International Journal of Applied Mathematics
& Statistical Sciences (IJAMSS)
Vol.1, Issue 1 Aug 2012 29-39
© IASET

SELECTION OF CONSTRAINTS IN NON-NEGATIVE LINEAR PROGRAMMING PROBLEMS USING INTERCEPT MATRIXS

PAULRAJ

Associate Professor of Mathematics, Madras Institute of Technology Campus, Anna University Chennai, Chromepet, Chennai - 600 044, Tamil Nadu, India

ABSTRACT

Linear Programming is one of the most important techniques adopted in modeling and solving different practical optimization problems that arise in industry, commerce and management. While formulating a linear programming model, systems analysis and researchers often tend to include all possible constraints. Although some of them may not be binding at the optimal solution. The Accuracy of number of equations and variable needed to model real-life situations are significantly large, and the solution process could be time-consuming even solving with computers. Pre-processing is an important technique in the practice of linear programming problem. Since, the reduction of large scale problem can save significant amount of computational effort during the solution of a problem. Many researchers have proposed algorithms for selecting necessary constraints in linear programming models. This paper proposes a heuristic approach for selecting constraints, a prior to the start of the solution process using Intercept Matrix. Rather some of the earlier methods developed for selecting constraints are explained and an improved method is also suggested using Intercept Matrix. The developed algorithm is implemented and the computational results are also presented. It shows that the proposed method reflects a significant decrease in the computational effort and is one of the best alternative to select the necessary constraints prior to solve non-negative linear programming problem.

KEYWORDS: Constraint Selection, Cosine Simplex Algorithm, Intercept Matrix, Largest Summation Rule.

INTRODUCTION

Linear Programming Problem (LPP) represents a mathematical model for solving numerous practical problems such as the optimal allocation of resources.

Linear Programming consist of two important ingredients

1. The objective function.
2. Constraint

both of which are linear.

The general form of LPP

Optimise

$Z = C^T x$

Subject to the constraints,

$Ax \leq b$

$x \geq 0$

where x represents the vector of variables (to be determined), while C and b are vectors of (known) coefficients and A is a (known) matrix of coefficients.If the technological coefficients a_{ij}'s ≥ 0, then the linear programming problems is said to be non-negative linear programming problems.

In solving a LP problem we tend to include all possible constraints it will increase the number of iteration and computational effort. To reduce the number of iterations and computational effort we use a new technique called constraint selection technique which minimizes the number of constraints used in solving LPP.

DEFINITIONS

Binding Constraint - Binding constraint (Scarce resource) is one which passes through the optimum solution point.

Redundant Constraint - A redundant constraint is one that can be removed without causing a change in the feasible region.

Nonbinding Constraints –Non binding constraints (abundant resource) is one which does not pass through optimum solution point.

Constraint Selection Technique - The constraint selection technique begins by solving a sequence of sub-problems using only a few of the original constraints [1-3] and [7-8]. If the solution obtained to this sub-problem satisfies the remaining constraints, it is optimal for the original LP. Otherwise, additional constraints must be incorporated in a larger sub-problem until a solution is obtained which satisfies all the original constraints.

Consider the following LPP,

Max $Z = C^T x$

Subject to the constraints,

$$Ax \leq b \qquad \text{……. (1)}$$

$$x \geq 0$$

The algorithm finds the optimal solution to (1) by solving a sequence of sub-problems of the form

Max $Z = C^T x$

Subject to the constraints,

$$A^K x \leq b^k \qquad \text{……. (2)}$$

$x \geq 0$

where, A^k is a sub-matrix of A. b^k consists of right-hand side values for the constraints corresponding to the rows in A^k.

The rest of the paper is organized as follows. Section 2 describes the already developed algorithm for selecting constraints in solving LPP. Section 3 proposes a new method for selecting constraints in LPP. The computational results are presented in section 4, followed by conclusion in section 5.

SOME PRECEDING METHODS FOR SELECTING CONSTRAINTS

In this section, some of the earlier methods developed for selecting constraints are explained and an improved method is suggested making use of matrix of intercept. The developed algorithm is implemented using computer programming language C.

