T-1601

SOLUTION OF A RESOURCE ALLOCATION PROBLEM WITH A MODIFICATION OF THE METHOD OF SENJU AND TOYODA

By

Franklin Ernest Grange II

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.

Signed: Franklin Ernest Grange II
Student

Golden, Colorado

Date: November 26, 1973

Approved: ____________________
Thesis Advisor

____
Head of Department

Golden, Colorado

Date: November 26, 1973

ABSTRACT

Discrete allocation of scarce resources among nonmutually exclusive investment alternatives to maximize net present value is a problem common to many firms of the mineral industry. Conventional methods of economic evaluation are often inadequate to select among such investments. This study presents an application of a variation of the integer programming heuristic method of Senju and Toyoda (1) to the analysis and solution of resource allocation problems. A resource economic, nonmathematical interpretation of the method is offered; a brief mathematical treatment of the original heuristic and the modification used is presented; and computational results obtained on a number of published solved problems are presented. Suggestions for further research are also made.

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

ACKNOWLEDGMENTS

I would like to thank the following people for their technical assistance and indulgence in the course of my study: Dr. Jean P. Mather, Dr. Frank J. Stermole, and Dr. R. E. D. Woolsey. All were members of my Committee.

In addition, I would like to thank the Colorado School of Mines and the State of Colorado for financial assistance during my residence as a graduate student at the Colorado School of Mines.

INTRODUCTION

A problem common to many firms of the mineral industry may be stated as follows. Given that any number of a group of projects may be chosen for investment, what projects should be chosen to maximize the total net present value of all investments made within the constraints imposed by the availabilities of several scarce resources? Each project requires some known amount (but possibly none) of each of the several scarce resources, which may include men, money, or materials. If part or all of one or more of a project's resource requirements is not allocated to the project, the project may not be done; that is, the decision to do a project necessarily implies that resources are being allocated to meet all of the project's resource requirements exactly.

As an example of such a situation, consider the problem facing a firm wishing to do many possible petroleum production development projects. After appraisal of external and internal financing and their relative costs, top management determines how much money is available for investment in each of several quarters in the future. District engineering staffs estimate future production volumes from the many investments, and from these data, future costs, future revenues, and, finally, net present values at the firm's minimum rate of return are developed for all projects. Additional estimates are made of the drilling and work-over rig hours required by each project in each month, and the available drilling and work-over rig hours in each month are estimated. Engineering man-hour requirements and availabilities are similarly estimated. Such basic data is obviously very difficult to obtain and highly subjective in nature.

Because of the inherent difficulty of getting some data in the first place, the emphasis of the decision-maker's efforts should be not on the mechanics of making the final decision, but on data collection and evaluation. In a more comprehensive version of the hypothetical problem, an integrated oil company may have 2000 or more projects to consider. Two thousand projects will yield 2^{2000} possible combinations of projects to do or not do; it is physically unreasonable to totally list the combinations and assess their total net present values and feasibilities subject to scarce resource constraints. The sheer magnitude of the problem may thus preclude finding the net present value-maximizing set of projects.

Faced with such a mountainous task, the decision-maker will invariably resort to a seat-of-the-pants approach. Sometimes, in fact, intuitively obvious approaches to this problem will yield the best or reasonably good solutions. Since the benefit of knowing the best (i. e., the net present value-maximizing) solution may not offset the cost of obtaining it, an intuitively comfortable method of finding a relatively good solution may possess both psychological and economic appeal.

On the other hand, the high stakes that are sometimes associated with these resource allocation problems may place a premium on the accuracy of the solution. Several sophisticated mathematical approaches will yield an optimum solution to the resource allocation problem, but for even small problems such methods are sure to try the patience of even the most understanding and best educated manager.

A good project selection method should thus be efficient, accurate, and simple; but these criteria are usually at cross purposes with one another. If cost and simplicity are unimportant, any degree of accuracy may be achieved. In high-stakes situations, the benefit of knowing the right answer may well justify the expected cost of finding it via some complex route. Despite clear economic benefits, however, the best solution may be impossible to implement, for "a manager would rather live with a problem he cannot tolerate than accept a solution he cannot understand." (2) Simplicity and efficiency are the real keys to implementation.

The heuristic method of Senju and Toyoda (1) is not only very simple and efficient, but reasonably accurate as well. The method has been successfully applied to real-world problems (1,3). Because the method of Senju and Toyoda is a heuristic, it may not always generate an optimal solution, but Wyman (4) showed that the expected cost of finding the optimal solution may exceed the expected net present value improvement over the Senju and Toyoda solution.

With one exception (3), literature treatment of the Senju and Toyoda method has been largely mathematical in nature. The "robustness" of the "simple" heuristic has impressed most previous researchers, and they have usually evolved a set of randomly generated benchmark problems to demonstrate their incredulity that something so simple works so well. Unfortunately, other accurate and relatively efficient heuristics for solving resource allocation problems exist, but few are as simple as the robust heuristic of Senju and Toyoda. The importance of the heuristic is not its ability to find a very good solution, but its simplicity and rational operation.

It is the purpose of this study to present a resource economic, nonmathematical rationalization of the Senju and Toyoda heuristic, and to demonstrate the robustness of the method on a number of published solved problems. The study is divided into four parts. First, a nonmathematical, verbal explanation of the heuristic's operation is presented. Second, a numerical example is solved and discussed. Next, a brief mathematical presentation of the original heuristic of Senju and Toyoda and the modification used in this study is given. Finally, computational experience and comments on other authors' approaches are presented.

RATIONAL DEVELOPMENT OF THE HEURISTIC

The rules used for project elimination in the Senju and Toyoda heuristic have a very intuitive appeal. Crucial to the operation of the algorithm is the project ranking based on the computation of the "effective gradient." The effective gradients, in turn, are based on three measures: the net present value of each project, the percentage of each resource availability that each project requires, and the excess percentage demand for each resource that exists at each stage of the algorithm.

Net present value (NPV) is a popular way for a firm to rank mutually exclusive investments, and NPV is an intuitively appealing way to rank nonmutually exclusive investments. Ranking is necessary, however, because limited resources required by the nonmutually exclusive investments constrain the firm from doing all investment projects under consideration. Some projects, furthermore, may have a high net present value but use resources inefficiently; that is, some projects have a lower net present value per present-worth investment dollar, or "bang-for-buck" ratio.
Because certain investments may have a relatively high NPV but use inordinately voluminous amounts of resources, several projects with smaller NPV's but more efficient resource uses might give a larger total NPV for the same or slightly less total resource investments than a single large-NPV project. Ranking based on net present value alone may thus be an

inadequate means of selecting a NPV-maximizing set of projects within the resource constraints; a more intuitively adequate ranking measure would be proportional to NPV and inversely proportional to resource consumption.

Two difficulties arise in constructing such a resource efficiency measure. First, whereas the present worths of investments made in several budgeting periods may be added to give a reasonable result, amounts of different kinds of resources may not be added (just as apples and oranges may not be added) to give a reasonable result. The second difficulty with a resource efficiency measure is that simple addition of the different resource requirements might give a ranking no different from a ranking based on one of the resources alone. An example of this problem might exist in the allocation of tens of thousands of operating dollars and hundreds of exploration geology staff man-hours among a number of nonmutually exclusive oil exploration projects. A resource efficiency ranking based on simple addition of the operating-dollar and the man-hour requirements will probably be identical to a ranking based on the operating dollars alone. Since the available man-hours still constrain project selection, the man-hour requirements should not be ignored in resource efficiency computation.

The percentage each project requires of the total amount of each resource available is thus a more reasonable basis for defining resource efficiency. In the first place, if any project requires more of any resource than is available, the project's percentage resource requirement exceeds 1, and the project must be immediately rejected. Secondly, scaling resource requirements by availabilities will give dimensionless quantities which may be added without compunction. Thirdly, the percentages will all be between 0 and 1, and no resource requirements will dominate the resource efficiency measure only on the basis of order of magnitude differences. A useful resource efficiency measure of a project will still be proportional to NPV but be inversely proportional to the project's percentage resource requirement for each resource.

Typically, some resources are more or less scarce than others, and an intuitively appealing measure of resource efficiency should penalize projects with relatively high requirements of relatively scarce resources. Resource scarcity, of course, is best measured in terms of how much is available opposed to how much is needed. The obvious dimensionless measure of scarcity is the sum of all percentage resource requirements for a resource, minus 1; if the sum is less than 1, however, the resource is in sufficient supply and scarcity is 0. Since a merit ranking should penalize projects that require scarce resources, resource efficiency of a project should vary inversely with great need of scarce resources.

The Senju and Toyoda method uses a merit ranking measure of a project that varies directly with the project's NPV and inversely with the sum of the products of each resource scarcity times the project's percentage requirement for the resource; the merit ranking measure is the "effective gradient" of the project. Thus, each project is ranked on the basis of how well it contributes to total net present value as opposed to how much the project consumes of resources that are more or less scarce. Restated in nonmathematical jargon, the steps of the Senju and Toyoda method are as follows:

STEP 0. Compute the percentage resource requirements of each project for each resource by dividing the resource requirements of each project by the respective resource availabilities.

STEP 1. For each resource, sum the percentage resource requirements over all projects to compute the total percentage demand for each resource. If the sum is less than 1 for any resource, that resource is in sufficient supply, does not constrain project selection, and may be ignored by setting the resource scarcity equal to 0. For each resource whose total percentage demand is greater than 1, subtract 1 from the sum, and make the scarcity equal to the difference. If all resource scarcities are equal to 0, stop; all projects may be done.

STEP 2. For each project, multiply each percentage resource requirement by the respective resource scarcity, and sum the products over all resources. The sum is the project's scarce-resource use.

STEP 3. Compute each project's resource efficiency by dividing the project's NPV by its scarce-resource use.

STEP 4. Reject the project with the smallest resource efficiency. For each resource, if the rejected project's percentage resource requirement is greater than or equal to the old resource scarcity, set the new resource scarcity equal to 0; otherwise, make the new resource scarcity equal to the old scarcity minus the rejected project's percentage requirement for that resource. If all new resource scarcities are 0, go to Step 5; otherwise, go to Step 2.

STEP 5. The current selected project set satisfies all the resources availability constraints, but sufficient amounts of resources may remain unallocated so that some rejected projects may be selected and still satisfy the resource constraints. For each resource, compute the percentage unallocated resource as 1 minus the sum of the percentage resource requirements of all selected projects; if the difference is zero for any resource, no more projects may be selected; otherwise, go to Step 6.

STEP 6. Examine each rejected project in order of decreasing NPV. For any resource, if the percentage resource requirement of the rejected project exceeds the corresponding percentage unallocated resource, the project cannot be selected; otherwise, select the project and for each resource subtract the project's percentage resource requirement from the percentage unallocated resource. Continue this examination of originally rejected projects until every one has been examined for selection.

EXAMPLE PROBLEM

An R & D manager considers the following proposals:

Proposal	Investments		Engineering manpower (man-years)			Net Present Value
	Year 1	Year 2	Year 1	Year 2	Year 3	
A	$50,000	$28,000	7.3	7.2	6.8	$1,070,000
B	27,000	13,000	8.2	8.7	9.3	820,000
C	43,000	29,000	5.4	4.7	3.1	790,000
D	68,000	0	6.2	6.2	5.0	960,000
E	18,000	33,000	2.3	3.2	4.5	500,000
F	29,000	41,000	5.6	4.5	5.0	550,000
G	37,000	13,000	6.3	6.3	6.3	1,000,000
H	47,000	26,000	8.4	9.3	5.2	900,000

What proposals should he elect to do if he has $200,000 and $100,000 to spend in the first and second years, respectively, and he can allocate 35.0 man-years in each of the three years?

