Technical Report

विद्याविनियोगाद्विकासः

IIM
AHMEDABAD

INDIAN INSTITUTE OF MANAGEMENT
AHMEDABAD

AN INTERACTIVE PROCEDURE FOR SUBSET SELECTION WITH ORDINAL PREFERENCES

by

R.K. Sarin

T.R.No. 93
Nov. 1975

Indian Institute of Management
Ahmedabad

No. 3

To

Chairman (Research)
IIMA

Technical Report

: An Interactive Procedure for Subset Selection with Ordinal Preferences

Name of the author : R.K. Sarin

Under which area do you like to be classified? Prod'n and Quantitative Methods

ABSTRACT (within 250 words:

Many decision situations involve the selection of a subset of alternatives (items, projects or actions) when the pay-offs of these decisions are difficult to quantify. An interactive procedure is developed to identify an optimal subset of alternatives when the decision-decision-maker can supply only ordinal preferences over the subsets. The prodedure enumerates the subsets of alternatives such that the numer of subset comparisons made explicitly by the decision-maker are minimized.

Please indicate restrictions if any that the author wishes to place upon this note.

Date: 20 oct. '75

R.K. Sarin
Signature of the author

No. 3

To

Chairman (Research)
IIMA

Technical Report

Title of the report : An Interactive Procedure for Subset Selection with Ordinal Preferences

Name of the author : R.K. Sarin

Under which area do you like to be classified? Prod'n and Quantitative Methods

ABSTRACT (within 250 words:

Many decision situations involve the selection of a subset of alternatives (items, projects or actions) when the pay-offs of these decisions are difficult to quantify. An interactive procedure is developed to identify an optimal subset of alternatives when the decision decision-maker can supply only ordinal preferences over the subsets. The prodedure enumerates the subsets of alternatives such that the numer of subset comparisons made explicitly by the decision-maker are minimized.

Please indicate restrictions if any that the author wishes to place upon this note.

Date: 20 oct. '75

R.K. Sarin
Signature of the author

Acknowledgments

I am very much indebted to Professor James Dyer and Professor Donald Erlenkotter for many helpful suggestions. This paper has been adopted from a part of my doctoral dissertation at the University of California, Los Angeles.

AN INTERACTIVE PROCEDURE FOR SUBSET SELECTION WITH ORDINAL PREFERENCES

In this paper an interactive procedure is developed for choosing a subset of alternatives (items, projects, or actions) from N available alternatives. It is assumed that the pay-offs for the alternatives cannot be expressed in numerical units. However, the decision-maker is willing to provide ordinal expressions of preference over the subsets. The procedure developed here attempts to minimize the number of subset comparisons made explicitly by the decision-maker. In section 1 an introduction and problem statement is given. Some basic definitions and theorems are stated in section 2. The interactive procedure is described in section 3. and a flow chart is given. In section 4 an example is presented to illustrate the procedure. The example has been devised to demonstrate all the features of the proposed procedure. In Appendix, a binary representation of the procedure is given that facilitates implementation on a computer.

1. Introduction and Problem Statement

The problem we consider is one in which a decision-maker has to choose a subset of alternatives (items, projects or actions) from N available alternatives. The chosen subset should also satisfy certain linear constraints (resource, manpower, etc.). If the pay-offs for the alternatives can be expressed in numerical units, the subset selection problem can be modeled as a linear or nonlinear integer program. In many decision situations, however, the pay-offs for the alternatives are difficult to quantify. In such situations, the decision-maker must express his preferences over the subsets of alternatives to identify a most preferred subset. Here we assume that he is willing to provide only ordinal expressions of preference over the subsets. Even these ordinal comparisons may require considerable thought by the decision-maker.

The subset selection problem can be represented as follows:

$y_1 \gtrsim y_2 \gtrsim y_3 \quad \cdots \gtrsim y_N$, select the alternatives satisfy the constraints:

$$(1) \qquad Ay \leqq b,$$

$$y_j = 0 \text{ or } 1 \text{ for } j = 1, \ldots, N.$$

Here A is an m by N coefficient matrix, b is an m by 1 resource vector, and y is an N by 1 vector whose j^{th} component is 1 if the alternative y_j is in the subset and 0 if the alternative y_j is not in the subset. $\gtrsim$ means "is preferred or indifferent to" and represents the decision-maker's preference indifference judgment.

There are 2^N possible y vectors (subsets), some of which may not satisfy the constraints in (1) and hence are infeasible. In a complete enumeration scheme the decision-maker has to make $F(F-1)/2$ paired comparisons between the subsets, where F is the number of feasible subsets. In most non-trivial decision situations the number of feasible subsets is large, and hence it is not practical for the decision-maker to make all possible paired comparisons between the subsets.

The task of evaluating the subsets of alternatives is greatly facilitated if certain assumptions are made about the decision-maker's preferences over the subsets. These assumptions allow the comparison between many subsets without directly involving the decision maker.