Danny C.Myers & Wei Shih et al (1988) proposes a new algorithm in constraint selection techniques which achieves considerable reductions in the constraints used in solving Linear Programming problem. This technique works well for a class of linear programs considered. This technique is simple but effective heuristic approach which obtains solutions to a class of linear programming problems with the use of only a small percentage of the original number of constraints. This method is guaranteed to converge in m/2 iterations where m is the number of constraints and is not dependent upon the problem structure.

Cosine Simplex Algorithm (Corley H.W. (2006))

Corley H.W, Jay Rosenberger, Wei-Chang Yeh, and T.K.Sung, et al (2006) use the cosine criterion at each iteration, and the current relaxed problem involves only a fraction of the original constraints. In small linear programming problems the cosine simplex algorithm solves them more efficiently than the standard simplex method. In addition, it can prevent cycling.

The cosine simplex algorithm is summarized as follows. For a given problem it begins by solving a relaxed problem consisting of the original objective function subject to a single constraint yielding a nonempty bounded feasible region. At each subsequent iterations of the algorithm, the most parallel constraint to the objective function among those constraints violated by the solution to the current relaxed problem is appended to it. When no constraints are violated, the solution of the current relaxed problem is optimal to the original problem.

Define row vector i of the matrix A by a_i so that constraint i of (1) becomes

$$a_i x \leq b_i,\ i = 1, 2, \ldots m. \tag{3}$$

Define $Cos\theta_i = \frac{a_i^T c}{\|a_i^T\| \|c\|}$ as the cosine of the angle θ_i between the normal vectors a_i for constraint i and c for the objective function.

Let $Cos\theta_r = \max_{i \in I} Cos\theta_i$. i = 1,2…m.

Assume that (1) has a nonempty feasible region with a constraint r in (3) for which $a_{rj} > 0$, j = 1, 2, . . n and $b_r > 0$. These assumptions guarantee that (1) has an optimal solution.

We call a constraint operative if it is a part of the current relaxed problem and inoperative otherwise.

The initial relaxed problem

Max $z = C^T x$

subject to the constraint

$$a_r\, x \leq b_r \tag{4}$$

$$x \geq 0.$$

ALGORITHM

1. Compute $Cos\theta_i = \frac{a_i^T c}{\|a_i^T\| \|c\|}$,i = 1,2,…m, $i \neq r$, and order the constraints according to decreasing Cosθ_i, where ties are broken arbitrarily. Let $Cos\theta_r = \max_{i \in I} Cos\theta_i$.

   i = 1,2…m.
2. Solve (4), which becomes relaxed problem (1), to obtain x^1. Set k = 1.
3. Check the inoperative constraints in decreasing order of Cosθ_i. Take the first one violated by x^k and go to Step 4. If none is found, stop since x^k satisfies problem (1).
4. Set k = k + 1. Append the violated constraint to the final tableau of relaxed problem k to obtain relaxed k+1 Apply the dual simplex algorithm to obtain a solution x^k Go to Step 2.

Largest Summation Rule (LSR) (Danny.C.Myers (1988))

The steps of the algorithm is as follows:

1. Set k= 0. Express each constraint in the following form by dividing each constraint by the corresponding right-hand side value.

   $$\sum_{j=1}^{n} \bar{a}_{ij}\, x_j \leq 1$$ where $\bar{a}_{ij} = a_{ij}/b_i$.
2. At the initial iteration we set $V_0=\{1,2,. . . m\}$. For each $i \in V_k$,

   Let

$$S_i = \sum_{j=1}^{n} |a_{ij}|$$ and calculate

$$S_{m1} = \max_{i \in V_k}\{S_i\}$$

$$S_{m2} = \max_{i \in V_k, i \neq m1}\{S_i\}$$

Select the constraints with indices m_1 and m_2 to add to the sub-problem at this iteration.Let x_j^k be solution obtained by solving the sub-problem at k iteration.

3. Define $V_k = \{ i : \sum_{j=1}^{n} \bar{a}_{ij} x_j^k \geq 1\}$ i.e., V_k is the index set of constraints which are violated at the current solution x_j^k. If $V_k = \varnothing$ stop. If V_k consists of only one element, add the constraint corresponding to this index and go to step 4.