STEP 0. Compute the percentage resource requirements.

Proposal	Investments		Engineering manpower (man-years)		
	Year 1	Year 2	Year 1	Year 2	Year 3
A	0.250	0.280	0.209	0.206	0.194
B	0.135	0.130	0.234	0.249	0.266
C	0.215	0.290	0.154	0.134	0.089
D	0.340	0	0.177	0.177	0.143
E	0.090	0.330	0.066	0.091	0.129
F	0.145	0.410	0.160	0.129	0.143
G	0.185	0.130	0.180	0.180	0.180
H	0.235	0.260	0.240	0.266	0.149

STEP 1. Compute the scarcity of each resource.

Investments

Year 1

(0.250+0.135+0.215+0.340+0.090+0.145+0.185+0.235)-1=0.595

Year 2

(0.280+0.130+0.290+0 +0.330+0.410+0.130+0.260)-1=0.830

Engineering manpower

Year 1

(0.209+0.234+0.154+0.177+0.066+0.160+0.180+0.240)-1=0.420

Year 2

(0.206+0.249+0.134+0.177+0.091+0.129+0.180+0.266)-1=0.432

Year 3

(0.094+0.266+0.089+0.143+0.129+0.143+0.180+0.149)-1=0.293

STEP 2. Compute the scarce-resource uses.

Proposal

A (0.250)(0.595)+(0.280)(0.830)+(0.209)(0.420)
+(0.206)(0.432)+(0.194)(0.293)=0.615

B (0.135)(0.595)+(0.130)(0.830)+(0.234)(0.420)
+(0.249)(0.432)+(0.266)(0.293)=0.472

C (0.215)(0.595)+(0.290)(0.830)+(0.154)(0.420)
+(0.134)(0.432)+(0.266)(0.293)=0.517

D (0.340)(0.595)+ 0 +(0.177)(0.420)
+(0.177)(0.432)+(0.143)(0.293)=0.395

E (0.090)(0.595)+(0.330)(0.830)+(0.066)(0.420)
+(0.091)(0.432)+(0.129)(0.293)=0.432

F (0.145)(0.595)+(0.410)(0.830)+(0.160)(0.420)
+(0.129)(0.432)+(0.143)(0.293)=0.591

G (0.185)(0.595)+(0.130)(0.830)+(0.180)(0.420)
+(0.180)(0.432)+(0.180)(0.293)=0.424

H (0.235)(0.595)+(0.260)(0.830)+(0.240)(0.420)
+(0.266)(0.432)+(0.149)(0.293)=0.615

STEP 3. Compute the resource efficiencies.

Proposal	
A	1,070,000÷0.615=1,739,837
B	820,000÷0.472=1,737,288
C	790,000÷0.517=1,528,046
D	960,000÷0.395=2,430,380
E	500,000÷0.432=1,157,407
F	550,000÷0.591= 930,626 ←
G	1,000,000÷0.424=2,358,491
H	900,000÷0.615=1,463,415

STEP 4. Proposal F has the smallest resource efficiency.

Compute the new resource scarcities.

Investment
- Year 1 0.595-0.145=0.450
- Year 2 0.830-0.410=0.420

Engineering manpower
- Year 1 0.420-0.160=0.260
- Year 2 0.432-0.129=0.303
- Year 3 0.293-0.143=0.150

STEP 2. Compute the new scarce-resource uses.

Proposal	
A	0.376
B	0.292
C	0.313
D	0.274
E	0.243
G	0.266
H	0.380

STEP 3. Compute the new resource efficiencies.

Proposal	
A	1,070,000÷0.203=5,270,936
B	820,000÷0.164=5,000,000
C	790,000÷0.164=4,817,073
D	960,000÷0.197=4,873,096
G	1,000,000÷0.155=6,451,613
H	900,000÷0.214=4,205,607 ←

STEP 4. Proposal H has the smallest resource efficiency.

Compute the new resource scarcities.

Investment

- Year 1 0.360-0.235=0.125
- Year 2 0.090-0.260=-0.170

Engineering manpower

- Year 1 0.194-0.240=-0.046
- Year 2 0.212-0.266=-0.054
- Year 3 0.021-0.149=-0.128

All resource scarcities but investment in year 1 are 0.

STEP 2. Compute the new scarce-resource uses.

Proposal	
A	(0.250)(0.125)+0+0+0+0=0.031
B	(0.135)(0.125)+0+0+0+0=0.017
C	(0.215)(0.125)+0+0+0+0=0.027
D	(0.215)(0.125)+0+0+0+0=0.043
G	(0.185)(0.125)+0+0+0+0=0.023

STEP 3. Compute the new resource efficiencies.

Proposal	
A	1,070,000÷0.031=34,516,129
B	820,000÷0.017=48,235,294
C	790,000÷0.027=29,259,259
D	960,000÷0.043=22,325,581 ←
G	1,000,000÷0.023=43,478,261

STEP 4. Proposal D has the smallest resource efficiency.

Compute the new resource scarcity.

Investment in Year 1 0.125-0.340=-0.215

STEP 5. Compute the percentage unallocated resources.

Selected proposals are A, B, C, and G.

Investments

Year 1 1-(0.250+0.135+0.215+0.185)=0.215

Year 2 1-(0.280+0.130+0.290+0.130)=0.170

Engineering manpower

Year 1 1-(0.209+0.234+0.154+0.180)=0.223

Year 2 1-(0.206+0.249+0.134+0.180)=0.231

Year 3 1-(0.194+0.266+0.089+0.180)=0.271

STEP 6. Consider the rejected proposals for selection.

Proposal	Net Present Value	Select?	Resources in short supply for selecting proposal
D	$960,000	No	Investment in year 1
H	900,000	No	Investments in years 1 and 2; manpower in years 1 and 2

Proposal	Net Present Value	Select?	Resources in short supply for selecting proposal
F	550,000	No	Investment in year 2
E	500,000	No	Investment in year 2

The total net present value of the selected proposals (A, B, C, and G) is

1,070,000+820,000+790,000+1,000,000=$3,680,000

Total resource requirements are

Investment

Year 1 50,000+27,000+43,000+37,000=$157,000

Year 2 28,000+13,000+29,000+13,000=$83,000

Engineering manpower

Year 1 7.3+8.2+5.4+6.3=27.2 man-years

Year 2 7.2+8.7+4.7+6.3=26.9 man-years

Year 3 6.8+9.3+3.1+6.3=25.5 man-years

A better solution is possible. By ranking the proposals by net present value and considering what proposals may be added subject to the resource constraints, proposals A, G, D, B, and E may be selected for a total net present value of $4,350,000. If, in place of D, proposal C is the last one dropped in the Example solution, proposal E may be added in Step 6, and the same better solution would have been found. Obviously, the method does not always give the optimum solution, but this example certainly shows how the method provides a systematic procedure for reaching a good feasible solution.

A COMPUTATIONAL MODIFICATION OF THE METHOD OF SENJU AND TOYODA

The heuristic of Senju and Toyoda addresses problems of the form

$$\text{Maximize} \quad \sum_{j=1}^{n} c_j x_j$$

$$\text{subject to} \quad \sum_{j=1}^{n} a_{ij} x_j \leq b_i \qquad i = 1,\ldots,m$$

$$x_j = 0 \text{ or } 1$$

$$c_j,\ a_{ij},\ b_i \geq 0$$

The algorithm starts with an infeasible solution with every variable in the basis (i. e., all x_j=1 at the outset.) Variables are chosen to leave the infeasible basis in order of an increasing "effective gradient" until a feasible basis remains. The effective gradient of a variable is a measure of how much the variable contributes to the objective function as opposed to how much it contributes to constraint infeasibility. The steps of the algorithm as originally suggested by Senju and Toyoda are listed in Table 1.

The resource allocation formulation results from the following specifications:

c_j = net present value of project j, $

b_i = availability of resource i (appropriate units)

Table 1. The Original Method of Senju and Toyoda

STEP 1. If problem is a minimization, formulate the 1's complement maximization problem.

STEP 2. Set all $x_j=1$.

STEP 3. For each constraint $i=1,\ldots,m$, find the surplus

$$s_i=\sum_{j=1}^{n} x_j a_{ij}/b_i - 1$$

If $s_i<0$, set $s_i=0$. If all $s_i=0$, go to STEP 6.

STEP 4. For each variable $j=1,\ldots,n$, find the effective gradient

$$g_j=c_j/\left(\sum_{i=1}^{m} s_i a_{ij}/b_i\right)$$

STEP 5. Find $g_k=\min_j(g_j)$, such that $x_k=1$. Set $x_k=0$. If the total number of variables left to examine in the basis is 5 or less, totally enumerate to find the final solution and STOP. If the number of variables left to examine is greater than 5, go to STEP 3.

STEP 6. Can any non-basic variables be added back into the basis without becoming infeasible? If not, STOP; otherwise, go to STEP 7.

STEP 7. Formulate a new problem from the current non-basic variables as follows. For each constraint $i=1,\ldots,m$, subtract the sum of the coefficients of the basic variables from the right-hand side. Formulate new constraints from the coefficients of the non-basic variables in each constraint. Write a new objective function from the current objective function coefficients of the non-basic variables. Go to STEP 2.

a_{ij} = requirement of project j for resource i (appropriate units)

x_j = decision variable for project j; x_j=0 if project j is not done, and x_j=1 if project j is done.

This formulation has been discussed extensively by Weingartner (5).

The resource allocation formulation suggests a simplification that may be introduced into the method of Senju and Toyoda. In the process of dropping variables from the infeasible basis, it is possible that the first feasible basis encountered may leave so much slack in the constraints that some of the non-basic variables may become basic without ruining feasibility. In devising a way to reassign rejected variables to the basis, Senju and Toyoda reasoned that "what's good for the goose is good for the gander," and suggested that once the first feasible solution has been obtained, a new problem be formulated from the non-basic variables and the slack capacities left in the constraints. Their procedure formulates a new problem each time a feasible solution is obtained and augments the solution to the original problem by the basic variables in the solution to the latest sub-problem. The procedure can be repeated as long as a basic feasible solution can be found for each subproblem; when a subproblem remains with no basic feasible solution, the method terminates. The original method of Senju and Toyoda is thus rather complicated in concept.

The "nested-sub-problem" approach described above might have mathematically-comfortable appeal, but it presents some computational intricacies. A heuristic method should be appealing in at least three ways: efficiency, accuracy, and simplicity. These three features are in conflict with one another in Senju and Toyoda's method. The nested-sub-problem approach improves the accuracy of the method at the expense of efficiency and simplicity. For some large problems with only a small number of basic variables in the feasible basis, the heuristic might be bested by an implicit enumeration scheme given a good bound and starting feasible basis by methods similar to Petersen's (6). In any event, a simpler approach to the reassignment of the rejected variables would be highly desirable.

In order to partially overcome the types of difficulties conceived above, a modification of the heuristic was investigated. Rather than solve a nest of sub-problems, after the first feasible solution is obtained, non-basic variables that can be added back to the basis without ruining feasibility become basic in order of decreasing objective function coefficients. The procedure is outlined in Figure 1. Although the suggested modification does not offer the mathematical comfort of Senju and Toyoda's original approach, the method is considerably simpler and has a common-sense, economic basis.

Figure 1. Modification of the Method of Senju and Toyoda

START → Is problem "max"? → No → Find new right-hand sides

$b_i = \sum_{j=1}^{n} a_{ij} - d_i,\ i=1,\ldots,m$

Is problem "max"? → Yes ↓

Set all $x_j=1$

Find $s_i = \sum_{j=1}^{n} (a_{ij}/b_i) - 1$

$i=1,\ldots,m$

Are any $s_i > 0$? → No → (to "Have all non-basic variables been examined?")

