	0	2	0	1	0	0	0	0
1	0	0	0	0	0	4	0	2	1	1	0	0	0	0	9	0	0	0	4	0	5	0	2	0	0	0	3	0	0	0
2	0	0	0	0	0	0	0	0	8	0	5	0	1	0	0	0	2	0	0	0	0	0	0	4	0	5	0	0	5	0
3	0	3	0	2	0	5	0	0	1	0	0	5	0	4	0	0	2	0	0	4	0	1	3	0	0	6	0	0	0	1
4	0	0	0	0	2	0	1	0	0	0	2	2	3	0	4	0	5	0	0	0	0	1	0	1	0	0	3	2	0	3
5	0	0	5	6	2	0	7	4	4	5	3	0	0	0	0	2	0	0	0	0	1	0	0	0	0	0	0	1	0	0
6	0	0	0	6	8	0	0	6	1	0	0	5	0	0	0	0	0	9	0	0	0	3	0	0	0	0	0	0	4	1
7	0	4	3	0	0	0	7	3	0	2	1	0	3	0	0	0	6	0	0	6	0	0	0	0	0	0	0	0	4	0
8	4	2	0	0	0	4	0	0	0	0	0	0	0	4	0	0	3	9	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	3	0	2	0	0	0	3	2	5	0	3	5	0	4	0	0	0	0	0	5	1	0	0	0	0	0	0	8
10	0	0	1	2	3	0	0	3	0	5	2	0	0	0	4	4	0	0	2	0	0	0	0	0	0	0	0	3	0	0
11	5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	0	0	7	0	5	0	0	0	0	2	6	1	0	4
12	0	0	0	0	1	0	1	0	0	0	0	1	2	0	0	0	0	0	0	4	0	5	8	0	6	5	0	7	0	0
13	0	0	0	0	0	3	2	0	0	0	0	5	1	0	1	0	0	0	5	0	0	0	4	3	5	0	0	0	5	0
14	9	2	6	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	3	6	0	0	5	7	0	4	4	0	1
15	0	7	0	0	0	2	0	0	0	1	0	0	5	5	0	2	0	0	0	1	1	3	0	5	0	0	2	0	0	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.9 Setup matrix used for small size problems

	To Product						
From Product	0	1	2	3	4	5	6
0	0	0	0	0	0	0	0
1	0	0	3	5	4	2	2
2	0	6	0	1	4	1	1
3	0	2	6	0	1	1	6
4	0	6	4	4	0	4	1
5	0	3	5	6	4	0	4
6	0	1	4	6	5	3	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table B.10 Setup matrix for large size problems

	To Product															
From Product	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	3	5	4	2	2	6	1	4	1	1	2	6	1	1
2	0	6	0	6	4	4	4	1	3	5	6	4	4	1	4	6
3	0	5	3	0	3	6	3	5	6	6	5	2	4	1	3	3
4	0	4	4	2	0	3	4	6	3	1	4	3	5	6	1	5
5	0	1	2	6	2	0	2	2	2	1	5	4	1	6	6	5
6	0	5	5	3	1	5	0	1	5	2	2	1	6	1	2	1
7	0	5	2	2	2	1	6	0	3	6	4	1	2	2	3	4
8	0	3	2	1	4	1	4	1	0	6	3	6	3	5	2	1
9	0	6	5	4	5	5	5	5	4	0	3	3	3	4	6	1
10	0	2	6	1	2	2	3	5	3	1	0	6	3	5	4	3
11	0	2	2	6	4	5	2	1	6	2	6	0	4	6	4	2
12	0	6	5	2	4	6	2	6	4	4	5	5	0	2	3	5
13	0	3	5	1	1	2	6	2	2	3	5	6	6	0	3	3
14	0	1	4	6	6	2	3	6	1	4	2	5	2	2	0	4
15	0	6	1	6	5	4	1	5	4	2	1	4	1	6	2	0

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX C

OUTPUT GENERATED BY ALGORITHMS

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 Experimental results

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
1	1	1	1	1	1	1	1	221	229	194	219	178	199	199	188	202	210
1	1	1	1	1	1	1	2	208	209	191	215	178	210	191	173	225	186
1	1	1	1	1	1	2	1	196	223	177	224	178	197	214	220	191	203
1	1	1	1	1	1	2	2	196	199	177	164	178	197	220	220	191	203
1	1	1	1	1	2	1	1	442	481	498	540	502	465	482	545	469	482
1	1	1	1	1	2	1	2	506	457	463	523	542	493	518	489	486	540
1	1	1	1	1	2	2	1	529	464	453	545	579	507	493	511	512	466
1	1	1	1	1	2	2	2	550	470	468	579	615	564	529	475	531	514
1	1	1	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	1	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	1	2	1	1	1	1	139	139	120	112	116	122	122	105	151	109
1	1	1	2	1	1	1	2	114	119	120	112	116	114	122	107	151	113
1	1	1	2	1	1	2	1	116	119	104	111	116	117	114	105	129	107
1	1	1	2	1	1	2	2	116	119	104	111	116	117	114	105	129	110
1	1	1	2	1	2	1	1	375	347	342	344	364	445	342	347	398	380
1	1	1	2	1	2	1	2	383	432	292	367	428	431	388	328	374	430
1	1	1	2	1	2	2	1	436	379	334	334	364	361	354	347	410	389
1	1	1	2	1	2	2	2	433	357	334	334	364	440	354	332	410	369
1	1	1	2	2	1	1	1	INF	131	98	98	105	106	128	INF	97	93
1	1	1	2	2	1	1	2	100	131	98	98	105	106	128	114	97	93
1	1	1	2	2	1	2	1	INF	155	98	124	111	109	128	INF	109	93
1	1	1	2	2	1	2	2	109	155	98	124	111	109	128	126	109	93
1	1	1	2	2	2	1	1	INF	464	368	409	345	INF	470	INF	341	333
1	1	1	2	2	2	1	2	404	469	564	432	363	446	470	440	394	361

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
1	1	1	2	2	2	2	1	INF	503	368	427	354	379	470	INF	368	333
1	1	1	2	2	2	2	2	413	509	368	427	390	514	470	387	368	333
1	1	2	1	1	1	1	1	833	816	840	867	864	848	848	870	847	927
1	1	2	1	1	1	1	2	829	809	817	896	861	818	818	834	801	890
1	1	2	1	1	1	2	1	885	952	803	882	1055	886	939	910	789	827
1	1	2	1	1	1	2	2	856	967	797	803	987	796	867	892	801	770
1	1	2	1	1	2	1	1	2181	2180	2019	2180	2172	1973	2021	2217	2045	2187
1	1	2	1	1	2	1	2	2216	2332	2224	2375	2297	2165	2120	2126	2333	2194
1	1	2	1	1	2	2	1	2121	2375	2292	2429	2475	2322	2273	2230	2321	2319
1	1	2	1	1	2	2	2	2201	2421	2319	2191	2446	2228	2107	2326	2173	2281
1	1	2	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	1	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	1	1	1	1	494	558	509	518	493	496	543	491	519	480
1	1	2	2	1	1	1	2	494	534	522	469	472	479	516	487	541	460
1	1	2	2	1	1	2	1	498	570	528	471	506	491	552	505	509	463
1	1	2	2	1	1	2	2	484	566	508	468	467	484	535	500	510	451
1	1	2	2	1	2	1	1	1486	1754	1699	1635	1586	1706	1811	1748	1562	1478
1	1	2	2	1	2	1	2	1598	1871	1701	1803	1833	1713	2054	1683	1750	1597
1	1	2	2	1	2	2	1	1648	1800	1680	1841	1854	1815	1826	1887	1797	1762
1	1	2	2	1	2	2	2	1738	1741	1632	1864	1818	1808	1895	1828	1832	1651
1	1	2	2	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
1	1	2	2	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	1	2	2	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	1	1	1	1	158	158	166	169	160	166	164	170	166	159
1	2	1	1	1	1	1	2	158	158	170	165	160	166	164	160	166	159
1	2	1	1	1	1	2	1	158	158	166	164	160	160	164	160	166	159
1	2	1	1	1	1	2	2	158	158	166	164	160	160	164	160	166	159
1	2	1	1	1	2	1	1	553	511	497	479	536	596	525	498	591	509
1	2	1	1	1	2	1	2	549	512	470	484	504	547	534	530	523	522
1	2	1	1	1	2	2	1	484	547	498	554	527	507	521	540	563	551
1	2	1	1	1	2	2	2	484	543	498	554	527	507	521	539	563	551
1	2	1	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	1	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	1	2	1	1	1	1	93	92	90	86	111	106	94	98	112	94
1	2	1	2	1	1	1	2	93	92	90	86	111	106	94	98	112	94
1	2	1	2	1	1	2	1	93	92	90	86	111	102	94	98	112	94
1	2	1	2	1	1	2	2	93	92	90	86	111	102	94	98	112	94
1	2	1	2	1	2	1	1	375	377	342	345	398	406	388	343	461	357
1	2	1	2	1	2	1	2	401	377	342	345	398	406	388	343	461	395
1	2	1	2	1	2	2	1	401	377	338	345	398	409	340	343	473	357
1	2	1	2	1	2	2	2	401	377	338	345	398	409	340	343	473	376
1	2	1	2	2	1	1	1	101	143	88	98	88	110	130	INF	97	78
1	2	1	2	2	1	1	2	101	143	88	98	88	110	164	109	97	78

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
1	2	1	2	2	1	2	1	107	146	88	117	88	110	139	INF	91	78
1	2	1	2	2	1	2	2	107	146	88	117	88	110	139	126	91	78
1	2	1	2	2	2	1	1	374	491	368	412	380	410	454	344	343	332
1	2	1	2	2	2	1	2	376	491	368	412	380	410	467	353	343	332
1	2	1	2	2	2	2	1	380	494	368	412	380	410	463	INF	325	332
1	2	1	2	2	2	2	2	385	494	368	412	380	410	488	398	325	332
1	2	2	1	1	1	1	1	690	709	711	723	800	794	647	736	702	732
1	2	2	1	1	1	1	2	690	709	711	723	821	803	647	736	702	732
1	2	2	1	1	1	2	1	711	733	745	712	1099	790	662	773	747	735
1	2	2	1	1	1	2	2	711	733	745	712	981	790	662	773	747	735
1	2	2	1	1	2	1	1	2199	2389	2198	2299	2333	2192	2217	2380	2274	2413
1	2	2	1	1	2	1	2	2171	2418	2283	2226	2476	2242	2193	2372	2163	2042
1	2	2	1	1	2	2	1	2245	2580	2439	2239	2712	2147	2152	2433	2697	2241
1	2	2	1	1	2	2	2	2245	2537	2439	2340	2741	2444	2208	2433	2697	2240
1	2	2	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	1	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	1	1	1	1	435	477	432	440	416	455	458	424	446	420
1	2	2	2	1	1	1	2	435	477	432	440	416	455	458	424	446	420
1	2	2	2	1	1	2	1	435	477	432	440	416	455	458	432	446	420
1	2	2	2	1	1	2	2	435	477	432	440	416	455	458	432	446	420
1	2	2	2	1	2	1	1	1563	1811	1609	1633	1664	1634	1710	1608	1741	1622
1	2	2	2	1	2	1	2	1623	1829	1619	1662	1616	1645	1728	1616	1756	1634
1	2	2	2	1	2	2	1	1620	1804	1687	1709	1613	1753	1696	1668	1764	1680
1	2	2	2	1	2	2	2	1623	1804	1725	1680	1608	1753	1696	1673	1824	1680

