

Algorithms for integer fractional programming

C. R. SESHAN AND V. G. TIKEKAR

Department of Applied Mathematics, Indian Institute of Science, Bangalore 560 012, India.

Received on September 5, 1979; Revised on February 6, 1980.

Abstract

In this paper, two algorithms are presented to solve integer linear fractional programming problems. The first algorithm is an extension of Bitran and Novaes method to solve a linear fractional program and the second algorithm is a refinement of the first. A numerical example is worked out by using both the algorithms, to illustrate the methods.

Key words : Linear fractional programming, parametric integer programming, Gomory method.

1. Introduction

In this paper two algorithms are presented for solving integer linear fractional programming problems. Their validity and convergence are also proved.

The first algorithm is obtained by modifying the algorithm given by Bitran and Novaes¹ for a linear fractional programming problem, to the integer case. The proofs of the theorems given here are very much simpler compared to the proofs given by Bitran and Novaes¹.

The second algorithm is a refinement of the first algorithm and it employs a simple check to decide whether the optimal solution, obtained at a particular iteration, to the parametric integer programming problem, is an optimal solution to the given integer fractional program. The second algorithm reduces by one, the number of iterations required to solve the given integer fractional program by the first algorithm. The same refinement can also be applied to the algorithm given by Bitran and Novaes¹ to solve the linear fractional programming problem.

Both the algorithms are based on the parametric approach to the fractional programming problem given by Dinkelbach² and Jagannathan³. To illustrate the algorithms a numerical example is solved using all the three algorithms.

2. Integer fractional programming problem

Consider the following integer linear fractional programming problem

$$\text{Maximise } (C^t X + \alpha)/(d^t X + \beta) \quad (1)$$

$$\text{subject to } AX = b, X \geq 0 \quad (2)$$

$$\text{and } X \text{ is integral} \quad (3)$$

where A is an $(m \times n)$ matrix, C, d, X are $(n \times 1)$ vectors, b is an $(m \times 1)$ vector, α, β are scalars and the superscript t denotes the transpose. Denote the problem given by the expressions (1), (2) and (3) as problem P .

Let

$$S = \{X \in R^n / AX = b, X \geq 0, X \text{ integral}\}.$$

It is assumed that (1) S is nonempty and bounded and (2) $d^t X + \beta > 0$ for all X in S .

For every real number λ , define the subsidiary integer programming problem $P(\lambda)$ as follows.

$$\text{Maximise } (C^t X + \alpha) - \lambda(d^t X + \beta) \quad (4)$$

subject to (2) and (3).

$F(\lambda)$ denotes the optimal value of the problem $P(\lambda)$.

3. Algorithm 1

Step 1 : Let X_0 be any feasible solution of P . Set $i = 0$.

Step 2 : Set $\lambda_{i+1} = (C^t X_i + \alpha)/(d^t X_i + \beta)$. Solve the integer programming problem $P(\lambda_{i+1})$. If X_i is an optimal solution of $P(\lambda_{i+1})$, then X_i is an optimal solution of P . Otherwise let X_{i+1} be an optimal solution of $P(\lambda_{i+1})$. Set $i = i + 1$ and repeat step 2.

3.1. Validity and convergence of the algorithm

Theorem 1 : λ_i is a strictly increasing sequence.

$$\text{Proof : } (C^t X_i + \alpha) - \lambda_{i+1}(d^t X_i + \beta) = 0. \quad (5)$$

Since X_i is not an optimal solution of $P(\lambda_{i+1})$ and X_{i+1} is an optimal solution of $P(\lambda_{i+1})$,

$$(C^t X_{i+1} + \alpha) - \lambda_{i+1}(d^t X_{i+1} + \beta) > 0. \quad (6)$$

Hence

$$\lambda_{i+2} = (C^t X_{i+1} + \alpha)/(d^t X_{i+1} + \beta) > \lambda_{i+1}.$$

Theorem 2 : If X_i is an optimal solution of $P(\lambda_{i+1})$, then X_i is an optimal solution of P .

