Sequencing Orders of Multiple Sized Stock Sheets

by

Jinsong Rao

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree of

MASTER OF SCIENCE

in Mechanical Engineering

Department of Mechanical and Industrial Engineering
University of Manitoba
Winnipeg, Manitoba

0-612-23467-3

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES
*****
COPYRIGHT PERMISSION PAGE

SEQUENCING ORDERS OF MULTIPLE SIZED STOCK SHEETS

BY

JINSONG RAO

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirements of the degree

of

MASTER OF SCIENCE

Abstract

This thesis studies the problem of two-dimensional, multiple sized stock sheets cutting. It is concerned primarily with the problem of how to sequence orders of multiple sized stock sheets in order to minimize the total trim-loss for a given bill of material. Existing optimal and heuristic procedures do not balance well the conflicting requirements of efficient stock sheet utilization and minimal computational effort. A critical study of these procedures, however, leads to a new procedure which is based on a single stock sheet layout approach. The procedure is designed for an IBM compatible microcomputer. It is compared with three of the better existing procedures using test data created by a random problem generator. Results show that the new procedure performs relatively well.

Acknowledgments

I would like to express my deep sense of gratitude and appreciation to my co-advisor, Dr. Neil Popplewell, for his excellent guidance, constructive criticisms, valuable suggestions and constant encouragement during the research and preparation of this thesis, and my entire program at the University of Manitoba. I would like to equally acknowledge my co-advisor, Dr. S. Balakrishnan, for his generous advice, support, and many valuable suggestions.

Sincere thanks are extended to Dr. A.D. Woodbury, who is on my examining committee, for his time, effort and suggestions. I would like to thank Mr. A.W. El-Bouri for his help and suggestions, particularly in computer programming. My thanks also go to the friendly technicians, B. Barrett and K. Tarte, in the Department of Mechanical and Industrial Engineering, for their help and assistance.

I wish to express my appreciation to my parents for their understanding and support. Finally, I wish to thank my wife, Danhong, for her continuous support, understanding, patience and encouragement in all ways.

TABLE OF CONTENTS

CHAPTER **PAGE**

CHAPTER	PAGE

CHAPTER **PAGE**

LIST OF FIGURES

FIGURE **PAGE**

LIST OF TABLES

TABLE **PAGE**

Nomenclature

Latin Letters

m	number of different order piece sizes
n	number of different stock sheet sizes
D	total number of order pieces
d	quantity of single sized order pieces
Q	total number of stock sheets
q	quantity of single sized stock sheets
L	length of a stock sheet
W	width of a stock sheet
M	number of stock sheets used to finish the BOM by following a certain sequence

Subscripts

i	stock sheet size
k	order piece size
j	level of a simulated tree
p	layout pattern

Miscellaneous

APSA	Average Piece-to-Stock Area ratio
ASSU	Average (percentage) Stock Sheet Utilization
B-B	Branch and Bound
BOM	Bill of Material
BSS	Basic Stock (sheet) Size
CPU	Central Process Unit
CPTL	Cumulative Percentage Trim-Loss
CSP	Cutting Stock Problem
CTL	Cumulative Trim-Loss
ID	a string of letters or digits identifying the node of a simulated tree
LAM	Largest Area Method
LTM	Least (total) Trim-loss Method
LTN	List of Terminal Notes
MU	Memory Unit that defines the space required to store all the information associated with a node during the sequence selection.
OH	Optimization Homotopy
PSTL	Pre-Specified Trim-Loss
PTL	Percentage Trim-Loss
SBH	Search-Based Heuristic

SQ	a sequential number specifying the order of the sheet node in the layout process
TL	Trim-Loss

Chapter 1. Introduction

How to economically cut a bill of material (BOM) from stock inventory is a common problem for manufacturers in the metal, leather, glass, electronic, shipbuilding, and lumber industries. In such industries, materials are usually ordered or produced in standard sizes and cut into the smaller pieces actually demanded. As the demanded sizes must not be greater than the standard sizes, material wastage can be reasonably expected. To minimize the cost caused by the resulting trim-loss, a company must decide, first, the standard sizes to be ordered (the so called assortment problem) and, second, how to cut the required sizes from the standard sizes so that wastage is not undesirably high (the trim-loss problem).

There are many characteristics that can be used to classify a Cutting Stock Problem (CSP). The most important one is dimensionality. There are a multiple of one, two, or three-dimensional problems that are similar in nature. Examples include one-dimensional problems encountered in pipe cutting, two-dimensional problems in sheet metal cutting, and three-dimensional problems in container loading. The problem considered in this research is a two-dimensional case in which a bill of material (BOM) containing rectangular pieces, known as order pieces, is to be cut from larger rectangular pieces, known as stock sheets. The task is to find an efficient way of cutting the order pieces from the stock sheets so that the trim-loss is minimized.

A two-dimensional CSP may be classified into either a single or a multiple stock sheet size problem depending upon the number of different stock sheet sizes available. It can also be classified into an assortment or a trim-loss problem according to the certainty of the stock sheets' dimensions. Each problem has its own particularities.

1.1 Single Stock Sheet Size Problem

A single stock sheet size problem involves stock sheets which are identical in shape and have the same dimensions. Probably it is the most investigated two-dimensional problem. Most researchers have tried to find a "good" layout pattern to minimize the trim-loss. Various approaches have been developed [1-6]. Their output would generally meet certain criteria and take the form of different patterns in addition to the number of repeats of these patterns. A pattern can be specified by the type of order pieces, the corresponding quantities and the placement in a stock sheet.

It is a common situation in manufacturing that there is more than one stock sheet size available. Therefore, the single stock sheet size problem can be considered to be the first step in solving the more complicated problem of multiple stock sheet sizes.

1.2 Multiple Stock Sheet Size Problem

A multiple stock sheet size problem refers to a BOM containing various sized order pieces that have to be cut from a number of different sized stock sheets. This problem is not simple because the layout problem has been demonstrated to be NP-complete [34], even for a single stock sheet when the order pieces are all rectangular. It is even more complicated when the BOM requires several sheets and there are multiple stock sheet sizes available. However, the objective of a multiple stock sheet size problem is still to find the best combination of all the available stock sheets to cut order pieces and minimize the trim-loss.

Two important aspects, the layout pattern on a single sheet and the sequence of selecting multiple sized stock sheets, need to be taken into account. The layout pattern on a single sheet is the same as that for the single stock size problem. That is, a procedure is developed to achieve a "good" pattern layout that produces minimal trim-loss on a single sheet. Then the procedure is repeated on additional sheets until the BOM is exhausted. After the layout pattern has been determined, the procedure to select the sequence of these stock sheets is yet another problem. Such a problem obviously does not exist for a single stock sheet size. For multiple stock sheet sizes, on the other hand, the sequence of the selected stock sheet sizes can have a considerable impact on the trim-loss.

The problems discussed earlier are all based on the assumption that the stock sheet sizes are known. However, in many situations, stock sheet sizes are not given or they can be supplied in numerous numbers. Then the problem becomes to determine, for a single known BOM, the best combination of stock sizes that are in inventory. This problem includes determining the dimensions of the inventoried stock sheets as well as the number of unique sizes to stock. The former problem is called a trim-loss problem, the latter is termed an assortment problem.

1.3 Assortment and Trim-loss Problem

The assortment problem may be illustrated by considering the following situation. A given BOM is to be cut from a set of stock sheets that have various sizes and quantities. Because of storage or manufacturing limitations, the economics of scale in production and storage or the cost of holding different stock sizes, a subset of the stock sheet sizes is to be stocked. Demands for an unstocked size are filled from a stocked size with an associated substitution cost. The problem is to determine the particular rules to follow in order to minimize the sum of all the relevant costs. On the other hand, the trim-loss problem deals with the way of cutting a BOM from a set of known stock sheets so as to minimize the trim-loss.

Although the assortment and trim-loss problems are different in nature, they can be connected by taking the trim-loss problem as a further step of the assortment problem. That is, once the stock sheet sizes have been determined by solving the assortment problem, the trim-loss can be reduced further by solving the trim-loss problem.

1.4 Problem Constraints and Formulation

Many constraints may be imposed on the way in which the problem can be formulated when dealing with the two-dimensional CSP. The most obvious constraint is that all the patterns allocated to a stock sheet must not overlap and they must lie entirely within the stock sheet. Other constraints may include the following requirements.

1. Cuts should be orthogonal and such that any cut has to be parallel to a side of the rectangular stock sheet.
2. A cut should be guillotined so that order pieces have to be cut in a way that each sequential cut can be performed from one edge to the opposite edge of a sheet. This constraint often arises because of limitations in the cutting machinery used in industries associated with glass and furniture. However, such constraints may not be necessary with the advent of advanced laser cutting techniques.

3. The number of each type of order piece appearing in a layout pattern should be limited. Indeed, the exact number demanded in the BOM must be met and an oversupply is not allowed.

4. The number of unique stock sheet sizes and the number of sheets of each unique size should be limited because of storage or manufacturing limitations.

The generalized form of the two-dimensional CSP can be formulated as (Carnieri [1]):

minimize $$\sum_{i=1}^{n} \sum_{p \in P(i)} C_i x_p^i \qquad (1.1)$$

subject to $$\sum_{i=1}^{n} \sum_{p \in P(i)} a_{kp}^i x_p^i \le d_k \qquad p \in P(i),\ i=1,\ldots,n,\ k=1,\ldots,m, \qquad (1.2)$$

$$\sum_{p \in P(i)} x_p^i \le q_i \qquad p \in P(i),\ i=1,\ldots,n, \qquad (1.3)$$

and $$x_p^i \ge 0 \qquad p \in P(i),\ i=1,\ldots,n. \qquad (1.4)$$

Here

n = number of different stock sheet sizes available

m = number of different order piece sizes

C_i = a cost associated with the stock sheet having size i

x_p^i = the number of stock sheets of size i that follow cutting pattern p, $p \in P(i)$, $i=1,\ldots,n$

$P(i)$ = the set of all cutting patterns for a stock sheet of size i ($i=1,\ldots,n$)

a^i_{kp} = the number of order pieces of size k obtained from a stock sheet of size i by following cutting pattern p, $p \in P(i)$, $i=1,\ldots,n$

d_k = the number of order pieces having size k, $k=1,\ldots,m$

q_i = the number of stock sheets having size i, $i=1,\ldots,n$

1.5 Solution Procedure

The solution procedures for a two-dimensional CSP may be grouped into two broad categories that involve either algorithmic or heuristic approaches. Algorithmic procedures include mainly linear programming, dynamic programming and a state space procedure like the branch and bound method. An algorithmic method guarantees an optimal solution for a specified problem but it is computationally expensive. Because of the complexity of a two-dimensional CSP, an exact solution is impossible in many cases or the computational effort is prohibitive by using an algorithmic procedure. Thus, instead of trying to reach an optimal solution, most researchers have turned to heuristic solutions.

A heuristic procedure cannot guarantee an optimal solution. It is a “short cut” and provides “rules of thumb” by which one can search for a “satisfactory” rather than an optimal solution. A heuristic procedure is judged to be acceptable if the solutions are “good enough”. That is, the solutions are believed to be within a tolerable deviance range from the optimal solution, if such a solution exists. Heuristic procedures are often incorporated into exact procedures to reduce the computational burden.

1.6 Factors Affecting Solution Procedure

There are two important factors that need to be considered in developing a new method or evaluating an existing method, namely the trim-loss and the computational complexity.

1. Trim-loss

Although there are many different situations in a two-dimensional problem, minimizing the trim-loss is always the main objective. It can be seen from the previously described problem formulation (see equations 1.1 through 1.4) that the objective function is to minimize the cost caused mainly by the trim-loss. There are many factors that affect the trim-loss. The major ones are the procedure used for a single stock sheet layout, the number of stock sheet sizes involved, and the distribution of order pieces in the BOM.

2. Computational Complexity

An algorithmic procedure, whether it involves linear or dynamic programming or a state space procedure, often needs tremendous computational effort. Even with the aid of advanced computer technology, the computer time and memory required are still very large. It is generally believed [34] that the computer time and memory grow exponentially as the number of stock sheet sizes increases. Therefore, many researchers have tried to develop heuristics to reduce both the computer time and memory but not at the expense of significantly increasing the trim-loss.

Throughout this thesis, the performance of a heuristic procedure is assessed by using the Average[1] (percentage) Stock Sheet Utilization (ASSU) and a computer's CPU time. The CPU time is based on an IBM compatible microcomputer that has a 90 MHz Pentium processor and 16 MB RAM. Although this time is not an absolute, it does provide an idea of a heuristic procedure's relative efficiency.

1.7 Goals and Scope of Research

Most research on the two-dimensional CSP deals with the layout of a BOM on the basis of a single stock sheet size. More recent research [34] has included the sequencing of stock sheets, that is selecting the proper sequence of stock sheets to be used for a BOM if more than one sheet is required and available in inventory. This study considers a two-dimensional CSP in which specified numbers of different rectangular order pieces are required to be cut from a set of multiple sized, rectangular stock sheets. It is based on a single stock sheet size approach that was formulated by El-Bouri et al [17]. The algorithm

[1] The "average" represents the arithmetic mean through this thesis.

and computer program developed by El-Bouri are adapted here to the multiple stock sheet size problem.

The main objective of this study is to develop a selection procedure for the cutting sequence of a set of predefined and multiple sized, stock sheets. Accomplishments include the following aspects.

1. A new procedure is proposed to sequence multiple sized stock sheets. This procedure achieves a relatively low trim-loss, compared with existing heuristic procedures, and it is relatively computationally inexpensive.
2. A computer program is developed. It is capable of automatically generating random test problems, searching for an effective cutting sequence for multiple sized stock sheets, and recording the output for further applications.

The following conditions detail the problem under study.

1. It is a two-dimensional, constrained problem. The requirement of the BOM must be satisfied exactly. There should be no restriction to guillotine-type cuts.
2. There is more than one rectangular stock sheet size available. The number of different stock sheet sizes and the dimensions of each sheet must be predefined.
3. The number of each sized stock sheet can be either infinite or finite.
4. The BOM is large enough to require more than one stock sheet.

The remainder of the thesis is organized as follows. Chapter 2 provides a review of the existing literature on two-dimensional CSP. Then three existing procedures of sequencing stock sheets, developed by Qu and Sanders [34], are detailed in Chapter 3. The first one is an exact Branch and Bound (B-B) procedure, the rest are the heuristic based. The reason these particular procedures were selected is that they are the only procedures that currently deal with a similar problem to that studied in this thesis. They can also be programmed easily on a microcomputer. A comparison of the three procedures serves as a basis for developing a new heuristic procedure.

A full description of the new heuristic procedure is presented in Chapter 4. It is developed to perform competitively over a wide range of problem mixes. Moreover, the procedure is expected to significantly reduce computer time and memory but with no more than a small increase in the trim-loss compared with the three existing procedures. This section also includes the development of a random problem generator that can automatically create the test problems (i.e. the BOMs as well as the stock sheets).

Chapter 5 discusses the results of testing the existing and new procedures. Tests are undertaken by using several test problems generated by the random problem generator. The results are compared and discussed. Finally, conclusions and recommendations are presented in Chapter 6.

Chapter 2. Literature Review

This chapter presents a survey of the substantial work done for two-dimensional CSP. The focus is aimed at publications related to single and multiple sized, rectangular stock sheets. The survey is divided into the following sections. Section 2.1 concentrates on single sized stock sheet problems that have been studied extensively. Different methodologies used in solving this type of problem are outlined. This survey is followed in section 2.2 by a review of multiple sized, stock sheet problems. Two inverse problems are discussed in detail. The first one is the assortment problem, in which there is freedom to select the sizes of the stock sheets. The second one, which is important but relatively less studied, is the trim-loss problem in which a set of known stock sheets are needed to be sequenced to minimize the trim-loss. Finally, a summary of the survey is given in section 2.3.

2.1 Single Size Stock Sheet

This section is divided into two parts; the first part reviews procedures leading to exact solutions, the second part surveys heuristic procedures.

2.1.1 Exact Solution Procedures

Many researchers have formulated the CSP by using linear and integer programming techniques or recursive procedures in trying to solve the problem exactly. Gilmore and Gomory [2], for example, formulated the two-dimensional, unconstrained[2], guillotine CSP as a linear programming problem in a manner similar to the one-dimensional CSP of their earlier works [3, 4]. They solved the problem in two stages[3], each stage being a one-dimensional knapsack problem. Standard dynamic programming procedures were used at each step. To solve the problem more quickly, Gilmore and Gomory [5] discussed the theory of knapsack functions and developed two algorithms. The two algorithms, although based on dynamic programming, were essentially iterative. The second algorithm provided a further improvement to the computation of the staged, two-dimensional, knapsack function. However, this improvement was shown later by Herz [6] to be incorrect. Beasley [7] also found a further, inadvertent error in the dynamic programming recursion.

A recursive algorithm was used by Herz [6] to solve the unconstrained, guillotine CSP. Recursion is better than an iterative algorithm because of its simple logic. In Herz's algorithm, a stock sheet was divided into subsheets such that each subsheet contained an order piece (dissection) from the BOM. In generating the optimal dissections, priority was

[2] Unconstrained means that there is no limit on the number of each type of order piece in the BOM.
[3] Two stages means that the first cut is parallel to one edge so as to form strips; each strip is then cut in the direction perpendicular to the first cut.

given to homogeneous dissections[4]. Upper bounds were set for the length and width of every dissected rectangle in order to increase the computational speed. Moreover, during recursion, the optimal value of each partial rectangle encountered was stored so that it could be used later if such a rectangle had to be considered. Herz reported that the recursive technique required 20% less computational time than the iterative algorithm of Gilmore and Gomory [5].

Beasley [7] developed an algorithm for an n-staged, unconstrained guillotine CSP by using dynamic programming recursion. The idea of "normal patterns" was used to improve computational efficiency. A recursive procedure for the non-staged CSP was also developed. In fact, the latter was just a staged problem where the number n was not fixed. It was demonstrated that this approach was capable of solving problems with as many as 50 order pieces in under 2 seconds on a CDC 7600 computer.

Christofides and Whitlock [8] provided a tree search algorithm for the two-dimensional CSP that had the guillotine restriction. They also constrained the maximum number of each type of order piece. An enumerative tree was generated in which each node represented the state of a cut-out rectangle whilst each branching represented a cut. An enumerative procedure was designed to generate "normal" cutting patterns without duplication due to symmetry or the ordering of the cuts. The search for the optimal solution was limited by the upper bound of the solution obtainable at each node. Tremendous computational effort was still needed, however, to search for an optimal

[4] Homogeneous dissection means that the order pieces have the same length and the same width.

solution even though duplicate cuts were eliminated. This was because a branch and bound tree search was used. For example, the average CPU time to generate a pattern involving 20 order piece sizes was over 2 minutes on a CDC 7600 computer. This result suggests that the procedure could be used to solve primarily problems that involve fewer than 10 to 20 order piece sizes. In a later paper, Christofides and Hadjiconstantinou [9] presented a tree-search algorithm that improved the algorithm developed by Christofides and Whitlock [8]. The new algorithm limited the size of the tree search by using an upper bound derived from a state space relaxation of a dynamic programming formulation. Although this algorithm achieved better results, it may be used to solve problems involving fewer than about 30 order piece sizes due to the still excessive computational burden.

Beasley [10] used Lagrange relaxation of a zero-one integer formulation as an upper bound in a tree search to find the optimal cutting patterns for non-guillotine cuts. Sub-gradient optimization and reduction tests made it possible to solve problems involving 10 order piece sizes on a CDC 7600 computer in anywhere from 10 to 229 seconds.

Although exact procedures can generally find an optimal solution, they all have the common disadvantage of computational inefficiency, especially for problems that involve over 30 order piece sizes. Therefore, instead of trying to reach an optimal solution, many researchers have turned to heuristically determined solutions.

2.1.2 Heuristic Procedures

Adamowicz and Albano [11] formulated a heuristic procedure for the two-dimensional constrained problem. To shorten the computational time, they generated candidate groupings of rectangular shapes, called strips, and selected a subset to fill a sheet or a subsheet. Thresholds were used to limit the number of strips generated; the strips were selected by dynamic programming. Albano and Orsini [12] upgraded this procedure by introducing more heuristics to cover the generation of non-homogeneous, uniform and quasi-uniform strips. All these heuristics were computationally efficient and they were able to produce a near-optimal solution, especially for high Average Piece to Stock Area (APSA) ratios.

Wang [13] also presented a n-stage algorithm for the two-dimensional, constrained CSP. Unlike a traditional method of dissecting the stock sheets, the algorithm successively builds rectangles to form a large guillotine pattern. However, those rectangles whose percentage waste exceeds a prescribed "aspiration level" are rejected. In this way, all possible guillotine rectangles are formed and the one producing the least trim-loss is chosen.

Oliverira and Ferreira [14] adapted Wang's algorithm into a branch and bound form. They pruned a partial solution as if cutting patterns were generated whose waste percentage was surely greater than the aspiration level. This algorithm was shown numerically to improve the performance of Wang's algorithm without degrading the solution.

Priority rules were used by Dietrich and Yakowitz [15] to solve a single sized CSP. Order pieces were classified as zero, one, or two-degree fits in relation to the stock sheet or subsheet to be cut. The classification corresponded to how many dimensions a piece had in common with the stock sheet. Then the classified pieces were prioritized so that a piece having a higher degree received higher priority. Six different rules were applied to select the order of pieces that initially had equal priority. Test results showed that the rule-based algorithm gave a higher average utilization. Computer times were reported in the range of 0.009 to 0.025 second per piece on an IBM PS/2.

Israni and Sanders [16] developed an algorithm based on the first fit decreasing method. Order pieces were arrayed in decreasing lengths (with ties resolved by decreasing widths). They were placed alternately along the length and width of a stock sheet, starting from the bottom left corner. A piece that could not fit in the available space was retained. When no further pieces could be allocated from the pre-ordered list, that particular stage was considered complete. The procedure was repeated for the residual sub-sheet. Human intervention was allowed after each step in order to fill gaps by using small pieces or by performing rotations, if useful. Computational times ranged from 5 to 15 minutes for randomly generated BOMs that included 135 to 279 order pieces. Most of the time was consumed by the intervention of the user.

El-Bouri et al. [17] described a search-based, heuristic procedure for the two-dimensional, constrained CSP. This procedure combined a tree-search routine with a priority rule system in order to decide the manner in which pieces were allocated to the available areas of a stock sheet. The search routine provided an enumerate method to find a low trim-loss layout for each processed subsheet. Priority rules were designed to allocate, in the earlier stages, specific pieces that had large piece to stock dimension ratios in order to avoid a higher overall trim-loss. The heuristic was compared with four heuristics published previously by Bengtsson [18], Dietrich and Yakowitz [15], Albano and Orsini [12], as well as Israni and Sanders [16]. Results showed that it performed better than these four heuristics in a majority of the tests.

An approximation algorithm for the two-dimensional guillotine CSP was presented by Macleod et al. [19]. Each order piece was considered, in turn, and assigned to a position on the stock sheet that made guillotine cuts feasible. The algorithm has the property that a position is located for the n-th rectangular piece if and only if such a feasible placement exists. A comparison with Wang's [13] algorithm demonstrated that improved results were achieved.

2.2 Multiple Size Stock Sheets

The solution procedures described in section 2.1 all assume that there is only one stock sheet size. However, it is a common situation in real-world applications that stock sheets are available in multiple sizes. In the flat glass industry, for example, two or three stock sheet sizes can be supplied typically to satisfy a customer's order that contains many order pieces of different sizes. This section provides a survey of the multiple sized, stock sheet CSP. The first part describes the literature that tries to determine the best combination of stock sheet sizes for a known, single BOM (the so called assortment problem). The second part reviews publications that aim to find a set of multiple sized stock sheets that minimize the overall trim-loss when a known single BOM is cut (the trim-loss problem).

2.2.1 Assortment Problem

Virtually all manufacturing operations need to determine the material sizes that need to be stocked to fulfill a BOM composed of smaller sizes. Because of storage or manufacturing limitations, or the cost associated with holding different sizes in stock, only a limited number of stock sizes are usually stocked. The issue of selecting the stock sizes is known as the assortment problem. The first attempt to solve this problem was given, perhaps, by Wolfson [20]. It was a one-dimensional case where the length of a steel bar needed to be selected. The selection was done by simple enumeration if the number of different stock sizes, n, was two. Dynamic programming could be used for larger values of n. It was shown that the trim-loss was reduced greatly by increasing n. However, Wolfson pointed out that a large value of n could impose higher stocking and production costs.

Page [21] considered the problem of cutting steel sections but only one size of a small, rectangular order piece could be cut from each rectangular stock. The problem was posed as a two-dimensional one involving the width and length of the stock sections. A heuristic approach, based on a dynamic programming relaxation procedure, was used to find the sizes of the rectangular stock.

Chambers and Dyson [22] indicated that a wise selection of stock sizes could reduce the trim-loss more effectively than a good cutting pattern. They presented a heuristic procedure for a two-dimensional assortment problem. The best width was selected first by presuming that all lengths were available. Then the lengths of stock were determined by assuming that the feasible sizes all had the same width. This procedure reduced the problem to essentially a one-dimensional form.

Beasley [23] presented an integer model in developing a heuristic algorithm for the two-dimensional assortment problem. Three procedures were used: a greedy procedure for generating two-dimensional cutting patterns, a linear program for choosing the cutting patterns, and an interchange procedure to decide the best subset of rectangular stock to cut. The algorithm initially found cutting patterns that included as many order pieces that had the same size as possible. Then more cutting patterns were discovered by using the greedy heuristic. Linear programming was employed to select a fixed number of patterns from those generated previously. Finally, the interchange heuristic was used to select the stock sheet sizes. Good Computer results indicated that this combined approach was capable of dealing with moderately sized (i.e. 10 different stock sheet sizes and 30 different order piece sizes), two-dimensional, guillotine cutting assortment problems.

Diegel and Bocker [24] provided decision rules for the assortment problem in the glass industry but limited themselves to guillotine cuts. They selected several "good" stock sizes for a given set of order pieces by simulating the cutting plans without counting the exact number of cuts required for each piece. When the actual cutting plans were produced, only the stock sheet sizes selected by the simulated run were accepted for the order pieces. Consequently, a lower loss was produced than if the cutting procedure had access to all the stock sheet sizes. Diegel and Bocker came to the quite unexpected conclusion that more choices mean worse choices. This was so because the cutting procedure took the stock sheet that happened to be best for the few order pieces wanted now, without taking account of those used on the next run. Their procedure was tested on several sets of genuine data. It produced consistently good results.

Pentico [25] addressed a discrete, two-dimensional assortment problem by considering situations involving concave production-inventory cost functions and substitution costs that were additive and proportional to the amount substituted. The characteristics of the optimal stocking pattern were presented. Three heuristic procedures were compared with an optimization technique, called DUALOC, developed by Erlenkotter [26]. It was found that the three procedures did not perform as well as DUALOC. On the other hand, they were relatively simple and provided results close to optimal. In addition, the new heuristics were modified easily to permit nonlinear concave variable costs which DUALOC could not accommodate.

A dynamic programming approach that was also based on a heuristic method was proposed by Gochet and Vandebroek [27] for a deterministic, two-dimensional assortment problem. The heuristic was applied to real-world, cardboard buying. Both the required set[5] and the extended set[6] were used to test the heuristic. Results indicated that it performed much better, with less computer time, than the existing heuristic method of Page [21].

Gemmill and Sanders [28] compared three commonly used heuristic methods developed by Beasley [23], as well as by Chambers and Dyson [22] for the assortment problem. They introduced an optimization homotopy (OH) method that was developed originally for a stochastic problem. It was found that, if the set of feasible sizes was relatively small (i.e. less than 19 stock sheet sizes), the three heuristic methods appeared to perform well. As the set of feasible stock sheet sizes grew and the incremental difference between feasible dimensions decreased, the application of OH tended to provide the best

[5] The quantity required for each order piece must be met exactly.
[6] Extra order pieces of each size were allowed.

results. However, it was computationally more complex than the three commonly used heuristic methods [22, 23].

Agrawal [29] presented a method, developed for an automobile press shop, to determine stock sheet sizes that minimize the total trim-loss. Order pieces of only one size were cut from one stock sheet to avoid coordination problems. However, the same sheet size could be used for another part at another time. Stock sheet sizes were determined in the following three steps. A sheet size was suggested first that was likely to give a low trim-loss for major parts. Then a trim-loss matrix was generated that gave the trim-loss for every combination of the suggested sheet sizes and parts. Finally, a heuristic was used to select a given number of sheet sizes. A "dominant" sheet size was introduced. It was defined as a sheet size which gave either the same or less trim-loss than other sheet sizes. For every part, sheet sizes that were not yet proved to be dominant were called "non-dominant" sheet sizes. By excluding the non-dominated sheet sizes, the problem size could be reduced without noticeably affecting the results. This method was used to determine stock sheet sizes for a steel sheet cutting shop. According to [29], the trim loss could be reduced by 90%.

Vasko and Wolf [30] introduced a practical approach to determine rectangular stock sheet sizes. They solved the problem in three phases. In phase one, a large number of stock sizes were generated that were ideally suited to the BOM. In phase two, these stock sizes were consolidated to an "acceptable" number. In phase three, a heuristic method was employed to lay out the BOM on the acceptable sizes of stock sheets and then find the best combination. It was demonstrated that their approach was efficient for a real-word BOM with 392 distinct order sizes and over 7700 order pieces.

Gemmill and Sanders [31] treated the two-dimensional assortment problem as the "portfolio" problem. Unlike other approaches, their method dealt with multiple possible BOMs. They analyzed the relationship between the average trim loss and the ratio, n, of the stock sheet size to the average piece size for BOMs having various standard deviations, s, from a specified average value. They derived a regression model to determine n and s for uniform and normal distributions of BOMs. General guidelines for the portfolio problem were presented and applied to complex problems. The results were found to compare favorably with those of the optimization homotopy (OH) algorithm [28].

Yanasse [32] considered a special case of the assortment problem in which there was only one dimension (i.e. the length) of a stock sheet involved. He proposed a procedure to generate a lower bound on the best solution that could be obtained from each of the stock lengths within an allowed range. This lower bound was updated each time a potentially good stock length was analyzed. The search stopped when the result was within a pre-specified value. This approach is quite simple and straightforward. It can be implemented easily when an optimizing program is already available that accommodates the cutting constraints. It was shown that the approach is better than a previously suggested method by Yanansse, Zinober and Harris [33].

In regard to the assortment problem, the most useful algorithms would appear to be those of Beasley [23] and Chambers and Dyson [22]. Their algorithms can be applied, in an identical manner, to either single or multi-dimensional problems in order to choose a "good" subset of stock sheet sizes from a finite set of feasible sizes. However, their approaches appear to perform well only when the set of feasible stock sheet sizes is less than 30. Therefore, researchers such as Gemmill and Sanders [28], Vasko and Wolf [30] have striven to solve the large assortment problems. Their approaches appear to perform good for the practical problems that are as large as 392 order piece sizes and 7700 total pieces.

2.2.2 Trim-loss Problem

The trim-loss problem described here is considered in conjunction with the two-dimensional CSP. It comprises a set of known, multiple sized stock sheets that are to be sequenced to complete a known, single BOM with minimal total trim-loss. Relatively less work has been done for this kind of problem compared with the assortment problem.

Qu and Sanders [34] presented three procedures to find the sequence of stock sheets for a two-dimensional BOM layout when (a) there is more than one stock sheet size, and (b) when the method for laying out each single sheet has been chosen already. The first procedure involved an optimization Branch and Bound (B-B) algorithm to determine the "best" sequence, i.e. the sequence of stock sheets producing the smallest total trim-loss. Although the B-B can find the optimal solution, it requires tremendous computer time and memory. To reduce the memory requirement, a second heuristic procedure, called PREORDER, was developed. It is a recursive, preorder search technique that finds a "good" sequence of stock sheets and requires little computer memory. To reduce further both the computational memory and time requirements, a third heuristic procedure, called STEP, was proposed. It employs a stepwise tree-search technique. Although the STEP procedure can significantly reduce both the computer memory and time requirements, it may substantially increase the trim-loss.

Yanasse et al. [33] suggested a pattern building heuristic procedure, combined with an enumeration scheme, to find the best mix of stock sheets and cutting patterns that satisfy a BOM with minimum wastage. The combined procedure identifies possible combinations of stock sheet sizes from which all the required order pieces could be cut. The procedure was tested by employing a limited number of problems. It was shown to produce a good solution but the computer run-time was a little long. For example, a problem involving 30 different types of order pieces, 60 total pieces, and 3 different stock sheet sizes needed a computer run-time of over two hours on a 286/AT PC machine [33].