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
1	2	2	2	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
1	2	2	2	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	1	1	1	1	1	220	226	196	210	INF	207	190	212	223	199
2	1	1	1	1	1	1	2	214	226	225	217	272	216	210	217	220	203
2	1	1	1	1	1	2	1	227	242	INF	224	244	209	193	210	222	202
2	1	1	1	1	1	2	2	237	242	240	224	244	234	205	230	222	202
2	1	1	1	1	2	1	1	491	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	1	1	2	1	2	539	602	482	487	557	588	480	582	549	523
2	1	1	1	1	2	2	1	524	INF	INF	442	INF	488	INF	542	INF	INF
2	1	1	1	1	2	2	2	584	681	616	623	674	677	598	608	647	683
2	1	1	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	1	2	1	1	2	340	348	312	334	320	345	358	280	324	282
2	1	1	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	1	2	1	2	2	500	396	435	488	503	527	475	566	339	467
2	1	1	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	1	2	2	1	2	719	599	707	693	729	632	642	693	713	841
2	1	1	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	1	2	2	2	2	904	1121	1038	1220	976	968	884	916	936	987
2	1	1	2	1	1	1	1	145	137	108	110	122	133	123	139	150	122
2	1	1	2	1	1	1	2	145	146	116	110	122	133	123	130	150	138
2	1	1	2	1	1	2	1	160	146	108	110	137	143	130	133	142	130
2	1	1	2	1	1	2	2	160	146	108	110	137	143	130	130	142	120
2	1	1	2	1	2	1	1	429	435	315	308	392	344	399	399	435	442
2	1	1	2	1	2	1	2	477	385	292	516	439	368	456	410	491	452

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
2	1	1	2	1	2	2	1	INF	444	342	376	374	368	INF	413	418	440
2	1	1	2	1	2	2	2	495	446	342	542	415	443	393	430	519	503
2	1	1	2	2	1	1	1	117	160	INF	INF	INF	INF	INF	INF	123	INF
2	1	1	2	2	1	1	2	160	170	117	177	143	138	229	186	164	INF
2	1	1	2	2	1	2	1	126	183	INF	INF	INF	INF	INF	INF	156	INF
2	1	1	2	2	1	2	2	126	206	139	230	157	148	322	285	172	INF
2	1	1	2	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	533	INF	INF
2	1	1	2	2	2	1	2	532	698	442	INF	463	494	437	481	336	485
2	1	1	2	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	1	2	2	2	2	2	813	738	429	716	654	1045	686	560	461	717
2	1	2	1	1	1	1	1	911	907	915	902	928	918	925	969	970	927
2	1	2	1	1	1	1	2	913	900	922	920	911	920	930	1006	935	897
2	1	2	1	1	1	2	1	1050	1115	INF	975	976	1054	1036	INF	INF	1066
2	1	2	1	1	1	2	2	1050	1033	1037	1039	998	1036	1054	1128	1029	1053
2	1	2	1	1	2	1	1	INF	INF	INF	INF	2177	INF	2283	INF	2349	INF
2	1	2	1	1	2	1	2	2194	2399	2592	2459	2342	2304	2460	2294	2472	2172
2	1	2	1	1	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	1	1	2	2	2	2499	2663	2430	2753	2487	2559	2348	2556	2810	2762
2	1	2	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	1	2	1	1	2	2035	1851	INF	INF	1907	INF	INF	1944	2026	INF
2	1	2	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	1	2	1	2	2	INF	INF	INF	INF	3777	INF	INF	INF	INF	INF
2	1	2	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	1	2	2	1	2	3708	4079	3705	3855	3654	3660	3523	3886	4011	3809
2	1	2	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	1	2	2	2	2	5339	6259	5755	5320	5760	5713	6151	6285	5816	6353
2	1	2	2	1	1	1	1	555	604	573	527	576	581	617	535	629	559
2	1	2	2	1	1	1	2	560	604	590	533	581	567	625	553	627	582
2	1	2	2	1	1	2	1	545	594	641	552	619	571	621	591	603	544
2	1	2	2	1	1	2	2	550	615	640	563	622	572	618	593	603	563

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
2	1	2	2	1	2	1	1	1697	1779	1931	1670	1966	1953	1894	1691	1952	1814
2	1	2	2	1	2	1	2	1677	2081	1871	1761	1859	1868	1920	1762	2012	1676
2	1	2	2	1	2	2	1	1779	1889	1846	1851	1964	INF	2060	1879	1981	1723
2	1	2	2	1	2	2	2	1772	1962	1881	1841	2066	2195	2074	1926	2062	1830
2	1	2	2	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	2	2	1	1	2	INF	INF	921	1198	INF	INF	INF	INF	INF	INF
2	1	2	2	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	2	2	1	2	2	2283	INF	INF	2025	INF	INF	INF	2922	INF	2572
2	1	2	2	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	2	2	2	1	2	2999	2863	3128	3066	2818	3286	2821	2771	2774	2901
2	1	2	2	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	1	2	2	2	2	2	2	5427	3266	3971	4146	4439	5821	4214	INF	5548	4135
2	2	1	1	1	1	1	1	169	172	170	149	160	177	157	167	166	166
2	2	1	1	1	1	1	2	169	172	170	149	160	177	157	167	166	166
2	2	1	1	1	1	2	1	177	170	168	161	160	177	157	167	178	165
2	2	1	1	1	1	2	2	177	170	168	161	160	177	157	167	178	165
2	2	1	1	1	2	1	1	507	446	481	502	454	366	426	451	454	433
2	2	1	1	1	2	1	2	479	446	433	501	458	387	404	437	454	431
2	2	1	1	1	2	2	1	436	477	480	516	442	468	487	451	509	471
2	2	1	1	1	2	2	2	436	477	480	516	442	468	507	453	509	471
2	2	1	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	1	1	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	1	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	1	1	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	1	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	590	INF	INF
2	2	1	1	2	2	1	2	618	639	612	702	576	618	603	649	559	596
2	2	1	1	2	2	2	1	INF	672	INF	INF	INF	INF	629	INF	753	INF
2	2	1	1	2	2	2	2	INF	725	797	INF	808	INF	629	808	749	841
2	2	1	2	1	1	1	1	99	97	90	86	111	106	94	97	114	101
2	2	1	2	1	1	1	2	99	97	90	86	111	106	94	94	114	95

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
2	2	1	2	1	1	2	1	99	97	90	86	111	109	94	97	114	101
2	2	1	2	1	1	2	2	99	97	90	86	111	109	94	94	114	95
2	2	1	2	1	2	1	1	409	396	326	309	370	401	315	352	472	398
2	2	1	2	1	2	1	2	409	380	326	309	370	401	319	382	472	370
2	2	1	2	1	2	2	1	424	348	326	309	370	405	354	352	434	416
2	2	1	2	1	2	2	2	424	348	326	309	370	405	346	364	434	400
2	2	1	2	2	1	1	1	101	143	88	98	123	110	130	105	97	78
2	2	1	2	2	1	1	2	101	143	88	98	123	110	164	109	97	78
2	2	1	2	2	1	2	1	107	146	88	117	116	110	139	120	91	78
2	2	1	2	2	1	2	2	107	146	88	117	116	110	139	126	91	78
2	2	1	2	2	2	1	1	391	454	363	INF	INF	INF	INF	363	365	INF
2	2	1	2	2	2	1	2	391	454	400	448	410	447	522	363	369	385
2	2	1	2	2	2	2	1	394	INF	391	INF	INF	460	INF	367	378	INF
2	2	1	2	2	2	2	2	394	457	391	485	486	445	556	394	378	390
2	2	2	1	1	1	1	1	651	625	643	669	682	663	636	662	680	657
2	2	2	1	1	1	1	2	651	625	643	669	682	666	635	662	680	667
2	2	2	1	1	1	2	1	651	631	649	673	687	673	623	670	678	661
2	2	2	1	1	1	2	2	651	631	649	673	687	672	622	670	678	670
2	2	2	1	1	2	1	1	1851	1999	1783	2001	1980	1897	1841	1915	1842	1801
2	2	2	1	1	2	1	2	1851	1981	1783	2022	1943	1884	1819	1915	1845	1768
2	2	2	1	1	2	2	1	1993	2175	2140	2127	2117	1918	1925	1961	1934	1928
2	2	2	1	1	2	2	2	1993	2175	2114	2122	2156	1918	1916	1961	1934	1939
2	2	2	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	1	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
2	2	2	2	1	1	1	1	431	474	426	432	413	442	439	425	454	413
2	2	2	2	1	1	1	2	431	474	426	432	413	442	439	425	454	413
2	2	2	2	1	1	2	1	431	474	434	432	413	442	439	425	454	413
2	2	2	2	1	1	2	2	431	474	434	432	413	442	439	425	454	413
2	2	2	2	1	2	1	1	1565	1789	1426	1536	1543	1560	1701	1525	1621	1596
2	2	2	2	1	2	1	2	1442	1789	1441	1527	1548	1582	1669	1505	1630	1564
2	2	2	2	1	2	2	1	1557	1816	1466	1539	1539	1620	1658	1556	1648	1577
2	2	2	2	1	2	2	2	1498	1788	1451	1566	1566	1620	1694	1530	1648	1549
2	2	2	2	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	1	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	1	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	2	1	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
2	2	2	2	2	2	2	2	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	1	1	1	1	1	1	164	144	176	161	198	156	167	172	194	171
3	1	1	1	1	1	1	2	174	154	197	174	208	170	192	181	203	196
3	1	1	1	1	1	2	1	192	168	182	183	186	175	182	197	188	184
3	1	1	1	1	1	2	2	239	264	256	223	234	229	225	257	226	238
3	1	1	1	1	2	1	1	417	418	403	375	386	379	387	357	475	383
3	1	1	1	1	2	1	2	422	406	458	439	444	455	382	398	480	458
3	1	1	1	1	2	2	1	432	495	544	365	410	466	427	435	515	386
3	1	1	1	1	2	2	2	558	776	709	660	578	637	684	594	655	574
3	1	1	1	2	1	1	1	230	207	217	233	198	213	230	INF	200	203
3	1	1	1	2	1	1	2	248	271	235	310	249	255	251	306	209	229
3	1	1	1	2	1	2	1	261	287	274	277	223	297	269	287	246	235
3	1	1	1	2	1	2	2	605	544	521	468	412	420	549	448	443	412
3	1	1	1	2	2	1	1	393	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	1	1	2	2	1	2	709	630	514	694	567	557	494	549	542	548