Yes ↓

If $s_i<0$, set $s_i=0$, $i=1,\ldots,m$

Find

$g_j = c_j / \sum_{i=1}^{m} [(a_{ij}/b_i) x_j s_i]$

$j=1,\ldots,n$

Find k such that $g_k = \min_j(g_j)$ <u>and</u> $x_k=1$. Set $x_k=0$. (If k not unique, set $x_k=0$ for k with smallest c_j.)

Find $s_i = s_i - a_{ik}/b_i$, $i=1,\ldots,m$ → (back to "Are any $s_i>0$?")

Have all non-basic variables been examined? → Yes → Is problem "max"?

No ↓

Among unexamined non-basic variables find k such that $c_k = \max_j(c_j)$ with $x_k=0$

Does $x_k=1$ ruin feasibility? → Yes → (back to "Have all non-basic variables been examined?")

No ↓

Set $x_k=1$ → (back to "Have all non-basic variables been examined?")

Is problem "max"? → Yes → Find $Z_x = \sum_{j=1}^{n} c_j x_j$ → STOP; best value obtained.

No ↓

If $x_j=1$, set $y_j=0$; otherwise, $y_j=1$

Find $Z_y = \sum_{j=1}^{n} c_j y_j$

STOP; best value obtained.

KEY TO NOTATION

- x_j=variable in maximization problem
- y_j=variable in minimization problem
- c_j=objective function coefficient
- d_i or b_i=right-hand side of <u>i</u>th constraint (d for min, b for max
- a_{ij}=coefficient of the <u>j</u>th variable in the <u>i</u>th constraint
- m=number of constraints
- n=number of variables
- s_i=surplus in the left-hand side of the <u>i</u>th constraint
- g_j=effective gradient of the <u>j</u>th variable

Applied to resource allocation problems, the method of Senju and Toyoda identifies the least desirable projects in order of increasing resource efficiencies. When a feasible set of projects is found, the remaining projects are those that use the resources most inefficiently compared to all the rest (namely, those chosen to be done). Since none of the projects rejected used resources efficiently compared to those chosen, it would seem to be wasted effort to recompute resource efficiencies with respect to the amounts of unallocated resources remaining; the obvious, faster, more direct method to choose among the rejected projects is in order of decreasing net present value. The modification thus considers the initially rejected projects for reassignment in order of maximum contribution to the total net present value.

COMPUTATIONAL RESULTS

In order to test the efficiency and accuracy of the modification, a number of published solved problems were collected. Motivation for using published solved problems was twofold. First, such problems provide generally available, reliable benchmarks against which the modified version of Senju and Toyoda's heuristic may be evaluated for accuracy. Second, the published problems provide other authors' solution methods against which the efficiency of the modified heuristic may be compared.

The problem identifications and test results are given in Table 2. All the problems were solved with a FORTRAN IV computer program run on the Colorado School of Mines PDP-10. The program and data files are presented in the Appendix. Comments on the results are given below:

Capital Budgeting Problems from (7), (8), and (6). The heuristic seems to be fairly accurate on all but Petersen's #3. The computation times are much lower than those the original authors reported.

Weingartner and Ness's Multi-Dimensional Knapsack Problems (9). The first six problems differ only in their right-hand sides and show the effect of constraint severity on calculation times.

Set Covering Problems from (10), (11), and (12). The heuristic appears to perform well when applied to this type

Table 2. Computational Results with Published Problems

Author/Problem	Size (m x n)	Published Optimum Solution	Best Solution Obtained	Calculation Time* (Seconds)	CPU Time** (Seconds)
Cochran, et al (7)	3 x 14	35.777	35.777	0.100	0.33
Cord (8)	2 x 25	1.67	1.48	0.317	0.53
Petersen (6) #1	10 x 6	3,800	3,800	0.050	0.32
#2	10 x 10	8,706.1	8,336.9	0.100	0.52
#3	10 x 15	4,015	3,245	0.166	0.55
#4	10 x 20	6,120	5,965	0.384	0.82
#5	10 x 28	12,400	11,950	0.500	1.03
#6	5 x 39	10,618	10,601	0.550	0.93
#7	5 x 50	16,537	16,531	0.950	1.43
Weingartner and					
Ness (9) #1	2 x 28	141,278	140,786	0.267	0.48
#2	2 x 28	130,883	130,883	0.317	0.53
#3	2 x 28	95,667	95,667	0.366	0.62
#4	2 x 28	119,337	115,797	0.217	0.43
#5	2 x 28	98,796	98,631	0.400	0.62
#6	2 x 28	130,623	130,233	0.300	0.52
#7	2 x 105	1,095,445	1,095,112	1.983	2.58
#8	2 x 105	624,319	609,944	4.917	5.43
Haldi (3) IBM #9†	35 x 15	9	9	0.734	1.97
IBM #4†	15 x 15	10	11	0.433	1.00
IBM #6†	31 x 31	18	19	2.766	4.18
Lemke, et al (4)†	15 x 32	14	15	1.300	2.13
Pierce (6)†	5 x 31	61	78	0.166	0.48
Senju and Toyoda (7)					
#1	2 x 8	2,600	2,500	0.017	0.18
#2	4 x 6	1,800	1,800	0.017	0.20
#3	30 x 60	7,770††	7,648	11.000	13.73
#4	30 x 60	8,698§	8,698	7.267	9.70

*Measured by internal program clock; excludes input/ouput. **Error of at least ±0.20 second.
†Minimization problems. ††7,772 and §8,722 reported in (14).

of problem. A comparison of the method's speed with the results from Trauth and Woolsey (13) and also Haldi (10) is especially encouraging.

Senju and Toyoda's Example Problems (7). These are included to demonstrate the accuracy of the modification opposed to the original authors' results. The modification did worse on the tighter version of the large test problem (#1 in Table 2; see Appendix), but since Senju and Toyoda totally enumerated when five variables remained for consideration, the slightly lower objective value obtained was not entirely unexpected.

The modification of Senju and Toyoda's method is nearly as effective on their own example problems as their original, more complicated method. The heuristic seems especially appealing, however, when compared to other authors' methods. Weingartner's dynamic programming method (9), for example, requires extensive core storage for lists of incumbent solutions. Processing times he reported on the six problems ranged from 1.6 to 2.4 CPU seconds. His program was written in assembly language. The FORTRAN IV heuristic's times ranged from 0.43 to 0.62 CPU seconds. Although the quantitative impact of these data is discounted by the large error involved in measuring CPU times, the relative efficiency of the heuristic is quite dramatic. Weingartner stated that when the 105-variable problem had right-hand sides of 2,000 for both constraints, the processing time became too excessive to allow the dynamic programming code to finish. The heuristic solved the same problem in 3.05 CPU seconds.

Cochran, et al (7), offered their sample problem in an article describing a time-shared capital-budgeting system whose project-selection program used an optimizing implicit enumeration scheme. The authors wisely point out the trade-off that exists between the expected benefit derived from possible new knowledge obtained and the cost of obtaining that knowledge through extensive use of their optimization program. The authors further report that their program took approximately 20 minutes to solve a 16-variable problem, and they consider that the largest problem for time-shared solution. (It is notable that the heuristic took 0.330 CPU seconds to solve their 14-variable problem.) They dismiss the inefficiency by stating that the project-selection program need only be run a few times annually so that such inefficiency can be tolerated.

The heuristic would seem to offer two distinct advantages over the optimization scheme. First, much larger problems may be solved in a far more reasonable length of time. Second, with a greater capability at lower cost, additional use of a heuristic project-selection program for sensitivity analysis could be justified. These advantages are gained, of course, at the expense of not being guaranteed the optimum solution, but as Senju and Toyoda suggested in (1), the uncertainty of the data presented in real applications may offset possible error in the heuristic solution. The ultimate test of suitability, however, lies in some sort of benefit-cost analysis.

CONCLUSIONS AND SUGGESTIONS FOR FURTHER APPLICATION

The simpler modification of the Senju and Toyoda method seems to perform comparably to the original and offers many advantages over other optimal methods. In addition, the modified heuristic has a clear nonmathematical resource economic interpretation. The heuristic is also a tool for extending the concept of budgeting of capital among non-mutually exclusive projects to the allocation of all of a firm's scarce resources that constrain investment decisions.

Although many applications for the Senju and Toyoda heuristic may be seen in resource allocation, other models than net present value-maximization may require modification of the Senju and Toyoda heuristic to accommodate negative coefficients. Despite mathematical research along such lines, future modified heuristics will require economic interpretation akin to the one offered here. Without such "translation," actual user acceptance may be slight in the ranks of nontechnical management.

BIBLIOGRAPHY

1. Senju, Shizuo, and Toyoda, Yoshiaki, "An Approach to Linear Programming with 0-1 Variables," Management Science, vol. 15, no. 9 (May 1969), pp. 481-493.

2. Bandeen, R. A., at 1973 meeting of the Canadian Operations Research Society, Ste. Adele, Quebec.

3. Woolsey, R. E. D., Quick and Dirty Methods in Operations Research, in preparation.

4. Wyman, F. Paul, "Binary Programming: A Decision Rule for Selecting Optimal vs. Heuristic Techniques," American Institute of Decision Sciences, 2nd Annual Midwest Regional Conference, April 1971.

5. Weingartner, H. Martin, Mathematical Programming and the Analysis of Capital Budgeting Problems, Prentice-Hall, Englewood Cliffs, N. J., 1963.

6. Petersen, Clifford C., "Computational Experience with Variants of the Balas Algorithm Applied to the Selection of R & D Projects," Management Science, vol. 13, no. 9 (May 1967), pp. 736-750.

7. Cochran, M. A., Pyle, E. B., III, Greene, L., C., Clymer, H. A., and Bender, A. D., "Investment Model for R & D Project Evaluation and Selection," IEEE Transactions on Engineering Management, vol. EM-18, no. 3 (August 1971), pp. 89-100.

8. Cord, Joel A., "A Method for Allocating Funds to Investment

Projects When Returns Are Subject to Uncertainty," Management Science, vol. 10, no. 2 (January 1964), pp. 335-341.

9. Weingartner, H. Martin and Ness, David N., "Methods for the Solution of the Multi-Dimensional 0/1 Knapsack Problem," Operations Research, vol. 15, no. 1 (January-February 1967), pp. 83-103.

10. Haldi, J., "25 Integer Programming Test Problems," Working Paper No. 43, Graduate School of Business, Stanford University (December 1964).

11. Lemke, C. E., Salkin, H. M., and Spielberg, K., "Set Covering by Single-Branch Enumeration with Linear-Programming Subproblems," Operations Research, vol. 19, no. 4 (July-August 1971), pp. 998-1022.

12. Pierce, J. F., "Application of Combinatorial Programming to a Class of All-Zero-One Integer Programming Problems," IBM Report 36.Y03, Cambridge, Massachusetts, July 1966.

13. Trauth, C. A., Jr., and Woolsey, R. E. D., "Integer Linear Programming: A Study in Computational Efficiency," Management Science, vol. 15, no. 9 (May 1969), pp. 481-493.

14. Wood, D. R., Blackburn, C. R., and Robertson, J. L., "Real-Time Capital Allocation," Institute of Management Science, 11th American Meeting, Los Angeles, October 19-21, 1970.

