Order Number 1330544

Multiple choice modular design problem experimental results and sensitivity analysis

Cheraghi, Seyed Hossein, M.S.

THE UNIVERSITY OF ARIZONA, 1987

U·M·I
300 N. Zeeb Rd.
Ann Arbor, MI 48106

PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark __√__.

1. Glossy photographs or pages ______
2. Colored illustrations, paper or print ______
3. Photographs with dark background ______
4. Illustrations are poor copy ______
5. Pages with black marks, not original copy ______
6. Print shows through as there is text on both sides of page ______
7. Indistinct, broken or small print on several pages __√__
8. Print exceeds margin requirements ______
9. Tightly bound copy with print lost in spine ______
10. Computer printout pages with indistinct print ______
11. Page(s) __________ lacking when material received, and not available from school or author.
12. Page(s) __________ seem to be missing in numbering only as text follows.
13. Two pages numbered ______. Text follows.
14. Curling and wrinkled pages ______
15. Dissertation contains pages with print at a slant, filmed as received ________
16. Other ____

MULTIPLE CHOICE MODULAR DESIGN PROBLEM
EXPERIMENTAL RESULTS AND SENSITIVITY ANALYSIS

by

SEYED HOSSEIN CHERAGHI

A Thesis Submitted to the Faculty of the

DEPARTMENT OF SYSTEMS AND INDUSTRIAL ENGINEERING

in Partial Fulfillment of the Requirements
For the Degree of

MASTER OF SCIENCE
WITH A MAJOR IN INDUSTRIAL ENGINEERING

In the Graduate College

THE UNIVERSITY OF ARIZONA

1 9 8 7

STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at the University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interest of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: S. R. Cherasgn

APPROVAL BY THESIS DIRECTOR

This thesis has been approved on the date shown below:

Jeff M. Goldberg

J. B. GOLDBERG
Assistant Professor, Department of
Systems and Industrial Engineering

5/1/87
Date

ACKNOWLEDGMENTS

I would like to express my special thanks to Dr. J. B. Goldberg, my thesis advisor, for his guidance and advise through all phases of this research. I also would like to thank Dr. T. C. John and Dr. S. Sen for spending their valuable time in evaluating this thesis.

TABLE OF CONTENTS

LIST OF TABLES

ABSTRACT

Shaftel's model for integer modular design cannot be applied to a situtation where there are choices for part substitution. In comparison Goldberg's model overcomes the part substitution limitation. In this thesis, we give experimental results that show that Goldberg's heuristic is an efficient heuristic with respect to its ability to find the optimal solution and the time required to solve problems. Sensitivity analysis in multiple choice modular design model is a complicated matter because of the structure of a model. We give analytical results for some cases where bounds can be found.

CHAPTER 1

INTRODUCTION

A company produces several kinds of end items, each of which requires a specific number of parts. These end items can be assembled directly from parts that are placed around the assembly line. This system is accepted and is commonly used by manufacturers. But there may be a more efficient way of assembling end items by first producing standard modules and then using these modules to produce the end items.

The following reasons justify interest in assembling end items from standard modules:

1- it may be easier to assemble an end item from a small number of modules than from several different parts.

2- it may be more efficient to repair a end item by looking for a broken part in the small area (module) than in the whole end item.

Using standard modules in assembling the end items may require more parts than are needed by the end items. The cost of giving away extra parts may be justified by the economic advantages of manufacturing standard modules.

An example of the application of modular design can be seen in electronic assembly technology. A company produces several kinds of televisions,radios and electronic calculators.

Each of these end items has a specific requirement, for example, of inductors, resistors and capacitors. The company can produce standard modules of circuits that can be used in each of the end items, and using different number of these modules, final end items can be produced.

The modular design problem was first presented by David Evans [1963] as follows:

> J end items are to be produced from I parts. Each end item is made up of a variety of parts with multiples of specific parts. The manufacturer has the option of producing all of the end items directly from the parts or producing standard modules of the parts and then producing the end items from these modules. It is assumed that only one standard module is used. The problem is to design the part mixture of the standard modules. The number of modules used in each end item must also be determined.

Evans' nonlinear formulation is given as follows:

Let R_{ij} = number of part i required in end item j, $i = 1 \ldots I$, $j = 1 \ldots J$

C_i = cost for part i, $i = 1 \ldots I$

d_j = demand for end item j, $j = 1 \ldots J$

Y_j = number of modules needed to produce end item j, $j = 1 \ldots J$.

X_i = number of part i used in the module, $i = 1 \ldots I$

The objective is to determine the values for X_i, Y_j that minimize the total cost of parts that required to satisfy the demand for end items. The formulation is:

Minimize

$$\sum_{i=1}^{I} C_i X_i \sum_{j=1}^{J} d_j Y_j \tag{1.1}$$

subject to:

$$X_i Y_j \geq R_{ij} \quad \text{for all pairs of i, j,} \tag{1.2}$$

$$X_i > 0, \quad i = 1 \ldots I \tag{1.3}$$

$$Y_j > 0, \quad j = 1 \ldots J \tag{1.4}$$

Equation (1.1) is the objective function and represents the total cost of parts used to satisfy the demand for end items. Constraint (1.2) tells us that the number of each part used in each end item, should be at least equal to the requirement of that part in each end item.