Those assumptions are:

(i) Given any two subsets y^1 and y^2, $y^1 \gtrsim y^2$ or $y^2 \gtrsim y^1$;

(ii) Given any three subsets y^1, y^2, and y^3, if $y^1 \gtrsim y^2$ and $y^2 \gtrsim y^3$, then $y^1 \gtrsim y^3$;

(iii) Given any four subsets y^1, y^2, y^3, and y^4 such that $y^1 \cap y^3 = \emptyset$ and $y^2 \cap y^4 = \emptyset$, if $y^1 \gtrsim y^2$ and $y^2 \gtrsim y^3$, then $y^1 \cup y^3 \gtrsim y^2 \cup y^4$.

The statements (i) and (ii) correspond to assumptions of connectivit and transitivity, respectively, on the decision-maker's preferences. According to assumption (iii), if project 1 is preferred to project 2 and project 3 is preferred to project 4, then the decision-maker woulc prefer projects 1 and 3 taken together to projects 2 and 4. This would be the case if the decision-maker's preference function defined on the set of alternatives could be written in the additive form (see Fishburn (8)), but the additivity assumption is only sufficient, not necessary, for our purposes.

Unfortunately, even under the above assumptions, the decision-maker may have to make a large number of paired comparisons for identifying a preferred subset. The number of such noncomparable subsets increases exponentially as N increases (see Bartee (1)). Table 1 gives the number of noncomparable paired subsets for various values of N.

Bartee (1) proposes the examination of all feasible subsets. Thus, all noncomparable subsets must be identified. These non-comparable subsets are presented to the decision-maker, who is asked to identify a preferred subset. However, as noted above, the evaluation of all non-comparable subsets would place a considerable information burden on the decision-maker.

Table 1

Number of Alternatives	Number of Sets in Power Set	Total Number of Paired Comparisons	Paired Subsets Not Comparable	
			Number	%
2	4	6	0	0
3	8	28	1	3.6
4	16	120	10	8.3
5	32	496	66	13.3
6	64	2016	364	18.1
7	128	8128	1821	22.4

Reference: Bartee, E.M., "Problem Solving with Ordinal Measurement," Management Science, Vol. 17, No. 10, June 1971.

We propose an interactive procedure for identifying a preferred subset of alternatives. The procedure is interactive in the sense that it progresses by seeking certain information from the decision-maker, and can be implemented on an on-line interactive computer. The central idea of the procedure is to minimize the number of subset comparisons made explicitly by the decision-maker. An implicit enumeration scheme is devised such that the decision-maker steps in only when the choice between two subsets of alternatives cannot be resolved with available information. The information sought from the decision-maker is then utilized efficiently to enumerate as many subsets of alternatives as possible until more information from the decision-maker is absolutely essential for further progress.

A procedure for the evaluation of subsets of alternatives has many important applications. In Research and Development project selection, various scoring models have been used for the optimal selection of a subset of projects (Dean (5), Gordon (10), Pearson (12), Sobin <u>et.al.</u> (13)). However, the project scores offer only an ordinal measure, and cannot measure "how much" one project is better than another. In many situations involving the selection of multi-attributed alternatives (projects), the attributes (objectives) and the outcomes of the alternatives on various attributes cannot be easily specified. In the context of criminal justice projects Emrich (7) classifies such projects as nonevaluable. Bartee (1) discusses in some detail the usefulness of an ordinal approach such as proposed in this paper for these decision situations. The subset selection problem has also been studied when the elements are product defects (Stillson (14)), objectives (Churchman <u>et.al.</u> (2, 3)) and criteria (Eckenrode (6)). It has arisen in the contexts of marketing (Green <u>et.al</u> (11)) field selection by graduate students (Coombs (4)) and psychology (Tversky (15, 16)).

2. Notation, Definitions and Theorems

Suppose that a preference ordering has been obtained on alternatives. Without loss of generality, we can now label the alternatives according to their rank. We thus have an ordered set S whose i^{th} element is the ith ranked alternative. For N available alternatives, the set S is $(1, 2, \ldots, i, \ldots, N)$.

Definition 1: Partial sequence: A partial sequence is an ordered set of numbers such that a higher number never precedes a lower one, and the same number is never repeated. Numbers in a partial sequence correspond to alternatives in S which are undertaken. The number of elements in the partial sequence S^k is given by $n(S^k)$. The j^{th} element from the left in the partial sequence S^k is denoted s_j^k. A partial sequence is feasible if it satisfies the constraints in (1).

Theorem 1: A partial sequence S^1 is unambiguously not inferior to another partial sequence S^2 if:

1. $n(S^1) \geqq n(S^2)$, and
2. $s_j^2 \geqq s_j^1$ for $j = 1$ to $n(S^2)$; that is, each element from the left in S^2 is not smaller than the corresponding element in S^1.

Proof: Conditions 1 and 2 imply that for every alternative in S^2 there is a corresponding alternative in S^1 with the same rank or a higher one. Thus, by invoking assumption (iii) in section 1. Theorem 1 is proved.

For example, if $S^1 = \langle 1, 3, 5 \rangle$ and $S^2 = \langle 2, 3, 6 \rangle$, then both conditions 1 and 2 are satisfied, and hence S^1 is <u>not inferior</u> to S^2. But, if $S^2 = \langle 2, 3, 4 \rangle$, then $s_3^2 \geq s_3^1$, and hence we cannot conclude that S^1 is <u>not inferior</u> to S^2.

