Discrete-Variable Extremum Problems
Author(s): George B. Dantzig
Source: *Operations Research*, Vol. 5, No. 2 (Apr., 1957), pp. 266-277
Published by: INFORMS
Stable URL: http://www.jstor.org/stable/167356

DISCRETE-VARIABLE EXTREMUM PROBLEMS

GEORGE B. DANTZIG
The Rand Corporation, Santa Monica, California

This paper reviews some recent successes in the use of linear programming methods for the solution of discrete-variable extremum problems. One example of the use of the multistage approach of dynamic programming for this purpose is also discussed.

A NUMBER of important scheduling problems, such as the assignment of flights for an airline or the arrangement of stations on an assembly line, require the study of an astronomical number of arrangements in order to determine which one is 'best.' The mathematical problem is to find some short-cut way of getting this best assignment without going through all the combinations. By allowing the unknown assignments to vary continuously over some range, mathematicians often arrive at pseudo solutions in which one or more assignments turn out to be fractions instead of whole numbers. It is common practice to adjust such values to whole numbers. Since mathematical models are often imperfect mirrors of reality, this approach is recommended for most practical problems. But since such procedures can occasionally give an answer far from the best, mathematicians have been working on improved techniques. The purpose of the present paper is to review some recent successes using linear programming methods in this difficult area. A thorough discussion of the multistage approach of dynamic programming is beyond the scope of this paper; one example is presented, however, in which this method provides an efficient algorithm.

To be more explicit, there are certain classes of problems that are combinatorial in nature and easy to formulate, but that mathematicians have had only partial success in solving. These arise often in the form of discrete-variable programming problems, such as:

1. The empty-containers problem.
2. The multistage machine-scheduling problem.[1]
3. The flight-scheduling problem.[2]
4. The fixed-charge problem.[3]
5. The traveling-salesman problem.[4]

Examples of problems that have yielded to analysis are the following:

6. The assignment problem.[5,6]
7. The problem of the shortest route in a network.
8. The tanker-scheduling problem.[7]

The purpose of this paper is to outline an approach that, we believe, has a high probability of verifying whether or not, in any particular numerical case, an optimum combination has been selected. The human mind seems to have a remarkable facility for scanning many combinations and arriving at what appears to be either a best one or a very good one. The actual number of combinations can be extremely large, however, making it almost impossible in many cases to verify that the choice is, indeed, a good one. Any ideas, therefore, that can help to verify that a conjectured solution is optimal should be of interest.

The technique presented here is not foolproof and cannot be guaranteed to work in all cases. But it is an approach that several people at Rand have found to work in each numerical case considered. HARRY MARKOWITZ and ALAN MANNE have used it, for example, on flight-scheduling problems, and the present author has employed it with SELMER JOHNSON on various scheduling and container problems and with RAY FULKERSON and SELMER JOHNSON in our article on the traveling-salesman problem.

THE MARRIAGE PROBLEM

TO ILLUSTRATE many of the concepts of this paper, we begin by describing the assignment problem in a popular form: Imagine that a group of hearty pioneer bachelors have gone on ahead of the womenfolk to form a colony. The prospective brides, who have not met the future grooms, now arrive; and after a short courtship in which the boys and girls get acquainted with one another, it is decided to have a ceremony and get things over with. In order to decide who marries whom, the prospective brides rank the various boys in order of preference. Based on their ratings, the selections are to be made. Of course, if each girl were to choose a different boy for top rating there would be no problem; it is only when more than one girl has her eye on one *particular* boy that the selection problem becomes interesting.

From a mathematical point of view, let a variable $x_{ij}=1$ mean that the ith bride (in the solution to the problem) marries the jth boy, and let $x_{ij}=0$ mean that she does not. Then we have certain equations that have to be satisfied:

$$\sum_{j=1}^{j=n} x_{ij}=1, \qquad (i=1,2,\cdots,n),$$

$$\sum_{i=1}^{i=n} x_{ij}=1. \qquad (j=1,2,\cdots n).$$

The equations of the first set state that each bride marries just one boy. Similarly, the equations of the second set state that each boy marries just one girl. Moreover, by explicitly stating that $x_{ij}=0$ or 1, we require a solution that does not admit fractional solutions; in other words, *bigamous* and, generally, *polygamous* solutions are ruled out.

