Artificial Neural Network And New Mathematical Approach To Solve Muti-Objective Linear Fractional Programming Problem

Khaled Elsharkawy

ABSTRACT: A new algorithm based on revised simplex method is designed to solve multiple objectives linear fractional programming (MOLFP), we put a condition for the feasible solution to be efficient that is at every iteration we check if each feasible point is efficient or not. Our algorithm can be used to convert the multi-objective linear fractional programming problem into linear programming problem and hence solving it. A simple example is given to illustrate the theory of the proposed algorithm and a suggestion to the solution using artificial neural network.

Key words: Multi-Objective Linear Programming (MOLP), Multi-Objective linear fractional Programming (MOLFP), Artificial Neural Networks (ANN).

1.INTRODUCTION

MOLFP problems arise when several linear functional objectives are to be optimized over a feasible region X. However for a single objective linear fractional programming the charnes and cooper transformation [2] can be used to transform the problem into linear programming problem. Few approaches have been reported [1, 7] for solving MOLFP problems. Kormbluth and steuer [7] considered this problem and presented a simplex-based solution procedure to find all weakly efficient vertices of the augmented feasible region. Also Benson [1] in his article showed that the procedure suggested by [7] for computing the numbers to find "break points" may not work all the time and he proposed a failsafe method for computing these numbers. In this paper we consider a MOLFP problem and proposed an algorithm for solving this problem based on the revised simplex method. Condition for the efficiency of a feasible solution is also given. The rest of this paper is organized as follows, section 2 is about definitions and notations of MOLP and MOLFP, section 3 is about Characterization of efficiency in MOLFP, section 4 is about the ANN in optimization, section 5 giving numerical example and section 6 gives the conclusion.

2. Definitions and Notations [10, 11, 12]

In a MOLFP problems there are several linear fractional objectives have to be optimized over a convex polytope X [8, 9], this problem can be formulated as:

(MOLFP)

$$Max\, Z(x) = \{z_1(x),..., z_k(x)\}$$

$$S.t \;\; x \in X = \{x \in R^n / Ax = b, x \geq 0\} \qquad (2.1)$$

$$Where \;\; z_1(x) = \frac{c_1^T x + \alpha_1}{d_1^T x + \beta_1} \quad for\; i = 1,2,..,k$$

Here, c_i, d_i are vectors in $R^n, \alpha_1 \; and \; \beta_1$ are scalars, A is an mxn matrix and $b \in R^m$. We also assume that X is a compact set and $d_1^T x + \beta_1 \succ 0$, i= 1, 2, K for every $x \in X$.

Solving MOLFP problem seeks for the set of solutions having the following definitions.

Definition 2.1: A solution x° is said to be efficient for MOLFP if $x^\circ \in X$ and there is no $x \in X$ such that $Z(x) \geq Z(x^\circ) \; and \; Z(x) \neq Z(x^\circ)$.

The neural network may be physically implemented by the use of electronic components or it may be simulated using software running on a personal computer. A neural network can be seen as a parallel implementation of a distributed processor that is composed of a set of simple processing units. These units have an inherit natural tendency to store experimental knowledge, and then make this knowledge available for use.

Definition 2.2: A solution x° is said to be weakly efficient for MOLFP if $x^\circ \in X$ and there is no $x \in X$ such that $Z(x) \geq Z(x^\circ)$.

Consider the MOLP problem [5, 6] $(MOLP)_1$

$$Max\, \bar{z}(x) = \{\bar{z}_1(x), \bar{z}_2(x),..., \bar{z}_k(x)\}$$

$$Subject\, to \;\; x \in X = \{x \in R^n / Ax = b, x \geq 0\} \qquad (2.2)$$

$$Where\; \bar{z}_1(x) = (c_1^T - z_1^\circ d_1^T)x, i = 1,2,..,k.$$

And z_1° is a given constant k vector. Since the level curve of each objective in MOLFP can be written as $(c_1^T - z_1^\circ d_1^T)x = \beta_1 z_1^\circ - \alpha_1$, i=1,2,…,k, we have the following proposition.

