T-1370

AN APPLICATION OF BINARY INTEGER PROGRAMMING TO THE LOCK-BOX PROBLEM

By

Fred F. Ciochetto Jr.

A Thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science (Mathematics)

Signed: ______________
Student

Golden, Colorado

Date: April 22, 1971

Approved: ______________
Thesis Advisor

____
Head of Department

Golden, Colorado

Date: April 22, 1971

ABSTRACT

One of the problems faced by the financial sector of many businesses that operate over wide geographic areas is that of minimizing "accounts receivable float." "Accounts receivable float" is accounts receivable revenue for which investment opportunity is being lost and is a product of customer remittances which are either within the postal system or in the process of clearing the banks on which they were drawn. One approach to this problem involves the location of lock-boxes (post-office boxes selected so that the time necessary to collect and have available the funds remitted to a firm by it's customers is a minimum) within a firm's distribution area.

The problem of optimally locating lock-boxes may be mathematically modeled as a binary integer problem of nm+m variables and 2n constraints, where n represents the number of customers to a firm and m represents the number of potential lock-box sites being considered. Several methods may be employed in solving this problem; but each quickly becomes unsatisfactory for larger problems, so another method, which uses an exhaustive, implicit-explicit enumeration scheme, is developed.

The efficiency of this method arises from the fact that, by carefully selecting a few solutions to be explicitly enumerated, many more solutions may be implicitly

enumerated and the computation greatly reduced. The method is generated by imposing three conditions on the method of Balas. The first condition constrains the method of Balas so that only feasible solutions are examined. The second and third conditions bound the search from below and from the right so that, whenever "completions" of a partial solution are feasible but less attractive than the current best solution, those "completions" are implicitly enumerated.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

I want to thank the members of my graduate committee, Associate Professor Robert E. D. Woolsey and Associate Professor Ardel J. Boes, for their assistance during the preparation of this dissertation. I am especially indebted for the help and guidance provided by the chairman of my graduate committee, Assistant Professor Hunter S. Swanson.

INTRODUCTION

One of the problems faced by the financial sector of many businesses that operate over wide geographic areas is that of minimizing "accounts receivable float," that is, inaccessible revenue for which investment opportunity is being lost. This "float" is a product of customer remittances which are either within the postal system or in the process of clearing the banks on which they were drawn. One attempt, which was presented by Levy[1], at solving this problem involves the location of lock-boxes within a firm's distribution area. These lock-boxes are post-office boxes selected so that the time necessary to collect and have available the funds remitted to a firm by its customers is a minimum.

To illustrate the function of lock-boxes, consider the following hypothetical situation. Firm A, located in city X, with a distribution area throughout the United States, does business with only one bank which is located in city X, and firm A's terms of sale are 15 days net. On the fifteenth day following the rendering of the invoice for its order, firm B, which is in city Y, remits a check to firm A. Upon receipt and deposit of this check in city X, the check must clear the bank in city Y on which it was drawn. With a clearing time of two days and a mailing time of three days, there is a minimum of 20 days time before the remittance is available to firm A.

Suppose firm A had established a lock-box in city Y to which customer B in city Y remits rather than sends the check to city X. In establishing the new lock-box, firm A selected a bank in city Y, which collects the remittances several times daily and deposits them in firm A's account. With a one-day clearing time and a one-day mailing time, firm A could have its funds available in 17 days. If this saving of three days' availability time could be extended to all of firm A's "accounts receivable," the cost savings associated with investing the funds for three days time at the current rate of interest would become quite significant.

The cost savings illustrated above does not, however, represent the cost of establishing the new lock-box in city Y. This cost usually consists of a fixed charge associated with the post-office box rental, a fixed charge associated with maintaining an account in the proper bank, and a variable charge associated with the volume of transactions processed by the bank for firm A. Therefore, the lock-box selection becomes a problem in seeking a minimum cost to a firm through balancing the cost savings from shortened fund availability against the cost of establishing new lock-boxes.

