

Gen. Math. Notes, Vol. 4, No. 2, June 2011, pp. 1-9 ISSN 2219-7184; Copyright © ICSRS Publication, 2011 www.i-csrs.org Available free online at http://www.geman.in

An Integer Solution of Fractional Programming Problem

S.C. Sharma¹ and Abha Bansal²

¹Department of Mathematics, University of Rajasthan, Jaipur-302055, India. E-mail: sureshchand26@gmail.com ²Department of Mathematics, Dungurpur Engineering College & Technology, Dungurpur (Raj.)- 314001, India E-mail: neelamchikusinghal2002@gmail.com

(Received: 2-6-11/Accepted: 23-06-11)

Abstract

The present paper describes a new method for solving the problem in which the objective function is a fractional function, and where the constraint functions are in the form of linear inequalities. The proposed method is based mainly upon simplex method, which is very easy to understand and apply. This can be illustrated with the help of numerical examples.

Keywords: Fractional programming, Simplex method.

1 Introduction

Fractional programming problem is that in which the objective function is the ratio of numerator and denominator. These types of problems have attracted considerable research and interest. Since these are useful in production planning, financial and corporate planning, health care and hospital planning etc.

Algorithms for solving linear fractional programming problems are well known by many authors [1, 2, 3]. Charnes-Copper [1] replaces a linear fractional program by one equivalent linear program, in which one extra constraint and one extra variable has been added. The usual simplex algorithm computes the optimum solution. Isbell-Marlow and Martos [3, 4] find the solution of a sequence of linear programs. Wagner-Yuan [8] showed that, when the feasible set is bounded. Chdhas-Rachael [2] solves a system of linear of inequalities in which

the objective function is expressed as one of the constraint along with the given set of linear constraints of the problem. Resently Tantawy [7] has suggested a feasible direction approach and a duality approach to solve a linear fractional programming problem.

Here our aim is to find the integer solution of fractional programming problems (i.e. objective function is the ratio of numerator and denominator of linear functions). For it, we use simplex method and branch and bound method. These methods are very easy to understand and apply. Preliminaries are given in the next section. The steps of the proposed algorithm are presented in section 3. Numerical examples have been worked out in section 4 of the paper and finally in section 5 we present the references.

2 **Preliminaries**

A maximization integer fractional programming problem may be stated as:

Max.
$$Z = \frac{c^{1}x + \alpha}{d^{1}x + \beta} \qquad \dots (2.1)$$

s.t.
$$Ax \le b$$

 $x \ge 0$ and an integer

where x, c, d are $n \times 1$ vectors, b is an $m \times 1$ vector, c^{1} , d^{1} denote transpose of vectors, c and d is an $m \times n$ matrix and α , β are scalars constants. It is assumed that the constraint. S = {x : Ax $\leq b$, x ≥ 0 and integer} is non- empty and bounded.

3 Algorithm

- Step 1: first, observes whether all the right side constants of the constraints are non-negative. If not, it can be changed into positive value on multiplying both the sides of the constraints by -1.
- Step 2: next converts the inequality constraints to equations by introducing the non-negative slack or surplus variables. The coefficients of slack or surplus variables are always taken zero in the objective function.

Step 3: constructs the simplex table by using the following notations.

Let x_B be the initial basic feasible solution of the given problem such that

$$B x_{B} = b$$
$$x_{B} = b B^{-1}$$
$$b = b$$

where $\mathbf{B} = (b_1, b_2, ..., b_r, b_s, ..., b_m)$. Further suppose that

$$Z_1 = c_B^1 x_B + \alpha$$
$$Z_2 = d_B^1 x_B + \beta$$

where c_B^1 and d_B^1 are the vectors having their components as the coefficients associated with the basic variables in the numerator and denominator of the objective function respectively.

Step 4: Now, compute the 'net evaluation' Δ_j for each variable x_j (column vector x_j) by the formula

$$\Delta_{j} = Z_{2}(c_{j} - Z_{j}^{(1)}) - Z_{1}(d_{j} - Z_{j}^{(2)})$$

Step 5: If all $\Delta_i \leq 0$, the optimal solution is obtained.

Step 6: If optimal solution is an integer solution then we get the required solution otherwise use Branch and Bound method.

4 Numerical Example

Example1. Find the integer solution of following fractional programming problem:-

max.
$$Z = \frac{x_1 + 2x_2}{2x_1 - x_2 + 2}$$

s.t. $-x_1 + 2x_2 \le 2$
 $x_1 + x_2 \le 4$

$$x_1, x_2 \ge 0$$
 and an integer.