The above result will be used extensively in our procedure. The comparison between two partial sequences using Theorem 1 will be termed element by element comparison (ebec).

<u>Definition 2</u>: Branch $i(B_i)$: Branch i is the set of all possible partial sequences of alternatives that include alternative i but do not include any alternative $j < i$. Branch i will contain exactly 2^{N-i} partial sequences. The maximum number of elements that any feasible partial sequence in B_i can possibly have is denoted n_i.

<u>Theorem 2</u>: The enumeration of branches i = 1 to N implies that all possible partial sequences have been enumerated.

<u>Proof</u>: $B_i \cap B_j = \emptyset$, $i \neq j$; that is, no partial sequence is repeated in two branches. The total number of partial sequences in $B_i = 2^{N-i}$. Therefore, the total number of partial sequences in all branches =

$$\sum_{i=1}^{N} 2^{N-i} = 2^N - 1,$$ which is exactly equal to the number of all possible partial sequences excluding $\langle \emptyset \rangle$, the null sequence. This proves Theorem 2.

<u>Definition 3</u>: Level ℓ: Level ℓ of a branch i (B_i) is a set of all possible partial sequences of alternatives such that each partial sequence contains exactly $n_i - \ell + 1$ elements. There will be n_i levels in B_i. The first level of B_i will contain partial sequences with exactly n_i elements, and the n_ith level will contain a single partial sequence with one element (the i^{th} alternative from S). Thus, the

set B_i is further partitioned into n_i subsets. The number of partial sequences in a level ℓ of the branch i is $\binom{N-i}{n_i - \ell}$.

Theorem 3: The enumeration of all levels of branch i implies that all feasible partial sequences in branch i have been enumerated.

Proof: $B_i^{\ell} \cap B_i^{m}$ (for $\ell \neq m$) $= \phi$. The number of partial sequences in all levels of B_i = $\sum_{\ell=1}^{n_i} \binom{N-i}{n_i - \ell}$, which is exactly equal to the number of partial sequences in B_i when no partial sequence contains more than n_i elements. Hence, Theorem 3 is proved.

Definition 4: Fathoming of Branch (Level): A branch (level) is called fathomed when either a non-inferior feasible partial sequence in the branch (level) has been identified or it is asserted that the branch (level) cannot contain a feasible partial sequence which is non-inferior to the incumbent (the incumbent is the best feasible partial sequence obtained so far).

Definition 5: Automatic Fathoming: A branch (level) is declared automatically fathomed when the branch (level) can be fathomed without any additional information, except the ordinal ranking of alternatives, from the decision-maker.

In the proposed procedure, the branch, level and sequence are generated in a particular order. Below we describe this order.

Branch Generation: Branches are generated for enumeration in increasing order, i.e., B_1, B_2, ..., B .

Level Generation: Levels in a given branch are also generated in increasing order. In a branch, i, first level 1 (B_i^1) consisting of all partial sequences of n_i alternatives is enumerated, and then level 2 (B_i^2), and so on till the last level consisting of only the i^{th} alternative is enumerated.

Sequence Generation: In a branch i, the first partial sequence of a level with x alternatives, i.e., the $(n_i - x + 1)^{th}$ level, will have elements i, i+1,..., i+x-1. The first partial sequence consists of x elements; left most element is i, and each next element is incremented by 1. For example, in branch 3, in a level with four projects, the first partial sequence will be <3, 4, 5, 6>. To generate further partial sequences in a level, right most element is incremented by 1 till the list of alternatives in S is exhausted. For example, the above partial sequence <3, 4, 5, 6> will be changed by incrementing 6 by 1 till last alternative N^{th} is considered (<3, 4, 5, 6>, <3, 4, 5, 7>, . . . , <3, 4, 5, N>). When the right most element reaches N, the partial sequence is initialized by incrementing the next left element (initialized partial sequence <3,4,6,7>) and the procedure is repeated (<3,4,6,7>, <3,4,6,8 >, ..., <3,4,6, N>). This is done until it is not possible to construct a valid partial sequence of x elements by incrementing the next left element (the left to right most element takes value N-1). Then, the next left element is picked and incremented by one, and the partial sequence initialized (<3,5,6,7>), and the procedure is repeated until again a valid sequence cannot be generated. The procedure stops when next right element to the first element takes a value (N-x+2). With this procedure, all possible partial sequences of a level are generated. Since no two partial sequences are identical, the procedure is non-redundant.

<u>Theorem 4:</u> If all levels of a branch are fathomed then the branch is fathomed.

<u>Proof:</u> Definition 4 and Theorem 3 imply Theorem 4.

<u>Theorem 5:</u> If the best feasible partial sequence (non-inferior feasible partial sequence) of a level is preferred to the first partial sequence of the next higher level, then all higher levels are fathomed. Hence, if incumbent, S*, is not inferior to the first partial sequence of the first level of a branch, then the branch is fathomed.

<u>Proof:</u> There is a strict preference ordering in the first partial sequences of levels.

$$S_1^1 \gtrsim S_2^1 \gtrsim \cdots \gtrsim S_\ell^1$$

where S_ℓ^1 = First partial sequence of the level ℓ .