The purpose of this thesis is to describe an exact procedure for finding the optimal number of lock-boxes and their locations through application of binary integer programming. The thesis begins with a discussion of a formulation of the problem in terms of binary integers and then proceeds to an evaluation of the various methods of solution available. Finally, the development of an exact method of solution is discussed, and the computational results from this method are examined.

MATHEMATICAL FORMULATION

The lock-box selection process may be mathematically modeled as the following binary integer problem:

Minimize

$$
\begin{aligned}
Y = {} & a(1,1)x(1,1) + a(1,2)x(1,2) + \dots + a(1,m)x(1,m) + \\
& a(2,1)x(2,1) + a(2,2)x(2,2) + \dots + a(2,m)x(2,m) + \\
& \vdots \\
& a(n,1)x(n,1) + a(n,2)x(n,2) + \dots + a(n,m)x(n,m) + \\
& b(1)y(1) + b(2)y(2) + \dots + b(m)y(m)
\end{aligned}
$$

subject to

$$
\begin{aligned}
(1)\quad & x(1,1) + x(1,2) + \dots + x(1,m) = 1 \\
& x(2,1) + x(2,2) + \dots + x(2,m) = 1 \\
& \vdots \\
& x(n,1) + x(n,2) + \dots + x(n,m) = 1\ , \\
(2)\quad & x(1,1) + x(2,1) + \dots + x(n,1) - M\,y(1) \le 0 \\
& x(1,2) + x(2,2) + \dots + x(n,2) - M\,y(2) \le 0 \\
& \vdots \\
& x(1,m) + x(2,m) + \dots + x(n,m) - M\,y(m) \le 0\ , \\
(3)\quad & x(I,J) = 0 \text{ or } 1 \quad \text{for } I=1,n \text{ and for } J=1,m\ ,
\end{aligned}
$$

and (4) $y(J) = 0$ or 1 for $J=1,m$;

where $x(I,J) = 1$ if check I is assigned to lock-box J, x(I,J) = 0 otherwise,

y(J) = 1 if any check has been assigned to lock-box J,

y(J) = 0 if no check has been assigned to lock-box J,

a(I,J) = the variable cost of assigning check I to lock-box J,

b(J) = the fixed charge associated with lock-box J,

and M = a large positive integer (999,999).

The problem may be stated more concisely as follows:

Minimize $\sum_{I=1}^{n} \sum_{J=1}^{m} a(I,J)x(I,J) + \sum_{J=1}^{m} b(J)y(J)$

subject to

(1') $\sum_{J=1}^{m} x(I,J) = 1$ for I=1,n ,

(2') $\sum_{I=1}^{n} x(I,J) - M\,y(J) \quad 0$ for J=1,m ,

(3') x(I,J) = 0 or 1 for I=1,n and for J=1,m ,

and (4') y(J) = 0 or 1 for J=1,m ,

where constraint 1' insures that no check is deposited in more than one lock-box, constraint 2' introduces the fixed charge (B(J)) to any lock-box that has been used, and constraints 3' and 4' merely restrict the range of the variables x and y to the integers zero and one.

To solve this problem, several different methods could be used. For example, the Gomory cutting-plane[2] method is readily adaptable to this problem if the variables are properly constrained. A second method[3], which applies a branch-and-bound technique, may also be used if the variables are properly constrained. While both of the above methods guarantee an optimal solution to the problem, they quickly become computationally unattractive for larger problems. A third method, which applies an exhaustive, implicit-explicit enumeration scheme, may also be used to guarantee an optimal solution to the problem. It is this third method, when fully implemented, that yields the best procedure for solving the problem.

In a problem of N variables, there are 2^N-1 solutions to examine under an exhaustive enumeration scheme. A systematic procedure for doing this was presented by Balas[4]. The efficiency of Balas' procedure is derived from the fact that, by carefully selecting a few solutions to be enumerated explicitly, many solutions can be enumerated implicitly and the computation greatly reduced. In particular, sets of solutions are ignored when the enumeration proceeds if

(1) "completions" of partial solutions are found to be infeasible, or

(2) "completions" of partial solutions are found to be feasible but less attractive than the current best solution;

where a "completion" of a partial solution is obtained by specifying values for those variables whose solution values are not specified in the current partial solution.