Now let d_{ij} be the rank given by the ith girl to the jth boy.

To determine the 'best' over-all selection, the x_{ij} are to be selected to satisfy the foregoing equations and, subject to these constraints, are to minimize the form

$$\sum_{i,j} d_{ij}\, x_{ij} = z.$$

The rating factor d_{ij}, better known as the 'disappointment' factor, measures the degree of disappointment if the ith bride happens to get the jth boy. Hence, as here defined, a best selection is one that minimizes the total disappointment.

THE STANDARD DISCRETE FORM

To GENERALIZE, the problems we are considering belong to the class of extremizing problems of linear programming in which, for given constants a_{ij}, b_i, and c_j ($i=1, 2, \cdots, m; j=1, 2, \cdots, n$), values $x_1, x_2, \cdots, x_n$ are to be determined, subject to the constraints

$$\sum_{j=1}^{j=n} a_{ij}x_j = b_i, \quad (x_j \geq 0; i=1, 2, \cdots, m) \tag{1}$$

in such a way as to minimize the linear form

$$\sum_{j=1}^{j=n} c_j x_j = z; \tag{2}$$

but for us the variables x_j are further constrained to satisfy

$$x_j = 0 \text{ or } 1. \tag{3}$$

For a programming problem to be discrete, it is not necessary that the variables be 0 or 1. In flight-scheduling problems, for example, the variables that represent the number of flights are required to be nonnegative *integers*. There exists, however, a very simple device by which such problems can be reduced to the 0-or-1 form if the variables have known upper bounds. Indeed, let x be a variable that can take on only nonnegative integral values and let the integer k be an upper bound for x, so that $x \leq k$; then x can be replaced by the sum

$$y_1 + y_2 + \cdots y_k. \qquad (y_j = 0 \text{ or } 1)$$

For this reason the representation (1), (2), and (3) of a discrete programming problem shall be referred to as the *standard discrete form.*

An important property of any set of points whose coordinates satisfy equation (3) is that the points form the vertices of a convex polyhedral set in the n-dimensional space ($x_1, x_2, \cdots, x_n$). This is perhaps intuitively obvious since a point such as $(0, 1, 0, \cdots, 0, \cdots, 1)$ is one of the vertices of the unit n cube (which of course is convex). It is well known that in linear programming problems an optimal solution exists that is an extreme point of feasible solutions. This suggests that in seeking a solution to the standard discrete problem we first weaken the hypothesis as follows:

$$\text{replace} \quad \underset{\text{Discontinuous range}}{x_j=0 \text{ or } x_j=1} \quad \text{by} \quad \underset{\text{Continuous range}}{0 \leq x_j \leq 1.} \tag{4}$$

Since the replacement given in (4) is less restrictive than the condition (3), it follows that the set of solutions to the linear programming problem (1) and (4) forms a convex polyhedral set C that contains the convex polyhedral set C^* whose vertices are the solutions of (1) and (3). It is easy to see, however, that every extreme point (vertex) of C^* is an extreme point of C (see open dots in Fig. 1); but there may be extreme points of C (see

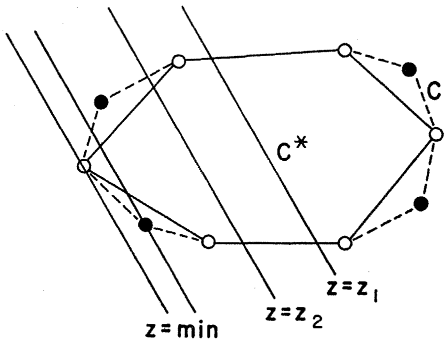

Fig. 1. Schematic representation (2-dimensional case) of the convex polyhedral set C^* whose vertices are the solutions of a discrete linear-programming problem, and of the convex polyhedral set C of solutions of the corresponding continuous problem.

closed dots in Fig. 1) that are outside of C^*. The parallel lines in Fig. 1 represent different positions of the hyperplane

$$c_1x_1+c_2x_2+\cdots c_nx_n=z=\text{constant},$$

and it is clear that, depending on the values of the c_j, an extreme point corresponding to $z=$minimum may belong to C^* (as in Fig. 1) or may belong to C and not to C^*.