Also, by the sequence generation procedure, the first partial sequence of a level is non-inferior to all other partial sequences in that level. By definition 4, all higher levels are fathomed. Similarly if S* is non-inferior to the first partial sequence of the first level than all levels in the branch are fathomed. By Theorem 4 the branch is fathomed.

<u>Theorem 6:</u> If for a given level in a branch i the partial sequence $<i, j, k, l>$ is found to be the first feasible partial sequence then, it is also the best feasible partial sequence in the level if the following holds:

(2) $\quad <i,j,k,l> \gtrsim <i,j,k+1,k+2>$, and

(3) $\quad <i,j,k,l> \gtrsim <i,j+1,j+2,j+3>$

Proof: <i,j,k,l> is the first feasible partial sequence, and hence all previous partial sequences were infeasible. Thus we need to examine only subsequent partial sequences. Recall from the sequence generation procedure that the subsequent partial sequences are generated by first incrementing l, and then incrementing k and initializing and repeating the procedure, and finally incrementing j and initializing and repeating the procedure.

Therefore, $\langle i,j,k,l\rangle \succsim (i,j,k,l')$ for all $l' > l$.
If (2) holds then

$\langle i,j,k,l\rangle \succsim (i,j,k',k'')$ for all $k' > k+1$ and $k'' > k+2$

If (3) holds then

$\langle i,j,k,l\rangle \succsim (i,j',j'',j''')$ for all $j' > j+1$, $j'' > j+2$ and $j''' > j+3$

Thus all partial sequences of the level are exhausted, and hence i,j,k,l is the best feasible partial sequence in the level.

It should be noted that the comparisons in (2) and (3) would sometimes be automatic, while at other times the decision-maker may be asked to make a choice. As before, for automatic comparison, element by element comparison (ebec) as in Theorem 1 should hold.

Example: Suppose S = (1,2,....,7), and $n_1 = 4$; then the number of partial sequences in B_1^1 is 20. These partial sequences are generated in the order given below:

<1,2,3,4>	<1,2,3,5>	<1,2,3,6>	<1,2,3,7>
<1,2,4,5>	<1,2,4,6>	<1,2,4,7>	
<1,2,5,6>	<1,2,5,7>		
<1,2,6,7>			
<1,3,4,5>	<1,3,4,6>	<1,3,4,7>	
<1,3,5,6>	<1,3,5,7>		

$\langle 1,3,6,7 \rangle$

$\langle 1,4,5,6 \rangle$ $\langle 1,4,5,7 \rangle$

$\langle 1,4,6,7 \rangle$

$\langle 1,5,6,7 \rangle$

Suppose $\langle 1,2,3,6 \rangle$ is found to be feasible. Obviously, it is preferred to $\langle 1,2,3,7 \rangle$, so we ask the decision-maker to compare $\langle 1,2,3,6 \rangle$ with $\langle 1,2,4,5 \rangle$ (equivalent to (2)).

If $\langle 1,2,3,6 \rangle \succsim \langle 1,2,4,5 \rangle$ then it is non-inferior to all subsequent partial sequences which have elements 1 and 2 (by sequence generation procedure). Now compare $\langle 1,2,3,6 \rangle$ with $\langle 1,3,4,5 \rangle$ (equivalent to (3)). If $\langle 1,2,3,6 \rangle \succsim \langle 1,3,4,5 \rangle$, then it is non-inferior to all subsequent partial sequences. Thus, $\langle 1,2,3,6 \rangle$ is the best feasible sequence in B_1^1 Note that if (2) holds; that is, $\langle 1,2,3,6 \rangle \succsim \langle 1,2,4,5 \rangle$, then, (3) holds automatically, since $\langle 1,2,4,5 \rangle \succsim \langle 1,3,4,5 \rangle$ by element by element comparison (ebe) as in Theorem 1.

3. Interactive Procedure

Using the results of section 2., we shall now describe the interactive procedure for optimal subset selection.

Step 1 Obtain the ordinal ranking of alternatives to construct the set S.

Step 2 Obtain an upper bound n_i on the maximum number of feasible alternatives for each branch $i=1$ to N. If n_{ij} = maximum number of alternatives in branch i which satisfy constraint j, $n_i = \text{Min}(n_{i1}, n_{i2}, \ldots, n_{im})$. In order to obtain n_{ij} fix the variable corresponding to the i^{th} ranked alternative in the j^{th} constraint to 1. Then successively fix the variables corresponding to lower than the i^{th} ranked alternative, starting with the smallest coefficient, to 1 until the j^{th} constraint is violated. The number of alternatives fixed at 1 before the j^{th} constraint is violated gives n_{ij}.

To obtain tighter bounds, the following 0-1 integer program needs to be solved for all i=1 to N.

$$\text{Max} \sum_{j=1}^{N} y_j$$

$$\text{s.t.} \quad Ay \leqq b$$
$$y_i = 1$$
$$y_j = 0 \text{ for } j < i$$
$$y_j = 0 \text{ or } 1 \text{ for } j \geq i$$

Then, $n_i = \sum_{j=1}^{N} y_j$. It should be noted that it may be more desirable to work with the looser upper bounds since the computational effort in solving the 0-1 integer programs is considerable.