When applied to an n-by-m lock-box problem, Balas' method must explicitly or implicitly enumerate 2^{nm+m} solutions. Convergence of this method to the optimal solution may be accelerated by the implementation of four basic ideas. The first of these ideas involves checking partial solutions of the Balas procedure for feasibility. If the constraints of the problem are ordered so that the "tightest" constraint is always tested first, an infeasibility in a partial solution may be readily determined, usually without testing all constraints. The "tightness" of a constraint is measured when all constraints in the problem are satisfied either by assigning all variables a value of zero or by assigning all variables a value of one. The least oversatisfied of these constraints is then chosen as the "tightest" constraint. Convergence is also accelerated by choosing as an entering variable for the next partial solution that variable which reduces infeasibility the most. In other words, the second idea specifies that the choice of an entering variable for the next partial solution is based on reduction in infeasibility. The third idea involves the addition of "surrogate" constraints to the problem. Glover(5) showed how the addition of these "surrogate" constraints may accelerate convergence of the Balas method. A few years later, Goeffrion(6) described a procedure to be followed to generate the "strongest surrogate" constraint. The last of these ideas involves starting Balas' procedure at a good feasible solution. The efficiency of this idea arises from the fact that sets of solutions are implicitly enumerated when "completions" of partial solutions are found to be feasible but less attractive than the current best solution. Thus, if Balas' procedure is started at a good feasible starting point, many solutions which were previously explicitly enumerated may be ignored. The starting solution to be used may be obtained through several different methods. One such method, due to Levy[7], uses a heuristic approach to the lock-box problem. Another method[8] which is readily adaptable to the lock-box problem, uses a heuristic approach to the capital budgeting problem.

With the implementation of the aforementioned ideas, Balas' procedure seemed quite suitable for the solution of small lock-box problems. However, as larger problems (5 by 5) were attempted, the time savings associated with the implementation of acceleration methods was greatly overshadowed by the time necessary to enumerate (either implicitly or explicitly) $2^{nm+m}-1$ solutions. This fact led to the search for a faster enumeration process.