Proposition 2.1: If x° solves MOLFP problem with objective function values $z_1(x^\circ) = z_1^\circ$, i=1,..k, then x° solves MOLP defined by (2.2) with objective function value

$$\bar{z}_1(x^\circ) = \beta_1 z_1(x^\circ) - \alpha_i, i = 1,2,..,k.$$

Proof: Straight forward.

Hence one can see a correspondence when solving MOLFP defined by (2.1) and MOLP defined by (2.2) respectively. Consider the MOLP problem and suppose x is a basic feasible solution of Ax=b, $x \geq 0$ with corresponding basic decomposition then

Ax=b, $x \geq 0 \Leftrightarrow Bx_B + NX_N = b, x_B \geq 0, x_N \geq 0$, if we construct a tableau for x in multiple objective form as

$$T(x) = \begin{pmatrix} I & B^{-1}N & B^{-1}b \\ 0 & R & z_B \end{pmatrix} \quad (2.3)$$

Here

$$x = \begin{pmatrix} B^{-1} & b & 0 \end{pmatrix}^T, \; z_B = \begin{cases} \dfrac{\bar{z}_1(x^\circ) + \alpha_1}{\beta_1} & i = 1,2,...,k, \; \beta_1 \neq 0 \\ \dfrac{-\bar{z}_1(x^\circ) + c_1^T(x^\circ)}{(d_1^T x^\circ)} & i = 1,2,...,k, \; \beta_1 = 0 \end{cases}$$

And R is a k.(n-m) matrix of reduced cost coefficient for each of the objectives

$$R = (c_{1N}^T - c_{1B}^T B^{-1} N) - z_1(x^\circ)(d_{1N}^T - d_{1B}^T B^{-1} N), i = 1,2,..,k \quad (2.4)$$

Consider the tableau $T(x^\circ)$ for a given basic feasible solution x°, then x° is efficient if the system

$$((c_{1N}^T - c_{1B}^T B^{-1} N) - z_1(x^\circ)(d_{1N}^T - d_{1B}^T B^{-1} N)) \; u_N \leq 0, u_N \geq 0 \quad (2.5)$$

Is consistent where u_N is an n-m vector and we have the following theorem.

Theorem 2.1:

If x° is a basic feasible solution of MOLP problem (2.2) with corresponding tableau $T(x^\circ)$ then x° is efficient for MOLFP problem (2.1) if there exist a k vector $\lambda \succ 0$ *such that*

$$\lambda^T R \leq 0. \quad (2.6)$$

Proof:

This theorem can be proved by Motzkin's theorem of the alternative [10], that the system (2.5) has no solution if there exist $\lambda \in R^k$ such that $\lambda^T R \leq 0$ holds.

Remark 2.1: If r^j denotes the j^{th} column of R then x° is inefficient if there exist a column aa $r^j \in R$ *and* $r^j \geq 0.$

Remark 2.2: The tableau in (2.3) reduces to the canonical tableau of MOLP problem if all

$d_1, i = 1,2,..,k$ are equal to zero. And our proposed algorithm to find an initial efficient solution can be summarized in the following steps:

Step 1: start with a feasible point x°

Step 2: construct the tableau $T(x^\circ)$

Step 3: solve the system $\lambda^T R \leq 0$, if there exist a k vector $\lambda \succ 0$ then we get the first efficient solution x°, otherwise choose another feasible point x^1 and go to step 2.

3. Characterization of efficiency in MOLFP [3, 4, 22, 23]

Suppose that x° is an efficient solution of MOLFP then we have the following theorem.

Theorem 3.1:

Consider $(MOLP)_2$

$$\begin{aligned} &\max \; F(x) = \{ f_1(x), f_2(x), ..., f_k(x) \} \\ &\textit{Subject to } \; x \in X = \{ x \in R^n / Ax = b, \; x \geq 0 \} \qquad (3.1) \\ &\textit{Where } \; f_1(x) = (c_1^T (d_1^T x^\circ) - d_1^T (c_1^T x^\circ)) x, \; i = 1,2,..,k. \end{aligned}$$

If x° is efficient solution for MOLFP then x° solves (3.1) with vector value equal zero.