4. Set k = k+1. Then, using the constraints identified in step 2 together with those constraints included in earlier iterations, solve (2) to obtain x^k and return to step 2.

INTERCEPT MATRIX METHOD

The initial relaxation of Intercept matrix includes those constraints which have an intercept on each coordinate axis that is closest to the origin.

The steps of the algorithm is as follows:

1. Construct a matrix of intercepts of all the decision variables formed by each of the resource constraints along the respective co-ordinate axis.

$$\theta_{ji} = \frac{b_i}{a_{ij}}, \qquad a_{ij} > 0$$

2. Identify the smallest of the intercept in each row of the matrix of intercept θ_{ji}.

3. Let $\beta_j = \min_{i \in I}\{\theta_{ji}\}$, for j Є J. Let k be the set containing the column number of corresponding β_j elements.

4. Solve the LPP with constraints corresponding to the elements of k. If the solution obtained from the above subproblem satisfies all the constraints, Stop. Otherwise go to Step 5.

5. Calculate θ_{ji} for which the constraints are violated at the current solution and calculate β_j and k. Let k_1 be the set containing the column number of corresponding β_j. Set $k = k + k_1$ and return to Step 4.

ILLUSTRATION

Selecting the constraints by Cosine Simplex Algorithm, Largest Summation Rule and proposed method is illustrated using following numerical example.

Consider the problem,

Maximize $Z = 3x_1 + 4x_2$

Subject to the constraints,

$$x_1 + 3x_2 \leq 15 \qquad (3.1)$$

$$2x_1 + x_2 \leq 10 \qquad (3.2)$$

$$2x_1 + 3x_2 \leq 18 \qquad (3.3)$$

$$x_1 + x_2 \leq 7 \qquad (3.4)$$

$$4x_1 + 5x_2 \leq 40 \qquad (3.5)$$

$$x_1, x_2 \geq 0.$$

SOLUTION

COSINE SIMPLEX ALGORITHM

ITERATION 1

Step 1: The values for Cos θ_i are 0.9886, 0.8944, 0.9984, 0.9899 and 0.9995 respectively, since the associated Cos θ_i for (3.5) is maximum.

Step 2: The first relaxed problem is to be solved by simplex method. We obtain x_1= 0.00, x_2=8.00, and Z=32.

Step 3: Constraints (3.3),(3.4)and (3.1) are inoperative constraint and here Cos θ_i for (3.3) is a maximum.

Step 4: Add the (3.3) constraint to the existing problem and update the simplex table. We obtain x_1=9.00 ,x_2=0.00, and Z=27.

ITERATION 2

Step 3: Constraint (3.4) is inoperative.

Step 4: Add the (3.4) constraint to the existing problem and we obtain x_1=3.00, x_2=4.00, and Z=25.

Number of constraint selected in Cosine Simplex Algorithm =3.

LARGEST SUMMATION RULE

ITERATION 1

Step 1: Set V_0={1,2,3,4,5}

Maximize Z= $3x_1$+$4x_2$.

Subject to the constraints

$$\frac{1}{15}x_1 + \frac{3}{15}x_2 \leq 1 \quad (3.6)$$

$$\frac{1}{5}x_1 + \frac{1}{10}x_2 \leq 1 \quad (3.7)$$

$$\frac{1}{9}x_1 + \frac{1}{6}x_2 \leq 1 \quad (3.8)$$

$$\frac{1}{7}x_1 + \frac{1}{7}x_2 < 1 \quad (3.9)$$

$$\frac{1}{10}x_1 + \frac{1}{8}x_2 \leq 1 \quad (3.10)$$

$$x_1, x_2 \geq 0.$$

Step 2: Compute

$$S_i = \sum_{j-1}^{n} |\bar{a}_{ij}|$$

The values for S_i are 0.266, 0.3, 0.277, 0.2857 and 0.255.

$S_{m1}= S_2= 0.3$ and $S_{m2}= S_4= 0.2857$

Select the constraint (3.2) and (3.4) constraint and solve by using simplex method. We obtain $x_1 = 0.00$,x_2=7.00, and Z=28.

Step 3: Constraints (3.1) and (3.3) are violated.

