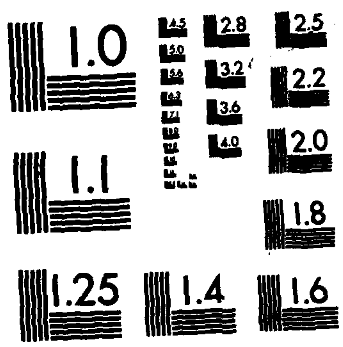
AD-A125 527 AN ENHANCED CONVERSION SCHEME FOR LEXICOGRAPHIC MULTIOBJECTIVE INTEGER PROGRAMS(U) NAVAL POSTGRADUATE SCHOOL MONTEREY CA J P IGNIZIO ET AL. DEC 82 1/1

UNCLASSIFIED NPS55-82-035 F/G 12/2 NL

END

FILMED

DTIC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

NPS55-82-035

NAVAL POSTGRADUATE SCHOOL

Monterey, California

AN ENHANCED CONVERSION SCHEME FOR LEXICOGRAPHIC, MULTIOBJECTIVE INTEGER PROGRAMS

by

James P. Ignizio

Lyn C. Thomas

December 1982

Prepared for:

Naval Postgraduate School
Monterey, Ca 93940

NAVAL POSTGRADUATE SCHOOL
Monterey, California

Rear Admiral J. J. Ekelund
Superintendent

David A. Schrady
Provost

James P. Ignizio, Professor
The Pennsylvania State University
University Park, Pennsylvania

Lyn C. Thomas, Professor
University of Manchester
Manchester, Great Britain

Reviewed by:

K. T. Marshall, Chairman
Department of Operations Research

Released by:

William M. Tolles
Dean of Research

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NPS55-82-035	2. GOVT ACCESSION NO. AD-A125527	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) AN ENHANCED CONVERSION SCHEME FOR LEXICOGRAPHIC, MULTIOBJECTIVE INTEGER PROGRAMS		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James P. Ignizio, Lyn C. Thomas		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School, Monterey, CA 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61152N; RR000-01-10 N0001482WR20043
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE December 1982
		13. NUMBER OF PAGES 19
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Energy Programming
Multiobjective Programming

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A number of approaches have been proposed (and several implemented) for the solution of lexicographic, multiobjective programming problems. These approaches may be divided into two classes. The first encompasses the development of algorithms specifically designed to deal directly with the initial model while the second attempts to transform, efficiently, the lexicographic, multiobjective model into an equivalent, single objective programming problem. This second approach would appear particularly attractive since it permits the use of conventional, readily available, mathematical programming software. In

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

this paper we address a particular form of the lexicographic, multiobjective model; specifically one in which all functions are linear and all variables integer. It is then shown how a recently developed scheme for the transformation of this model may be substantially improved. As a result, lexicographic, multiobjective integer linear programs may be easily converted into conventional linear integer programs wherein the magnitude of the objective function coefficients are minimized.

ABSTRACT

A number of approaches have been proposed (and several implemented) for the solution of lexicographic, multiobjective programming problems. These approaches may be divided into two classes. The first encompasses the development of algorithms specifically designed to deal directly with the initial model while the second attempts to transform, efficiently, the lexicographic, multiobjective model into an equivalent, single objective programming problem. This second approach would appear particularly attractive since it permits the use of conventional, readily available, mathematical programming software. In this paper we address a particular form of the lexicographic, multiobjective model; specifically one in which all functions are linear and all variables integer. It is then shown how a recently developed scheme for the transformation of this model may be substantially improved. As a result, lexicographic, multiobjective integer linear programs may be easily converted into conventional linear integer programs wherein the magnitude of the objective function coefficients are minimized.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the support and assistance provided them, during the conduct of this research, through National Research Council Senior Research Associateships, under the sponsorship of the Naval Postgraduate School. All work was performed at the Naval Postgraduate School in Monterey, California.

1. INTRODUCTION

In a recent paper, Sherali [3] introduced a weighting factor scheme for the conversion of certain lexicographic, multiobjective programming problems into equivalent single objective models. The specific problem addressed was one in which:

(1) all functions (i.e., objectives and constraints) are linear

(2) all variables are restricted to integer values

(3) the multiple objectives are preemptively ordered

That is, we seek the solution of a Lexicographic, Multiobjective Integer Linear Programming, or LMOILP, problem. The LMOILP problem is given as:

$$\text{maximize } \{C^*x: A^*x=b,\ 0<x<u \text{ and integer}\} \qquad \text{(P)}$$

where:

C is a (K,n) matrix whose rows represent K preemptively ordered objective functions