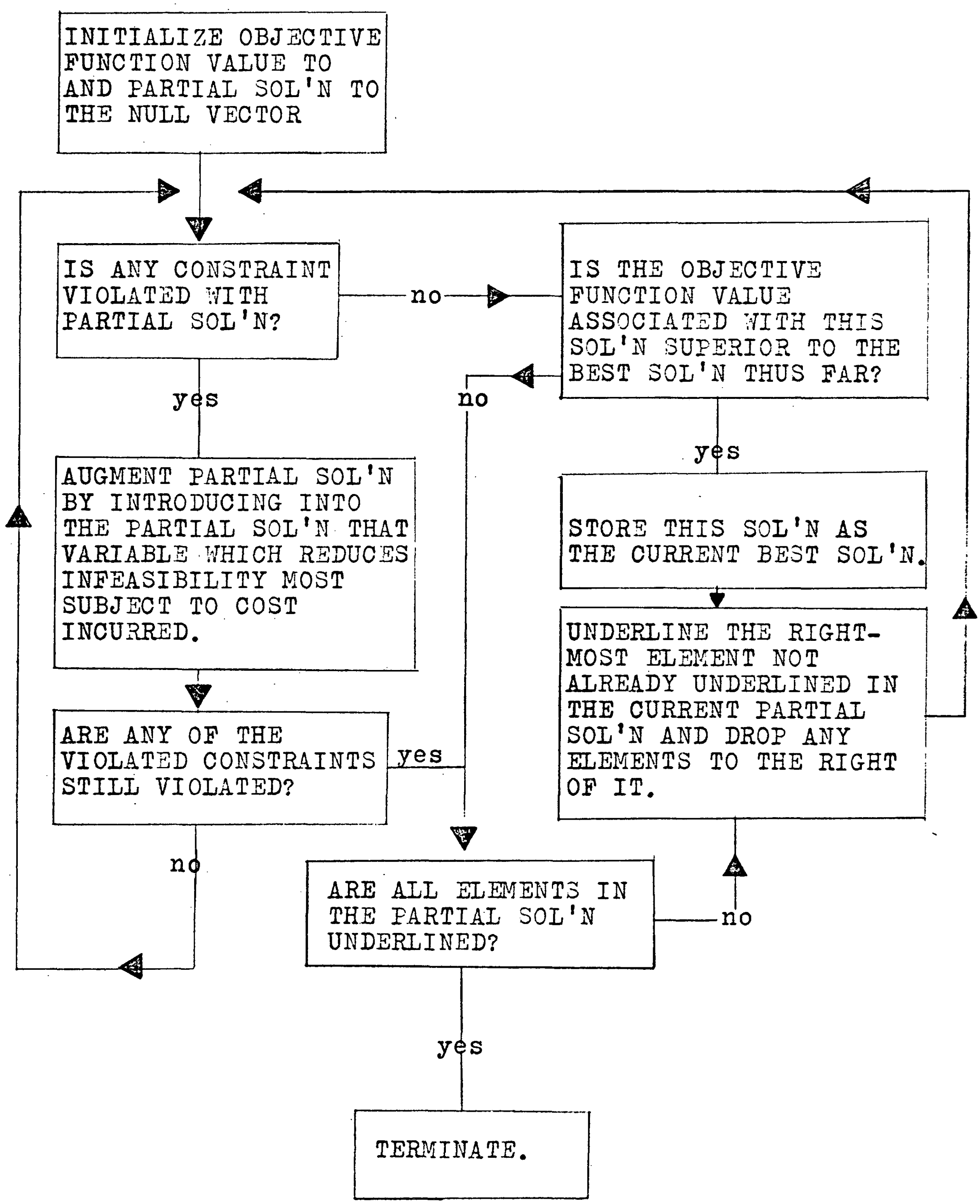
PROBLEM SOLUTION

Notice in the Balas procedure that even with the addition of "tight surrogate" constraints much time is spent in enumeration of infeasible solutions. In particular, if all solutions which assign the same check to multiple lock-boxes or which fail to assign each check to some lock-box may be ignored, then m^n rather than $2^{nm+m}-1$ solutions must be enumerated. For example, if infeasible solutions were ignored, approximately $1/2^{77}$ or 2^{33} of all the possible solutions (2^{110}) would be enumerated in a 10 by 10 lock-box problem. Figure 1 shows a simplified flow diagram for the enumeration of feasible solutions.

Convergence of this method may be further accelerated with the addition of the accessories developed for the Balas procedure. As in Balas' method, this procedure implicitly enumerates sets of solutions when "completions" of partial solutions are found to be feasible but less attractive than the current best solution. To illustrate the effects of implicit enumeration, consider a generalized lock-box problem whose solution sets are partially depicted by the "tree" in figure 2. If this "tree" could be searched for the optimal solution so that a "violation" is produced at stage B, for example, rather than at stage D, then many more solutions could be implicitly enumerated at one time, A "violation" occurs whenever "completions" of a