One of the problems with Evans' model is that it does not consider the possibility of part substitution. Sometimes it is technically possible and economically advantageous to substitute some parts with others that have different technical characteristics and costs, but can satisfy the same requirement. This is the basic subject to be discussed in this thesis. We examine Goldberg's [1986] model and associated solution heuristic for incorporating multiple choice part selection decisions. We look at various sensitivity analyses such as varying the cost parameter of a group of parts, varying the demand parameter of end items and varying the technological requirements for a group of part-end item pairs.

In the next chapter, we will review the literature and discuss the Shaftel's algorithm to solve an integer version of Evans' model. Shaftel's algorithm is a subroutine in Goldberg's heuristic.In the third chapter we will discuss Goldberg's model for multiple choice modular design and his heuristic to solve this model. An optimal procedure for solving the multiple-choice modular design problem is given, and the computational effectiveness of the Goldberg's heuristic is tested. In the last chapter we discuss sensitivity analysis when various parameters in the model are changed.

CHAPTER 2

LITERATURE REVIEW

In this chapter we discuss the literature concerned with solving Evans' model. We also discuss Shaftel's algorithm for solving the integer variable extension of Evans' model.

Several authors have derived algorithms to solve Evans' model. Passy [1970] and Smeers [1974] solve the model as a geometric programing problem. Shaftel and Thompson [1977] derive an efficient solution procedure that is similar to the transportation algorithm of linear programing. Their algorithm, as they claim, can solve problems as large as 30,000 constraints (300 parts, 100 end items) in 480 seconds of computational time (UNIVAC 1108).

All of the above approaches solve the modular design problem as a problem with continuous values. Evans believes that, once the continuous version of the problem is solved, an intelligent rounding procedure will give an optimum integer solution.

Abrams and Charnes [1972] give a procedure to round off the continuous solutions to the modular design problem. Samuelson and Thompson [1974] use "branch and bound" to solve the integer version of the problem. Both above approaches make use of the Shaftel and Thompson's algorithm.

Shaftel [1971] gives an integer branch and bound technique to solve the modular design problem. His procedure is fully described here. Mathematically Evans' model with integer variables is stated as follows:

Minimize

$$\sum_{i=1}^{I} C_i X_i \sum_{j=1}^{J} d_j Y_j \qquad (2.1)$$

subject to:

$$X_i Y_j \geq R_{ij}, \text{ for all pairs of } i, j, \qquad (2.2)$$

$$X_i > 0, \quad i = 1 \ldots I \qquad (2.3)$$

$$Y_j > 0, \quad j = 1 \ldots J \qquad (2.4)$$

$$X_i, Y_j \text{ are both integer.} \qquad (2.5)$$

Throughout this thesis X and Y are considered as vectors and X_i and Y_j are the elements of these vectors unless otherwise specified.

Shaftel's algorithm is a search procedure that uses partial enumeration. It can be used to solve small modular design problems for their optimal integer solution. In Shaftel's method complete enumeration is reduced to partial enumeration by using two bounding techniques, "masking" and "dominance". The bounding techniques reduces the total number of necessary searches enormously.

A complete search algorithm simply looks at all possible Y values and find the corresponding X using the following formula.

$$X_i = \max_j \; [R_{ij}/Y_j] \;, \quad i = 1 \ldots I \qquad (2.6)$$

(Where [a] is the smallest integer greater than a).
Then the solution to the problem would be (X , Y) that minimizes (2.1). The reason this is optimal is that since X_i is integer, for all i, any integer value less than X_i calculated in (2.6) will not satisfy some constraints of the form $X_i Y_j \geq R_{ij}$, and also any integer value grater than X_i found in (2.6) will simply increase the value of objective function, because all other parameters in the objective function (C_i , $\sum_{j=1}^{J} d_j Y_j$) are fixed and only the value of X is increased.

Shaftel proves that there is a bound on the maximum value of Y_j .

Lemma 1: For the modular design problem the maximum number of solutions to be searched is equal to $\prod_{j=1}^{n} \mathrm{Max}_i [R_{ij}]$, where R_{ij} is an element of the requirement matrix.

Proof: We have $X_i = \mathrm{Max}_i [R_{ij}/Y_j]$. As Y_j increases and becomes greater than R_{ij}, the lowest value that X_i can be is one. Now assume that all the elements of Y remain constant (when $Y_j > R_{ij}$) except one element is increased, then there will not be any change in the value of X and the only change that occurs is that the value of the objective function will increase. Hence the maximum number of solutions that must be considered will be $\prod_{j=1}^{n} \mathrm{Max}_i [R_{ij}]$.■

As was mentioned before, the search is reduced through masking and dominance. Masking and dominance are two types of bounding techniques which are used to eliminate unnecessary searches for those solutions that have a higher value of objective function than one already found.

Masking occurs when $X'_i = X_i$, for all i, while $Y'_j \geq Y_j$, for all j, and $Y'_k > Y_k$ for at least one k. When this happens, the value of objective function will only increase because the first part of the objective function ($\sum_{i=1}^{I} C_i X_i$) will be fixed, and the second part ($\sum_{j=1}^{J} d_j Y_j$) is increased. Thus; Since the value of objective function is increased by changing Y to Y′, then Y′ is masked by Y.