Proof:

Suppose x° solves the MOLFP problem defined by (2.1) then for each level curve, we have

$(c_1^T - z_1^\circ d_1^T) x = \beta_1 z_1^\circ - \alpha_1$, i=1,2,…,k

$$(c_1^T - (\frac{c_1^T x + \alpha_1}{d_1^T x + \beta_1} d_1^T) x^\circ = \beta_1 \, z_1^\circ - \alpha_1$$

$$(c_1^T(d_1^T x^\circ + \beta_1) - (c_1^T x^\circ + \alpha_1)\, d_1^T))\frac{x^\circ}{d_1^T x^\circ + \beta} = \beta_1\, z_1^\circ - \alpha_i$$

and we have

$$(c_1^T(d_1^T x^\circ) - d_1^T(c_1^T x^\circ))\frac{x^\circ}{d_1^T x^\circ + \beta} + \beta_1\, z_1^\circ - \alpha_i = \beta_1\, z_1^\circ - \alpha_i$$

Hence, we conclude that, if x° is efficient then

$$(c_1^T(d_1^T x^\circ) - d_1^T(c_1^T x^\circ))\frac{x^\circ}{d_1^T x^\circ + \beta} = 0.$$

Let the j^{th} column of R is denoted by r^j and starting from an efficient tableau $T(x^\circ)$.

Recall that pivoting in the j^{th} column in the tableau $T(x^\circ)$ from a new basis with corresponding basic feasible solution $\bar{x}$ which is an extreme point of X. If θ is the standard scalar defined by the simplex algorithm [9] then

$$\frac{c_1^T\,\bar{x} + \alpha_1}{d_1^T\,\bar{x} + \beta_1} = \frac{c_1^T\,x^\circ + \alpha_1}{d_1^T x^\circ + \beta_1} - \theta\, r^{-j}$$

Where r^{-j} in the j^{th} column of R with each component of it is multiplied by $\dfrac{1}{d_1^T\,\bar{x} + \beta_1}, i = 1,2,.., k.$

If we denote the edge of X determined by pivoting column into the basis be E^j and this edge will be efficient edge [6] if and only if the linear program

$$\max\{w = e^T s\}$$
$$Subjectto\ \{Ru = s + r^j, u \geq 0, s \geq 0.\}$$

Has maximum objective value w=0. Suppose that when pivoting in column j an adjacent efficient $\bar{x}$ of X to x° exists, this adjacent efficient solution can be found by solving the linear program $(MOLP)_3$

$$\begin{aligned} &\max \theta \\ &Subjectto\ Ax = b \\ &(c_1^T - z_1^\circ d_1^T)\, x + \theta\{(c_{1B}^T B^{-1} N_j - c_{1N_j}) - z_1^\circ (d_{1B}^T B^{-1} N_j - d_{1N_j})\} = \beta_1 z_1^\circ - \alpha_1,\ i = 1,2,.., k. \quad (3.2) \\ &x \geq 0,\ \theta \geq 0. \end{aligned}$$

The above consideration can be used in the following procedure for obtaining all efficient extreme points of X.

Step 1: let x° be an efficient solution of X with corresponding tableau $T(x^\circ)$.

