

ADAPTIVE PROJECTION METHODS FOR LINEAR FRACTIONAL PROGRAMMING

AHLEM BENNANI*, DJAMEL BENTERKI AND HASSINA GRAR

Abstract. In this paper, we are interested in solving a linear fractional program by two different approaches. The first one is based on interior point methods which makes it possible to solve an equivalent linear program to the linear fractional program. The second one allows us to solve a variational inequalities problem equivalent to the linear fractional program by an efficient projection method. Numerical tests were carried out by the two approaches and a comparative study was carried out. The numerical tests show clearly that interior point methods are more efficient than of projection one.

Mathematics Subject Classification. 90C32, 35R35, 90C51.

Received February 12, 2020. Accepted August 10, 2020.

1. INTRODUCTION

Linear fractional programming problems (LFP) are a class of mathematical programming, where the constraints are linear functions and the objective function is the ratio of two linear or affine functions. This field of mathematical programming was developed by the Hungarian mathematician Martos in 1960. Such programming problems have recently been a subject of wide interest, it has applications in various fields such as operational research, data base, engineering, combinatorial optimization, economy... etc. Many resolution methods have been proposed to solve (LFP), among them, there are three major techniques:

- **Direct resolution:** (LFP) is treated in its initial form without changing neither the objective function or the set of constraints. This technique interested many mathematicians among them Swarup [16], Gimore and Gomory [7], Martos [12], Swarup [17], Sharma *et al.* [15] and Cambini and Martein [4].
- **The parametric approach:** contrary to the direct resolution, we build a problem with a simplified objective function which is a linear combination of the numerator and the denominator through a parameter without changing the set of constraints. It was firstly identified by Isbell and Merlow [9], then generalized in (1967) by Dinkelbach [6] for nonlinear fractional programs.
- **The resolution of an equivalent programming problem:** a change of variables allows to simplify the problem, then solving the equivalent program by well known algorithms. This technique was introduced firstly by Charnes and Cooper [5], and developed later by Bitrand *et al.* [3], Babul Hasan *et al.* [1], Saha *et al.* [14].

Keywords. Linear fractional programming, variational inequalities problem, interior point methods, projection methods.

Laboratory of Fundamental and Numerical Mathematics LMFN, Department of Mathematics, Faculty of Sciences, Setif-1 Ferhat Abbas University, Setif 19000, Algeria.

*Corresponding author: ahlem.bennani@outlook.fr

The purpose of this paper is to solve (LFP) by two projection approaches. The first one consists at converting (LFP) in a suitable way into a (LP), then solve it by the projection method of interior points introduced by Ye–Lustig [10]. The second approach is to convert (LFP) in an adequate way to a variational inequalities problem (VIP) then solve it by an efficient projection method introduced by Grar *et al.* [8].

The paper is organized as follows: In Section 2, we describe the mathematical formulation of (LFP), then in Section 3, we present the first approach based on interior point methods where we convert (LFP) into (LP) and we recall the algorithm of Ye–Lustig for (LP). In Section 4, we present the second approach where we convert (LFP) into (VIP) and give Grar’s algorithm. Section 5 is reserved to numerical tests and making a comparison between the two projection approaches. The last section represents a general conclusion on the work carried out.

2. DESCRIPTION OF A LINEAR FRACTIONAL PROGRAMMING PROBLEM (LFP)

A linear fractional programming problem (LFP) is written as:

$$(\text{LFP}) \begin{cases} \min G(x) = \frac{c^t x + \alpha}{d^t x + \beta} \\ Ax = b \\ x \geq 0, \end{cases}$$

where $c, d \in \mathbb{R}^n$; $\alpha, \beta \in \mathbb{R}$; $A \in \mathbb{R}^{m \times n}$ with $\text{rank}(A) = m < n$; $b \in \mathbb{R}^m$ and $x \in \mathbb{R}^n$ with $d^t x + \beta > 0$.

Lemma 2.1 ([18]). *The objective function G is at a time pseudoconvex and pseudoconcave.*

Properties

- Since G is at a time pseudoconvex and pseudoconcave so, it is also strictly quasiconvex and strictly quasiconcave respectively.
- Since G is continuous, quasiconvex and quasiconcave so, at least one optimal solution of (LFP) is attained at an extreme point of the polyhedron $P = \{x \in \mathbb{R}^n / Ax = b, x \geq 0\}$.
- Since G is pseudoconvex and pseudoconcave so, if $\nabla G(\bar{x}) = 0$, then \bar{x} is a global optimum.