Dominance is defined as a property that $[R_{ik}/Y_k] \leq [R_{ij}/Y_j]$ for all i and some $j \neq k$. When Y exceeds a certain value and the above inequality is satisfied, then all the values Y′ greater than Y will only increase the value of objective function. Therefore they are masked by Y if the remaining Y values are held constant. A small sample is solved by Shaftel using the above procedure. A problem with requirement matrix of size (4*3) with $Max_i[R_{ij}]$ of (34,23,44) is solved by in about six seconds on the Univac 1108. In this problem the total number of searches is reduced from 34,408 possible solutions to 850 possible solutions. A problem with 4,359,480 solutions was solved in eleven seconds. Shaftel mentions that his procedure is able to solve problems as big as 4 parts and 10 end items with requirements of up to 500 units.

Goldberg [1984] develops a model in which there are several parts that can satisfy the same requirement. In this case decisions that are to be made, besides the number of parts in a module, will be the choice of parts to be used. This model is called "multiple choice modular design" and is described in the next chapter.

CHAPTER 3

MULTIPLE CHOICE MODULAR DESIGN MODEL AND EXPERIMENTAL RESULTS OF THE GOLDBERG'S HEURISTIC

In this chapter we describe Goldberg's model for multiple-choice modular design and his heuristic to solve the model. We present the procedure used to find the optimal solution for the multiple-choice modular design model and we give experimental results of the heuristic on 110 small problems.

3.1 Multiple Choice Modular Design

Goldberg [1986] extends Evans' model to allow for several groups of parts such that every part in each group can be substituted for any other part in that group. Each part has a fixed cost, a variable cost and a strength. The strength of a part is the number of that part equivalent to one unit of the group-end item requirement. The group-end item requirements is the matrix that shows the total requirement of any group of parts in each end item. As in Evans' model there will be only one standard module. Decisions to be made are:

1. Number of modules for each end item.
2. Parts from each group to be used.
3. Quantity of each part used in the module.

Mathematically the problem is given as follows:

Variables:

Let X_{si} = number of part i from group s used in the module, $s = 1 \ldots S$, $i = 1 \ldots I$.

Y_j = number of modules used in the production of end item j, $j= 1 \ldots J$

$$\delta_{si} = \begin{cases} 1, & \text{if } X > 0 \text{ (if part i from group s is chosen).} \\ 0, & \text{otherwise} \end{cases}$$

Parameters:

Let F_{si} = fixed cost associated with part i from group s, $i = 1 \ldots I$, $s = 1 \ldots S$.

C_{si} = variable cost associated with part i from group s, $i = 1 \ldots I$, $s = 1 \ldots S$.

d_j = demand for end item j, $j = 1 \ldots J$.

R_{sj} = total strength of group s required in end item j, $s = 1 \ldots S$, $j = 1 \ldots J$

ω_{si} - per unit strength of part i in group s, $i = 1 \ldots I$, $s = 1 \ldots S$.

It is assumed that only one part from each group can be chosen to form the module, and it is also assumed that from each group there must be a part in some end item. The objective is to find Y and X such that to minimize the total part cost. The model is:

Minimize

$$\sum_{s=1}^{S}\sum_{i=1}^{I_s} \delta_{s\,i} F_{s\,i} \; + \; \sum_{j=1}^{J} d_j Y_j \quad \sum_{s=1}^{S}\sum_{i=1}^{I_s} C_{s\,i} X_{s\,i} \tag{3.1}$$

subject to:

$$\sum_{i=1}^{I_s} \omega_{s\,i} X_{s\,i} Y_j \;\geq\; R_{s\,j}\,, \qquad \text{for each pair of } (s,\ j) \tag{3.2}$$

$$\sum_{i=1}^{I_s} \delta_{s\,i} = 1, \qquad s = 1 \ldots S, \tag{3.3}$$

$$X_{s\,i} - M\delta_{s\,i} \leq 0, \qquad \text{for each pair of } (s,\ i) \tag{3.4}$$

$$X_{s\,i} \geq 0, \qquad s = 1 \ldots S,\ i = 1 \ldots I_s \tag{3.5}$$

$$Y_j \geq 1, \qquad j = 1 \ldots J \tag{3.6}$$

$$\delta_{s\,i} \in (0,1) \tag{3.7}$$

$$X_{s\,i}\,,\ Y_j \qquad \text{are integer} \tag{3.8}$$

Equation (3.1) is the objective function and gives the total cost of parts used to form end items. Constraint (3.2) indicates the total number of parts from each group used in every end item has to satisfy the strength requirement for that end item - group pair. Constraint (3.3) along with constraint (3.4) imply that only one part from each group can be chosen. Constraints (3.5) and (3.6) are non negativity constraints. Constraint (3.8) tells us that $X_{s\,i}$, Y_j, for all s,i, and j, must be integer.

If the values of δ_{si} are given and feasible,then the model would be the same as Shaftel's model. So if one is able to choose the right part from each group, then Shaftel's algorithm will give the optimal solution for determining the number of each part in the module. Goldberg's heuristic tries to select the correct set of parts using a greedy procedure. And then uses Shaftel's algorithm to solve the resultant modular design problem. The process repeats until no improvement can be found. The heuristic is given as follows:

Step 0: Choose the initial value for vector Y. Set the current Y to this vector.

Step 1: Using current Y, choose one part from each group that minimizes the following function:

$$F_{si} + C_{si}(\mathrm{Max}_j [R_{sj}/\omega_{si} \ Y_j]) \sum_{j=1}^{J} d_j Y_j , \ _s = 1 \ldots S, \ i = 1 \ldots I$$

where [a] is the smallest integer greater than a.