A is a (m,n) constraint matrix

x, b, and u are n, m, and n column vectors, respectively

The LMOILP problem of (P) is also known as the lexicographic vectormax programming problem. Note that (P) should not be confused with a multiobjective model having somewhat similar form; specifically the lexicographic goal programming problem. For comparison, the lexicographic linear goal programming model is shown as (G), below:

$$\text{satisfy } \{C^*x>b': \ A^*x=b,\ x \text{ continuous or integer}\} \qquad \text{(G)}$$

where:

C^*x are the set of original, preemptively ordered objectives

b' represents the goal levels, or target values for each of the original objectives. Thus, $C^*x \geq b'$ denotes a vector of goals which are to be preemptively satisfied,

Although a conversion scheme for (G) is also possible, our interest in this paper shall be restricted to the LMOILP model as shown in (P). Further details with regard to model (G) may be found in the references [1,2].

Aggregation of Objectives

Except for the multiple, preemptively ordered objectives, (P) would be a linear integer programming model. Consequently, one approach to the solution of (P) is to first transform it into an equivalent single objective model via the aggregation of all objectives into a single, equivalent objective function. We shall denote the transformed problem as (P'), where the general form of (P') is given as:

$$\text{maximize } \{wC)^*x: \ A^*x=b, \ 0 \leq x \leq u \text{ and integer}\} \qquad (P')$$

To accomplish this transformation, we must determine w, a K order column vector of weights so as to insure that the solution to (P') is the same as that which would be obtained by solving (P). The determination of w is made more difficult by recalling that the objectives are preemptively ordered. However, if such a weighting may be found, we may then use conventional (i.e., single objective) algorithms and readily available software to solve the new problem. Sherali [3] has devised algorithms which accomplish such a transformation. That is, he shows how the preemptively ordered multiple objectives may be aggregated into a single, equivalent objective for which the solution satisfies the preemptive ordering of the original set of objectives. A drawback of his approach is that the magnitude of the coefficients of the aggregate objective function may be enormous, thus limiting the practical implementation of the scheme.

Purpose and Overview

The primary purpose of this paper is to present an approach to the LMOILP problem which provides an "optimal" aggregate objective function. That is, the function is optimal in the sense that the magnitude of the largest coefficient is minimized. As a consequence, the aggregation of objectives in the LMOILP problem may, in may instances, be transformed from simply an academic proposal into a practical, implementable end result. In this paper we present this method, demonstrate it on a numerical example, and compare it with Sherali's approach.

2. BACKGROUND

The most promising approach that has been proposed, thus far, for the aggregation of objectives in the LMOILP problem is the scheme developed, as mentioned earlier, by Sherali. In his paper, Sherali presents two algorithms where one dominates the other in the sense of always obtaining smaller maximum aggregate objective function coefficients. That algorithm, denoted in [3] as algorithm 2, is given below:

<u>The Sherali Algorithm</u>

Step 1: When required in the algorithm, the upper bound of objective z(k), denoted as UB[z(k)], is found by:

$$UB[z(k)] = \sum_{j=1}^{n} u(j)|c(j,k)| \tag{1}$$

where:

UB[j(k)] is the UB of the k-th objective

u(j) is the UB of the j-th variable

c(j,k) is the coefficient of the j-th variable in the k-th objective

Note that in many cases the problem structure may permit the easy derivation of a far tighter upper bound than that given by (1) and, for such cases, the tighter upper bound may be used without need to change the remaining steps of the algorithm.

Step 2: Initialize:

Set F(K) = c(K)*x

Set w(K) = 1

Set p (a counter) = K-1

Step 3: Compute:

$$w(p) = 1 + UB[F(p+1)] \qquad (2)$$

$$F(p) = F(p+1) + w(p)*z(p) \qquad (3)$$

Step 4: Check for termination: If $p = 1$, stop. The aggregate objective is $F(1)$. However, if p exceeds 1, then set $p = p-1$ and return to step 3.

<u>Example</u>

In order to demonstrate the Sherali algorithm, we consider the following numerical example. Later, we shall compare the results obtained here with those of the enhanced method. The LMOILP problem under consideration is given as:

$$\text{maximize } z(1) = x(1) + x(2) + x(3)$$

$$\text{maximize } z(2) = 200*x(1) + 150*x(2) + 250*x(3)$$

$$\text{maximize } z(3) = 180*x(1) + 155*x(2) + 240*x(3)$$

subject to:

$$A*x = b$$

$$x(j) \le 10 \text{ and integer for all } j$$

Since the form of the constraint set (i.e., A*x=b) plays no part in the derivation of the weights, we use the general form in the above example. Now (again recall that the three objectives above have been <u>preemptively ordered</u>), applying the Sherali algorithm gives:

$$F(3) = 180*x(1) + 155*x(2) + 240*x(3)$$

$$w(3) = 1$$

Thus, w(2) = 1 + UB[F(3)]

