ISSN 2250-3153

An Efficient Method for Easy Computation by Using θ - Matrix by Considering the Integer Values for Solving Integer Linear Fractional Programming Problems

V.Seerengasamy[*], Dr.K.Jeyaraman[**]

[*] Professor of Mathematics, PSNA College of Engineering and Technology, Dindigul-624 622, Tamil Nadu, India.
[**] Dean of Science and Humanities, PSNA College of Engineering and Technology, Dindigul-624622, Tamil Nadu, India

Abstract- To minimize the computational effort needed in solving a Linear Fractional programming problem a new approach has been proposed. Here we use θ matrix for finding the solution of the integer linear fractional programming problems.

Index Terms- Integer Linear Fractional Programming Problems, θ matrix and Promising variables.

I. INTRODUCTION

To solve Integer Linear Fractional Programming Problems with reduced computational effort, a new method of approach has been proposed. In this method, among the decision variables, the variables which can enter into the basis are identified and ordered based on the maximum contribution to the objective function. The ordered decision variables one by one are allowed to enter into the basis by checking whether it is still giving an improved solution.

II. GENERAL INTEGER LINEAR FRACTIONAL PROGRAMMING PROBLEMS IN MATRIX FORM

The general Integer Linear Fractional Programming Problems is given by

$$\text{Extremize } Z = \frac{C^T X + c_0}{D^T X + d_0}$$

Subject to

$$AX\ (\leq\ =\ \geq)\ P^0$$

$$X \geq 0 \text{ and are integer}$$

Where

$$A^{mxn} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & . & . & . & a_{1n} \\ a_{21} & a_{22} & a_{23} & . & . & . & a_{2n} \\ a_{31} & a_{32} & a_{33} & . & . & . & a_{3n} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ a_{m1} & a_{m2} & a_{m3} & . & . & . & a_{mn} \end{bmatrix} \quad X^{nx1} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ . \\ . \\ . \\ x_n \end{bmatrix} \quad P^0 = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ . \\ . \\ . \\ b_m \end{bmatrix}$$

Let the columns corresponding to the matrix A be denoted by P^1, P^2, P^3……P^n where

$$P^1 = \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ . \\ . \\ . \\ a_{m1} \end{bmatrix} \quad P^2 = \begin{bmatrix} a_{12} \\ a_{22} \\ a_{32} \\ . \\ . \\ . \\ a_{m2} \end{bmatrix} \quad P^3 = \begin{bmatrix} a_{13} \\ a_{23} \\ a_{33} \\ . \\ . \\ . \\ a_{m3} \end{bmatrix} \ldots\ldots\ldots\ldots\ldots P^n = \begin{bmatrix} a_{1n} \\ a_{2n} \\ a_{3n} \\ . \\ . \\ . \\ a_{mn} \end{bmatrix}$$

$C^T = (c^1, c^2, c^3 \ldots\ldots c^n)$, $D^T = (d^1, d^2, d^3 \ldots\ldots d^n)$ and c^0, d^0 are scalars.

III. Approach

In this new approach to solve Integer Linear Fractional Programming Problems, three phases are included and those phases are given below.

Phase I:

Promising decision variables are identified to enter in the basis and those promising variables are ordered based on the contribution to the objection function.

Phase II:

The arranged promising variables are allowed to enter into the basis in the arranged order after checking whether the newly entering variables will improve the objective function of the problem, keeping the feasibility.

Phase III:

Finding the improved solution vector.

The above three phases are repeated till the optimum solution reached. In this method, the promising variables are identified and arranged based on the maximum contribution to the objective function by considering the **integer values** of the θ matrix entries. The step by step procedure is as given below.

Step 1: Let iteration = 0
Step 2: Perform phase I
Step 3: Perform phase II
Step 4: If the set J is empty then, Perform phase III
Step 5: stop.

Phase I - Ordering of Promising variables

Step 1. Using the intercepts of the decision variables along the respective axes with respect to the chosen basis a matrix is called θ matrix is to be constructed. A typical intercept for the j^{th} variable, x^j due to the i^{th} the resource, b^i is $\left\{\frac{b_i}{a_{ij}}\right\}$ $a^{ij} > 0$