Step 2: for each $r^{-j} \in R$ satisfying $\bar{R}u \geq r^{-j}$ solve the linear programming $(MOLP)_3$ to find all adjacent extreme points to x°

4. Integration between ANN and operational research

Neural networks have often been formulated to solve difficult optimization problems. A problem is not an optimization problem if we can practically enumerate all theAlternatives and apply an evaluation formulation to each one, also optimization Techniques are used in the field of neural networks. Many learning algorithms are based on minimization of error function [13]. There are two basic approaches to use neural network, the more popular approach is to formulate combinatorial optimizationtask in terms of minimizing a cost or energy function that involve either binary variables [14] or continuous variables. Neural network models have been developed or interpreted As minimization machines, before using network to solve problem one must express The problem as mathematical function that is to be minimized. The other basic approach is to design competition-based neural networks in which neurons are completed to be active under certain conditions. The interconnected networks of analog processors can be used for solution of constrained optimization problems including the linear and non-linear programming problems. In multi-criteria decision maker realm, Jun Wang [15] develops a neural network approach for modeling decision maker's fuzzy preferences structures. The result shows that the proposed neural network approach is capable of modeling a variety of fuzzy preferences structures. On the other hand [16] propose a new interactive procedure for solving MOPP based upon FFANN. The computational Results indicate that the interactive FFANN procedure produces good solutions and Is robust with regard to the neural network architecture, in 1995 [17] presents a neural Network technique based on back-propagation learning algorithm under monotonic Function constrains for modeling multi-criteria multi-person decision making problems. Another work has been done by [18], who describes how the effectiveness of visual Interactive simulation model can be enhanced if all results from the simulation are Stored and used to train a neural network, also ANN has been used in managerialapplications such as bank failure prediction [12], classification problem in discriminate Analysis [19] and the decision tree to evaluate different alternatives [20]. From this Survey, we can decide that the ANN succeed in solving some complexities of decisionmaking problems. In addition it increases the effectiveness and capabilities of the Available mathematical programming models, although the disadvantages of applying artificial neural networks (ANN) in this field can be summarized into two points:

1-It takes a long time in training process, especially with large amount of data.

2-It needs many historical data to be trained.

5. Illustrative example

	x_1	x_2	x_3	x_4	solution
x_3	1	0	1	0	4
x_4	0	1	0	1	4
	0	-1	-2	0	-8
	0	1	-3	0	-12
	0	$\frac{1}{3}$	$\frac{3}{5}$	0	2

This example is taken from [24]. Consider the MOLFP problem

$$\max z_1(x) = -2x_1 + x_2$$
$$\max z_2(x) = -3x_1 - x_2$$
$$\max z_3(x) = \frac{x_1 + x_2 - 2}{-x_1 + 2x_2 + 5}$$
$$Subjectto\ \{x_1 \le 4, x_2 \le 4, x_i \ge 0,\ i = 1,2.\}$$

5.1. Using artificial neural network

We can solve the above numerical example using many

	x_1	x_2	x_3	x_4	solution
x_3	1	0	1	0	4
x_4	0	1	0	1	4
	0	0	-2	1	-4
	0	0	-3	-1	-16
	0	0	$\frac{5}{39}$	$\frac{-1}{3}$	$\frac{2}{3}$

methods, one of these methods is our new mathematical approach but transfer this problem to weighting sum approach problem P (w) so our problem becomes:

$$P(w):\ \max Z(x) = [w_1 Z_1(x) + w_2 Z_2(x) + w_3 Z_3(x)]$$
$$x \in X, where\ X\ is \{x_1 \le 4, x_2 \le 4, x_i \ge 0, i = 1,2\}$$

.

By choosing different values for w_1, w_2 and w_3 we can solve the problem algebraically also we can give these Values to the ANN to train it then solve the weighting problem and we shall concentrate about this point in another article.

$$\left\{\begin{array}{l} x^1 = (0,0),\ z_1 = (0,0,\frac{-2}{5}) \\ -2\lambda_1 - 3\lambda_2 + \frac{3\lambda_3}{5} \le 0 \\ \lambda_1 - \lambda_2 + \frac{9\lambda_3}{65} \le 0 \end{array}\right\}$$