3. CONVERSION OF (LFP) INTO (LP)

In this part, we present a formulation of (LFP) by converting it into a linear program (LP), then solving it via an approach of interior point methods.

Lemma 3.1. *The linear fractional programming problem (LFP) can be written in the following equivalent linear programming problem (LP)*

$$(\text{LP}) \begin{cases} \min c_1^t x + \alpha' \\ Ax = b \\ x \geq 0, \end{cases}$$

where

$$c_1 = c - \tilde{z}d \quad \text{and} \quad \alpha' = \alpha - \tilde{z}\beta,$$

with $\tilde{z} = \frac{c^t \bar{x} + \alpha}{d^t \bar{x} + \beta}$ is an upper bound of the optimal value of the objective function and \bar{x} is a given strictly feasible solution of (LFP).

Proof. Let z^* be the optimal value of (LFP), then

$$(\text{LFP}) \begin{cases} \min \frac{c^t x + \alpha}{d^t x + \beta} = z^* \\ Ax = b \\ x \geq 0 \end{cases} \iff \begin{cases} \min [c^t x + \alpha - z^*(d^t x + \beta)] = 0 \\ Ax = b \\ x \geq 0, \end{cases}$$

which gives us

$$\begin{cases} \min (c^t - z^* d^t) x + (\alpha - z^* \beta) = 0 \\ Ax = b \\ x \geq 0. \end{cases}$$

In the numerical implementation, we approximate z^* at each iteration k , by an upper bound \tilde{z} such as:

$$z^* \leq \tilde{z},$$

where

$$\tilde{z} = \frac{c^t \tilde{x} + \alpha}{d^t \tilde{x} + \beta}$$

with \tilde{x} is a given strictly feasible solution of (LFP).

Then, (LFP) can be written in the equivalent (LP) form as follows:

$$(LP) \begin{cases} \min c_1^t x + \alpha' \\ Ax = b \\ x \geq 0, \end{cases}$$

where

$$c_1 = c - \tilde{z}d \quad \text{and} \quad \alpha' = \alpha - \tilde{z}\beta.$$

□

In order to solve the obtained linear programming problem (LP), we apply the following Ye–Lustig’s variant of interior points methods.

3.1. Ye–Lustig’s algorithm

Consider the standard linear programming problem:

$$(LP) \begin{cases} \min c^t x \\ Ax = b \\ x \geq 0, \end{cases}$$

where $A \in \mathbb{R}^{m \times n}$ with $\text{rank}(A) = m < n$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$.

In this part, we give the description of Ye–Lustig’s algorithm for solving the linear program (LP). For more details, see [10, 11].

3.1.1. Description of Ye–Lustig’s algorithm

Begin algorithm

Initialization

Let $x_0 > 0$ a strictly feasible solution for (LP), P_0 the projection of the cost vector on the kernel of the constraints matrix, $k = 0$ and $\varepsilon > 0$, be a given precision.

– **While** $\frac{\|P_k\|}{|c^t x_0|} > \varepsilon$ **do**

(1) Build the constraints’s matrix

$$B_k = [A_k, -b], \quad \text{where} \quad A_k = AD_k \quad \text{and} \quad D_k = \text{diag}(x_k).$$

(2) Compute the projection P_k

$$P_k = \left[I_{n+1} - B_k^t (B_k B_k^t)^{-1} B_k \right] \begin{pmatrix} D_k c \\ -c^t x_k \end{pmatrix} \quad \text{and take} \quad d_k = \frac{P_k}{\|P_k\|}.$$

(3) Compute the current iterate

$$y^{(k+1)} = \frac{e_{n+1}}{n+1} - \alpha_k r d_k, \quad r = \frac{1}{\sqrt{n(n+1)}}.$$

Where α_k is the displacement step with $0 < \alpha_k < 1$ and $e_{n+1} = (1, 1, \dots, 1) \in \mathbb{R}^{n+1}$.

(4) Take

$$x_{k+1} = \frac{D_k y^{(k+1)} [n]}{y_{n+1}^{(k+1)}}, \quad k = k + 1.$$

Where $y^{(k+1)} [n]$ is the vector of the first n components of $y^{(k+1)}$.