A remarkable property of the 'assignment' problem (and the same holds true for the 'shortest-route' problem, which we shall describe in a moment, and of the tanker-scheduling problem) is that

$$C^*=C. \tag{5}$$

Indeed, this result holds true for a general class of 'transportation' problems of which these are special cases. Thus, in the marriage problem, when we replace the condition $x_{ij}=0$ or 1 by $0\leq x_{ij}\leq 1$, we are, in effect, allowing the class of possible solutions to be extended from the monogamous to the polygamous situation in which sharing of mates is possible. The fact that $C^*=C$ states that monogamy will turn out to be the best of all relations in the sense that it will minimize over-all 'disappointment' as defined above.†

THE SHORTEST-ROUTE PROBLEM

Now a second example that has this fundamental property—namely, that the extreme points of the continuous version of the programming problem are the same as those of the discrete form—is the 'shortest-route problem.' Let us suppose there is a package originating in Los Angeles that is to be delivered to Boston along one of the different routes shown in Fig. 2, where the lengths d_{ij} of various links of the transportation network are as shown. We are interested in having the package delivered over the shortest route. The package is allowed to be shipped from city to city

† At the summer, 1955, meeting of ORSA at Los Angeles where this paper was first presented, I (the author) was interviewed by the press, and the following event took place. The reporter turned out to be the brother of my little girl's piano teacher, and so we became quite chummy. I explained to him that linear programming *models* originated in the Air Force and I described the uses of linear programming in industry. It became obvious that this veteran *Hollywood* reporter was having a hard time seeing how to make the material into exciting news copy. In desperation I suggested, "How about something with sex appeal?" "Now you're talking," he said. "Well," I continued, "an interesting by-product of our work with linear programming models is a *proof*—perhaps the first mathematical proof in history—that of all the forms of marriage (monogamy, bigamy, polygamy), *monogamy* is the best of all possible relations." "You say monogamy is the best of all possible relations?" he queried. "Yes," I replied. "Man," he said, shaking his head in the negative, "you've been working with the wrong kind of *models*."

Aside from using an anecdote to interpret fractional solutions, a serious reader may wonder whether the above may be construed as a "proof" of the superiority of one of the accepted mores of Western civilization. Polygamy, it should be pointed out, usually means several wives for one man. Thus, contrary to our assumptions, the sum of the variables associated with a man could exceed unity while that of a woman could be zero or one. Our model is entirely inadequate under this condition for establishing such a proof.

However, in another sense it may have more validity. In our civilization divorce is commonplace. Hence, fractions may be interpreted as the proportion of a person's life spent with one mate or another. Here the only essential assumption is that the rating factors depend on the pairings, and the total is the weighted sum of the individual ratings. This contains the implicit assumption that the over-all rating is the same whether a person married A first and B later or the other way around. It also assumes that the over-all rating for being married at different times to A, B, and C is the weighted average of the individual ratings for A, B, and C. To be realistic, the model should be extended to include a state of being single, along with its corresponding rating. This is easily done. Under this extension, and under the stated assumption, the general conclusion is that an optimum marriage pattern for a group of n men and n women exists that is free from divorce. Thus there is an assignment, where each individual remains married to just one person (or remains single) throughout his adult life, that is better (or at least not worse) over-all than any assignment involving divorce. This result is quite apart from the adverse effect of divorce on children.

along the lines of the network until it arrives at Boston. For $i \neq j$, let $x_{ij}=1$ mean that the package is shipped directly from city i to city j, and let $x_{ij}=0$ mean it is not. Let x_{ii} be the total quantity shipped into the city i, which we require to be the same as the total quantity shipped out.

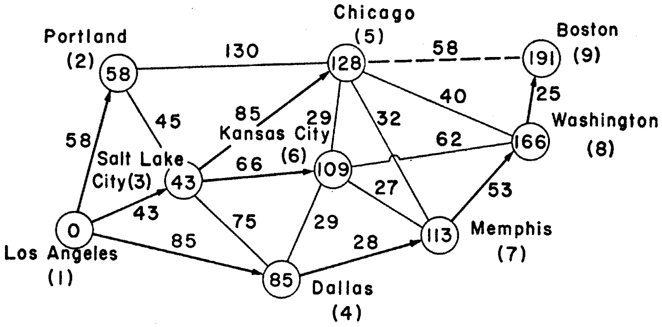

Fig. 2. Illustration of the shortest-route problem.