Step 2: Based on the parts chosen in step 1, solve the Shaftel's algorithm. Denote this solution as (X^*, Y^*).

Step 3: If Y^* is equal to the current Y, then stop, (X^*, Y^*) are the final solution to the problem. Else, set the current Y equal to Y^* and go to step 1.

Goldberg shows that the objective decreases in each iteration of the heuristic and that an optimal solution will contain only one part from each group. Nothing is mentioned about the computational effectiveness of the heuristic. To test effectiveness, an optimal procedure has been developed. It is discussed next.

3.2- OPTIMAL PROCEDURE TO SOLVE THE GOLDBERG´S MODEL

The heuristic procedure given in the first section of this chapter should be tested for both computational time and its ability to find the optimal solution. For this purpose and for the purpose of sensitivity analysis given in the next chapter, we devise a procedure to find the optimal solution.

As it was mentioned in the previous section of this chapter, if one could find the right part from each group, then the Shaftel´s algorithm could be easily used to solve the model to optimality. The procedure given in this section is a complete search procedure similar to Shaftel´s algorithm. In this procedure all the possible values for Y are enumerated and the best Y are chosen. Given the Y values, X´s will be found from each group to minimize :

$$F_{si} + C_{si} X_{si} \sum_{j=1}^{J} d_j Y_j , \quad s = 1 \ldots S, \ i = 1 \ldots I_s \qquad (3.9)$$

where $X_{si} = \text{Max}_j [R_{sj} / \omega_{si} Y_j]$, for all s, i.

and [a] is the smallest integer greater than a.

Finally the objective function is evaluated for each potential solution. The complete search procedure is given as follows:

Step 0 : set Y = 1

Step 1 : Find the corresponding part number and the quantity of that part from each group that minimizes (3.9). Denote the current solution as (X , Y).

Step 2 : Calculate the objective of (X , Y) and see if it is the best so far. If it is the best, store (X , Y) as the incumbent.

Step 3 : increment the value of only one element of Y by one and go to step 1. Continue this step until all values of Y are considered.

As in Shaftel's algorithm, the number of searches necessary is bounded.

Lemma 2: For any multiple choice modular design problem with the given requirement of R_{sj} and strength of ω_{si} for each part in each group, the total number of search required is given by

$$\prod_{j=1}^{n} \text{Max}_s \, \text{Max}_i \, [R_{sj} / \omega_{si}]$$

Proof: proof is exactly the same as the one given for Shaftel's algorithm, where R will be replaced by $(R_{sj} \; / \; \omega_{si})$.■

To reduce the total number of searches, masking and dominance can be used. Masking and dominance used here are different from those used in Shaftel's algorithm. In Shaftel's procedure, there are no groups, so masking occurs as soon as we get the same value for the number of parts for two sets Y and Y' with Y ≥ Y'. But here we have several different parts in each group, thus when Y changes we have no idea what part from each group will be chosen, so masking cannot take place. Therefore masking is the property that:

1- the same part is used from each group.

2- the quantity of these parts are not changed ($X'_{si} = X_{si}$, for all s for a chosen i) while $Y'_j \geq Y_j$, for all J, and $Y'_k > Y_k$ for at least one k.

In this situation since the same part is chosen and the quantity of that part has not changed, then the value of the objective function will only increase as Y_j increases. The objective value using Y is equal to

$$Z = \sum_{s=1}^{S}\sum_{i=1}^{I_s} F_{si} + \sum_{j=1}^{J} d_j Y_j \quad \sum_{s=1}^{S}\sum_{i=1}^{I_s} C_{si} X_{si}$$

and the objective value using Y' is equal to

$$Z' = \sum_{s=1}^{S}\sum_{i=1}^{I_s} F_{si} + \sum_{j=1}^{J} d_j Y'_j \quad \sum_{s=1}^{S}\sum_{i=1}^{I_s} C_{si} X'_{si}$$

Since the same parts are chosen, F, C, ω and X are the same for both cases, and because $Y'_j > Y_j$, for at least one j, $\sum_{j=1}^{J} d_j Y'_j > \sum_{j=1}^{J} d_j Y_j$, so Z' > Z and Y' can be masked.

Dominance is the property that:

$[R_{sk}/\omega_{si} Y_k] \leq [R_{sj}/\omega_{si} Y_j]$, for all s and all i and some $k \neq j$.

After Y_k exceeds a certain value, then all the values greater than Y_k will only increase the value of objective function and therefore are masked by Y_k if the remaining Y_j values are fixed. Masking and dominance are used in step 2 of the optimal procedure to avoid calculating redundant objective values. However in our programs only masking is implemented.

3.3 Experimental results of Goldberg's heuristic

In this section, we compare the Goldberg's heuristic against the optimal procedure. The purpose of this comparison is to find the overall quality of the heuristic. To do so, two computer programs are written, one for the heuristic and one for the optimal procedure. The programs are written in fortran 77 computer language. All computer runs are made on the University of Arizona VAX 11/780 computer.

To compare the two procedures, 110 sample problems were generated based on the following characteristics:

- Fixed costs (F) are integer numbers between (5 - 1200).
- Part group-end item requirements (R) are integers between (1 - 30).
- Demand for end items (d) are integers between (5 - 1200).
- Strength of each part (ω) is a continuous number between (1 - 5).