Generating a cutting pattern with minimum trim-loss is not the complete story in the glass industry. The sequence in which patterns are cut must also be decided. For instance, several patterns must be cut to fulfill an order for many pieces of a given size, *X*. After cutting one of these patterns, a stock of size *X* pieces is obtained. However, the stack cannot be removed until the required number of size *X* pieces has been cut from the other patterns. Such incomplete stacks pose handling and space problems. Consequently, the cutting of the pattern must be sequenced to reduce the queue length of the stacks. Madsen [35] solved the sequencing problem in two stages by employing a traveling salesman approach. The first stage consisted of solving the CSP without the sequencing constraints. In the second stage, a sequencing problem that was formulated as a travelling-saleman model was used to order the cutting patterns in an optimal or near-optimal way. The results from fourteen test runs indicated that the procedure reduced the average and maximum order spreads by approximately 31% and 18 %, respectively. Yuen [36] suggested two heuristics to solve the sequencing problem in two steps. In the first step, pieces were selected that were involved in one or more unsequenced patterns from the available pieces. In the second step, the heuristics selected the next pattern to sequence from candidate or unsequenced patterns that involved the selected piece from step one. Test results showed that the average maximum queue decreased by around 35% with a 1 to 2 % increase in the trim-loss.

In summary, few approaches can be found from the literature that tackle the trim-loss problem. The most useful procedures would seem to be those of Qu and Sanders

[34]. They were designed to select a sequence of known, multiple sized stock sheets that are required to complete a known, single BOM. Qu and Sanders' approach is unique because the stock sheet selection is treated as a sequential decision problem that is solved by using combinatorial optimization branch and bound methods and associated heuristics. The procedure used to choose the sequence of stock sheets is independent of the procedures used to perform the layout function. Hence, any single sheet layout procedure could be fit into their approach. The problem studied in this thesis is similar to that of Qu and Sanders. Therefore, their procedures are detailed in the next chapter. The goal is to uncover and rectify the weaknesses.

2.3 Summary

This chapter has presented a general survey of two-dimensional CSP. It has been seen that there have been many attempts to solve the CSP and related problems. Solution procedures range from traditional linear and dynamic programming to new techniques of artificial intelligence such as a depth-first, tree search. On the other hand, all these techniques can be divided into two groups. The first group includes linear and dynamic programming. Such procedures generally create optimal solutions. But, as the size of the problem increases, a high computational burden becomes apparent. A typical example would be Gilmore and Gomory's algorithm [2]. The second group, which is essentially heuristic in nature, aims to reduce this burden but the quality of a solution deteriorates to a near-optimal one. Typical examples would include Wang's algorithm [13], Dietrich and Yakowitz's priority rules-based heuristic [15], Israni and Sanders' first fit decreasing method [16], El-Bouri et al.'s search-based heuristic [17], and Qu and Sanders' procedures [34].

In summary, the two-dimensional CSP has been studied greatly, especially for a single sized stock sheet problem and the assortment problem. However, few studies can be found for solving a trim-loss problem that involves multiple sized, stock sheets.

Chapter 3. Comparison of Three Existing Procedures

The literature survey of Chapter 2 revealed that relatively little work has been done to predict the trim-loss from two-dimensional, multiple sized, rectangular stock sheets. To further explore this problem, the three procedures of Qu and Sanders [34] are detailed in this chapter. The main reason these particular procedures were selected is that they are the only procedures that deal with a similar problem to that studied in this thesis. In addition, the simple logic of Qu and Sanders' procedures makes them relatively easy to program on a microcomputer. The performances of the three procedures are evaluated based on the:

- quality of solution, which is measured by the average stock sheet utilization; and
- the computational efficiency, suggested by the CPU time and memory requirements.

It is intended that the strengths and weakness of these procedures be uncovered so that an improved procedure can be developed.

3.1 General Description of the Three Procedures

Qu and Sanders [34] approached the selection stock sheets as a sequential decision that they resolved by using the combinatorial optimization method of a Branch and Bound (B-B) algorithm or associated heuristics (the PREORDER and STEP procedures).

Procedures to choose the sequence of the stock sheets are independent of the procedure used to perform the layout. Therefore, any single sheet layout procedure can be employed. The layout procedure they selected for a single sheet was the Decreasing Length, Perpendicular Strip Packing (DLPER) approach designed by Israni and Sanders [16].

3.1.1 Problem Definition and Research Objective

The problem that the three procedures handle is described next. Suppose a BOM contains total of ***D*** rectangular order pieces that belong to ***m*** distinct sizes. It is required that d_k of order pieces be cut for each size from a set of the rectangular stock sheets. There are ***n*** distinct sizes of stock sheets and a sufficient stock sheets are in storage to cut the entire BOM. More than one stock sheet is required to fulfill the BOM.

The process of laying out the BOM involves several stages. At the beginning and every consecutive stage, there are ***n*** distinct sheet sizes to choose from and only one stock sheet to be chosen. When the current sheet is filled and the BOM is still incomplete, a new sheet is started and the procedure is repeated until the BOM is completed. Although the method for laying out the BOM on each single sheet is identical, the trim-loss will usually vary for different stock sheet sizes. Thus, by selecting different stock sheet sizes, different sequences are obtained that produce different total trim-losses. The set of all the possible sequence alternatives can be illustrated by an upside-down tree. The root of the tree represents the start of the layout and there are ***n*** branches. Except for the root node, each node of the tree represents a layout on a sheet. The depth along each path, from the root to the leaf or end node, represents the total number of stock sheets, M, required to complete the BOM following that particular sequence. The number of distinct sequences equals the total number of leaf nodes.

The individual trim-losses for each of the M sheets in a particular sequence is represented by TL_i so that the total trim-loss is: $TL_1+TL_2+...+TL_{M-1}$. Note that the trim-loss on the last sheet, TL_M, is not counted because it usually does not reflect the packing density. Indeed, this remaining portion can usually be kept and used for future BOMs. The objective is to find the sequence that will minimize the total trim-loss but meet the order piece requirement of the BOM.

3.2 Branch and Bound (B-B)

The B-B procedure produces an exact solution for the trim-loss. It uses a top-down approach to check every branch under the tree, from the root to a leaf node. (Note, however, that the simulated tree will be illustrated as an upside-down tree in subsequent figures.)

3.2.1 Description of the B-B Procedure

Step 1

Starting from the root the procedure goes down to the first level. One stock sheet of each of the *n* sizes available on the first level is selected and filled with the same BOM. The trim-loss of each sheet is recorded as the *CTL*. The *CTL* represents the cumulative trim-loss of the stock sheet associated with the node.

Step 2

The nodes on the first level are put into a List of Terminal Nodes (LTN) in an ascending order that corresponds to the values of their *CTL*. Then the node having the least *CTL* is selected as the current node and it is taken from the LTN.

Step 3

The layout of the BOM continues from the current node down to the next (lower) level. *n* more stock sheets are laid out by using the parts of the BOM remaining from the previous layout up to the current node. If the BOM is completed at any node, the procedure is finished. Otherwise, the resulting *n* nodes below the current node, each with its *CTL* incremented from the last layout, are added into the LTN. The LTN now includes all the nodes that have been created except for the current node. As in Step 2, the nodes in the LTN are sorted in ascending order of the values of their *CTL*s. The node having the least *CTL* is selected as the current node and it is removed from the LTN.

Step 4

Step 3 is repeated until the BOM is completed.

3.2.2 An Illustrative Example

An example of selecting a sequence by using the B-B procedure is shown in Figure 3.1. The procedure has been coded in Turbo Prolog [37]. The method for the layout of a single sheet is the Search-Based Heuristic (SBH) developed by El-Bouri et al. [17]. This example involves only two stock sheet sizes; size 1 which is 149x52 and size 2 which is 141x51. Both stock sheet sizes have arbitrary but consistent units of distance. The BOM is created by a random problem generator that will be described in the next chapter. Table 3.1 details the BOM considered here.

Figure 3.1 presents a simplified overview of the example. The layout can be seen from this figure to be completed from the available two stock sheet sizes at the 4th level along the tree. The ***ID*** employed in Figure 3.1 is a digital string that identifies the node of a simulated tree. On the other hand, ***SQ*** indicates the increasing sequential order of the nodes visited by the procedure. The procedure starts branching at the root, laying the same BOM on stock sheet sizes 1 and 2 at the first level. Two nodes, i.e. node 1 (***ID***='1') and node 2 (***ID***='2'), are obtained. The ***CTL*** of nodes 1 and 2 are 1133 and 786, respectively. The two nodes are then put into the LTN in ascending order values of their ***CTL***s, i.e. node 2 first and, secondly, node 1. Node 2, which has the least ***CTL***, is selected

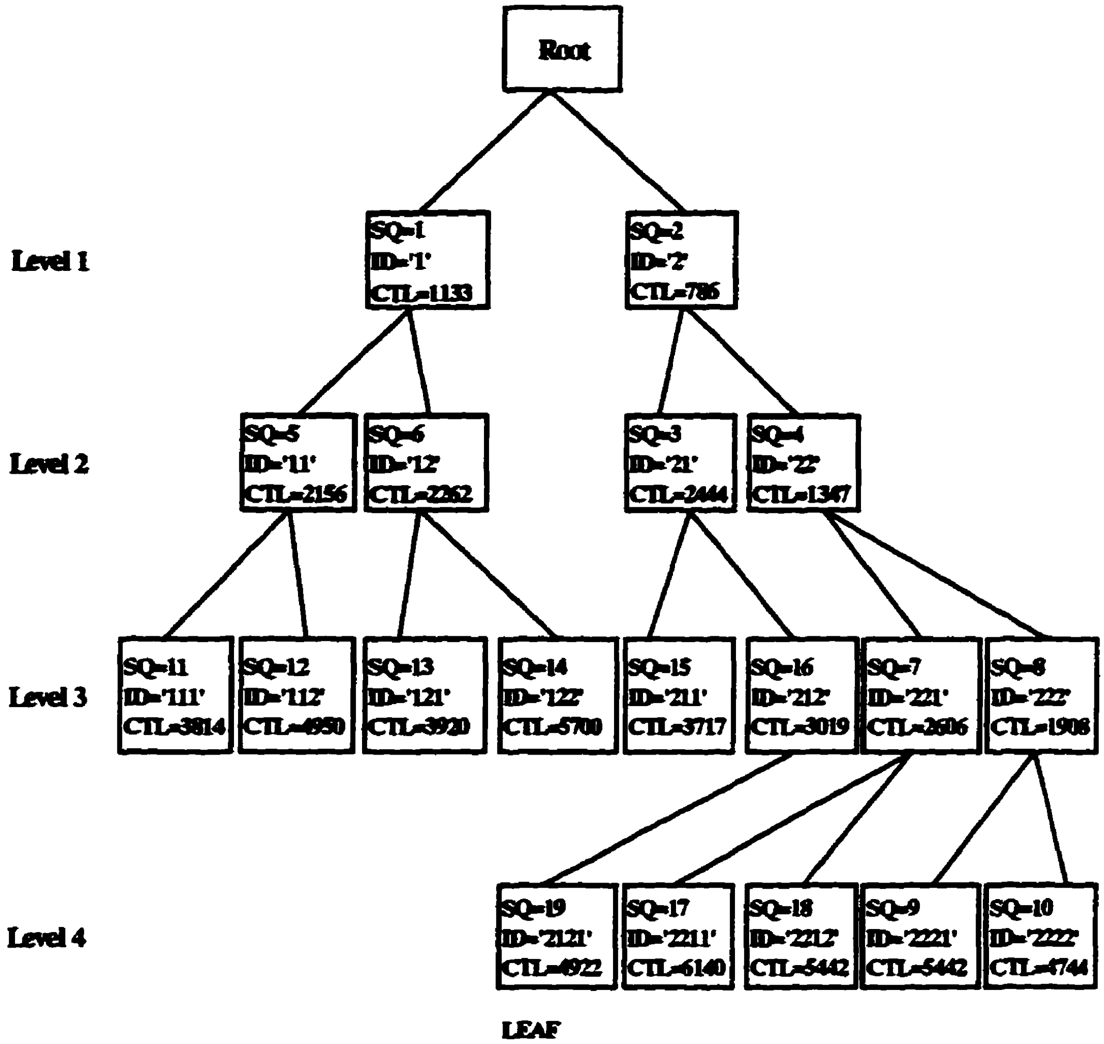

Figure 3.1. Example of the B-B Procedure.

Table 3.1. BOM #1.

Piece #	Length	Width	Quantity Required
1	45	21	8
2	43	14	7
3	42	29	9
4	38	26	1
5	19	10	5

from the LTN, as the current node and it is removed from the LTN. The procedure continues from the current node (node 2) to the next lower level (i.e. level 2). Two more stock sheets, one for each size, are employed to lay out the partly finished BOM remaining from the previous layout upto node 2. The resulting two nodes (nodes 21 and 22), each with its own ***CTL*** incremented from the previous layout (i.e. 2444 for node 21 and 1347 for node 22), are added into the LTN. Consequently, the LTN now contains the three nodes 1, 21, and 22. The nodes in the LTN are sorted again in the order of their increasing ***CTL*** values. Again, the node that has the least ***CTL*** (node 1, at this time) is selected as the current node. It is removed from the LTN. From the current node, the previously described procedures of sheet laying, updating the LTN, and selection of the current node are repeated. The B-B procedure is stopped at node 2121 in level 4; at which point the BOM is completed. Node 2121 in the final level 4 is a leaf node so that it is not counted in the solution. The node next to the leaf node, i.e. node 212, corresponds to the final solution. In other words, the procedure achieves the least ***CTL*** by following the sequence '212' to fulfill the BOM.

3.2.3 Discussion

It can be seen from the last section that the LTN must be stored in the computer's main memory because it is expanded continuously as more nodes are added. The information needed to be kept for each node are the:

- layout pattern;
- order pieces of the BOM that are not satisfied yet;
- ***CTL, ID, SQ***, and whether the BOM is complete or not.

The worst-case memory requirement of the B-B is related to the maximum number of nodes that need to be stored before the final sequence is found. As the final sequence can appear only in the node at the last level (i.e. level M), all the nodes at the (M-1)th level need to be stored. Therefore, the maximum Memory Unit[7] (MU) is the number of nodes at the (M-1)th level, i.e. $n^{(M-1)}$. For large problems (n > 3), this requirement can become a major difficulty. Therefore, the second procedure of PREORDER is designed to reduce the computer memory requirement.

3.3 PREORDER

The PREORDER procedure includes an original and a modified version, both of which were developed by Qu and Sanders [34]. The original PREORDRDER procedure produces an exact solution but the modified PREORDER procedure is an approximate heuristic one. Both are described in the following sections. However, only the modified version will be studied further because the original procedure requires much more computer time than the B-B procedure to find the optimal solution.

[7] A memory unit is the computer space required to store the information associated with a node.

3.3.1 Description of the Original Procedure

Step 1

The procedure follows any one path under the tree by starting from the root and proceeding down the tree to the leaf node, i.e. the BOM layout is completed by using any sequence of stock sheet sizes. The leaf node is defined as the "best node" and the resulting *CTL*, as well as the sequence and pattern layout, are stored.

Step 2

The same BOM is laid out by using any of the remaining paths. The leaf node of the new path is defined as the current node. If the resulting *CTL* of the current node is less than the *CTL* of the best node, the best node is updated by using the current node. Otherwise, the previous best node is retained.

Step 3

Step 2 is repeated until all paths are exhausted.

This procedure uses only two memory units: one for the best node and the other for the current node. However, the procedure takes more time to find the best sequence than the B-B procedure because every node under the tree must be examined.

3.3.2 A Modified Version of the Procedure

To reduce the number of nodes that need to be examined, Qu and Sanders [34] introduced an assumption that the Percentage Trim Loss (***PTL***) of a single sheet does not decrease along any path down the tree. This assumption is based on the observation that many order pieces of the BOM are not satisfied at the upper levels of the tree so that they can be selected later to fill gaps on a stock sheet. Towards the completion of the layout (i.e. at the bottom of the tree), fewer sizes of order pieces are available for placement on each stock sheet so that the later sheets of the layout sequence will have a generally higher ***PTL***.

To explain the PREORDER procedure, it is assumed that two sizes of stock sheet are available. The area of sheet 1 is 20% larger than that of sheet 2. The procedure involves the following steps.

<u>Step 1</u>

First, only one size of sheet, say size 1 illustrated in Figure 3.2, is selected. The BOM is laid on the same size 1 sheets until the BOM is completed. Each time a sheet is finished, the ***PTL*** of that sheet is calculated. Assume that a total of ***M*** sheets of size 1 have been used. Then the resulting sequence is 1...11.

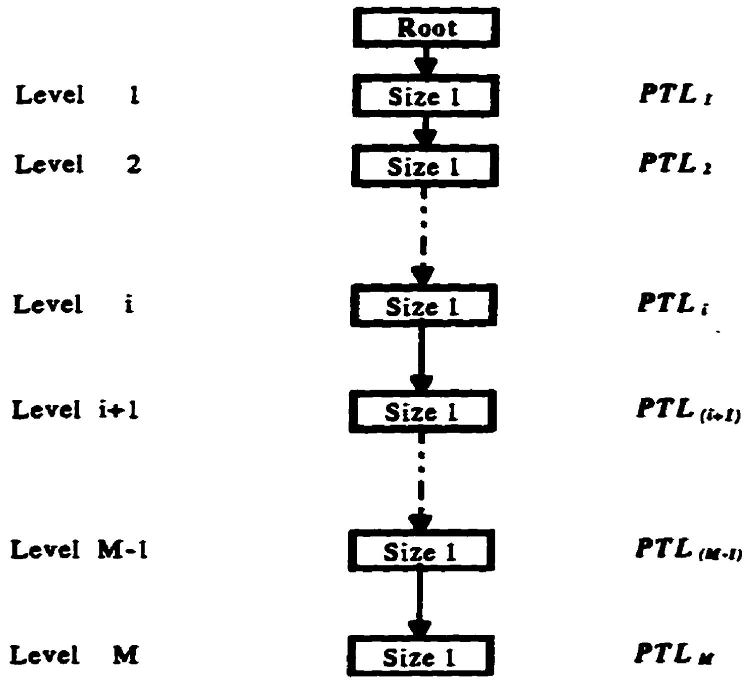

Figure 3.2. Showing the BOM First Laid on Size 1 Sheets in the PREORDER Procedure.

Step 2

Now the questions to be confronted are as follows. Is sequence 1...11 the best? Is it possible that, by replacing some of the size 1 sheets in the sequence, a better layout can be achieved? To answer this question, the procedure back up one level from the bottom of Figure 3.2. Then a size 2 sheet is used below level (*M-1*) instead of size 1 sheet. As a result, node A is obtained on level *M* in Figure 3.3. If node A completes the BOM but gives a *PTL* that is less than PTL_M(i.e. the *PTL* of the size 1 sheet on level *M*), that means below level (*M-1*), the size 2 sheet is better than the corresponding size 1 sheet. Therefore, the size 2 sheet is chosen on level *M*. The new sequence 1...12 is stored. If, conversely, the *PTL* is larger than PTL_M, the previous sequence 1...11 is retained.

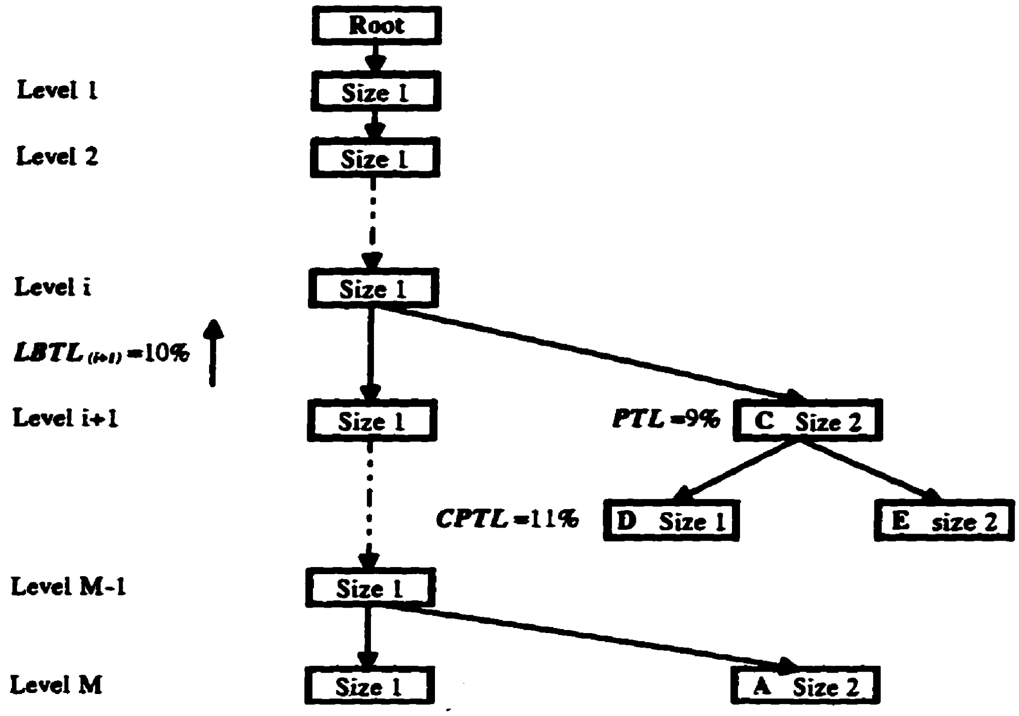

Figure 3.3. The PREORDER Procedure.

Step 3

The procedure traces continuously up the tree, level by level, from level (*M-1*). Assume that the best sequence has been found below level (*i+1*) and that the average ***PTL*** of the sequence from the (bottom) level ***M*** to level (*i+1*) is 10%. This average ***PTL*** is called the Local Best Trim Loss ($LBTL_{(i+1)}$) from level ***M*** to level (*i+1*). A size 2 sheet is tried at node **C** on level (*i+1*) in Figure 3.3. If the layout of node **C** gives a ***PTL*** greater or equal to $LBTL_{(i+1)}$=10% then, according to the assumption that the ***PTL*** does not decrease along any path down the tree, all the nodes under **C** will have ***PTL***s of at least 10%. Therefore, by proceeding further down from node **C**, the procedure will not generate a path that improves the one found already. The procedure will stop at node **C** and go back to level *i*. On the other hand, if the ***PTL*** of node **C**, which is assumed in Figure 3.3 to be 9%, is less than $LBTL_{(i+1)}$-10%, a better path than the one found already may be determined by proceeding down from node C. In this case, the procedure goes down one level to lay out the remaining BOM by using size 1 sheet at node D. The cumulative *PTL* (*CPTL*) is calculated and compared with the $LBTL_{(i+1)}$. If the *CPTL* of node D is greater than $LBTL_{(i+1)}$, the procedure stops at node D. Otherwise, it continues down the tree until the condition $CPTL > LBTL_{(i+1)}$ is met or the BOM is finished. As in Step 2, each of the two stock sheet sizes must be examined below the *i*th level and the best path generating the smallest total trim-loss is stored.

Step 4

Step 3 is repeated until the root of the tree is reached.

3.3.3 Discussion

From the previous description of the PREORDER procedure, it can be seen that the original version always finds the optimal solution. However, the procedure is computationally expensive because all the nodes of the tree must be examined. The modified version is based on the assumption that the *PTL* does not decrease along any path down the tree. It can reduce the computing time by limiting the nodes that need to be examined. If the above assumption is valid, the PREORDER procedure will find the optimal solution and it will do so quicker and using less computer memory than the B-B procedure. According to Qu and Sanders [34], PREORDER achieves the same trim-loss as the B-B procedure but uses 50% less computing time for a problem involving 200 order pieces and two stock sheet sizes. Unfortunately, this improvement does not always happen. The trim-loss on each sheet is affected not only by the composition of the BOM but also by the method used for the layout of a single sheet. For example, a single sheet layout procedure may fill order pieces that lead to a high trim-loss at the early stage of the sheet layout in order to minimize the total trim-loss. Therefore, a later sheet may have a trim-loss which is less than that of a previous sheet. This situation makes the assumption invalid and leads to a non-optimal solution. In addition, the time requirement of PREORDER may still be large even though the number of nodes visited is limited.

3.4 STEP

STEP was designed to reduce both the computer time and memory requirements needed by the B-B and PREORDER procedures. It also uses a top-down approach.

3.4.1 Description of STEP

Step 1

Starting from the root, the procedure goes down to the first level. One stock sheet of each of the *n* sizes is selected and filled with the same BOM. The percentage trim-loss of the stock sheet associated with each node is recorded as *PTL*.

Step 2

The nodes on the first level are put temporarily into a List of Terminal Nodes (LTN) in the same ascending order as the values of their ***PTL***s. Then the node having the least ***PTL*** is selected as the current node and the LTN is emptied.

Step 3

The layout of the BOM continues from the current node down to the next lower level. ***n*** more stock sheets are laid out by using the partly completed BOM remaining from the previous layout upto the current node. If the BOM is completed at any node, the procedure is finished. Otherwise, the resulting ***n*** nodes below the current node, each with their own calculated ***PTL***, are put into the empty LTN. Consequently, the LTN includes all the nodes in the current level. As in Step 2, the nodes in the LTN are sorted in the same order as the ascending values of their ***PTL***s. The node having the least ***PTL*** is selected as the current node and the LTN is emptied.

Step 4

Step 3 is repeated until the BOM is completed.

3.4.2 An Illustrative Example

An example of the sequence selection obtained from STEP is shown in Figure 3.4. The procedure has been coded in Turbo Prolog [37]. The method for the layout of a single sheet is the Search-Based Heuristic (SBH) developed by El-Bouri et al. [17]. This example uses the same test problem as that employed in section 3.2. Two stock sheet sizes are available. It can be seen from Figure 3.4 that the layout is completed at the 5th level. The ***ID*** given in the figure is a digital string that identifies the node of a tree. Moreover, the ***SQ*** indicates the sequential order of the node visited by the procedure. The procedure starts branching at the root, laying the same BOM on both stock sheet sizes at the first level. Two nodes, node 1 (***ID***='1') and node 2 (***ID***='2') are obtained. The ***PTL***s of nodes 1 and 2 are 15% and 10%, respectively. The two nodes are put into the LTN in the same order as the ascending values of their ***PTL***s. From the LTN, node 2, which has the least ***PTL***, is selected as the current node and the LTN is emptied. The procedure continues from the current node (node 2) to the next lower level (level 2), again using both size 1 and 2 stock sheets. The partly completed BOM remaining from the previous layout up to node 2 is laid out. The resulting two nodes (nodes 21 and 22), each with its ***PTL*** calculated, are put into the LTN. Now, the LTN contains nodes 21 and 22. The nodes in the LTN are sorted again in order of their increasing ***PTL***s. The node that has the least ***PTL*** (node 22 at this time) is selected as the current node and the LTN is emptied again. Procedures for sheet laying, LTN updating, and current node selection are repeated from the current node. STEP stopped at node 22221 in level 5 when the BOM is completed. Node 22221 is the leaf node so that it is not counted in the solution. Hence the previous node, node 2222, corresponds to the final solution.

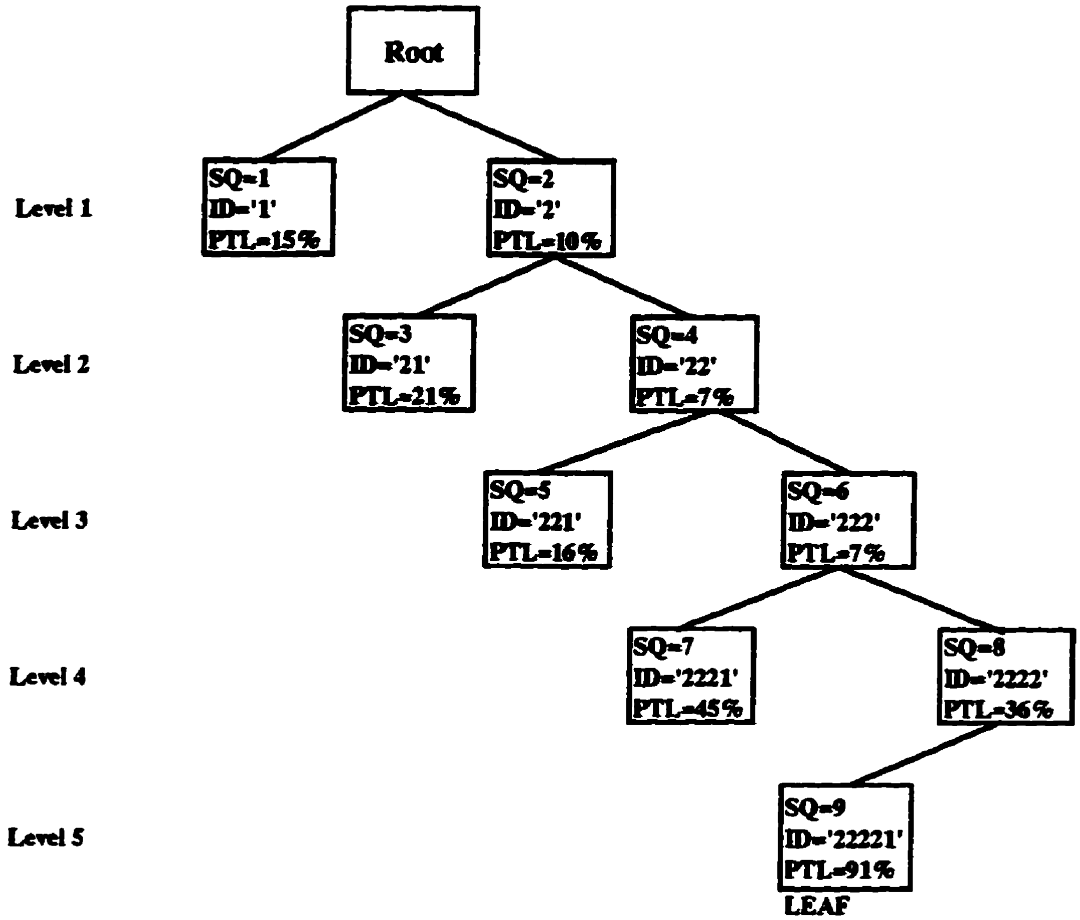

Figure 3.4. Example of the STEP Procedure.

3.4.3 Discussion

The STEP and B-B procedures seem alike. Both employ a top-down approach and both use a LTN to control the node to be examined. Actually, they are quite different. The B-B procedure takes the node that has the least *CTL* as the current node from all the nodes visited (including those in the previous levels). Once the current node has been found, no matter the level, the B-B procedure always backtracks to that node. Unlike the

B-B procedure, the STEP procedure is a non-backtracking "greedy" procedure. It searches for the node that has the least *CTL* within the current level. This node is taken as the current node before proceeding down to next level. No backtracking to previous levels is allowed. Consequently only n stock sheets are involved at each level for the STEP procedure (i.e. at any level only n different nodes need be remembered) so that the memory requirement is reduced to n units. The time unit[8] is $n \cdot M$ where M is the number of the levels required to complete a BOM. It is clear that computational time and memory requirements are generally reduced significantly by using STEP. On the other hand, although STEP is fast and memory-conserving, it can provide substantially non-optimal results because there is no backtracking.

In summary, the B-B procedure can find the "best" sequence. However, it is computationally expensive. PREORDER reduces computer memory requirements. STEP further reduces both the computer memory and time requirements.

3.5 Test and Analysis

To compare the performance of the three procedures, the Average (percentage) Stock Sheet Utilization (ASSU), CPU time and Memory Units (MU) are used. The calculation of the maximum MU is given in Table 3.2 [34].

[8] A time unit is the time required to lay out a single stock sheet.

Table 3.2. Calculation of the Maximum MU.

Procedure	MU
B-B	$n^{(M-1)}$
PREORDER	2
STEP	n

Table 3.2 indicates that when the number of different stock sheet sizes *n* is greater than two and the BOM is large enough to require more than two stock sheets (*M*>2), the PREORDER is the best procedure in terms of the MU. The B-B requires the most MU and the requirement of the MU increases exponentially as the *n* and *M* increase. The MU demanded by the STEP procedure depends on the *n* only.

3.5.1 Test Data

The test problems used for evaluating and comparing the three procedures are created by employing a random problem generator which will be described in the next chapter. The test data is generated randomly in eight different categories. The classification of these categories is based on the Average Piece-to-Stock Area (APSA) ratio and the number of different the stock sheet sizes. Table 3.3 details the classification of the categories. The APSA ratio is defined as the arithmetic mean area of the BOM pieces to the arithmetic mean area of the available stock sheets, regardless of the quantity of each sheet size. Four APSA ratios, ranging from 0.04 to 1.00, are identical to those used by Dietrich and Yakowitz [15]. Each of the first four categories showing in Table 3.3

involves two different stock sheet sizes. The second four categories involve three stock sheet sizes. Problems that involve more than three stock sheet sizes are not considered because the computer memory requirement of the B-B procedure is too large for the computer system (an IBM compatible, personal computer having a 90 MHz Pentium processor and 16 MB of RAM). However, a problem that involves more than three stock sheet sizes is used to assess the performances of the PREORDER and STEP procedures.