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
3	1	1	1	2	2	2	1	INF	INF	INF	INF	448	INF	INF	INF	INF	INF
3	1	1	1	2	2	2	2	1007	867	886	891	871	841	855	804	805	756
3	1	1	2	1	1	1	1	131	61	80	107	106	82	93	135	78	111
3	1	1	2	1	1	1	2	131	82	66	91	104	79	114	98	96	81
3	1	1	2	1	1	2	1	146	75	102	131	106	109	108	110	74	149
3	1	1	2	1	1	2	2	146	167	70	109	104	93	108	98	119	79
3	1	1	2	1	2	1	1	316	207	250	308	220	306	251	372	314	244
3	1	1	2	1	2	1	2	465	294	304	273	251	288	405	318	283	343
3	1	1	2	1	2	2	1	403	308	304	344	402	330	240	439	319	339
3	1	1	2	1	2	2	2	590	490	334	402	430	519	326	462	367	378
3	1	1	2	2	1	1	1	125	207	104	119	121	INF	97	77	116	99
3	1	1	2	2	1	1	2	122	180	274	119	144	180	139	268	124	170
3	1	1	2	2	1	2	1	174	INF	140	143	151	INF	108	83	114	144
3	1	1	2	2	1	2	2	258	211	133	143	98	282	211	423	132	305
3	1	1	2	2	2	1	1	360	INF	351	INF	INF	INF	INF	385	INF	INF
3	1	1	2	2	2	1	2	501	390	329	506	340	479	352	463	337	343
3	1	1	2	2	2	2	1	427	INF	389	INF	INF	INF	INF	368	INF	291
3	1	1	2	2	2	2	2	888	510	724	443	546	747	809	728	445	478
3	1	2	1	1	1	1	1	901	844	917	833	905	839	859	941	846	900
3	1	2	1	1	1	1	2	972	1018	1368	863	1323	1031	929	961	958	1074
3	1	2	1	1	1	2	1	1077	1011	1048	967	1072	1112	1018	1101	965	1007
3	1	2	1	1	1	2	2	1284	1260	1451	1376	1135	1395	1614	1278	1448	1257
3	1	2	1	1	2	1	1	2125	1970	1905	1928	2087	2104	1983	1983	1881	1858
3	1	2	1	1	2	1	2	2043	2122	2046	1972	2045	2047	2118	2231	2131	1942
3	1	2	1	1	2	2	1	2289	2228	2323	2144	2283	2267	2366	2153	1977	2082
3	1	2	1	1	2	2	2	2532	2789	2880	2870	2590	2547	2673	2488	2318	2640
3	1	2	1	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	1	2	1	1	2	1576	1802	1562	1631	1843	1727	1675	1627	1613	1565
3	1	2	1	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	1	2	1	2	2	3607	3278	2710	2222	2863	3115	2608	2839	2701	2405

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
3	1	2	1	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	1	2	2	1	2	2860	3032	3352	3215	2972	2855	3194	3244	3338	2771
3	1	2	1	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	1	2	2	2	2	4760	5812	5483	5462	6419	4392	4615	5425	4867	5399
3	1	2	2	1	1	1	1	642	557	612	541	676	598	632	584	610	567
3	1	2	2	1	1	1	2	732	715	600	503	651	744	609	625	634	576
3	1	2	2	1	1	2	1	645	651	748	628	722	866	738	730	734	663
3	1	2	2	1	1	2	2	892	1001	816	602	805	1097	738	842	817	681
3	1	2	2	1	2	1	1	1652	1733	1629	1696	1667	1867	1921	1485	1739	1763
3	1	2	2	1	2	1	2	1589	1525	1743	1538	1616	1739	1614	1463	1643	1884
3	1	2	2	1	2	2	1	1713	1717	1667	1924	1763	2013	1936	1728	1852	1888
3	1	2	2	1	2	2	2	1730	1893	1971	2107	2307	2071	2052	1854	2147	2135
3	1	2	2	2	1	1	1	INF	INF	INF	INF	INF	INF	INF	INF	1083	INF
3	1	2	2	2	1	1	2	1279	996	1358	1273	1134	1355	1308	1195	1506	1218
3	1	2	2	2	1	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	2	2	1	2	2	2387	1684	2062	1810	1948	1811	2031	1952	1919	2009
3	1	2	2	2	2	1	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	2	2	2	1	2	2451	2546	2745	2580	2890	2638	2542	2918	2646	2972
3	1	2	2	2	2	2	1	INF	INF	INF	INF	INF	INF	INF	INF	INF	INF
3	1	2	2	2	2	2	2	4215	3583	3870	5175	4138	4786	5133	5036	3948	5138
3	2	1	1	1	1	1	1	56	50	65	72	38	71	59	74	49	73
3	2	1	1	1	1	1	2	56	50	65	72	38	71	59	74	49	73
3	2	1	1	1	1	2	1	58	46	58	69	38	92	57	67	49	67
3	2	1	1	1	1	2	2	58	46	58	69	38	92	57	67	49	67
3	2	1	1	1	2	1	1	231	275	231	279	208	269	238	264	214	319
3	2	1	1	1	2	1	2	331	297	252	254	208	269	238	269	214	300
3	2	1	1	1	2	2	1	240	376	215	258	258	249	223	292	263	245
3	2	1	1	1	2	2	2	240	420	235	258	262	249	223	294	263	245
3	2	1	1	2	1	1	1	121	86	71	100	74	58	87	84	68	68
3	2	1	1	2	1	1	2	121	86	71	100	74	58	87	84	68	68

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
3	2	1	1	2	1	2	1	130	114	71	83	82	66	114	112	89	77
3	2	1	1	2	1	2	2	130	114	71	83	82	66	114	112	89	77
3	2	1	1	2	2	1	1	347	290	367	258	INF	339	302	355	311	374
3	2	1	1	2	2	1	2	346	318	301	281	401	344	311	279	403	290
3	2	1	1	2	2	2	1	452	346	386	375	338	340	343	333	338	432
3	2	1	1	2	2	2	2	489	463	464	405	382	450	335	362	395	407
3	2	1	2	1	1	1	1	53	30	27	49	56	56	55	46	52	49
3	2	1	2	1	1	1	2	53	30	27	49	50	56	55	40	52	35
3	2	1	2	1	1	2	1	53	30	27	49	56	56	55	46	52	49
3	2	1	2	1	1	2	2	53	30	27	49	50	56	55	40	52	35
3	2	1	2	1	2	1	1	383	226	138	169	188	286	126	221	232	202
3	2	1	2	1	2	1	2	383	247	120	169	170	260	126	258	250	198
3	2	1	2	1	2	2	1	269	212	132	169	165	289	135	270	232	236
3	2	1	2	1	2	2	2	269	212	132	169	170	235	135	228	232	204
3	2	1	2	2	1	1	1	50	56	53	42	54	55	73	37	82	58
3	2	1	2	2	1	1	2	50	56	53	42	54	45	73	37	82	58
3	2	1	2	2	1	2	1	55	56	53	42	54	70	73	37	64	67
3	2	1	2	2	1	2	2	55	56	53	42	54	45	73	37	64	67
3	2	1	2	2	2	1	1	256	358	298	159	246	284	247	250	295	185
3	2	1	2	2	2	1	2	256	308	298	159	246	375	247	250	322	199
3	2	1	2	2	2	2	1	293	447	INF	159	276	305	283	216	267	230
3	2	1	2	2	2	2	2	293	507	369	159	406	323	283	301	267	230
3	2	2	1	1	1	1	1	370	372	390	376	373	374	364	369	350	354
3	2	2	1	1	1	1	2	370	372	390	376	374	369	355	369	350	349
3	2	2	1	1	1	2	1	357	379	398	371	376	360	342	372	347	362
3	2	2	1	1	1	2	2	357	379	398	371	376	355	333	372	347	357
3	2	2	1	1	2	1	1	1711	1339	1418	1295	1419	1533	1536	1394	1441	1365
3	2	2	1	1	2	1	2	1711	1330	1418	1276	1468	1481	1491	1394	1441	1368
3	2	2	1	1	2	2	1	1257	1299	1556	1306	1417	1467	1375	1414	1446	1395
3	2	2	1	1	2	2	2	1271	1299	1556	1293	1438	1488	1374	1414	1446	1386

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
3	2	2	1	2	1	1	1	489	429	504	498	545	506	521	465	487	514
3	2	2	1	2	1	1	2	497	429	519	498	545	506	521	465	487	514
3	2	2	1	2	1	2	1	483	473	530	622	591	542	550	516	513	538
3	2	2	1	2	1	2	2	483	473	530	622	591	542	550	516	513	538
3	2	2	1	2	2	1	1	1675	1808	1773	1721	1970	1780	1653	1853	1841	1878
3	2	2	1	2	2	1	2	1705	1819	1825	1689	1932	1754	1752	1882	1821	1866
3	2	2	1	2	2	2	1	1982	1894	1936	1736	2150	1761	1826	1974	2212	1920
3	2	2	1	2	2	2	2	2231	1894	1936	1833	1938	1812	1878	1949	2064	1816
3	2	2	2	1	1	1	1	295	310	316	311	313	336	320	313	283	312
3	2	2	2	1	1	1	2	295	310	316	311	313	336	320	313	283	312
3	2	2	2	1	1	2	1	295	310	316	311	313	336	320	313	284	312
3	2	2	2	1	1	2	2	295	310	316	311	313	336	320	313	284	312
3	2	2	2	1	2	1	1	1366	1121	1257	1236	1350	1268	1121	1124	1180	1191
3	2	2	2	1	2	1	2	1367	1109	1220	1177	1347	1267	1121	1154	1192	1143
3	2	2	2	1	2	2	1	1294	1146	1258	1249	1290	1292	1131	1189	1205	1174
3	2	2	2	1	2	2	2	1326	1137	1236	1199	1240	1278	1201	1201	1205	1126
3	2	2	2	2	1	1	1	325	387	INF	281	330	306	327	380	284	276
3	2	2	2	2	1	1	2	325	382	438	281	330	290	325	372	284	276
3	2	2	2	2	1	2	1	324	359	401	292	339	314	337	442	276	300
3	2	2	2	2	1	2	2	324	359	401	292	339	290	332	536	276	300
3	2	2	2	2	2	1	1	1439	1308	1390	INF	INF	1503	1360	INF	INF	1303
3	2	2	2	2	2	1	2	1413	1323	1388	1138	1195	1363	1272	1408	1414	1273
3	2	2	2	2	2	2	1	1543	1245	1366	INF	INF	INF	1521	1364	1245	1380
3	2	2	2	2	2	2	2	1544	1300	1486	1136	1637	1319	1521	1595	1242	1373
4	1	1	1	1	1	1	1	390	409	426	387	388	412	408	459	386	394
4	1	1	1	1	1	1	2	390	409	426	387	388	412	408	459	386	394
4	1	1	1	1	1	2	1	408	427	453	414	406	430	420	483	413	418
4	1	1	1	1	1	2	2	408	427	453	414	406	430	420	483	413	418
4	1	1	1	1	2	1	1	614	623	606	620	607	607	596	624	626	616
4	1	1	1	1	2	1	2	614	623	606	620	607	607	596	624	626	616