The expanded form of θ matrix is

$$\begin{array}{c} \\ x_1 \\ x_2 \\ \\ x_j \\ \\ x_n \end{array} \begin{array}{c} \begin{array}{ccccccc} S^1 & S^2 & . & . & S^i & . & . & S^m \end{array} \\ \begin{bmatrix} \frac{b_1}{a_{11}} & \frac{b_2}{a_{21}} & . & . & \frac{b_i}{a_{i1}} & . & . & \frac{b_m}{a_{m1}} \\ \frac{b_1}{a_{12}} & \frac{b_2}{a_{22}} & . & . & \frac{b_i}{a_{i2}} & . & . & \frac{b_m}{a_{m2}} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ \frac{b_1}{a_{1j}} & \frac{b_2}{a_{2j}} & . & . & \frac{b_i}{a_{ij}} & . & . & \frac{b_m}{a_{mj}} \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ \frac{b_1}{a_{1n}} & \frac{b_2}{a_{2n}} & . & . & \frac{b_i}{a_{in}} & . & . & \frac{b_m}{a_{mn}} \end{bmatrix} \end{array}$$

Each row of the θ matrix consists of m number of intercepts of the decision variable along their respective axes and each column consists of intercepts formed by the decision number of promising variables in each of the m constraints.

Step 2. The minimum integer intercepts and its position in each row of θ matrix is found out. If there are more one minimum integer intercept then one of them is selected arbitrarily. Multiply the minimum intercept of the variable corresponding to a row with the corresponding contribution coefficient in the objective function both in the numerator and denominator and the objective function value $\left(\frac{c_j x_j + c_0}{d_j x_j + d_0}\right)$ is calculated.

Step 3. Let λ = 0. J is a set consisting of the subscript of the promising variables.

Step 4. Select the variable whose $\left(\frac{c_j x_j + c_0}{d_j x_j + d_0}\right)$ value is the largest integer. If the same largest integer $\left(\frac{c_j x_j + c_0}{d_j x_j + d_0}\right)$ value occurs, for more than one variable then the variable that has maximum contribution including the fractional value is taken as the promising variable. If that is also same then select any one arbitrarily.

Step 5. Let it be x_R.Then x_R is selected as the promising variable.

Step 6. Increment λ by 1.The subscript of the variable x_R is stored as the l^{th} element in set J.

Step 7. The row corresponding to the variable x_R as well as the other rows whose minimum occurs in the column at which the minimum for x_R occurs are deleted.

Step 8. Step 4 to 7 is repeated till either all the rows or all the columns are deleted.

Step 9. The set of variables collected in Steps 4 to 7 are the ordered promising variables.

Let J= {Subscripts of the promising variables arranged in the descending order $\left(\frac{c_j x_j + c_0}{d_j x_j + d_0}\right)$ value}.

Let λ be the total number of elements in the set J.

Phase II – Arranged variables are allowed to enter into the basis

The arranged promising variables are allowed to enter into the basis one by one based on the entering criteria. The step by step procedure is given below.

Step 1. Let $k=1$, X^B is the solution vector and flag (=0) is the flag vector

$$\text{flag} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{bmatrix}_{n\times 1} ; \qquad P^{0-old} = P^0$$

Step 2. Iteration is incremented by 1.

Step 3. The k^{th} element in the set J is selected and let it be j. Then the entering variable is x_j.

Step 4. Computation of x_j value.

The value with which x_j can enter into the basis is computed by using the following formula, θ_k =

$$\min \left\{ \text{int} \left\{ \frac{(P_{0-old})_i}{(P_j)_i} \right\} ; \ (P^j)^i > 0 \right\} \quad i = 1,2,3….m$$

Step 5. If θ_k =1 and k=1 then the value of α =1 else the value of α is chosen between 0 to 1 (Let α =0.5)

Compute S = int ($\alpha\ \theta_k$)

ε_k = int (1- α) θ_k

Step 6. S is added to the j th element of the vector X^B and 1 is added to flag ^{j}

Step 7. P^0 vector is modified using the relation

$$(P^{0-new})^i = (P^{0-old})^i - (P^j)^i\ S \quad i = 1,2,….m$$

Step 8. (P^{0-old}) is replaced by (P^{0-new})

If k=1 or S $\le$ 1 go to step 16

Step 9. Check whether k th element is still promising among the remaining list of (λ-k) promising variables in set J using the following steps.

Step 10. Let r =1

Step 11. Select the (k + r) th element in this set J. Let it be q. Then the variable corresponding position x^q .

Step 12. Find θ_q using the formula

$$\theta_q = \min \left\{ \text{int} \left\{ \frac{(P_{0-old})_i}{(P_j)_i} \right\} ; \ (P^j)^i > 0 \right\}$$

$$\varepsilon_q = \left(\frac{c_q\theta_q + c_0}{d_q\theta_q + d_0}\right)$$

Step 13. If $\varepsilon_k < \varepsilon_q$ goto step 15
Step 14. Increment r by one
If $r \leq (\lambda - k)$ then goto step 11. Else goto step 16.