Step 3 In step 3, partial sequences are generated, and an optimal solution is identified. In the enumeration of partial sequences, sometimes the intervention of the decision-maker is required. The flow chart of the procedure is given in Figure 1. Below, we describe each of the four substeps of the procedure.

Step 3.1 Before evaluating a branch i, we check whether the branch can be fathomed. Let S* be the incumbent solution (initially $S^* = \langle \emptyset \rangle$). If S* is non-inferior to the first partial sequence of branch i containing n_i elements, $S^1 = \langle i, i+1, \ldots, i+n_i - 1 \rangle$, then branch i is fathomed. This can be established by using Theorem 1. Otherwise, the decision-maker compares S* and S^1. If he prefers S* to S^1 then branch i is fathomed. If these tests fail, branch i needs to be evaluated.

Step 3.2 In a manner similar to step 3.1, it is determined whether a level can be fathomed without any evaluation. The first partial sequence of a level ℓ of branch i is $\langle i, i+1, \ldots, i+n_i - \ell \rangle$

(say S^1). If the level cannot be automatically fathomed, the decision-maker is asked to compare S* with S^1. If he prefers S* to S^1, then level and all higher levels are fathomed (Theorem 5). Otherwise, level needs to be evaluated.

Step 3.3 In order to determine a feasible partial sequence, the partial sequences are generated according to the sequence generation procedure described earlier, and, checked for feasibility. Generation of all possible sequences in a level and testing them for feasibility requires considerable computational burden. Fortunately, many of these partial sequences can be implicitly enumerated. In problem statement (1), the constraint set is represented by $Ay \leq b$, $y_j = 0$ or 1. Denote, a_{jk} as an element of matrix A. For generating partial sequences in a level containing n elements, an implicit enumeration scheme is as follows.

The first partial sequence of a level with n elements in branch i is $\langle i, i+1, \ldots, i+n-1 \rangle$. Fix the first two elements of this partial sequence (i, i+1) at 1 and test for feasibility by successively fixing at 1 the remaining n-2 elements, starting from the one with the lowest coefficient. This can be accomplished by rearranging the constraints in increasing order of a_{jk}'s for $k > i+1$, so that a_{jt} is the t^{th} ranked coefficient for constraint j. We check whether

$$\sum_{t=1}^{n-2} a_{jt} > b_j - a_{ji} - a_{j,\, i+1}$$

holds for any j=1 to m. If the inequality holds for any one of the m constraints, then no feasible solution in the level with i and i+1 as the first two elements can exist. In this case, increment the second element of the partial sequence by one and fix it at 1, and repeat the process to check for inequality until the inequality does not hold. Note that in a partial sequence with n elements, the second element can be incremented only up to N-n+2. When the first two elements of a feasible partial sequence are determined, these are fixed at 1 and the third element is incremented and the process is repeated. Similarly, the next higher element is considered until the n-1 elements are fixed, and for the n^{th} element the inequality does not hold. The partial sequence thus obtained is the first feasible partial sequence. All partial sequences prior to this partial sequence are infeasible.

Various other ideas can be used in this implicit enumeration. More powerful tests can be devised for particular constraint sets. For example, in many situations it may be possible to introduce a surrogate constraint that is redundant for the problem (1), but is powerful for testing feasibility implicitly (e.g. see Glover [9]).

Once a feasible partial sequence is obtained, we determine whether the level can be fathomed. The element immediately to the left of the right most element of this feasible partial sequence is incremented by one, and the initialized partial sequence is compared with the feasible partial sequence. If the feasible partial sequence is found to be non-inferior, then it is compared with the partial sequence generated by incrementing the next left element by one and initializing. If the feasible partial sequence is non-inferior to all n-2 partial sequence so generated, then the level is fathomed (Theorem 6.6). Otherwise, starting from the initialized partial sequence to which the feasible partial sequence is not non-inferior, further partial sequences are generated (step 3.3).

When a feasible partial sequence is again found, it is compared with the previous feasible partial sequence and one of the two is dropped. But, now the survivor is compared with the initialized feasible sequences generated from the second feasible partial sequence. The process is repeated until the level is fathomed, and the best feasible partial sequence in the level is obtained. This partial sequence is compared with the incumbent and the incumbent is updated. As shown in the flow diagram (Figure 1), the next level or next branch (if all levels are enumerated) is considered.

Proof of Convergence

We have to show that the generation of partial sequences is non-redundant, and that the procedure terminates only when all 2^N solutions have been enumerated.

Non-redundant

We know that any two branches cannot have a common partial sequence (Definition 2). We also know that any two levels in a branch cannot have a common partial sequence (Definition 3). Within a level the same partial sequence is not generated twice (sequence generation). If a feasible solution is obtained in a level and it is fathomed, then the next level is considered. Therefore partial sequences are non-redundant. But if the level cannot be fathomed, then the partial sequences are generated again starting from a higher partial sequence (some element of the feasible partial sequence is incremented and the partial sequence is initialized) and hence the partial sequence cannot be repeated. Thus, the procedure is non-redundant.