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
4	1	1	1	1	2	2	1	683	695	681	701	676	676	647	711	713	694
4	1	1	1	1	2	2	2	683	695	681	701	676	676	647	711	713	694
4	1	1	1	2	1	1	1	388	432	414	477	319	482	284	465	306	346
4	1	1	1	2	1	1	2	388	432	414	477	319	482	284	465	306	346
4	1	1	1	2	1	2	1	472	477	486	531	433	557	413	516	474	457
4	1	1	1	2	1	2	2	472	477	486	531	433	557	413	516	474	457
4	1	1	1	2	2	1	1	827	638	635	588	628	709	746	747	610	740
4	1	1	1	2	2	1	2	827	638	635	588	636	709	746	747	610	740
4	1	1	1	2	2	2	1	971	842	821	813	915	964	899	903	871	917
4	1	1	1	2	2	2	2	971	842	821	813	915	964	899	903	871	917
4	1	1	2	1	1	1	1	486	424	456	489	427	512	542	556	406	471
4	1	1	2	1	1	1	2	486	424	456	489	427	512	542	556	406	471
4	1	1	2	1	1	2	1	603	454	525	519	484	554	587	688	454	552
4	1	1	2	1	1	2	2	603	454	525	519	484	554	587	688	454	552
4	1	1	2	1	2	1	1	670	594	660	822	725	626	873	767	602	631
4	1	1	2	1	2	1	2	670	594	660	822	725	626	873	767	602	631
4	1	1	2	1	2	2	1	808	657	789	912	854	713	960	932	671	736
4	1	1	2	1	2	2	2	808	657	789	912	854	713	960	932	671	736
4	1	1	2	2	1	1	1	643	460	536	403	574	414	638	400	501	551
4	1	1	2	2	1	1	2	643	460	536	401	574	414	638	400	501	551
4	1	1	2	2	1	2	1	940	541	743	662	736	564	956	571	864	962
4	1	1	2	2	1	2	2	940	541	743	662	736	564	956	571	864	962
4	1	1	2	2	2	1	1	860	611	770	633	762	648	840	637	733	897
4	1	1	2	2	2	1	2	860	611	770	633	762	648	840	637	733	897
4	1	1	2	2	2	2	1	1388	890	1049	912	957	957	1224	925	1060	1311
4	1	1	2	2	2	2	2	1388	890	1049	912	957	957	1224	925	1060	1311
4	1	2	1	1	1	1	1	2729	2845	2759	2991	2978	2522	2838	2776	2889	2767
4	1	2	1	1	1	1	2	2729	2845	2759	2991	2978	2522	2838	2776	2889	2767
4	1	2	1	1	1	2	1	2795	2980	2858	3099	3068	2627	2970	2839	2961	2866
4	1	2	1	1	1	2	2	2795	2980	2858	3099	3068	2627	2970	2839	2961	2866

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
4	1	2	1	1	2	1	1	4438	5056	4090	4191	3904	3613	4011	4648	3915	4903
4	1	2	1	1	2	1	2	4438	5056	4090	4191	3904	3613	4011	4648	3915	4903
4	1	2	1	1	2	2	1	4624	5227	4372	4437	4195	3745	4260	4816	4212	5164
4	1	2	1	1	2	2	2	4624	5227	4372	4437	4195	3745	4260	4816	4212	5164
4	1	2	1	2	1	1	1	1661	1808	1543	1531	1459	1721	1701	1738	1691	1838
4	1	2	1	2	1	1	2	1661	1798	1535	1531	1459	1721	1701	1738	1691	1838
4	1	2	1	2	1	2	1	2498	2977	2195	2902	2101	2399	2475	3310	2753	2603
4	1	2	1	2	1	2	2	2498	2977	2195	2902	2101	2399	2475	3310	2753	2603
4	1	2	1	2	2	1	1	2747	3005	2891	2857	3028	3252	3025	2972	2940	3007
4	1	2	1	2	2	1	2	2747	3005	2891	2857	3028	3252	3025	2972	2940	3007
4	1	2	1	2	2	2	1	3731	3782	3554	3607	3664	3822	4342	3833	3825	3895
4	1	2	1	2	2	2	2	3731	3782	3554	3607	3664	3822	4342	3833	3825	3895
4	1	2	2	1	1	1	1	2991	3190	2802	3010	2922	3268	3085	3170	3096	3295
4	1	2	2	1	1	1	2	2991	3190	2802	3010	2922	3268	3085	3170	3096	3295
4	1	2	2	1	1	2	1	3558	3430	3054	3625	3198	3517	3559	3566	3564	3754
4	1	2	2	1	1	2	2	3558	3430	3054	3625	3198	3517	3559	3566	3564	3754
4	1	2	2	1	2	1	1	4512	4572	4015	4661	4280	4457	4607	5167	4474	4423
4	1	2	2	1	2	1	2	4512	4572	4015	4661	4280	4457	4607	5167	4474	4423
4	1	2	2	1	2	2	1	5073	4815	4282	5150	4616	4895	5087	5668	4777	4822
4	1	2	2	1	2	2	2	5073	4815	4282	5150	4616	4895	5087	5668	4777	4822
4	1	2	2	2	1	1	1	3277	2934	3118	2613	2612	2448	2212	2973	2573	3269
4	1	2	2	2	1	1	2	3273	2932	3118	2613	2612	2448	2212	2973	2573	3269
4	1	2	2	2	1	2	1	5517	5116	5266	4071	4250	3873	3334	5106	3548	5636
4	1	2	2	2	1	2	2	5517	5116	5266	4071	4250	3873	3334	5106	3548	5636
4	1	2	2	2	2	1	1	4356	3608	3779	3493	3673	3445	3454	4174	3578	4518
4	1	2	2	2	2	1	2	4356	3608	3779	3493	3673	3445	3454	4174	3578	4518
4	1	2	2	2	2	2	1	5934	4991	4895	4333	4636	4735	4360	5668	4619	5967
4	1	2	2	2	2	2	2	5934	4991	4895	4333	4636	4735	4360	5668	4619	5967
4	2	1	1	1	1	1	1	345	135	369	312	119	330	237	185	291	194
4	2	1	1	1	1	1	2	345	135	369	312	119	330	237	185	291	194

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
4	2	1	1	1	1	2	1	408	150	456	357	119	396	297	194	321	206
4	2	1	1	1	1	2	2	408	150	456	357	119	396	297	194	321	206
4	2	1	1	1	2	1	1	533	444	509	516	473	601	497	609	557	501
4	2	1	1	1	2	1	2	533	444	509	516	473	601	497	609	557	501
4	2	1	1	1	2	2	1	554	468	539	546	497	625	515	651	587	525
4	2	1	1	1	2	2	2	554	468	539	546	497	625	515	651	587	525
4	2	1	1	2	1	1	1	565	550	598	520	580	539	539	554	601	541
4	2	1	1	2	1	1	2	565	550	596	518	580	531	539	554	601	541
4	2	1	1	2	1	2	1	1138	1108	1202	1046	1168	1080	1088	1124	1210	1090
4	2	1	1	2	1	2	2	1138	1108	1202	1046	1168	1080	1088	1124	1210	1090
4	2	1	1	2	2	1	1	582	477	628	562	616	569	544	597	597	515
4	2	1	1	2	2	1	2	582	477	628	562	616	569	544	597	597	515
4	2	1	1	2	2	2	1	831	669	898	766	874	905	730	864	864	671
4	2	1	1	2	2	2	2	831	669	898	766	874	905	730	864	864	671
4	2	1	2	1	1	1	1	404	357	369	292	319	307	321	345	437	479
4	2	1	2	1	1	1	2	404	357	369	292	319	307	321	345	437	479
4	2	1	2	1	1	2	1	608	507	507	397	466	439	465	516	584	677
4	2	1	2	1	1	2	2	608	507	507	397	466	439	465	516	584	677
4	2	1	2	1	2	1	1	663	606	660	527	587	618	569	648	679	558
4	2	1	2	1	2	1	2	663	606	660	527	587	618	569	648	679	558
4	2	1	2	1	2	2	1	822	651	756	563	695	669	653	792	718	666
4	2	1	2	1	2	2	2	822	651	756	563	695	669	653	792	718	666
4	2	1	2	2	1	1	1	786	514	558	553	648	601	784	560	738	775
4	2	1	2	2	1	1	2	786	514	558	553	646	601	784	560	738	775
4	2	1	2	2	1	2	1	1623	928	1104	1123	1312	1189	1468	1145	1470	1570
4	2	1	2	2	1	2	2	1623	928	1104	1123	1312	1189	1468	1145	1470	1570
4	2	1	2	2	2	1	1	836	738	622	577	603	754	927	620	765	759
4	2	1	2	2	2	1	2	828	714	622	577	603	754	927	620	765	759
4	2	1	2	2	2	2	1	1434	1095	940	811	846	1180	1503	866	1236	1218
4	2	1	2	2	2	2	2	1434	1095	940	811	846	1180	1503	866	1236	1218