= 1 + 1800 + 1550 + 2400 = 5751

And, F(2) = F(3) + w(2)*z(2)

= 180*x(3) + 155*x(2) + 240*x(3)

+ 1150200*x(1) + 862650*x(2) + 1437750*x(3)

Or, F(2) = 1150380*x(1) + 862805*x(2) + 1437990*x(3)

Finally, w(1) = 1 + UB[F(2)] = 34511751

F(1) = F(2) + w(1)*z(1)

And thus:

F(1) = 35,662,131*x(1) + 31,374,556*x(2) + 35,949,741*x(3)

Limitations

While the Sherali algorithm, as described above, will accomplish objective function aggregation in LMOILP, its primary drawback is made obvious by the small numerical example. That is, the coefficients of the aggregated objective, F(1), range in size from 31,374,556 up to 35,949,741. In many real world problems, the size of such coefficients will be far larger. Large enough, in fact, to create an integer overflow on the digital computer as well as other practical difficulties in algorithm implementation. As such, the first question that arises, in concern to this or any other approach, is in regard to the possibility for development of a (quick and easy) method for minimizing the magnitude of the largest coefficient in the aggregate objective. In the next section, we show that there does exist a simple, efficient approach to accomplish this goal.

3. PROBLEM STATEMENT

We may replace our original statement of the LMOILP problem, i.e., (P), with the following equivalent formulation:

$$\text{maximize } \{C*x - y*S*x): \; A*x=b, \; 0<x<u \text{ and integer}\} \quad \text{(PE)} \quad \text{where:}$$

$$S*x = \begin{bmatrix} 0 \\ c(1)*x \\ c(2)*x \\ \cdot \\ \cdot \\ \cdot \\ c((K-1)*x \end{bmatrix} \tag{4}$$

and y is a m order column weighting vector.

The replacement of (P) by (PE) is made possible by the preemptive ordering of the multiple objectives. That is, in the LMOILP, the optimization of a higher level objective preempts that of any lower objective. Consequently, relative to any lower level objective, the higher level objective is a constant. As such, S*x, as given in (4) provides for the development of just one possible equivalent formulation of (P). However, for our purposes, it is the one we shall use to develop the enhanced aggregate objective scheme since it provides a very simple and speedy approach for conversion.

With reference to (PE), our goal is to determine the vector of weights, y , so as to minimize the magnitude of the largest coefficient in the aggregate objective, F(1), when the Sherali algorithm is employed. To accomplish this, we need only deal with two objectives since any number of objectives may be dealt with by combining two at a time (i.e., the two lowest ranked objectives are first aggregated. Next, this aggregate objective is combined with the third lowest ranked objective, and so on.).

Given two objectives, z(1) and z(2) (where they have been preemptively ordered) wherein:

$$z(1) = c(1)*x$$

$$z(2) = c(2)*x$$

we express these, using the form (PE), as:

$$z(1) = c(1)*x$$

$$z'(2) = c(2)*x - y(2)*[c(2)*x]$$

Now, if the Sherali algorithm is applied to z(1) and z'(2), the resulting aggregate objective is given as:

$$F'(1) = w(1)*c(1)*x + \{c(2)*x - y(2)*[c(1)*x]\}$$

but, from (2), we may replace w(1) by 1 + UB[z'(2)]. Thus F'(1) may be rewritten as:

$$F'(1) = \{1+UB[z'(2)]\}*[c(1)*x] + \{c(2)*x-y(2)*[c(1)*x]\}$$

Further, from (1), we may state F(1) as:

$$F'(1) = \{1 + \sum_{j=1}^{n} u(j)|c(j,2)-y(2)*c(j,1)|\}*[c(1)*x] + \{c(2)*x-y(2)*[c(1)*x]\} \quad (5)$$

Using (5), the coefficient of each variable, x(j), in the aggregate objective function, F'(1), may be written as a function of y(2) as shown below:

$$c'(j) = \{1 + \sum_{j=1}^{n} u(j)|c(j,2)-y(2)*c(j,1)|\}*[c(j,1)] + [c(j,2)-y(2)*c(j,1)] \quad (6)$$

We thus wish to find y(2) so as to minimize the maximum value of c'(j) for any j . This may be stated as:

$$\text{minmax } \{c'(j)\} \text{ over all } j \tag{7}$$

Some Observations and Resulting Simplifications

To solve (7), we may immediately note that:

(a) Relation (6) and, consequently, (7) are obviously convex.

(b) The single unknown is y(2)

(c) The values of y(2) are restricted to those which will maintain <u>integer</u> coefficients in the resultant, aggregate objective function. See [3].