Proof : Since X_i is an optimal solution of $P(\lambda_{i+1})$, it follows from (5) that

$$(C^t X + \alpha) - \lambda_{i+1} (d^t X + \beta) \leq 0 \text{ for all } X \text{ in } S \quad (7)$$

Hence

$$(C^t X + \alpha)/(d^t X + \beta) \leq (C^t X_i + \alpha)/(d^t X_i + \beta) \text{ for all } X \text{ in } S.$$

Therefore X_i is an optimal solution of P .

Theorem 3 : The algorithm converges in a finite number of steps.

Proof : Since λ_i is a strictly increasing sequence, the X_i do not repeat themselves until termination. Since the number of integer solutions of P is finite, $X_i = X_{i+1}$ for some i and hence the algorithm terminates.

Example 1 : Consider the following problem P_1 .

$$\text{Maximise } (7x_1 + 9x_2 + 3)/(3x_1 + 4x_2 + 2) \quad (8)$$

subject to

$$2x_1 + 3x_2 + x_3 = 6 \quad (9)$$

$$3x_1 + 2x_2 + x_4 = 5 \quad (10)$$

$$x_1, x_2 \geq 0 \text{ and integral} \quad (11)$$

$X_0 = (0, 0, 6, 5)$ is a feasible solution of P_1

Set $\lambda_1 = 3/2$. Consider the problem $P(\lambda_1)$

$$\text{Maximise } (5/2) x_1 + 3x_2$$

subject to (9), (10) and (11)

$X_1 = (0, 2, 0, 1)$ solves $P(\lambda_1)$. Set $\lambda_2 = 21/10$

Consider the problem $P(\lambda_2)$

$$\text{Maximise } (7/10) x_1 + (6/10) x_2 - (12/10)$$

subject to (9), (10) and (11).

$X_2 = (1, 1, 1, 0)$ solves $P(\lambda_2)$. Set $\lambda_3 = 19/9$

Consider the problem $P(\lambda_3)$

Maximise $(6/9)x_1 + (5/9)x_2 - (11/9)$

subject to (9), (10) and (11)

$X_2 = (1, 1, 1, 0)$ solves the problem $P(\lambda_3)$.

Hence the algorithm terminates and X_2 is an optimal solution of P_1 .

4. Algorithm 2

The first algorithm terminates when X_i is an optimal solution of $P(\lambda_{i+1})$ for some i . In Algorithm 1, even after finding the optimal solution X_i , we have to solve one more integer program before deciding that X_i is an optimal solution of P . If we had some means of checking the optimality of X_i at each iteration, we can stop the algorithm at the last but one iteration. Such a check is provided in algorithm 2. All the quantities required for the check can be easily computed from the final simplex tableau at each iteration.

At each iteration, we solve the problem $P(\lambda)$ for some value of λ by Gomory method (see Hadley⁵) and let \bar{X} be the optimal solution of $P(\lambda)$. Let B be the basis matrix corresponding to the optimal solution \bar{X} [The matrix B also includes the rows corresponding to the cuts introduced by Gomory method in solving $P(\lambda)$]. Let

$$Z_j^1 = C_B^t B^{-1} a_j, \quad Z_j^2 = d_B^t B^{-1} a_j$$

for those columns a_j of A not in the basis B

(The columns of A also contain the columns corresponding to the slack variables corresponding to the Gomory cuts) (see Hadley⁶).

Algorithm 2 :

Step 1 : Let X_0 be any feasible solution of P . Set $i = 0$.

Step 2 : Set $\lambda_{i+1} = (C^t X_i + \alpha) / (d^t X_i + \beta)$.

Solve the problem $P(\lambda_{i+1})$ by Gomory method.

If X_i is an optimal solution of $P(\lambda_{i+1})$, then X_i is an optimal solution of P . Otherwise let X_{i+1} be an optimal solution of $P(\lambda_{i+1})$. Go to step 3.

