

REDUCTION OF DIMENSIONALITY IN DYNAMIC PROGRAMMING-BASED SOLUTION METHODS FOR NONLINEAR INTEGER PROGRAMMING

BALASUBRAMANIAN RAM

Department of Industrial Engineering
North Carolina A & T State University
Greensboro, NC 27411

and

A.J.G. BABU

Industrial Systems Department
University of South Florida
Tampa, FL 33620

(Received June 5, 1986)

ABSTRACT. This paper suggests a method of formulating any nonlinear integer programming problem, with any number of constraints, as an equivalent single constraint problem, thus reducing the dimensionality of the associated dynamic programming problem.

KEY WORDS AND PHRASES. Dynamic Programming, Integer Programming.
1980 AMS SUBJECT CLASSIFICATION CODE. 90C39, 90C10.

1. INTRODUCTION.

There are numerous application areas in which it is possible to model the situation under study by formulating a discrete-variable nonlinear optimization model. In general, these situations can be represented by a nonlinear objective function with nonlinear constraints. Examples of such situations include facilities location, investment analysis and transportation problems [1]. The problem can be stated as follows:

$$\text{Maximize } f(x_1, x_2, \dots, x_n) \quad (1.1)$$

$$\text{Subject to } g_i(x_1, x_2, \dots, x_n) \leq b_i \quad \text{for } i = 1, 2, \dots, m \quad (1.2)$$

$$x_i \geq 0 \quad \text{for } i = 1, 2, \dots, n \quad (1.3)$$

$$x_i \text{ is integer for } i = 1, 2, \dots, n \quad (1.4)$$

There are four broad classes of solution methods available for problems (1.1)-(1.4): dynamic programming-based methods, branch and bound methods, linear approximation methods and hybrid methods.

Dynamic Programming-based methods require that either the objective function (1.1) or the constraint set (1.2) be separable in the decision variables. Direct dynamic programming methods require that each of the constraints in constraint set (1.2) be separable while the hypersurface algorithm of Cooper and Cooper [1-3] require that the objective function (1.1) be separable.

Wei Shih [4] has developed a branch-and-bound method for the separable programming problem with the additional restriction that the component functions of the objective function satisfy the law of diminishing returns. This solution procedure was proven to yield the optimal solution; the proof was given by Mjelde [5]. Shih's procedure was reported to be faster than both dynamic programming and exhaustive search methods.

Attempts to linearize the nonlinear integer programming problem involve a radical increase in the number of variables and constraints. Methods to achieve more economical linear representations of 0-1 polynomials have been undertaken by Glover and Woolsey [6]. In spite of the developments, this method seems to be not very encouraging except for problems containing certain special structures.

Aust [7] suggested a dynamic programming-branch and bound hybrid approach for nonlinear integer programming problems with separable objective function and separable constraints. The method involves partitioning the m constraints into p disjoint set of constraints.

2. DYNAMIC PROGRAMMING METHODS.

The direct dynamic programming-based methods for problems (1.1)-(1.4), for the single-constraint case give rise to one-dimensional dynamic programming problems (problems requiring one state variable per stage). Problems with multiple constraints give rise to multi-dimensional dynamic-programming problems. The storage requirement for tables increases very rapidly with increase in the number of constraints. This is referred to as the curse of dimensionality.

A number of methods have been proposed to reduce the problem of dimensionality including lagrange multiplier method, successive approximation method and polynomial approximation method. A detailed discussion of these methods can be found in Cooper [8].

Cooper and Cooper [1] developed the hypersurface algorithm to reduce the dimensionality problem. The basic notion behind the algorithm is to search objective function hypersurfaces to see if it contains any feasible points. The lattice points for each hypersurface are found by solving a one-dimensional dynamic programming problem. The method requires a separable objective function but does not impose any restriction on the form of the constraints.

This paper suggests a method to reduce the storage requirement, which is a serious problem in dynamic programming-based methods, especially for problems with a large number of constraints.

3. REDUCTION OF DIMENSIONALITY BY CONSTRAINT AGGREGATION.

A nonlinear integer programming problem with several constraints can be reduced to a problem with a single constraint if the original constraint set can be reduced to an equivalent single constraint. The following theorem provides a method of aggregating both linear and nonlinear constraints with mild restrictions on the type of nonlinear constraints. (See Babu and Ram [9] for a similar result in linear integer programming).