This situation gives rise to a system of constraints on the variables very similar to the system for the assignment problem; namely, we have

$$\left.\begin{array}{ll}
x_{12}+x_{13}+x_{14} & =1, \\
x_{21}-x_{22}+x_{23} \quad +x_{25} & =0, \\
x_{31}+x_{32}-x_{33}+x_{34}+x_{35}+x_{36} & =0, \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\
\qquad\qquad x_{85}+x_{86}+x_{87}-x_{88}+x_{89} & =0,
\end{array}\right\} \quad (6)$$

$$\left.\begin{array}{ll}
x_{12}-x_{22}+x_{32} \quad +x_{52} & =0, \\
x_{13}+x_{23}-x_{33}+x_{43}+x_{53}+x_{63} & =0, \\
x_{14} \quad +x_{34}-x_{44} \quad +x_{64}+x_{74} & =0, \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \\
\qquad\qquad x_{59} \qquad +x_{89} & =1.
\end{array}\right\} \quad (7)$$

The first equation in (6) states that the amount shipped out of Los Angeles is unity. The last equation in (7) states that the amount shipped into Boston is unity. Equating the x_{ii} in the ith equation of (6) with the x_{ii} in the $(i-1)$st equation of (7) yields the assertion that the amount shipped out equals the amount shipped in for each intermediate city, $i=2, 3, \cdots, n-1$.

Subject to the constraints (6) and (7), and to the further constraint that $x_{ij}=0$ or 1 for $i \neq j$, we are to minimize the form

$$\sum_{i=1}^{i=n} \sum_{j=1}^{j=n} d_{ij}\, x_{ij}=z; \qquad (n=9) \quad (8)$$

that is, we are to minimize the total distance that the package travels.

Thus, the shortest-route problem is very similar to the marriage problem. Now if, in the shortest-route problem, we replace the condition $x_{ij}=0$ or 1 by $0 \leq x_{ij} \leq 1$, it is again true that $C^* = C$, and accordingly this problem also can be solved by standard linear programming techniques. A simple graphical procedure for this will now be described.

First of all, draw a *tree* in the network shown in Fig. 2. By drawing a tree is meant the following: Start from Los Angeles and draw some route conjectured to be optional—for example, a route across the Southern states and then up along the East Coast. Draw arrows indicating this route, then fan out from cities along this route to other cities, drawing more arrows. Repeat until each city in the network is reached either directly or indirectly from Los Angeles. Be sure that the links associated with the arrows *do not form loops.* An example of a tree is shown by the heavy arrows in Fig. 2. Each such tree corresponds to a basis in the simplex method, and the so-called prices (which are analogous to Lagrange multipliers) are computed in the following manner.

At city i put a number u_i which constitutes the distance from Los Angeles if we go along the routes formed by the tree. For example, from Los Angeles to Memphis, city (7), the distance is 85 plus 28; hence $u_7 = 113$ (circled entry at Memphis). Similarly, from Los Angeles to Kansas City along the tree is 43 plus 66, which gives 109, and so forth. The next step is to test whether the tree and route in the tree from Los Angeles to Boston is a solution to the 'shortest-route problem.' To do this, we must determine whether the circled numbers are actually the shortest-route distances from Los Angeles when compared with other routes in the network. If they are, we have an optimal solution. To test whether these are minimum distances for each *directed link* of the network, joining city i to city j, we compare the distance d_{ij} with $u_j - u_i$. If for all directed links we have $u_j - u_i \leq d_{ij}$, then the routes along the tree are the shortest distances from Los Angeles to any other city. If not, the solution can be improved. Consider, for example, the Chicago-Boston link. Here $u_9 - u_5 = 63$ and $d_{5,9} = 58$, so that $u_9 - u_5 > d_{5,9}$ and the tree shown in Fig. 2 is not optimal. Indeed, for a route going along the branches, the total distance to Boston is 191. However, if it went via Chicago, and then to Boston, it would be 128 plus 58, which is 186; this is 5 units less. In this case we could get a better solution by *inserting* an arrow between Chicago and Boston, recording 186 at Boston, and removing the arrow between Washington and Boston. Again, we could test whether the 128 at Chicago could be improved by noting whether a shorter route could be obtained, by considering the Portland–Chicago link. However, here 58 plus 130 is not less than 128, so we try the Kansas City–Chicago link. But here again 109 plus 29 is not less than 128. By continuing in this manner, we can eventually arrive at a situation where it is not possible to find any directed link that leads to an improvement of the distance shown in any circle. If so, we have arrived at an optimal solution. For the example at hand, the optimal tree is the same as that shown in Fig. 2, except that the arrow from Washington to Boston is dropped and one from Chicago to Boston is inserted. The 191 at Boston is changed to 186. The values of the x_{ij} are unity along the path in the final tree from Los Angeles to Boston and are zero elsewhere. Hence the optimal path is from Los Angeles to Salt Lake City, then to Chicago, and finally to Boston.