Table 3.3. Classification of Test Categories.

Category #	APSA	# of Stock Sheet Sizes
1	1.00	2
2	0.25	2
3	0.10	2
4	0.04	2
5	1.00	3
6	0.25	3
7	0.10	3
8	0.04	3

Testing the B-B and PREORDER procedures is very time consuming because of the computational complexity. Therefore, only five problems are tested for each category. The size of the BOMs is also limited in that the number of distinct order piece sizes ranges from 8 to 50. The total number of the order pieces is between 10 and 205. The length of an order piece ranges from 1 to 80 units; the width (width <= length) of an order piece is from 1 to 69 units. Stock sheet lengths and widths range from 29 to 116 units, respectively. The available quantity of stock sheets of each size is assumed to be unlimited.

3.5.2 Test Results

A test was undertaken by using the computer programs written to mimic the three procedures. Forty problems that belonged to the eight different categories were tested for each procedure. The results of each problem were recorded as:

- the Average (percentage) Stock Sheet Utilization (ASSU);
- the CPU time for the sequence selection; and
- the Memory Unit (MU) required by each procedure.

The results produced by the three procedures are presented in Figures 3.5 through 3.7 as well as in Table 3.4. Figure 3.5 shows a comparison of the ASSU for each category. On the other hand, Figure 3.6 gives the corresponding average CPU times. Figure 3.7 compares the CPU times for the PREORDER and STEP procedures when the number of stock sheet sizes is greater than three. Two to six different stock sheet sizes are involved in this test. The comparison of the memory units required by the three procedures is presented in Table 3.4.

Table 3.4. Comparison of the Number of Memory Units.

Category #	APSA	# of stock sheet sizes	B-B	PREORDER	STEP
1	1.00	2	21	2	2
2	0.25	2	26	2	2
3	0.10	2	16	2	2
4	0.04	2	13	2	2
5	1.00	3	92	2	3
6	0.25	3	179	2	3
7	0.10	3	71	2	3
8	0.04	3	49	2	3

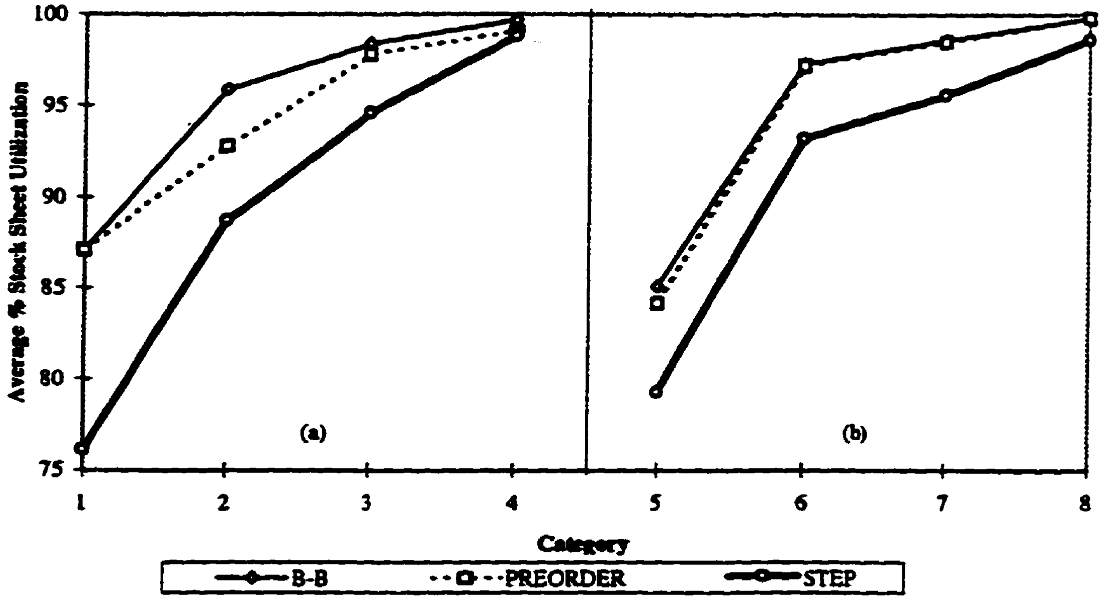

Figure 3.5. Comparison of the ASSU for (a) Two and (b) Three Stock Sheet Sizes.

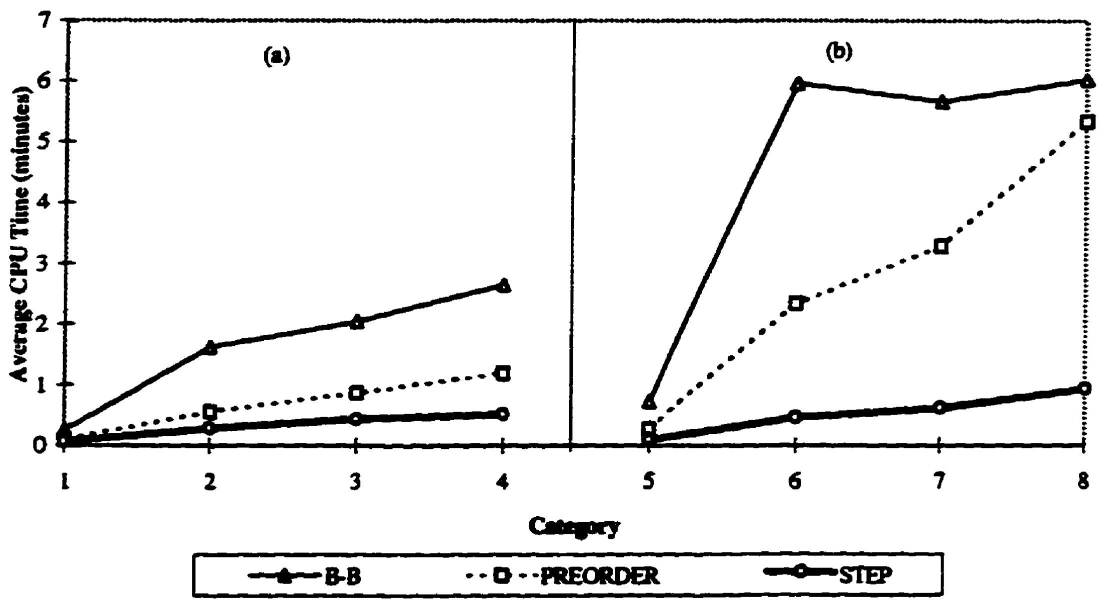

Figure 3.6. Comparison of the CPU Time for (a) Two and (b) Three Stock Sheet Sizes.

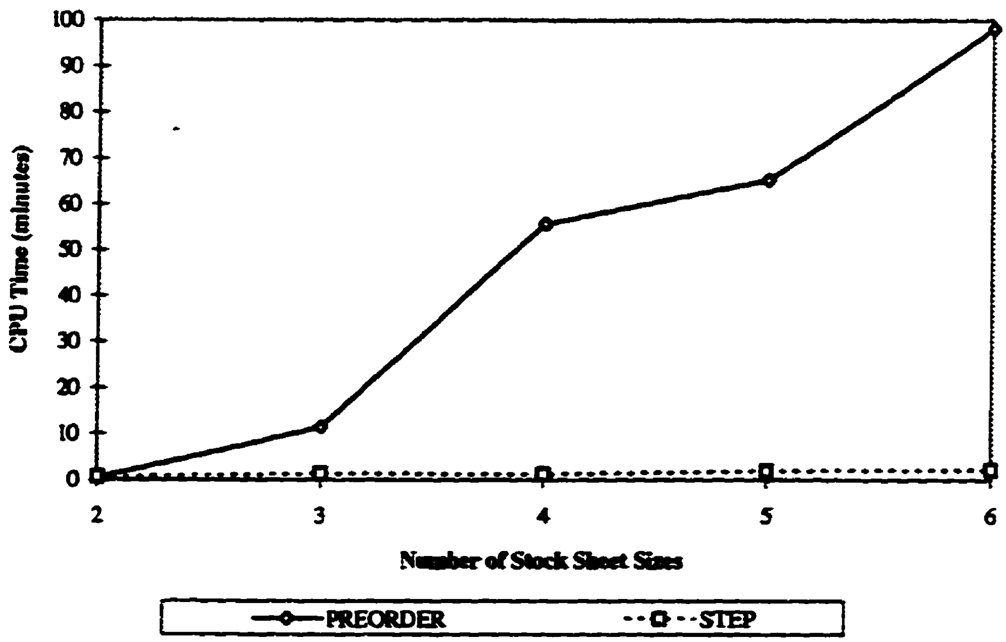

Figure 3.7. Comparison of the CPU Times for the PREORDER and STEP Procedures.

3.5.3 Analysis of the Results

The following observations are based on the data shown in Figures 3.5 through 3.7 and Table 3.4.

1. Figure 3.5 indicates that the B-B procedure achieves the highest ASSU in all categories. On the other hand, it needs the most CPU time, as shown in Figure 3.6. Table 3.4 also shows that it requires the most memory units. This is because, as mentioned early, the list of terminal nodes has to be kept in the computer's main memory and every node in the list needs to be checked before the best sequence can be found.

2. The PREORDER procedure is comparable, in terms of ASSU, with the B-B procedure in all categories except category 2. Because it only needs two memory units, the memory problem of the B-B procedure is resolved by the PREORDER procedure. However, the time requirement remains very large, especially for APSA ratios less than 0.25. Indeed, Figure 3.7 shows that as the complexity of the problem (i.e. the number of stock sheet sizes) increases, the time requirement grows exponentially.
3. The major drawback of STEP is that it degrades the quality of a solution even though it uses the least CPU time and a small memory. It can be seen from Figure 3.5 that the ASSU decreases in all categories compared with the other two procedures. The decrease ranges from about 1% to 10%.
4. Table 3.4 indicates that the B-B procedure uses the most MU. It must be pointed out that the actual number of MU of the B-B did not reach the maximum because the optimal solution was found before the last node was examined.

3.5.4 Conclusions

The following conclusions can be drawn based on the comparison of the results.

1. The B-B procedure is unsuitable for solving a problem in which there are more than three stock sheet sizes and 250 order pieces.

2. PREORDER is a good alternative to the B-B procedure if the computing time is not a major issue.
3. STEP has a very high computational efficiency. However, the quality of the solution needs to be improved.

3.6 Summary

Three previously published procedures [34] were introduced and evaluated in this chapter. They are the B-B, the PREORDER, and the STEP procedures. The principles behind these procedures were detailed. They were evaluated by using problems generated by a random problem generator. It was found that the B-B procedure achieved the best sequence but with large computer time and memory requirements. PREORDER significantly reduced the memory requirements and kept the quality of the solution. However, the computer time was still very large. Although STEP significantly improved the computational efficiency, it degraded the quality of the solution.

Chapter 4. The Pre-specified Trim-loss Procedure

The intent of this chapter is to (a) present a full description of a new procedure for sequencing orders of multiple sized stock sheets, (b) evaluate this procedure by testing it using four different heuristic polices on various test problems, and (c) draw conclusions that lead to the further studies discussed in the next chapter. First, the principles and computer programming procedures for the new procedure are detailed. Then the new procedure is tested four times by using 25 categories of data created by a random problem generator. Each time, one of four different heuristic polices is implemented in the new procedure. A small group of five test problems is used for each category to identify appropriate heuristic polices. Finally, conclusions are drawn from the test results.

4.1 The New Procedure

The new procedure for sequencing orders of multiple sized stock sheets has been designed to overcome the shortcomings of the existing procedures discussed in the previous chapter. It was developed to emphasize the following three principles.

1. All stock sheet sizes must be pre-examined individually against the BOM before the stock sheet sequence selection begins. Then an appropriate stock sheet size that best suits the BOM is found.
2. The number of stock sheet sizes checked must be minimized during the sequence selection so that the computational effort is reduced.
3. The procedure must be monitored continuously by using the trim-loss produced from the stock sheet being examined currently. Then sheets generating an undesirably high trim-loss are eliminated.

The first principle implies that a specific BOM favors a particular stock sheet size. In other words, there is a stock sheet size among all the sizes involved that tends to have the least total trim-loss for a particular BOM. This stock sheet size is defined as the Basic Stock (sheet) Size, BSS. The second principle is considered implicitly in the new procedure by limiting the search of the stock sheets to the BSS sheets. The third principle is fulfilled by means of a heuristic tree search that is constrained to a Pre-Specified Trim-Loss (PSTL). Even when the BSS has been found and used for a BOM layout, the trim-losses of every individual BSS sheet that complete the BOM are quite different. Hence, the trim-loss of each BSS sheet must be monitored so that the high trim-loss BSS sheets, determined by comparing their trim-losses with the PSTL, may be replaced quickly by other stock sheet sizes. Therefore, the total trim-loss can be reduced further.

The new procedure is termed a "pre-specified trim-loss procedure" because it uses a PSTL to control the selection of the stock sheets. Like existing procedures, the PSTL procedure uses a top-down tree search but differently. A tree search basically enumerates all the possible sequence alternatives that are feasible for a given problem. Then it selects the one that best satisfies the objectives. However, to check all the sequence alternatives is not feasible because the number is enormous, especially for a large BOM and many stock sheet sizes. Therefore, the PSTL procedure attempts to limit the number of stock sheets that are examined in order to minimize the computational effort while maintaining a quality solution.

The PSTL procedure has been coded in Turbo Prolog [37], a computer language designed for artificial intelligence. The program takes a BOM and a set of multiple sized stock sheets as input and produces a specific sequence of stock sheets as well as a graphic pattern layout for each stock sheet. A file that contains the overall average stock sheet utilization and the coordinate data of each pattern layout is also provided.

4.2 Programming Language

Prolog was chosen as the programming language for the following reasons.

1. The PSTL procedure employs an existing heuristic, namely the Search-Based Heuristic (SBH) developed by El-Bouri et al [17], for a single sheet pattern layout. This procedure was coded already in Prolog and it provides a good single sheet pattern layout by extensively using a tree search technique.
2. The PSTL procedure needs to keep a dynamic database to select the order pieces from the BOM and sequence the stock sheet sizes from a group of candidate sizes. Prolog provides excellent support for a dynamic database in which the dimensions and up-to-the-minute quantities of each order piece and stock sheet size are stored as a database predicate (fact).
3. The PSTL procedure is itself a tree search procedure that employs a backtracking technique. The built-in backtracking capability of Prolog is an important consideration in choosing the programming language.

4.3 Pre-specified Trim-loss Procedure

As mentioned earlier, the number of possible combinations of stock sheets is very large due to the many possible cutting patterns. Therefore, it is generally not feasible to examine all the possible combinations. Accordingly, a heuristic PSTL procedure is adopted.

The PSTL procedure introduces the idea of BSS and PSTL to reduce the complexity of the problem. By constraining the search mainly to the BSS sheets, it is possible to control the scope of the search. By using the PSTL, the total trim-loss can be controlled. The PSTL procedure is basically composed of a BSS selection, a PSTL calculation, and a stock sheet sequence selection. In these routines, the SBH subroutine is called extensively in order to achieve a single sheet layout pattern.

4.3.1 Input Data

The program takes a BOM and a set of multiple sized stock sheets as input. The BOM usually includes a number of smaller rectangles that vary in size (length and width) and quantity. The stock sheets are all rectangular. They are supplied in multiple sizes but the length, width, and quantity of each size are given. The input data are arranged as a set of predicates (facts) within the Prolog database. A predicate, defined by Current_piece (Id,L,W,Q), is used to specify a particular BOM piece in the database [17]. The variables L, W, and Q represent, respectively, the length, width and quantity of piece type 'Id". Similarly, the multiple sized stock sheets are stored as the predicate Stock_size_multiple (Id,L,W,C1,C2,Q). Here C1 and C2 are the (rectangular) coordinates of the upper left corner of a stock sheet. They are used as a reference for the location of the stock sheet when presenting graphical output. The predicates can be updated or restored dynamically. For example, the predicate Stock_size_multiple (Id,L,W,C1,C2,Q) can be updated so that the variable "Q" is reduced by one upon allocation of one stock sheet of type "Id". On the other hand, "Q" is increased by one if the type "Id" stock sheet is not chosen in the final sequence.

4.3.2 Single Stock Sheet Pattern Layout - the SBH Routine

The layout method for a single sheet is an important part of the PSTL procedure. It provides necessary information such as the layout pattern and the trim-loss of the sheet being examined for the PSTL procedure. A good single sheet layout method not only helps to achieve a good solution but it also improves the computational efficiency. In this study, the SBH was chosen for the following reasons.

1. The SBH uses a tree search combined with priority rules to decide how order pieces are allocated to the available areas of a stock sheet. It utilizes a degree-of-fit concept as part of an algorithmic search to control the scope of the search and avoid a computational explosion. The priority rules are designed to guarantee the allocation of specific pieces in the earlier stages in order to avoid a higher overall trim-loss. These features make SBH an efficient method for a single sheet layout.
2. Test results show that the SBH outperformed, in terms of computational efficiency and average sheet utilization, existing procedures in the majority of problem categories tested [11, 15, 16].
3. The SBH has been programmed previously and access to the source code is available. Therefore, it can be employed readily in the PSTL procedure.

4.3.3 Basic Stock Sheet Size - the BSS Selection Routine

The purpose of this routine is to identify a single stock sheet size, among the stocked sizes, that is most suitable for a particular BOM. The BSS selection examines all the available stock sheet sizes and searches for the best size, namely the one producing the least total trim-loss in filling a BOM. When a given stock sheet is cut into order pieces, there may be a trim-loss. This trim-loss varies with the stock sheet size, the sizes and quantities of the order pieces to be produced, and the method of laying out a single sheet pattern. In the case of a previously specified BOM and single sheet pattern layout method, the proper selection of the stock sheet size becomes a major issue in controlling the trim-loss.

One way to determine the BSS is to lay out, by using the SBH routine, the same BOM on single sized stock sheets for each of the available stock sheet sizes. The BSS selection routine includes the following two steps. In the first step, all the stock sheet data are entered and sorted by descending area. Then the total trim-losses of each stock sheet size are obtained. They are compared to find the BSS which correspond to the lowest total trim-loss. It is assumed that there are enough stock sheets of each size to complete the BOM at the BSS selection stage. Details are given next.

1. All information about the stock sheets are entered into a computer from either the computer's keyboard or a pre-prepared data file. The information includes the

number of different stock sheet sizes, the length and width of each size and the available quantity of sheets for each size. Then the stock sheet sizes are sorted in descending areas. They are stored in the Prolog database as the predicate series of Stock_sizes_multiple (Id,L,W,C1,C2,Q). The "Id" is the series number from 1 to the number of different stock sheet sizes available.

2. The same BOM is laid out on single sized stock sheets by using SBH on each of the different stock sheet sizes. For a given stock sheet size, layouts proceed one sheet at a time. The cumulative trim-loss (*CTL*) of each sheet, incremented from the previous sheet's layout, is recorded. After the BOM is completed, the ***CTL*** after laying out the next to the last sheet is set as the total trim-loss for that size. The trim-loss on the last sheet is discounted because the last sheet usually does not reflect the packing density. The resulting ***CTL***s and the information about the stock sheet size being examined are stored in a predicate series Total_trim_loss (I,N,CTL). The "*N*" and "***CTL***" respectively represent the number of different stock sheets of constant size "*I*" needed to satisfy the BOM and the total trim-loss for this size. The BSS selection continues with the next (constant) stock sheet size until all the available stock sheet sizes are checked. Then the stock sheet sizes are sorted again in the ascending order of their ***CTL***s. This information is stored in the predicate series Total_trim_loss (I,N,CTL). After all the available stock sheet sizes are examined, the first element in the Total_trim_loss(I,N,CTL) which has the least total trim-loss is chosen as the final BSS.

The above method of finding the BSS is called the Least Total trim-loss Method (LTM). Such a method may be inefficient when the BOM is very long or the number of available stock sheet sizes is very large because every stock sheet size needs to be checked against the BOM. Alternatively, the Largest Area Method (LAM) is used. LAM simply chooses the stock sheet size having the largest area as the BSS. This method is based on the observation that a larger stock sheet tends to have less trim-loss than a smaller sheet. This occurs because, as the area of a stock sheet increases relative to those of the BOM pieces, better utilization can be expected because the order pieces are more likely to fit in the sheet's available space. However, when the BSS is selected by LAM, a stock sheet's aspect ratio needs to be considered too because a long, narrow stock sheet may result in a higher trim-loss than a short, wide sheet having a smaller area. Therefore, a stock sheet that has an aspect ratio (length to width) greater than five is removed when LAM is used to determine the BSS.

The BSS selection can be considered a special case of the assortment problem discussed in Chapters 1 and 2. It is special because only one stock sheet size is selected from a set of pre-specified, stock sheet sizes for the BOM layout. On the other hand, the BSS is not the final solution. It is only the first step in the PSTL procedure.

4.3.4 Pre-Specified Trim Loss - the PSTL Calculation

Although the use of the BSS sheets can give the lowest total trim loss, this does not mean that the lowest trim loss is achieved on each BSS sheet. Figure 4.1, for example, shows the trim loss when SBH is used to lay out the simple BOM detailed in Table 4.1. Three different stock sheet sizes are employed but each size is considered separately. Size 1 is 149x52, size 2 is 141x51, and size 3 is 130x50. These dimensions are again in arbitrary but consistent units.

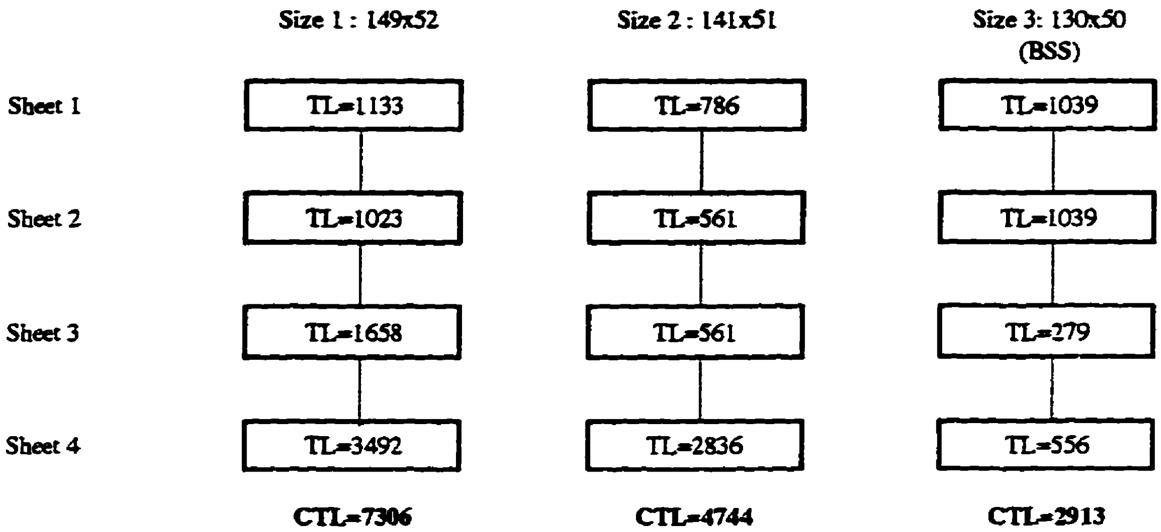

Figure 4.1. List of Trim-losses.

Table 4.1. List of the BOM.

Piece #	Length	Width	Quantity
1	45	21	8
2	43	14	7
3	42	29	9
4	38	26	1
5	19	10	5

It can be seen from Figure 4.1 that size 3 (130x50) produces the lowest total trim-loss (*CTL*=2913). Consequently, this particular size forms the BSS. However, the trim losses of the first and second BSS sheets (an identical 1039) are higher than those of the size 2 sheets (786 and 561). Therefore, there is still a possibility of further reducing the total trim-loss by replacing some of the size 3 BSS sheets with other sizes. To do this, the PSTL procedure needs to dynamically monitor the trim loss on every BSS sheet and decide which one should be replaced. Other size sheets should also be examined simultaneously to decide which size could be used for the substitution.

A particular value called the Pre-Specified Trim Loss (*PSTL*) is used to find the BSS sheets needed to be replaced. All the trim-losses of the BSS sheets are compared with this value. Any BSS sheet that has a trim-loss greater than this value may be replaced by another size stock sheet. The ***PSTL*** is also used to decide which size stock sheet could be used to replace the high trim-loss BSS sheet. Any other sheet that has a trim-loss less than the ***PSTL*** could be a candidate. The determination of the ***PSTL*** is critical because different ***PSTL***s produce different solutions. It can be seen from Figure 4.1, for instance, that the trim-losses of the BSS sheets of this example range from 279 to 1039. If the ***PSTL*** is set below 279, all the BSS sheets need to be replaced. On the other hand, if the ***PSTL*** is set above 1039, no BSS sheet needs to be replaced. If all BSS sheets are replaced, the total trim loss is increased. However, if none are replaced, the problem becomes a single sized, stock sheet problem. Therefore, the high trim-loss BSS sheets should be identified and replaced by other sizes. In other words, the ***PSTL*** must set at a value between 279 and 1039.

An average of the trim-losses of the BSS sheets could be used as the ***PSTL***. Unlike an extreme value, such as the smallest or the largest value, an average value is a measure of central tendency. The commonly used measures of central tendency are the arithmetic mean (normally abbreviated to mean) and the median. For any set of measurements the mean is computed by adding all the data values and dividing the resulting total by the number of values in the data set. The median conveys the notion of the middle value by dividing the distribution into two halves. The use of these two measures is compared in this thesis. The calculation of the ***PSTL*** by using these two measures is given below.

1. Mean measure: Suppose that there are n BSS sheets having the trim losses TL_1, TL_2, ..., TL_n. Then,

$$PSTL = \frac{1}{n}\sum_{i=1}^{n} TL_i \,. \qquad (4.1)$$

2. Median measure: Suppose that the trim losses of each BSS sheet are arranged in an array from the smallest to the largest loss. If the number of BSS sheets is odd, then the ***PSTL*** is the center value of this array. If the number of sheets is even, then the ***PSTL*** is the arithmetic mean of the two central values.

The set of trim losses corresponding to the BSS sheets shown in Figure 4.1 is used to demonstrate the calculation of the ***PSTL***. In Figure 4.1, there are four (n=4) trim losses of the BSS (i.e. size 3) sheets, namely TL_1=1039, TL_2=1039, TL_3=279, and TL_4=556. The mean of this set is: $PSTL=(TL_1+TL_2+TL_3+TL_4)/4=(1039+1039+279+556)/4=728.3$. To find the median, the data are arranged in ascending order, i.e. 279, 556, 1039, 1039. The number of the data is even so that the median is the mean of the two center values of 556 and 1039. That is $PSTL=(556+1039)/2=797.5$.

4.3.5 Sequencing Orders of Stock Sheets - the Sequencing Routine

After the BSS and PSTL have been determined, the procedure sequences the BOM order for multiple sized stock sheets. That is, appropriate stock sheet sizes and their sequence are found for the layout of a given BOM such that the total trim-loss is minimized.

The sequencing procedure uses a top-down, tree search that starts from the tree's root with a new BOM and finishes at the end node when the BOM is completed. The procedure goes down the tree, level by level. A particular level represents a layout pattern of a single stock sheet whose size is determined by the PSTL procedure. At each consecutive level, the BOM is updated by removing the order pieces that were filled at the previous level. Within a given level, the same BOM is employed when different sized stock sheets are tried. Then only a single sized stock sheet is selected. The search for the appropriate stock sheet always starts with the BSS sheets at each level. The trim loss of a

BSS sheet is recorded as the trim loss, ***TL***. Then the ***TL*** is compared with the ***PSTL***. If ***TL*** is less than or equal to the ***PSTL***, the BSS sheet is chosen at the current level. Then the procedure goes down to the next level. Otherwise, the procedure tries another stock sheet size, any size other than the BSS, to lay out the same BOM as that used by the BSS sheet. The condition that the ***TL*** is not greater than the ***PSTL*** is checked. The first stock sheet size that satisfies this condition is chosen. If all the stock sheet sizes have been examined and none can satisfy the condition, then the stock size producing the least ***TL*** (including the BSS) is chosen. The procedure continues down to the next level and it is repeated, level by level, until the BOM is completed. The combination of the stock sheets obtained at each level is the solution to the problem.

Figure 4.2 gives an example of the PSTL sequencing procedure when four stock sheet sizes are available. In this example, the values of the ***PSTL*** and ***TL***s are hypothetical. They are used only for the easy illustration of the PSTL sequencing procedure. The procedure starts sequencing from the root. The BSS has been found as size 1 and the ***PSTL*** is 18. At level 1, a BSS sheet is chosen because its ***TL*** (i.e. 10) is less than the ***PSTL*** of 18. Three sizes, sizes 1 through 3, are tried at level 2. The size 3 sheet is chosen because its ***TL*** (i.e. 15) is less than the ***PSTL***. All four sizes are examined at level 3. No size has a ***TL*** less than the ***PSTL***. Therefore, the size 2 sheet is chosen because it has the least ***TL***. The procedure stops at level 4 when the BOM is satisfied.

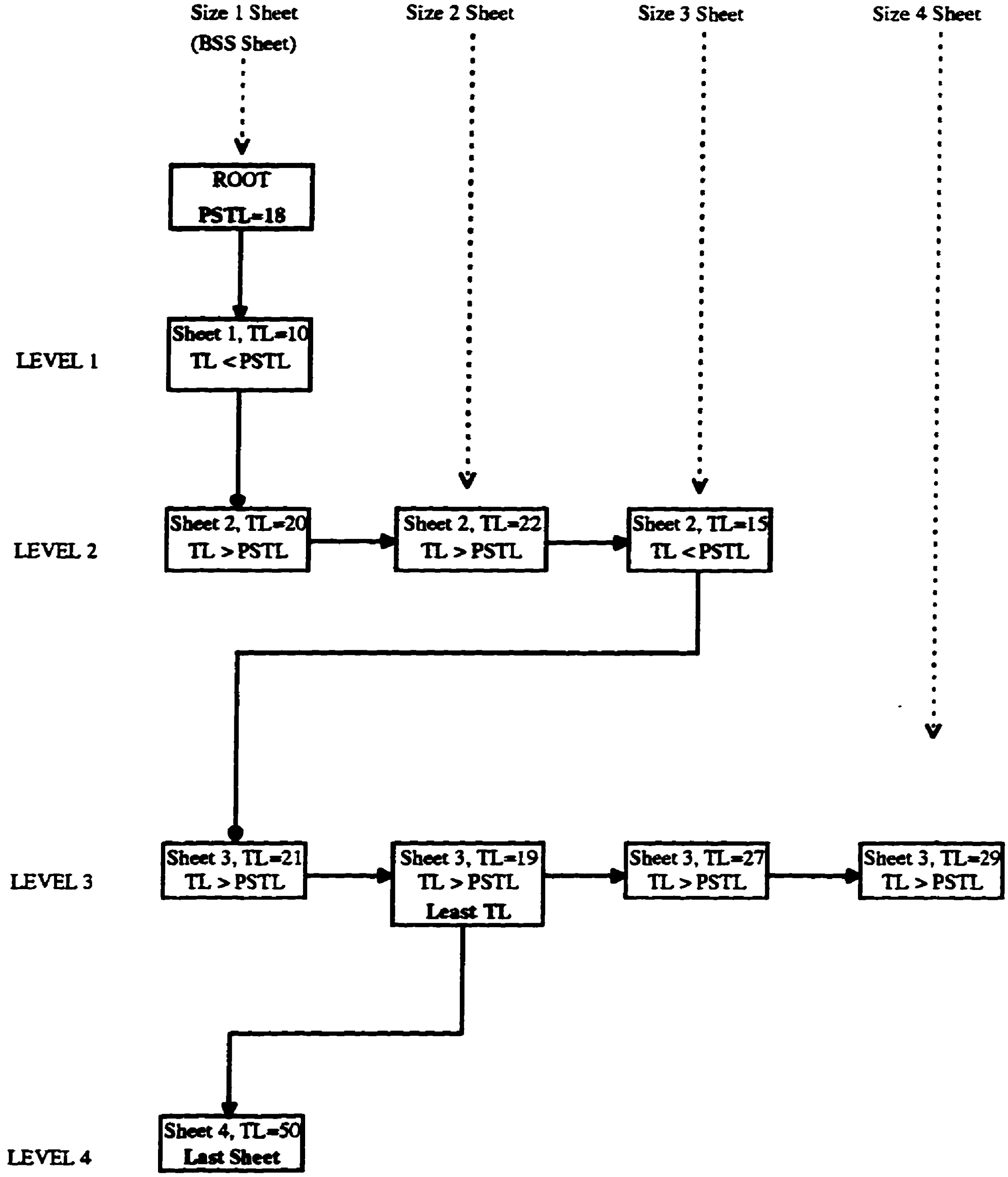

Figure 4.2. Example of the Sequence Selection Procedure.