Complete 2^N Enumeration

In steps 3.2, 3.3, and 3.4 of our procedure all levels of a branch are evaluated. By Theorem 3 all feasible partial sequences in branch i are thus evaluated. Infeasible partial sequences are implicitly enumerated. In step 3.1, all branches i=1 to N are evaluated and hence by Theorem 1, all 2^N-1 partial sequences are enumerated. But, we start with the <Ø> partial sequence, and hence all 2^N partial sequences are enumerated.

4. Numerical Example

In this section we solve a numerical example using the proposed procedure. The example has been devised to demonstrate all of the features of the procedure.

Example Choose a subset of projects satisfying the constraints:

$$20y_1 + 20y_2 + 10y_3 + 30y_4 + 10y_5 + 10y_6 + 10y_7 + 10y_8 \leq 60$$
$$50y_1 + 50y_2 + 30y_3 + 20y_4 + 30y_5 + 10y_6 + 20y_7 + 10y_8 \leq 150$$
$$y_1 + y_5 = 1$$
$$y_6 - y_7 = 0$$

y_j = 0 or 1 for all j=1 to 8.

Step 1 Suppose, y_1 is most preferred, y_2 second most preferred and so on until y_8 is least preferred.
S = (1,2,...,8)

Step 2* $n_1 = 5$, $n_2 = 5$, $n_3 = 5$, $n_4 = 4$, $n_5 = 4$, $n_6 = 0$, $n_7 = 0$, $n_8 = 0$

Step 3 Branch 1
Level 1 First partial sequence <1,2,3,4,5>

Step 3.3
Is there a feasible partial sequence with 1,2 — No
Is there a feasible partial sequence with 1,3 — Yes
Is there a feasible partial sequence with 1,3,4 — No
Is there a feasible partial sequence with 1,3,5 — No
Is there a feasible partial sequence with 1,3,6 — Yes

*For example, to obtain n_3, fix y_3, y_5, y_6, y_7 and y_8 at 1; it is noted that the first constraint is violated if y_4 is also fixed at 1. Thus, only five variables including y_3 can be fixed at 1 without violating any constraint, hence $n_3 = 5$.

Is there a feasible partial sequence with 1,3,6,7 — Yes
Is there a feasible partial sequence with 1,3,6,7,8 — Yes

Step 3.4 — Compare <1,3,6,7,8> with <1,4,5,6,7>.
Suppose decision-maker prefers <1,3,6,7,8>, then level is fathomed S* = <1,3,6,7,8>.

Level 2

Step 3.2 — Compare <1,3,6,7,8> with <1,2,3,4>.
Suppose decision-maker prefers <1,2,3,4>, then level 2 cannot be fathomed.

Step 3.3
Is there a feasible solution with 1,2 — Yes
Is there a feasible solution with 1,2,3 — Yes
Is there a feasible solution with 1,2,3,4 — No
Is there a feasible solution with 1,2,3,5 — No
Is there a feasible solution with 1,2,3,6 — No
Is there a feasible solution with 1,2,3,7 — No
Is there a feasible solution with 1,2,3,8 — Yes

Step 3.4 — Compare <1,2,3,8> with <1,2,4,5>.
Suppose decision-maker prefers <1,2,4,5>, then generate further partial sequences starting from <1,2,4,5>.

Step 3.3
Is there a feasible solution with 1,2,4 — No
Is there a feasible solution with 1,2,5 — No
Is there a feasible solution with 1,2,6 — Yes
Is there a feasible solution with 1,2,6,7 — Yes

Step 3.4 — Compare <1,2,6,7> with <1,2,3,8> (comparison of two feasible solutions).
Suppose decision-maker prefers <1,2,3,8>, then, to fathom level 2.

Compare <1,2,3,8> with <1,2,7,8>.

<1,2,3,8> is non-inferior to <1,2,7,8> (element by element comparable as in Theorem 1).

Compare <1,2,3,8> with <1,3,4,5>.

Suppose decision-maker prefers <1,2,3,8> , then level 2 is fathomed.

To determine incumbant S* compare <1,3,6,7,8> with <1,2,3,8> .

Suppose decision-maker prefers <1,2,3,8> , then S* = <1,2,3,8> .

Level 3

p 3.2 Compare <1,2,3,8> with <1,2,3> .

<1,2,3,8> is preferred (ebec)

Level 3 and subsequent levels are fathomed (Theorem 5).

Branch 2

Step 3.1 Compare <1,2,3,8> with <2,3,4,5,6> .

Suppose decision-maker prefers <2,3,4,5,6>, then, branch 2 cannot be fathomed.

Step 3.3

Is there a feasible solution with 2,3	–	Yes
Is there a feasible solution with 2,3,4	–	No
Is there a feasible solution with 2,3,5	–	Yes
Is there a feasible solution with 2,3,5,6	–	Yes
Is there a feasible solution with 2,3,5,6,7	–	Yes

Step 3.4 Compare <2,3,5,6,7> with <2,3,5,7,8> (element by element comparable)

Compare <2,3,5,6,7> with <2,3,6,7,8> (element by element comparable)

Compare <2,3,5,6,7> with <2,4,5,6,7> (element by èlement comparable)

Level 1 is fathomed.