Since the maximum number of solutions needed to be searched is $\prod_{j=1}^{n} \text{Max}_s \text{Max}_i [R_{sj} / \omega_{si}]$, values for R_{sj} and ω_{si} were chosen such that the computer was able to carry out the computation within a reasonable period of time.

The sizes of the sample problems are given in table 1.

Table 1

Size of the Sample Problems

Size (end item,group)	Total number of parts	Average # of possible solutions	Standard deviation	Number
A = (4, 6)	12	26355	20000	23
B = (4, 5)	12	25500	18326	21
C = (4, 4)	12	31460	36214	21
D = (4, 3)	12	25696	23037	21
E = (4, 2)	12	20853	21380	21
F = (9, 5)	15	1362625	1055623	5

The first task is to check the ability of the heuristic to find the optimal solution. Table 2 displays the number of times that heuristic has been able to find the optimal solution for different sizes of problems. It also shows the average percent deviation from optimality in the case where the heuristic does not give an optimal solution.

Table 2

Results of the Optimality Test for the Heuristic

Size	total number of test	number of times optimal	Percent optimal	average percent off from optimality
A	23	19	82.6	0.319
B	21	15	71.43	0.356
C	21	18	85.7	0.284
D	20	17	85	0.62
E	20	12	60	0.42
F	5	3	60	0.74
Total	110	84	76.36	0.456

It can be seen from table 2 that, 76.36% of the time, the heuristic has found the optimal solution. As shown in Table 2, the overall average deviation of non optimal solutions from the optimality is 0.456%. This deviation is calculated as a percentage of the range of the maximum possible value of objective function and the optimal solution. The following formula is used:

$$PD = \frac{(HV - OV) * 100}{(MPV - OV)}$$

where PD = percent deviation

HV = heuristic objective value

OV = optimal value of objective function

MPV = lower bound for the maximum possible value of objective function.

Because of the bounding techniques used, all the possible solutions are not checked and so MPV might be lower than the actual maximum possible value of the objective function. Therefore, one can say that the overall actual average deviation of the heuristic should be less than 0.456% and that this value is the upper bound of the deviation.

The standard deviation calculated for these deviations is 0.226, and shows a narrow distribution of the deviations from optimality. Table 2 also tells us that as the total number of parts in each group increases, the possibility of getting optimal solution decreases. This can be explained since the reason that heuristic can not find the optimal solution is that it is unable to find the right part in some cases. Since the number of parts increases, the possibility of finding the right part decreases if they are close in terms of ω,C, and F.

In order to be able to find the size of the problems that can be solved by Goldberg's heuristic, we need to know the number of times that the heuristic solves a Shaftel's modular design problem. Table 3 displays the result.

Table 3

Number of Iterations for Goldberg's Heuristic

Number of iterations	Number of problems	Percent of the total
1	30	27.2
2	78	71
3	2	1.8
more than 3	None	None

From Table 3, we see that in more than 71% of the cases the heuristic has terminated after two iterations of Shaftel's procedure. And since Shaftel's procedure is the base for Goldberg's heuristic, we conclude that Goldberg's heuristic can solve problems with the same size as Shaftel's algorithm. Shaftel's algorithm solves problems with (4 * 10) requirement matrix with the requirement of up to 500 units, therefore Goldberg's heuristic will be able to solve problems with the same requirement matrix and $\max_n \max_i [R_{sj}/\omega_{si}] \le 500$. Note that number of parts in each group does not have any effect on the size of the problem since computer has to do very simple calculations to those among parts in a group. Also if different subroutine is used in place of shaftel's algorithm, then larger problems can be solved with the heuristic.

It is obvious that the optimal procedure requires considerably more time than the heuristic to solve the problems. In one case the problem with (4 * 4) requirement matrix and with the total possible number of solutions equal to 1,388,520 was solved by heuristic in 24.7 seconds of cpu time. But it took more than 13 minutes of cpu time for the optimal procedure and the problem was still unsolved.

We obtained the following conclusions from the above experiment:

- heuristic has given an optimal solution for more than 76% of the sample problems and when the heuristic does not find the optimal solution, its objective value is less than 0.45% off of the optimal value. Considering the time needed to solve the problem by both procedures and the size of the problem that can be solved by both procedures, it can be concluded that Goldberg's heuristic is a more efficient procedure. This conclusion is based on the sample problems tested. Sample problems were not very large problems and only in 5 cases, problems with 9 end items were tested. Conclusion for larger problems are speculative, however it is not uncommon for heuristics to perform well on large problems given they perform well on reasonable sized problems.

In the next chapter, we discuss sensitivity analysis in the multiple choice model. We also look at cases such as varying the cost parameter of a group of parts, varying the demand parameter of any end items and varying the technological requirements for a group - end item pair.

CHAPTER 4

SENSITIVITY ANALYSIS

When applying linear or nonlinear programming, numerical data are usually stochastic. They are only rough estimates of quantities that are difficult to measure or predict. Over time, various changes in some parameters may occur, for example, the demand of a product might change, the cost of a specific part might change and the requirement of a part in the end item might change because of the change in the design of the end item. Whenever a change occurs, one would like to know the impact on the design and the production system. In this chapter we describe the behavior of the multiple choice model when any parameter within the system changes.