4.3.6 Presentation of the Results - Data Output

Two kinds of output are generated by a data output routine [17]. The first is an on screen, graphical display of the layout of each sheet. Figure 4.3 presents a sample of the graphical display. The figure shows two types of output information, the geometry of the sheet's layout and information about the stock sheet, order pieces, sheet utilization, etc.

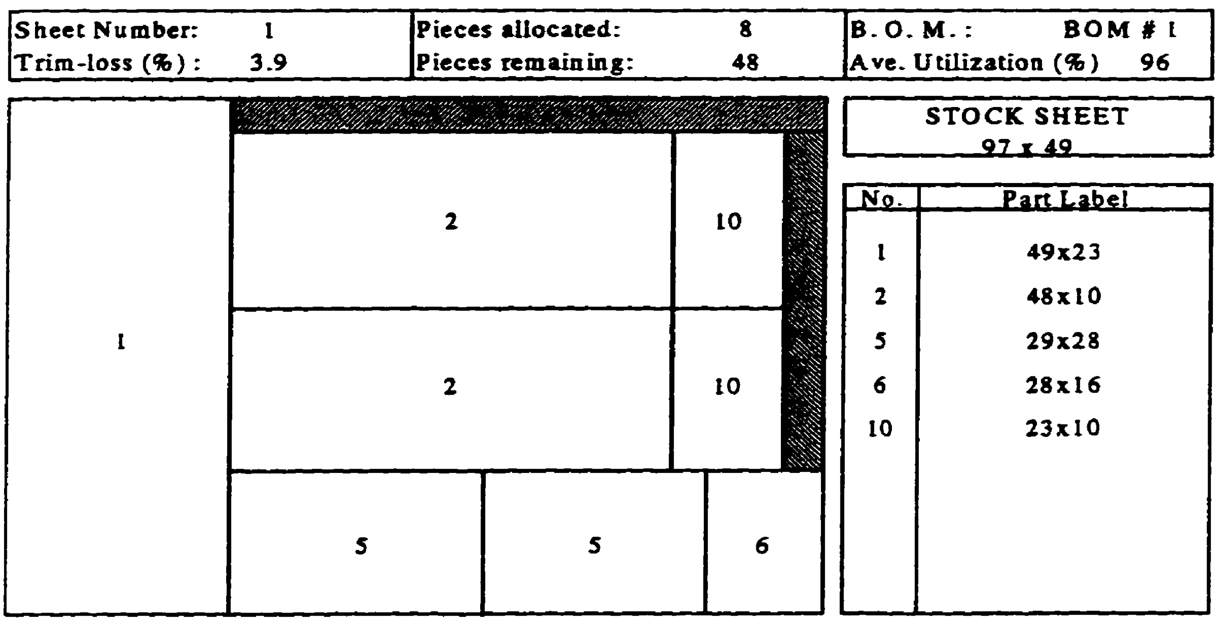

Figure 4.3. Sample of the Graphical Display.

The second kind of output is the digital representation of the sheet layout. The co-ordinates of each order piece on the sheet are stored in a data file. This file satisfies a request made to optimize the actual cutting path for a given BOM [38].

4.3.7 Description of the Iterative PSTL Procedure

Having introduced the principles of the BSS selection, the PSTL calculation and the sequencing routines, the PSTL procedure can be described with the aid of the flowchart shown in Figure 4.4. This procedure starts with the input of the information about the BOM and the available stock sheet sizes. Then the BSS is determined by calling the BSS selection routine and the PSTL is calculated by calling the PSTL routine. Iteration of the sequence selection starts by initiating the Prolog database. The bill of material is loaded first (in order of the descending piece areas) in a list named BOMLIST. The list of stock sheet sizes, called STOCKLIST, is loaded with the stock sheet sizes and their corresponding quantities placed in the ascending order of their total trim-losses obtained at the BSS selection. Obviously, the BSS is the first element of the STOCKLIST. An empty list (TLLIST) is also created in order to store the single sheet trim-losses for all the sheets examined within each level. The iteration continues by calling the SBH routine to lay out the BOM on a BSS sheet. The resulting ***TL*** is entered into TLLIST and compared with the ***PSTL*** to check whether the ***TL*** is less than or equal to the ***PSTL***. If the ***TL*** of the BSS sheet is less than or equal to the ***PSTL***, the BSS sheet is chosen and the procedure goes to next (lower) level to start a new BSS sheet. Otherwise, the procedure selects a stock sheet size other than the BSS in the STOCKLIST to lay out the BOM and start a new iteration within the current level. The iteration ends when either one of the following conditions is satisfied:

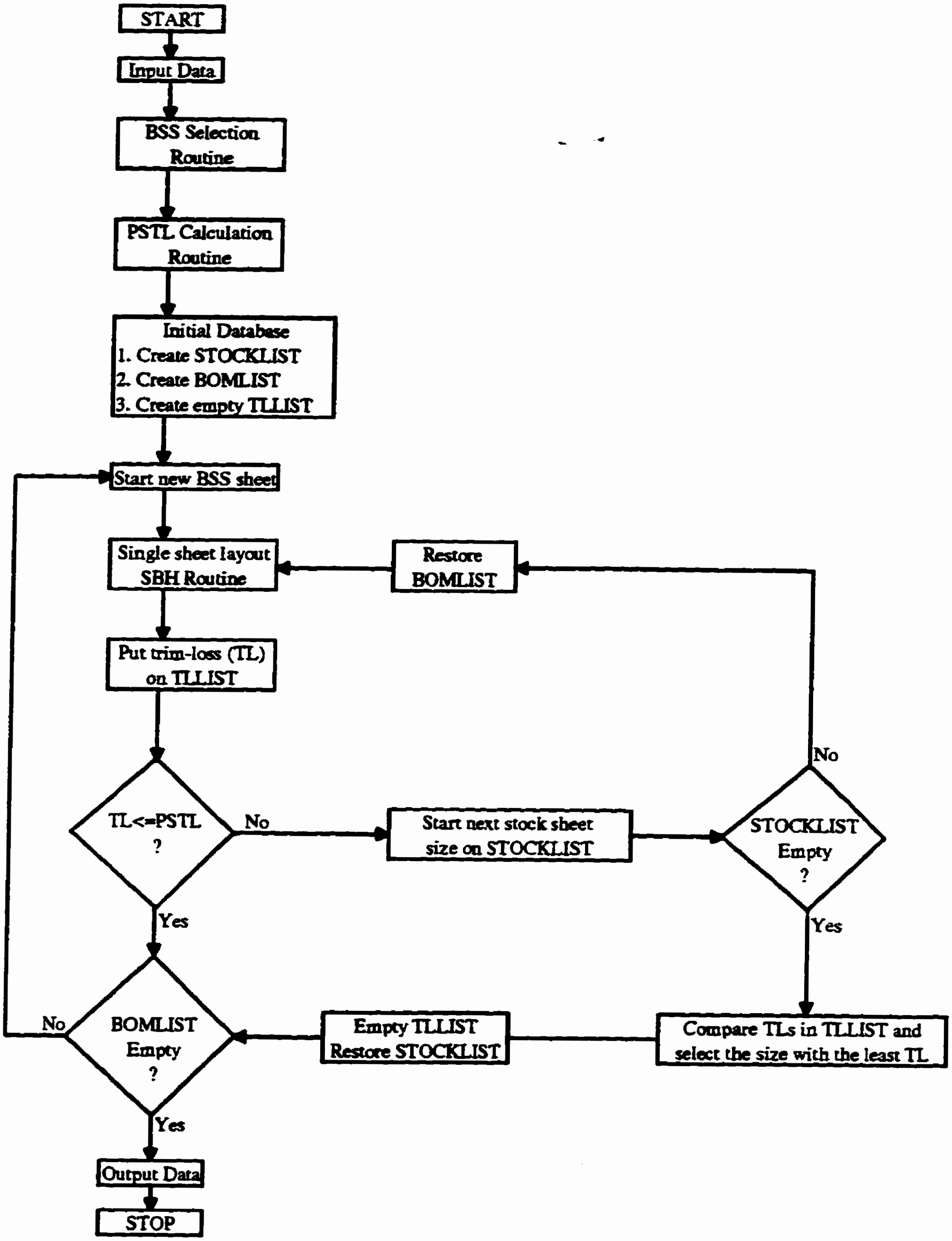

Figure 4.4. Flowchart of the PSTL Procedure.

1. the first size sheet is found which has a ***TL*** less than or equal to the ***PSTL***; or
2. all the stock sheet sizes have been checked.

If all the stock sheet sizes have been examined and no size satisfies the condition that the ***TL*** is less than or equal to the ***PSTL***, then the size generating the least ***TL*** is chosen. During the iteration, the BOM is updated, essentially instantly, with every allocation of an order piece. The BOMLIST is checked after each iteration. If no more order pieces are left, the entire procedure is terminated. Then the PSTL procedure outputs the results.

4.4 Evaluation of the PSTL Procedure

Various test problems are needed to systematically evaluate the performance of the PSTL procedure. Using test data sets and results published in the literature is one approach. However, a review of the literature revealed that very few publications give the details of test data sets. Therefore, a comparison of the results gained from such limited published data clearly represents an inadequate basis for drawing general inferences. Therefore, to obtain statistically reliable results, it is necessary to conduct tests on large data sets, each set having similar characteristics.

4.4.1 Classification of Test Problems

Test problems are classified according to the Average Piece to Stock Area (APSA) ratio and the number of different stock sheet sizes available. The APSA is the ratio of the mean area of the BOM pieces to the mean area of the available stock sheets, regardless of the available quantity of each sheet size. Test problems are classified into five different categories, each specified by an APSA ratio. The five APSA ratios considered are 1.00, 0.50, 0.25, 0.10 and 0.04. They are chosen because they represent five typical combinations of the BOM and stock sheet sizes [15]. Each of the five categories is classified further into five sub-categories. A sub-category is specified by the number of stock sheet sizes available in it. As there is no limit placed on the number of stock sheet sizes that can be used, the number of sub-categories is unlimited. However, to simplify the problem, the number of stock sheet sizes is limited to six in this study. In other words, the number of stock sheet sizes available for each individual BOM is limited to 2, 3, 4, 5, or 6. In fact, it is a typical situation in manufacturing that a small number of stock sheet sizes are kept in inventory in order to limit storage and manufacturing costs. Therefore, the actual number of categories involved in this test is 5x5=25. Table 4.2 gives the details of the 25 categories. It can be seen from this table that categories 1 to 5 have the same APSA ratio (1.00) but there are five different stock sheet sizes (2 to 6). Similarly, categories 6 to 10 have an APSA ratio of 0.50 and, again 2 to 6 stock sheet sizes. Categories 11 to 15 have an APSA of 0.25. The lowest APSA considered in categories 21 to 25 is 0.04.

Table 4.2. Classification of the Test Categories.

Category No.	APSA	No. of Stock Sheet Sizes
1	1.00	2
2	1.00	3
3	1.00	4
4	1.00	5
5	1.00	6
6	0.50	2
7	0.50	3
8	0.50	4
9	0.50	5
10	0.50	6
11	0.25	2
12	0.25	3
13	0.25	4
14	0.25	5
15	0.25	6
16	0.10	2
17	0.10	3
18	0.10	4
19	0.10	5
20	0.10	6
21	0.04	2
22	0.04	3
23	0.04	4
24	0.04	5
25	0.04	6

4.4.2 Random Problem Generator

A procedure for randomly generating problems is developed. The generation of the test problems includes both the creation of the BOMs and the stock sheet data sets. The procedure uses a framework similar to that described in [39]. The main difference is that the current procedure provides data for two-dimensional, multiple sized stock sheets rather than for a single three-dimensional container. The procedure includes the following steps.

Step 1

Input the following parameters:

1. target area, A_c, a parameter to control the size of the BOM or the stock sheet set;
2. the number, n, of different rectangular piece sizes available;
3. lower and upper limits on all the length and width dimensions, a_j, b_j, j=1,2, of the rectangular order pieces for the BOM or the whole set of rectangular stock sheets;
4. limit of the aspect ratio, L, i.e. the ratio of a order piece or stock sheet's length to width.

Step 2

The random number generator is initialized. An important utility that digital computer systems should provide is the ability to generate random numbers. A Prolog function, Random [37], is selected as the random number generator. The numbers generated are used as the seed numbers to create the rectangles'[9] dimensions.

Step 3

An integer n different sizes of rectangles are created. Each rectangle's length and width are generated by using random seed numbers, the lower and upper limits, a_j and b_j, as well as the aspect ratio limit, L. The available number of each rectangle is set to one at this step and the total area of all the rectangles (A) is calculated.

[9] A rectangle is a rectangular order piece or stock sheet.

Step 4

The total area of all the rectangles is compared with the target area A_c. If $A \geq A_c$, the procedure is terminated. If $A < A_c$, more rectangles are required to reach the target area. Because the number of rectangle sizes, n, and their dimensions are generated already in Step 3, the only way to enlarge A is to increase the numbers of the rectangles. Therefore, the quantity of the rectangles that belong to one of the n different sizes is increased by one. The selection of the size to be increased is random. A size indicator k (k=1,...,n) is set by a random number that is produced, again, by using the random number generator. The area of this new rectangle is added into A. Then step 4 is repeated until the target area is reached.

Figure 4.5 contains a flowchart detailing the different steps involved in generating a set of rectangles. It can be seen that the total area, A, of all the rectangle cannot exceed the target area, A_c. However, A will be close to A_c. The procedure tends to lead to approximately similar numbers, q_i, for the different sized rectangles. Here q_i represents the quantity of rectangle i (i=1,.., n), for the different rectangle sizes. Minor modifications involving the introduction of weighting factors into the formula defining indicator, k, for the size of a rectangle would allow rectangles to be generated in other (user-specified) proportions. In other words, by controlling the assignment of k during each iteration, the quantity of rectangles of size k could be controlled. The first 10 randomly generated numbers in any sequence are always discarded to ensure that similar seed values do not lead to similar random numbers.

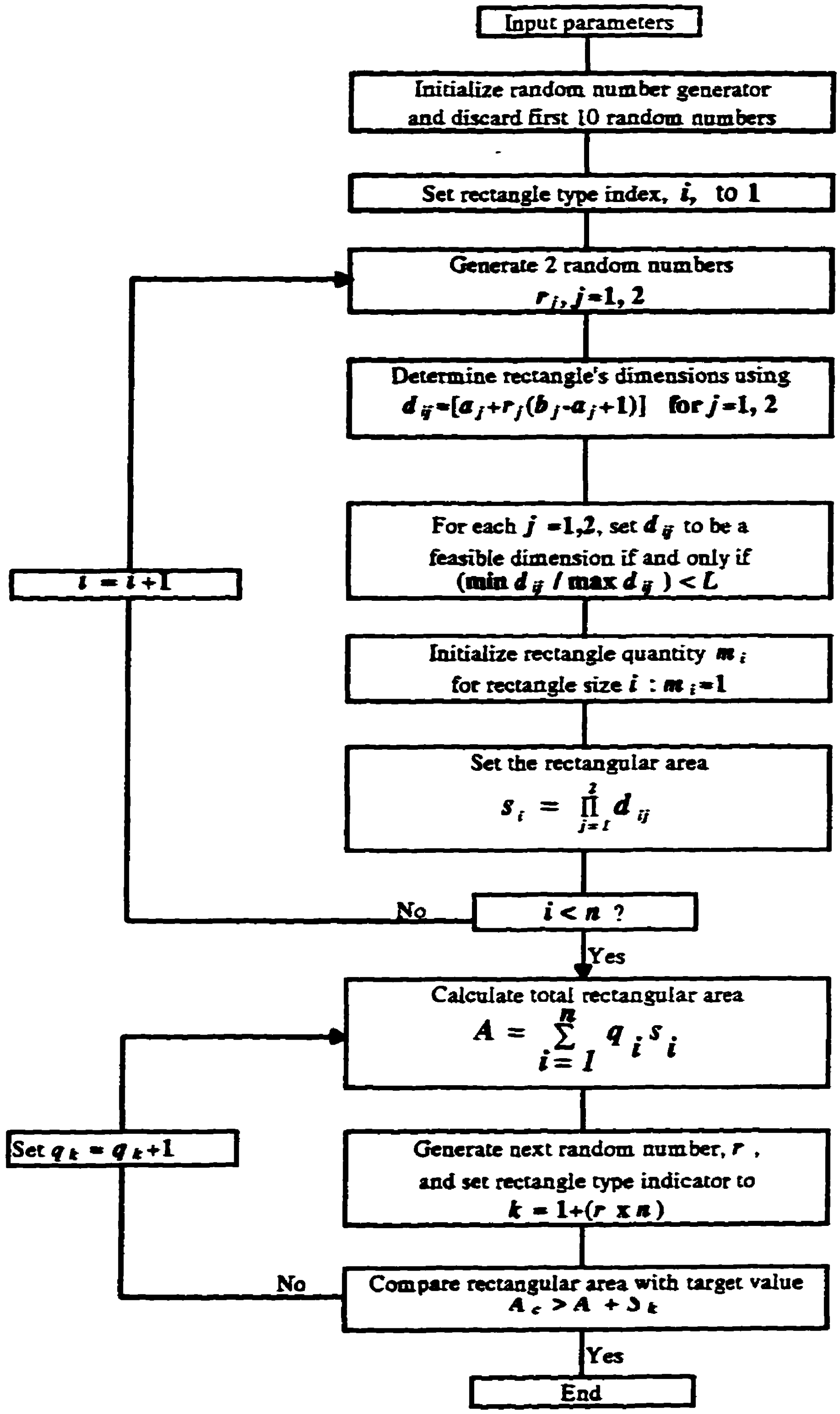

Figure 4.5. Flowchart for the Random Problem Generator.

For the generation of stock sheet data set, the following two factors must be taken into account.

1. The smallest length and width of any stock sheet size must be larger than the largest length and width, respectively, of any order piece in the BOM.
2. The required APSA ratio must be defined to generate a set of stock sheet data.

Five test problems are considered for each APSA category. Therefore, a total of 125 test problems are created. The BOMs generated vary in their structure. The number of different sizes demanded in the BOMs ranges from 8 to 50. The quantity for each of these sizes ranges from 1 to 20 pieces. The total number of order pieces in the BOMs varies from 10 to 205. The number of stock sheet sizes is between 2 to 6. A stock sheet's length and width ranges from 25 to 120 (arbitrary distance) units. Within the same APSA category, the standard deviation of the stock sheet areas is in the range of 10 to 50% of the overall mean stock sheet area. An order piece's length ranges from 1 unit to the stock sheet's width. The quantity of order pieces in a BOM is almost equal with a slight variance of one or two pieces for each size. However, the number of different order piece sizes varies from BOM to BOM.

4.4.3 Test Results for the PSTL Procedure

The PSTL procedure described previously was tested by using test problems created by the random problem generator. The procedure was tested four separate times. Each time one of the following four different heuristic polices was implemented in the PSTL routine.

1. The first heuristic is LTM-MEAN in which the Least Total trim loss Method (LTM) is used to select the BSS. The arithmetic mean (MEAN) is used to calculate the PSTL.
2. The second heuristic is LTM-MEDIAN which uses the LTM but, in this case, the median (MEDIAN) is used for the PSTL calculation.
3. The third heuristic is LAM-MEAN in which the Largest Area Method (LAM) is used for the BSS selection. The mean is used for the PSTL calculation.
4. The fourth heuristic is LAM-MEDIAN which uses LAM for the BSS selection and MEDIAN for the PSTL calculation.

The test results for each of the four different heuristics are presented in Figures 4.6 through 4.9. An abscissa in these figures represents the category numbers in which the 25 categories are divided into the 5 sections denoted by (a) through (e). Each section includes 5 different categories that have the same APSA ratio but a different (2 to 6) number of stock sheet sizes. Figure 4.6 shows the Average (percentage) Stock Sheet Utilization

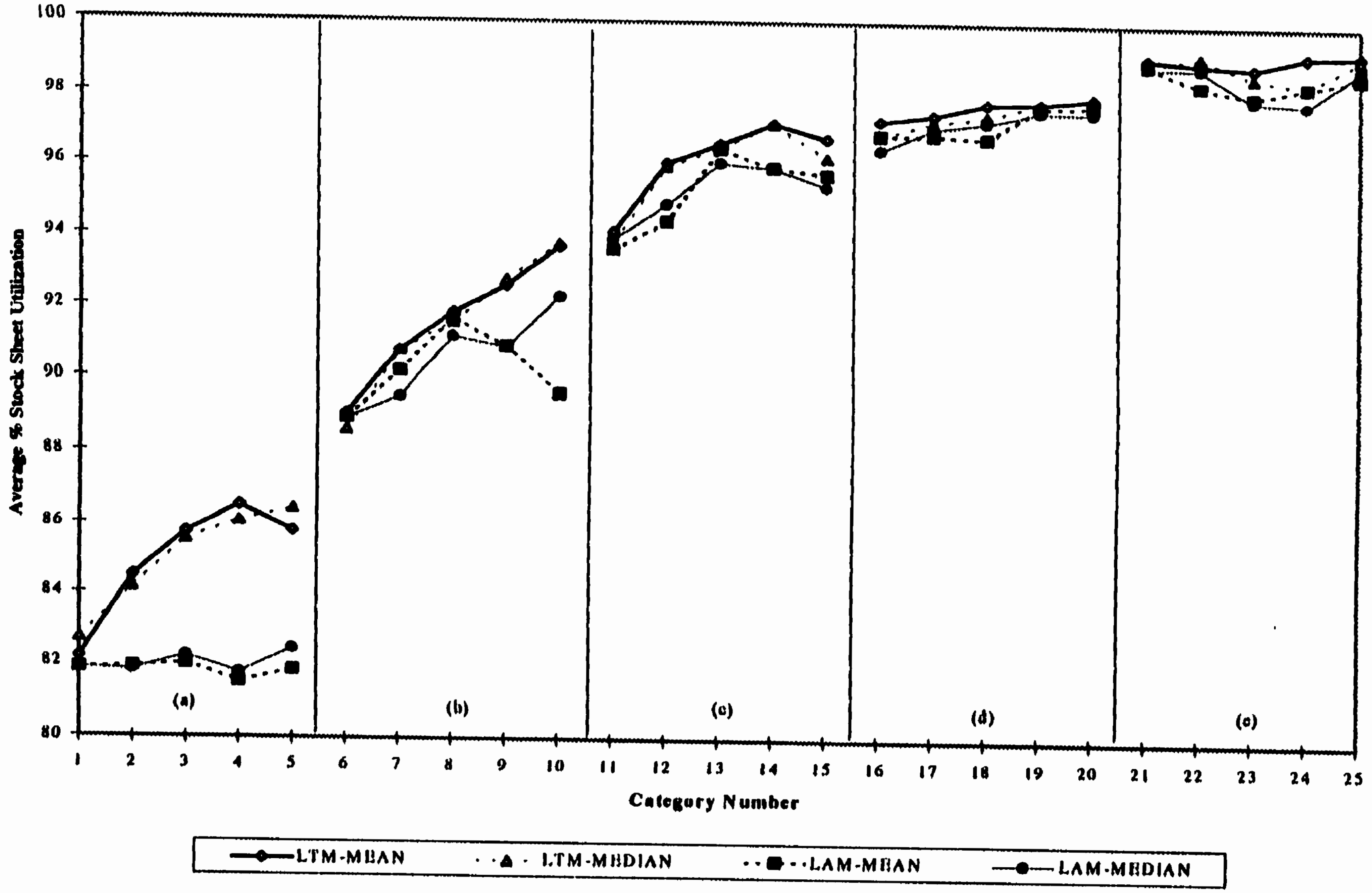

Figure 4.6. Average Sheet Utilizations for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

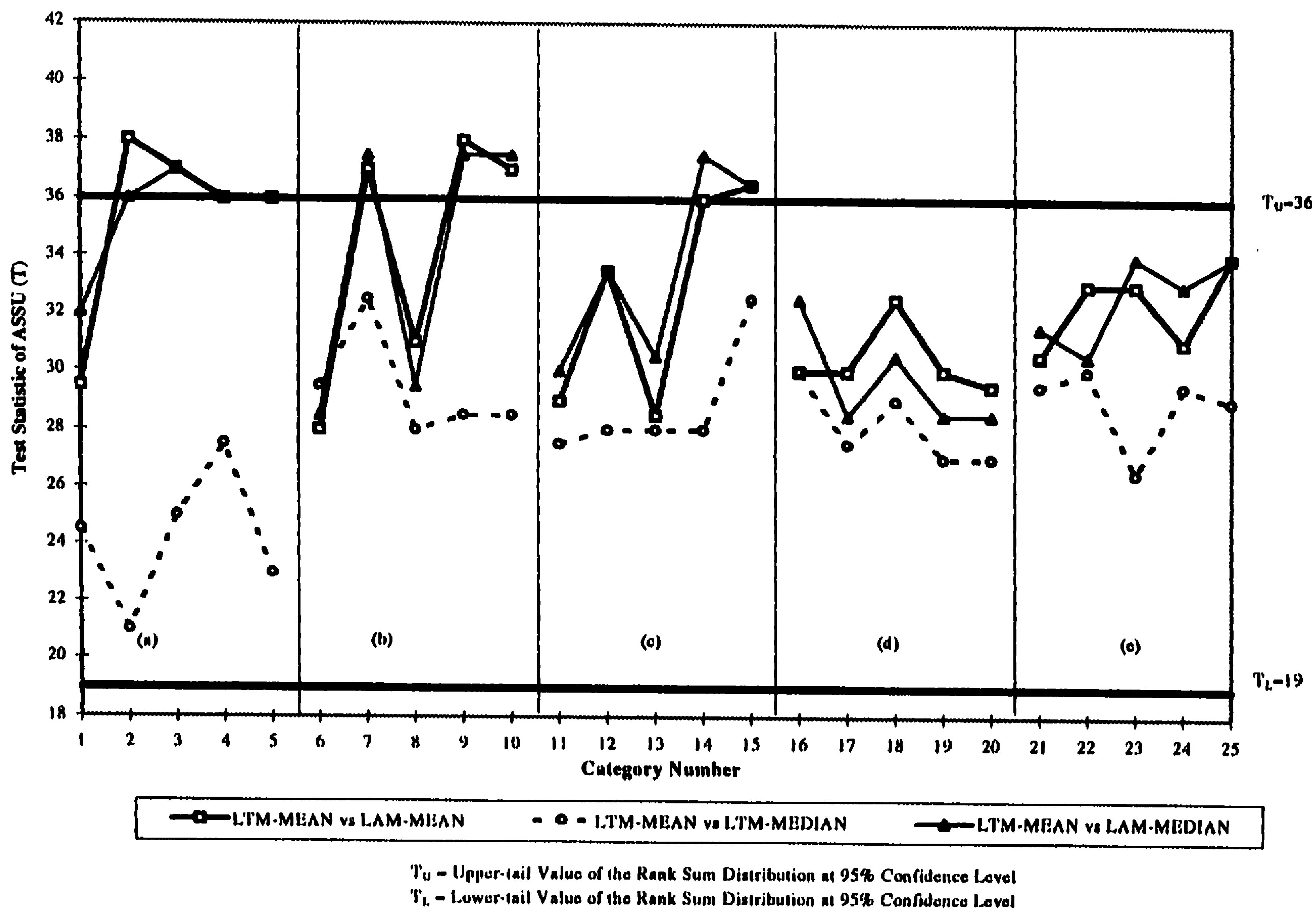

Figure 4.7. Wilcoxon Rank Sum Test of Average Sheet Utilizations for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

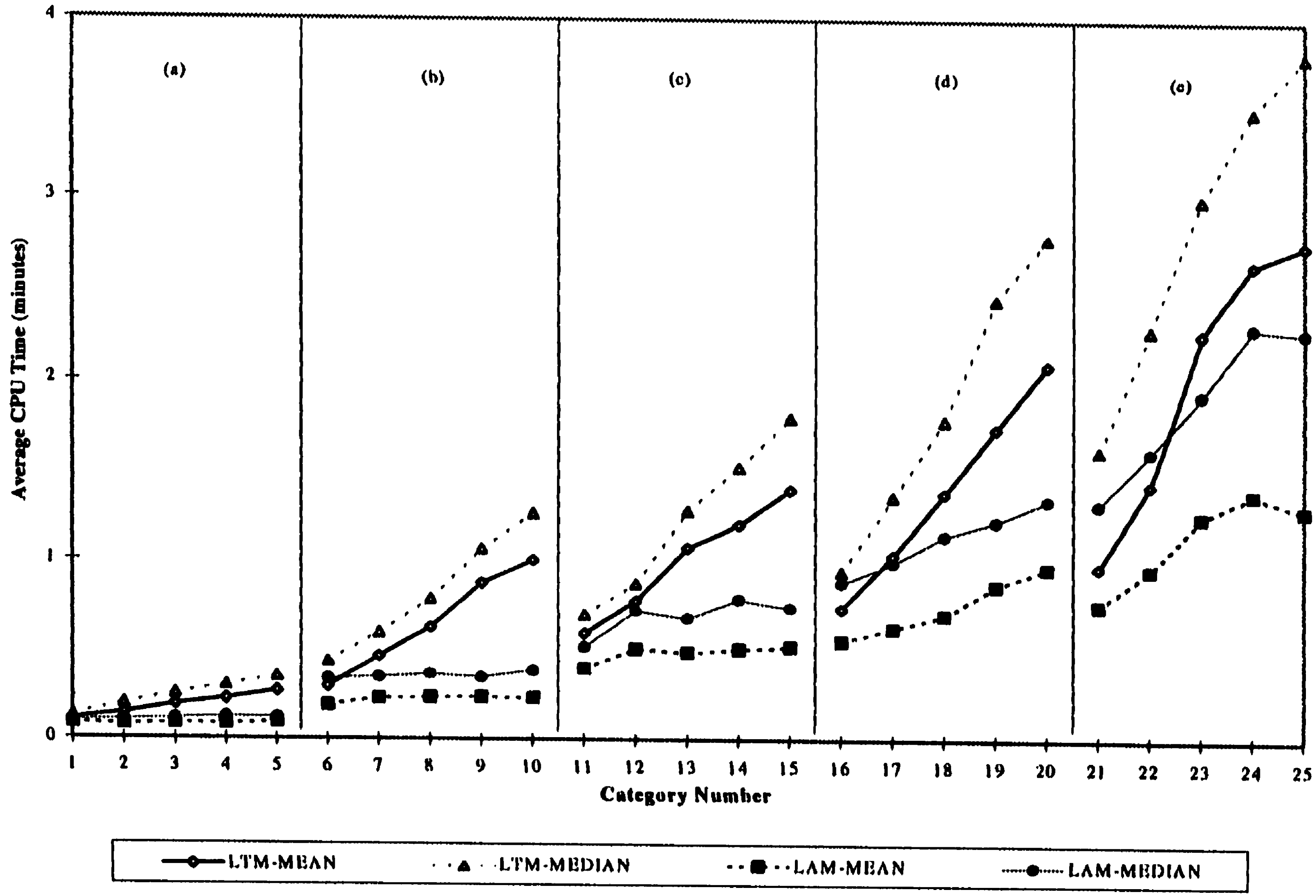

Figure 4.8. Average CPU Times for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

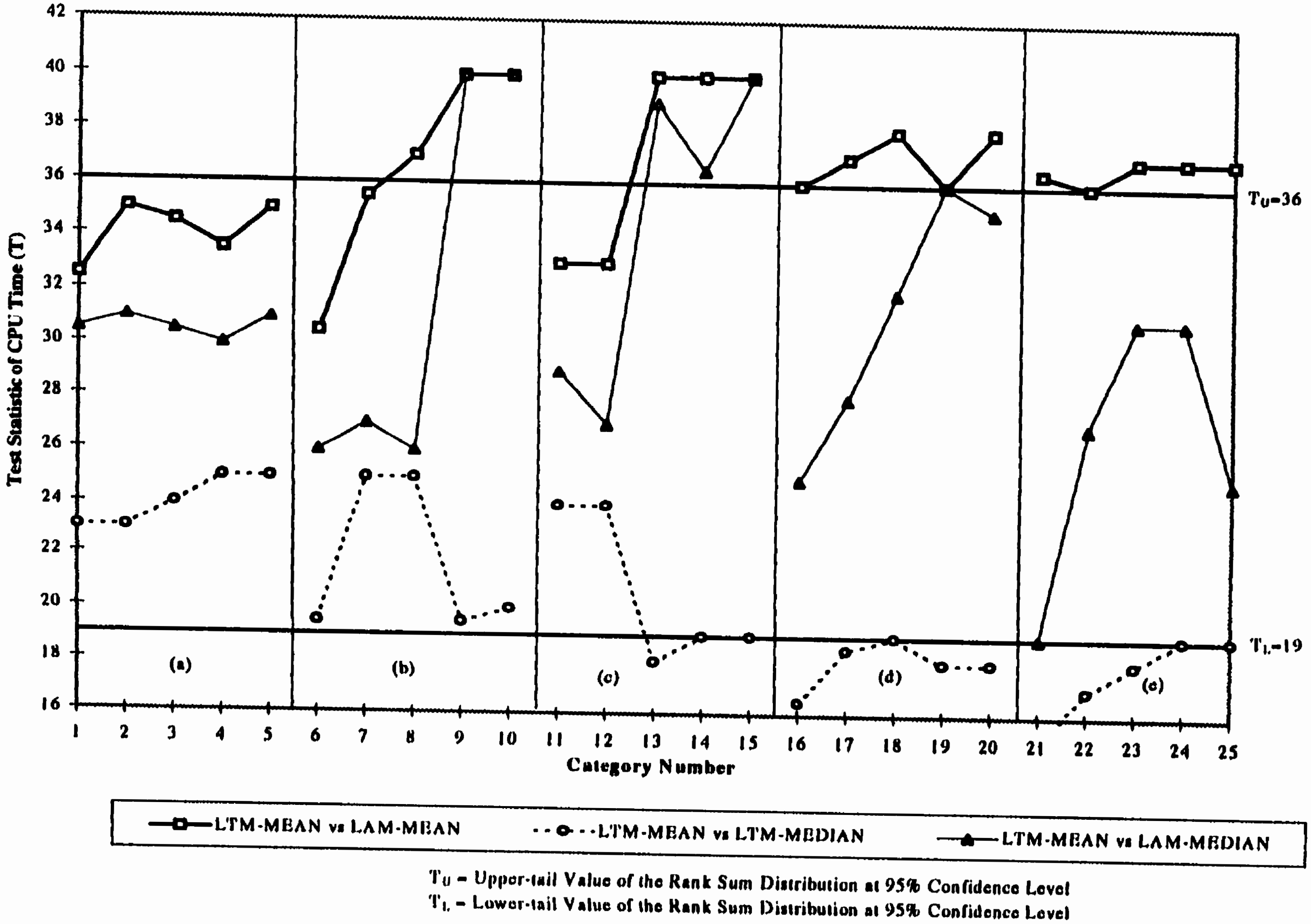

Figure 4.9. Wilcoxon Rank Sum Test of Average CPU Times for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

(ASSU) in each category. Figure 4.7 presents the results of the Wilcoxon rank sum test [40] on the sample data of the ASSU. The Wilcoxon rank sum test is a nonparametric statistical technique that compares the bias of two test methods. It determines if the difference between the results produced by the two methods is significant (two-tailed test) or identifies the better of two methods (one-tailed test). In this chapter, the two-tailed test was conducted. Figures 4.8 and 4.9 give the corresponding mean values of the CPU times and the results of the Wilcoxon rank sum test. The standard deviations are not compared in this test because it is believed that a small sample size (5) may not provide meaningful results. The original test results of all data sets for the four heuristic policies are presented in Tables C.1 through C.25 in the Appendix C. It is noted that only five data sets are tested in each category.