To update S^* compare $\langle 1,2,3,8 \rangle$ with $\langle 2,3,5,6,7 \rangle$.

Suppose decision-maker prefers $\langle 2,3,5,6,7 \rangle$, then, $S^* = \langle 2,3,5,6,7 \rangle$.

<u>Level 2</u>

<u>Step 3.2</u> Compare $\langle 2,3,5,6,7 \rangle$ with $\langle 2,3,4,5 \rangle$

Suppose decision-maker prefers $\langle 2,3,5,6,7 \rangle$, then level and subsequent levels are fathomed.

<u>Branch 3</u>

<u>Step 3.1</u> Compare $\langle 2,3,5,6,7 \rangle$ with $\langle 3,4,5,6,7 \rangle$

Branch 3 is fathomed automatically.

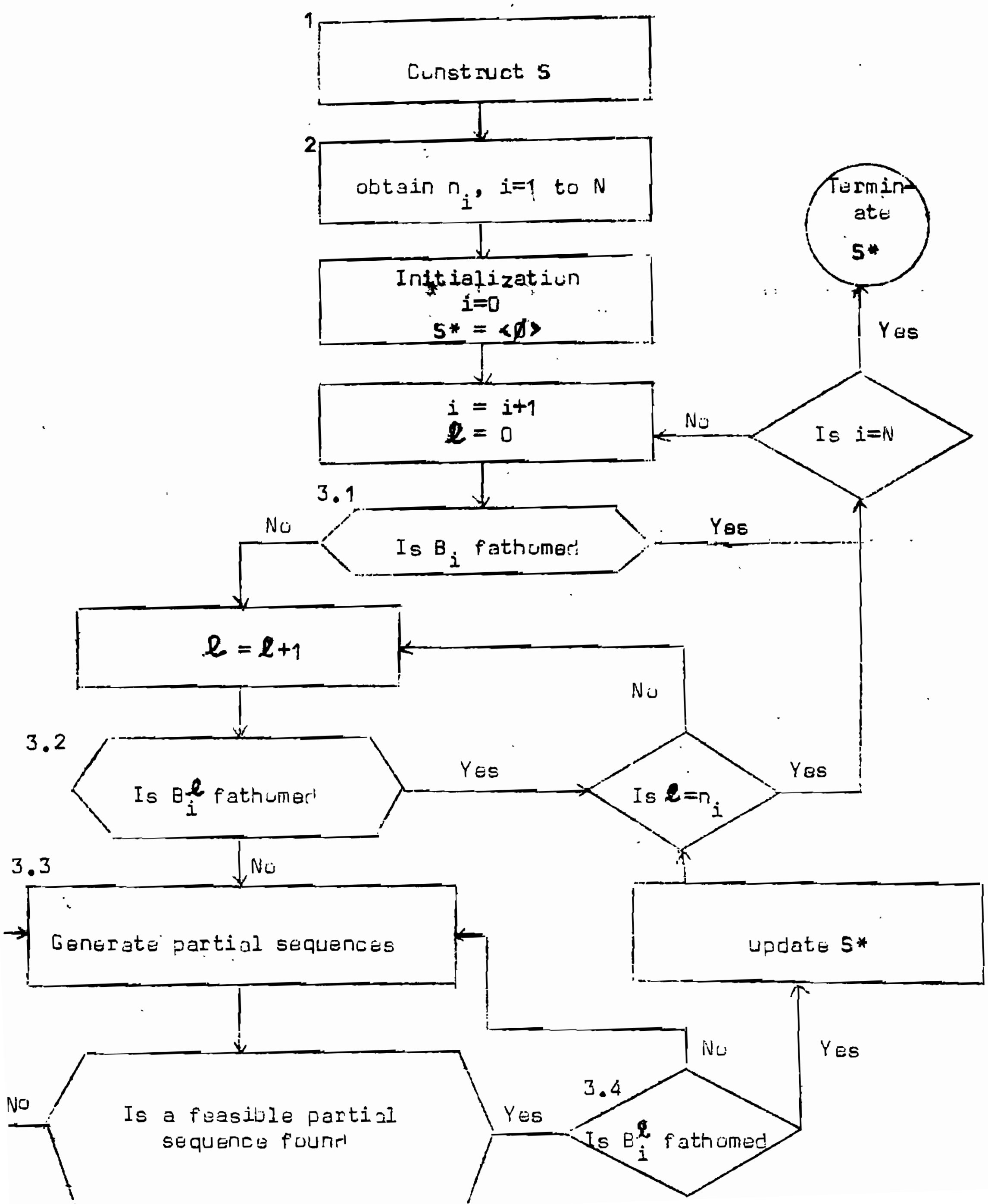
Branches 4,5,6,7 and 8 are also automatically fathomed.

Optimal Solution $= \langle 2,3,5,6,7 \rangle$.