FIGURE 1

FLOW DIAGRAM FOR LOCK-BOX PROCEDURE



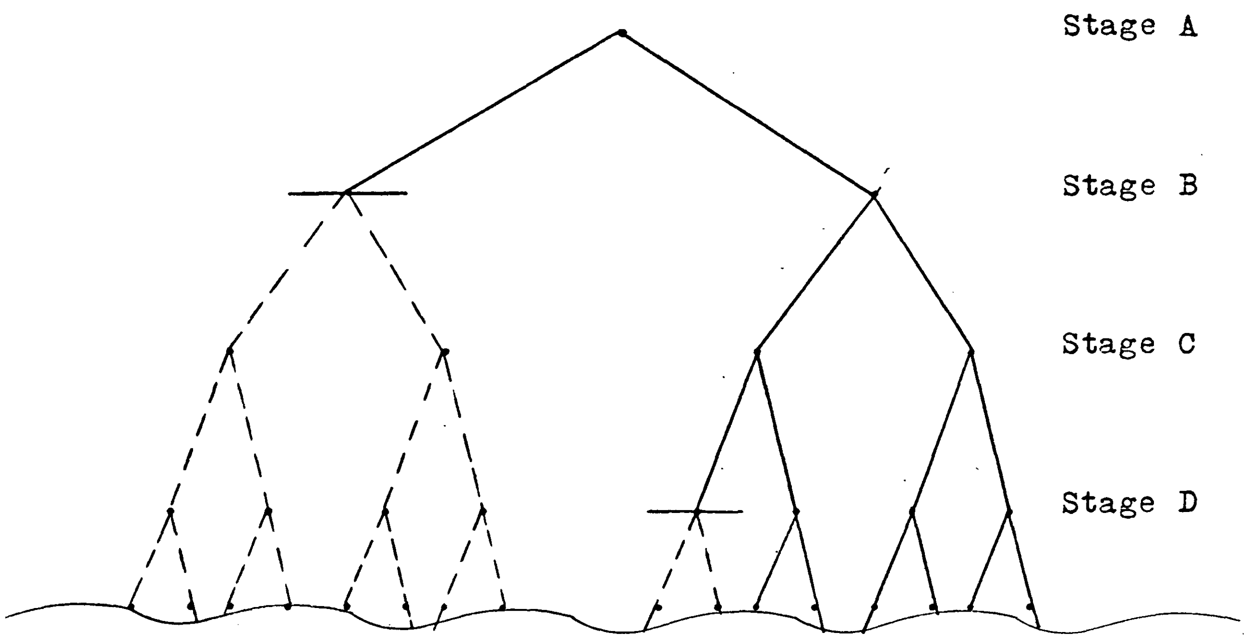


Fig. 2 Effect of raising the bound. Solid lines indicate paths that must be explicitly enumerated while dashed lines indicate paths that may be implicitly enumerated. Horizontal bars indicate position of the bound.

partial solution are less attractive than the current best solution. In particular, the number of solutions which are implicitly enumerated is a direct function of how quickly a "violation" is produced in the search. This fact suggests that the constraints should be reordered so that the most easily violated constraint is always tested first. This ordering scheme, however, proved unsatisfactory for larger problems even though most solutions were implicitly enumerated. For example, a realistic 10 by 10 lock-box problem supplied by the Denver and Rio Grande Western Railroad Company ran for more than 4 hours actual execution time on a PDP 10 computer before execution was interupted prior to completion. However, a second, slightly different approach resulted in an execution time of only 3.7 seconds for the same 10 by 10 lock-box problem. This second method bases the choice of an entering variable to a partial solution on the ratio of reduction in infeasibility to cost. As mentioned above, execution time on the 10 by 10 lock-box problem, using the second approach, was 3.7 seconds, but more importantly, of 10^{10} possible solutions only 3,000 solutions were explicitly enumerated. As larger problems were encountered, the addition of a constraint involving the fixed charges B(J) proved helpful in accelerating convergence. The extent to which an n-by-m lock-box problem may be implicitly enumerated is a function of the range of the A(I,J) values and will be discussed in the final chapter.

COMPUTATIONAL RESULTS

Based on the range of the coefficients in the 10 by 10 lock-box problem, other problems were constructed to test the lock-box procedure. The A(I,J) variable costs were determined as the amount of interest lost to a firm due to the use of funds as compensating balances to the banks. Thus, the variable cost of assigning check I to a lock-box J is derived from the following formula:

A(I,J) = (RPY(I)*DAYS(I,J)*INT/365) + ((CPI(J)*IPY(I))/(((100-RREQ)/100)*ECR(J))) * INT

where

RPY(I) = the total amount of remittances from a customer I in a one-year time period,

DAYS(I,J) = the elapsed time in days before a remittance, which is processed through lock-box J, from customer I is available for use.

