AN EXTENSION OF UPPER BOUNDED TECHNIQUE FOR A LINEAR FRACTIONAL PROGRAM [1])

S. S. Chadha

The problem considered here is a generalization of my earlier work [2, 3]. The generalization is in the sense that the constraint set of the problem under investigation has a large number of equations in each block, rather than one as in [2, 3], coupled together by relatively few connecting equations. The objective function which is the ratio of two linear functions is to be maximized. From the basis of the given problem a working basis is obtained which is much smaller in size than the original one. Two sets of pricing vectors are computed to test the optimality of the solution at any stage of the algorithm.

1. INTRODUCTION

A decomposition principle for solving a linear fractional program has already been given in [1]. A technique for solving a large block diagonal structured linear fractional program, where each of the block contains just one equation, has also been given in [2,3]. The method proposed here solves a linear fractional program with more than one equation in each block. The special structured program is the following :

Maximize

$$Z = \frac{C_0 x_0 + C_1 x_1 + C_2 x_2 + \cdots + C_L x_L}{D_0 x_0 + D_1 x_1 + D_2 x_2 + \cdots + D_L x_L + \alpha}$$

subject to

$$\begin{aligned}
A_0' x_0 + A_1' x_1 + A_2' x_2 + \cdots\cdots\cdots + A_L' x_L &= b^0 \\
A_1 x_1 \qquad\qquad\qquad\qquad\qquad &= b^1 \\
A_2 x_2 \qquad\qquad\qquad &= b^2 \\
\ddots \qquad A_L x_L &= b^L
\end{aligned}$$

$$x \geq 0.$$

Fig. 1

$X \; [x_0, x_1, x_2, \cdots, x_L]$ is a vector of N components, x_i, C_i, D_i have each n_i components such that $N = n_0 + n_1 + \cdots + n_L$. A_i' and A_i are $m \times n_i$ and $m_i \times n_i$ matrices respectively. M, the total number of equations, is given by $M = m + m_1 + \cdots + m_L$. The system in full is given in fig. 2.

1) My sincere thanks are due to Prof. R. S. Varma, D. Sc. F. N. I. Dean, Faculty of Mathematics, University of Delhi, and Dr. R. N. Kaul for their valuable guidance in my work.

Maximize $$Z = \frac{c_1 x_1 + c_2 x_2 + \cdots + c_{n_0} x_{n_0} + c_{n_0+1} x_{n_0+1} + \cdots + c_{n_1} x_{n_1} + \cdots + \cdots + c_{n_{L-1}+1} x_{n_{L-1}+1} + \cdots + c_{n_L} x_{n_L}}{d_1 x_1 + d_2 x_2 + \cdots + d_{n_0} x_{n_0} + d_{n_0+1} x_{n_0+1} + \cdots + d_{n_1} x_{n_1} + \cdots + \cdots + d_{n_{L-1}+1} x_{n_{L-1}+1} + \cdots + d_{n_L} x_{n_L}}.$$

subject to

$$\begin{array}{l}
A_1^1 x_1 + A_1^2 x_2 + \cdots + A_1^{n_0} x_{n_0} + A_1^{n_0+1} x_{n_0+1} + \cdots + A_1^{n_1} x_{n_1} + \cdots + \cdots + A_1^{n_{L-1}+1} x_{n_{L-1}+1} + \cdots + A_1^{n_L} x_{n_L} = b_1 \\
\vdots \\
A_m^1 x_1 + A_m^2 x_2 + \cdots + A_m^{n_0} x_{n_0} + A_m^{n_0+1} x_{n_0+1} + \cdots + A_m^{n_1} x_{n_1} + \cdots + \cdots + A_m^{n_{L-1}+1} x_{n_{L-1}+1} + \cdots + A_m^{n_L} x_{n_L} = b_m \\
\qquad A_{m+1}^{n_0+1} x_{n_0+1} + \cdots + A_{m+1}^{n_1} x_{n_1} = b_{m+1} \\
\qquad \vdots \\
\qquad A_{m_1}^{n_0+1} x_{n_0+1} + \cdots + A_{m_1}^{n_1} x_{n_1} = b_{m_1} \\
\qquad \qquad \ddots \\
\qquad \qquad A_{m_{L-1}+1}^{n_{L-1}+1} x_{n_{L-1}+1} + \cdots + A_{m_{L-1}+1}^{n_L} x_{n_L} = b_m \\
\qquad \qquad \vdots \\
\qquad \qquad A_{m_L}^{n_{L-1}+1} x_{n_{L-1}+1} + \cdots + A_{m_L}^{n_L} x_{n_L} = b_{m_L}
\end{array}$$

$$(x_j \geq 0)$$

Fig. 2

No doubt the decomposition principle [1] can be used with greater advantage over the methods proposed in [5, 7]; but here, by taking the advantage of the very special structure of the problem we extend the device given in [3] to solve such type of problems. The computation is carried out with the help of a working basis, which has been derived from the basis of the full system and is of considerably smaller in size. This indeed results in a substantial saving in computations. Following are the usual assumptions under which the algorithm works :