The procedure identified a preferred subset of projects by seeking paired comparisons (eight in the above example) between subsets from th decision-maker. The decision-maker's revealed preferences can be checked for consistency by verifying the assumptions (ii) and (iii) of section 1. Also, at termination his choices (paired comparisons) are to be shown to him to verify whether he maintains the same preferences. In case the decision-maker chooses to revise his previous choices, the procedure should be repeated. The imolementation of this procedure on an interactive computer would greatly facilitate its execution. The computer can do the computations and present to the decision-maker the choices when further computations are not possible. The decision-maker then makes a choice and informs the computer. On the basis of this information further computations are carried out. The procedure thus progresses interactively until a prefered subset is identified.

Figure 1

Flowchart of the Algorithm for Subset Selection with Ordinal Preferences



APPENDIX

Here we shall demonstrate how binary arithmetic can be used to program the proposed procedure on a computer. The motivation for this is in the simplicity of programming while working with binary numbers. However, for clarity and simplicity in conceptual understanding, we did not use binary algebra in the main text.

Consider a vector y whose j^{th} component is 1 if the j^{th} alternative from S is undertaken; otherwise it is 0. y corresponds to a partial sequence.

<u>Testing for a non-inferior partial sequence</u>: y^1 is non-inferior to y^2 if for each 1 in y^2 there is a corresponding 1 in the similar position or to the left of this position in y^1. This is identical to stating that y^1 has a better or similar alternative corresponding to each alternative undertaken in y^2. Note that this comparison cannot be regarded as one binary number being larger than another.

<u>Generation of sequence</u>: The partial sequences are generated by moving the right most 1 to the next right position and initializing and repeating the process. For example, the partial sequences in the first level of the first branch with $n_1 = 3$ and $N = 5$ are as follows:

(1 1 1 0 0), (1 1 0 1 0), (1 1 0 0 1)
(1 0 1 1 0), (1 0 1 0 1)
(1 0 0 1 1)

<u>Testing for fathoming a level</u>: If a feasible partial sequence y^1 is found, then similar to Theorem 6, the level is fathomed if y^1 is non-inferior to partial sequences generated by successively moving the 1 to the next right position and initializing. For example, if (1 1 0 1 1 0 1) is found feasible, then to fathom the level it should be compared with

(1 1 0 1 0 1 1), (1 1 0 0 1 1 1) and (1 0 1 1 1 1 0). The 4 sub-steps of step 3 are as follows:

Step 3.1: A prior evaluation of a branch to determine whether the branch can be fathomed is done by comparing the incumbent with the first partial sequence in the branch. Testing for non-inferior partial sequence has been described above. If branch cannot be automatically fathomed, the decision-maker steps in to make a choice.

Step 3.2: This is similar to step 3.1 above.

Step 3.3: Generation of sequences and testing for feasibility is identical to that described earlier except that in binary representation the elements are moved to right position as described above instead of incrementing by one.

Step 3.4: Testing for fathoming a level has been described above.

REFERENCES

1. Bartee, E. M., "Problem Solving with Ordinal Measurement," Management Science, Vol. 17, No. 10, June 1971.

2. Churchman, C. W., and Ackoff, R. L., "An Approximate Measure of Value," Operations Research, Vol. 2, 172-187, 1954.

3. ______________________, and Arnoff, E. L., Introduction to Operations Research, John Wiley and Sons, New York, 1957

4. Coombs, Clyde H., A Theory of Data, New York; Wiley, 1964.

5. Dean, Burton V., "Scoring and Profitability Models for Evaluating and Selecting Engineering Projects," Operations Research, Vol. 13, No. 4, 550-69, 1965.

6. Eckenrode, R. T., "Weighting Multiple Criterion," Management Science, Vol. 12, 180-192, Nov. 1965.

7. Emrich, Robert, "A Proposed Evaluation Plan for 1973," Unpublished report to the California Council on Criminal Justice, 1973.

8. Fishburn, P. C., "Methods of Estimating Additive Utilities," Management Science, Vol. 13, 435-453, 1967.

9. Glover, F., "A Multiphase-Dual-Algorithm for the Zero-One Integer Programming Problem, Operations Research, 13, 879-919, 1965.

10. Gordon, J. B., "Conversion of Ordinal Values to Relative Values for Research and Exploratory Development Project Proposals," Proceedings of the Sixteenth Military Operations Research Symposium, Office of Naval Research, 1966.

11. Green, Paul E., Wind, Yoram, and Jain, Arun K., "Preference Measurement of Item Collections," Journal of Marketing Research, 9, 371-377, Nov. 1972.

12. Pearson, A. W., "The Use of Ranking Formulae in R & D Projects," R & D Management (U.K.), No. 2, 69-73, Feb. 1972.

13. Sobin, B., and Proschan, A., "Search and Evaluation Methods in Research and Exploratory Development" in H. D. Learner (Ed.), Research Programm Effectiveness, Gordon and Breach; New York,

14. Stillson, P., "A Method for Defect Evaluation," Industrial Quality Control, Vol. 11, 9-12, 1954.

15. Tversky, Amos, "Elimination by Aspects: A Theory of Choice," Psychological Review, 281-299, July 1972.

16. ________, "Choice by Elimination," Journal of Mathematical Psychology, 341-367, Nov. 1972.