APPENDIX

```
      DIMENSION A(60,105),B(60),C(105),X(105),S(60),G(105),Y(10
      DOUBLE PRECISION FMT1,FMT2
      INTEGER X,Y,OUT
C
C        SENJU & TOYODA PROGRAM FOR 0-1 INTEGER PROGRAMMING PROBLEMS
C
C
C
C
C        PROBLEM FORM MUST BE AS FOLLOWS:
C
C                MAXIMIZE
C                   OR    C(1)*X(1) + C(2)*X(2) + ... + C(N)*X(N)
C                MINIMIZE
C
C
C                SUBJECT TO CONSTRAINTS OF THE FORM
C                                                             .LE.
C                A(I,1)*X(1) + A(I,2)*X(2) + ... + A(I,N)*X(N) OR B(I)
C                                                             .GE.
C
C
C                WHERE   I = 1,...M
C
C                        X = 0 OR 1
C
C                AND A, B, AND C ARE ALL POSITIVE.
C
C        DATA FORMAT:
C
C           RECORDS 1 & 2 -- TWO INFORMATION CARDS FOR FILE IDENTIFICA-
C                                TION THAT ARE NOT READ AS DATA
C                RECORD 3 -- THE WORD "MAX" OR "MIN"
C                RECORD 4 -- NUMBER OF VARIABLES, NUMBER OF CONSTRAINTS
C                RECORD 5 -- FORMAT FOR READING OBJECTIVE FUNCTION
C                                AND CONSTRAINT COEFFICIENTS (CONSTRAINT
C                                COEFFICIENTS ARE READ ONE INEQUALITY AT
C                                A TIME TO READ LIKE THE OBJECTIVE
C                                FUNCTION)
C                RECORD 6 -- FORMAT FOR READING RIGHT-HAND SIDES
C                RECORD 7 -- OBJECTIVE FUNCTION COEFFICIENTS
C                RECORD 8 -- RIGHT-HAND SIDES OF CONSTRAINTS
C                RECORD 9 -- COEFFICIENTS OF CONSTRAINTS, ONE CONSTRAINT
C          THROUGH 9+M-1          AT A TIME
C
C
C        INPUT ON FILE 10, OUTPUT ON FILE 11
C
C        MAXIMUM PROBLEM SIZE --- 60 X 60
C
C
```

```
1ØØØ  FORMAT(//A3)
1     FORMAT(3A1Ø)
2     FORMAT(2G)
3     FORMAT(/' BEST VALUE OF THE OBJECTIVE FUNCTION IS ',F15.3//' BASI
     1S IS AS FOLLOWS:'/' ',6ØI1)
3ØØ   FORMAT(//F15.3,' SECONDS FOR CALCULATION.')
3Ø1   FORMAT(/' CONSTRAINT ',I2,1H','S SLACK IS ',F1Ø.4)
      READ(1Ø,1ØØØ)MAXMIN
      READ(1Ø,2)N,M
      READ(1Ø,1)(FMT1(I),I=1,3)
      READ(1Ø,1)(FMT2(I),I=1,3)
      READ(1Ø,FMT1)(C(J),J=1,N)
      READ(1Ø,FMT2)(B(I),I=1,M)
      DO 4 I=1,M
4     READ(1Ø,FMT1)(A(I,J),J=1,N)
      CALL RTIME(IN)
      IF(MAXMIN.EQ.'MAX')GO TO 7
C
C
C       PROBLEM IS MINIMIZATION; CHANGE FORM TO
C               SOLVE 1'S COMPLEMENT PROBLEM
C
C
      DO 6 I=1,M
      B(I)=-B(I)
      DO 6 J=1,N
6     B(I)=B(I)+A(I,J)
C
C
C       INITIALIZE X(J)=1
C
C
7     DO 8 J=1,N
8     X(J)=1
C
C
C       COMPUTE SURPLUS IN LEFT-HAND SIDES
C
C
      DO 1Ø I=1,M
1Ø2   S(I)=-1
      DO 11 J=1,N
11    S(I)=S(I)+A(I,J)/B(I)
1Ø    IF(S(I).LT.Ø)S(I)=Ø
C
C
C       IF ANY SURPLUSES ARE NON-ZERO, ANOTHER VARIABLE
C               MUST LEAVE THE BASIS
C
C
2Ø1   DO 12 I=1,M
```

```
12    IF(S(I).GT.Ø)GO TO 13
C
C
C       TRY TO ADD BACK NON-BASIC VARIABLES
C               WITHOUT BECOMING INFEASIBLE
C
C
      DO 21 I=1,M
      S(I)=B(I)
      DO 21 J=1,N
21    S(I)=S(I)-A(I,J)*X(J)
      DO 22 J=1,N
22    IF(X(J).EQ.Ø)Y(J)=1
23    XMAX=Ø
      DO 24 J=1,N
      XMAX=AMAX1(XMAX,(C(J)*Y(J)))
24    IF(XMAX.EQ.(C(J)*Y(J)).AND.XMAX.NE.Ø)K=J
      IF(XMAX.EQ.Ø)GO TO 25
      DO 27 I=1,M
27    IF(S(I).LT.A(I,K))GO TO 28
      DO 29 I=1,M
29    S(I)=S(I)-A(I,K)
      X(K)=1
28    Y(K)=Ø
      GO TO 23
25    IF(MAXMIN.EQ.'MAX')GO TO 3Ø
C
C
C       CHANGE BASIS TO ITS COMPLEMENT
C               FOR MINIMIZATION PROBLEM
C
C
      DO 26 J=1,N
      IF(X(J))26,261,262
261   X(J)=1
      GO TO 26
262   X(J)=Ø
26    CONTINUE
C
C
C       COMPUTE VALUE OF THE OBJECTIVE FUNCTION
C
C
3Ø    DO 14 J=1,N
14    OBJFUN=OBJFUN+C(J)*X(J)
      CALL RTIME(OUT)
      XOUT=FLOAT((OUT-IN))/1ØØØ
      WRITE(4,3ØØ)XOUT
      WRITE(4,3)OBJFUN,(X(J),J=1,N)
      WRITE(4,3Ø1)(I,S(I),I=1,M)
      STOP
```

```
C
C
C       COMPUTE GRADIENTS
C
C
13    DO 15 J=1,N
      G(J)=0
      DO 16 I=1,M
16    G(J)=A(I,J)/B(I)*X(J)*S(I)+G(J)
      IF(G(J).NE.0)G(J)=C(J)/G(J)
15    XLEG=AMAX1(XLEG,G(J))
C
C
C       FIND VARIABLE WITH THE LEAST EFFECTIVE GRADIENT
C               AND DROP IT FROM THE BASIS
C
C
      DO 18 J=1,N
18    IF(G(J).NE.0)XLEG=AMIN1(XLEG,G(J))
      DO 19 J=1,N
      IF(X(J).EQ.0.OR.G(J).GT.XLEG)GO TO 19
      X(J)=0
C
C
C       SUBTRACT A(I,J) FROM THE SURPLUS IN EACH CONSTRAINT
C
C
      DO 20 I=1,M
      S(I)=S(I)-A(I,J)/B(I)
20    IF(S(I).LT.0)S(I)=0
      GO TO 201
19    CONTINUE
      END
```

COCHRAN, ET AL, "INVESTMENT MODEL FOR R&D PROJECT EVALUATION AND SELECT-ION," IEEE TRANS, ON ENG, MGT,, V, EM-18, NO, 3, AUGUST 1971, P, 89.

```
MAX
14,3
(1ØG/4G)
(3G)
1.291,2.Ø35,.497,3.347,9.136,8.554,6.976,5.531,4.971,2.224
3.ØØ4,5.Ø97,6.Ø14,.976
1.6,2.1,2.9
.23,.31,.15,.18,.31,.1,.14,.55,.44,.12
.23,.35,.31,.11
.35,.33,.3,.29,.39,.2,.33,.8,.73,.28
.29,.39,.46,.16
.76,.45,.4,.53,.56,.3,.71,1.Ø9,.94,.44
.35,.51,.63,.21
```

```
CORD, J., "A METHOD FOR ALLOCATING FUNDS TO INVESTMENT PROJECTS WHEN RE-
TURNS ARE SUBJECT TO UNCERTAINTY," MGT. SCI., V. 1Ø, NO. 2, 1964, P. 335
MAX
25,2
(2(1ØG/),5G)
(2G)
.11,.17,.3,.Ø8,.27,.13,.1,.49,.44,.41
.41,.5,.41,.25,.Ø6,.19,.Ø6,.38,.39,.47
.43,.21,.49,.15,.12
2ØØ,.ØØ7
1Ø,37,8,99,12,66,31,85,63,73
98,11,83,88,99,65,8Ø,74,69,9
91,8Ø,44,12,63
.ØØØ4,.ØØ49,.Ø4,.Ø625,.ØØ36,.ØØØ4,.Ø196,.ØØ25,.Ø1,.ØØØ9
.ØØ49,.Ø625,.ØØ16,.Ø324,.Ø324,.ØØ36,.ØØØ4,.Ø225,.Ø529,.ØØ16
.Ø196,.ØØ36,.Ø121,.Ø225,.Ø441
```

```
PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM
APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V, 13, #9, 1967.
OPTIMUM SOLUTION = 3800
MAX
6,10
(6G)
(10G)
100,600,1200,2400,500,2000
80,96,20,36,44,48,10,18,22,24
8,12,13,64,22,41
8,12,13,75,22,41
3,6,4,18,6,4
5,10,8,32,6,12
5,13,8,42,6,20
5,13,8,48,6,20
0,0,0,0,8,0,
3,0,4,0,8,0
3,2,4,0,8,4
3,2,4,8,8,4
```

```
PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM
APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V. 13, #9, 1967.
OPTIMUM SOLUTION = 8706.1
MAX
10,10
(10G)
(10G)
600.1,310.5,1800,3850,18.6,198.7,882,4200,402.5,327
450,540,200,360,440,480,200,360,440,480
20,5,100,200,2,4,60,150,80,40
20,7,130,280,2,8,110,210,100,40
60,3,50,100,4,2,20,40,6,12
60,8,70,200,4,6,40,70,16,20
60,13,70,250,4,10,60,90,20,24
60,13,70,280,4,10,70,105,22,28
5,2,20,100,2,5,10,60,0,0
45,14,80,180,6,10,40,100,20,0
55,14,80,200,6,10,50,140,30,40
65,14,80,220,6,10,50,180,30,50
```

PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V, 13, #9, 1967.
OPTIMUM SOLUTION = 612Ø
MAX
15,1Ø
(15G)
(1ØG)
1ØØ,22Ø,9Ø,4ØØ,3ØØ,4ØØ,2Ø5,12Ø,16Ø,58Ø,4ØØ,14Ø,1ØØ,13ØØ,65Ø
55Ø,7ØØ,13Ø,24Ø,28Ø,31Ø,11Ø,2Ø5,26Ø,275
8,24,13,8Ø,7Ø,8Ø,45,15,28,9Ø,13Ø,32,2Ø,12Ø,4Ø
8,44,13,1ØØ,1ØØ,9Ø,75,25,28,12Ø,13Ø,32,4Ø,16Ø,4Ø
3,6,4,2Ø,2Ø,3Ø,8,3,12,14,4Ø,6,3,2Ø,5
5,9,6,4Ø,3Ø,4Ø,16,5,18,24,6Ø,16,11,3Ø,25
5,11,7,5Ø,4Ø,4Ø,19,7,18,29,7Ø,21,17,3Ø,25
5,11,7,55,4Ø,4Ø,21,9,18,29,7Ø,21,17,35,25
Ø,Ø,1,1Ø,4,1Ø,Ø,6,Ø,6,32,3,Ø,7Ø,1Ø
3,4,5,2Ø,14,2Ø,6,12,1Ø,18,42,9,12,1ØØ,2Ø
3,6,9,3Ø,29,2Ø,12,12,1Ø,3Ø,42,18,18,11Ø,2Ø
3,8,9,35,29,2Ø,16,15,1Ø,3Ø,42,2Ø,18,12Ø,2Ø

```
PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM
APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V, 13, #9, 1967.
OPTIMUM SOLUTION = 6120
MAX
20,10
(15G/5G)
(10G)
100,220,90,400,300,400,205,120,160,580,400,140,100,1300,650
320,480,80,60,2550
550,700,130,240,280,310,110,205,260,275
8,24,13,80,70,80,45,15,28,90,130,32,20,120,40
30,20,6,3,180
8,44,13,100,100,90,75,25,28,120,130,32,40,160,40
60,55,10,6,240
3,6,4,20,20,30,8,3,12,14,40,6,3,20,5
0,5,3,0,20
5,9,6,40,30,40,16,5,18,24,60,16,11,30,25
10,13,5,1,80
5,11,7,50,40,40,19,7,18,29,70,21,17,30,25
15,25,5,1,100
5,11,7,55,40,40,21,9,18,29,70,21,17,35,25
20,25,5,2,110
0,0,1,10,4,10,0,6,0,6,32,3,0,70,10
0,0,0,0,0
3,4,5,20,14,20,6,12,10,18,42,9,12,100,20
5,6,4,1,20
3,6,9,30,29,20,12,12,10,30,42,18,18,110,20
15,18,7,2,40
3,8,9,35,29,20,16,15,10,30,42,20,18,120,20
20,22,7,3,50
```

PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V, 13, #9, 1967.
OPTIMUM SOLUTION = 124ØØ

```
MAX
28,1Ø
(15G/13G)
(1ØG)
1ØØ,22Ø,9Ø,4ØØ,3ØØ,4ØØ,2Ø5,12Ø,16Ø,58Ø,4ØØ,14Ø,1ØØ,13ØØ,65Ø
32Ø,48Ø,8Ø,6Ø,255Ø,31ØØ,11ØØ,95Ø,45Ø,3ØØ,22Ø,2ØØ,52Ø
93Ø,121Ø,272,462,532,572,24Ø,4ØØ,47Ø,49Ø
8,24,13,8Ø,7Ø,8Ø,45,15,28,9Ø,13Ø,32,2Ø,12Ø,4Ø
3Ø,2Ø,6,3,18Ø,22Ø,5Ø,3Ø,5Ø,12,5,8,18
8,44,13,1ØØ,1ØØ,9Ø,75,25,28,12Ø,13Ø,32,4Ø,16Ø,4Ø
6Ø,55,1Ø,6,24Ø,29Ø,8Ø,9Ø,7Ø,27,17,8,28
3,6,4,2Ø,2Ø,3Ø,8,3,12,14,4Ø,6,3,2Ø,5
Ø,5,3,Ø,2Ø,3Ø,4Ø,1Ø,Ø,5,Ø,Ø,1Ø
5,9,6,4Ø,3Ø,4Ø,16,5,18,24,6Ø,16,11,3Ø,25
1Ø,13,5,1,8Ø,6Ø,5Ø,2Ø,3Ø,1Ø,5,3,2Ø
5,11,7,5Ø,4Ø,4Ø,19,7,18,29,7Ø,21,17,3Ø,25
15,25,5,1,1ØØ,7Ø,55,2Ø,5Ø,15,15,6,2Ø
5,11,7,55,4Ø,4Ø,21,9,18,29,7Ø,21,17,35,25
2Ø,25,5,2,11Ø,7Ø,55,2Ø,5Ø,2Ø,15,6,2Ø
Ø,Ø,1,1Ø,4,1Ø,Ø,6,Ø,6,32,3,Ø,7Ø,1Ø
Ø,Ø,Ø,Ø,0,3Ø,1Ø,Ø,1Ø,1Ø,5,Ø,1Ø
3,4,5,2Ø,14,2Ø,6,12,1Ø,18,42,9,12,1ØØ,2Ø
5,6,4,1,2Ø,5Ø,3Ø,5,2Ø,2Ø,1Ø,1Ø,2Ø
3,6,9,3Ø,29,2Ø,12,12,1Ø,3Ø,42,18,18,11Ø,2Ø
15,18,7,2,4Ø,6Ø,5Ø,25,25,25,15,1Ø,28
3,8,9,35,29,2Ø,16,15,1Ø,3Ø,42,2Ø,18,12Ø,2Ø
2Ø,22,7,3,5Ø,6Ø,55,25,3Ø,25,15,1Ø,28
```

```
PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM
APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V, 13, #9, 1967,
OPTIMUM SOLUTION = 10618
MAX
39,5
(3(10G/),9G)
(5G)
560,1125,300,620,2100,431,68,328,47,122
322,196,41,25,425,4260,416,115,82,22
631,132,420,86,42,103,215,81,91,26
49,420,316,72,71,49,108,116,90
600,500,500,500,600
40,91,10,30,160,20,3,12,3,18
9,25,1,1,10,280,10,8,1,1
49,8,21,6,1,5,10,8,2,1
0,10,42,6,4,8,0,10,1
16,92,41,16,150,23,4,18,6,0
12,8,2,1,0,200,20,6,2,1
70,9,22,4,1,5,10,6,4,0
4,12,8,4,3,0,10,0,6
38,39,32,71,80,26,5,40,8,12
30,15,0,1,23,100,0,20,3,0
40,6,8,0,6,4,22,4,6,1
5,14,8,2,8,0,20,0,0
8,71,30,60,200,18,6,30,4,8
31,6,3,0,18,60,21,4,0,2
32,15,31,2,2,7,8,2,8,0
2,8,6,7,1,0,0,20,8
38,52,30,42,170,9,7,20,0,3
21,4,1,2,14,310,8,4,6,1
18,15,38,10,4,8,6,0,0,3
0,10,6,1,3,0,3,5,4
```

```
PETERSEN, "COMPUTATIONAL EXPERIENCE WITH VARIANTS OF THE BALAS ALGORITHM
APPLIED TO THE SELECTION OF R&D PROJECTS," MGT SCI, V, 13, #9, 1967.
OPTIMUM SOLUTION = 16537
MAX
5Ø,5
(4(1ØG/),1ØG)
(5G)
56Ø,1125,3ØØ,62Ø,21ØØ,431,68,328,47,122
322,196,41,25,425,426Ø,416,115,82,22
631,132,42Ø,86,42,1Ø3,215,81,91,26
49,42Ø,316,72,71,49,1Ø8,116,9Ø,738
1811,43Ø,3Ø6Ø,215,58,296,62Ø,418,47,81
8ØØ,65Ø,55Ø,55Ø,65Ø
4Ø,91,1Ø,3Ø,16Ø,2Ø,3,12,3,18
9,25,1,1,1Ø,28Ø,1Ø,8,1,1
49,8,21,6,1,5,1Ø,8,2,1
Ø,1Ø,42,6,4,8,Ø,1Ø,1,4Ø
86,11,12Ø,8,3,32,28,13,2,4
16,92,41,16,15Ø,23,4,18,6,Ø
12,8,2,1,Ø,2ØØ,2Ø,6,2,1
7Ø,9,22,4,1,5,1Ø,6,4,Ø
4,12,8,4,3,Ø,1Ø,Ø,6,28
93,9,3Ø,22,Ø,36,45,13,2,2
38,39,32,71,8Ø,26,5,4Ø,8,12
3Ø,15,Ø,1,23,1ØØ,Ø,2Ø,3,Ø
4Ø,6,8,Ø,6,4,22,4,6,1
5,14,8,2,8,Ø,2Ø,Ø,Ø,6
12,6,8Ø,13,6,22,14,Ø,1,2
8,71,3Ø,6Ø,2ØØ,18,6,3Ø,4,8
31,6,3,Ø,18,6Ø,21,4,Ø,2
32,15,31,2,2,7,8,2,8,Ø
2,8,6,7,1,Ø,Ø,2Ø,8,14
2Ø,2,4Ø,6,1,14,2Ø,12,Ø,1
38,52,3Ø,42,17Ø,9,7,2Ø,Ø,3
21,4,1,2,14,31Ø,8,4,6,1
18,15,38,1Ø,4,8,6,Ø,Ø,3
Ø,1Ø,6,1,3,Ø,3,5,4,Ø
3Ø,12,16,18,3,16,22,3Ø,4,Ø
```

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL
0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.
OPTIMUM SOLUTION = 141278
MAX
28,2
(10G/10G/8G)
(2G)
1898,440,22507,270,14148,3100,4650,30800,615,4975
1160,4225,510,11880,479,440,490,330,110,560
24355,2885,11748,4550,750,3720,1950,10500
600,600
45,0,85,150,65,95,30,0,170,0
40,25,20,0,0,25,0,0,25,0
165,0,85,0,0,0,0,100
30,20,125,5,80,25,35,73,12,15
15,40,5,10,10,12,10,9,0,20
60,40,50,36,49,40,19,150

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL
0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.
OPTIMUM SOLUTION = 130883
MAX
28,2
(10G/10G/8G)
(2G)
1898,440,22507,270,14148,3100,4650,30800,615,4975
1160,4225,510,11880,479,440,490,330,110,560
24355,2885,11748,4550,750,3720,1950,10500
500,500
45,0,85,150,65,95,30,0,170,0
40,25,20,0,0,25,0,0,25,0
165,0,85,0,0,0,0,100
30,20,125,5,80,25,35,73,12,15
15,40,5,10,10,12,10,9,0,20
60,40,50,36,49,40,19,150

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL 0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.
OPTIMUM SOLUTION = 95677

```
MAX
28,2
(10G/10G/8G)
(2G)
1898,440,22507,270,14148,3100,4650,30800,615,4975
1160,4225,510,11880,479,440,490,330,110,560
24355,2885,11748,4650,750,3720,1950,10500
300,300
45,0,85,150,65,95,30,0,170,0
40,25,20,0,0,25,0,0,25,0
165,0,85,0,0,0,0,100
30,20,125,5,80,25,35,73,12,15
15,40,5,10,10,12,10,9,0,20
60,40,50,36,49,40,19,150
```

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL 0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.

```
OPTIMUM SOLUTION = 199337
MAX
28,2
(10G/10G/8G)
(2G)
1898,440,22507,270,14148,3100,4650,30800,615,4975
1160,4225,510,11880,479,440,490,330,110,560
24355,2885,11748,4550,750,3720,1950,10500
300,600
45,0,85,150,65,95,30,0,170,0
40,25,20,0,0,25,0,0,25,0
165,0,85,0,0,0,0,100
30,20,125,5,80,25,35,73,12,15
15,40,5,10,10,12,10,9,0,20
60,40,50,36,49,40,19,150
```

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL 0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.

```
OPTIMUM SOLUTION = 98796
MAX
28,2
(10G/10G/8G)
(2G)
1898,440,22507,270,14148,3100,4650,30800,615,4975
1160,4225,510,11880,479,440,490,330,110,560
24355,2885,11748,4550,750,3720,1950,10500
600,300
45,0,85,150,65,95,30,0,170,0
40,25,20,0,0,25,0,0,25,0
165,0,85,0,0,0,0,100
30,20,125,5,80,25,35,73,12,15
15,40,5,10,10,12,10,9,0,20
60,40,50,36,49,40,19,150
```