4.4.4 Analysis of Results

The following observations and analysis are presented based on the data shown in Figures 4.6 through 4.9.

1. Figure 4.6 shows a comparison of the ASSU for the four heuristic policies. It can be seen from this figure that the ASSU generally increases as the APSA ratio decreases, regardless of the heuristic variant used. In those categories that have the same APSA ratio, the ASSU grows as the number of different stock sheet sizes increases. This trend happens because, as the APSA ratio decreases, the mean area of the piece to be

allocated from the BOM decreases in relation to that of the stock sheet. Hence, a better utilization can be expected. It is obvious that, the smaller is the order piece, the better this piece can fit into the smaller available space on the stock sheet. When the APSA ratio remains constant, the greater is the number of different stock sheet sizes that are available, the greater is the possibility that a smaller trim loss layout can be found.

2. The results presented in Figure 4.6 indicate that the two variants for the BSS selection, namely LTM and LAM, have quite different impacts on the ASSU. LTM generally achieves a higher ASSU than LAM. This holds true particularly for those categories with a high APSA ratio and a large number of different stock sheet sizes. The distinction between LTM and LAM becomes less significant as the APSA ratio and the number of different stock sheet sizes decrease. This behavior can be explained by the main feature of LTM and LAM, that is, the way that the BSS is selected. LTM examines all the different stock sheet sizes and searches for the BSS by laying out the BOM on each single sized stock sheet so that the most favored stock sheet size can be found. Unlike LTM, LAM does not check all the stock sheet sizes against the BOM. It simply chooses the stock sheet size corresponding to the largest area as the BSS. In other words, it presumes that the stock sheet having the largest area produces the least total trim-loss. The chance of this happening depends upon the different categories of the test problems. It is more likely true in categories involving few different stock sheet sizes, say 2 or 3. It also has a higher possibility to be true in the categories

having a low APSA ratio, say 0.04 or 0.10. For low APSA ratios, the larger is the stock sheet, the less trim loss could be achieved. Figure 4.6 also shows that the two variants for the PSTL calculation, namely MEAN and MEDIAN, have little difference on the ASSU. Consequently, the method of calculating the PSTL has little effect.

3. Figure 4.6 only shows the mean values of the stock sheet utilizations. Therefore, it is necessary to conduct a significance test between these variants by using a Wilcoxon rank sum test [40]. A conclusion of whether two methods produce significantly different results is based on a "test statistic", T. Details of computing T are given more conveniently in Appendix A. The T is compared with two critical values, T_U (upper-tail value of the rank sum distribution), and T_L (lower-tail value of the rank sum distribution). If $T \leq T_L$ or $T \geq T_U$, it can be concluded that the two methods generate statistically different results. Otherwise, they are not statistically different. The critical values can be found in standard statistical tables [40] by relating them to the sample space and the confidence level (95% in this test). Figure 4.7 presents the results of the Wilcoxon rank sum test for the ASSU between two of the four heuristic policies, namely, LTM-MEAN and LTM-MEDIAN, LAM-MEAN and LAM-MEDIAN. The critical values, which are based on a 5 sample space and a 95% confidence level, are $T_L = 19$ and $T_U = 36$ [40]. Statistically, there is invariably no difference, according to Figure 4.7, between LTM-MEAN and LTM-MEDIAN because all the test statistics of the two heuristics are between the two critical values. However, the difference between LTM-MEAN and LAM-MEAN or LAM-MEDIAN varies in different

categories. Generally, they are not significantly different at low APSA ratio categories (category 16-25). Conversely, they are different in high APSA ratio categories (category 1-15) except when few (2 or 3) stock sheet sizes are available.

4. The average CPU time taken on a Pentium 90 based, IBM compatible microcomputer is shown in Figure 4.8 for all the four heuristic policies and each category. It can be seen that LTM generally demands more computer time than LAM, especially in the categories with a low APSA ratio (less than 0.25) and many (greater than 3) different stock sheet sizes. This is because all the different stock sheet sizes are examined when finding the BSS by using LTM. With the same BSS selection, the use of MEDIAN for calculating the PSTL requires more CPU time than that of MEAN. This is because it is somewhat more complicated to find the PSTL for the MEDIAN than for the MEAN. Differences between CPU times are insignificant at a high APSA ratio (around 1.00) but they are exaggerated as the APSA ratio decreases, especially for more than 3 different stock sheet sizes.

5. Figure 4.9 presents the results of the Wilcoxon rank sum test for the average CPU times. This figure shows that the test statistics for LTM-MEAN and LTM-MEDIAN, LAM-MEAN, LAM-MEDIAN are between the T_L an T_U at high APSA ratio categories (category 1-8). (The critical value is based again on a 5 sample space and a 95% confidence level.) As the APSA ratio decreases, most of the test statistics are greater than the critical value T_L or less than the T_U. This indicates that, at higher

APSA ratio categories, there is no significant difference between LTM-MEAN and the other heuristics. However, there is a significant difference at APSA ratios below about 0.25.

4.4.5 Conclusions

Based on the test results, the following conclusions can be drawn.

1. The APSA ratio and the number of different stock sheet sizes are two major factors that should be considered in implementing heuristics for the PSTL procedure. Generally speaking, the ASSU and average CPU time of the PSTL procedure increase as the APSA ratio decreases or as the number of different stock sheet sizes increases, regardless of heuristic policy.
2. Depending on the APSA ratio and the number of different stock sheet sizes, the four variants (i.e. LTM or LAM for the BSS selection and MEAN or MEDIAN for the PSTL calculation) differ in their effectiveness. In terms of the ASSU, LTM is more effective than LAM at APSA ratios above about 0.50, especially for situations involving more than 3 different stock sheet sizes. The difference in ASSU becomes less significant as the APSA ratio increases. LTM demands more CPU time than LAM. However, the increase in CPU time is not significant at APSA ratios above 0.25, especially for 2 or 3 different stock sheet sizes. On the other hand, the difference becomes more apparent as the APSA ratio decreases. Therefore, LTM

should be used at APSA ratios above 0.10 and LAM should be chosen at APSA ratios below 0.10.

3. The variants of MEAN and MEDIAN generate little difference in the ASSU but they are quite different in their demands for CPU time. MEAN achieves a slightly higher ASSU than MEDIAN but requires much less CPU time. Therefore, MEAN should be used for a PSTL calculation.

In conclusion, for the final design of the PSTL procedure, the hybrid heuristic policy of LTM-MEAN should be chosen at APSA ratios above 0.10, especially for fewer than 3 different stock sheet sizes. Conversely, LAM-MEAN should be chosen at APSA ratios below 0.10 and for more than 3 different stock sheet sizes.

4.5 Summary

A new procedure was developed and evaluated in this chapter. The procedure used the BSS to limit the scope of the search for 'good' basic stock sheet sizes. The PSTL was employed to dynamically control the selection of the stock sheets in order to reduce the total trim loss. The new procedure was tested on various problems that were created by using a random test problem generator. Four different heuristic policies, namely LTM-MEAN, LTM-MEDIAN, LAM-MEAN, and LAM-MEDIAN were implemented and tested. Results were compared in terms of the average stock sheet utilization and the average CPU time. It was found that LTM-MEAN performed the best at APSA ratios above about 0.10 and for 2 or 3 different stock sheet sizes. Conversely, LAM-MEAN is the best choice for APSA ratios below about 0.10 and for more than 3 different stock sheet sizes.

Chapter 5. Analysis of the Pre-specified Trim-loss Procedure

This chapter describes and evaluates the final form of the PSTL procedure which was discussed in the previous chapter. The objective is to design a procedure in which the conflicting demands for both a "high quality" solution and minimal computational effort can be satisfied. The development of the final form is briefly described first. Then the performance of the PSTL procedure is compared with that of the three previously published procedures discussed in Chapter 3.

5.1 Final Design of the PSTL Procedure

According to the evaluation presented in the previous chapter, two of the four heuristic policies, namely LTM-MEAN and LAM-MEAN, are superior in terms of both ASSU and CPU time. Based on this general conclusion, the detailed form of the PSTL procedure is designed as follows.

1. LTM-MEAN is always chosen, regardless of the APSA ratio if the number of different stock sheet sizes is fewer than three. When only one or two different stock

sheet sizes are available, LTM-MEAN achieves a higher ASSU and requires only slightly more CPU time than LAM-MEAN. The difference in the CPU times between these two policies is insignificant according to the test results presented in the previous chapter.

2. If the number of different stock sheet sizes is larger than two, the choice between LTM-MEAN or LAM-MEAN depends upon the APSA ratio. Test results given in the previous chapter indicate that, as the APSA ratio decreases below 0.10, the difference between LTM-MEAN and LAM-MEAN becomes insignificant in terms of ASSU. However, the difference in the CPU times needed to implement these two policies is significant. LTM-MEAN requires much more CPU time than LAM-MEAN. In other words, when the APSA ratio is less than 0.10, it is more advantageous, on balance, to use LAM-MEAN. The 0.10 is chosen as a critical value to control the selection of the heuristic policy.

Figure 5.1 illustrates a simple logic for the selection of the heuristic policy.

5.2 Testing the PSTL Procedure

The PSTL procedure, implemented as described in the previous section, is evaluated next by comparing its performance with the three previously published procedures described in Chapter 3. They are the PREORDER, STEP, and Branch and Bound (B-B) procedures [34]. The comparison is based upon the results achieved by the PSTL and the three existing procedures for the same test categories utilized in the last chapter.

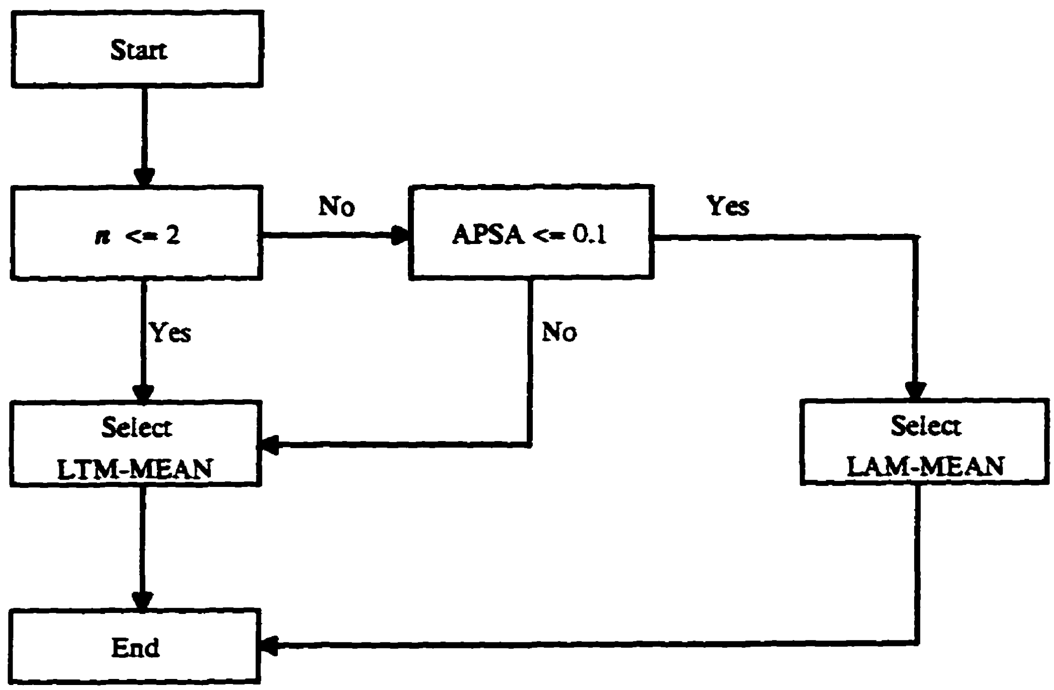

Figure 5.1. Selection of the Heuristic Policy.

5.2.1 Test Problems

The test problems used to compare the PSTL and the three existing procedures take the same classification and characteristics as those described in Chapter 4. That is, there are 25 categories of test problems with five different APSA ratios (0.04 to 1.00) and five sets of different stock sheet sizes. Each set includes a particular number of stock sheet sizes. The number of different stock sheet sizes ranges from 2 to 6. There are 30 problems for each category. The mean values of the test results (the ASSU and the CPU times) are used to evaluate the procedures. In other words, the performance of the procedures is compared based on the 30 sample problems for each category. The question to ask is whether a sample space of 30 is large enough to draw conclusions about the entire population? The answer is positive. According to the Central Limit Theorem of statistics [41]: "The distribution of the sample mean, $\bar{X}$, of a random sample drawn from practically any population with mean μ and variance σ^2 can be approximated by means of a normal distribution with mean μ and variance σ^2/n, providing the sample size n is large". The key aspect in this theorem is that the distribution of the sample mean must be normal. In most instances the tendency towards normality is so strong that the approximation is fairly satisfactory with a sample of about 30 [41]. With larger samples, of course, the approximation is even more satisfactory. However, larger samples lead to more computational effort. Therefore, 30 problems are tested for each category. All the test problems are created by using the random problem generator. As described in Chapter 3, the three existing procedures have been reprogrammed for an IBM compatible microcomputer so that they can be compared directly with the PSTL procedure on the same computer system. The computer system, again, is an IBM compatible computer having a 90 MHz Pentium processor and 16 MB of RAM.

5.2.2 Results

Test results are presented in terms of the Average (percentage) Stock Sheet Utilization (ASSU) and CPU time for each category. For the PSTL, PREORDER and STEP procedures, the results of all 25 categories are presented. However, for the B-B procedure, only the results in categories involving two or three different stock sheet sizes are given. It was found that the B-B procedure couldn't be tested on the computer system if the number of different stock sheet sizes exceeded three. This is because significantly more computer memory is required for the B-B procedure. Therefore, the comparison of the PSTL and B-B procedures is limited to two and three different stock sheet sizes. The original test results of all data sets for the PSTL, PREORDER and STEP procedures are presented in Table C.1 through C.25 in Appendix C.

5.3 Comparison With the PREORDER and STEP Procedures

Figures 5.2 through 5.7 compare the results of the PSTL procedure with those of the PREORDER and STEP procedures. Figure 5.2 compares all three procedures for each of the 25 categories in terms of the ASSU. The comparison of the standard deviations of the ASSU is presented in Figure 5.3. Figure 5.4 gives the results of testing the difference of the ASSU between the PSTL and STEP or PREORDER procedures by using independent sample data. A conclusion of whether two procedures produce a significant

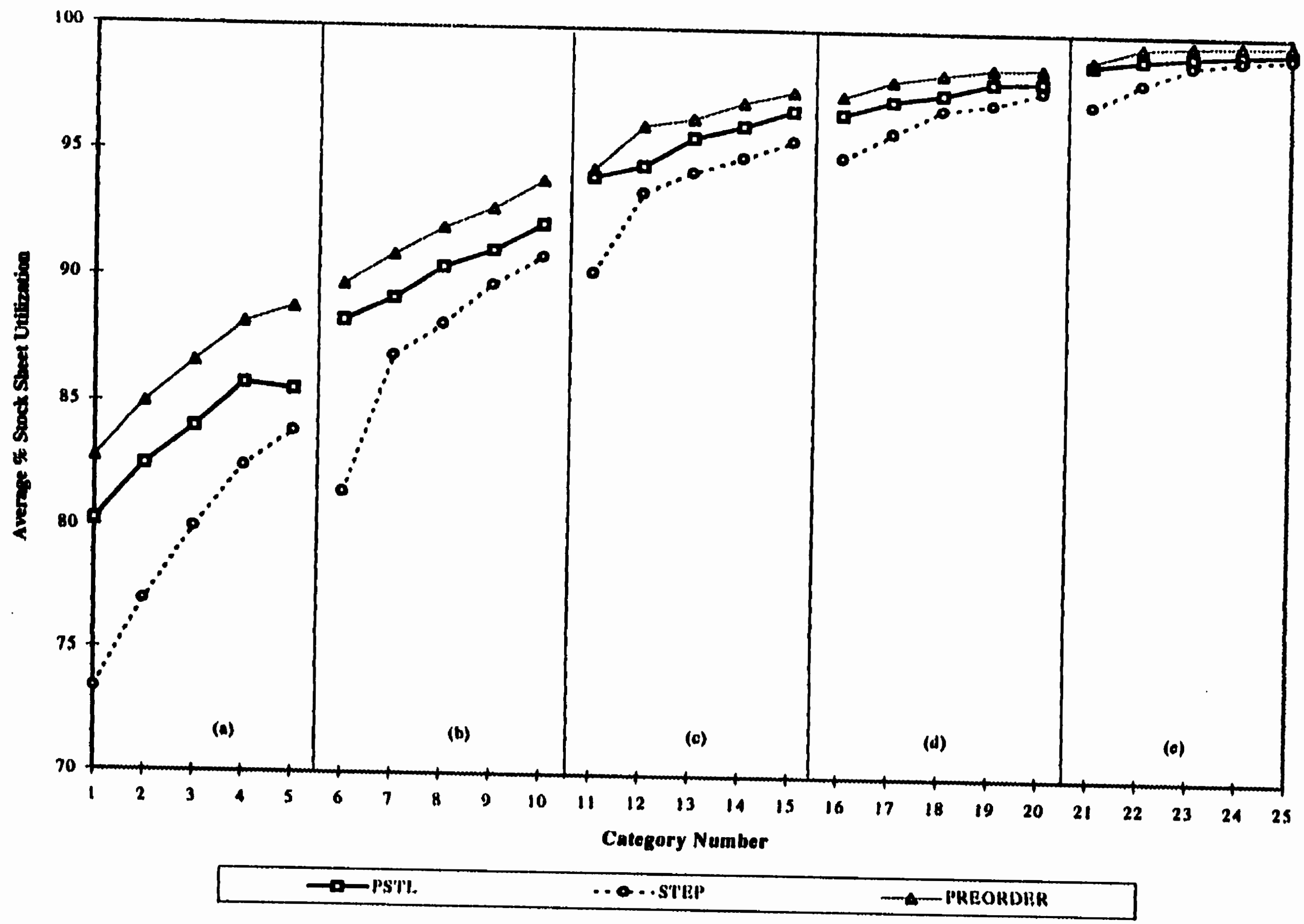

Figure 5.2. Average Sheet Utilizations for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

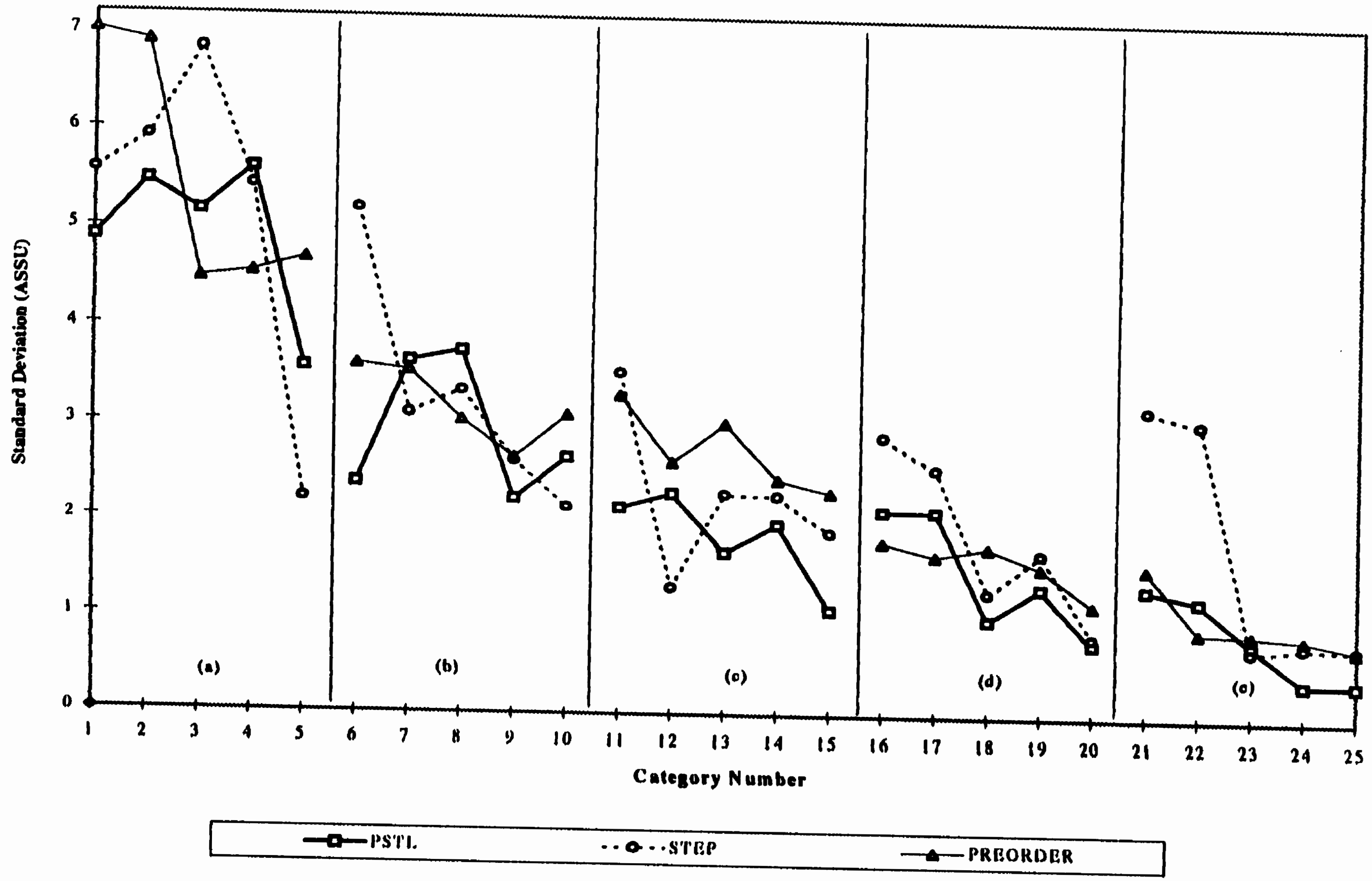

Figure 5.3. Standard Deviations of the ASSU for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

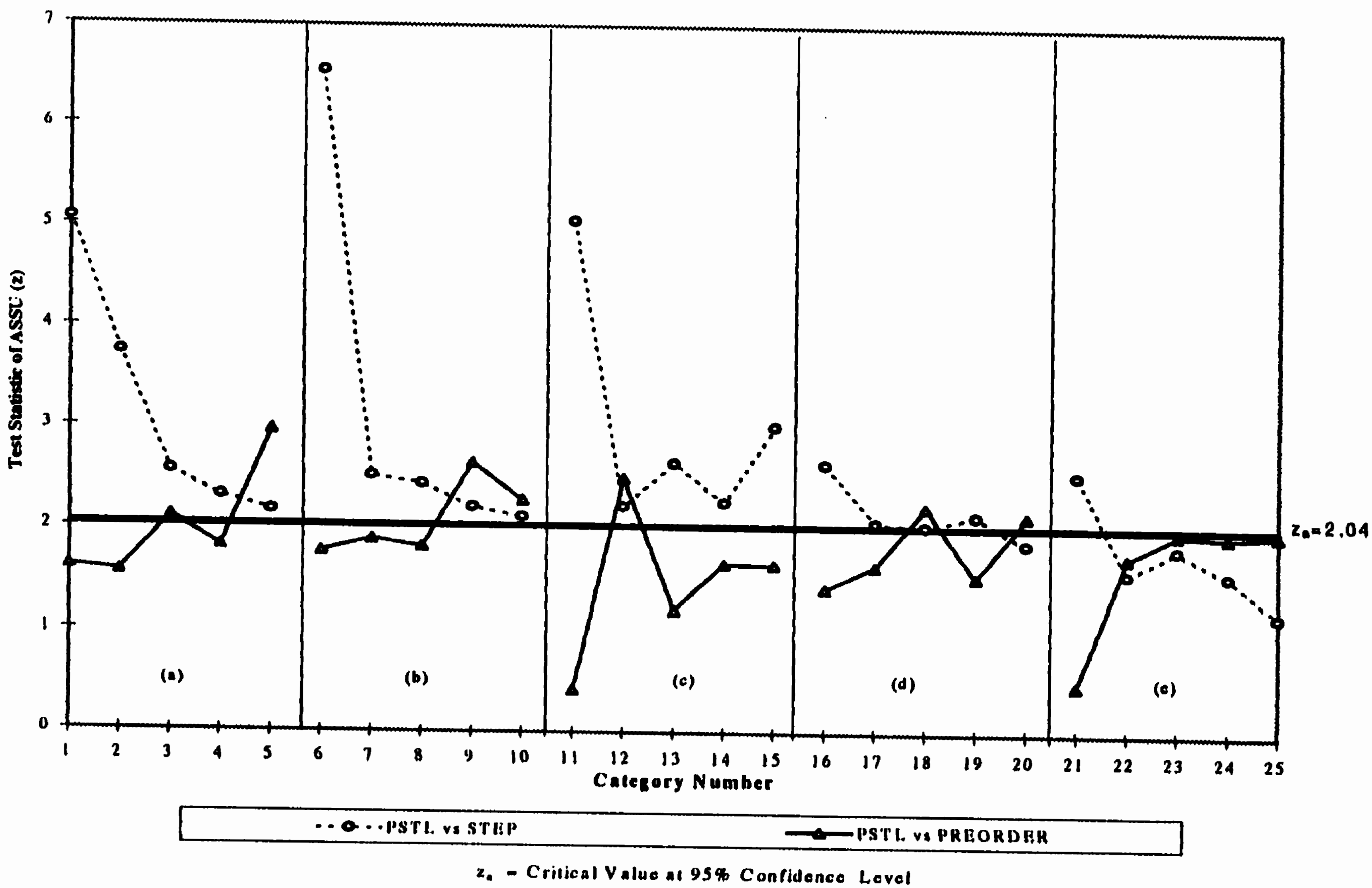

Figure 5.4. Difference Between the ASSU for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

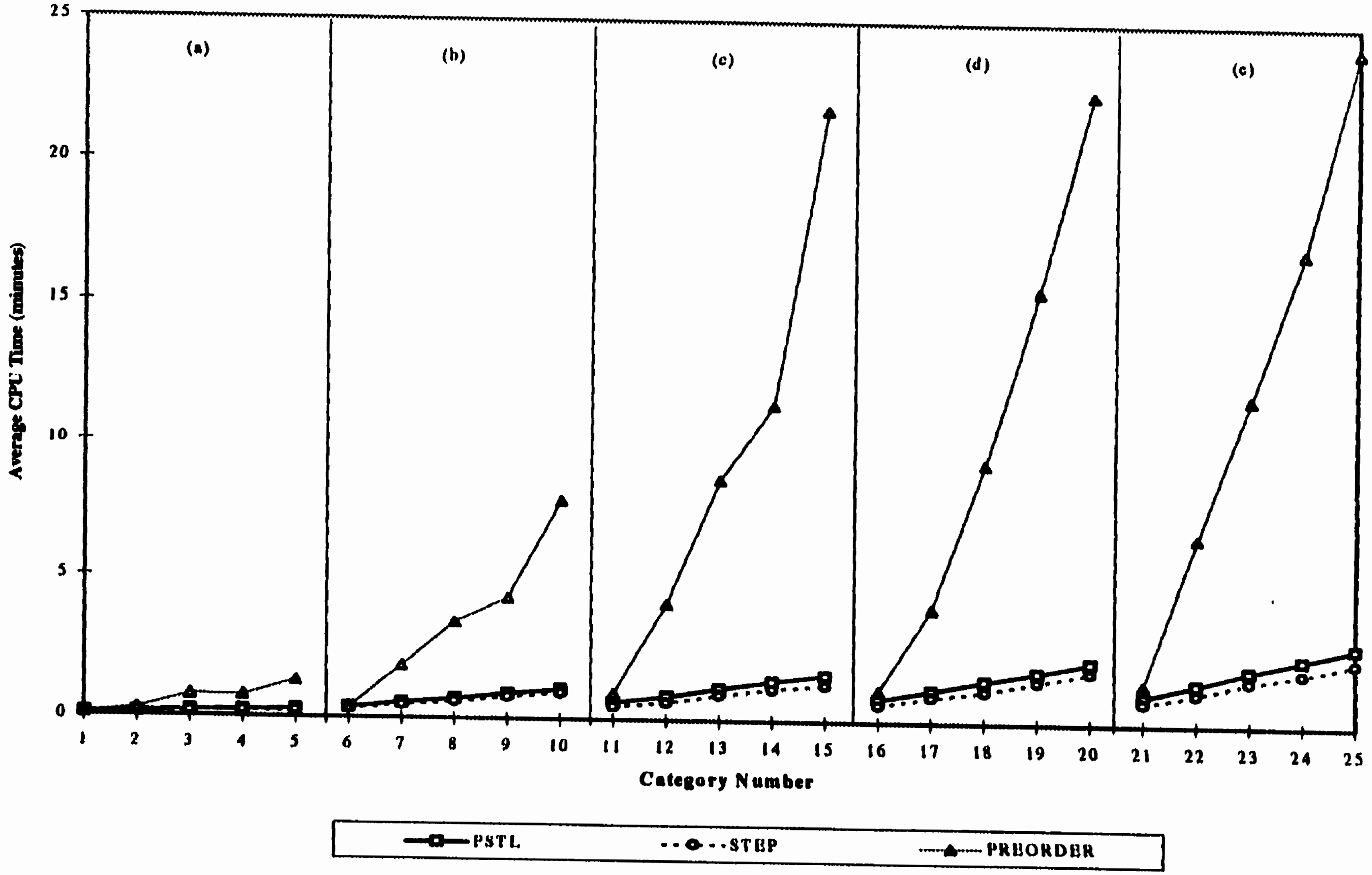

Figure 5.5. Average CPU Times for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