THEOREM: The system of equations (3.1) is equivalent to the equation (3.2).

$$\sum_{j=1}^n a_{ij} x_j^{r_{ij}} = b_i \quad \text{for } i = 1, 2, \dots, m \quad (3.1)$$

$$\sum_{i=1}^m (\ell_n p_i) \left(\sum_{j=1}^n a_{ij} x_j^{r_{ij}} \right) = \sum_{i=1}^m (\ell_n p_i) (b_i) \quad (3.2)$$

where a_{ij} , b_i , r_{ij} , are integer constants x_j are integer variables and p_i are distinct prime numbers other than unity.

PROOF: Consider a solution (x_1, x_2, \dots, x_n) to the system (3.1). Multiplying each equation in (3.1) by constants $\ell_n p_i$ and adding we get equation (3.2).

We now prove that (3.2) implies (3.1). Assume (3.2) holds. Let B_i be any real numbers satisfying

$$\sum_{j=1}^n a_{ij} x_j^{r_{ij}} = B_i \quad (3.3)$$

Since a_{ij} , x_j , r_{ij} are integers, B_i are integers. Rewriting (3.2), we have

$$\sum_{i=1}^m B_i p_i = \sum_{i=1}^m b_i p_i \quad (3.4)$$

Suppose $B_i < b_i$ for $i \in I_1$ where $I_1 \subseteq I = \{1, 2, \dots, n\}$ and $B_i > b_i$ for $i \in I_2$ where $I_2 \subseteq I$. I_1, I_2 are clearly disjoint. After cancelling like terms on both sides of (3.4) we have,

$$\sum_{i \in I_2} p_i (B_i - b_i) = \sum_{i \in I_1} p_i (b_i - B_i) \quad (3.5)$$

Since $(B_i - b_i)$, $i \in I_2$ and $(b_i - B_i)$, $i \in I_1$ are integers and p_i are prime numbers other than unity, equation (3.5) cannot be satisfied. This implies that $B_i = b_i$ for all $i \in I$. Hence any solution (x_1, x_2, \dots, x_n) to (3.2) will also be a solution to (3.1). Thus, the system of equations (3.1) and the equation (3.2) are equivalent.

The above theorem gives a method of reducing a system of nonlinear constraints satisfying the following conditions, to a system consisting of a single constraint.

- (i) Each of the constraints must be separable in the variables.
- (ii) The exponents of the variables must be integers.
- (iii) The constraint coefficients and the right hand side constants in the constraints must be integers.

4. COMPUTATIONAL CONSIDERATIONS.

The use of the above theorem for problem (1.1) - (1.4) satisfying the conditions stated in Section 3, will result in a one-dimensional dynamic programming problem in place of a multi-dimensional problem. When the number constraints in the original problem is large the storage requirement for a computer implementation of the method is dramatically reduced. It must be mentioned here that in a computer implementation, the one-dimensional approach will involve additional table search as the state variables take on non-integer values and hence cannot be used directly as array indices. The irrational nature of the constraint multipliers $\ell_n p_i$, causes errors in evaluation of the state variable value. This problem can be circumvented by specifying a carefully chosen tolerance value.

REFERENCES

1. COOPER, L. and COOPER, M.W. Nonlinear Integer Programming, Computers and Mathematics with Applications, 1 (1975), 215-222.
2. COOPER, M. W. An Improved Algorithm for Nonlinear Integer Programming, Technical Report IEOR 77005, Southern Methodist University, Dallas, Texas (1977).
3. COOPER, M. W. The Use of Dynamic Programming Methodology for the Solution of a Class of Nonlinear Programming Problems, Naval Research Logistics Quarterly, 27 (1980), 89-95.
4. SHIH, WEI. A Branch and Bound Procedure for a Class of Discrete Resource Allocation Problems with Several Constraints, Operations Research Quarterly, 28 (1977), 439-451.
5. MJELDE, K. M. The Optimality of an Incremental Solution of a Problem Related to Distribution of Effort, Operations Research Quarterly, 26 (1975), 867-870.
6. GLOVER, F. and WOOLSEY, E. Converting the 0-1 Polynomial Programming Problem to a 0-1 Linear Program, Operations Research, 22 (1974), 180-182.
7. AUST, R. J. Dynamic Programming Branch and Bound Algorithm for Pure Integer Programming, Computers and Operations Research, 3 (1976), 27-28.
8. COOPER, L. and COOPER, M. W. Introduction to Dynamic Programming, Pergamon Press, Inc., New York (1981).
9. BABU, A. J. G. and RAM, B. On the Aggregation of Constraints in Integer Programs, Abstracts of American Mathematical Society, 2 (1981), 483.