– **End while**

End algorithm

4. CONVERSION OF (LFP) INTO (VIP)

Consider the following variational inequalities problem:

$$\text{VIP}(F, C) \begin{cases} \text{find } \bar{x} \in C, \text{ such that} \\ \langle F(\bar{x}), \bar{x} - x \rangle \geq 0 \quad \forall x \in C, \end{cases}$$

where C is a nonempty closed convex set in \mathbb{R}^n and F is a continuous mapping from \mathbb{R}^n to itself and $\langle \cdot, \cdot \rangle$ denotes the usual inner product in \mathbb{R}^n .

In this part, we present a formulation of (LFP) by converting it into a variational inequalities problem (VIP), then solving it *via* the projection approach of Grar [8].

4.1. The relation between (LFP) and (VIP)

In this part, we present a formulation of (LFP) by converting it into a variational inequalities problem (VIP), then resolving it *via* an approach of projection type. The link between (VIP) and (LFP) is essentially based on the condition of optimality of (LFP).

Lemma 4.1. *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable function and C a bounded convex set of \mathbb{R}^n . We define the following optimization problem:*

$$(\text{OP}) \begin{cases} \min f(x) \\ x \in C. \end{cases}$$

So, if \bar{x} is a local minimum solution of (OP), then:

$$\langle \nabla f(\bar{x}), x - \bar{x} \rangle \geq 0, \quad \forall x \in C. \quad (\text{VIP}(\nabla f, C))$$

Proof. Since C is convex, then $\forall x \in C, \forall t \in [0, 1]$, we have

$$f(tx + (1-t)\bar{x}) = f(\bar{x} - t(x - \bar{x})) \geq f(\bar{x}),$$

because \bar{x} is the local minimum of (OP).

Furthermore, using Taylor's development of f in the neighborhood of $\bar{x} \in C$

$$\begin{aligned} f(\bar{x} - t(x - \bar{x})) &\simeq f(\bar{x}) + \nabla^t f(\bar{x})(\bar{x} + t(x - \bar{x}) - \bar{x}) \\ &\simeq f(\bar{x}) + t \nabla^t f(\bar{x})(x - \bar{x}). \end{aligned}$$

So, we have:

$$0 \leq f(\bar{x} - t(x - \bar{x})) - f(\bar{x}) \simeq t \nabla f(\bar{x})^t (x - \bar{x})$$

which gives inequality (VIP($\nabla f, C$)). □

Lemma 4.2. *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be a differentiable convex function, then:*

$$\bar{x} \text{ is a solution of } \text{VIP}(\nabla f, C) \iff \bar{x} \text{ is a solution of (OP).}$$

Proof. (1) Suppose that \bar{x} is a solution of (OP), then according to Lemma 4.1, \bar{x} is a solution $\text{VIP}(\nabla f, C)$.

(2) Conversely, suppose that \bar{x} is a solution of $\text{VIP}(\nabla f, C)$.

According to the definition of convexity of f in terms of ∇f , we have:

$$\forall x, y \in C : f(x) - f(y) \geq \langle \nabla f(y), x - y \rangle.$$

For $y = \bar{x}$, we get

$$f(x) - f(\bar{x}) \geq \langle \nabla f(\bar{x}), x - \bar{x} \rangle \geq 0,$$

then

$$f(x) - f(\bar{x}) \geq 0, \forall x \in C.$$

So, \bar{x} is a solution of (OP). □

Remark 4.3. – We deduce that the necessary and sufficient condition for \bar{x} to be a solution of (OP) is equivalent for \bar{x} to be a solution of (VIP), with $F = \nabla f$.

- The same results remain true when f is pseudoconvex.
- We are interested in this work to the particular case where we replace (OP) by (LFP) with a pseudoconvex objective function.

Proposition 4.4. *\bar{x} is a solution of (LFP) if and only if \bar{x} is a solution of the following variational inequalities problem:*

$$\text{VIP}(\nabla G, C) \left\{ \begin{array}{l} \text{find } \bar{x} \in C, \text{ such that} \\ \langle \nabla G(\bar{x}), x - \bar{x} \rangle \geq 0 \forall x \in C. \end{array} \right.$$

Proof. It suffices to take $F = \nabla G$ and remark that G is pseudoconvex then use Lemma 4.2. □

In order to solve the obtained variational inequalities problem $\text{VIP}(\nabla G, C)$, we apply the projection algorithm of Grar *et al.* [8].

4.2. Grar’s algorithm

In this part, we give the description of Grar’s algorithm for solving the variational inequalities problems (VIP). For more details see [8].

4.2.1. Description of Grar’s algorithm

Begin algorithm

Initialization

Select any $\sigma, \gamma, \beta \in (0, 1)$ and let $\varepsilon > 0$, be a given tolerance.