INT = the annual rate of interest applicable to compensating balances (may be short-term rate or investment rate depending upon the firm's cash situation),

CPI(J) = the cost, per item processed, assessed by the bank associated with lock-box J,

IPY(I) = the number of remittances per year from customer I, AFC(J) = the annual fixed charge assessed by the bank associated with lock-box J for maintaining an account for the firm,

RREQ = the current reserve requirement,

ECR(J) = the earnings credit rate for the bank associated with lock-box J,

and

APO(J) = the cost incurred through post-office box rental.

The lock-box fixed charge may be similiarly represented by the following formula:

B(J) = (AFC(J)/(ECR(J)*(100-RREQ)/100) * INT + APO(J).

To illustrate the concept of compensating balances, suppose a firm (I) maintains an account in bank J. The variable cost incurred by firm I is represented by CPI(J)*IPY(I). Now, consider funds that are deposited in and for the use of bank J. Of the funds deposited, only (100-RREQ) percent of these funds are available to be invested by bank J. Furthermore, bank J specifies that it is able to earn ECR(J) percent return on its investments so that ECR(J)*(100-RREQ)/100 represents the return on each dollar invested by bank J. Therefore, to insure that bank J receives full compensation for its services, firm I must deposit funds in the amount of (CPI(J)*IPY(I))/ECR(J)*(100-RREQ)/100) for the use of bank J. Thus, actual cost to firm I is represented by interest lost on the funds deposited for the use of bank J.

As mentioned previously, the extent of implicit enumeration in an n-by-m lock-box problem is a function of the range of the A(I,J) values. In particular, the extent of implicit enumeration in any lock-box problem depends upon the range of the variable RPY(I) (the amount of business per year from customer I). To examine the effect of varying the range of this variable, several different size problems were solved. The range of RPY(I) was then halved, and the problems were solved again. This procedure was repeated until the range of RPY(I) was 1/8 of the original range of RPY(I). Computational results of these tests are presented in table 1.

In conclusion, it is the author's opinion that the method just developed has wide application to fixed-charge assignment problems of all sizes. Even though the method guarantees an optimal solution to these problems, the author's experience indicates that this method also provides a very good solution quickly which is an extremely attractive feature when dealing with larger problems.

TABLE 1

COMPUTATIONAL RESULTS

TEST PROBLEM	SIZE	RANGE			
		R	R/2	R/4	R/8
1	10-10	1.57 1	2.23 27	1.52 108	1.35 314
2	10-10	1.27 4	1.18 24	1.35 24	1.48 48
3	20-20	3.92 4	3.87 48	7.58 3249	20.55 21282
4	20-20	4.60 138	5.83 180	6.30 672	----- -----
5	30-30	9.25 48	9.47 576	14.37 3657	30.22 23977
6	30-30	12.15 3284	16.33 5090	401.90 406545	-----

Key to table 1:

no	ss-ss	ttt.tt cccc

where

no represents the test problem number,

ss-ss represents the size of the test problem,

ttt.tt represents the execution time in seconds on a PDP 10 computer,

and

cccc represents the number of solutions which were explicitly enumerated.

LITERATURE CITED

1. Levy, F. K., An application of heuristic problem solving to accounts receivable management: Management Science, v. 12, no. 6, p. 236-244 (1966).

2. Gomory, R. E., An algorithm for the mixed integer problem: Rand Corporation, RM-2597 (1960).

3. Land, A. H. and A. Doig, An automatic method of solving discrete programming problems: Econometrica, v. 28, p. 497-520 (1960).

4. Balas, E., An additive algorithm for solving linear programs with zero-one variables: Operations Research, v. 13, p. 517-546 (1965).

5. Glover, F., A multiphase-dual algorithm for solving the zero-one integer programming problems: Operations Research, v. 13, no. 6, p. 879-919 (1965).

6. Geoffrion, Arthur M., An improved implicit enumeration approach for integer programming: Rand Corporation, RM-5644-PR, June (1968).

7. Levy, F. K., An application of heuristic problem solving to accounts receivable management: Management Science, v. 12, no. 6, p. 236-44 (1966).

8. Senju, S. and Y. Toyoda, An approach to linear programming with zero-one variables: Management Science, v. 15, no. 4, p. 196-207 (1968).