1. Denominator of the objective function keeps the same sign (say positive), throughout the feasible region.
2. The given system is of full rank, i.e. none of the given equations is redundant.
3. We assume that the constraint set is non-void.
4. The proposed algorithm is of importance if we assume that $L \ggg m$.
5. Underscoring of a matrix or a vector is done to distinguish it from that of the reduced system.

Following closely the notations and terminology of [2, 3, 4, 6] some definitions and theorems have been set out in the next section, while the algorithm in detail is given in section 3 of the paper.

2. DEFINITIONS AND THEOREMS

The system of equations which couple together the different blocks is refered as the linking set. The i-th set of columns of variables S_i is a set of columns or the set of the components of X that a linking set has in common with the i-th block. $\underline{B}$ denotes the basis of the full system and is given by $\underline{B} = [\underline{A}^{j_1}, \underline{A}^{j_2}, \cdots \underline{A}^{j_M}]$ while the contribution of the set S_i towards the basis $\underline{B}$ is denoted by

$$\begin{bmatrix} B_i' \\ 0 \\ B_i \\ 0 \end{bmatrix}.$$

Theorem 1. *The set S_l will definitely contribute at least m_l vectors for the basis of the full system.*

Proof. Since $\underline{B}$ forms the basis of the full system, therefore, for an M components vector $[0, 0,\ldots,0, b_{m_l+1},\ldots,b_{m_{l+1}},0,\ldots,0]^T$ we can find scalars λ_i such that

$$\sum_{i=1}^{M} \lambda_i \underline{A}^{ji} = [0, 0,\ldots,0,b_{m_l+1},\ldots,b_{m_{l+1}}, 0,\ldots,0]^T$$

$$i.e.\quad \sum \lambda_i A_k^{ji} = b_k\,;\quad k = m_l+1,\ldots,m_{l+1}\,;\quad 0 \leq l \leq L-1\,;$$

which implies that there are m_{l+1} or more A^{ij} in the $(l+1)$ th block, since each of the blocks is assumed to be of the full rank. Hence the result.

Theorem 2. *The numbers of the sets S_i containing more than m_i basic variables cannot exceed m.*

Proof. Since the basis is of the full rank, therefore, the basis consists of $m+\sum_{i=1}^{L} m_i$ vectors. This, combined with the result of the previous theorem, establishes the desired result.

A set S_i which contributes more than m_i variables towards the basis of the ful system is known as an essential set and S_0 is always included in it. The rest of the sets are known as inessential sets.

Reduced system and the Working Basis.

Knowing the initial basic feasible solution (using phase 1) we can separate the sets S_i ; $i = 1, 2, \ldots, L$ into two classes, essential and inessential. The reduced system is obtained from the full system by deleting the sub-sets $A_i X_i = b^i$ for which S_i is inessential. The set of the essential basis columns restricted to the reduced system will be our working basis, B. It follows from the construction of the working basis that the working basis will be a basis for the reduced system. A thing of special interest here is that the size of the working basis changes from step to step; further, one observes that the number of blocks in the working basis may also change with the change in step, however, their upper bound is m.

3. THE ALGORITHM

If $\underline{B} = [\underline{A}^{ji} ; i = 1, \ldots, M]$ is the basis of the full system then let $B = [A^{ki} ; i = 1, 2, \ldots M_B]$ denotes our working basis, where members of B are obtained from the corresponding members of $\underline{B}$ by removing their appropriate components. The values of the inessential basic variables are obtained by solving the subproblems $A_i X_i = b^i$ that belong to the inessential sets which are solved independently by putting equal to zero the variables not associated with the basis. In order to test the optimality of the solution which is associated with the basis $\underline{B}$, we define

$$[\pi, \mu^{(1)}, \mu^{(2)}, \ldots, \mu^{L}] = [\pi_1, \pi_2, \ldots, \pi_m, \mu^{(1)}_{m+1} \ldots \mu^{(1)}_{m_1}, \ldots \mu^{(L)}_{m_{L-1}+1} \ldots \mu^{(L)}_{m_L}]$$