Step 15. k is replaced by k + r and ε_k is replaced by ε_q

If $k < \lambda$ goto step 10

Step 16. If $flag^k \leq 1$ goto step 3.
Else goto Perform Phase I.

Phase III – Determination of new (improved) solution vector to the Integer Linear Fractional Programming Problems

Except for the most promising variable in the solution set obtained in phase II the values of remaining variables are set to zero. Taking this as starting solution, phase I and II are performed until improved solution is obtained. If there is no improvement the next promising variable value along with the most promising variable also is retained and the remaining basic variables made to zero. Phase III is repeated until the basic variables list exhausted.

IV. Algorithm

Stage I. The basic variables are arranged according to the descending order of their contribution to the objective function

Step 1. $\lambda = 0$, k =0, n^1 is is the number of basic variables having nonzero values in the solution
Step 2. X is the solution vector obtained in phase II.

Step 3. If λ^{th} element in X, ie $X^{\lambda} > 0$, then

Multiply $\left(\frac{c_\lambda x_\lambda + c_0}{d_\lambda x_\lambda + d_0}\right)$, let it be stored as k^{th} row 0^{th} column element of array W and λ is stored as k^{th} row 1^{th} column element of array W. k is incremented by one.

Step 4. λ is incremented by one

Step 5. If $\lambda < n^1$ then goto step 3.
Step 6. The array W is sorted in the descending order based on the 0^{th} column values of W

Stage II. Finding the solution by assigning all the variable values except one in the basis to zero level.
Step 7. k = 0
Step 8. $\lambda = 0$
Step 9. i = 0
Step 10. If $i \neq k$ then

$J = W^{i1}$

$X^{j} = 0$

Step 11. i is incremented by one

Step 12. If $i < n^1$ then goto step 10

Step 13. Now P^c is the current resource vector or (RHS) and corresponding objective function value Z^1 is calculated.

Stage III. Find the new solution
Step 14. Use phase I and phase II and find the new solution X which is stored as $Y(\lambda)$ and the corresponding objective function value Z^2 is stored as $V(\lambda)$.

Step 15. λ is incremented by one

Step 16. If $\lambda < n^1$ then goto step 9

Step 17. Find the largest of $V(\lambda)$ and its position pos, where $(0 \leq \lambda < n^1)$, Let it be stored in Z^3 **Step 18.** If $Z^3 > Z$, then Replace X by Y (pos)

goto step 7. else if $k < n^1$ then increment k by 1.
goto step 8.

V. Numerical Examples

Solve the following Integer linear fractional Programming Problem.
Maximize

$$Z = \frac{4x_1 + 17x_2 + 24x_3 + 23x_4 + 19x_5 + 13x_6 + 2}{2x_1 + 3x_2 + 4x_3 + 6x_{4+}3x_5 + 3x_6 + 50}$$

Subject to the constraints
$4x_1 + 2x_2 + 5x_3 + 7x_4 + 7x_5 + 7x_6 \leq 325$
$4x_1 + 2x_2 + 4x_3 + 9x_4 + 9x_5 + x_6 \leq 400$
$9x_1 + 3x_2 + x_3 + 4x_4 + 7x_5 + 9x_6 \leq 425$
$5x_1 + 3x_2 + x_3 + 2x_4 + 4x_5 + 6x_6 \leq 425$

Where $x_1, x_2, x_3, x_4, x_5, x_6 \geq 0$ and all are integers
Solution

$$A = \begin{pmatrix} 4 & 2 & 5 & 7 & 7 & 7 \\ 4 & 2 & 4 & 9 & 9 & 1 \\ 9 & 3 & 1 & 4 & 7 & 9 \\ 5 & 3 & 1 & 2 & 4 & 6 \end{pmatrix} \quad P_0 = \begin{pmatrix} 325 \\ 400 \\ 425 \\ 425 \end{pmatrix}, P_1 = \begin{pmatrix} 4 \\ 4 \\ 9 \\ 5 \end{pmatrix}, P_2 = \begin{pmatrix} 2 \\ 2 \\ 3 \\ 3 \end{pmatrix}, P_3 = \begin{pmatrix} 5 \\ 4 \\ 1 \\ 1 \end{pmatrix}$$

$X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix}$, $P_4 = \begin{pmatrix} 7 \\ 9 \\ 4 \\ 2 \end{pmatrix}$, $P_5 = \begin{pmatrix} 7 \\ 9 \\ 7 \\ 4 \end{pmatrix}$, $P_6 = \begin{pmatrix} 7 \\ 1 \\ 9 \\ 6 \end{pmatrix}$

$C^T = (4,17,24,23,19,13)$, $D^T = (2,3,4,6,3,5,50)$,$C_0 = 2$, $D_0 = 50$.