THE KNAPSACK PROBLEM

In certain types of problems, we can get extreme-point solutions for which not all the values of the x_{ij} are either zero or one. When any of the x_{ij} have fractional values, the corresponding extreme points are referred to as *fractional extreme points*. Now an example of this occurs in the knapsack problem. In this problem a person is planning a hike and has decided not to carry more than 70 lb of different items, such as bed roll, geiger counters (these days), cans of food, etc.

We try to formulate this in mathematical terms. Let a_j be the weight of the jth object and let b_j be its relative value determined by the hiker in comparison with the values of the other objects he would like to have on his trip. Let $x_j=1$ mean that the jth item is selected, and $x_j=0$ mean that it is not selected. We express the weight limitation by

$$\sum_{j=1}^{j=n} a_j x_j \leq 70, \tag{9}$$

with

$$x_j=0 \text{ or } 1, \tag{10}$$

and wish to choose the x_j so that the total value

$$\sum_{j=1}^{j=n} b_j x_j = z \tag{11}$$

is a maximum.

Now we can show this pictorially in the plane (Fig. 3) if one coordinate axis measures weight a and the other measures value b. Each object then is represented by a point having coordinates (a_j, b_j). The problem, graphically, is to select a subset of these points that represents the set of items that he carries with him on his hike; the others he rejects. Let us see what type of graphical solution is obtained if the condition $x_j=0$ or 1 is replaced by the condition that the variables can lie anywhere in the interval from 0 to 1. Then the problem can be solved by regular linear programming methods; indeed, because of its very simple form it admits an immediate solution: Rotate clockwise a ray with the origin as pivot point and b axis as starting position. Items corresponding to points swept out by the ray are selected in turn until the sum of their weights exceeds the weight limitation. If for the jth item the weight limitation would first be exceeded, the value x_j is chosen as that fractional part of its weight a_j that would make the sum come exactly to 70 lb. With the exception of this one item all the items swept out by the ray have the value $x_j = 1$, while those not swept out have the value $x_j = 0$. It will be noted that this is

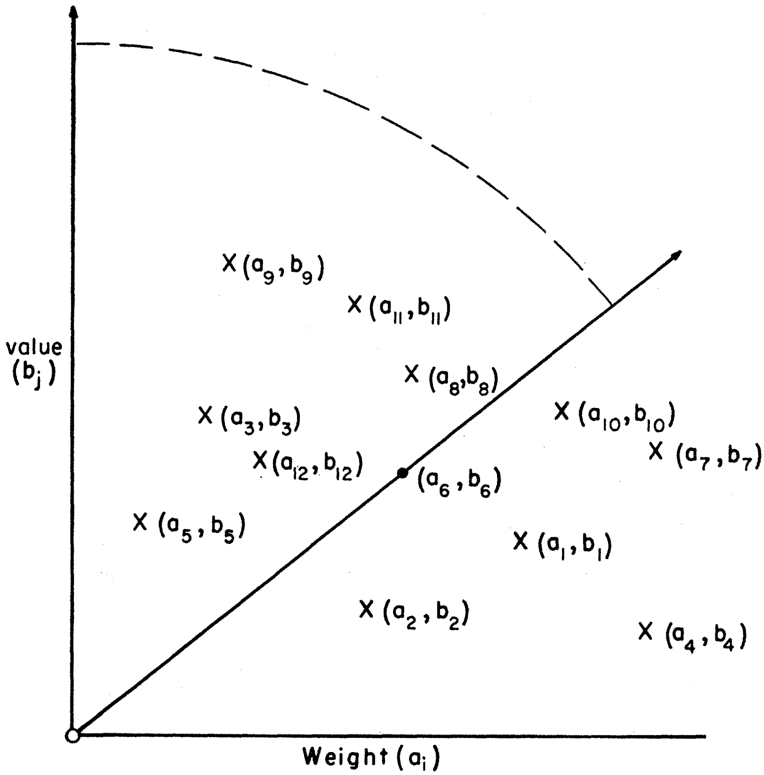

Fig. 3. Graphical solution of the knapsack problem.