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
4	2	2	1	1	1	1	1	1986	2053	2707	2335	2610	2641	2578	2424	2509	2239
4	2	2	1	1	1	1	2	1972	2039	2705	2331	2610	2641	2578	2424	2509	2239
4	2	2	1	1	1	2	1	4189	4307	6212	5262	5901	5785	5824	5265	5686	4906
4	2	2	1	1	1	2	2	4189	4307	6212	5262	5901	5785	5824	5265	5686	4906
4	2	2	1	1	2	1	1	2436	2473	2514	2651	2558	2564	2649	2521	2660	2659
4	2	2	1	1	2	1	2	2436	2473	2514	2651	2558	2564	2649	2521	2660	2659
4	2	2	1	1	2	2	1	2514	2626	2664	2810	2684	2708	2859	2644	2816	2779
4	2	2	1	1	2	2	2	2514	2626	2664	2810	2684	2708	2859	2644	2816	2779
4	2	2	1	2	1	1	1	3619	3579	3066	3649	3061	3043	3116	3746	3538	3160
4	2	2	1	2	1	1	2	3619	3575	3066	3649	3061	3043	3116	3746	3538	3160
4	2	2	1	2	1	2	1	8827	8729	7413	8932	7411	7375	7505	9071	8647	7603
4	2	2	1	2	1	2	2	8827	8729	7413	8932	7411	7375	7505	9071	8647	7603
4	2	2	1	2	2	1	1	2045	1912	1974	1836	1708	1928	1890	2076	1842	2027
4	2	2	1	2	2	1	2	2085	1912	1974	1828	1708	1928	1890	2076	1842	2027
4	2	2	1	2	2	2	1	3540	2824	2577	2992	2077	2501	2658	3546	2484	2699
4	2	2	1	2	2	2	2	3540	2824	2577	2992	2077	2501	2658	3546	2484	2699
4	2	2	2	1	1	1	1	2972	2583	3103	3380	2472	2765	2831	3470	2577	2858
4	2	2	2	1	1	1	2	2962	2565	3099	3380	2470	2765	2831	3470	2577	2858
4	2	2	2	1	1	2	1	6607	5250	6558	6953	5326	5951	5789	7628	5370	6203
4	2	2	2	1	1	2	2	6607	5250	6558	6953	5326	5951	5789	7628	5370	6203
4	2	2	2	1	2	1	1	3068	3134	3042	3055	2509	2813	2930	3363	3437	3338
4	2	2	2	1	2	1	2	3068	3134	3042	3055	2509	2813	2930	3363	3437	3338
4	2	2	2	1	2	2	1	3812	3500	3510	3631	2947	3203	3563	3993	4067	3896
4	2	2	2	1	2	2	2	3812	3500	3510	3631	2947	3203	3563	3993	4067	3896
4	2	2	2	2	1	1	1	5346	5498	5351	4785	5230	4793	5278	5386	5499	5152
4	2	2	2	2	1	1	2	5314	5482	5337	4785	5214	4793	5278	5362	5499	5152
4	2	2	2	2	1	2	1	12457	13009	12513	11427	12465	11405	12784	12646	13218	11932
4	2	2	2	2	1	2	2	12457	13009	12513	11427	12465	11405	12784	12646	13218	11932
4	2	2	2	2	2	1	1	3481	3387	3287	2796	3128	3181	2838	3169	2899	3922
4	2	2	2	2	2	1	2	3481	3387	3287	2796	3128	3181	2838	3169	2899	3922

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
4	2	2	2	2	2	2	1	6046	6240	5744	4752	5807	5146	5229	5161	4999	7405
4	2	2	2	2	2	2	2	6046	6240	5744	4752	5807	5146	5229	5161	4999	7405
5	1	1	1	1	1	1	1	419	415	436	427	412	415	400	433	442	427
5	1	1	1	1	1	1	2	419	415	436	427	412	415	400	433	442	427
5	1	1	1	1	1	2	1	545	538	580	562	532	538	508	574	592	562
5	1	1	1	1	1	2	2	545	538	580	562	532	538	508	574	592	562
5	1	1	1	1	2	1	1	467	463	484	475	460	463	448	481	490	475
5	1	1	1	1	2	1	2	467	463	484	475	460	463	448	481	490	475
5	1	1	1	1	2	2	1	593	586	628	610	580	586	556	622	640	610
5	1	1	1	1	2	2	2	593	586	628	610	580	586	556	622	640	610
5	1	1	1	2	1	1	1	703	692	721	700	733	742	694	679	700	736
5	1	1	1	2	1	1	2	703	692	721	700	733	742	694	679	700	736
5	1	1	1	2	1	2	1	1195	1172	1234	1192	1258	1273	1180	1150	1192	1261
5	1	1	1	2	1	2	2	1195	1172	1234	1192	1258	1273	1180	1150	1192	1261
5	1	1	1	2	2	1	1	751	740	769	748	781	790	742	727	748	784
5	1	1	1	2	2	1	2	751	740	769	748	781	790	742	727	748	784
5	1	1	1	2	2	2	1	1243	1220	1282	1240	1306	1321	1228	1198	1240	1309
5	1	1	1	2	2	2	2	1243	1220	1282	1240	1306	1321	1228	1198	1240	1309
5	1	1	2	1	1	1	1	496	388	476	467	500	407	486	436	427	476
5	1	1	2	1	1	1	2	496	388	476	467	500	407	486	436	427	476
5	1	1	2	1	1	2	1	700	484	659	638	707	521	675	580	562	659
5	1	1	2	1	1	2	2	700	484	659	638	707	521	675	580	562	659
5	1	1	2	1	2	1	1	544	436	524	515	548	455	534	484	475	524
5	1	1	2	1	2	1	2	544	436	524	515	548	455	534	484	475	524
5	1	1	2	1	2	2	1	748	532	707	686	755	569	723	628	610	707
5	1	1	2	1	2	2	2	748	532	707	686	755	569	723	628	610	707
5	1	1	2	2	1	1	1	927	618	568	693	626	683	805	717	743	875
5	1	1	2	2	1	1	2	927	618	568	693	626	683	805	717	743	875
5	1	1	2	2	1	2	1	1647	1029	928	1173	1040	1160	1402	1227	1274	1538
5	1	1	2	2	1	2	2	1647	1029	928	1173	1040	1160	1402	1227	1274	1538

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
5	1	1	2	2	2	1	1	975	666	616	741	674	731	853	765	791	923
5	1	1	2	2	2	1	2	975	666	616	741	674	731	853	765	791	923
5	1	1	2	2	2	2	1	1695	1077	976	1221	1088	1208	1450	1275	1322	1586
5	1	1	2	2	2	2	2	1695	1077	976	1221	1088	1208	1450	1275	1322	1586
5	1	2	1	1	1	1	1	3989	4057	4011	4094	3956	4000	3950	3985	4070	4033
5	1	2	1	1	1	1	2	3989	4057	4011	4094	3956	4000	3950	3985	4070	4033
5	1	2	1	1	1	2	1	5429	5563	5481	5639	5366	5437	5354	5422	5591	5518
5	1	2	1	1	1	2	2	5429	5563	5481	5639	5366	5437	5354	5422	5591	5518
5	1	2	1	1	2	1	1	4121	4189	4143	4226	4088	4132	4082	4117	4202	4165
5	1	2	1	1	2	1	2	4121	4189	4143	4226	4088	4132	4082	4117	4202	4165
5	1	2	1	1	2	2	1	5561	5695	5613	5771	5498	5569	5486	5554	5723	5650
5	1	2	1	1	2	2	2	5561	5695	5613	5771	5498	5569	5486	5554	5723	5650
5	1	2	1	2	1	1	1	6538	6498	6454	6480	6568	6310	6619	6620	6409	6474
5	1	2	1	2	1	1	2	6538	6498	6454	6480	6568	6310	6619	6620	6409	6474
5	1	2	1	2	1	2	1	11293	11226	11134	11190	11362	10852	11467	11474	11041	11175
5	1	2	1	2	1	2	2	11293	11226	11134	11190	11362	10852	11467	11474	11041	11175
5	1	2	1	2	2	1	1	6670	6630	6586	6612	6700	6442	6751	6752	6541	6606
5	1	2	1	2	2	1	2	6670	6630	6586	6612	6700	6442	6751	6752	6541	6606
5	1	2	1	2	2	2	1	11425	11358	11266	11322	11494	10984	11599	11606	11173	11307
5	1	2	1	2	2	2	2	11425	11358	11266	11322	11494	10984	11599	11606	11173	11307
5	1	2	2	1	1	1	1	4268	3888	3919	4044	4120	3996	4216	4486	4005	4239
5	1	2	2	1	1	1	2	4268	3888	3919	4044	4120	3996	4216	4486	4005	4239
5	1	2	2	1	1	2	1	5993	5226	5281	5541	5695	5442	5887	6421	5460	5925
5	1	2	2	1	1	2	2	5993	5226	5281	5541	5695	5442	5887	6421	5460	5925
5	1	2	2	1	2	1	1	4400	4020	4051	4176	4252	4128	4348	4618	4137	4371
5	1	2	2	1	2	1	2	4400	4020	4051	4176	4252	4128	4348	4618	4137	4371
5	1	2	2	1	2	2	1	6125	5358	5413	5673	5827	5574	6019	6553	5592	6057
5	1	2	2	1	2	2	2	6125	5358	5413	5673	5827	5574	6019	6553	5592	6057
5	1	2	2	2	1	1	1	6604	5907	6318	6473	6187	6237	6367	6969	6541	6832
5	1	2	2	2	1	1	2	6604	5907	6318	6473	6187	6237	6367	6969	6541	6832

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
5	1	2	2	2	1	2	1	11440	10044	10857	11177	10597	10689	10951	12159	11293	11902
5	1	2	2	2	1	2	2	11440	10044	10857	11177	10597	10689	10951	12159	11293	11902
5	1	2	2	2	2	1	1	6736	6039	6450	6605	6319	6369	6499	7101	6673	6964
5	1	2	2	2	2	1	2	6736	6039	6450	6605	6319	6369	6499	7101	6673	6964
5	1	2	2	2	2	2	1	11572	10176	10989	11309	10729	10821	11083	12291	11425	12034
5	1	2	2	2	2	2	2	11572	10176	10989	11309	10729	10821	11083	12291	11425	12034
5	2	1	1	1	1	1	1	419	415	436	427	412	415	400	433	442	427
5	2	1	1	1	1	1	2	419	415	436	427	412	415	400	433	442	427
5	2	1	1	1	1	2	1	545	538	580	562	532	538	508	574	592	562
5	2	1	1	1	1	2	2	545	538	580	562	532	538	508	574	592	562
5	2	1	1	1	2	1	1	467	463	484	475	460	463	448	481	490	475
5	2	1	1	1	2	1	2	467	463	484	475	460	463	448	481	490	475
5	2	1	1	1	2	2	1	593	586	628	610	580	586	556	622	640	610
5	2	1	1	1	2	2	2	593	586	628	610	580	586	556	622	640	610
5	2	1	1	2	1	1	1	703	692	721	700	733	742	694	679	700	736
5	2	1	1	2	1	1	2	703	692	721	700	733	742	694	679	700	736
5	2	1	1	2	1	2	1	1195	1172	1234	1192	1258	1273	1180	1150	1192	1261
5	2	1	1	2	1	2	2	1195	1172	1234	1192	1258	1273	1180	1150	1192	1261
5	2	1	1	2	2	1	1	751	740	769	748	781	790	742	727	748	784
5	2	1	1	2	2	1	2	751	740	769	748	781	790	742	727	748	784
5	2	1	1	2	2	2	1	1243	1220	1282	1240	1306	1321	1228	1198	1240	1309
5	2	1	1	2	2	2	2	1243	1220	1282	1240	1306	1321	1228	1198	1240	1309
5	2	1	2	1	1	1	1	496	388	476	467	500	407	486	436	427	476
5	2	1	2	1	1	1	2	496	388	476	467	500	407	486	436	427	476
5	2	1	2	1	1	2	1	700	484	659	638	707	521	675	580	562	659
5	2	1	2	1	1	2	2	700	484	659	638	707	521	675	580	562	659
5	2	1	2	1	2	1	1	544	436	524	515	548	455	534	484	475	524
5	2	1	2	1	2	1	2	544	436	524	515	548	455	534	484	475	524
5	2	1	2	1	2	2	1	748	532	707	686	755	569	723	628	610	707
5	2	1	2	1	2	2	2	748	532	707	686	755	569	723	628	610	707