Recalling from chapter 3, the model is represented as:

Minimize

$$\sum_{s=1}^{S}\sum_{i=1}^{I_s} \delta_{si} F_{si} + \sum_{j=1}^{J} d_j Y_j \sum_{s=1}^{S}\sum_{i=1}^{I_s} C_{si} X_{si} \qquad (4.1)$$

subject to:

$$\sum_{i=1}^{I_s} \omega_{si} X_{si} Y_j \geq R_{sj}, \text{ for all } (s, j) \text{ pairs} \qquad (4.2)$$

$$\sum_{i=1}^{I_s} \delta_{si} = 1, \qquad s = 1 \ldots S, \qquad (4.3)$$

$$X_{si} - M\delta_{si} \leq 0, \qquad \text{for each } (s, i) \text{ pair}, \qquad (4.4)$$

$$X_{si} \geq 0, \qquad s = 1 \ldots S,\ i = 1 \ldots I \qquad (4.5)$$

$$Y_j \geq 1, \qquad j = 1 \ldots J \qquad (4.6)$$

$$\delta_{si} \in (0,1) \qquad (4.7)$$

$$X_{si},\ Y_j \qquad \text{are integer} \qquad (4.8)$$

After Y is found, X is calculated from each group such that to minimize the following part selection formula defined by "T":

$$T = F_{si} + C_{si} X_{si} \sum_{j=1}^{J} d_j Y_j, \qquad s= 1 \ldots S,\ i = 1 \ldots I_s, \qquad (4.9)$$

where

$$X_{si} = \mathrm{MAx}_j\, [R_{sj} / \omega_{si} Y_j\,], \text{ for all } s, i.$$

In the above problem there are five parameters that can vary during the time; F,d,c,ω and R. F,C and ω are part related parameters and have the property that if the value of one of them for any single part in any group is changed, there will be no effect on the value of "T" for other parts. R is the group related parameter and has a property that any change in the value of R for any group-end item pair will effect on "T" for parts in the affected group. And finally d is an end-item related parameter and has a property that any change in the value of d will effect "T" for all parts in all groups. These properties have a very important role in sensitivity analysis of the model. We now discuss the sensitivity analysis of the model for each parameter.

4.1 Part Related Parameters

F,C and ω are part related parameters. The sensitivity of the model to F in some cases is different from that to C and ω. ω is an inverse parameter since high ω values are good while high C values are not. Therefore any result obtained for C, can be "inversed" to include ω. In this part we discuss the sensitivity of the multiple choice model on F and C. One can then extend the result for C to ω considering the above statement.

4.1.1 Non-used Parts

The value of fixed cost (variable cost) of one or several unchosen parts in one or more groups is increased. In this case there will be no change in the module design. This is obvious since the value of objective function for the current optimal solution does not change while for some of the nonoptimal solutions, the value of objective function increases.

4.1.2 Proportional Change in all Parts in all Groups

When the value of fixed cost for all parts in all groups are increased (decreased) proportionally, that is if $F \rightarrow \alpha F$, for $\alpha>1$ $(\alpha<1)$, then if the fixed cost for the chosen part in each group is the smallest (largest) fixed cost for all parts in that group, the design remains unchanged.

Proof: Assume that from each group, part k is chosen to be in optimal solution. Also assume that part h is any other part in each group. Then the following inequality holds for each group.

$$F_k + C_k X_k \sum_{j=1}^{J} d_j Y_j \leq \alpha F_h + C_h X_h \sum_{j=1}^{J} d_j Y_j \qquad (4.10)$$

Now if fixed costs are increased proportionally (F αF), then we need to prove that

$$\alpha F_k + C_k X_k \sum_{j=1}^{J} d_j Y_j \leq \alpha F_h + C_h X_h \sum_{j=1}^{J} d_j Y_j \qquad (4.11)$$

and show that part k will still be selected. Inequality (4.9) can be written as

$$(\alpha F_k - F_k) + F_k + C_k X_k \sum_{j=1}^{J} d_j Y_j \leq (\alpha F_h - F_h) + F_h + C_{h x h} \sum_{j=1}^{J} d_j Y_j \qquad (4.12)$$

If $F_k < F_h$ then $(\alpha - 1)F_k < (\alpha - 1)F_h$, for $\alpha > 1$,

or $\alpha F_k - F_k < \alpha F_h - F_h$. (4.13)

From (4.10) and (4.13) it is concluded that inequlity (4.12) and as a result inequality (4.11) holds as long as $\alpha > 1$.■

If the above condition for chosen parts does not hold, then the design may change. This follows since as F changes proportionally ($F \rightarrow \alpha F$), T_{si} does not.

When the value of variable cost (C) for all parts in all groups are increased (decreased) proportionally, then if the value of $C_{si}X_{si}$ for the chosen part in each group is the smallest (largest) for all parts in that group, design remains unchanged. Proof is the same as the one for F, except that α is multiplied by its corresponding variable C.

4.1.3 Proportional Change in all Parts in one Group

If the value of F for all parts in one specific group is changed proportionally at the same rate, then results obtained in part A of 4.1.2 will still hold. But if the value of C for all parts in a specific group is changed proportionally, the results obtained in part B of 4.1.2 does not hold and the design may change. This difference between F and C comes from the fact that F only appears in part selection formula and does not have any role in solving Shaftel's algorithm, therefore as long as the same part in a group is chosen, Shaftel's algorithm will give the same result. In contrast any change in C will affect both part selection formula and Shaftel's result and is considered as a irregular change that is described next.