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL 0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.

```
OPTIMUM SOLUTION = 130623
MAX
28,2
(10G/10G/8G)
(2G)
1898,440,22507,270,14148,3100,4650,30800,615,4975
1160,4225,510,11880,479,440,490,330,110,560
24355,2885,11748,4550,750,3720,1950,10500
562,497
45,0,85,150,65,95,30,0,170,0
40,25,20,0,0,25,0,0,25,0
165,0,85,0,0,0,0,100
30,20,125,5,80,25,35,73,12,15
15,40,5,10,10,12,10,9,0,20
60,40,50,36,49,40,19,150
```

```
WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL
0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83,
OPTIMUM SOLUTION = 1095445
MAX
105,2
(10(10G/),5G)
(2G)
41850,38261,23800,21697,7074,5587,5560,5500,3450,2391
761,460,367,24785,47910,30250,107200,4235,9835,9262
15000,6399,6155,10874,37100,27040,4117,32240,1600,4500
70610,6570,15290,23840,16500,7010,16020,8000,31026,2568
2365,4350,1972,4975,29400,7471,2700,3840,22400,3575
13500,1125,11950,12753,10568,15600,20652,13150,2900,1790
4970,5770,8180,2450,7140,12470,6010,16000,11100,11093
4685,2590,11500,5820,2842,5000,3300,2800,5420,900
13300,8450,5300,750,1435,2100,7215,2605,2422,5500
8550,2700,540,2550,2450,725,445,700,1720,2675
220,300,405,150,70
3000,3000
75,40,365,95,25,17,125,20,22,84
75,50,15,0,0,12,0,10,0,50
0,0,10,0,0,50,60,150,0,0
75,0,102,0,0,40,60,0,165,0
0,0,45,0,0,0,25,0,150,0
0,0,158,0,85,95,0,89,20,0
0,0,0,0,0,80,0,110,0,15
0,60,5,135,0,0,25,0,300,35
100,0,0,25,0,0,225,25,0,0
0,0,0,0,0,5,0,60,0,100
0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,5,10,10,50,2,5,5
10,5,6,11,41,30,5,40,2,6
100,10,25,39,30,13,30,15,60,5
5,10,5,15,91,24,10,15,90,15
60,5,55,60,50,75,100,65,15,10
30,35,50,15,45,80,40,110,80,80
36,20,90,50,25,50,35,30,60,10
150,110,70,10,20,30,104,40,40,94
150,50,10,50,50,16,10,20,50,90
10,15,39,20,20
```

```
WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL
0/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.
OPTIMUM SOLUTION = 624319
MAX
105,2
(10(10G/),5G)
(2G)
41850,38261,23800,21697,7074,5587,5560,5500,3450,2391
761,460,367,24785,47910,30250,107200,4235,9835,9262
15000,6399,6155,10874,37100,27040,4117,32240,1600,4500
70610,6570,15290,23840,16500,7010,16020,8000,31026,2568
2365,4350,1972,4975,29400,7471,2700,3840,22400,3575
13500,1125,11950,12753,10568,15600,20652,13150,2900,1790
4970,5770,8180,2450,7140,12470,6010,16000,11100,11093
4685,2590,11500,5820,2842,5000,3300,2800,5420,900
13300,8450,5300,750,1435,2100,7215,2605,2422,5500
8550,2700,540,2550,2450,725,445,700,1720,2675
220,300,405,150,70
500,500
75,40,365,95,25,17,125,20,22,84
75,50,15,0,0,12,0,10,0,50
0,0,10,0,0,50,60,150,0,0
75,0,102,0,0,40,60,0,165,0
0,0,45,0,0,0,25,0,150,0
0,0,158,0,85,95,0,89,20,0
0,0,0,0,0,80,0,110,0,15
0,60,5,135,0,0,25,0,300,35
100,0,0,25,0,0,225,25,0,0
0,0,0,0,0,5,0,60,0,100
0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,5,10,10,50,2,5,5
10,5,6,11,41,30,5,40,2,6
100,10,25,39,30,13,30,15,60,5
5,10,5,15,91,24,10,15,90,15
60,5,55,60,50,75,100,65,15,10
30,35,50,15,45,80,40,110,80,80
36,20,90,50,25,50,35,30,60,10
150,110,70,10,20,30,104,40,40,94
150,50,10,50,50,16,10,20,50,90
10,15,39,20,20
```

WEINGARTNER, H. M., "METHODS FOR THE SOLUTION OF THE MULTI-DIMENSIONAL Ø/1 KNAPSACK PROBLEM," ORSA JOUR., V. 15, NO. 1, JANUARY 1967, P. 83.

```
OPTIMUM SOLUTION = 624319
MAX
1Ø5,2
(1Ø(1ØG/),5G)
(2G)
4185Ø,38261,238ØØ,21697,7Ø74,5587,556Ø,55ØØ,345Ø,2391
761,46Ø,367,24785,4791Ø,3Ø25Ø,1Ø72ØØ,4235,9835,9262
15ØØØ,6399,6155,1Ø874,371ØØ,27Ø4Ø,4117,3224Ø,16ØØ,45ØØ
7Ø61Ø,657Ø,1529Ø,2384Ø,165ØØ,7Ø1Ø,16Ø2Ø,8ØØØ,31Ø26,2568
2365,435Ø,1972,4975,294ØØ,7471,27ØØ,384Ø,224ØØ,3575
135ØØ,1125,1195Ø,12753,1Ø568,156ØØ,2Ø652,1315Ø,29ØØ,179Ø
497Ø,577Ø,818Ø,245Ø,714Ø,1247Ø,6Ø1Ø,16ØØØ,111ØØ,11Ø93
4685,259Ø,115ØØ,582Ø,2842,5ØØØ,33ØØ,28ØØ,542Ø,9ØØ
133ØØ,845Ø,53ØØ,75Ø,1435,21ØØ,7215,26Ø5,2422,55ØØ
855Ø,27ØØ,54Ø,255Ø,245Ø,725,445,7ØØ,172Ø,2675
22Ø,3ØØ,4Ø5,15Ø,7Ø
2ØØØ,2ØØØ
75,4Ø,365,95,25,17,125,2Ø,22,84
75,5Ø,15,Ø,Ø,12,Ø,1Ø,Ø,5Ø
Ø,Ø,1Ø,Ø,Ø,5Ø,6Ø,15Ø,Ø,Ø
75,Ø,1Ø2,Ø,Ø,4Ø,6Ø,Ø,165,Ø
Ø,Ø,45,Ø,Ø,Ø,25,Ø,15Ø,Ø
Ø,Ø,158,Ø,85,95,Ø,89,2Ø,Ø
Ø,Ø,Ø,Ø,Ø,8Ø,Ø,11Ø,Ø,15
Ø,6Ø,5,135,Ø,Ø,25,Ø,3ØØ,35
1ØØ,Ø,Ø,25,Ø,Ø,225,25,Ø,Ø
Ø,Ø,Ø,Ø,Ø,5,Ø,6Ø,Ø,1ØØ
Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,5,1Ø,1Ø,5Ø,2,5,5
1Ø,5,6,11,41,3Ø,5,4Ø,2,6
1ØØ,1Ø,25,39,3Ø,13,3Ø,15,6Ø,5
5,1Ø,5,15,91,24,1Ø,15,9Ø,15
6Ø,5,55,6Ø,5Ø,75,1ØØ,65,15,1Ø
3Ø,35,5Ø,15,45,8Ø,4Ø,11Ø,8Ø,8Ø
36,2Ø,9Ø,5Ø,25,5Ø,35,3Ø,6Ø,1Ø
15Ø,11Ø,7Ø,1Ø,2Ø,3Ø,1Ø4,4Ø,4Ø,94
15Ø,5Ø,1Ø,5Ø,5Ø,16,1Ø,2Ø,5Ø,9Ø
1Ø,15,39,2Ø,2Ø
```

HALDI, J., "25 INTEGER PROGRAMMING TEST PROBLEMS," WORKING PAPER NO. 43, GRADUATE SCHOOL OF BUSINESS, STANFORD U., (IBM TEST PROBLEM #9)

```
OPTIMUM SOLUTION = 9
MIN
15,35
(15G)
(2ØG/15G)
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
Ø,Ø,1,1,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,1,1,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,Ø,Ø,1,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,1,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø
Ø,1,1,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø
Ø,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,1,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,1,Ø,1,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,1,Ø,1,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø,1,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø,1,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,1,Ø,Ø,1,Ø,Ø
Ø,Ø,Ø,Ø,Ø,1,1,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø,Ø,Ø,Ø,Ø,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,1,Ø,1,Ø,Ø,Ø,1,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,1,Ø,Ø,Ø,1,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,1,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,1
1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø
Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1
Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,1
Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø,Ø,Ø
Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø,Ø
1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1
Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,1,Ø,Ø
Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,1,Ø
Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,1
Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1,Ø
1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø
Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø
Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø
Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1
```

```
HALDI, J., "25 INTEGER PROGRAMMING TEST PROBLEMS," WORKING PAPER NO.
43, GRADUATE SCHOOL OF BUSINESS, STANFORD U., (IBM TEST PROBLEM #4)
OPTIMUM SOLUTION = 1Ø
MIN
15,15
(15G)
(15G)
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
6,6,6,6,5,5,5,5,5,5,4,4,4,4,3
1,Ø,Ø,Ø,1,1,1,Ø,Ø,Ø,1,1,1,Ø,1
Ø,1,Ø,Ø,1,Ø,Ø,1,1,Ø,1,1,Ø,1,1
Ø,Ø,1,Ø,Ø,1,Ø,1,Ø,1,1,Ø,1,1,Ø
Ø,Ø,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,1,1,1
1,1,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,1,1,Ø
1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø
1,Ø,Ø,1,1,1,Ø,Ø,1,1,1,Ø,Ø,1,Ø
Ø,1,1,Ø,1,1,Ø,Ø,1,1,Ø,1,1,Ø,Ø
Ø,1,Ø,1,1,Ø,1,1,Ø,1,1,Ø,1,Ø,Ø
Ø,Ø,1,1,Ø,1,1,1,1,Ø,1,1,Ø,Ø,Ø
1,1,1,Ø,Ø,Ø,1,Ø,1,1,1,Ø,Ø,Ø,1
1,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1
1,Ø,1,1,1,Ø,Ø,1,1,Ø,Ø,Ø,1,Ø,1
Ø,1,1,1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,1,1
1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1,Ø
```

HALDI, J., "25 INTEGER PROGRAMMING TEST PROBLEMS," WORKING PAPER NO. 43, GRADUATE SCHOOL OF BUSINESS, STANFORD U., (IBM TEST PROBLEM #6)
OPTIMUM SOLUTION = 18

```
MIN
31,31
(31G)
(31G)
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
1Ø,1Ø,9,1Ø,9,9,8,1Ø,9,9,8,9,8,8,7,1Ø,9,9,8,9,8,8,7,9,8,8,7,8,7,7,6
1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1
Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1
1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø
Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1
1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø
Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø
1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1,1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1,1,1,1,1
1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø
Ø,1,1,Ø,Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,1,1,Ø,Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø
1,1,Ø,Ø,1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,Ø,1,1,Ø,Ø,1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1
Ø,Ø,Ø,1,1,1,1,1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1,1,1,1,1,Ø,Ø,Ø,Ø
1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1
Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,1,1,Ø,Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,1,1
1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,Ø,1,1,Ø,Ø,1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,Ø,1,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø
Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø
1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1
Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø
1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1
Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1
1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1,Ø,1,Ø,1
Ø,1,1,Ø,Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,1,1,Ø,Ø,1,1
1,1,Ø,Ø,1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,1,Ø,Ø,1,Ø,1,1,Ø,Ø,1,1,Ø
Ø,Ø,Ø,1,1,1,1,1,1,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,1,1
1,Ø,1,1,Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,Ø,1,Ø,1,1,Ø,1,Ø
Ø,1,1,1,1,Ø,Ø,1,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø,Ø,1,1,Ø,Ø,1,1,1,1,Ø,Ø
1,1,Ø,1,Ø,Ø,1,1,Ø,Ø,1,Ø,1,1,Ø,1,Ø,Ø,1,Ø,1,1,Ø,Ø,1,1,Ø,1,Ø,Ø,1
```

```
LEMKE, ET AL, "SET COVERING BY SINGLE-BRANCH ENUMERATION WITH LINEAR-
PROGRAMMING SUBPROBLEMS," ORSA JOUR., V, 19, #4, JULY-AUG 1971, P. 998,
MIN
32,15
(2(1ØG/),12G)
(15G)
1,1,1,1,1,1,1,1,2,2
2,2,2,2,2,3,3,3,3,4
4,4,4,5,5,5,6,6,6,7,8,9
1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø
Ø,1,Ø,Ø,1,Ø,Ø,1,Ø,Ø,Ø,1
Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,1,Ø,Ø,1,Ø,Ø,Ø,1,Ø
1,Ø,1,Ø,Ø,Ø,1,Ø,1,1,1,1
Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,1,Ø,1,Ø,Ø,Ø,1,Ø,1
Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,Ø,Ø,1,Ø,Ø,1,Ø,Ø,Ø,1
Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø
1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø
Ø,1,1,Ø,Ø,Ø,1,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1
Ø,Ø,Ø,1,Ø,1,Ø,Ø,Ø,1,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø,Ø
Ø,1,Ø,Ø,Ø,Ø,1,Ø,Ø,Ø
Ø,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,1,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1
Ø,Ø,1,1,Ø,Ø,1,Ø,Ø,Ø,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø
Ø,Ø,1,1,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1,1,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,1,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,1,1,Ø,Ø,1,Ø,Ø
Ø,Ø,1,1,Ø,Ø,1,Ø,Ø,1,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,1,Ø
Ø,1,Ø,1,Ø,1,Ø,Ø,1,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1
Ø,Ø,Ø,Ø,1,Ø,Ø,1,Ø,1,Ø,1
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,1,Ø,Ø,1,Ø
```