Step 3 : Set $\bar{X} = X_{i+1}$ and $\bar{\lambda} = \lambda_{i+1}$.

Compute the quantities $Z_j^1 - C_j$ and $Z_j^2 - d_j$ corresponding to the non-basic variables x_j from the final simplex tableau for the problem $P(\bar{\lambda})$

$$\text{Compute } \mu = \min_{Z_j^2 - d_j > 0} \left(\frac{Z_j^1 - C_j}{Z_j^2 - d_j} \right). \quad (12)$$

If $Z_j^2 - d_j \leq 0$ for every j , take $\mu = +\infty$.

Step 4 : (a) If $\mu = +\infty$, then \bar{X} is an optimal solution of the problem P . (b) If $\mu \neq +\infty$, compute

$$F(\mu) = (C^t \bar{X} + \alpha) - \mu (d^t \bar{X} + \beta).$$

If $F(\mu) \leq 0$, then \bar{X} is an optimal solution of the problem P . (c) If $F(\mu) > 0$, set $i = i + 1$ and go to step 2.

4.1. Validity and convergence of algorithm 2

The convergence of this algorithm is proved in Section 3. The validity of the algorithm depends on the following results.

Theorem 4 : \bar{X} is an optimal solution and λ , the optimal value of P if and only if $F(\lambda) = 0$ and \bar{X} is an optimal solution of $P(\lambda)$.

Theorem 5 : For every X in S , the set $A(X) = \{\lambda \in R \mid X \text{ is an optimal solution of } P(\lambda)\}$ is a convex subset of R and hence is an interval in R .

Theorem 6 : In each interval $A(X)$, $F(\lambda)$ is a piecewise linear function of λ and hence is a continuous function of λ .

Theorem 7 : $F(\lambda)$ is a strictly decreasing function of λ and

$$\lim_{\lambda \rightarrow +\infty} F(\lambda) = -\infty \quad \text{and} \quad \lim_{\lambda \rightarrow -\infty} F(\lambda) = +\infty.$$

Theorem 8 : If there exists λ_1 and λ_2 such that $F(\lambda_1) > 0$ and $F(\lambda_2) < 0$ and both $P(\lambda_1)$ and $P(\lambda_2)$ have same optimal solution \bar{X} , then \bar{X} is an optimal solution of P .

Proof of validity of the algorithm

If X_i is an optimal solution of $P(\lambda_{i+1})$, by Theorem 2, X_i is an optimal solution of P . Since

$$\bar{\lambda} = \lambda_{i+1} = (C^t X_i + \alpha) / (d^t X_i + \beta),$$

$$(C^t X_i + \alpha) - \bar{\lambda} (d^t X_i + \beta) = 0.$$

Hence $F(\bar{\lambda}) > 0$. Since \bar{X} is an optimal solution of $P(\bar{\lambda})$

$$(Z_j^1 - C_j) - \bar{\lambda}(Z_j^2 - d_j) \geq 0 \text{ for all } j. \quad (13)$$

For $\lambda_1 \geq \bar{\lambda}$, \bar{X} is an optimal solution of $P(\lambda_1)$ if

$$(Z_j^1 - C_j) - \lambda_1(Z_j^2 - d_j) \geq 0 \text{ for all } j. \quad (14)$$

(14) is automatically satisfied if $Z_j^2 - d_j \leq 0$ for all j .

(14) is satisfied only when

$$\frac{Z_j^1 - C_j}{Z_j^2 - d_j} \geq \lambda_1 \text{ if } Z_j^2 - d_j > 0 \quad (15)$$

(i.e.), only when $\mu \geq \lambda_1$ where μ is defined as in (12).