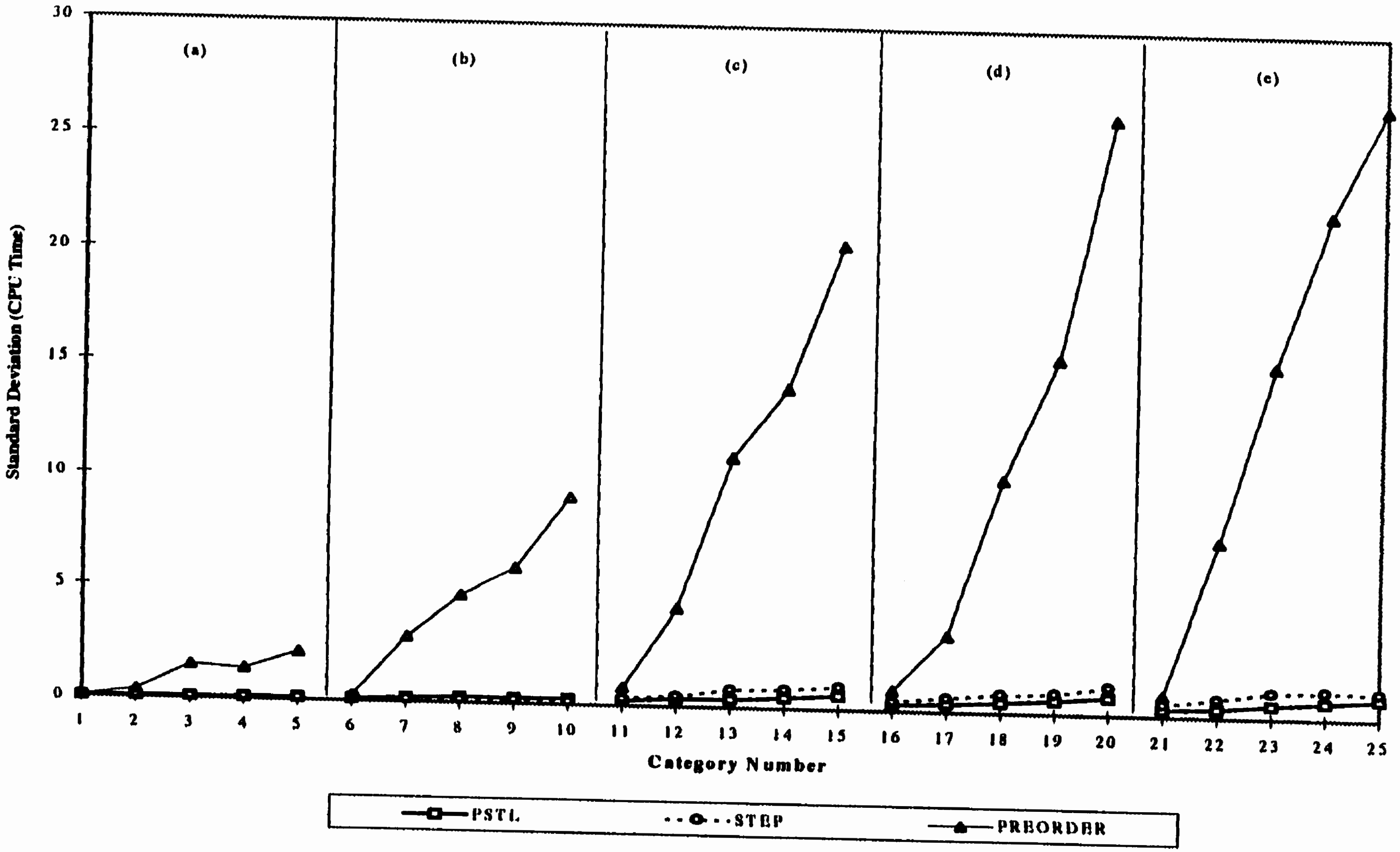

Figure 5.6. Standard Deviations of the average CPU Times for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

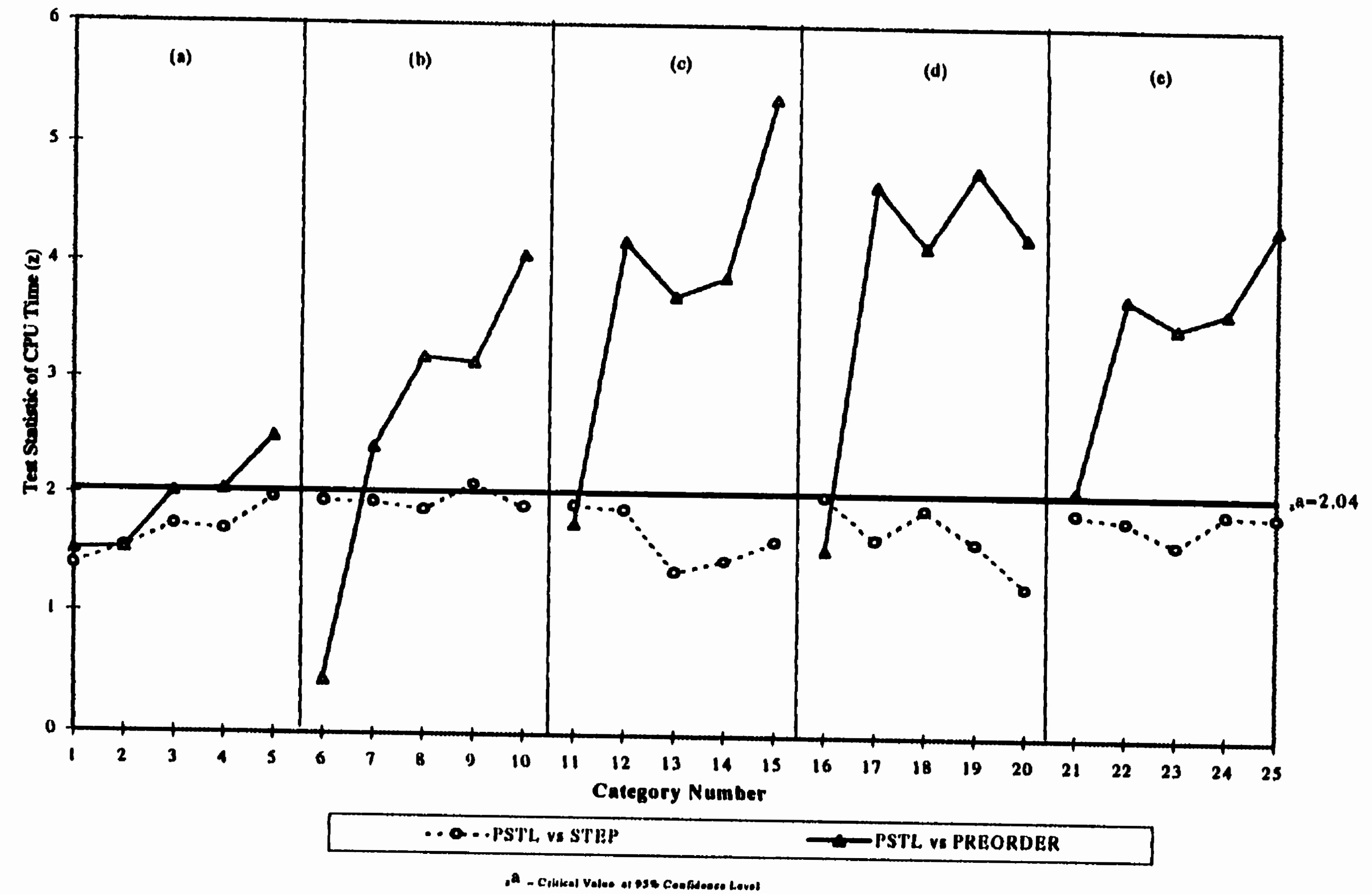

Figure 5.7. Difference Between the Average CPU Times for APSA Values of (a) 1.00, (b) 0.50, (c) 0.25, (d) 0.10 and (e) 0.04.

difference in the ASSU is based on the comparison of the test statistic, z, and the critical value z_α. The calculation of z is given in Appendix A. The z_α, which is based on the sample space and the confidence level, can be found in standard statistical tables given in reference [40]. Figure 5.5 compares all three procedures in terms of the average CPU times for each category. Figure 5.6 shows the comparison of the standard deviations of the average CPU times. Figure 5.7 presents the results of testing the difference of the average CPU times. The results shown in these figures highlight the following important observations.

1. Figure 5.2 indicates that the ASSU generally increases as the APSA ratio decreases, regardless of the procedure. In categories that have the same APSA ratio, the ASSU grows as the number of different stock sheet sizes increases. It can also be seen from Figure 5.2 that the ASSU of the PSTL procedure is invariably higher than that of the STEP procedure but not as high as the PREORDER procedure. The differences are quite obvious at APSA ratios greater than about 0.25 but become less significant as the APSA ratio decreases. This behavior indicates that the PSTL procedure is better than the STEP procedure but not as good as the PREORDER procedure in terms of the ASSU.

2. The standard deviation is a useful parameter to compare the different procedures. The results presented in Figure 5.3 indicate that the standard deviation decreases as the APSA ratio decreases, regardless of the procedure. The three procedures generally

give comparable standard deviations except that the STEP procedure produces a high standard deviation for the Categories 1, 6, 11, 16 and 21 involving small number of stock sheet sizes. In other words, there may be more volatility with STEP in these categories. This observation is confirmed by Figure 5.4 in which the difference of the ASSU generated by the PSTL and STEP procedures is quite significant in those categories.

3. Figure 5.4 shows the results from testing the difference of ASSU. It reveals that the difference between the ASSU from PSTL and PREORDER is not statistically significant except for categories 3, 5, 9 10, 18 and 20 because the statistic (z) is greater than the critical value z_α (2.04) which corresponds to a 30 sample space and a 95% confidence level [40]. Conversely, the difference in the ASSU from the PSTL and STEP procedures is significant except in the low APSA ratio categories 20, 22, 23, 24 and 25 because the statistic (z) is then less than the critical value z_α.

4. Figure 5.5 shows the average CPU times taken by the three procedures. It can be seen that the PSTL procedure always demands slightly more average CPU times than the STEP procedure. However, the difference between these two procedures is insignificant. This behavior is confirmed by testing the difference of the average CPU times shown in Figure 5.7. It can be seen from this figure that the test statistics z between the PSTL and STEP procedures are below the critical value z_α of 2.04 which corresponds to a 30 sample space and a 95% confidence level [40]. In other words,

there is 95% confidence that there is no difference between the PSTL and STEP procedures in terms of their average CPU times. The requirement of the average CPU times for the PSTL and PREORDER procedures are quite different. This requirement depends upon the available number of different stock sheet sizes and the APSA ratio of the BOM. When there are only two different stock sheet sizes, the difference in the CPU times of the PSTL and PREORDER procedures is insignificant. However, as the number of different stock sheet sizes increases, the difference increases exponentially, especially for APSA ratios lower than 0.25. PREORDER demands, on average, much more CPU time than the PSTL procedure in all categories except those involving two different stock sheet sizes. This behavior is also confirmed by the results of testing the difference of the average CPU times presented in Figure 5.7.

5. Figure 5.6 shows that the PSTL procedure produces the lowest standard deviation for the average CPU times. The difference between the PSTL and STEP procedures is insignificant. However, the difference between the PSTL and PREORDER procedures is quite significant, especially in low (less than 0.25) APSA ratio categories. Furthermore, PREORDER is the most volatile procedure in terms of the average CPU times.

5.4 Comparison with Branch and Bound (B-B) Procedure

As mentioned in Chapter 3, the B-B procedure requires tremendous computer memory. If the number of different stock sheet sizes is larger than three, the B-B procedure cannot be executed on the current computer system. Therefore, the comparison of the PSTL and the B-B procedures is limited to those categories that have two and three different stock sheet sizes. The results are compared in Tables 5.1 and 5.2 in terms of the ASSU and CPU times.

The performance of the PSTL procedure is seen to compare well, giving very comparable ASSU values especially at APSA ratios lower than 0.10. The PSTL procedure requires much less CPU time than the B-B procedure. Therefore, the major improvement of the PSTL procedure, over the B-B procedure, is that it greatly improves computational efficiency with, at most, a slight 3% or so decrease in ASSU for the categories tested.

Table 5.1. Comparison of the ASSU with the B-B Procedure.

Procedure	Category									
	1	2	6	7	11	12	16	17	21	22
PSTL	80.76	82.49	88.29	89.60	94.08	94.54	96.70	97.25	98.81	99.06
B-B	84.07	85.10	90.32	91.47	95.90	97.07	98.15	98.59	99.62	99.82

Table 5.2. Comparison of the CPU Times (minutes) with the B-B Procedure.

Procedure	Category									
	1	2	6	7	11	12	16	17	21	22
PSTL	0.11	0.15	0.36	0.53	0.61	0.82	0.79	1.10	0.98	1.40
B-B	0.58	0.87	1.21	3.78	1.62	5.96	2.04	5.67	2.66	6.00

5.5 Conclusions

Based on the comparison of the test results, the following conclusions can be drawn.

1. The computational efficiency of the PSTL procedure is much higher than that of the PREORDER procedure, especially at APSA ratios lower than 0.25 and more than three different stock sheet sizes. It is also comparable to the STEP procedure in all categories.

2. The ASSU of the PSTL procedure is comparable with that of the PREORDER procedure and better than the STEP procedure.

Chapter 6. Conclusions and Recommendations

A Pre-Specified Trim Loss (PSTL) heuristic procedure was developed for solving the cutting of two-dimensional, multiple sized, stock sheets. The PSTL procedure was aimed at finding a "good" stock sheet cutting sequence that led to a low trim loss with high computational efficiency. The major effort was given to the design of heuristic polices to select different stock sheet sizes for a specific BOM. Observations from the tests of the different heuristic polices led to the conclusion that the effect of the heuristic polices on ASSU diminishes as the APSA ratio and the number of different stock sheet sizes decreases below about 0.10 and three, respectively. However, the corresponding effect on the CPU time increases as the APSA ratio decreases below about 0.50 and the number of different stock sheet sizes increases beyond three. This was the main consideration in the design of the PSTL procedure. Two Prolog routines, namely the BSS selection and PSTL calculation, were developed to implement the heuristic polices.

Another innovative feature of the PSTL procedure is the use of a random problem generator for generating test problems. It allowed the BOMs and different stock sheet data sets to be created that have a wide range and mix of piece sizes and quantities for the fairer evaluation of the procedures.

The PSTL procedure was compared with three previously published procedures of Qu and Sanders [34]. It provided a comparable ASSU and much higher computational efficiency compared with the B-B and PREORDER procedures. It also achieved a better ASSU and a comparable CPU time compared with the STEP procedure.

6.1 Conclusions

The following conclusions can be drawn from this research.

1. The APSA ratio and the number of stock sheet sizes are the major factors that should be considered in implementing heuristics for cutting of two-dimensional, multiple sized stock sheets.
2. A pre-examination of different stock sheet sizes, relative to the BOM, is essential in processing problems involving many different stock sheet sizes. The greater is the number of different stock sheet sizes, the greater is the advantage the BSS selection procedure.
3. The determination of the PSTL is critical. It affects not only the quality of the solution but the computational efficiency.
4. Comprehensive tests with randomly generated problems provide a more detailed picture of the performance of the PSTL procedure.

6.2 Recommendations

Recommendations for future study are provided below.

1. Upgrade the Prolog routines of the PSTL procedure by using more advanced Prolog programming tools to improve the capacity of the internal database. Then it will be possible to solve large size problems.
2. Further develop the basic stock sheet size selection procedure to enable it to determine the BSS when different stock sheet sizes are not specified beforehand.
3. Truly integrate the PSTL procedure with the cutting path optimization routine developed by Berscheid [38]. This system should be able to automatically determine the sequence of the different stock sheets for a specific BOM.
4. Extend the PSTL procedure to three-dimensional, multiple sized, container loading problems.

REFERENCES

1. Carnieri, C., Mendoza, G.A. and Gavinho, L.G., "Solution Procedures for Cutting Lumber into Furniture Parts", *European Journal of Operational Research*, no. 73, pp. 495-501,1994.

2. Gilmore, P. C. and Gomory, R. E., "Multistage Cutting Stock Problems of Two and More Dimensions", *Operations Research*, vol. 13, pp. 94-120, 1965.

3. Gilmore, P. C. and Gomory, R. E. , "A Linear Programming Approach to the Cutting Stock Problem", *Operations Research*, vol. 9, pp. 849-859, 1961.

4. Gilmore, P. C. and Gomory, R. E. , " A Linear Programming Approach to the Cutting Stock Problem - Part II ", *Operations Research*, vol. 11, pp. 863-888, 1963.

5. Gilmore, P. C. and Gomory, R. E., " Theory and Computation of Knapsack Functions", *Operations Research*, vol. 14, pp. 1045-1074, 1966.

6. Herz, J.L., " Recursive Computational Procedure for Two-dimensional Stock-cutting", *IBM Journal of Research and Development*, vol. 16, pp. 462-469, 1972.

7. Beasley, J.E., "Algorithm for Unconstrained Two-dimensional Guillotine Cutting", *Journal of Operational Research Society*, vol. 36, pp. 297-306, 1985.

8. Christofides, N. and Whitlock, C., "An Algorithm for Two-Dimensional Cutting Problems", *Operations Research*, vol. 25, pp. 31-44, 1977.

9. Christofides, N. and Hadjiconstantinou, E., "An Exact Algorithm for Orthogonal 2-D Cutting Problems Using Guillotine Cuts", *European Journal of Operational Research*, no. 83, pp. 21-38, 1995.

10. Beasley, J.E., "An Exact Two-Dimensional Non-Guillotine Cutting Tree Search Procedure", *Operations Research*, vol. 33, no. 1, pp. 49-64, 1985.

11. Adamowicz, M. Albano, A., "A Solution of the Rectangular Cutting-Stock Problem", *IEEE Trans. on System, Man, and Cybern*, vol. 6, pp. 302-310, 1976.

12. Albano, A., Orsini, R., "A Heuristic Solution of the Rectangular Cutting Stock Problems", *The Computer Journal*, vol. 33, pp. 338-343, 1980.

13. Wang. P. Y., "Two Algorithms for Constrained Two-Dimensional Cutting stock Problems", *Operations Research*, vol. 31, no. 3, pp. 573-586, 1983.

14. Oliveira, J.F. and Ferreira, J.S., "An Improved Version of Wang,s Algorithm for Two-Dimensional Cutting Problems", *European Journal of Operational Research*, no. 44, pp. 256-266, 1990.

15. Dietrich, R.D. and Yakowitz, S.J., "A Rule-based Approach to the Trim-loss Problem", *International Journal of Production Research*, vol. 29, no. 2, pp. 401-415, 1991.

16. Israni, S. and Sanders, J., "Two-dimensional Cutting Stock Problem Research: A Review and a New Rectangular Layout Algorithm", *Journal of Manufacturing Systems*, vol. 1, no. 1, pp. 169-182, 1982.

17. El-Bouri, A., Popplewell, N., Balakrishnan, S., and Alfa, A., "A Search-based Heuristic for the Two-dimensional Bin-packing Problem", *INFOR*, vol. 32, no. 4, pp. 265-274, Nov. 1994.

18. Bengtsson, B.E.., "Packing Rectangular Pieces - A Heuristic Approach", *The Computer Journal*, no. 25, pp. 353-357, 1982.

19. MacLeod, B., Moll, R., Girkar, M., Hanifi, N., "An Algorithm for the 2D Guillotine Cutting Stock Problem", *European Journal of Operational Research*, no. 68, pp. 400-412, 1993.

20. Wolfson, M.L., "Selecting the Best Length to Stock", *Operations Research*, vol. 13, pp. 570-585, 1965.

21. Page, E., "A Note On a Two-dimensional Dynamic Programming Problem", *Operational Research Quarterly*, vol. 26, pp. 321-324, 1975.

22. Chambers, M.L. and Dyson, R.G., "The Cutting Stock Problem in the Flat Glass Industry - Selection of Stock Size", *Operational Research Quarterly*, vol. 27, pp. 949-957, 1976.

23. Beasley, J.E., "An Algorithm for the Two-dimensional Assortment Problem", *European Journal of Operational Research*, vol. 19, no. 2, pp. 253-267, 1985.

24. Diegel, A. and Bocker, H.J., "Optimal Dimensions of Virgin Stock in Cutting Glass to Order", *Decision Sciences*, vol. 15, pp. 260-274, 1984.

25. Pentico, D.W., "The Discrete Two-dimensional Assortment Problem", *Operations Research*, vol. 36. no. 2, pp. 324-332, 1988.

26. Elenkotter, D., "A Dual-Based Procedure for Uncapacitated Facility Location", *Operations Research*, no. 26, pp. 992-1009, 1978.

27. Gochet, W. and Vandebroek, M., "A Dynamic Programming Based Heuristic for Industrial Buying of Cardboard", *European Journal of Operational Research*, no. 38, pp. 104-112, 1989.

28. Gemmill D.D. and Sanders, J.L., "A Comparison of Solution Methods for the Assortment Problem", *International Journal of Production Research*, vol. 29, no. 12, pp. 2521-2527, 1991.

29. Agrawal, P.K., "Determining Stock-sheet-sizes to Minimize Trim Loss", *European Journal of Operational Research*, no. 64, pp. 423-431, 1993.

30. Vasko, F. J. and Wolf F. E. "A Practical Approach for Determining Rectangular Stock Sizes", *Journal of Operational Research Society*, vol. 45, no. 3, pp. 281-286, 1994.

31. Gemmil, D.D. and Sanders, J.L., "Approximate Solutions for the Cutting Stock 'Portfolio' Problem", *European Journal of Operational Research*, no. 44, pp. 167-174, 1990.

32. Yanasse, H.H., "A Search Strategy for the One-size Assortment Problem", *European Journal of Operational Research*, no. 74, pp. 135-142, 1994.

33. Yanasse, H.H., Zinober, A.S.I. and Harris, R.G., "Two-dimensional Cutting Stock with Multiple Stock Sizes", *Journal of the Operational Research Society*, vol. 42, no. 8, pp. 673-683, 1991.

34. Qu, W. and Sanders, J.L., "Sequence Selection of Stock Sheets in Two-dimensional Layout Problems", *International Journal of Production Research*, vol. 27 no. 9, pp. 1553-1571, 1989.

35. Madsen,O.B.G., "An Application of Traveling-salesman Routines to Solve Pattern-allocation Problem", *Journal of the Operational Research Society*, vol. 39, pp. 249-256, 1988.

36. Yuen, B.J., "Heuristics for Sequencing Cutting Patterns", *European Journal of Operational Research*, no. 55, pp. 183-190, 1991.

37. Borland International, Turbo Prolog: Version 2.0 User's Guid. Scotts Valley, Calif. 1988 .

38. Berscheid, R, "Cutting Path Strategies for Two-dimensional Straight Line Cut Configurations", Master of Engineering Project Report, Department of Mechanical and Industrial Engineering, University of Manitoba, 1996.

39. Bischoff, E. and Ratclife, M., "Issures in the Development of Approaches to Container Loading", *International Journal of Management Science*, vol. 23, no. 4, pp. 377-390, 1995.

40. Mendenhall, W., and Sincich, T., Statistics for Engineering and the Science. Prentice Hall. Englenwood cliffs, New Jersey, 1995.

41. Khazanie, R., Elementary Statistics in a World of Applications. Scott, Foresman and Company, Glenview, Illinois, 1990.

APPENDIX A. Calculation of the Test Statistics

A.1 Test Statistic for Wilcoxon Rank Sum Test (T)

The test statistic of the Wilcoxon rank sum test is calculated according to the following steps [40].

1. Rank the n_1+n_2 observations in two samples from the smallest (rank 1) to the largest (rank n_1+n_2). The n_1 and n_2 are the sample size of sample 1 and sample 2, respectively.
2. Calculate T_1 and T_2, the rank sums associated with sample 1 and sample 2, respectively.
3. If $n_1 < n_2$, select T_1 as the test statistic T. If $n_1 > n_2$, select T_2 as the test statistic T. If $n_1 = n_2$, select either T_1 or T_2 as the test statistic T.

The following table shows an example of the calculation of the test statistic, T, for category # 1 by using the Wilcoxon rank sum test.

Table A.1. Example of the Wilcoxon Rank Sum Test.

LTM-MEAN		LAM-MEDIAN	
ASSU	Rank	ASSU	Rank
83.20	8	80.64	4
82.31	7	80.15	2
80.51	3	82.14	6
81.00	5	80.00	1
83.83	9	86.32	10
$n_1=5$	$T_1=32$	$n_2=5$	$T_2=23$

A.2 Statistic for Testing the Difference of the Means of Two Independent Samples (z)

The test statistic (z) for testing the difference of the means of two independent samples is calculated by using the following equation [40]:

$$\text{Test statistic, } z = \frac{X_1 - X_2}{\sqrt{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)}} \tag{A.1}$$

where

X_1	=	mean of sample 1
X_2	=	mean of sample 2
s_1	=	standard deviation of sample 1
s_2	=	standard deviation of sample 2
n_1	=	size of sample 1
n_2	=	size of sample 2.

APPENDIX B. Application of the Software

This appendix presents a simple example of the application of the software to sequence the orders of multiple sized stock sheets.

B.1. Bill of Material and Stock Sheets

The bill of material and the stock sheets used in the example are listed in the following two tables.

Table B.1. List of Bill of Material.

Piece #	Length*	Width*	Quantity Required
1	95	15	12
2	82	50	9
3	77	56	7
4	76	48	1
5	50	46	11

Table B.2. List of Stock Sheets.

Stock Sheet Size #	Length	Width	Quantity Available
1	179	100	unlimited
2	163	123	unlimited
3	160	113	unlimited

* Arbitrary but consistent units are needed.

B.2. Sequencing Orders by Using the Least Total Trim-loss Method (LTM)

1. Determining the basic stock sheet size (BSS) and the pre-specified trim-loss (PSTL).

The same BOM is laid on single sized sheets by using each of the three stock sheet sizes. The trim-loss for each sheet and the number of sheets of each size that are used are shown in Table C.3.

Table B.3. List of Trim-losses for Each Size of Sheet.

	Size 1: 179x100	Size 2: 163x123	Size 3: 160x113
Sheet #	Trim-loss	Trim-loss	Trim-loss
1	0.42	10.16	16.36
2	0.42	13.59	4.61
3	6.98	13.59	16.21
4	8.59	13.59	16.21
5	19.98	24.44	16.21
6	27.94	24.44	15.79
7	27.94	35.92 (last)	25.27
8	75.91 (last)		63.63 (last)
Total	92.27	99.81	110.66
Mean	13.18	16.64	15.81

The total trim-loss and the mean of the trim-loss for each size are also presented in Table C.3. Note that the last sheet is not counted. Then the stock sheet size that has the least total trim-loss (Size 1) is selected as the BSS, and the mean value of the BSS sheet (13.18) is specified as the PSTL.

2. Sequencing orders of the stock sheets.

After the BSS and the PSTL have been determined, multiple sized stock sheets are sequenced by the software in the manner shown in Table C.4.

Table B.4. Sequencing Orders of Multiple Sized Stock Sheets by LTM.

Series #	Size of the Shee	Trim-loss	Trim-loss <= PSTL ?	Sequence
1	1	0.42	Yes (Size 1 is accepted)	1
2	1	0.42	Yes (Size 1 is accepted)	11
3	1	6.98	Yes (Size 1 is accepted)	111
4	1	8.59	Yes (Size 1 is accepted)	1111
5	1	19.98	No (Size 1 is rejected)	11111
6	3	16.36	No (Size 3 is rejected)	11113
7	2	9.79	Yes (Size 2 is accepted)	11112
8	1	19.98	No (Size 1 is rejected)	111121
9	3	16.36	No (but Size 3 is accepted, because it has the least trim-	111123
10	2	28.56	No (Size 2 is rejected)	111122
Final sequence				111123

B.3. Sequencing Orders by Using Largest Area Method (LAM)

1. Determining the basic stock sheet size (BSS) and the pre-specified trim-loss (PSTL).

The area of the sheet for each size is calculated. The one that has the largest area is chosen as the BSS. In this example, the BSS is size 2: 163x123=20049. Then the BOM is laid on the size 2 sheets. As shown in Table C.3, the mean trim-loss of the size 2 sheets is 16.64. This value is specified as the PSTL.

2. Sequencing orders of the stock sheets.

After the BSS and the PSTL have been determined, multiple sized stock sheets are sequenced by the software in the manner shown in Table C.5.

Table B.5. Sequencing Orders of Multiple Sized Stock Sheets by LAM.