Step 4: Add (3.1) and (3.3) constraints to the existing problem

update the simplex table. We obtain x_1=3.00 ,x_2=4.00, and Z=25.

Number of constraint selected in Largest Summation Rule = 4.

INTERCEPT MATRIX METHOD

ITERATION 1

Step 1: Let θ_{ji} be

Decision variables	1	2	3	4	5	βj	k
x1	15	5	9	7	10	5	2
x2	5	10	6	7	8	5	1

Solving the constraints (3.2) and (3.1), we obtain x_1= 3.00,

x_2= 4.00, and Z = 25.

Number of constraints selected in this method = 2.

COMPUTATIONAL RESULTS

The efficiency of the algorithm is tested by solving LPP using Simplex method and various methods in constraint selection techniques. The following table 1 shows a comparison of number of constraints selected in different methods. Table 2 provides a comparison of computational efforts. The tables 1 and 2 shows that the proposed method (Intercept matrix method) is efficient to select the constraints in a given LP with minimum computational effort. Figure 1 shows number of constraints used in solving linear programming problems and figure 2 shows the amount of computational effort reducing the intercept method over the previous methods.

Here the problems have the canonical form with $c_j = 1$ for $j = 1,2,\dots n$, $b_i = 100$ for $i = 1,2,\dots m$ and a_{ij} is generated uniformly within the interval (0,100) for $i = 1,2,\dots m,$ $j = 1,2,\dots n$.

Table1. Comparison of No. of Constraints Selected in Solving LPP

SI. No.	Size of LPP		No. of constraints selected in solving LPP		
	m	n	Method1	Method2	Method3
1	5	2	2	4	2
2	3	2	1	2	2
3	4	3	3	2	2
4	3	2	2	2	1
5	3	2	3	3	3
6	3	2	3	2	2
7	4	3	3	2	1
8	4	3	4	2	2
9	4	5	4	2	2
10	5	2	2	2	2
11	5	4	2	2	3
12	3	3	2	3	2
13	7	10	4	2	2
14	16	6	4	4	4
15	20	5	4	2	3
16	25	6	3	2	2
17	30	3	4	2	1
18	40	2	1	2	1
19	45	3	3	3	2
20	50	5	4	2	1

m- No. of constraints, n- No. of variables, Method1 - Cosine Simplex Algorithm, Method2 – Largest Summation Rule, Method3 – Intercept Matrix Method.

Table2. Comparison of Computational effort in Solving LPP

SI. No.	Size of LPP		No. of multiplication/divisions taken to solve LPP			
	M	n	Method1	Method2	Method3	Method4
1	5	2	175	153	176	60
2	3	2	32	44	36	33
3	4	3	98	246	74	69
4	3	2	62	90	55	34
5	3	2	91	205	164	143
6	3	2	62	158	55	52
7	4	3	51	187	53	39
8	4	3	98	461	74	69
9	4	5	177	942	132	125
10	5	2	118	114	65	60
11	5	4	141	210	98	93
12	3	3	71	128	139	89
13	7	10	481	1405	285	278
14	16	6	774	1173	439	550
15	20	5	591	1188	253	247
16	25	6	888	1501	362	337
17	30	3	1117	1286	235	195
18	40	2	1845	414	211	172
19	45	3	4690	1093	411	317
20	50	5	2961	2250	583	521

m- No. of constraints, n- No. of variables, Method1 - Simplex method, Method2 – Cosine Simplex Algorithm,

Method3 – Largest Summation Rule, Method4 – Intercept Matrix Method.