Let $x^0 \in C, k = 0$

Compute $z^k = \text{Proj}_C(x^k - \beta F(x^k))$ and $r(x^k, \beta) = x^k - \text{Proj}_C(x^k - \beta F(x^k))$

- **While** $\|r(x^k, \beta)\| > \varepsilon$ **do**

(1) Compute

$$y^k = (1 - \alpha_k)x^k + \alpha_k z^k.$$

Where $\alpha_k = \gamma^j$ with j being the smallest nonnegative integer satisfying:

$$\langle F(x^k - \gamma^j r(x^k, \beta)), r(x^k, \beta) \rangle \geq \sigma \|r(x^k, \beta)\|^2.$$

(2) Compute the current iterate

$$x^{k+1} = Proj_C(x^k - \lambda_k F(y^k)).$$

Where λ_k is chosen that:

$$\langle F(y^k), Proj_C(x^k - \lambda_k F(y^k)) - y^k \rangle \leq 0.$$

(3) Take $k = k + 1$.

– **End while**

End algorithm

5. NUMERICAL TESTS

To give some insight into the behavior of the two projection approaches and compare their effectiveness to each other, we implemented them on some test problems with fixed and variable sizes which are described below.

We used the Matlab Language with a precision $\varepsilon = 10^{-6}$.

We designate by:

- k : The number of iterations required to obtain the solution.
- Time: The execution time in second.

5.1. Examples with fixed size

Example 1

Consider the linear fractional programming problem

$$(LFP) \begin{cases} \min Z = \frac{7x_1 + 9x_2 + 3}{3x_1 + 4x_2 + 2} \\ 2x_1 + 3x_2 \leq 6 \\ 3x_1 + 2x_2 \leq 5 \\ x_1, x_2 \geq 0. \end{cases}$$

Results obtained via Ye–Lustig’s algorithm

Optimal solution found after 3 iterations and 0,0008s

$$x^* = (0.5999, 1.6000)^t.$$

The optimal value is $Z^* = 2,1176$.

Results obtained via Grar’s algorithm

Optimal solution found after 1 iteration and 0,0731s

$$x^* = (0.6000, 1.6000)^t.$$

The optimal value is $Z^* = 2,1176$.

Example 2

Consider the linear fractional programming problem below:

$$(LFP) \begin{cases} \max Z = \frac{x_1 + 2x_2 + 4x_3 + 5x_4 + 8x_5}{2x_1 + 5x_2 + 3x_3 + 4x_4 + 6x_5 + 1} \\ 2x_1 + x_2 + 3x_3 + x_4 + x_5 \leq 15 \\ x_1 + 2x_2 + x_5 \leq 8 \\ 3x_1 + 5x_2 + 2x_4 \leq 10 \\ 2x_2 + 4x_3 + 3x_4 + x_5 \leq 21 \\ x_1, x_2, x_3, x_4, x_5 \geq 0. \end{cases}$$

Results obtained via Ye–Lustig’s algorithm

Optimal solution found after 12 iterations and 0,0253s

$$x^* = (0.00001, 0.00002, 2.33333, 0.00009, 8.00000)^t.$$

The optimal value is $Z^* = 1, 3$.

Results obtained via Grar’s algorithm

Optimal solution found after 9 iterations and 1,3309s

$$x^* = (0.00000, 0.00003, 2.33333, 0.00009, 8.00000)^t.$$

The optimal value is $Z^* = 1, 3$.

Example 3

Consider the linear fractional programming problem

$$(\text{LFP}) \begin{cases} \min Z = \frac{x_1 + x_2 + 1}{5x_1 + x_2 + 1} \\ -5x_1 - 2x_2 \leq -6 \\ x_1 \leq 3 \\ x_2 \leq 3 \\ x_1, x_2 \geq 0. \end{cases}$$

Results obtained via Ye–Lustig’s algorithm

Optimal solution found after 5 iterations and 0,0026s

$$x^* = (3.000, 0.00000)^t.$$

The optimal value is $Z^* = 0, 25$.

Results obtained via Grar’s algorithm

Optimal solution found after 1 iteration and 0,0456s

$$x^* = (3.000, 0.00000)^t.$$

The optimal value is $Z^* = 0, 25$.