Hence \bar{X} is an optimal solution of $P(\lambda_1)$ for $\lambda \leq \lambda_1 \leq \mu$. If $\mu = +\infty$, then by continuity of F and Theorem 7, it follows that there exists some $\lambda_1 > \bar{\lambda}$ such that $F(\lambda_1) = 0$ and \bar{X} is an optimal solution of $P(\lambda_1)$. Hence \bar{X} is an optimal solution of P . If $F(\mu) \leq 0$, then it follows from Theorem 8, that \bar{X} is an optimal solution of P .

Example 2: Consider the Problem P_1 given in Example 1. $X_0 = (0, 0, 6, 5)$ is a feasible solution of P_1 . Set $\lambda_1 = 3/2$. $X_1 = (0, 2, 0, 1)$ solves $P(\lambda_1)$. The final simplex tableau for the problem $P(\lambda_1)$ is given in Table I. Third row in Table I corresponds to the Gomory cut introduced in solving $P(\lambda_1)$ and x_5 is the slack variable corresponding to the cut.

$$\bar{\lambda} = \lambda_1 = 3/2. \quad \bar{X} = X_1 = (0, 2, 0, 1).$$

$$C_B = (9, 7, 0). \quad d_B = (4, 3, 0).$$

$$Z_3^1 - C_3 = 2, \quad Z_3^2 - d_3 = 1.$$

$$Z_5^1 - C_5 = 1, \quad Z_5^2 - d_5 = 1/3$$

$$\mu = \min\{2, 3\} = 2.$$

$$F(\bar{\lambda}) = 6 > 0 \quad \text{and} \quad F(\mu) = 1 > 0.$$

Set $\lambda_2 = 21/10$. $X_2 = (1, 1, 1, 0)$ solves $P(\lambda_2)$.

The final simplex tableau corresponding to $P(\lambda_2)$ is given in Table II. Third row in Table II corresponds to the Gomory cut introduced in solving $P(\lambda_2)$ and x_5 is the slack variable corresponding to the cut.

$$\bar{\lambda} = \lambda_2 = 21/10. \quad \bar{X} = X_2 = (1, 1, 1, 0).$$

$$C_B = (9, 7, 0), \quad d_B = (4, 3, 0)$$

Table I

Basis	Value of basic variables	x_1	x_2	x_3	x_4	x_5
x_2	2	0	1	1	0	$-2/3$
x_1	0	1	0	-1	0	1
x_4	1	0	0	1	1	$-5/3$

Table II

Basis	Value of basic variables	x_1	x_2	x_3	x_4	x_5
x_3	1	0	1	0	-1	1
x_1	1	1	0	0	1	$-2/3$
x_3	1	0	0	1	1	$-5/3$

$$Z_4^1 - C_4 = 2, \quad Z_4^2 - d_4 = -1$$

$$Z_5^1 - C_5 = 13/3, \quad Z_5^2 - d_5 = 2.$$

$$\mu = 13/6.$$

$$F(\bar{\lambda}) = 1/10 > 0 \quad \text{and} \quad F(\mu) = -3/6 < 0.$$

Hence $\bar{X} = (1, 1, 1, 0)$ is an optimal solution of P .

Note : Using the first algorithm, we had to solve three integer programs, whereas with the second one two integer programs sufficed.

References

1. BITRAN, G. R. AND NOVAES, A. G. Linear programming with a fractional objective function, *Opns. Res.*, 1973, 21, 22-29.
2. KANTI SWARUP Linear fractional functional programming, *Opns. Res.*, 1965, 13, 1029-1036.

3. DINKELBACH, W. On nonlinear fractional programming, *Manage. Sci.*, 1967, 13, 492-498.
4. JAGANNATHAN, R. On some properties of programming problems in parametric form pertaining to fractional programming, *Manage. Sci.*, 1966, 12, 609-615.
5. HADLEY, G. *Nonlinear and Dynamic Programming*, Addison Wesley, Reading, Massachusetts, 1970, Chap. 8, pp. 272-276.
6. HADLEY, G. *Linear Programming*, Addison Wesley, Reading, Massachusetts, 1962, Chap. 3, pp. 84-87.