Solution:

After adding slack variables x_3 and x_4 , the constraints become

$$-x_{1} + 2x_{2} + x_{3} = 2$$

$$x_{1} + x_{2} + x_{4} = 4$$

$$x_{1}, x_{2}, x_{3}, x_{4} \ge 0$$

			c _j	1	2	0	0	Min.
								ratio
			dj	2	-1	0	0	$\begin{array}{c c} x_{ij} \\ y_{ij} \end{array}$
d _B	c _B	x _B	В	X ₁	X ₂	X ₃	X 4	
0	0	X3	2	-1	2	1	0	2/2
0	0	X4	4	1	1	0	1	4/1
	$Z_1 = 0$	Z = 0						
	Z ₂ = 2							
			$c_{j} - Z_{j}^{(1)}$	1	2	0	0	
			$\overline{d_j - Z_j^{(2)}}$	2	-1	0	0	
			$\overline{\Delta}_{j}$	2	4	0	0	
					$\mathbf{\Lambda}$	\checkmark		

Where

$$Z_1 = c_B^{(1)} x_B + \alpha$$
$$= \begin{pmatrix} 0\\0 \end{pmatrix} \begin{pmatrix} 2\\4 \end{pmatrix} + 0 = 0$$

$$Z_2 = d_B^{(1)} x_B + \beta$$
$$= \begin{pmatrix} 0 \\ 0 \end{pmatrix} \begin{pmatrix} 2 \\ 4 \end{pmatrix} + 2 = 2$$

$$\Delta_{j} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

Since $Z_{1} = 0$ and $Z_{2} = 2$, therefore $\Delta_{j} = 2(c_{j} - Z_{j}^{(1)})$ as shown in table 1.

			c _j	1	2	0	0
			dj	2	-1	0	0
d _B	c _B	XB	В	x ₁	x ₂	X3	X ₄
-1	2	x ₂	1	-1/2	1	1/2	0
0	0	X 4	3	3/2	0	-1/2	1
	$Z_1 = 2$	Z = 2					
	$Z_2 = 1$						
			$c_j - Z_j^1$	2	0	-1	0
			$d_j - Z_j^2$	3/2	0	1/2	0
			Δ_j	-1	0	-2	0

$$Z_1 = c_B^{(2)} x_B + \alpha$$
$$= \begin{pmatrix} 2 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 3 \end{pmatrix} + 0 = 2$$

$$Z_2 = d_B^{(2)} x_B + \beta$$
$$= \begin{pmatrix} -1 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 3 \end{pmatrix} + 2 = 1$$

$$\Delta_{j} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$\Delta_{1} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$= 1(2) - 2(3/2) = -1$$

$$\Delta_{2} = 0$$

$$\Delta_{3} = 1(-1) - 2(1/2) = -2$$

$$\Delta_{4} = 0$$

Here $x_1 = 0$, $x_2 = 1$ and $Z = Z_1 / Z_2 = 2$. Since all $\Delta_j \le 0$, therefore the current solution is the optimal basic feasible solution.

Example2. Find the integer solution of following fractional programming problem:-

max. Z =
$$\frac{2x_1 + x_2}{3x_1 + x_2 + 6}$$

s.t.

 $5x_1 + 3x_2 \le 6$ $7x_1 + x_2 \le 6$

 $x_1, x_2 \ge 0$ and an integer.

Solution: After adding slack variables x_3 and x_4 , the constraints become

 $5x_1 + 3x_2 + x_3 = 6$ $7x_1 + x_2 + x_4 = 6$ $x_1, x_2, x_3, x_4 \ge 0$

	Table-3									
			c _j	2	1	0	0	Min.		
								ratio		
			dj	3	1	0	0	$\begin{array}{c} x_{ij} \\ y_{ij} \end{array}$		
d _B	c _B	XB	В	X ₁	X2	X3	X 4			
0	0	X3	6	5	3	1	0	6/5		
0	0	X4	6	7	1	0	1	6/7		
	$Z_1 = 0$	Z = 0								
	$Z_2 = 6$									

		$c_{j} - Z_{j}^{(1)}$	2	1	0	0	
		$d_{j} - Z_{j}^{(2)}$	3	1	0	0	
		Δ_{j}	12	6	0	0	
 1	1		\wedge			\mathbf{V}	1

Where

$$Z_1 = c_B^{(1)} x_B + \alpha$$
$$= \begin{pmatrix} 0\\0 \end{pmatrix} \begin{pmatrix} 6\\6 \end{pmatrix} + 0 = 0$$

$$Z_2 = d_B^{(1)} x_B + \beta$$
$$= \begin{pmatrix} 0\\ 0 \end{pmatrix} \begin{pmatrix} 6\\ 6 \end{pmatrix} + 6 = 6$$

$$\Delta_{j} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

Since $Z_{1} = 0$ and $Z_{2} = 6$, therefore $\Delta_{j} = 6(c_{j} - Z_{j}^{(1)})$ as shown in table 1.