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
5	2	1	2	2	1	1	1	927	618	568	693	626	683	805	717	743	875
5	2	1	2	2	1	1	2	927	618	568	693	626	683	805	717	743	875
5	2	1	2	2	1	2	1	1647	1029	928	1173	1040	1160	1402	1227	1274	1538
5	2	1	2	2	1	2	2	1647	1029	928	1173	1040	1160	1402	1227	1274	1538
5	2	1	2	2	2	1	1	975	666	616	741	674	731	853	765	791	923
5	2	1	2	2	2	1	2	975	666	616	741	674	731	853	765	791	923
5	2	1	2	2	2	2	1	1695	1077	976	1221	1088	1208	1450	1275	1322	1586
5	2	1	2	2	2	2	2	1695	1077	976	1221	1088	1208	1450	1275	1322	1586
5	2	2	1	1	1	1	1	3989	4057	4011	4094	3956	4000	3950	3985	4070	4033
5	2	2	1	1	1	1	2	3989	4057	4011	4094	3956	4000	3950	3985	4070	4033
5	2	2	1	1	1	2	1	5429	5563	5481	5639	5366	5437	5354	5422	5591	5518
5	2	2	1	1	1	2	2	5429	5563	5481	5639	5366	5437	5354	5422	5591	5518
5	2	2	1	1	2	1	1	4121	4189	4143	4226	4088	4132	4082	4117	4202	4165
5	2	2	1	1	2	1	2	4121	4189	4143	4226	4088	4132	4082	4117	4202	4165
5	2	2	1	1	2	2	1	5561	5695	5613	5771	5498	5569	5486	5554	5723	5650
5	2	2	1	1	2	2	2	5561	5695	5613	5771	5498	5569	5486	5554	5723	5650
5	2	2	1	2	1	1	1	6538	6498	6454	6480	6568	6310	6619	6620	6409	6474
5	2	2	1	2	1	1	2	6538	6498	6454	6480	6568	6310	6619	6620	6409	6474
5	2	2	1	2	1	2	1	11293	11226	11134	11190	11362	10852	11467	11474	11041	11175
5	2	2	1	2	1	2	2	11293	11226	11134	11190	11362	10852	11467	11474	11041	11175
5	2	2	1	2	2	1	1	6670	6630	6586	6612	6700	6442	6751	6752	6541	6606
5	2	2	1	2	2	1	2	6670	6630	6586	6612	6700	6442	6751	6752	6541	6606
5	2	2	1	2	2	2	1	11425	11358	11266	11322	11494	10984	11599	11606	11173	11307
5	2	2	1	2	2	2	2	11425	11358	11266	11322	11494	10984	11599	11606	11173	11307
5	2	2	2	1	1	1	1	4268	3888	3919	4044	4120	3996	4216	4486	4005	4239
5	2	2	2	1	1	1	2	4268	3888	3919	4044	4120	3996	4216	4486	4005	4239
5	2	2	2	1	1	2	1	5993	5226	5281	5541	5695	5442	5887	6421	5460	5925
5	2	2	2	1	1	2	2	5993	5226	5281	5541	5695	5442	5887	6421	5460	5925
5	2	2	2	1	2	1	1	4400	4020	4051	4176	4252	4128	4348	4618	4137	4371
5	2	2	2	1	2	1	2	4400	4020	4051	4176	4252	4128	4348	4618	4137	4371

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

Table C.1 (Continued)

Factors								Replication									
A	B	C	D	E	F	G	H	1	2	3	4	5	6	7	8	9	10
5	2	2	2	1	2	2	1	6125	5358	5413	5673	5827	5574	6019	6553	5592	6057
5	2	2	2	1	2	2	2	6125	5358	5413	5673	5827	5574	6019	6553	5592	6057
5	2	2	2	2	1	1	1	6604	5907	6318	6473	6187	6237	6367	6969	6541	6832
5	2	2	2	2	1	1	2	6604	5907	6318	6473	6187	6237	6367	6969	6541	6832
5	2	2	2	2	1	2	1	11440	10044	10857	11177	10597	10689	10951	12159	11293	11902
5	2	2	2	2	1	2	2	11440	10044	10857	11177	10597	10689	10951	12159	11293	11902
5	2	2	2	2	2	1	1	6736	6039	6450	6605	6319	6369	6499	7101	6673	6964
5	2	2	2	2	2	1	2	6736	6039	6450	6605	6319	6369	6499	7101	6673	6964
5	2	2	2	2	2	2	1	11572	10176	10989	11309	10729	10821	11083	12291	11425	12034
5	2	2	2	2	2	2	2	11572	10176	10989	11309	10729	10821	11083	12291	11425	12034

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

APPENDIX D

RESULTS OF TUKEY TESTS FOR SIGNIFICANCE

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=1; e=1; f=1; g=1; h=1.

Alpha= 0.05 df= 89 MSE= 1.278904
Critical Value of Studentized Range= 4.590
Minimum Significant Difference= 1.6504
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.89011
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	7.2610	10	9
A	7.2610	10	10
A	6.9480	10	7
B	4.1770	10	8
B	3.4940	10	1
B	3.4078	9	3
B	2.9620	10	5
B	2.8430	10	4
B	2.8080	10	2
C	1.0000	10	6

c=1; d=1; e=1; f=1; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 1.400902
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.7173
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	7.2610	10	9
A	7.2610	10	10
A	6.9480	10	7
B	4.1770	10	8
B	3.8770	10	3
B	3.4180	10	1
B	3.2020	10	5
B	2.8430	10	4
B	2.7950	10	2
C	1.0000	10	6

c=1; d=1; e=1; f=1; g=2; h=1.

Alpha= 0.05 df= 89 MSE= 2.11628
Critical Value of Studentized Range= 4.590
Minimum Significant Difference= 2.1231
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.89011

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	9.7040	10	9
	A	9.7040	10	10
	B	7.4720	10	7
	C	4.8480	10	8
	C	3.9078	9	3
	C	3.5310	10	1
	C	3.2260	10	5
D	C	2.9440	10	4
D	C	2.8370	10	2
	D	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=1; e=1; f=1; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 2.134932
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 2.1201
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	9.7040	10	9
	A	9.7040	10	10
	B	7.4720	10	7
	C	4.8480	10	8
	C	4.2100	10	5
	C	4.0270	10	3
	C	3.4030	10	1
D	C	2.9440	10	4
D	C	2.8370	10	2
	D	1.0000	10	6

c=1; d=1; e=1; f=2; g=1; h=1.

Alpha= 0.05 df= 81 MSE= 0.08294
Critical Value of Studentized Range= 4.507
Minimum Significant Difference= 0.4105
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	2.4670	10	7
B	A	2.1380	10	2
B	A	2.1040	10	8
B	C	1.9690	10	1
B	C	1.8910	10	10
B	C	1.8910	10	9
B	C	1.8210	10	4
	C	1.6080	10	5
	D	1.0000	10	6

c=1; d=1; e=1; f=2; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 0.087763
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.4298
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	2.3770	10	7
B	A	2.0850	10	3
B	A	2.0270	10	8
B	A	2.0000	10	2
B	A	1.9480	10	1
	B	1.8230	10	10
	B	1.8230	10	9
	B	1.7160	10	4
	B	1.6870	10	5
	C	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=1; e=1; f=2; g=2; h=1.

Alpha= 0.05 df= 81 MSE= 0.095523
Critical Value of Studentized Range= 4.507
Minimum Significant Difference= 0.4405
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	2.6790	10	7
B	A	2.3450	10	10
B	A	2.3450	10	9
B	C	2.1530	10	8
B	C	2.0600	10	2
B	C	1.9740	10	1
	C	1.8490	10	4
	C	1.7460	10	5
	D	1.0000	10	6

c=1; d=1; e=1; f=2; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 0.114818
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.4917
Means with the same letter are not significantly different.

Tukey Grouping			Mean	Sample Size	Solution Approach
		A	2.6270	10	7
	B	A	2.4450	10	5
	B	A	2.4370	10	3
B	A	C	2.2990	10	10
B	A	C	2.2990	10	9
	B	C	2.1130	10	8
	B	C	2.0370	10	1
	B	C	2.0190	10	2
		C	1.8240	10	4
		D	1.0000	10	6

c=1; d=1; e=2; f=1; g=1; h=1.

Alpha= 0.05 df= 53 MSE= 2.258074
Critical Value of Studentized Range= 4.181
Minimum Significant Difference= 2.0052
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.818182
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	9.0880	10	9
A	9.0880	10	10
A	7.1420	10	8
B	4.9840	10	7
C	2.7356	9	5
C	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=1; e=2; f=1; g=1; h=2.

Alpha= 0.05 df= 63 MSE= 2.016383
Critical Value of Studentized Range= 4.307
Minimum Significant Difference= 1.9341
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	9.0880	10	10
	A	9.0880	10	9
	B	7.1240	10	8
	C	4.9840	10	7
	C	4.1210	10	3
	C	3.2340	10	5
	D	1.0000	10	6

c=1; d=1; e=2; f=1; g=2; h=1.

Alpha= 0.05 df= 54 MSE= 6.008689
Critical Value of Studentized Range= 4.178
Minimum Significant Difference= 3.2388
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	13.623	10	9
	A	13.623	10	10
	A	12.594	10	8
	B	5.428	10	7
C	B	2.968	10	5
	C	1.000	10	6

c=1; d=1; e=2; f=1; g=2; h=2.

Alpha= 0.05 df= 63 MSE= 5.493531
Critical Value of Studentized Range= 4.307
Minimum Significant Difference= 3.1924
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	13.623	10	10
	A	13.623	10	9
	A	12.594	10	8
	B	5.428	10	7
	B	5.293	10	5
	B	5.253	10	3
	C	1.000	10	6

c=1; d=1; e=2; f=2; g=1; h=1.

Alpha= 0.05 df= 44 MSE= 0.129235
Critical Value of Studentized Range= 4.022
Minimum Significant Difference= 0.4623
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.782609
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	2.2330	10	10
	A	2.2330	10	9
B	A	2.0210	10	7
	B	1.6670	10	8
	C	1.0000	9	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=1; e=2; f=2; g=1; h=2.