and

$$[\varphi, \lambda^{(1)}, \lambda^{(2)}, \ldots \lambda^{L}] = [\varphi_1, \varphi_2, \ldots, _m, \lambda^{(1)}_{m+1} \ldots \lambda^{(1)}_{m_1}, \ldots \lambda^{(L)}_{m_{L-1}+1} \ldots \lambda^{(L)}_{m_L}]$$

as our set of pricing vectors corresponding to the numerator and the denominator of the objective function respectively. These are determined [1, 5, 7] from the relations

$$[\pi, \mu^{(1)}, \ldots, \mu^{(L)}] = -C_{\underline{B}} \underline{B}^{-1}$$

and

$$[\varphi, \lambda^{(1)}, \ldots, \lambda^{(L)}] = -D_{\underline{B}} \underline{B}^{-1}$$

where $C_{\underline{B}}$ and $D_{\underline{B}}$ are the cost vectors associated with the basis $\underline{B}$. Practically there would have been no advantage if we were to find the prices as given. We shall try to take the help of our working basis B to get these prices. As has been shown in[8] we find that $[\pi_1, \pi_2, \ldots, \pi_m]$, $[\varphi_1 \varphi_2, \ldots, \varphi_m]$ and those $\mu^{(i)}$ and $\lambda^{(i)}$ which are associated with the essential sets can be obtained from,

$$-C_B B^{-1} \quad \text{and} \quad -D_B B^{-1},$$

where C_B and D_B are the cost (row) vectors associated with the working basis B. The values of $\mu^{(i)}$ and $\lambda^{(i)}$ when i belongs to an inessential set are given by

$$\mu^{(i)} = -[C_{Bi} + \pi B_i'] B_i^{-1}$$
$$\lambda^{(i)} = -[D_{Bi} + \pi B_i'] B_i^{-1},$$

where C_{Bi} and D_{Bi} are the cost vectors associated with the basic variables of the inessential set i.

After having determined the multipliers, we are now ready to test the optimality of the solution, for which we compute Δ_j, where j refers to non basic columns.

$$\Delta_j = Z_2 [c_j + (\pi, \mu^{(1)}, \ldots \mu^{(L)}) \underline{A}^j] - Z_1 [d_j + (\varphi, \lambda^{(1)}, \lambda^{(2)} \ldots \lambda^{(L)}) \underline{A}^j].$$

Here Z_1 and Z_2 are the values of the numerator and the denominator of the objective function at the basic feasible solution. This solution will stay as an optimal one [1, 5, 7] if all $\Delta_j \leq 0$, otherwise we find $\Delta_s = \text{Max}_j \, \Delta_j$; $(\Delta_j > 0)$ and thus $\underline{A}^s$ qualifies for entry into the basis. Now to determine the vector which has to leave the basis we express the column $\underline{A}^s$ and the vector $\underline{b} = (b^0, b^1, ..., b^{(L)})$ in terms of the basis $\underline{B}$. If $\underline{A}^{-s}$ and $\underline{b}^{-1}$ denote the representation of $\underline{A}^s$ and $\underline{b}$ in terms of the basis $\underline{B}$, then we have

(3.1)
$$\underline{A}^s = \underline{B}\,\underline{A}^{-s} \quad ; \quad \underline{b} = \underline{B}\,\underline{b}^{-1}$$
or
$$\underline{A}^s = \sum_{i=1}^{M} q_i^s \, \underline{A}^{ji} \quad ; \quad \underline{b} = \sum_{i=1}^{M} b_i^* \, \underline{A}^{ji} .$$

Thus the column which has to leave the basis is given by $\frac{b_\gamma^*}{q_\gamma^s} = \text{Min}_i \frac{b_i^*}{q_i^s}$, $(q_i^s > 0)$. Now $\underline{A}^s$ enters the basis and vector $\underline{A}^\gamma$ will leave the basis. As has been shown in [6] the solution of (3.1) can be decomposed into two parts according as the set S_σ to which $\underline{A}^s$ belongs is essential or inessential. For the sake of completeness a brief account of the same is given in the following lines.

In the case where S_σ is essential, solving (3.1) is equivalent to solve

(3.2)
$$B\,A^{-s} = A^s \quad ; \quad B\,b^{-1} = b.$$

But, however, if S_σ is inessential the equations to solve are

(3.3)
$$B\,A^{-s} = \begin{bmatrix} A'^s \\ 0 \end{bmatrix} - \begin{bmatrix} B'_\sigma \\ 0 \end{bmatrix} A^{-s},$$
$$B\,b^{-1} = b - \sum \begin{bmatrix} B'_i \\ 0 \end{bmatrix} b_i^{-1}$$

rather than (3.1).