Ø,Ø,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,1,Ø,Ø
Ø,Ø,Ø,1,Ø,1,Ø,1,Ø,Ø,1,1

```
PIERCE, J. F., "APPLICATION OF COMBINATORIAL PROGRAMMING TO A CLASS
OF ALL-ZERO-ONE INTEGER PROGRAMMING PROBLEMS," JULY 1966.
MIN
31,5
(2(1ØG/),11G)
(5G)
68,55,57,6Ø,64,49,5Ø,51,54,54
4Ø,62,45,53,58,5Ø,32,45,34,34
23,26,42,31,6,17,47,36,34,28,11
1,1,1,1,1
1,1,1,1,1,1,1,1,1,1
1,1,1,1,1,1,Ø,Ø,Ø,Ø
Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,1,1,Ø,1,1,1,Ø,1,Ø
1,Ø,Ø,Ø,Ø,Ø,1,1,1,1
1,1,1,1,Ø,Ø,Ø,Ø,Ø,Ø,Ø
1,Ø,1,1,1,Ø,Ø,1,1,Ø
Ø,1,Ø,1,Ø,Ø,Ø,1,1,1
1,Ø,Ø,Ø,1,1,1,1,Ø,Ø,Ø
1,1,1,1,Ø,1,Ø,Ø,Ø,1
Ø,1,Ø,Ø,1,Ø,1,1,1,Ø
Ø,1,Ø,Ø,Ø,Ø,1,1,1,1,Ø
1,1,Ø,1,1,Ø,1,1,Ø,1
Ø,Ø,1,Ø,Ø,Ø,1,1,Ø,1
Ø,Ø,1,Ø,Ø,1,1,Ø,1,Ø,1
```

SENJU AND TOYODA, "AN APPROACH TO LINEAR PROGRAMMING WITH 0-1 VARI-
ABLES," MGT SCI, V. 15, NO, 4, 1968, P, B-196.
OPTIMUM SOLUTION = 2600
MAX
8,2
(8G)
(2G)
100,400,200,800,300,600,400,500
24,30
6,2,3,4,9,6,5,1
2,8,2,6,3,5,6,7

```
SENJU AND TOYODA, "AN APPROACH TO LINEAR PROGRAMMING WITH 0-1 VARI-
ABLES," MGT. SCI., V. 15, NO. 4, 1968, P. B-196.
OPTIMUM SOLUTION = 1800
MAX
6,4
(6G)
(4G)
500,900,300,600,100,300
16,16,16,16
1,4,5,7,5,6
5,3,2,8,4,9
4,3,4,4,2,5
6,5,3,7,2,5
```

SENJU AND TOYODA, "AN APPROACH TO LINEAR PROGRAMMING WITH 0-1 VARI-
ABLES," MGT SCI, V. 15, NO, 4, 1968, P, B-196.
BEST KNOWN VALUE OF OPTIMUM = 7772
MAX
60,30
(5(10G/),10G)
(2(10G/),10G)
2,77,6,67,930,3,6,270,33,13
110,21,56,974,47,734,238,75,200,51
47,63,7,6,468,72,95,82,91,83
27,13,6,76,55,72,300,6,65,39
63,61,52,85,29,640,558,53,47,25
3,6,568,6,2,780,69,31,774,22
6000,6000,6000,6000,6000,6000,6000,6000,6000,4000
6000,6000,6000,6000,6000,6000,6000,6000,6000,4000
6000,6000,6000,6000,6000,6000,6000,6000,6000,4000
47,774,76,56,59,22,42,1,21,760
818,62,42,36,785,29,662,49,608,116
834,57,42,39,994,690,27,524,23,96
667,490,805,46,19,26,97,71,699,465
53,26,123,20,25,450,22,979,75,96
27,41,21,81,15,76,97,646,898,37
73,67,27,99,35,794,53,378,234,32
792,97,64,19,435,712,837,22,504,332
13,65,86,29,894,266,75,16,86,91
67,445,118,73,97,370,88,85,165,268
758,21,255,81,5,774,39,377,18,370
96,61,57,23,13,164,908,834,960,87
36,42,56,96,438,49,57,16,978,9
644,584,82,550,283,340,596,788,33,350
55,59,348,66,468,983,6,33,42,96
464,175,33,97,15,22,9,554,358,587
71,23,931,931,94,798,73,873,22,39
71,864,59,82,16,444,37,475,65,5
47,114,26,668,82,43,55,55,56,27
716,7,77,26,950,320,350,95,714,789
430,97,590,32,69,264,19,51,97,33
571,388,602,140,15,85,42,66,778,936
61,23,449,973,828,33,53,297,75,3,
54,27,918,11,620,13,28,80,79,3
61,720,7,31,22,82,688,19,82,654
809,99,81,97,830,826,775,72,9,719
740,860,72,30,82,112,66,638,150,13
586,590,519,2,320,13,964,754,70,241
72,12,996,868,36,91,79,221,49,690
23,18,748,408,688,97,85,777,294,17
698,53,290,3,62,37,704,810,42,17
983,11,45,56,234,389,712,664,59,15
22,91,57,784,75,719,294,978,75,86
105,227,760,2,190,3,71,32,210,678
41,93,47,581,37,977,62,503,32,85

31,36,30,328,74,31,56,891,62,97
71,37,978,93,9,23,47,71,744,9
619,32,214,31,796,103,593,16,468,700
884,67,36,3,93,71,734,504,81,53
609,14,293,31,75,59,99,11,67,306
96,218,845,303,3,319,86,724,22,838
82,5,330,58,55,66,53,916,89,56
33,27,13,57,6,87,21,12,15,290
206,420,32,880,854,417,770,4,12,952
604,13,96,910,34,460,76,16,140,100
876,622,559,39,640,59,6,244,232,513
644,7,813,624,990,274,808,372,2,694
804,39,5,644,914,484,1,8,43,92
16,36,538,210,844,520,33,73,100,284
650,85,894,2,206,637,324,318,7,566
46,818,92,65,520,721,90,53,174,43
320,812,382,16,873,678,29,92,755,827
27,218,143,12,57,480,154,944,7,730
12,65,67,39,390,32,39,318,47,86
45,51,59,21,53,43,25,7,42,27
310,45,72,53,798,304,354,79,45,44
52,76,45,26,27,968,86,16,62,85
790,208,390,36,62,83,93,16,574,150
99,7,920,860,12,404,31,560,37,32
9,62,7,43,17,77,73,368,66,82
11,51,97,26,83,426,92,39,66,2
23,93,85,660,85,774,77,77,927,868
7,554,760,104,48,202,45,75,51,55
716,752,37,95,267,91,5,956,444,529
96,99,17,99,62,7,394,580,604,89
678,476,97,234,1,608,19,69,676,51
410,89,414,81,130,491,6,238,79,43
5,288,910,204,948,19,644,21,295,11
6,595,904,67,51,703,430,95,408,89
11,495,844,13,417,570,9,429,16,939
430,270,49,72,65,66,338,994,167,76
47,211,87,39,1,570,85,134,967,12
553,63,35,63,98,402,664,85,458,834
3,62,508,7,1,72,88,45,496,43
750,222,96,31,278,184,36,7,210,55
653,51,35,37,393,2,49,884,418,379
75,338,51,21,29,95,790,846,720,71
728,930,95,1,910,5,804,5,284,128
423,6,58,36,37,321,22,26,16,27
218,530,93,55,89,71,828,75,628,67
66,622,440,91,73,790,710,59,83,968
129,632,170,67,613,608,43,71,730,910
36,92,950,138,23,95,460,62,189,73
65,943,62,554,46,318,13,540,90,53
967,654,46,69,26,769,82,89,15,87
46,59,22,840,66,35,684,57,254,230

21,586,51,19,984,156,23,748,760,65
339,892,13,13,327,65,35,246,71,178
83,3,34,624,788,200,980,882,343,550
708,542,53,72,86,51,700,524,577,948
132,900,72,51,91,150,22,110,154,148
99,75,21,544,110,11,52,840,201,2
6,663,22,20,89,10,93,964,924,73
501,398,3,2,279,5,288,80,91,132
620,628,57,79,2,874,36,497,846,22
350,866,57,86,83,178,968,52,399,628
869,26,710,37,81,89,6,82,82,56
96,66,46,13,934,49,394,72,194,408
5,541,88,93,36,398,508,89,66,16
71,466,7,95,464,41,69,130,488,695
82,39,95,53,37,200,87,56,268,71
304,855,22,564,47,26,26,370,569,2
494,2,25,61,674,638,61,59,62,690
630,86,198,24,15,650,75,25,571,338
268,958,95,898,56,585,99,83,21,600
462,940,96,464,228,93,72,734,89,287
174,62,51,73,42,838,82,515,232,91
25,47,12,56,65,734,70,48,209,71
267,290,31,844,12,570,13,69,65,848
72,780,27,96,97,17,69,274,616,36
554,236,47,7,47,134,76,62,824,55
374,471,478,504,496,754,604,923,330,22
97,6,2,16,14,958,53,480,482,93
57,641,72,75,51,96,83,47,403,32
624,7,96,45,97,148,91,3,69,26
22,45,42,2,75,76,96,67,688,2
2,224,83,69,41,660,81,89,93,27
214,458,66,72,384,59,76,538,15,840
65,63,77,33,92,32,35,832,970,49
13,8,77,75,51,95,56,63,578,47
33,62,928,292,2,340,278,911,818,770
464,53,888,55,76,31,389,40,864,36
35,37,69,95,22,648,334,14,198,42
73,594,95,32,814,45,45,515,634,254
42,29,15,83,55,176,35,46,60,296
262,598,67,644,80,999,3,727,79,374
19,780,400,588,37,86,23,583,518,42
56,1,108,83,43,720,560,81,674,25
96,218,6,69,107,534,158,56,5,938
9,938,274,76,298,9,518,571,47,175
63,93,49,94,42,26,79,50,718,926
419,810,23,363,519,339,86,751,7,86
47,75,55,554,3,800,6,13,85,65
99,45,69,73,864,95,199,924,19,948
214,3,718,56,278,1,363,86,1,22
56,114,13,53,56,19,82,88,99,543
674,704,418,670,554,282,5,67,63,466

```
491,49,67,154,956,911,77,635,2,49
53,12,79,481,218,26,624,954,13,580
130,608,37,91,78,743,1,950,45
41,718,36,30,534,418,452,359,759,88
29,499,55,974,93,56,108,257,93,171
13,92,63,714,9,84,890,16,930,967
748,5,7,6,327,894,33,629,448,21
9,19,7,535,75,3,27,928,21,7
864,27,73,61,25,75,876,16,92,22
248,11,86,944,872,996,252,2,800,334
93,107,254,441,930,744,97,177,498,931
694,800,9,36,6,539,35,79,130,860
710,7,630,475,903,552,2,45,97,974
17,36,77,843,328,22,76,368,39,71
35,850,96,93,87,56,972,96,594,864
344,76,17,17,576,629,780,640,56,65
43,196,520,86,92,31,6,593,174,569
89,718,83,8,790,285,780,62,378,313
519,2,85,845,931,731,42,365,32,33
65,59,2,671,26,364,854,526,570,630
33,654,95,41,42,27,584,17,724,59
42,26,918,6,242,356,75,644,818,168
964,12,97,178,634,21,3,586,47,382
804,89,194,21,610,168,79,96,87,266
482,46,96,969,629,128,924,812,19,2
468,13,9,120,73,7,92,99,93,418
224,22,7,29,57,33,949,65,92,898
200,56,12,31,296,185,272,91,77,37
734,911,27,310,59,33,87,872,73,79
920,85,59,72,888,49,12,79,538,947
462,444,828,935,518,894,13,591,22,920
23,93,87,490,32,63,870,393,52,23
63,634,39,83,12,72,131,69,984,87
86,99,52,110,183,704,232,674,384,47
804,99,83,81,174,99,77,708,7,623
114,1,750,49,284,492,11,61,6,449
429,52,62,482,826,147,338,911,30,984
35,55,21,264,5,35,92,128,65,27
9,52,66,51,7,47,670,83,76,7
79,37,2,46,480,608,990,53,47,19
35,518,71,69,32,87,96,240,52,310
86,73,52,31,83,544,16,15,21,774
224,7,83,680,554,310,96,844,29,61
```