Alpha= 0.05 df= 72 MSE= 0.093456
Critical Value of Studentized Range= 4.415
Minimum Significant Difference= 0.4268
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	2.3520	10	9
	A	2.3520	10	10
B	A	2.1650	10	3
B	A	2.1390	10	7
B	A	1.9290	10	4
	B	1.8030	10	5
	B	1.7610	10	8
	C	1.0000	10	6

c=1; d=1; e=2; f=2; g=2; h=1.

Alpha= 0.05 df= 45 MSE= 0.096517
Critical Value of Studentized Range= 4.018
Minimum Significant Difference= 0.3948
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	3.4540	10	10
A	3.4540	10	9
B	2.4460	10	7
B	2.2220	10	8
C	1.0000	10	6

c=1; d=1; e=2; f=2; g=2; h=2.

Alpha= 0.05 df= 69 MSE= 0.080494
Critical Value of Studentized Range= 4.421
Minimum Significant Difference= 0.4071
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.491525
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	3.0690	10	9
	A	3.0690	10	10
	B	2.4190	10	3
C	B	2.1760	10	7
C	B	2.0860	10	5
	C	1.9680	10	8
	C	1.9271	7	4
	D	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=2; e=1; f=1; g=1; h=1.

Alpha= 0.05 df= 90 MSE= 3.688227
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 2.7865
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	10.6290	10	7
A	10.1890	10	10
A	10.1890	10	9
A	8.1850	10	8
B	2.8740	10	3
B	2.7850	10	1
B	2.2130	10	4
B	2.1690	10	2
B	2.1380	10	5
B	1.0000	10	6

c=1; d=2; e=1; f=1; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 4.113345
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 2.9427
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	11.2870	10	7
A	10.8270	10	10
A	10.8270	10	9
A	8.7560	10	8
B	3.1280	10	3
B	2.8230	10	1
B	2.3260	10	4
B	2.3020	10	2
B	2.1670	10	5
B	1.0000	10	6

c=1; d=2; e=1; f=1; g=2; h=1.

Alpha= 0.05 df= 90 MSE= 6.275634
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 3.6348
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	13.783	10	9
A	13.783	10	10
A	12.070	10	7
A	11.613	10	8
B	2.976	10	3
B	2.545	10	1
B	2.435	10	5
B	2.219	10	4
B	2.162	10	2
B	1.000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=2; e=1; f=1; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 7.390149
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 3.9444
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	14.662	10	9
A	14.662	10	10
A	12.848	10	7
A	12.433	10	8
B	3.119	10	3
B	2.700	10	1
B	2.583	10	5
B	2.332	10	4
B	2.295	10	2
B	1.000	10	6

c=1; d=2; e=1; f=2; g=1; h=1.

Alpha= 0.05 df= 90 MSE= 0.562097
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.0878
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	3.6790	10	7
	A	3.1350	10	8
B	A	2.6250	10	9
B	A	2.6250	10	10
B	C	1.9900	10	3
B	C	1.9460	10	2
B	C	1.8850	10	4
B	C	1.8770	10	1
	C	1.4100	10	5
	C	1.0300	10	6

c=1; d=2; e=1; f=2; g=1; h=1.

Alpha= 0.05 df= 90 MSE= 0.844415
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.3333
Means with the same letter are not significantly different.

	Tukey Grouping		Mean	Sample Size	Solution Approach
		A	3.6680	10	7
	B	A	3.1200	10	8
B	A	C	2.6250	10	9
B	A	C	2.6250	10	10
B	D	C	2.1740	10	3
B	D	C	1.9610	10	2
B	D	C	1.9550	10	1
B	D	C	1.8650	10	4
	D	C	1.6440	10	5
		D	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=2; e=1; f=2; g=2; h=1.

Alpha= 0.05 df= 88 MSE= 0.807384
Critical Value of Studentized Range= 4.591
Minimum Significant Difference= 1.3207
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.756098
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.1700	10	7
	A	3.5390	10	8
	A	3.4580	10	9
	A	3.4580	10	10
	B	1.9537	8	3
	B	1.9040	10	2
	B	1.8710	10	1
	B	1.8710	10	4
	B	1.7100	10	5
	B	1.0000	10	6

c=1; d=2; e=1; f=2; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 0.646059
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.1662
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.3230	10	7
	A	3.6790	10	8
	A	3.5780	10	9
	A	3.5780	10	10
	B	2.3580	10	3
	B	2.2020	10	5
C	B	1.9850	10	2
C	B	1.9470	10	1
C	B	1.9410	10	4
	C	1.0000	10	6

c=1; d=2; e=2; f=1; g=1; h=1.

Alpha= 0.05 df= 77 MSE= 4.184745
Critical Value of Studentized Range= 4.514
Minimum Significant Difference= 2.9956
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.501466
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	13.5370	10	10
	A	13.5370	10	9
B	A	11.9770	10	8
	B	9.4100	10	7
	C	2.1956	9	5
	C	1.9980	10	4
	C	1.8650	8	1
	C	1.8322	9	2
	C	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=2; e=2; f=1; g=1; h=2.

Alpha= 0.05 df= 89 MSE= 3.954504
Critical Value of Studentized Range= 4.590
Minimum Significant Difference= 2.9022
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.89011
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	13.8130	10	9
	A	13.8130	10	10
B	A	12.2160	10	8
	B	9.5720	10	7
	C	3.3900	10	5
	C	3.1722	9	3
	C	2.1000	10	4
	C	2.0430	10	1
	C	2.0350	10	2
	C	1.0000	10	6

c=1; d=2; e=2; f=1; g=2; h=1.

Alpha= 0.05 df= 76 MSE= 11.81509
Critical Value of Studentized Range= 4.515
Minimum Significant Difference= 5.0721
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.364162
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	23.240	10	8
A	22.484	10	9
A	22.484	10	10
B	13.461	10	7
C	2.456	8	5
C	2.045	10	4
C	2.004	8	1
C	1.854	9	2
C	1.000	10	6

c=1; d=2; e=2; f=1; g=2; h=2.

Alpha= 0.05 df= 89 MSE= 9.685818
Critical Value of Studentized Range= 4.590
Minimum Significant Difference= 4.542
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.89011

Tukey Grouping	Mean	Sample Size	Solution Approach
A	24.183	10	8
A	23.405	10	9
A	23.405	10	10
B	13.908	10	7
C	4.338	10	5
C	3.897	9	3
C	2.228	10	1
C	2.149	10	4
C	2.097	10	2
C	1.000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=2; e=2; f=2; g=1; h=1.

Alpha= 0.05 df= 64 MSE= 0.544886
Critical Value of Studentized Range= 4.540
Minimum Significant Difference= 1.5571
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 4.632353
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	3.1890	10	10
	A	3.1890	10	9
B	A	3.0290	10	7
B	A	2.9070	10	8
B	C	1.6229	7	1
B	C	1.5800	10	2
B	C	1.5400	1	5
	C	1.3420	5	4
	C	1.0000	10	6

c=1; d=2; e=2; f=2; g=1; h=2.

Alpha= 0.05 df= 89 MSE= 0.461552
Critical Value of Studentized Range= 4.590
Minimum Significant Difference= 0.9915
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.89011
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	3.1000	10	9
A	3.1000	10	10
A	2.9470	10	7
A	2.8150	10	8
B	1.8000	9	3
B	1.7070	10	1
B	1.6660	10	4
B	1.6220	10	5
B	1.5550	10	2
B	1.0000	10	6

c=1; d=2; e=2; f=2; g=2; h=1.

Alpha= 0.05 df= 66 MSE= 1.234238
Critical Value of Studentized Range= 4.535
Minimum Significant Difference= 1.8377
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 7.517401
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	4.9290	10	10
A	4.9290	10	9
A	4.1180	10	8
A	4.0210	10	7
B	1.4787	8	1
B	1.4467	9	2
B	1.4060	5	4
B	1.3725	4	5
B	1.0000	9	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=1; d=2; e=2; f=2; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 1.213862
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.5986
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.6280	10	9
	A	4.6280	10	10
B	A	3.8940	10	8
B	A	3.7830	10	7
B	C	2.4140	10	3
	C	2.1690	10	5
	C	1.5500	10	4
	C	1.4610	10	1
	C	1.3950	10	2
	C	1.0180	10	6

c=2; d=1; e=1; f=1; g=1; h=1.

Alpha= 0.05 df= 90 MSE= 0.099806
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.4584
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	10.8840	10	9
	A	10.8840	10	10
	B	7.6160	10	7
	C	6.5240	10	8
	D	2.5150	10	3
E	D	2.3820	10	5
E	D	2.3220	10	1
E	F	1.9620	10	2
	F	1.7810	10	4
	G	1.0000	10	6

c=2; d=1; e=1; f=1; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 0.124711
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.5124
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	10.9400	10	9
	A	10.9400	10	10
	B	7.6530	10	7
	C	6.5510	10	8
	D	2.8550	10	5
E	D	2.5220	10	3
E	F	2.2800	10	1
	F	1.9810	10	2
	F	1.7920	10	4
	G	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=1; e=1; f=1; g=2; h=1.

Alpha= 0.05 df= 87 MSE= 0.528572
Critical Value of Studentized Range= 4.592
Minimum Significant Difference= 1.0782
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.589041
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	14.9810	10	9
	A	14.9810	10	10
	A	14.5760	10	8
	B	7.9440	10	7
	C	2.8586	7	3
	C	2.8370	10	5
	C	2.4410	10	1
	C	2.1020	10	2
D	C	1.8020	10	4
	D	1.0000	10	6

c=2; d=1; e=1; f=1; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 0.584392
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.1092
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	15.0680	10	9
	A	15.0680	10	10
	A	14.6640	10	8
	B	7.9890	10	7
	C	3.7210	10	5
D	C	2.8770	10	3
	D	2.3440	10	1
D	E	2.0830	10	2
D	E	1.8140	10	4
	E	1.0000	10	6

c=2; d=1; e=1; f=2; g=1; h=1.

Alpha= 0.05 df= 83 MSE= 0.049005
Critical Value of Studentized Range= 4.598
Minimum Significant Difference= 0.3575
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 8.108108
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	2.9850	10	7
	A	2.8880	10	10
	A	2.8880	10	9
	B	1.7890	10	8
C	B	1.5970	10	2
C	B	1.5500	3	3
C	B	1.4750	10	1
	C	1.3760	10	5
C	D	1.3190	10	4
	D	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=1; e=1; f=2; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 0.048486
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.3195
Means with the same letter are not significantly different.

Tukey Grouping			Mean	Sample Size	Solution Approach
		A	2.99700	10	7
		A	2.90100	10	10
		A	2.90100	10	9
		B	1.79800	10	8
	C	B	1.66100	10	3
C	B	D	1.58000	10	2
C	B	D	1.56700	10	1
	C	D	1.44700	10	5
	E	D	1.31600	10	4
		E	1.00000	10	6

c=2; d=1; e=1; f=2; g=2; h=1.