Example 4

Consider the linear fractional programming problem below:

$$(\text{LFP}) \begin{cases} \min Z = \frac{\sum_{j=1}^n c_j x_j}{\sum_{j=1}^n d_j x_j} \\ \sum_{j=1}^n x_j = N \\ 0 \leq x_j \leq 1, \quad j = 1, \dots, n. \end{cases}$$

(1) $n = 5$.

$$c^t = (2, -1, -3, 5, -2), \quad d^t = (1, 2, 2, 3, 4), \quad N = 3.$$

Results obtained via Ye–Lustig’s algorithm

Optimal solution found after 5 iterations and 0,0021s

$$x^* = (0.00000, 0.99999, 0.99999, 0.00000, 0.99999)^t.$$

The optimal value is $Z^* = -0, 74999$.

Results obtained via Grar's algorithm

Optimal solution found after 1 iteration and 0,0095s

$$x^* = (0.00000, 1.00000, 1.00000, 0.00000, 1.00000)^t.$$

The optimal value is $Z^* = -0,75$.

(2) $n = 10$.

$$c^t = (1, 1, 1, 2, 2, 3, 0, 0, 1, 1), \quad d^t = (1, 1, 1, 1, 1, 1, 1, 1, 1, 1), \quad N = 5.$$

Results obtained via Ye–Lustig's algorithm

Optimal solution found after 5 iterations and 0,0031s

$$x^* = (0.5999, 0.5999, 0.5999, 0.0000, 0.0000, 0.0001, 0.9999, 0.9999, 0.5999, 0.5999)^t.$$

The optimal value is $Z^* = 0,6$.

Results obtained via Grar's algorithm

Optimal solution found after 1 iteration and 0,0582s

$$x^* = (0.6, 0.6, 0.6, 0.0000, 0.0000, 0.0001, 1.0000, 1.0000, 0.6000, 0.6000)^t.$$

The optimal value is $Z^* = 0,6$.

(3) $n = 15$.

$$c^t = (1, 2, 3, 3, 0, 0, 1, 6, 0, 4, 0, 1, 5, 2, 1).$$

$$d^t = (1, 2, 2, 4, 6, 1, 1, 0, 0, 5, 8, 2, 3, 1, 1), \quad N = 8.$$

Results obtained via Ye–Lustig's algorithm

Optimal solution found after 7 iterations and 0,0066s

$$x^* = (0.9999, 0.0000, 0.0000, 0.0000, 0.9999, 0.9999, 0.9999, 0.0000, 0.9999, 0.0000, 0.9999, 0.9999, 0.0000, 0.0000, 0.9999)^t.$$

The optimal value is $Z^* = 0,2$.

Results obtained via Grar's algorithm

Optimal solution found after 2 iterations and 0,0957s

$$x^* = (0.9999, 0.0000, 0.0000, 0.0000, 1.0000, 0.9999, 0.0000, 1.0000, 0.0000, 1.0000, 1.0000, 0.0000, 0.0000, 0.9999, 0.0000)^t.$$

The optimal value is $Z^* = 0,2$.

The following table summarizes the results obtained *via* the two projection algorithms, Ye–Lustig's algorithm and Grar's algorithm according to different examples.

Example	Ye–Lustig's algorithm		Grar's algorithm	
	k	Time (s)	k	Time (s)
1	3	0.0008	1	0.0731
2	12	0.0253	9	1.3309
3	5	0.0026	1	0.0456
	5	0.0021	1	0.0095
4	5	0.0031	1	0.0582
	7	0.0066	2	0.0957

5.2. Example with variable size

Example 5

Consider the linear fractional programming problem with variable size

$$(\text{LFP}_m) \begin{cases} \max Z = \frac{c^t x + 2m}{d^t x + 1} \\ Ax = b \\ x \geq 0, \end{cases}$$

where

$$\begin{cases} c_i = \begin{cases} -1 + 2m, & i = 1, \dots, m \\ 2m, & i = m + 1, \dots, n \end{cases} \\ d_i = 1, i = 1, \dots, n \quad n = 2m \\ A[i, j] = \begin{cases} 1 & \text{If } j = i \text{ or } j = m + i \\ 0 & \text{Otherwise.} \end{cases} \\ b_i = 2, i = 1, \dots, m. \end{cases}$$

The optimal solution is

$$x_i^* = \begin{cases} 2, & i = 1, \dots, m \\ 0, & i = m + 1, \dots, n. \end{cases}$$

The following table summarizes the results obtained *via* the two projection algorithms, Ye–Lustig’s algorithm and Grar’s algorithm according to different values of m .