Series #	Size of the Sheet	Trim-loss	Trim-loss <= PSTL ?	Sequence
1	2	10.16	Yes (Size 2 is accepted)	2
2	2	13.59	Yes (Size 2 is accepted)	22
3	2	13.59	Yes (Size 2 is accepted)	222
4	2	13.59	Yes (Size 2 is accepted)	2222
5	2	24.44	No (Size 2 is rejected)	22222
6	3	16.21	Yes (Size 3 is accepted)	22223
7	2	24.44	No (Size 2 is rejected)	222232
8	3	16.21	Yes (Size 3 is accepted)	222233
Final sequence				222233

APPENDIX C. Test Results

Table C.1. Test Results for Category # 1 (APSA Ratio: 1.00, Number of Stock Sheet Sizes: 2).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	10	10	83.20	0.08	85.89	0.08	80.64	0.06	80.64	0.07	83.20	0.08	86.95	0.13	69.51	0.06
2	8	12	82.31	0.06	71.31	0.08	82.15	0.04	80.15	0.06	82.31	0.06	74.41	0.07	68.12	0.14
3	12	13	80.51	0.15	80.51	0.17	82.14	0.13	82.14	0.14	80.51	0.15	88.25	0.07	82.80	0.07
4	9	16	81.00	0.11	90.00	0.15	81.00	0.08	80.00	0.10	81.00	0.11	90.71	0.09	86.98	0.12
5	12	14	83.83	0.15	85.83	0.17	83.32	0.11	86.32	0.12	83.83	0.15	91.04	0.12	73.22	0.08
6	10	11									83.57	0.09	87.90	0.10	69.53	0.04
7	9	11									71.09	0.05	70.00	0.06	67.91	0.14
8	13	12									79.11	0.15	78.45	0.06	70.63	0.07
9	10	15									79.47	0.13	83.32	0.09	71.36	0.08
10	11	15									84.91	0.12	90.85	0.06	70.08	0.07
11	9	10									79.39	0.08	85.96	0.08	70.45	0.05
12	9	13									72.59	0.05	71.48	0.06	69.34	0.15
13	11	14									79.05	0.12	83.35	0.06	73.08	0.05
14	10	17									81.49	0.13	88.90	0.11	71.62	0.10
15	10	12									81.63	0.12	76.92	0.10	73.02	0.06
16	10	10									85.76	0.08	88.89	0.07	77.91	0.04
17	8	12									70.69	0.06	69.60	0.07	67.52	0.03
18	12	13									79.93	0.12	84.06	0.09	73.39	0.05
19	11	14									80.56	0.11	84.59	0.07	66.77	0.06
20	10	15									87.88	0.14	89.75	0.05	83.46	0.06
21	8	9									77.93	0.09	83.05	0.11	72.11	0.04
22	10	12									71.09	0.05	70.00	0.06	68.93	0.04
23	12	13									83.31	0.10	83.31	0.20	72.15	0.07
24	11	10									76.13	0.13	79.98	0.12	71.46	0.07
25	9	13									82.99	0.16	84.16	0.14	80.77	0.25
26	9	9									82.74	0.09	85.92	0.15	74.15	0.04
27	9	13									70.38	0.05	69.30	0.08	68.25	0.05
28	11	14									85.78	0.14	85.70	0.15	72.65	0.10
29	12	15									83.13	0.16	88.37	0.10	80.75	0.35
30	11	16									86.38	0.15	88.09	0.08	84.54	0.10

Table C.2. Test Results for Category # 2 (APSA Ratio: 1.00, Number of Stock Sheet Sizes: 3).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	10	10	83.20	0.12	85.89	0.12	84.31	0.06	84.31	0.06	83.20	0.12	89.31	0.19	73.51	0.06
2	8	12	82.31	0.07	71.31	0.10	80.21	0.04	80.15	0.07	82.31	0.07	70.21	0.07	69.66	0.24
3	12	13	82.29	0.09	87.43	0.29	82.78	0.12	82.78	0.14	82.29	0.09	85.08	0.08	86.24	0.12
4	9	16	87.77	0.14	89.37	0.21	82.77	0.07	83.37	0.13	87.77	0.14	89.37	0.13	86.98	0.18
5	12	14	86.84	0.29	86.84	0.24	79.42	0.10	78.42	0.11	86.84	0.29	86.84	0.42	80.11	0.12
6	10	11									84.35	0.12	89.75	0.18	75.64	0.06
7	9	11									71.09	0.07	70.26	0.14	69.44	0.20
8	13	12									81.17	0.21	85.17	0.11	79.26	0.12
9	10	15									79.25	0.16	87.23	0.14	72.96	0.09
10	11	15									85.37	0.16	88.92	0.43	82.68	0.11
11	9	10									84.81	0.12	90.81	0.05	72.53	0.07
12	9	13									72.59	0.07	76.04	0.13	70.65	0.18
13	11	14									87.64	0.15	86.86	0.11	79.23	0.80
14	10	17									83.24	0.17	88.65	0.23	76.22	0.12
15	10	12									82.15	0.17	87.48	0.10	72.82	0.10
16	10	10									85.76	0.12	90.51	0.19	84.25	0.07
17	8	12									73.78	0.09	73.78	0.21	68.01	0.05
18	12	13									87.64	0.17	84.01	0.09	81.00	0.08
19	11	14									76.90	0.15	80.56	0.18	72.56	0.10
20	10	15									83.63	0.21	89.75	0.10	79.44	0.11
21	8	9									86.93	0.12	85.83	0.30	75.53	0.07
22	10	12									74.97	0.08	75.85	0.21	68.40	0.05
23	12	13									84.94	0.15	91.16	0.06	80.88	0.08
24	11	10									80.17	0.15	89.49	0.15	73.39	0.11
25	9	13									86.71	0.23	86.30	0.98	80.77	0.30
26	9	9									82.74	0.13	92.20	0.08	75.66	0.06
27	9	13									70.38	0.08	69.30	0.19	67.72	0.06
28	11	14									92.59	0.18	90.00	0.12	85.89	0.11
29	12	15									86.99	0.22	92.60	0.21	84.37	0.40
30	11	16									86.38	0.22	86.38	1.93	83.67	0.10

Table C.3. Test Results for Category # 3 (APSA Ratio: 1.00, Number of Stock Sheet Sizes: 4).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	10	10	83.20	0.05	85.89	0.16	84.92	0.50	85.92	0.07	83.20	0.05	89.31	0.23	85.82	0.07
2	8	12	80.80	0.10	74.08	0.12	79.75	0.04	80.75	0.07	80.08	0.10	77.08	0.16	68.35	0.30
3	12	13	86.29	0.31	87.43	0.38	83.11	0.18	81.11	0.14	86.29	0.31	89.47	0.10	86.24	0.15
4	9	16	88.77	0.14	89.92	0.30	83.77	0.04	80.92	0.16	88.77	0.14	89.92	0.45	85.28	0.20
5	12	14	90.42	0.33	90.42	0.28	78.42	0.12	82.42	0.12	90.42	0.33	88.48	0.42	80.72	0.15
6	10	11									83.99	0.16	86.38	0.27	84.33	0.07
7	9	11									73.85	0.10	78.14	0.25	70.26	0.25
8	13	12									85.98	0.25	87.67	0.14	86.20	0.14
9	10	15									79.25	0.21	85.09	0.63	70.62	0.13
10	11	15									89.26	0.25	93.39	0.14	87.57	0.17
11	9	10									86.95	0.13	87.75	0.23	81.61	0.09
12	9	13									75.42	0.10	77.48	0.19	71.55	0.22
13	11	14									80.43	0.23	87.42	0.58	82.61	0.11
14	10	17									82.94	0.25	88.90	0.36	76.57	0.16
15	10	12									92.04	0.26	87.48	0.21	84.49	0.16
16	10	10									85.76	0.16	90.51	0.39	75.99	0.09
17	8	12									76.58	0.11	79.37	0.35	70.69	0.07
18	12	13									83.22	0.20	90.18	0.18	81.33	0.11
19	11	14									82.09	0.20	90.12	0.40	74.24	0.13
20	10	15									88.86	0.26	91.25	0.16	87.63	0.16
21	8	9									86.93	0.16	85.83	0.51	77.88	0.15
22	10	12									77.82	0.11	79.84	0.62	76.37	0.12
23	12	13									85.16	0.19	88.20	0.25	84.14	0.12
24	11	10									84.02	0.24	85.07	0.31	73.60	0.16
25	9	13									86.71	0.30	88.26	6.27	86.78	0.32
26	9	9									82.74	0.18	85.26	0.67	82.95	0.10
27	9	13									73.12	0.10	80.55	0.54	64.75	0.10
28	11	14									92.99	0.21	90.00	0.08	84.57	0.14
29	12	15									89.14	0.22	90.13	1.29	88.77	0.42
30	11	16									86.19	0.28	91.19	6.36	87.74	0.20

Table C.4. Test Results for Category # 4 (APSA Ratio: 1.00, Number of Stock Sheet Sizes: 5).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	10	10	88.43	0.04	85.89	0.20	84.92	0.04	82.92	0.07	88.43	0.04	88.44	0.63	85.82	0.10
2	8	12	78.46	0.05	74.46	0.15	80.10	0.04	78.10	0.07	78.46	0.05	77.90	0.30	71.49	0.12
3	12	13	87.15	0.35	87.43	0.45	82.38	0.22	82.38	0.14	87.15	0.35	90.75	0.13	82.29	0.22
4	9	16	86.96	0.25	91.06	0.33	84.77	0.04	84.92	0.19	86.96	0.25	91.11	2.04	88.04	0.23
5	12	14	91.63	0.40	91.63	0.34	75.42	0.08	80.42	0.13	91.63	0.40	90.63	0.13	87.91	0.18
6	10	11									87.10	0.20	89.96	0.61	84.33	0.11
7	9	11									74.23	0.12	81.02	0.28	76.26	0.28
8	13	12									91.94	0.25	87.67	0.16	83.77	0.19
9	10	15									86.85	0.24	95.55	0.27	76.64	0.16
10	11	15									88.22	0.28	92.26	0.10	86.44	0.21
11	9	10									86.95	0.19	88.71	0.45	83.80	0.10
12	9	13									75.80	0.12	78.29	0.43	77.17	0.28
13	11	14									85.80	0.22	87.74	0.17	82.28	0.15
14	10	17									84.84	0.29	91.14	0.61	80.49	0.17
15	10	12									92.04	0.31	92.04	2.08	85.76	0.18
16	10	10									85.76	0.19	90.69	0.93	86.75	0.11
17	8	12									76.58	0.14	82.07	0.55	76.81	0.08
18	12	13									92.22	0.21	87.74	0.15	88.59	0.13
19	11	14									90.82	0.26	90.13	0.16	87.35	0.15
20	10	15									88.02	0.30	88.34	0.16	77.61	0.18
21	8	9									84.43	0.20	90.89	0.49	87.44	0.20
22	10	12									78.20	0.13	79.51	0.46	76.37	0.18
23	12	13									92.14	0.23	92.14	0.42	85.53	0.15
24	11	10									90.26	0.28	91.48	0.40	86.50	0.17
25	9	13									84.64	0.33	88.75	0.27	79.59	0.38
26	9	9									83.17	0.19	88.17	0.60	86.14	0.13
27	9	13									73.49	0.13	80.96	0.48	67.58	0.18
28	11	14									92.99	0.24	90.50	0.63	90.40	0.20
29	12	15									88.40	0.28	9.52	0.48	88.77	0.50
30	11	16									86.19	0.34	90.59	7.56	86.58	0.20

Table C.5. Test Results for Category # 5 (APSA Ratio: 1.00, Number of Stock Sheet Sizes: 6).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	10	10	88.43	0.06	85.89	0.24	83.92	0.04	83.92	0.08	88.43	0.06	89.62	1.09	86.85	0.11
2	8	12	80.46	0.04	74.46	0.17	78.88	0.03	80.88	0.05	80.46	0.04	77.90	0.34	74.45	0.20
3	12	13	87.15	0.44	88.81	0.52	81.70	0.30	81.70	0.15	87.15	0.44	91.64	0.16	82.98	0.23
4	9	16	87.36	0.46	91.42	0.39	85.77	0.04	85.27	0.17	87.36	0.46	91.42	0.38	88.04	0.31
5	12	14	85.63	0.30	91.63	0.40	78.94	0.05	80.42	0.14	85.63	0.30	91.63	0.10	88.05	0.17
6	10	11									87.10	0.24	90.39	0.11	83.26	0.11
7	9	11									80.23	0.14	81.02	0.38	83.26	0.32
8	13	12									85.94	0.30	89.03	0.24	83.16	0.23
9	10	15									85.11	0.27	93.87	0.57	83.10	0.18
10	11	15									87.60	0.32	92.26	1.68	83.28	0.24
11	9	10									89.08	0.20	90.87	0.12	83.80	0.12
12	9	13									75.80	0.14	78.29	0.57	83.17	0.30
13	11	14									85.80	0.27	91.35	0.26	83.73	0.17
14	10	17									86.20	0.35	90.03	0.65	84.49	0.20
15	10	12									89.17	0.34	93.19	3.61	84.70	0.22
16	10	10									85.76	0.22	91.31	0.75	84.18	0.13
17	8	12									82.58	0.17	82.07	0.81	82.81	0.10
18	12	13									89.22	0.26	90.22	8.74	83.59	0.16
19	11	14									89.56	0.32	92.24	1.37	84.82	0.22
20	10	15									87.96	0.32	90.15	0.43	84.97	0.20
21	8	9									79.95	0.25	87.18	0.27	83.14	0.25
22	10	12									78.20	0.15	79.51	0.61	83.60	0.22
23	12	13									87.14	0.29	92.14	8.40	83.17	0.17
24	11	10									88.79	0.33	91.56	0.35	84.79	0.21
25	9	13									84.28	0.38	91.70	3.80	84.62	0.40
26	9	9									85.58	0.23	90.58	0.40	84.19	0.14
27	9	13									83.49	0.16	80.96	0.30	83.91	0.20
28	11	14									89.99	0.29	90.13	0.60	84.57	0.26
29	12	15									88.40	0.37	90.78	0.30	83.98	0.51
30	11	16									85.22	0.37	90.47	0.50	84.16	0.25

Table C.6. Test Results for Category # 6 (APSA Ratio: 0.50, Number of Stock Sheet Sizes: 2).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	18	24	89.53	0.15	83.20	0.48	89.53	0.17	89.53	0.43	89.53	0.15	92.18	0.17	81.75	0.20
2	25	25	90.94	0.30	95.94	0.40	90.94	0.19	88.94	0.30	90.94	0.30	91.07	0.54	73.86	0.30
3	26	29	88.92	0.43	90.08	0.43	87.92	0.15	89.92	0.27	88.92	0.43	88.32	0.16	86.28	0.31
4	21	21	86.53	0.13	86.53	0.34	88.96	0.17	87.96	0.29	86.53	0.13	86.62	1.08	83.32	0.30
5	21	26	89.52	0.42	87.42	0.50	87.42	0.25	88.42	0.36	89.52	0.42	84.99	0.33	77.03	0.20
6	19	23									87.31	0.32	94.68	0.15	75.97	0.17
7	24	26									88.58	0.43	87.93	0.10	74.79	0.33
8	25	30									84.76	0.50	90.27	0.24	83.13	0.30
9	20	22									83.91	0.27	86.49	0.11	83.29	0.30
10	22	25									90.22	0.37	91.46	0.17	84.58	0.28
11	17	25									88.83	0.31	90.84	0.38	80.61	0.20
12	23	26									91.75	0.46	91.75	0.70	84.20	0.36
13	27	30									90.76	0.58	86.64	0.20	84.42	0.25
14	20	24									87.66	0.30	86.55	0.12	66.47	0.30
15	22	23									84.27	0.29	86.98	0.18	84.41	0.33
16	20	25									86.76	0.34	89.85	0.18	81.49	0.40
17	25	28									91.75	0.41	87.65	0.18	79.22	0.33
18	26	28									92.14	0.46	86.44	0.17	91.95	0.40
19	18	24									88.26	0.31	88.26	0.43	73.31	0.30
20	22	28									88.81	0.35	81.15	0.13	85.59	0.32
21	19	24									87.98	0.33	91.47	1.09	86.07	0.32
22	24	27									86.44	0.38	95.96	0.48	80.92	0.30
23	26	31									88.37	0.56	96.21	0.27	88.37	0.20
24	18	22									90.46	0.29	90.46	0.46	81.60	0.30
25	23	27									87.92	0.31	94.37	0.15	78.95	0.50
26	19	27									91.85	0.28	95.64	0.17	83.60	0.30
27	26	30									87.81	0.35	91.95	0.92	82.92	0.25
28	25	29									87.90	0.46	87.45	0.24	87.54	0.30
29	19	28									84.69	0.39	85.47	0.32	80.82	0.65
30	21	26									84.05	0.39	91.66	0.32	76.54	0.40

Table C.7. Test Results for Category # 7 (APSA Ratio: 0.50, Number of Stock Sheet Sizes: 3).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN ASSU (%)	LTM-MEAN CPU Time (min.)	LTM-MEDIAN ASSU (%)	LTM-MEDIAN CPU Time (min.)	LAM-MEAN ASSU (%)	LAM-MEAN CPU Time (min.)	LAM-MEDIAN ASSU (%)	LAM-MEDIAN CPU Time (min.)	PSTL ASSU (%)	PSTL CPU Time (min.)	PREORDER ASSU (%)	PREORDER CPU Time (min.)	STEP ASSU (%)	STEP CPU Time (min.)
1	18	24	90.79	0.20	90.79	0.54	90.79	0.10	90.79	0.32	90.79	0.20	94.76	0.19	88.15	0.22
2	25	25	90.70	0.28	96.70	0.63	90.70	0.21	88.70	0.31	90.70	0.28	92.70	0.45	89.68	0.43
3	26	29	90.88	0.81	90.08	0.72	89.16	0.49	90.66	0.51	90.88	0.81	95.11	3.09	85.75	0.50
4	21	21	90.76	0.24	87.56	0.44	89.86	0.13	88.86	0.23	90.76	0.24	88.38	0.24	85.06	0.42
5	21	26	90.82	0.75	88.82	0.60	90.57	0.20	88.57	0.33	90.82	0.75	89.04	0.76	85.91	0.21
6	19	23									88.44	0.42	93.63	0.35	91.18	0.38
7	24	26									89.47	0.62	93.94	1.19	87.62	0.46
8	25	30									93.02	0.79	91.21	0.24	86.08	0.47
9	20	22									83.91	0.36	88.03	0.14	84.65	0.43
10	22	25									91.36	0.59	91.81	4.76	82.81	0.60
11	17	25									88.91	0.43	91.41	2.08	85.22	0.38
12	23	26									91.75	0.66	91.21	2.05	88.18	0.41
13	27	30									93.32	0.92	95.00	7.68	90.34	0.51
14	20	24									86.17	0.36	90.87	1.36	80.94	0.50
15	22	23									86.59	0.58	92.85	3.70	81.81	0.60
16	20	25									88.36	0.50	93.17	0.38	84.53	0.56
17	25	28									91.75	0.59	90.76	1.76	84.34	0.51
18	26	28									94.77	0.71	88.90	0.46	88.39	0.55
19	18	24									88.74	0.37	86.07	0.14	86.41	0.40
20	22	28									88.81	0.51	81.15	0.18	88.06	0.48
21	19	24									90.71	0.44	91.58	0.35	90.38	0.46
22	24	27									75.68	0.64	87.75	0.18	95.03	0.47
23	26	31									92.35	0.76	95.77	11.73	89.34	0.41
24	18	22									89.15	0.49	86.55	0.14	82.59	0.41
25	23	27									90.35	0.51	92.50	0.26	88.38	0.60
26	19	27									91.02	0.41	95.64	0.51	89.58	0.45
27	26	30									89.70	0.46	91.64	0.35	84.57	0.26
28	25	29									87.90	0.69	90.59	9.52	87.54	0.39
29	19	28									84.69	0.34	82.88	0.13	85.47	0.70
30	21	26									83.98	0.53	92.78	0.58	90.35	0.56

Table C.8. Test Results for Category # 8 (APSA Ratio: 0.50, Number of Stock Sheet Sizes: 4).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	18	24	90.79	0.20	91.63	0.77	91.80	0.13	90.80	0.27	90.79	0.20	94.06	3.86	90.23	0.23
2	25	25	92.70	0.21	96.70	0.77	91.70	0.14	92.70	0.34	92.70	0.21	96.70	0.90	85.44	0.60
3	26	29	92.12	1.19	92.72	0.78	91.72	0.57	92.72	0.44	92.12	1.19	95.49	5.01	83.83	0.68
4	21	21	91.56	0.30	87.56	0.78	91.67	0.15	89.15	0.32	91.56	0.30	89.15	4.81	88.25	0.55
5	21	26	92.13	1.18	89.24	0.79	91.55	0.16	90.44	0.41	92.13	1.18	90.54	0.34	87.55	0.25
6	19	23									93.19	0.54	93.72	2.37	89.23	0.46
7	24	26									89.86	0.71	95.10	2.70	85.84	0.62
8	25	30									93.57	0.90	92.85	0.36	93.80	0.63
9	20	22									88.31	0.47	89.56	2.63	83.30	0.57
10	22	25									91.36	0.70	94.07	0.12	87.63	0.67
11	17	25									88.44	0.52	93.44	2.68	89.31	0.46
12	23	26									91.75	0.83	89.75	0.17	86.79	0.58
13	27	30									93.32	1.18	95.36	12.95	86.63	0.64
14	20	24									86.17	0.52	90.87	2.84	84.03	0.69
15	22	23									88.89	0.59	82.00	0.17	88.79	0.78
16	20	25									90.18	0.56	93.56	1.31	88.83	0.70
17	25	28									87.70	0.83	95.76	3.10	87.39	0.65
18	26	28									94.31	1.14	90.57	0.76	92.55	0.74
19	18	24									88.74	0.55	89.32	6.84	85.02	0.59
20	22	28									88.81	0.66	88.61	0.16	86.77	0.60
21	19	24									91.88	0.55	91.83	0.87	92.26	0.56
22	24	27									75.68	0.88	90.97	5.04	95.03	0.59
23	26	31									92.35	1.02	96.93	21.47	91.26	0.55
24	18	22									89.15	0.60	90.35	0.86	82.98	0.57
25	23	27									92.53	0.53	92.57	0.08	93.27	0.72
26	19	27									91.65	0.51	94.32	3.10	87.12	0.55
27	26	30									94.95	0.48	91.86	2.35	83.76	0.30
28	25	29									92.95	0.89	90.38	13.12	85.46	0.41
29	19	28									93.90	0.49	90.01	2.10	91.63	0.70
30	21	26									83.92	0.65	91.39	0.15	91.44	0.60

Table C.9. Test Results for Category # 9 (APSA Ratio: 0.50, Number of Stock Sheet Sizes: 5).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	18	24	94.04	0.76	94.57	1.00	91.36	0.15	91.36	0.25	94.04	0.76	96.04	0.51	90.63	0.25
2	25	25	93.70	1.00	96.70	1.10	92.70	0.27	91.20	0.37	93.70	1.00	96.70	1.78	85.75	0.75
3	26	29	90.40	1.16	91.40	1.00	90.39	0.34	92.72	0.45	90.40	1.16	96.72	5.27	92.24	0.75
4	21	21	92.15	0.49	89.15	1.09	89.52	0.19	88.52	0.32	92.15	0.49	92.42	0.35	86.96	0.80
5	21	26	92.72	0.93	92.03	1.12	90.56	0.21	90.55	0.31	92.72	0.93	89.59	1.38	86.14	0.76
6	19	23									93.19	0.75	95.04	0.54	88.34	0.65
7	24	26									90.86	0.77	96.70	5.20	92.01	0.89
8	25	30									93.57	1.12	93.32	0.52	93.80	0.86
9	20	22									87.56	0.55	88.03	0.19	87.40	0.76
10	22	25									91.36	0.85	93.12	27.72	87.63	0.86
11	17	25									90.44	0.67	92.53	3.61	89.31	0.88
12	23	26									89.10	1.05	90.88	0.17	88.92	0.66
13	27	30									91.62	1.29	95.51	10.92	86.98	0.75
14	20	24									86.53	0.61	91.68	4.20	84.60	0.78
15	22	23									88.89	0.81	88.11	0.65	89.34	0.90
16	20	25									89.26	0.70	94.00	0.69	88.23	0.86
17	25	28									89.69	1.01	95.76	5.61	94.64	0.90
18	26	28									93.16	1.34	90.57	0.92	91.63	0.95
19	18	24									88.74	0.67	90.88	0.40	85.96	0.94
20	22	28									91.76	0.79	88.61	0.16	88.79	0.80
21	19	24									91.88	0.76	92.66	5.59	92.26	0.75
22	24	27									84.99	1.01	90.97	2.09	91.26	0.76
23	26	31									92.35	1.33	96.93	12.90	91.26	0.70
24	18	22									92.39	0.69	90.90	2.09	91.23	0.72
25	23	27									92.53	0.64	93.83	15.87	91.38	0.97
26	19	27									93.02	0.64	95.18	4.33	93.10	0.60
27	26	30									91.95	0.76	91.86	8.53	87.81	0.31
28	25	29									91.52	1.32	90.53	0.93	92.85	0.40
29	19	28									93.90	0.64	94.01	6.13	90.05	0.75
30	21	26									90.56	0.82	91.39	0.15	91.44	0.78

Table C.10. Test Results for Category # 10 (APSA Ratio: 0.50, Number of Stock Sheet Sizes: 6).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN ASSU (%)	LTM-MEAN CPU Time (min.)	LTM-MEDIAN ASSU (%)	LTM-MEDIAN CPU Time (min.)	LAM-MEAN ASSU (%)	LAM-MEAN CPU Time (min.)	LAM-MEDIAN ASSU (%)	LAM-MEDIAN CPU Time (min.)	PSTL ASSU (%)	PSTL CPU Time (min.)	PREORDER ASSU (%)	PREORDER CPU Time (min.)	STEP ASSU (%)	STEP CPU Time (min.)
1	18	24	93.04	0.86	94.57	1.29	90.60	0.12	92.36	0.27	93.04	0.86	94.62	8.30	91.73	0.40
2	25	25	96.70	1.01	96.70	1.23	91.70	0.29	92.30	0.40	96.70	1.01	96.70	3.58	93.54	0.81
3	26	29	94.79	1.39	94.05	1.20	93.73	0.20	92.23	0.42	94.79	1.39	95.70	1.15	93.13	0.82
4	21	21	91.30	0.68	91.30	1.29	85.18	0.26	92.22	0.40	91.30	0.68	93.25	13.26	86.79	0.95
5	21	26	92.72	1.06	92.32	1.31	86.87	0.27	92.31	0.40	92.72	1.06	91.94	2.34	86.21	0.88
6	19	23									93.19	0.88	94.64	4.02	90.86	0.84
7	24	26									96.22	0.99	96.70	8.32	95.00	0.96
8	25	30									93.57	1.40	96.97	33.65	92.80	0.96
9	20	22									90.93	0.69	88.03	0.19	90.62	0.87
10	22	25									91.36	1.03	93.86	22.99	90.70	0.99
11	17	25									90.87	0.80	94.29	5.52	89.57	0.96
12	23	26									90.73	1.16	95.34	7.84	92.10	0.82
13	27	30									94.64	1.53	94.80	0.70	91.36	0.88
14	20	24									86.53	0.74	91.67	10.17	84.60	0.80
15	22	23									91.36	1.04	84.52	0.47	92.54	1.05
16	20	25									91.32	0.87	96.08	2.97	88.99	0.90
17	25	28									92.74	1.08	95.75	12.42	92.59	0.97
18	26	28									94.61	1.55	96.57	0.92	91.18	1.06
19	18	24									88.74	0.78	92.30	5.14	90.11	1.01
20	22	28									91.76	0.89	94.69	0.46	89.47	1.08
21	19	24									91.88	0.89	96.65	11.60	92.49	0.99
22	24	27									89.89	0.95	96.58	3.81	91.55	1.03
23	26	31									91.52	1.39	96.93	32.05	91.55	0.98
24	18	22									92.39	0.84	90.33	0.14	90.94	0.96
25	23	27									92.39	0.89	93.83	23.35	91.83	0.98
26	19	27									93.02	0.75	94.74	11.22	90.14	0.90
27	26	30									94.95	0.97	95.86	7.44	90.55	0.97
28	25	29									94.88	1.58	94.53	0.93	91.78	0.50
29	19	28									94.06	0.75	86.51	0.11	90.05	0.99
30	21	26									83.84	0.97	92.78	0.48	91.44	0.95

Table C.11. Test Results for Category # 11 (APSA Ratio: 0.25, Number of Stock Sheet Sizes: 2).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	24	44	94.02	0.38	97.63	0.51	93.97	0.32	93.97	0.38	94.02	0.38	94.63	0.71	92.45	0.96
2	28	42	96.48	0.25	97.65	0.53	96.47	0.33	93.88	0.59	96.48	0.25	94.93	0.23	89.44	1.08
3	36	42	96.82	0.60	91.43	0.67	96.82	0.28	93.55	0.47	96.82	0.60	95.35	0.42	89.56	0.24
4	32	39	93.15	1.13	93.15	1.02	90.73	0.48	92.73	0.53	93.15	1.13	84.34	0.21	88.89	0.30
5	19	41	90.21	0.57	89.37	0.72	90.21	0.55	95.49	0.57	90.21	0.57	91.93	1.23	83.37	0.21
6	23	45									94.49	0.40	96.64	1.28	93.88	1.07
7	29	43									95.95	0.41	96.98	1.29	88.40	0.99
8	35	44									95.43	0.71	96.88	1.49	88.46	0.19
9	31	40									91.20	0.89	90.22	0.29	91.03	0.21
10	20	40									93.35	0.58	93.13	0.19	86.42	0.23
11	22	46									90.94	0.39	96.98	0.34	93.88	0.33
12	26	40									96.81	0.41	97.59	0.92	90.40	0.27
13	34	41									95.71	1.04	96.49	0.46	94.63	0.33
14	33	38									95.06	0.87	95.75	1.11	88.47	0.30
15	20	43									90.92	0.50	89.46	0.18	92.38	0.33
16	22	42									93.60	0.53	96.18	1.62	93.96	0.40
17	30	46									96.55	0.50	96.14	0.27	97.06	0.33
18	36	48									96.22	0.83	99.42	3.15	94.39	0.40
19	30	44									91.13	1.12	92.00	0.60	90.68	0.30
20	22	47									94.18	0.38	92.93	0.14	87.80	0.32
21	20	47									94.62	0.68	96.45	2.02	92.18	0.32
22	25	38									97.22	0.40	97.98	0.38	89.96	0.30
23	33	39									95.58	0.85	97.28	0.69	91.19	0.20
24	33	39									91.80	0.66	91.98	0.29	86.44	0.30
25	19	44									91.63	0.42	93.06	0.16	81.94	0.50
26	23	46									92.82	0.49	94.17	0.62	86.82	0.30
27	29	47									95.76	0.42	97.82	2.42	94.13	0.25
28	35	49									96.27	0.67	93.28	0.65	94.33	0.30
29	29	48									92.28	0.63	93.34	2.99	88.75	0.65
30	21	46									92.11	0.48	88.15	0.28	86.23	0.40

Table C.12. Test Results for Category # 12 (APSA Ratio: 0.25, Number of Stock Sheet Sizes: 3).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	24	44	95.98	0.51	95.98	0.61	95.98	0.28	95.98	0.38	95.98	0.51	97.63	0.56	94.64	0.33
2	28	42	95.63	0.44	98.63	0.54	96.47	0.37	93.82	0.77	95.63	0.44	95.35	2.49	96.80	0.36
3	36	42	96.80	0.79	96.80	0.85	92.36	0.48	96.24	0.75	96.80	0.79	96.80	0.62	94.13	0.46
4	32	39	95.85	1.47	94.58	1.44	93.57	0.74	94.58	0.87	95.85	1.47	90.22	5.12	90.67	1.74
5	19	41	96.00	0.62	93.83	0.89	93.83	0.64	93.83	0.78	96.00	0.62	92.25	2.95	89.68	0.39
6	23	45									94.49	0.61	98.16	0.35	93.46	0.34
7	29	43									98.53	0.49	98.58	0.41	93.81	0.23
8	35	44									95.29	1.08	96.90	5.99	94.57	0.54
9	31	40									87.88	1.49	95.05	2.73	93.72	0.82
10	20	40									93.03	0.69	95.94	1.79	92.15	0.34
11	22	46									93.45	0.56	97.60	0.51	94.42	0.45
12	26	40									93.04	0.88	97.83	2.84	93.42	0.27
13	34	41									95.28	1.24	97.57	7.06	93.00	1.54
14	33	38									95.06	0.25	95.35	5.21	92.35	0.69
15	20	43									95.59	0.57	90.11	2.59	92.88	0.38
16	22	42									93.60	0.69	97.81	0.49	94.14	0.40
17	30	46									92.97	0.73	98.54	4.19	93.47	0.22
18	36	48									96.22	1.34	97.87	0.85	93.24	0.61
19	30	44									91.13	1.44	93.00	13.69	93.59	1.75
20	22	47									92.73	0.50	97.75	1.79	93.84	0.29
21	20	47									97.79	0.88	98.77	0.96	94.40	0.61
22	25	38									97.28	0.60	99.24	2.71	94.77	0.23
23	33	39									92.98	1.26	97.08	11.35	93.10	1.96
24	33	39									91.06	1.24	94.82	17.29	93.36	1.10
25	19	44									91.63	0.56	96.67	3.13	94.99	0.29
26	23	46									96.75	0.66	97.68	1.38	93.87	0.24
27	29	47									95.76	0.47	98.78	1.32	92.52	0.25
28	35	49									96.27	1.05	94.57	10.91	92.15	0.70
29	29	48									94.83	0.87	95.83	9.64	93.71	0.37
30	21	46									93.16	0.60	90.27	3.72	92.84	0.39

Table C.13. Test Results for Category # 13 (APSA Ratio: 0.25, Number of Stock Sheet Sizes: 4).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	24	44	96.74	0.87	96.74	1.48	96.74	0.29	96.74	0.48	96.74	0.87	98.11	0.73	95.41	0.42
2	28	42	97.34	1.06	97.34	1.47	97.14	0.32	97.93	0.50	97.34	1.06	95.19	0.72	97.19	0.51
3	36	42	98.80	1.09	98.26	1.14	98.80	0.54	94.26	0.71	98.80	1.09	97.97	28.53	94.83	0.51
4	32	39	95.77	1.40	95.77	1.18	95.34	0.64	96.00	0.91	95.77	1.40	97.24	11.19	91.96	2.86
5	19	41	94.34	0.93	94.34	1.13	94.34	0.62	95.34	0.75	94.34	0.93	97.25	6.43	89.88	0.48
6	23	45									94.49	0.85	97.26	0.53	92.99	0.41
7	29	43									97.57	0.68	98.58	0.63	95.91	0.37
8	35	44									95.54	1.66	98.24	13.17	94.99	0.73
9	31	40									92.21	1.81	96.56	7.77	95.65	0.90
10	20	40									93.03	0.98	96.03	0.39	95.07	0.65
11	22	46									94.27	0.71	97.64	0.82	93.36	0.49
12	26	40									96.95	0.85	97.83	4.41	95.64	0.33
13	34	41									95.72	1.47	98.53	1.03	95.82	2.56
14	33	38									96.12	1.26	97.60	0.68	92.36	0.75
15	20	43									97.13	1.03	91.03	10.93	93.25	0.41
16	22	42									95.74	1.01	98.75	10.48	94.95	0.57
17	30	46									98.13	0.83	99.21	7.24	97.47	0.70
18	36	48									96.01	1.71	99.25	46.28	97.55	0.71
19	30	44									95.13	1.63	93.19	1.41	91.48	2.75
20	22	47									92.73	0.72	91.64	0.18	93.04	0.34
21	20	47									97.79	1.19	98.24	10.84	95.26	0.66
22	25	38									97.28	0.78	99.90	7.65	97.55	0.36
23	33	39									95.75	1.57	98.17	32.50	94.59	2.96
24	33	39									92.83	1.59	97.01	26.83	91.36	2.09
25	19	44									93.45	0.67	87.83	0.51	96.98	0.31
26	23	46									95.64	0.78	97.68	2.83	91.69	0.30
27	29	47									96.16	0.49	99.00	9.65	96.11	0.30
28	35	49									96.26	1.47	90.10	3.08	95.48	0.88
29	29	48									95.66	1.15	95.83	9.64	89.16	0.50
30	21	46									95.92	0.82	92.53	2.69	92.08	0.45