4.1.4 Used part

The value of F or C for a chosen part is increased. The behavior of the model for any irregular change and any proportional change that does not have the condition mentioned in 4.1.2, is very complicated. We describe the behavior of the model for any irregular change in the value of a fixed cost by the following example.

Example 1:

Number of end items = 4

Number of groups of parts = 6

Number of parts in each group = 2

Group - end item requirements are given in table 4, also group cost and strength data are given in table 5. The optimal solution to this problem is given by:

$Z^* = 5176$

$Y^* = (2,1,2,2)$ and

$X^*_{12} = 1,\ X^*_{21} = 2,\ X^*_{32} = 1,$

$X^*_{41} = 2,\ X^*_{52} = 2,\ X^*_{62} = 2.$

The second best solution for this problem is given as:

$Z^1 = 5225.3$

$Y^1 = (1,1,1,1)$ and

$X^1_{12} = 2,\ X^1_{21} = 4,\ X^1_{32} = 2,$

$X^1_{41} = 3,\ X^1_{52} = 2,\ X^1_{61} = 3.$

As we see from the optimal solution and table 5, the condition required in 4.1.2 for chosen parts does not hold in the second group. The fixed cost for an unchosen part is less than fixed cost for a chosen part. The value of T_{si} for this group for both set of solutions are:

For solution set 1 (optimal solution) we have:

$T^*_{21} = 200 + (1)(2)(148) = 496$

$T^*_{22} = 0 + (1.476)(3)(148) = 655.2$

For solution set 2 we have:

$T^1_{21} = 200 + (1)(4)(89) = 556$

$T^1_{22} = 0 + (1.476)(5)(89) = 658.8$

We now change the value of fixed cost for part 1 from 200 to 301. Consider the changes for both sets of solutions. For solution set 1 we have:

$$T^{*}_{21} = 301 + (1)\ (2)\ (148) = 597$$

$$T^{*}_{22} = 0 \quad + (1.476)\ (3)\ (148) = 655.34$$

For the solution set 2 we have:

$$T^{1}_{21} = 301 + (1)\ (4)\ (89) = 657$$

$$T^{1}_{22} = 0 + (1.476)(5)(89) = 656.82$$

As we can see by changing the value of F_{21}, the same part is still chosen in the optimal solution set, but is now changed to the second part in solution set 2. Changes in the value of F_{21} for a chosen part (part number 1 in group number 2) will increase the value of objective function for the optimal solution set, while there will be no change in the value of objective function for the solution set 2. If the value of F_{21} is increased by 149 additional units solution set 2 will become optimal.

One has to be careful in this analysis since the second best solution may not become the optimal solution. It depends on the situation of the chosen parts and their T values. In some cases the third best solution might become optimal before the second set. We will present an example of this case shortly.

From the above discussion the following results can be deduced. These results can be extended to C by substituting δF by δC and F by C, where δC = change in the value of $C_{ki} X_{ki} \sum_{j=1}^{J} d_j Y_j$ as C_{ki} changes.

when the value of fixed cost for a chosen part in any group is increased, then there are three possibilities:

a) If the same part is chosen in every solution set, then the optimal solution will not change if the amount of increase is less than the smallest difference between the two smallest values of "T" in group k over all possible solution sets.

Let i = the selected part in all solution sets, and

$\delta T = \min\ [\min_j (T_{kj} - T_{ki})]$, over all solution sets.

δF = increase in a fixed cost .

Then as long as $\delta F \le \delta T$, design will not change.

b) If the same part is not chosen in any other solution sets, then as long as the local optimality condition holds, that is the chosen part in optimal set does not change, and

$$\delta F \le Z^1 - Z^*,$$

(where Z^* = optimal value of the objective function and Z^1 = the second best value of the objective function), the optimal solution will not change.

c) If in some solution sets the same part and in others different parts are chosen, then as long as

$$\delta F \le \min\ \{\delta T,\ (Z^1 - Z^*)\}$$

the optimal solution will not change. This is concluded from the results of parts a and b.

Proof a: since i appears in all solution sets, any increase in F_{ki} will increase the objective value of all sets the same amount until a chosen part in some sets changes. The limit for changing F_{ki} and having the same part in all solution sets in the optimal solution is δT. As soon as δF becomes greater than δT, different parts in some solution sets will be chosen and therefore the optimal solution might change. The above case is described in the following example.

Example 2: consider the same problem as in example 1 and assume that k=1. We look at three solution sets: For solution set 1 (optimal), $Y^*=(2,1,2,2)$ and $Z^* = 5176$, we have:

$T^*_{11} = 900 + (0.5)(1)(148) = 974$

$T^*_{12} = 200 + (2)(1)(148) = 496$ chosen part.

For the second best solution set, $Y^1=(1,1,1,1)$ and $Z^1=5224.5$ we have:

$T^1_{11} = 900 + (0.5)(2)(89) = 989$

$T^1_{12} = 200 + (2)(2)(89) = 556$ chosen part.

For the third best solution set with $Y^2=(2,2,3,2)$ and $Z^2=5233.5$ we have:

$T^2_{11} = 900 + (0.5)(1)(193) = 996.5$

$T^2_{12} = 200 + (2)(1)(193) = 586$ chosen part.

As we can see from the above calculations i = 2.