Phase - I

To find θ Matrix

$$\begin{array}{ccc} d_j & c_j & x_j \\ 2 & 4 & x_1 \\ 3 & 17 & x_2 \\ 4 & 24 & x_3 \\ 6 & 23 & x_4 \\ 3 & 19 & x_5 \\ \theta = 3 & 13 & x_6 \end{array} \begin{bmatrix} 81 & 100 & 47 & 85 \\ 162 & 200 & 141 & 141 \\ 65 & 100 & 425 & 425 \\ 46 & 44 & 106 & 212 \\ 46 & 44 & 60 & 106 \\ 46 & 400 & 47 & 70 \end{bmatrix} \begin{array}{c} \dfrac{c_j x_j + 2}{d_j x_j + 50} \\ 1.319 \\ 5.072 \\ 5.039 \\ 3.229 \\ 4.604 \\ 3.191 \end{array}$$

Arrangement of promising variables

J = {2, 3, 5}

Phase – II

$X_2 \rightarrow$ promising variable

θ = minimum $\left\{ int\left(\dfrac{\frac{325}{2,400}}{2}, \dfrac{\frac{425}{3,425}}{3} \right) \right\}$ = minimum (162,200,141,141) = 141

∴ S = 70

$Z = \dfrac{1190 + 2}{210 + 50} = \dfrac{1192}{260}$ = 4.585

$(P_0 - new)_1 = 325 - 70 \times 2 = 185$

$(P_0 - new)_2 = 400 - 70 \times 2 = 260$

$(P_0 - new)_3 = 425 - 70 \times 3 = 215$

$(P_0 - new)_4 = 425 - 70 \times 3 = 215$

$X_2 \rightarrow$ promising vari able

θ = minimum $\left\{ int\left(\dfrac{\frac{185}{2,260}}{2}, \dfrac{\frac{215}{3,215}}{3} \right) \right\}$ = minimum (92, 130, 71, 71) = 71

∴ S = 35

$Z = \dfrac{17 \times 35 + 1192}{3 \times 35 + 260} = \dfrac{595 + 1192}{105 + 260} = \dfrac{1787}{365}$ = 4.896

$(P_0 - new)_1 = 115$,$(P_0 - new)_2 = 190$,$(P_0 - new)_3 = 110$,$(P_0 - new)_4 = 110$

Repeating this procedures Phase I and Phase II we get the solution as

The current solution is $x_2 = 130$, $x_3 = 13$

Maximum $Z = \dfrac{130 \times 17 + 24 \times 13 + 2}{3 \times 130 + 4 \times 13 + 50} = \dfrac{2524}{492}$ = 5.130

Phase - III

x_2 is retained 130 and remaining variables are set to zero ,that is $x_3 = 0$.

Now $P_0 = \begin{pmatrix} 65 \\ 140 \\ 35 \\ 35 \end{pmatrix}$

Following similarly we get the final solution

The OptimaPhase - I

To find θ Matrix

$$\frac{c_j x_j + 2}{d_j x_j + 50}$$

d_j	c_j	x_j					
2	4	x_1					
3	17	x_2	16	35	3	7	0.25
4	24	x_3	32	70	11	11	2.28
6	23	x_4	13	35	35	35	3.08
3	19	x_5	9	15	8	17	2.01
θ = 3	13	x_6	9	15	5	8	1.49
			9	140	3	5	0.69

Arrangement of promising variables

J = { 3, 2 }

Phase – II

$X_3 \rightarrow$ promising variable

θ = minimum $\left\{ int \left(\frac{\frac{65}{5,140}}{4}, \frac{\frac{35}{1,35}}{1} \right) \right\}$ = minimum (13, 35, 35, 35) = 13

∴ S = 6

Z = $\frac{6 \times 24 + 17 \times 130 + 2}{6 \times 4 + 3 \times 130 + 50} = \frac{144 + 2212}{24 + 440} \frac{2356}{464}$ = 5.08

$(P_0 - new)_1 = 35, (P_0 - new)_2 = 116, (P_0 - new)_3 = 29, (P_0 - new)_4 = 29$