Table C.14. Test Results for Category # 14 (APSA Ratio: 0.25, Number of Stock Sheet Sizes: 5).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	24	44	97.29	1.04	97.29	1.56	96.74	0.33	96.74	0.52	97.29	1.04	97.63	11.48	96.43	0.71
2	28	42	98.18	0.78	98.18	1.56	94.29	0.43	96.17	0.54	98.18	0.78	98.57	0.91	98.01	0.61
3	36	42	97.29	1.14	98.26	1.57	97.29	0.37	95.26	0.84	97.29	1.14	98.24	17.49	95.31	1.02
4	32	39	97.03	1.96	96.00	1.48	95.34	0.70	95.39	1.14	97.03	1.96	97.24	17.39	96.87	3.20
5	19	41	96.04	1.09	96.04	1.47	96.04	0.68	96.04	0.85	96.04	1.09	97.25	4.58	89.88	0.58
6	23	45									94.78	1.08	97.37	0.63	93.88	0.81
7	29	43									99.07	0.67	99.21	4.33	97.30	0.36
8	35	44									98.92	1.64	97.34	0.86	96.14	1.15
9	31	40									91.04	2.64	94.22	0.57	93.61	1.61
10	20	40									93.03	1.20	96.56	9.71	91.57	0.89
11	22	46									94.27	0.95	98.13	0.54	98.51	0.66
12	26	40									98.11	0.97	98.61	6.25	97.22	0.64
13	34	41									96.61	1.83	98.14	19.65	94.96	2.10
14	33	38									97.49	2.03	97.42	11.29	95.06	0.90
15	20	43									97.13	1.31	97.03	12.65	93.25	0.53
16	22	42									95.74	1.33	98.75	10.55	94.95	0.72
17	30	46									98.46	1.09	96.14	1.00	98.03	0.54
18	36	48									96.01	2.03	99.25	16.32	96.12	1.00
19	30	44									95.13	2.09	96.19	1.07	93.15	3.10
20	22	47									92.73	0.94	97.64	4.32	93.04	0.55
21	20	47									97.06	1.22	98.77	1.65	92.55	0.56
22	25	38									97.60	0.97	99.90	11.37	98.02	0.50
23	33	39									95.75	1.94	97.28	1.70	95.18	3.10
24	33	39									95.31	1.51	97.84	0.82	92.43	3.09
25	19	44									93.45	0.83	88.50	12.41	94.79	0.61
26	23	46									95.87	0.99	98.51	21.64	92.20	0.42
27	29	47									98.69	0.79	98.78	2.03	98.61	0.48
28	35	49									96.26	2.03	95.10	30.00	92.71	0.84
29	29	48									96.64	1.76	97.78	50.00	94.27	0.95
30	21	46									93.92	1.06	90.06	60.00	93.36	0.52

Table C.15. Test Results for Category # 15 (APSA Ratio: 0.25, Number of Stock Sheet Sizes: 6).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	24	44	96.74	1.08	96.74	1.73	95.74	0.37	96.74	0.63	96.74	1.08	98.35	9.67	96.43	0.82
2	28	42	97.60	1.06	98.60	1.85	95.29	0.47	94.86	0.57	97.60	1.06	98.57	1.04	98.25	0.78
3	36	42	97.29	1.48	94.66	1.75	96.29	0.40	94.70	0.75	97.29	1.48	98.24	50.41	97.00	1.37
4	32	39	96.55	2.24	96.55	1.81	95.84	0.67	96.16	1.02	96.55	2.24	97.24	48.38	92.22	4.21
5	19	41	95.61	1.14	94.61	1.87	95.61	0.65	94.61	0.67	95.61	1.14	97.25	9.39	92.59	0.72
6	23	45									97.68	1.15	98.91	27.49	95.16	1.04
7	29	43									97.75	0.87	99.59	8.03	97.30	0.45
8	35	44									97.61	2.06	98.92	58.18	96.16	1.27
9	31	40									97.05	3.30	92.89	30.00	96.89	1.78
10	20	40									96.78	1.15	97.36	23.31	93.73	0.95
11	22	46									96.73	1.05	98.13	0.54	96.38	0.88
12	26	40									98.11	1.19	98.61	10.20	97.11	0.59
13	34	41									96.61	2.16	99.11	70.27	96.32	1.27
14	33	38									96.88	2.36	97.88	33.19	97.51	1.48
15	20	43									97.13	1.54	98.99	3.75	93.83	0.67
16	22	42									95.74	1.49	97.80	0.68	95.43	0.80
17	30	46									98.46	1.31	99.25	12.15	98.03	0.66
18	36	48									96.01	2.34	99.25	31.37	96.12	1.17
19	30	44									95.13	2.44	96.19	1.26	93.58	4.10
20	22	47									96.53	0.97	98.55	5.17	94.61	0.54
21	20	47									95.41	1.49	99.05	26.60	93.95	0.66
22	25	38									99.05	1.00	99.90	16.79	98.02	0.59
23	33	39									97.95	2.04	98.94	39.36	96.23	4.00
24	33	39									95.31	1.87	97.94	15.66	94.48	0.79
25	19	44									94.66	1.06	90.60	10.48	96.15	0.76
26	23	46									95.87	1.17	98.51	15.43	92.45	0.78
27	29	47									98.80	0.98	92.77	10.09	98.95	0.53
28	35	49									96.26	2.71	98.32	72.08	93.18	2.00
29	29	48									96.29	2.34	97.65	9.44	96.42	0.97
30	21	46									96.36	1.15	92.35	5.93	93.36	0.64

Table C.16. Test Results for Category # 16 (APSA Ratio: 0.10, Number of Stock Sheet Sizes: 2).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	29	90	97.87	0.73	97.17	0.99	97.87	0.55	97.19	1.26	97.87	0.73	98.77	0.43	95.77	0.40
2	41	91	99.51	0.38	99.42	0.89	99.51	0.35	99.42	0.56	99.51	0.38	99.64	0.97	96.74	0.35
3	19	62	98.30	0.68	97.00	0.83	98.30	0.52	97.00	0.56	98.30	0.68	98.13	0.30	96.04	0.36
4	30	63	92.21	1.09	92.21	0.96	92.21	0.78	92.21	0.96	92.21	1.09	95.26	2.34	91.14	0.89
5	36	76	98.46	0.76	98.46	1.02	96.46	0.52	96.46	1.02	98.46	0.76	97.24	0.30	93.34	0.45
6	30	91									96.91	0.93	98.12	3.50	97.86	0.41
7	42	92									98.36	0.69	99.85	1.80	98.40	0.30
8	20	63									97.96	0.45	98.91	0.18	95.35	0.31
9	31	64									93.30	0.91	94.35	0.32	91.82	0.49
10	37	77									95.69	1.03	95.52	0.31	94.09	0.50
11	28	89									98.75	1.32	99.36	0.75	95.89	0.43
12	40	90									99.09	0.52	98.84	0.32	97.48	0.37
13	18	61									94.03	0.76	96.49	0.32	90.58	0.37
14	29	62									93.32	0.84	96.67	2.89	94.78	0.90
15	35	75									94.10	1.13	94.27	0.39	95.85	0.50
16	32	94									99.33	0.68	95.92	0.41	97.17	1.01
17	44	94									96.09	0.72	97.96	0.70	99.09	0.41
18	22	65									96.30	0.46	97.47	1.13	96.22	0.32
19	33	66									97.05	0.84	97.90	2.14	96.26	0.46
20	39	79									93.46	1.06	93.97	0.44	95.52	0.51
21	27	88									98.62	0.62	98.85	0.37	95.56	0.34
22	38	90									99.17	0.50	99.95	1.51	98.28	1.85
23	20	62									96.29	0.64	96.51	0.31	92.76	0.26
24	31	63									94.97	1.12	98.36	0.71	95.15	1.70
25	33	76									95.91	1.02	97.91	1.75	85.26	0.48
26	34	95									99.15	0.41	98.93	1.32	96.04	0.52
27	43	93									95.54	0.52	97.57	1.34	97.63	0.35
28	21	66									95.46	0.45	97.57	1.37	93.48	0.26
29	32	67									98.52	0.96	93.64	0.31	94.34	0.54
30	38	89									97.20	1.48	98.64	3.28	90.66	1.85

Table C.17. Test Results for Category # 17 (APSA Ratio: 0.10, Number of Stock Sheet Sizes: 3).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	29	90	97.87	0.92	96.49	1.39	97.84	0.58	96.49	1.34	97.87	0.92	98.77	0.56	96.79	0.63
2	41	91	99.51	0.57	99.90	1.40	99.51	0.40	99.90	0.51	99.51	0.57	100	2.34	99.56	0.45
3	19	62	95.54	0.92	95.54	1.23	94.82	0.49	95.00	0.60	95.54	0.92	97.64	1.52	96.18	0.50
4	30	63	95.84	1.52	95.84	1.40	95.84	0.97	95.84	1.41	95.84	1.52	97.61	6.69	90.82	0.85
5	36	76	98.46	1.22	98.46	1.38	96.46	0.64	98.11	1.08	98.46	1.22	98.46	5.36	94.88	0.70
6	30	91									98.34	1.33	99.00	0.81	96.45	0.70
7	42	92									99.49	0.82	99.60	0.50	99.34	0.61
8	20	63									97.96	0.63	97.95	0.40	96.59	0.40
9	31	64									91.70	1.30	95.21	5.75	89.93	0.82
10	37	77									97.17	1.34	98.42	9.61	95.08	0.86
11	28	89									99.69	1.31	99.47	0.81	96.06	0.60
12	40	90									100	0.76	100	4.87	97.74	0.49
13	18	61									94.80	0.85	97.13	1.81	95.64	0.42
14	29	62									93.32	1.32	97.50	10.97	96.24	1.04
15	35	75									97.42	1.32	98.53	9.18	95.89	0.81
16	32	94									98.21	1.16	95.84	0.91	97.17	2.10
17	44	94									99.96	0.83	97.85	11.05	99.50	0.55
18	22	65									97.30	0.61	97.47	2.20	96.22	0.41
19	33	66									93.46	1.17	98.88	7.40	90.12	0.61
20	39	79									95.27	1.61	95.87	0.53	97.16	0.83
21	27	88									98.62	0.85	99.78	7.31	96.75	0.60
22	38	90									99.89	0.56	99.95	3.47	99.54	2.85
23	20	62									98.39	0.69	97.58	0.36	96.59	0.37
24	31	63									94.97	1.73	98.29	4.11	94.24	2.68
25	33	76									97.70	1.60	98.09	4.92	95.90	0.81
26	34	95									97.54	0.86	99.70	1.46	97.01	0.54
27	43	93									95.54	0.85	98.00	2.55	98.83	0.45
28	21	66									97.75	0.81	98.49	0.20	96.38	0.42
29	32	67									98.52	1.54	92.02	6.94	91.18	0.80
30	38	89									97.20	1.86	2.10	6.56	95.71	2.10

Table C.18. Test Results for Category # 18 (APSA Ratio: 0.10, Number of Stock Sheet Sizes: 4).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	29	90	98.05	1.24	96.49	1.85	95.77	0.55	95.96	1.11	98.05	1.24	99.59	14.13	98.55	0.86
2	41	91	99.71	0.82	99.44	1.84	99.77	0.33	99.92	0.59	99.71	0.82	99.90	0.69	99.64	0.56
3	19	62	96.47	1.06	96.45	1.78	96.59	0.58	96.59	0.64	96.47	1.06	98.26	5.96	96.03	0.68
4	30	63	96.30	2.05	96.30	1.66	96.30	1.14	96.30	1.66	96.30	2.05	97.87	35.36	91.49	0.94
5	36	76	98.46	1.73	98.46	1.82	95.62	0.84	97.45	1.71	98.46	1.73	98.75	35.75	96.85	0.87
6	30	91									98.34	1.63	99.53	17.58	97.49	0.74
7	42	92									96.85	1.09	99.62	0.79	96.74	0.72
8	20	63									97.96	0.88	97.95	0.59	96.53	0.55
9	31	64									96.71	2.09	98.21	12.31	96.43	1.08
10	37	77									97.46	1.60	99.03	14.06	96.54	0.95
11	28	89									98.11	2.09	99.69	6.96	96.65	1.88
12	40	90									97.00	0.98	100	0.67	97.75	0.57
13	18	61									96.80	1.06	96.79	0.46	97.21	0.52
14	29	62									96.32	1.75	97.50	16.30	96.24	2.30
15	35	75									96.25	1.87	94.98	0.62	97.35	1.16
16	32	94									98.76	1.55	96.98	1.08	97.47	0.67
17	44	94									96.96	1.13	100	4.62	97.55	0.64
18	22	65									97.85	0.83	99.02	2.07	97.76	0.43
19	33	66									96.46	1.68	98.88	9.91	97.64	1.00
20	39	79									96.27	2.04	94.49	24.49	96.41	0.87
21	27	88									98.72	1.08	98.95	1.20	97.58	0.82
22	38	90									99.77	0.89	99.57	0.57	96.90	3.60
23	20	62									98.39	0.89	98.74	12.41	97.16	0.52
24	31	63									96.97	2.21	98.78	8.44	96.08	3.80
25	33	76									97.74	1.65	98.60	28.72	96.40	0.84
26	34	95									96.28	1.08	99.70	2.93	97.18	0.64
27	43	93									97.72	0.89	100	0.75	97.83	0.62
28	21	66									97.75	1.17	98.49	3.16	96.96	0.50
29	32	67									98.52	2.06	92.40	4.81	97.24	1.23
30	38	89									97.20	2.35	98.64	12.00	96.19	2.50

Table C.19. Test Results for Category # 19 (APSA Ratio: 0.10, Number of Stock Sheet Sizes: 5).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN ASSU (%)	LTM-MEAN CPU Time (min.)	LTM-MEDIAN ASSU (%)	LTM-MEDIAN CPU Time (min.)	LAM-MEAN ASSU (%)	LAM-MEAN CPU Time (min.)	LAM-MEDIAN ASSU (%)	LAM-MEDIAN CPU Time (min.)	PSTL ASSU (%)	PSTL CPU Time (min.)	PREORDER ASSU (%)	PREORDER CPU Time (min.)	STEP ASSU (%)	STEP CPU Time (min.)
1	29	90	98.05	1.36	98.50	2.32	97.84	0.58	97.00	0.72	98.05	1.36	98.77	0.82	95.59	1.14
2	41	91	99.78	1.04	99.44	2.56	99.77	0.36	99.92	0.78	99.78	1.04	100	4.84	99.64	0.80
3	19	62	98.39	1.45	98.39	2.13	97.68	0.53	97.68	0.58	98.39	1.45	98.91	10.82	96.61	0.90
4	30	63	96.30	2.70	96.30	2.61	96.30	1.48	96.30	2.24	96.30	2.70	97.86	39.02	95.26	2.01
5	36	76	96.59	2.18	96.50	2.56	96.95	1.34	96.95	1.79	96.59	2.18	99.22	46.26	99.22	2.05
6	30	91									98.34	2.07	99.53	38.97	97.54	1.00
7	42	92									100	1.13	100	6.47	99.74	1.02
8	20	63									97.96	1.12	97.95	0.79	96.62	0.79
9	31	64									95.89	2.71	98.32	22.50	97.94	1.40
10	37	77									97.46	2.06	99.03	31.81	96.37	1.15
11	28	89									98.11	2.45	99.69	29.50	97.97	2.89
12	40	90									98.56	1.50	100	19.78	97.74	0.71
13	18	61									97.46	1.29	97.09	0.60	94.67	0.57
14	29	62									95.26	2.02	99.26	11.29	94.79	3.20
15	35	75									97.04	2.44	94.98	60.00	97.03	2.19
16	32	94									99.16	1.90	97.67	1.58	98.97	0.88
17	44	94									99.96	1.45	99.95	1.01	99.55	0.82
18	22	65									97.85	1.07	99.02	2.35	97.42	0.53
19	33	66									95.97	2.13	98.88	11.49	97.98	2.06
20	39	79									96.27	2.49	96.49	28.07	95.31	1.11
21	27	88									99.16	1.40	99.21	0.99	97.38	1.08
22	38	90									99.87	0.86	100	1.80	97.09	3.80
23	20	62									98.58	1.07	98.74	13.79	98.39	0.66
24	31	63									98.82	2.64	98.82	4.70	98.68	1.13
25	33	76									97.03	2.07	98.60	25.22	95.23	0.97
26	34	95									99.87	0.98	99.87	13.41	97.70	0.83
27	43	93									99.72	1.12	100	3.41	98.83	0.79
28	21	66									97.75	1.36	98.65	5.49	96.68	0.66
29	32	67									98.52	2.48	93.04	9.88	96.95	2.56
30	38	89									97.20	2.87	98.64	19.23	92.56	3.60

Table C.20. Test Results for Category # 20 (APSA Ratio: 0.10, Number of Stock Sheet Sizes: 6).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	29	90	98.87	1.86	98.80	2.35	98.17	0.56	97.50	0.67	98.87	1.86	98.99	1.08	95.59	1.38
2	41	91	99.78	1.29	99.81	2.81	99.77	0.40	99.81	0.60	99.78	1.29	99.00	4.86	99.64	0.95
3	19	62	98.39	1.75	98.39	2.06	97.48	0.63	97.11	0.80	98.39	1.75	98.91	10.82	97.15	1.07
4	30	63	96.30	3.04	96.30	3.61	96.30	1.69	96.30	2.51	96.30	3.04	98.30	55.39	95.26	1.53
5	36	76	96.59	2.50	96.70	3.05	96.95	1.51	96.95	2.11	96.59	2.50	98.68	53.55	99.22	0.26
6	30	91									98.41	2.66	98.05	1.82	97.54	1.36
7	42	92									98.00	1.27	99.00	7.70	97.74	1.18
8	20	63									97.96	1.35	98.48	1.08	97.62	1.02
9	31	64									97.89	3.19	98.41	32.55	97.88	1.93
10	37	77									98.46	2.63	98.41	39.36	97.37	1.47
11	28	89									98.11	3.12	99.69	70.41	97.87	3.10
12	40	90									98.56	1.77	100	4.12	97.24	0.86
13	18	61									98.15	1.49	98.67	6.92	97.42	0.79
14	29	62									96.06	3.25	98.86	0.73	97.96	4.10
15	35	75									98.04	2.89	98.98	80.00	98.19	3.60
16	32	94									98.69	2.28	97.67	1.58	97.09	1.09
17	44	94									98.00	1.64	100	8.28	97.66	0.96
18	22	65									97.68	1.20	99.66	0.32	97.56	0.77
19	33	66									98.17	2.12	98.88	17.83	98.07	3.90
20	39	79									97.76	2.84	97.47	42.37	98.18	1.32
21	27	88									97.69	1.43	98.52	1.75	97.38	1.30
22	38	90									98.37	0.96	100	2.35	97.95	4.50
23	20	62									98.58	1.28	98.74	31.80	98.39	0.84
24	31	63									98.82	3.13	98.22	4.17	98.18	1.34
25	33	76									97.06	2.85	98.70	73.05	97.47	1.16
26	34	95									98.37	1.31	99.87	20.42	98.26	2.01
27	43	93									98.72	1.34	99.00	4.47	98.33	0.95
28	21	66									98.31	1.28	98.65	7.82	97.56	0.78
29	32	67									98.52	3.05	93.55	12.53	97.85	3.80
30	38	89									98.20	3.48	97.64	76.23	97.24	4.60

Table C.21. Test Results for Category # 21 (APSA Ratio: 0.04, Number of Stock Sheet Sizes: 2).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	32	153	99.18	1.00	99.18	1.84	99.18	1.00	99.18	1.61	99.18	1.00	99.42	2.90	99.19	0.50
2	36	135	99.61	1.10	99.05	1.93	99.13	0.88	99.05	1.17	99.61	1.10	99.61	1.85	98.58	0.46
3	42	135	97.13	0.98	97.00	1.67	97.13	0.69	97.00	1.37	97.13	0.98	97.18	0.39	97.46	1.50
4	27	172	99.88	0.75	99.84	0.58	99.43	0.33	99.43	0.76	99.88	0.75	99.43	0.30	99.13	0.30
5	50	205	99.92	1.00	99.90	2.10	99.92	0.84	99.90	1.68	99.92	1.00	99.92	0.53	99.72	0.36
6	33	154									98.51	1.65	99.74	2.95	99.74	1.90
7	37	136									99.24	1.56	99.43	1.39	98.03	1.10
8	43	136									99.22	0.71	92.54	0.51	91.84	1.98
9	28	173									99.92	0.54	99.92	0.26	99.19	0.19
10	51	206									98.39	0.70	99.85	0.53	96.15	0.37
11	34	155									99.53	1.28	100	1.47	91.18	1.20
12	38	137									99.35	0.81	99.96	2.05	99.46	0.58
13	44	137									94.08	1.23	100	0.56	97.82	0.88
14	29	174									98.96	0.43	98.95	0.25	99.97	0.23
15	52	207									99.77	0.98	98.00	0.62	96.18	0.39
16	35	156									98.87	1.17	99.54	2.55	99.44	0.56
17	39	138									98.33	1.12	99.91	2.12	94.51	1.10
18	45	138									99.54	1.20	98.63	1.01	98.86	0.40
19	30	175									97.72	0.66	97.94	0.64	98.67	0.26
20	53	208									99.65	0.78	100	2.46	93.71	0.41
21	30	151									99.07	1.21	99.93	2.22	99.34	0.51
22	34	133									99.40	1.54	99.51	2.50	93.32	1.80
23	40	133									98.33	0.89	98.32	0.46	97.95	1.60
24	25	170									99.63	0.53	99.49	0.25	98.82	0.17
25	48	203									99.98	0.68	99.00	2.09	93.22	0.45
26	31	151									99.29	1.32	99.73	2.59	86.81	1.10
27	35	134									98.93	1.06	99.68	3.07	98.99	0.50
28	41	134									95.14	1.37	96.06	1.20	99.27	0.34
29	26	171									97.95	0.63	97.98	0.45	99.29	0.17
30	49	204									99.66	0.56	100	0.94	99.45	1.40

Table C.22. Test Results for Category # 22 (APSA Ratio: 0.04, Number of Stock Sheet Sizes: 3).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	32	153	99.18	1.90	99.18	2.15	97.98	1.19	97.98	2.84	99.18	1.90	99.42	7.13	98.63	1.69
2	36	135	99.27	1.05	98.90	2.37	99.17	1.50	99.17	1.71	99.27	1.05	99.77	3.33	99.45	0.64
3	42	135	97.13	1.63	99.53	2.93	97.13	0.84	99.53	1.91	97.13	1.63	99.64	11.55	97.60	2.50
4	27	172	99.88	1.53	98.75	2.33	98.05	0.49	98.05	0.77	99.88	1.53	100	2.89	97.31	0.40
5	50	205	99.64	1.05	99.64	1.61	99.64	0.71	99.64	0.83	99.64	1.05	100	1.80	100	0.61
6	33	154									99.41	1.64	100	5.85	100	2.98
7	37	136									99.54	1.95	99.73	4.29	99.02	2.10
8	43	136									99.22	1.57	95.98	4.37	92.66	2.99
9	28	173									99.92	0.82	100	1.71	99.92	0.35
10	51	206									98.39	1.10	100	5.23	96.15	0.51
11	34	155									99.83	1.88	100	2.93	99.40	0.66
12	38	137									99.61	1.53	99.96	4.39	99.46	0.76
13	44	137									94.08	1.89	100	0.94	95.94	0.98
14	29	174									98.99	0.68	99.80	3.02	99.97	0.30
15	52	207									100	1.13	99.00	1.18	96.16	0.60
16	35	156									99.40	1.99	99.85	4.23	99.55	1.23
17	39	138									99.47	1.58	99.91	4.03	99.43	0.80
18	45	138									99.92	1.71	99.92	38.70	96.99	1.53
19	30	175									99.68	0.94	97.97	5.56	99.45	0.39
20	53	208									99.65	1.11	100	6.25	93.71	0.65
21	30	151									99.04	1.58	99.93	4.92	99.45	0.70
22	34	133									99.91	1.70	99.91	5.94	98.68	2.84
23	40	133									96.75	1.82	99.92	23.88	85.33	0.86
24	25	170									98.96	0.83	100	5.42	99.87	0.30
25	48	203									100	0.93	100	0.89	99.80	0.76
26	31	151									99.88	1.67	99.99	6.71	99.69	2.10
27	35	134									98.93	1.53	99.85	8.29	99.68	1.13
28	41	134									98.63	1.50	98.00	17.97	100	0.59
29	26	171									97.95	0.89	98.00	5.35	99.92	0.30
30	49	204									99.66	0.97	99.97	1.94	99.97	0.46

Table C.23. Test Results for Category # 23 (APSA Ratio: 0.04, Number of Stock Sheet Sizes: 4).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	32	153	99.18	3.00	99.59	3.01	98.59	2.46	99.15	3.37	99.18	3.00	99.59	25.28	98.59	2.05
2	36	135	99.27	2.78	99.25	2.88	99.51	1.04	99.51	1.26	99.27	2.78	99.77	12.50	99.67	1.01
3	42	135	96.61	2.29	94.84	3.22	97.10	0.96	95.85	2.07	96.61	2.29	98.31	55.64	99.01	3.60
4	27	172	99.87	1.63	99.93	2.94	98.05	0.60	98.05	0.77	99.87	1.63	100	4.94	98.49	0.62
5	50	205	99.64	1.59	99.64	2.94	97.25	1.19	97.25	2.18	99.64	1.59	100	2.63	100	0.98
6	33	154									99.69	2.31	100	9.00	99.00	3.97
7	37	136									99.54	2.60	99.73	6.46	98.63	3.24
8	43	136									99.22	2.06	95.98	6.34	98.01	3.96
9	28	173									99.73	1.00	100	3.11	98.92	0.45
10	51	206									99.78	1.89	100	5.27	98.15	0.76
11	34	155									99.83	2.41	100	5.02	98.40	0.90
12	38	137									99.61	2.07	99.96	7.26	98.46	1.05
13	44	137									98.98	1.81	100	1.28	96.60	1.37
14	29	174									99.10	0.86	100	2.04	99.97	0.41
15	52	207									98.00	1.63	100	1.68	98.55	0.82
16	35	156									99.40	2.39	99.85	9.81	98.74	2.27
17	39	138									99.84	2.18	99.91	9.79	99.43	1.32
18	45	138									99.82	2.96	99.93	65.00	98.57	1.26
19	30	175									99.51	1.31	97.37	2.01	99.97	0.51
20	53	208									99.65	1.47	100	4.33	98.71	0.96
21	30	151									99.42	2.10	99.93	5.27	98.59	1.03
22	34	133									98.91	2.22	99.91	10.65	99.36	3.86
23	40	133									100	1.52	100	34.72	98.79	1.09
24	25	170									99.34	0.87	100	6.86	98.87	0.39
25	48	203									98.00	1.51	100	1.24	98.00	0.78
26	31	151									99.88	2.16	99.99	7.50	98.69	3.01
27	35	134									99.73	2.30	99.85	9.95	99.48	2.24
28	41	134									98.63	2.14	99.00	30.56	99.00	0.67
29	26	171									97.95	1.06	100	5.91	98.92	0.41
30	49	204									98.00	1.25	100	2.91	99.97	0.57

Table C.24. Test Results for Category # 24 (APSA Ratio: 0.04, Number of Stock Sheet Sizes: 5).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	32	153	99.03	2.92	99.03	3.44	99.59	2.69	99.15	4.16	99.03	2.92	99.72	12.88	98.59	2.72
2	36	135	99.38	2.51	99.00	3.79	99.51	1.22	99.51	1.49	99.38	2.51	99.77	7.24	99.67	1.28
3	42	135	98.61	3.88	94.84	3.81	97.63	1.06	93.36	2.41	98.61	3.88	98.64	78.13	95.96	1.97
4	27	172	99.58	1.90	99.58	3.20	99.05	0.66	99.05	1.07	99.58	1.90	100	5.62	98.49	0.73
5	50	205	99.64	1.95	99.64	3.25	96.25	1.27	98.25	2.32	99.64	1.95	100	3.91	100	1.22
6	33	154									99.79	2.42	100	10.41	100	3.90
7	37	136									99.42	2.71	99.73	10.15	99.32	3.20
8	43	136									99.22	2.58	96.30	11.22	99.00	4.10
9	28	173									99.66	1.45	100	5.14	99.92	0.52
10	51	206									99.78	2.30	100	8.37	99.15	1.07
11	34	155									99.83	2.89	100	7.83	98.64	1.21
12	38	137									99.61	2.66	99.96	11.22	99.46	1.24
13	44	137									98.98	2.25	100	1.83	98.34	1.61
14	29	174									99.10	1.12	100	4.01	99.00	0.46
15	52	207									99.00	1.97	100	2.01	98.55	1.03
16	35	156									99.40	2.92	99.85	12.35	99.72	1.18
17	39	138									99.84	2.87	99.91	11.70	99.09	2.48
18	45	138									99.54	4.21	100	95.21	98.56	3.95
19	30	175									98.77	1.19	97.17	30.00	98.97	0.62
20	53	208									99.65	1.91	100	5.52	98.71	1.20
21	30	151									99.42	2.61	99.93	6.55	99.59	2.34
22	34	133									98.88	2.46	99.91	17.01	98.70	3.90
23	40	133									99.00	2.07	100	50.00	99.24	2.21
24	25	170									99.67	1.07	100	7.76	99.37	0.46
25	48	203									99.00	1.79	100	1.49	99.00	0.95
26	31	151									98.88	2.79	99.99	8.83	99.69	3.80
27	35	134									99.73	2.46	99.97	10.28	99.73	3.32
28	41	134									98.63	3.16	99.00	31.40	99.00	0.80
29	26	171									99.72	1.08	100	40.00	99.92	0.51
30	49	204									99.00	1.69	100	2.96	99.00	0.76

Table C.25. Test Results for Category # 25 (APSA Ratio: 0.04, Number of Stock Sheet Sizes: 6).

BOM #	Number of Order Piece Sizes	Total Number of Order Pieces	LTM-MEAN		LTM-MEDIAN		LAM-MEAN		LAM-MEDIAN		PSTL		PREORDER		STEP	
			ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)	ASSU (%)	CPU Time (min.)
1	32	153	99.03	4.39	99.03	4.20	99.21	1.96	99.21	3.05	99.03	4.39	96.79	23.02	99.01	2.86
2	36	135	99.38	3.04	99.00	3.53	99.53	1.51	99.53	1.91	99.38	3.04	99.77	9.87	99.67	1.70
3	42	135	98.78	2.89	98.78	4.02	97.13	1.25	98.33	2.74	98.78	2.89	99.64	65.31	95.96	2.23
4	27	172	99.58	1.05	99.58	3.59	98.46	0.79	98.46	1.15	99.58	1.05	100	1.86	98.78	0.90
5	50	205	99.64	2.34	99.64	3.32	99.00	0.93	98.99	2.48	99.64	2.34	100	4.39	100	2.10
6	33	154									99.53	3.91	100	13.91	99.00	3.80
7	37	136									99.41	3.80	99.79	14.21	99.17	4.20
8	43	136									99.22	3.15	100	8.90	99.00	3.90
9	28	173									99.66	1.79	100	6.97	99.92	0.61
10	51	206									98.93	2.59	100	12.08	98.53	1.34
11	34	155									98.93	3.54	100	9.68	99.64	2.26
12	38	137									98.61	3.27	100	19.99	99.46	2.70
13	44	137									98.98	3.62	100	2.28	98.72	2.90
14	29	174									99.10	1.33	100	3.89	99.00	0.52
15	52	207									99.00	2.33	100	3.33	98.55	2.30
16	35	156									99.67	3.15	99.85	31.75	99.72	2.80
17	39	138									99.79	3.68	99.91	16.52	99.51	3.50
18	45	138									99.79	3.67	100	100	99.53	4.10
19	30	175									99.84	1.57	98.17	60.00	99.90	0.81
20	53	208									99.65	2.29	100	6.07	99.08	1.42
21	30	151									99.42	3.02	99.93	12.38	99.79	1.73
22	34	133									99.88	3.00	99.91	29.52	99.36	2.03
23	40	133									99.00	2.35	100	80.00	99.47	3.45
24	25	170									99.99	1.24	100	6.27	99.87	0.54
25	48	203									99.00	2.10	100	2.35	99.00	2.24
26	31	151									99.88	3.40	99.99	25.24	99.69	3.90
27	35	134									99.73	3.07	99.97	21.63	99.86	3.98
28	41	134									99.50	4.04	99.00	50.00	99.00	0.89
29	26	171									99.72	1.33	98.02	80.00	99.00	0.67
30	49	204									99.00	2.03	100	3.55	99.00	0.92