very close to the kind of solution desired: All the x_j's are either 0 or 1 with the exception of the one that has a fractional value.

Now at this point the question is natural, "What happens if the solution is *rounded*?" The effect of rounding up or down is of course to change the total weight carried to a number different from 70 lb. If the model is imperfect—in other words, if the hiker does not really mean exactly 70 lb—this may be a satisfactory way of getting rid of the fractional solution; this is particularly true if the weight of individual items is small relative to 70 lb. For most practical problems, the rounding procedure is probably all that is needed. Our object here, however, is to explore ways of getting an exact mathematical solution.

The first of these is due to R. BELLMAN,[8] and is straightforward; it uses the functional-equation approach of dynamic programming. It is recommended where there are only a few items and only one kind of limitation. Extensions to two or more limitations—say one on total weight and another on total volume—can be done, but there would be a considerable increase in the amount of computational work. The method consists in ordering the items in any arbitrary way and determining what items would be carried if (*a*) the weight limitation were $w=1, 2, \cdots$, or 70, and (*b*) the selections were restricted to only the first k items. For example, if $k=1$ and $w<a_1$ (where a_1 is the weight of the first item) then the item would not be selected; but if $w \geq a_1$, it would be. From this it is easy to decide, for every total weight $w=1, 2, \cdots, 70$, what the selections would be for the first two items ($k=2$), and thence inductively for $k=3, 4, \cdots, n$. Indeed, suppose we wish to determine whether we select the $(k+1)$st item if our weight limitation is w when we know how to make the selections for the first k items for any weight $w=1, 2, \cdots, 70$.

Let $F_k(w)$ be the highest total value that can be obtained with the first k objects under weight limitation w. Then, under the same weight limitation w, the highest total value that can be obtained with first $k+1$ objects is $F_k(w)$ if the $(k+1)$st object is not selected, but is $b_{k+1}+f_k(w-a_{k+1})$ if the $(k+1)$st object is selected. Hence, the $(k+1)$st object is or is not selected depending on which of these is higher. Thus not only is the selection for $k+1$ objects determined, but it also is clear that $F_{k+1}(w)$ is given for $w \geq a_{k+1}$ by

$$F_{k+1}(w)=\max[F_k(w),\, b_{k+1}+F_k(w-a_{k+1})] \tag{12}$$

and $F_{k+1}(w)=F_k(w)$ for $w<a_{k+1}$. The procedure is iterated for each w and repeated for $k=1, 2, \cdots, n$.

The linear programming approach consists in putting such additional linear-inequality constraints on the system that the fractional extreme points of C where the total value z is maximized will be excluded, while the set of extreme points of the convex hull C^* of admissible solutions will be unaltered. The procedure would be straightforward except that the rules for generating the *complete* set of the additional constraints is not known. For practical problems, however, rules for generating a partial set of constraints is often sufficient to yield the required solution.

Let us suppose, as in Fig. 3, that the ray swept out items 5, 3, 12, 11, 8, and 9 before the weight limitation was reached, but that when it reached item 6 the weight limitation was exceeded, so that

$$a_5+a_3+a_{12}+a_{11}+a_8+a_9<70,$$

$$a_5+a_3+a_{12}+a_{11}+a_8+a_9+a_6>70.$$

We wish to exclude the fractional extreme-point solution

$$\begin{aligned} x_5=x_3=x_{12}=x_{11}=x_8=x_9=1 \quad &\text{for} \quad 0<x_6<1, \\ x_j=0 \quad &\text{for} \quad \text{all other } j. \end{aligned} \tag{13}$$