Following similarly we get

$X_3 \rightarrow$ promising variable

θ = minimum $\left\{ int \left(\frac{\frac{5}{5,92}}{4}, \frac{\frac{23}{1,23}}{1} \right) \right\}$ = minimum (1, 23, 23, 23) = 1

∴ S = 1

Z = $\frac{1 \times 24 + 2500}{1 \times 4 + 488} = \frac{2524}{492}$ = 5.13

$(P_0 - new)_1 = 0, (P_0 - new)_2 = 88, (P_0 - new)_3 = 22, (P_0 - new)_4 = 22$

Phase - I

To find θ Matrix

$$\frac{c_j x_j + 2524}{d_j x_j + 492}$$

d_j c_j x_j

$$\theta = \begin{matrix} 2 & 4 & x_1 \\ 3 & 17 & x_2 \\ 4 & 24 & x_3 \\ 6 & 23 & x_4 \\ 3 & 19 & x_5 \\ 3 & 13 & x_6 \end{matrix} \begin{bmatrix} 0 & 22 & 2 & 4 \\ 0 & 44 & 7 & 7 \\ 0 & 22 & 22 & 22 \\ 0 & 9 & 5 & 11 \\ 0 & 9 & 3 & 5 \\ 0 & 88 & 2 & 3 \end{bmatrix} \begin{matrix} - \\ - \\ - \\ - \\ - \\ - \end{matrix}$$

Arrangement of promising variables

J = { }

Here the Phase – III will not improve the solution so, the optimal solution is
$x_2 = 130$, $x_3 = 13$

$$\text{Maximize } Z = \frac{130 \times 17 + 24 \times 13 + 2}{3 \times 130 + 4 \times 13 + 50} = \frac{2524}{492} = 5.13$$

VI. CONCLUSION

In this a new approach to solve Integer Linear Fractional Programming problem has been discussed. The above algorithm rendered best optimal solution. In Future this method can be applied in Mixed Integer Linear Fractional programming problem to get a better optimal solution.

REFERENCES

[1] C.Audet, P.Hansen, B.Jaumard and G.Savard, Journal of Optimization theory and Application. Vol.93, No.2, (1997) 273-300.

[2] A.I.Barros, J.B.G. Frenk, S.Schaible and S. Zhang, A new algorithm for generalized fractional programs Mathematical Programming 72 (1996), 2, 147-175.

[3] A.Charles, W.W. Cooper An explicit general solution in linear fractional programming, Vol.20 449-467 September 1973.

[4] Erik B.Bajalinov Linear Fractional programming theory, methods, Applications and Siftware.

[5] Fengquiyou & Ignacio Grossmam. Solving Mixed-Integer Linear Fractional Programming Problems with Dinkelbach’s Algorithm and MINLP methods

[6] Hamdy A.Taha, “Operations Research- An Introduction”, Seventh Edition, Prentice-Hall of India Private Limited, 2004

[7] H.Ishii, T. Ibaraki and H.Mine, Fractional knapsack problems, Mathematical Programming 13 (1976), 3, 255-271.

[8] Kanti Swarup, Gupta P.K.Manmohan, “Operations Research”, Sultan Chand and Sons,2010.

[9] G.KarthiKeyan,”Design of a new computer oriented algorithm to solve linear programming problems”, Ph.D., thesis, Alagappa University, India, May 2011.

[10] Pant. J.C. Operation Research, Introduction to optimization, 7th edition 2008.

[11] Er. Prem Kumar Gupta and Dr. D.S. Hira, Problems in Operation Research (Principles and solutions) S. Chand & company, Ram Nagar, New Delhi.

[12] Stancu–Minasian, I.M. Fractional programming theory, methods and applications series, Mathematics and its Application Vol. 409 (1997)432p.

[13] Suresh Chandra, M. Chandra Mohan, A note on integer linear fractional programming, Volume 27 (1980)171-174.

[14] L.Vicente, G.Savard and S.Judics, Journal of Optimization Theory and Applications, 89, No.3 (1996) 597-614.

[15] Wukfred Candler and Robert Townsley, Computers and Operations Research, 9(1982) 59-76.

AUTHORS

First Author – V.Seerengasamy, Professor of Mathematics, PSNA College of Engineering and Technology, Dindigul-624 622, Tamil Nadu, India.

Second Author – Dr.K.Jeyaraman, Dean of Science and Humanities, PSNA College of Engineering and Technology, Dindigul-624622, Tamil Nadu, India.