As a result, we may easily solve (7) for the optimal value of y(2) by simply employing a discrete search scheme (e.g., Fibonacci search). This makes the construction of the optimal form of the aggregate objective both simple and practical. In the section to follow, we demonstrate this concept via the example previously solved by the Sherali algorithm.

4. EXAMPLE

In Section 2, we used the Sherali algorithm to construct an aggregate objective function for a LMOILP with three objectives. Further, we noted the substantial increase in the magnitude of the aggregate objective coefficients, relative to the coefficients of the original objective functions. Here, we shall utilize the results of Section 3, specifically relationship (7), so as to develop the optimal aggregate objective function for this same example.

Recall that the example of Section 2 was:

maximize z(1) = x(1) + x(2) + x(3)

maximize z(2) = 200*x(1) + 150*x(2) + 250*x(3)

maximize z(3) = 180*x(1) + 155*x(2) + 240*x(3)

subject to:

A*x=b

x(j) < 10 and integer for all j

We work first with the two lowest ranked objectives, z(2) and z(3). From (6) we note that:

c'(j) =

{1+10[|180-200*y(3)|+|155-150*y(3)|+|240-250*y(3)|]}

*c(j,2) + [c(j,3)-y(3)*c(j,2)]

Listed below are the values for the aggregate objective function formed from z(2) and z(3) for several values of y(3):

y(3)	c'(1)	c'(2)	c'(3)
0	1,150,380	862,805	1,437,990
1	70,180	52,555	87,740
2	1,250,420	937,795	1,563,010

Since (7) is a convex function, we note that the optimal value for y(3) is y(3) = 1. Note also that, for y(3) = 0, we obtain the same coefficients as found earlier (when combining these two objectives) via the Sherali algorithm.

Next, we wish to combine z(1) with the composite objective formed from z(2) and z(3) when y(3) = 1. Denoting that composite objective as F'(2), we have:

$$z(1) = x(1) + x(2) + x(3)$$

$$F'(2) = 70180*x(1) + 52555*x(2) + 87740*x(3)$$

and c'(j) is then given by:

$$c'(j) = \{1+10*[|70180-y(2)|+|52555-y(2)|+|87740-y(2)|]\} *c(j,1) + [c(j,2)-c(j,1)*y(2)]$$

Using any discrete search algorithm, we determine the final, aggregate objective function for the three original objectives. This function, denoted as F'(1) is given as:

$$F'(1) = 351,851*x(1) + 334,226*x(2) + 369,411*x(3)$$

Recall that the Sherali algorithm produced, for the same example, the aggregate objective shown below:

$$F(1) = 35,662,131*x(1) + 31,374,556*x(2) + 35,949,741*x(3)$$

When the above aggregate objective is compared to that found, in Section 2, via the Sherali algorithm, the difference is obviously substantial. That is, the results from the Sherali algorithm are two orders of magnitude greater than that determined by the enhanced scheme. Given a larger number of objective functions, the differences can be even more pronounced.

5. CONCLUSIONS AND SUMMARY

In an earlier paper, Sherali presented an algorithm for the conversion of the LMOILP problem into an equivalent conventional linear integer program. We have demonstrated, in this paper, that such an aggregation scheme may be substantially enhanced via a simple and practical method. Although not illustrated, the enhanced approach (or the Sherali algorithm) may be even further improved by replacing the naive upper bound of (1) by a tighter upper bound. Such improved upper bounds may be found, for many integer programming models, via relatively simple and straight forward means. It should be recognized, however, that although the procedure proposed herein will provide for enhanced aggregate objective function development, it is optimal only for the form of the matrix S as shown in (4). Other approaches, using more complex forms of S can be developed to provide for even further improvement. Work in this area is, in fact, still in progress. However, as of this time, it is not yet clear that the additional improvement (i.e., reduction in the size of objective function coefficients) is worth the sometimes considerable additional effort and complexity.

6. REFERENCES

[1] J. P. Ignizio, Linear Programming in Single and Multiple Objective Systems (Prentice-Hall, Englewood Cliffs, New Jersey, 1982).

[2] J. P. Ignizio, Goal Programming and Extensions (D.C. Heath, Lexington, MA, 1976).

[3] H. D. Sherali, Equivalent weights for lexicographic multiobjective programs: Characterizations and computations, European Journal of Operational Research. 11 (1982) 367-379.

DISTRIBUTION LIST

	NO. OF COPIES
Library, Code 0142 Naval Postgraduate School Monterey, CA 93940	4
Dean of Research Code 012A Naval Postgraduate School Monterey, CA 93940	1
Library, Code 55 Naval Postgraduate School Monterey, CA 93940	1
Professor J. P. Ignizio Code 55 Naval Postgraduate School Monterey, CA 93940	45