It is clear that for an *admissible solution* not *all* $x_j=1$ for the seven points $j=5, 3, 12, 11, 8, 9, 6$, for otherwise the weight limitation would be violated. This means that the sum of these variables cannot exceed 6, or

$$x_5+x_3+x_{12}+x_{11}+x_8+x_9+x_6 \leq 6. \tag{14}$$

Since the fractional extreme-point solution (13) does not satisfy this constraint, it is clear that the effect of adding the particular inequality (14) is to exclude this fractional extreme point. Form (11) is maximized under conditions (9) and $0 \leq x_j \leq 1$, but with the constraint (14) added. Again a new fractional extreme point may turn up for the new convex C, and it will be necessary again to seek a condition that will exclude it. For the most part the conditions added will be other partial sums of the x_j similar to (14). However, at times more subtle relations will be required until at last an extreme point is obtained that is admissible.

Since the discovery of these more subtle relations is more an art than a science, the reader may dismiss the whole approach as worthless. However, experiments with many problems by the author and others indicate that very often a practical problem can be solved using only such obvious supplementary conditions as (14). In an experiment with a number of randomly chosen traveling-salesman problems involving nine cities, simple upper bounds and so-called simple 'loop conditions' on the variables were sufficient to yield the desired discrete solution. This indicates that the probability of need of the more subtle constraints is low in any given problem. To appreciate the power of this procedure it should be noted that for each nine-city case solved the tour that minimized the total distance covered was one from among 362,880 ways of touring nine cities and was selected in about two hours of hand-computation time.

REFERENCES

1. S. M. Johnson, "Optimal Two- and Three-Stage Production Schedules with Setup Times Included," *Naval Res. Log. Quart.* **1**, 61–68 (1954).
2. A. Manne and H. Markowitz, "On the Solution of Discrete Programming Problems," Rand Paper P-711, August 1, 1955.
3. W. Hirsch and G. B. Dantzig, "The Fixed Charge Problem," Rand Research Memorandum RM-1383, December 1, 1954.
4. G. B. Dantzig, D. R. Fulkerson, and S. M. Johnson, "Solution of a Large-Scale Traveling Salesman Problem," *Opns. Res.* **2**, 393–410 (1954).
5. H. W. Kuhn, "The Hungarian Method for the Assignment Problem," *Naval Res. Log. Quart.* **2**, 83–98 (1955).

6. L. R. Ford, Jr., and D. R. Fulkerson, "A Simple Algorithm for Finding Network Flows and an Application to Hitchcock Transportation Problem," Rand Paper P-743, September 26, 1955.
7. G. B. Dantzig and D. R. Fulkerson, "Minimizing the Number of Tankers to Meet a Fixed Schedule," *Naval Res. Log. Quart.* **1**, 217–222 (1954).
8. R. Bellman, "Some Applications of the Theory of Dynamic Programming—A Review," *Opns. Res.* **2**, 275–288 (1954).

DYNAMIC PROGRAMMING AND THE NUMERICAL SOLUTION OF VARIATIONAL PROBLEMS

RICHARD BELLMAN
The Rand Corporation, Santa Monica, California

THE PURPOSE of this paper is to illustrate the application of the theory of dynamic programming to the numerical solution of a large class of variational problems of the type occurring in a variety of applications. Many of these problems either escape in entirety the techniques of the classical calculus of variations, or else can be treated only with the aid of quite detailed and complicated analysis.

Although the methods we shall employ are equally suited to treat the problem of determining the analytic structure of the solution, in this paper we shall focus exclusively upon computational techniques.

Our aim is to present a simple, readily applicable technique, requiring no mathematical background beyond elementary calculus, which can be used to compute the solution of a variety of problems in a routine fashion, with no regard to linear or nonlinear, stochastic or deterministic features of the underlying processes.

REPRESENTATIVE PROBLEMS

LET US begin by describing a number of representative variational problems arising in various branches of economic, industrial, and engineering study, and then point out the difficulties of these problems, as far as conventional treatment is concerned.

A. A Classical Variational Problem

Consider the problem of determining the minimum of the functional

$$J(x)=\int_0^T F(x, dx/dt, t)\, dt \tag{1}$$

over all functions $x(t)$ satisfying the condition that $x(0)=c$.

Let us assume that F satisfies appropriate conditions (which, at the moment, are of no interest to us), so that the minimum is attained for a function satisfying the Euler equation