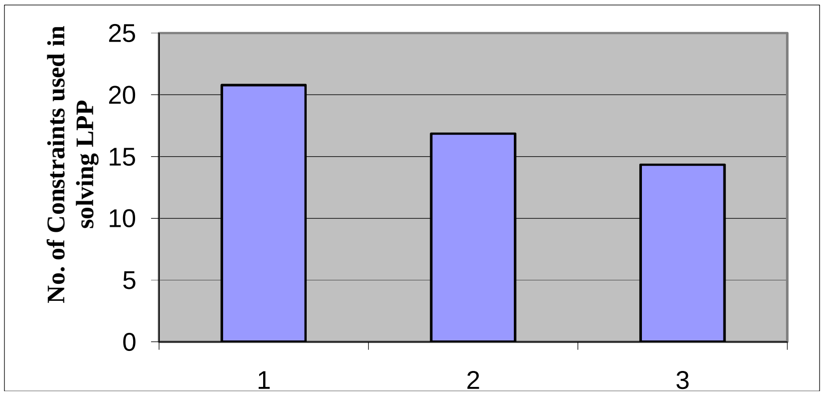

Figure 1: Comparison of No. of Constraints Selected in Solving LPP

1 - Cosine Simplex Algorithm, 2 – Largest Summation Rule,

3 – Intercept Matrix Method.

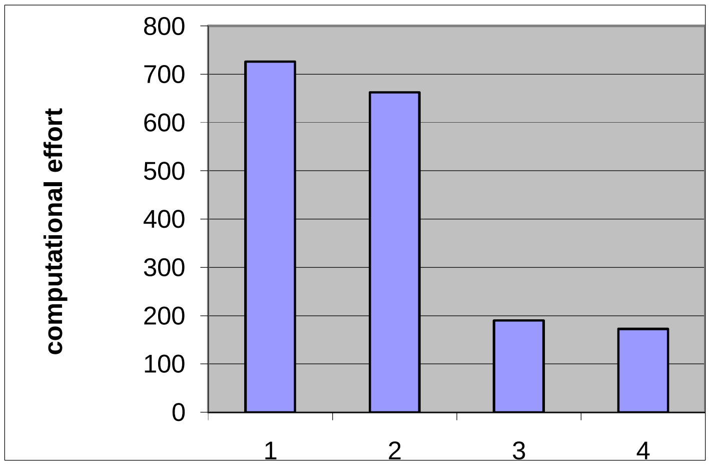

Figure 2: Comparison of Average Computational effort in Solving LPP

1 - Simplex method, 2 – Cosine Simplex Algorithm,

3 – Largest Summation Rule, 4 – Intercept Matrix Method.

In Linear programming problem, if (0,0,…0)ε R^n is a feasible point of a given problem, then we can apply the intercept method to select the constraint.

If $\{x_1,x_2,\ldots.x_n\}$, $x_i \neq 0$ (atleast one of the decision variable is not equal to zero). In that case LSR has to perform n/2 iteration to identify n constraint and in Cosine simplex method we have to perform n iteration, whereas in case of intercept method, we can identify n constraint in the first iteration.

CONCLUSIONS

In this paper, we have presented a Intercept matrix method to reduce the number of constraints considered in solving a given LPP. A large number of randomly generated test problems indicate that the Intercept matrix method have less computational efforts than the existing algorithms.

REFERENCES

1. Corley H.W, Jay Rosenberger, Wei-Chang Yeh, and T.K.Sung,(2006),'The Cosine Simplex algorithm'.The International Journal of Advanced Manufacturing Technology; Vol.27 pp (1047-1050).

2. Danny C.Myers and Wei Shih(1988), 'A constraint selection technique for a class of linear programs'. Operation Research Letters, Vol. 7, pp. 191-195.

3. Danny C.Myers,(1992), 'A Dual Simplex Implementation of a constraint selection Algorithm for linear programming'. The Journal of the Operational Research Society, Vol.43. (feb.,1992),pp.177-180.

4. Dantzig G.B,(1951) 'Maximization of a linear function of variables subject to linear inequalities,' in T.C. Koopmans,editor, Activity Analysis of Production and Allocation, JohnWiley, New York, pp.339-347.

5. Richard J.Caron, Arnon Boneh, and Shahar Boneh, 'Advances in sensitivity Analysis and parametric programming,' edited by Harvey J.Greenberg.

6. aha,H.A(2006) Operations Research, An Introduction, 8th Edition, Prentic Hall of India, New Delhi.

7. Wei Li, Haohao Li. (2011) On Simplex Method with Most-Obtuse-Angle Rule and Cosine

8. Rule, Applied Mathematics and Computation, (217), 7867-7873.

9. Yoshihiro Yamamoto.S (2010), "A New Method for Solving a linear programming problem", 49th IEE Conference on Decision and control, December 15 – 17.