			c _j	2	1	0	0	Min. ratio
			dj	3	1	0	0	x _{ij} y _{ij}
d _B	c _B	XB	В	x ₁	x ₂	X3	X 4	
0	0	X ₃	12/7	0	16/7	1	-5/7	3/4
3	2	x ₁	6/7	1	1/7	0	1/7	6
	$Z_1 = 12/7$	Z=1/5						
	Z ₂ = 60/7							
			$c_j - Z_j^1$	0	5/7	0	-2/7	
			$d_j - Z_j^2$	0	4/7	0	-3/7	
			Δ_{j}	0	252/49	0	-12/7	

$$\wedge \quad \downarrow$$

$$Z_{1} = c_{B}^{(2)} x_{B} + \alpha$$

$$= \begin{pmatrix} 0 \\ 2 \end{pmatrix} \begin{pmatrix} 12/7 \\ 6/7 \end{pmatrix} + 0 = 12/7$$

$$Z_{2} = d_{B}^{(2)} x_{B} + \beta$$

$$= \begin{pmatrix} 0 \\ 3 \end{pmatrix} \begin{pmatrix} 12/7 \\ 6/7 \end{pmatrix} + 6 = 60/7$$

$$\Delta_{j} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$\Delta_{1} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$= 0$$

$$\Delta_{2} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$= \frac{60}{7} \times \frac{5}{7} - \frac{12}{7} \times \frac{4}{7} = \frac{252}{49}$$

$$\Delta_{3} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$= 0$$

$$\Delta_{4} = Z_{2} (c_{j} - Z_{j}^{(1)}) - Z_{1} (d_{j} - Z_{j}^{(2)})$$

$$= 0$$

			c _j	2	1	0	0
			dj	3	1	0	0
d _B	c _B	XB	В	x ₁	x ₂	X3	X4
1	1	x ₂	3/4	0	1	7/16	-5/16
3	2	x ₁	3/4	1	0	-1/16	3/16
	$Z_1 = 9/4$	Z=1/4					

Z ₂ =9					
	$c_j - Z_j^1$	0	0	-5/16	-1/16
	$d_j - Z_j^2$	0	0	-1/4	-1/4
	$\overline{\Delta}_{j}$	0	0	-9/4	0

$$Z_{1} = c_{B}^{(3)}x_{B} + \alpha$$

$$= \begin{pmatrix} 1\\ 2 \end{pmatrix} \begin{pmatrix} 3/4\\ 3/4 \end{pmatrix} + 0 = 9/4$$

$$Z_{2} = d_{B}^{(3)}x_{B} + \beta$$

$$= \begin{pmatrix} 1\\ 3 \end{pmatrix} \begin{pmatrix} 3/4\\ 3/4 \end{pmatrix} + 6 = 9$$

$$\Delta_{j} = Z_{2}(c_{j} - Z_{j}^{(1)}) - Z_{1}(d_{j} - Z_{j}^{(2)})$$

$$\Delta_{1} = Z_{2}(c_{j} - Z_{j}^{(1)}) - Z_{1}(d_{j} - Z_{j}^{(2)})$$

$$= 0$$

$$\Delta_{2} = Z_{2}(c_{j} - Z_{j}^{(1)}) - Z_{1}(d_{j} - Z_{j}^{(2)})$$

$$= 0$$

Here $x_1 = 3/4$, $x_2 = 3/4$ and $Z = Z_1/Z_2 = 1/4$. Since all $\Delta_j \le 0$, therefore the current solution is the optimal basic feasible solution. But x_1 and x_2 are not integer value so make them integer, we use Branch and Bound method



Hence we get the integer solution of the given problem. The optimal solution is $x_1 = 0$, $x_2 = 2$ and max. Z = 1/4.

References

- [1] Carnes and W.W. Cooper, Programming with linear fractional functional, *Naval Research Logistics Quartely*, 9 (1962), 181-186.
- [2] S.S. Chadha, V. Chadha and R. Caldie, A study of linear inequalities applications and algorithms, *Presented at the International Conference on Operation Research for Development [ICORD]*, Anna University, Chennai, India, December (2002), 27-30.
- [3] J.R. Isbell and W.H. Marlow, Atrition games, *Naval Research Logistics Quartely*, 3 (1956), 1-99.
- [4] Martos, Hyperbolic programming, *Naval Research Logistics Quartely*, 11 (1964), 135-155.
- [5] J.K. Sharma, A.K. Gupta and M.P. Gupta, Extension of simplex technique for solving fractional programming problems, *Indian J. Pure appl. Math.*, 11(8) (1980), 961-968.
- [6] K. Swarup, Linear fractional functional programming, *Operation Research*, 13 (1965), 1029-1036.
- [7] S.F. Tantawy, Using feasible directions to solve linear fractional programming problems, *Australian Journal of Basic and Applied Sciences*, 1(2) (2007), 109-114.
- [8] H.M. Wagner and J.S.C. Yuan, Algorithm equivalence in linear fractional programming, *Management* Science, 14 (1968), 301-306.