SENJU AND TOYODA, "AN APPROACH TO LINEAR PROGRAMMING WITH 0-1 VARI-
ABLES," MGT SCI, V. 15, NO, 4, 1968, P, B-196.
BEST KNOWN VALUE OF OPTIMUM = 8722
MAX
60,30
(5(10G/),10G)
(2(10G/),10G)
2,77,6,67,930,3,6,270,33,13
110,21,56,974,47,734,238,75,200,51
47,63,7,6,468,72,95,82,91,83
27,13,6,76,55,72,300,6,65,39
63,61,52,85,29,640,558,53,47,25
3,6,568,6,2,780,69,31,774,22
10000,10000,10000,10000,10000,10000,10000,10000,10000,7000
10000,10000,10000,10000,10000,10000,10000,10000,10000,7000
10000,10000,10000,10000,10000,10000,10000,10000,10000,7000
47,774,76,56,59,22,42,1,21,760
818,62,42,36,785,29,662,49,608,116
834,57,42,39,994,690,27,524,23,96
667,490,805,46,19,26,97,71,699,465
53,26,123,20,25,450,22,979,75,96
27,41,21,81,15,76,97,646,898,37
73,67,27,99,35,794,53,378,234,32
792,97,64,19,435,712,837,22,504,332
13,65,86,29,894,266,75,16,86,91
67,445,118,73,97,370,88,85,165,268
758,21,255,81,5,774,39,377,18,370
96,61,57,23,13,164,908,834,960,87
36,42,56,96,438,49,57,16,978,9
644,584,82,550,283,340,596,788,33,350
55,59,348,66,468,983,6,33,42,96
464,175,33,97,15,22,9,554,358,587
71,23,931,931,94,798,73,873,22,39
71,864,59,82,16,444,37,475,65,5
47,114,26,668,82,43,55,55,56,27
716,7,77,26,950,320,350,95,714,789
430,97,590,32,69,264,19,51,97,33
571,388,602,140,15,85,42,66,778,936
61,23,449,973,828,33,53,297,75,3,
54,27,918,11,620,13,28,80,79,3
61,720,7,31,22,82,688,19,82,654
809,99,81,97,830,826,775,72,9,719
740,860,72,30,82,112,66,638,150,13
586,590,519,2,320,13,964,754,70,241
72,12,996,868,36,91,79,221,49,690
23,18,748,408,688,97,85,777,294,17
698,53,290,3,62,37,704,810,42,17
983,11,45,56,234,389,712,664,59,15
22,91,57,784,75,719,294,978,75,86
105,227,760,2,190,3,71,32,210,678
41,93,47,581,37,977,62,503,32,85

31,36,30,328,74,31,56,891,62,97
71,37,978,93,9,23,47,71,744,9
619,32,214,31,796,103,593,16,468,700
884,67,36,3,93,71,734,504,81,53
609,14,293,31,75,59,99,11,67,306
96,218,845,303,3,319,86,724,22,838
82,5,330,58,55,66,53,916,89,56
33,27,13,57,6,87,21,12,15,290
206,420,32,880,854,417,770,4,12,952
604,13,96,910,34,460,76,16,140,100
876,622,559,39,640,59,6,244,232,513
644,7,813,624,990,274,808,372,2,694
804,39,5,644,914,484,1,8,43,92
16,36,538,210,844,520,33,73,100,284
650,85,894,2,206,637,324,318,7,566
46,818,92,65,520,721,90,53,174,43
320,812,382,16,873,678,29,92,755,827
27,218,143,12,57,480,154,944,7,730
12,65,67,39,390,32,39,318,47,86
45,51,59,21,53,43,25,7,42,27
310,45,72,53,798,304,354,79,45,44
52,76,45,26,27,968,86,16,62,85
790,208,390,36,62,83,93,16,574,150
99,7,920,860,12,404,31,560,37,32
9,62,7,43,17,77,73,368,66,82
11,51,97,26,83,426,92,39,66,2
23,93,85,660,85,774,77,77,927,868
7,554,760,104,48,202,45,75,51,55
716,752,37,95,267,91,5,956,444,529
96,99,17,99,62,7,394,580,604,89
678,476,97,234,1,608,19,69,676,51
410,89,414,81,130,491,6,238,79,43
5,288,910,204,948,19,644,21,295,11
6,595,904,67,51,703,430,95,408,89
11,495,844,13,417,570,9,429,16,939
430,270,49,72,65,66,338,994,167,76
47,211,87,39,1,570,85,134,967,12
553,63,35,63,98,402,664,85,458,834
3,62,508,7,1,72,88,45,496,43
750,222,96,31,278,184,36,7,210,55
653,51,35,37,393,2,49,884,418,379
75,338,51,21,29,95,790,846,720,71
728,930,95,1,910,5,804,5,284,128
423,6,58,36,37,321,22,26,16,27
218,530,93,55,89,71,828,75,628,67
66,622,440,91,73,790,710,59,83,968
129,632,170,67,613,608,43,71,730,910
36,92,950,138,23,95,460,62,189,73
65,943,62,554,46,318,13,540,90,53
967,654,46,69,26,769,82,89,15,87
46,59,22,840,66,35,684,57,254,230

21,586,51,19,984,156,23,748,760,65
339,892,13,13,327,65,35,246,71,178
83,3,34,624,788,200,980,882,343,550
708,542,53,72,86,51,700,524,577,948
132,900,72,51,91,150,22,110,154,148
99,75,21,544,110,11,52,840,201,2
6,663,22,20,89,10,93,964,924,73
501,398,3,2,279,5,288,80,91,132
620,628,57,79,2,874,36,497,846,22
350,866,57,86,83,178,968,52,399,628
869,26,710,37,81,89,6,82,82,56
96,66,46,13,934,49,394,72,194,408
5,541,88,93,36,398,508,89,66,16
71,466,7,95,464,41,69,130,488,695
82,39,95,53,37,200,87,56,268,71
304,855,22,564,47,26,26,370,569,2
494,2,25,61,674,638,61,59,62,690
630,86,198,24,15,650,75,25,571,338
268,958,95,898,56,585,99,83,21,600
462,940,96,464,228,93,72,734,89,287
174,62,51,73,42,838,82,515,232,91
25,47,12,56,65,734,70,48,209,71
267,290,31,844,12,570,13,69,65,848
72,780,27,96,97,17,69,274,616,36
554,236,47,7,47,134,76,62,824,55
374,471,478,504,496,754,604,923,330,22
97,6,2,16,14,958,53,480,482,93
57,641,72,75,51,96,83,47,403,32
624,7,96,45,97,148,91,3,69,26
22,45,42,2,75,76,96,67,688,2
2,224,83,69,41,660,81,89,93,27
214,458,66,72,384,59,76,538,15,840
65,63,77,33,92,32,35,832,970,49
13,8,77,75,51,95,56,63,578,47
33,62,928,292,2,340,278,911,818,770
464,53,888,55,76,31,389,40,864,36
35,37,69,95,22,648,334,14,198,42
73,594,95,32,814,45,45,515,634,254
42,29,15,83,55,176,35,46,60,296
262,598,67,644,80,999,3,727,79,374
19,780,400,588,37,86,23,583,518,42
56,1,108,83,43,720,560,81,674,25
96,218,6,69,107,534,158,56,5,938
9,938,274,76,298,9,518,571,47,175
63,93,49,94,42,26,79,50,718,926
419,810,23,363,519,339,86,751,7,86
47,75,55,554,3,800,6,13,85,65
99,45,69,73,864,95,199,924,19,948
214,3,718,56,278,1,363,86,1,22
56,114,13,53,56,19,82,88,99,543
674,704,418,670,554,282,5,67,63,466

491,49,67,154,956,911,77,635,2,49
53,12,79,481,218,26,624,954,13,580
130,608,37,91,78,743,1,950,45
41,718,36,30,534,418,452,359,759,88
29,499,55,974,93,56,108,257,93,171
13,92,63,714,9,84,890,16,930,967
748,5,7,6,327,894,33,629,448,21
9,19,7,535,75,3,27,928,21,7
864,27,73,61,25,75,876,16,92,22
248,11,86,944,872,996,252,2,800,334
93,107,254,441,930,744,97,177,498,931
694,800,9,36,6,539,35,79,130,860
710,7,630,475,903,552,2,45,97,974
17,36,77,843,328,22,76,368,39,71
35,850,96,93,87,56,972,96,594,864
344,76,17,17,576,629,780,640,56,65
43,196,520,86,92,31,6,593,174,569
89,718,83,8,790,285,780,62,378,313
519,2,85,845,931,731,42,365,32,33
65,59,2,671,26,364,854,526,570,630
33,654,95,41,42,27,584,17,724,59
42,26,918,6,242,356,75,644,818,168
964,12,97,178,634,21,3,586,47,382
804,89,194,21,610,168,79,96,87,266
482,46,96,969,629,128,924,812,19,2
468,13,9,120,73,7,92,99,93,418
224,22,7,29,57,33,949,65,92,898
200,56,12,31,296,185,272,91,77,37
734,911,27,310,59,33,87,872,73,79
920,85,59,72,888,49,12,79,538,947
462,444,828,935,518,894,13,591,22,920
23,93,87,490,32,63,870,393,52,23
63,634,39,83,12,72,131,69,984,87
86,99,52,110,183,704,232,674,384,47
804,99,83,81,174,99,77,708,7,623
114,1,750,49,284,492,11,61,6,449
429,52,62,482,826,147,338,911,30,984
35,55,21,264,5,35,92,128,65,27
9,52,66,51,7,47,670,83,76,7
79,37,2,46,480,608,990,53,47,19
35,518,71,69,32,87,96,240,52,310
86,73,52,31,83,544,16,15,21,774
224,7,83,680,554,310,96,844,29,61