Assume there are only these 3 sets of solutions, we have $\delta T = 410.5$. Now as long as $\delta F \le 410.5$, the same amount will be added to all objective values, because the same part is still chosen. But as soon as δF becomes greater than 410.5, the chosen part will change in solution set 3. If we add 469 extra units to the fixed cost of part 2, we will have

For solution set 1:

$T^*_{11} = 900 + (0.5)(1)(148) = 974$

$T^*_{12} = 669 + (2)(1)(148) = 965$ chosen part.

For solution set 2:

$T^1_{11} = 900 + (0.5)(2)(89) = 989$ new part

$T^1_{12} = 669 + (2)(2)(89) = 1025.$

For solution set 3:

$T^2_{11} = 900 + (0.5)(1)(193) = 996.5$ new part

$T^2_{12} = 669 + (2)(1)(193) = 1059.$

As we can see, in solution set 2 and 3, the chosen part is changed and the third best solution has become optimal.

Proof b: Assume i is the chosen part. Since it does not appear in any other solution sets, any increase in the value of its fixed cost will only increase Z^* but has no effect on the objective of other solution sets. This increase continues until either Z^* hits the closest objective value, Z^1. At this point the optimal set will possibly change since it is not economical to use part i.

4.2 Group Related Parameter and Product Related Parameter

As it was mentioned before, R_{sj} is a group-end item related parameter, therefore any change in the value of R_{sj} for one group- end item will affect all parts in that group.

If R_{sj} increases small so that X and Y remain feasible, then they are optimal and the module design stays unchanged. This is true since the value of objective function for currently optimal solution stays the same but for other solutions this value may increase.

If all the values of R_{nj} are changed proportionally, because of the nonlinearity of function T_{ni} and the maximum integer part ($\max_j$ $[R_{sj}/\omega_{si}Y_j]$) in the part selection formula, no prediction can be given and the problem may have to be resolved. If the value of R_{nj} for one group-end item pair is changed, then the chosen part and consequently the quantity of that part will probably change and as a result, the design can change very easily.

d_j is a product related parameter and any change in the demand of any product will have effect on all parts, since all products are using the same module. If all demands for all parts are changed proportionally, then this sensitivity analysis is the same as the part parameter, C, sensitivity analysis discussed earlier in this chapter. Any other change in the value of d_j might change the optimal solution and the problem has to be resolved.

In general analytical results for the sensitivity of Shaftel's algorithm are difficult to obtain. Therefore, it is difficult to obtain bounds for changing R_{sj} or d_j.

4.3 Simultaneous Changes

Besides the cases mentioned above, there are some cases in which two parameters change simultaneously. These are

1. If the value of ω_{si} for a part is proportionally increased and the value of variable cost (C_{si}) for that part is also increased at the same rate, then the value of T_{si} will not change and therefore, the design will stay the same.
2. If the values of ω_{si} for all parts in one group are increased proportionally, and at the same time the values of R_{sj} for the same group are increased at the same rate, then the value of T_{si} for parts in that group will not change and thus design will stay the same (since the ratio of R_{sj}/ω_{si} remains constant).
3. If demands for all end items are increased proportionally and at the same time the variable costs for all parts in all groups are decreased at the same rate, then the value of T_{si} for all parts

will not change, and as a result the same Y's and X's will be optimal.

CHAPTER 5

CONCLUSION

In this thesis, we discuss the multiple choice modular design problem and Goldberg's heuristic to solve the model. Goldberg's heuristic uses shaftel's algorithm, for solving the integer version of modular design problem, as a subroutine. Goldberg's heuristic is tested in both computational time and ability to find the optimal solution. Heuristic gives an optimal solution in more than 75% of the cases tested and is able to solve problems as large as Shaftel's algorithm.

Sensitivity analysis on different parameters is discussed. The structure of the objective function makes this task very difficult. However, in some special cases analytical results can be obtained.

Table 4

Part Group End-Item Requirement

END ITEMS

PART GROUP	1	2	3	4
1	8	2	12	4
2	17	7	0	19
3	8	5	10	12
4	7	6	11	2
5	12	13	18	5
6	14	1	13	9

Table 5

Group Cost And Strength Data

GROUP	PART	F	C	ω
1	1	900	0.5	9.0
	2	200	2.0	6.0
2	1	200	1.0	6.0
	2	0	1.476	4.5
3	1	900	0.4	4.5
	2	300	1.0	7.5
4	1	800	1.5	4.5
	2	1100	1.5	4.5
5	1	1500	1.8	7.5
	2	700	1.0	10.5
6	1	1500	1.2	6.0
	2	1200	1.0	4.5

REFERENCES

Evans, D.E., 1963, "Modular Design - A Special Case in Nonlinear Programming", Operations Research, volume 11, pp. 637-647.

Goldberg, J.B., 1984, "The Modular Design Problem with Linear Separable Side Constraints: Heuristic and Applications". P.H.D Dessertation, University of Michigan.

Goldberg, J.B., 1986, "Applications and Extensions of the Modular Design Problem". Working Paper #86-023, University of Arizona.

Rutenberg, D.P., and Shaftel, T.L., 1971, "Product Design: Subassemblies for Multiple Markets", Management Science, Volume 18, pp. B220-B231.

Shaftel, T.L., 1971, " An Integer Approach to Modular Design", Operations Research, Volume 19, pp 130-134.

Shaftel, T.L. and Tompson, G.L., "A Simplex-Like Algorithm for the Continuous Modular Design Problem", Operations Research, Volume 25, pp 788-805.