Size (m, n)	Ye–Lustig’s algorithm		Grar’s algorithm	
	k	Time (s)	k	Time (s)
(50, 100)	3	0.0516	2	0.1793
(100, 200)	3	0.2941	2	1.0173
(200, 400)	2	1.5522	3	8.5241
(300, 600)	2	5.9558	4	26.0963
(500, 1000)	2	31.0989	5	85.7836
(1000, 2000)	2	247.4904	9	617.5747
(1500, 3000)	2	461.9488	13	2270.8168

Comments

Through the numerical tests and for different dimensions, the results show that:

- For small and medium sized examples, the execution time in Ye–Lustig’s algorithm is sensibly low compared to Grar’s algorithm, although it requires fewer iterations.
- For large sized example, we notice the superiority of Ye–Lustig’s algorithm compared to Grar’s algorithm. Indeed, the number of iterations and the execution time in Ye–Lustig’s algorithm are significantly inferior to those of Grar’s algorithm. We record a significant reduction in execution time in Ye–Lustig’s algorithm.

This shows clearly the practical effectiveness of the interior point methods already known for large size problems.

6. CONCLUSION

In this article, we have applied two different approaches to solve a Linear Fractional Program (LFP). The numerical results obtained by the two methods are encouraging and of very good quality.

On the other hand, the interior points method has shown its success and its advantage in terms of efficiency, robustness and above all its speed (execution time) compared to the second approach.

This contribution confirms the efficiency of interior point methods to solve a non-convex nonlinear optimization problem (LFP).

It opens the way to generalize the use of interior point methods to solve other more complex non-convex problems, such as Fractional Quadratic Programming problem (FQP).

REFERENCES

- [1] M. Babul Hasan and S. Acharjee, Solving LFP by converting it into a single LP. *Int. J. Oper. Res.* **8** (2011) 1–14.
- [2] E.B. Bajalinov, Linear-Fractional Programming Theory, Methods, Applications and Software. Kluwer Academic Publishers, Boston, MA (2003).
- [3] G.R. Bitrand and A.J. Novas, Linear programming with a fractional objective function. *J. Oper. Res.* **21** (1973) 22–29.
- [4] A. Cambini and L. Martein, A modified version of Martos’s algorithm for the linear fractional problem. *Methods Oper. Res.* **53** (1986) 33–44.
- [5] A. Charnes and W.W. Cooper, Programming with linear functional. *Nav. Res. Logist. Q.* **9** (1962) 181–186.
- [6] W. Dinkelbach, On nonlinear fractional programming. *Manage. Sci.* **13** (1967) 492–498.
- [7] P.C. Gilmore and R.E. Gomory, Linear programming approach to the cutting stock problem-part 2. *Oper. Res.* **11** (1963) 863–867.
- [8] H. Grar and D. Benterki, New effective projection method for variational inequalities problem. *RAIRO: OR* **49** (2015) 805–820.
- [9] J.R. Isbell and W.H. Marlow, Attrition games. *Nav. Res. Logist. Q.* **3** (1956) 1–99.
- [10] I.J. Lustig, A practical approach to Karmarkar’s algorithm, Technical report sol 85-5 System optimization laboratory; Department of Operations Research Stanford. University of Stanford, CA (1985).
- [11] I.J. Lustig, Feasibility issues in a primal-dual interior point method for linear programming. *Math. Program.* **49** (1991) 145–162.
- [12] B. Martos, Hyperbolic programming. *Nav. Res. Logist. Q.* **11** (1964) 135–155.
- [13] A. Nagih and G. Plateau, Problèmes fractionnaires: tour d’horizon sur les applications et méthodes de résolution. *RAIRO: OR* **33** (1999) 383–419.
- [14] S.K. Saha, M.R. Hossain, M.K. Uddin and R.N. Mondal, A new approach of solving linear fractional programming problem (LFP) by using computer algorithm. *Open J. Optim.* **4** (2015) 74–86.
- [15] J.K. Sharma, A.K. Gupta and M.P. Gupta, Extension of simplex technic for solving programming problems. *Indian J. Pure Appl. Math.* **11** (1980) 961–968.
- [16] K. Swarup, Linear fractional functional programming. *Oper. Res.* **110** (1962) 380–387.
- [17] K. Swarup, Linear fractional functional programming. *Oper. Res.* **13** (1965) 1029–1036.
- [18] O. Zerdani, L’optimisation non linéaire multiobjectif, Thèse de doctorat, Université Mouloud Mammeri, Tizi-Ouzou (2013).