The following four exhaustive cases can be considered for updating. Let $\underline{A}^s \in S_\sigma$ and $\underline{A}^\gamma \in S_\sigma$.

a) Both the sets S_ϱ and S_σ are inessential. This can happen only when S_ϱ will coincide with S_σ, and in this case the working basis will not be affected at all. The only change will be in the values of those basic variables belonging to the set S_σ.

b) Both the sets S_ϱ and S_σ are essentials. In this case the pivoting is done with the help of the working basis B. $\hat{B}^{-1}$, the new value of B^{-1}, is given by $B^{-1}\,E\,\hat{B}^{-1}$ where

$$E = \begin{bmatrix} 1 & \cdots\cdots & -k_1/k_s & \cdots\cdots & 0 \\ 0 & & -k_2/k_s & \cdots\cdots & 0 \\ 0 & & \vdots & & \vdots \\ \vdots & & \vdots & & \vdots \\ 0 & \cdots\cdots & 1/k_s & \cdots\cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & & -k_{MB}/k_s & \cdots & 1 \end{bmatrix}$$

when
$$A^s = k_1 A^{h_1} + k_2 A^{h_2} + \ldots + k_s A^{h_s} + \ldots + k_{MB} A^{h_{MB}}$$

c) S_Q is inessential but S_σ is essential. In the light of our discussion we observe that this can never happen.

d) S_Q is essential but S_σ is inessential. This case is resolved by introducing S_σ in the essential sets. The size of the working basis is increased by m_σ, and then pivot with the help of the new working basis as in b). A close study of this case shows that : 1) if the set S_Q is contributing more than $m_Q + 1$ vectors towards the basis $\underline{B}$, and the numbers of the essential sets associated with $\underline{B}$ is less than m, (which will certainly be the case) then the size of the working basis will remain increased by m_σ. 2) if S_Q is contributing just $m_Q + 1$ vectors towards the basis $\underline{B}$, and the numbers of the essential sets associated with $\underline{B}$ will be m (this will definitely happen when we let the set S_Q become inessential which will decrease the size of the basis by m_Q.

REFERENCES

[1] CHADHA, S. S. : *A Decomposition Principle for Fractional Programming*, OPSEARCH, 4, 3, 1967.

[2] CHADHA, S. S. : *A Generalized Upper Bounded Technique for a Linear Fractional Program*, to appear in METRIKA, 1971.

[3] CHADHA, S. S. : *Upper Bounded Technique for a Linear Fractional Program*, communicated.

[4] DANTZIG, GEORGE B. AND VAN SLYKE, R. M. : *Generalized Upper Bounded Technique-I* ORC 64-17 (RR), Operations Research Center, BERKELEY, UNIVERSITY OF CALIFORNIA, 1964.

[5] BELA, MARTES : *Hyperbolic Programming*, Translated by ANDREW AND VERONIKA WHINSTON, Naval Res. Log. Quart II, 1964.

[6] SAKAROVITCH, M. AND SAIGAL, R. : *An Extension of Generalized Upper Bounding Techniques for Structured Linear Programs*, SIAM J. Appl. Math., 15, 4, 1967.

[7] KANTI, SWARUP : *Linear Fractional Programming*, Operations Research 13, 1965.

HANS RAJ COLLEGE,
UNIVERSITY OF DELHI,
DELHI-7, (INDIA)

(Manuscript received October 1, 1969)

ÖZET

Bu araştırmaya konu teşkil edilen problem, yazarın daha önceki çalışmalarının bir genelleştirilmesidir [2,3]. Genelleşme, [2,3] teki durumdan farklı olarak, her blokta tek bir sınırlayıcı şart denkleminin yerine, aralarında bâzı bağıntıları bulunan birer denklem sisteminin verilmesinden ibarettir. Gâye fonksiyonu lineer iki fonksiyonun oranı olarak düşünülmekte ve bunun maksimum kılınması istenmektedir. Verilen problemin bazından hareket edilerek, bu bazdan daha az sayıda eleman ihtiva eden bir çalışma bazı bulunmaktadır. Çözüm algoritmasının her kademesinde, çözümün optimallığını kontrol etmek için iki fiatlandırma vektörü hesaplanmaktadır.