Alpha= 0.05 df= 81 MSE= 0.051454
Critical Value of Studentized Range= 4.507
Minimum Significant Difference= 0.3233
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.0440	10	10
	A	4.0440	10	9
	B	3.2540	10	7
	C	1.9510	10	8
D	C	1.7200	10	2
D	C	1.6680	10	1
	D	1.5930	10	5
	D	1.4580	10	4
	E	1.0000	10	6

c=2; d=1; e=1; f=2; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 0.052913
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.3338
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.0350	10	9
	A	4.0350	10	10
	B	3.2490	10	7
	C	1.9480	10	8
	C	1.8930	10	5
	C	1.8610	10	3
D	C	1.7470	10	2
D	C	1.6290	10	1
	D	1.4550	10	4
	E	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=1; e=2; f=1; g=1; h=1.

Alpha= 0.05 df= 45 MSE= 0.55786
Critical Value of Studentized Range= 4.018
Minimum Significant Difference= 0.9491
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	13.1560	10	10
A	13.1560	10	9
B	6.8230	10	8
C	3.3870	10	7
D	1.0000	10	6

c=2; d=1; e=2; f=1; g=1; h=2.

Alpha= 0.05 df= 58 MSE= 0.470839
Critical Value of Studentized Range= 4.319
Minimum Significant Difference= 1.0019
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 8.75
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	13.0970	10	10
A	13.0970	10	9
B	6.7930	10	8
C	4.0480	5	3
C	3.3690	10	7
C	3.3510	10	5
D	1.0000	10	6

c=2; d=1; e=2; f=1; g=2; h=1.

Alpha= 0.05 df= 45 MSE= 2.380181
Critical Value of Studentized Range= 4.018
Minimum Significant Difference= 1.9605
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	21.0740	10	10
A	21.0740	10	9
B	15.3450	10	8
C	4.9370	10	7
D	1.0000	10	6

c=2; d=1; e=2; f=1; g=2; h=2.

Alpha= 0.05 df= 54 MSE= 2.202189
Critical Value of Studentized Range= 4.178
Minimum Significant Difference= 1.9608
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	21.0740	10	9
A	21.0740	10	10
B	15.3450	10	8
C	5.3650	10	5
C	4.9370	10	7
D	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=1; e=2; f=2; g=1; h=1.

Alpha= 0.05 df= 45 MSE= 0.017622
Critical Value of Studentized Range= 4.018
Minimum Significant Difference= 0.1687
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	3.75300	10	10
A	3.75300	10	9
B	1.68200	10	7
C	1.08800	10	8
C	1.01500	10	6

c=2; d=1; e=2; f=2; g=1; h=2.

Alpha= 0.05 df= 63 MSE= 0.012254
Critical Value of Studentized Range= 4.307
Minimum Significant Difference= 0.1508
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	3.72300	10	10
A	3.72300	10	9
B	2.12600	10	3
C	1.73100	10	5
C	1.66800	10	7
D	1.08100	10	8
D	1.01300	10	6

c=2; d=1; e=2; f=2; g=2; h=1.

Alpha= 0.05 df= 45 MSE= 0.093108
Critical Value of Studentized Range= 4.018
Minimum Significant Difference= 0.3877
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	5.9040	10	10
A	5.9040	10	9
B	1.9810	10	7
C	1.4540	10	8
D	1.0040	10	6

c=2; d=1; e=2; f=2; g=2; h=2.

Alpha= 0.05 df= 63 MSE= 0.08188
Critical Value of Studentized Range= 4.307
Minimum Significant Difference= 0.3897
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	5.8880	10	10
A	5.8880	10	9
B	3.0500	10	3
B	2.7310	10	5
C	1.9750	10	7
D	1.4420	10	8
E	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=2; e=1; f=1; g=1; h=1.

Alpha= 0.05 df= 90 MSE= 0.314511
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.8137
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	13.2710	10	9
	A	13.2710	10	10
	B	9.9320	10	7
	B	9.3430	10	8
	C	1.9430	10	5
	C	1.8550	10	3
D	C	1.6450	10	1
D	C	1.4180	10	2
D	C	1.4020	10	4
	D	1.0000	10	6

c=2; d=2; e=1; f=1; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 0.319858
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.8206
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	13.2710	10	9
	A	13.2710	10	10
	B	9.9320	10	7
	B	9.3320	10	8
	C	2.0590	10	5
	C	1.8790	10	3
D	C	1.6040	10	1
D	C	1.4180	10	2
D	C	1.4020	10	4
	D	1.0000	10	6

c=2; d=2; e=1; f=1; g=2; h=1.

Alpha= 0.05 df= 90 MSE= 1.181922
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.5774
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	19.8430	10	8
A	18.3230	10	9
A	18.3230	10	10
B	11.2220	10	7
C	2.2910	10	5
C	1.8940	10	3
C	1.6400	10	1
C	1.4200	10	2
C	1.4040	10	4
C	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=2; e=1; f=1; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 1.197751
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 1.588
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	19.8430	10	8
	A	18.3230	10	9
	A	18.3230	10	10
	B	11.2220	10	7
	C	2.6660	10	5
D	C	1.9110	10	3
D	C	1.6030	10	1
D	C	1.4200	10	2
D	C	1.4040	10	4
	D	1.0000	10	6

c=2; d=2; e=1; f=2; g=1; h=1.

Alpha= 0.05 df= 90 MSE= 0.065586
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.3716
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	3.7230	10	7
	A	3.4990	10	10
	A	3.4990	10	9
	B	2.5330	10	8
	C	1.5080	10	3
	C	1.4110	10	5
D	C	1.3680	10	2
D	C	1.3580	10	1
D	C	1.3080	10	4
	D	1.0000	10	6

c=2; d=2; e=1; f=2; g=1; h=2.

Alpha= 0.05 df= 90 MSE= 0.062049
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.3614
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	3.7580	10	7
	A	3.5300	10	10
	A	3.5300	10	9
	B	2.5560	10	8
	C	1.5390	10	3
	C	1.4650	10	1
	C	1.3920	10	2
D	C	1.3590	10	5
D	C	1.3070	10	4
	D	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=2; e=1; f=2; g=2; h=1.

Alpha= 0.05 df= 89 MSE= 0.073258
Critical Value of Studentized Range= 4.590
Minimum Significant Difference= 0.395
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.89011
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.7710	10	9
	A	4.7710	10	10
	B	4.0350	10	7
	C	2.9670	10	8
	D	1.5556	9	3
	D	1.4930	10	5
	D	1.4680	10	1
E	D	1.3930	10	2
E	D	1.3110	10	4
	E	1.0000	10	6

c=2; d=2; e=1; f=2; g=2; h=2.

Alpha= 0.05 df= 90 MSE= 0.062163
Critical Value of Studentized Range= 4.588
Minimum Significant Difference= 0.3618
Means with the same letter are not significantly different.

Tukey Grouping		Mean	Sample Size	Solution Approach
	A	4.7990	10	9
	A	4.7990	10	10
	B	4.0570	10	7
	C	2.9840	10	8
	D	1.6720	10	5
	D	1.6190	10	3
	D	1.4680	10	1
	D	1.4080	10	2
E	D	1.3140	10	4
	E	1.0000	10	6

c=2; d=2; e=2; f=1; g=1; h=1.

Alpha= 0.05 df= 44 MSE= 23.43193
Critical Value of Studentized Range= 4.022
Minimum Significant Difference= 6.225
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.782609
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	18.537	10	10
A	18.537	10	9
A	14.926	10	8
B	7.936	10	7
C	1.000	9	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=2; e=2; f=1; g=1; h=2.

Alpha= 0.05 df= 54 MSE= 4.875797
Critical Value of Studentized Range= 4.178
Minimum Significant Difference= 2.9176
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	19.9600	10	9
A	19.9600	10	10
B	16.0920	10	8
C	8.6230	10	7
D	3.9230	10	5
E	1.0000	10	6

c=2; d=2; e=2; f=1; g=2; h=1.

Alpha= 0.05 df= 45 MSE= 17.01174
Critical Value of Studentized Range= 4.018
Minimum Significant Difference= 5.2412
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	37.231	10	8
A	33.454	10	9
A	33.454	10	10
B	13.631	10	7
C	1.000	10	6

c=2; d=2; e=2; f=1; g=2; h=2.

Alpha= 0.05 df= 54 MSE= 20.09254
Critical Value of Studentized Range= 4.178
Minimum Significant Difference= 5.9226
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	37.088	10	8
A	33.301	10	9
A	33.301	10	10
B	13.547	10	7
C	5.880	10	5
C	1.000	10	6

c=2; d=2; e=2; f=2; g=1; h=1.

Alpha= 0.05 df= 41 MSE= 1.055287
Critical Value of Studentized Range= 4.035
Minimum Significant Difference= 1.3953
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 8.823529
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	3.7220	10	10
A	3.7220	10	9
B	2.1780	10	7
B	1.8590	10	8
B	1.0000	6	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

c=2; d=2; e=2; f=2; g=1; h=2.

Alpha= 0.05 df= 63 MSE= 0.086889
Critical Value of Studentized Range= 4.307
Minimum Significant Difference= 0.4015
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	5.0080	10	10
A	5.0080	10	9
B	2.8960	10	7
C	2.4410	10	8
C	2.2420	10	3
C	2.0530	10	5
D	1.0000	10	6

c=2; d=2; e=2; f=2; g=2; h=1.

Alpha= 0.05 df= 42 MSE= 4.570735
Critical Value of Studentized Range= 4.030
Minimum Significant Difference= 2.8391
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.210526
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	6.5120	10	10
A	6.5120	10	9
B	3.3180	10	8
B	2.9500	10	7
B	1.0000	7	6

c=2; d=2; e=2; f=2; g=2; h=2.

Alpha= 0.05 df= 62 MSE= 0.519292
Critical Value of Studentized Range= 4.309
Minimum Significant Difference= 0.9898
WARNING: Cell sizes are not equal.
Harmonic Mean of cell sizes= 9.84375
Means with the same letter are not significantly different.

Tukey Grouping	Mean	Sample Size	Solution Approach
A	8.0390	10	10
A	8.0390	10	9
B	4.0310	10	8
B	3.5690	10	7
B	3.3011	9	3
B	3.2260	10	5
C	1.0000	10	6

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.

AUTOBIOGRAPHICAL STATEMENT

Padmanabhan Soundar was born on March 31, 1968 in Madras, India. He received a Bachelor of Technology in Mechanical Enginering from the Indian Institute of Technology at Madras in 1989 and a Master of Science in Industrial Engineering from the University of Missouri at Columbia in 1991.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.