5.2. Using our new mathematical approach

For this example the four extreme points

$$x^1 = (0,0), x^2 = (4,0), x^3 = (4,4), x^4 = (0,4)$$

are all efficient extreme points for this problem, there corresponding tableau are given below

	x_1	x_2	x_3	x_4	solution
x_3	1	0	1	0	4
x_4	0	1	0	1	4
	2	-1	0	0	0
	3	1	0	0	0
	$\frac{-3}{5}$	$\frac{-9}{65}$	0	0	$\frac{-2}{5}$

$$\left\{\begin{array}{l} x^2 = (4,0),\ z_2 = (-8,-12,2) \\ 2\lambda_1 + 3\lambda_2 - \frac{3\lambda_3}{5} \le 0 \\ \lambda_1 - \lambda_2 - \frac{\lambda_3}{3} \le 0 \end{array}\right\}$$

$$\left\{\begin{array}{l} x^3 = (4,4),\ z_3 = (-4,-16,\frac{2}{3}) \\ 2\lambda_1 + 3\lambda_2 - \frac{5\lambda_3}{39} \le 0 \\ -\lambda_1 + \lambda_2 + \frac{\lambda_3}{3} \le 0 \end{array}\right\}$$

	x_1	x_2	x_3	x_4	solution
x_3	1	0	1	0	4
x_4	0	1	0	1	4
	2	0	0	1	4
	3	0	0	-1	-4
	$\frac{-5}{39}$	0	0	$\frac{9}{65}$	$\frac{2}{13}$

$$\left\{\begin{array}{l} x^4 = (0,4),\ z_4 = (4,-4,\frac{2}{13}) \\ -2\lambda_1 - 3\lambda_2 + \frac{5\lambda_3}{39} \le 0 \\ -\lambda_1 + \lambda_2 - \frac{9\lambda_3}{65} \le 0 \end{array}\right\}$$

6. Conclusion

In this paper, we introduced a new method to solve multi-objective fractional linear programming problem, first we introduce the mathematical model of MOLFP and give a several definitions about the efficient solutions solved this model then we give a MOLP model and give a several definitions about the efficient solutions solved this model, we introduced theorems to correlate the MOLFP and MOLP models then deduce algorithm to deduce initial efficient solution solved the final MOLP, finally we introduced a procedure to deduce all the extreme points solved the final MOLFP, we then introduced a numerical example satisfying all these procedure, in the future i hope to use FF-ANN to solve MOLFP directly (may be as I discussed in section 5.1).

REFERENCES

(Periodical style)

[1] Benson, H. P," Finding certain weakly-efficient vertices in multiple objective linear fractional programming", Management Science, vol. 3, no.2, 1985, pp, 240-245.

[2] Charnes, A., & Cooper, W. W," Programming with linear fractional functional. Naval Research Logistics", 9(3-4), 1962, pp, 181-186.

[3] Moumita, D., & De, P. K, "A note on fuzzy multi-objective linear fractional programming problem", International Journal of Computer Science and Network, vol.3, no. 6, 2014, pp, 568-572.

[4] Muruganandam, S., & Ambika, P, "Harmonic mean technique to solve multi objective fuzzy linear fractional programming problems", Global Journal of Pure and Applied Mathematics, vol. 13, no.10, 2017,pp, 7321-7329.

[5] Jain, S, "Modeling of Gauss elimination technique for multi-objective linear programming problem", Journal of Novel Applied Sciences, 1, 2012, pp, 25-29.

[6] Ecker J. G, and Hegner N.S and Kouada I.A," Generating all maximal efficient faces for multiple objective linear programs", Journal of optimization Theory and Applications, vol.30, (1980), pp.353-381.

[7] Kornbluth, J. S., & Steuer, R. E, "Multiple objective linear fractional programming", Management Science, vol.27, no.9, 1981, pp,1024-1039.

[8] Jain, S, "Modeling of Gauss elimination technique for multi-objective fractional programming problem", South Asian Journal of Mathematics, vol.4, no.3, 2014, pp, 148-153.

[9] Guzel, N., & Sivri, M, "Proposal of a solution to multi objective linear fractional programming problem", Sigma Journal of Engineering and Natural Sciences, vol.2, 2005, pp, 43–50.

[10] Chen, S. H., & Hsieh, C. H," Graded mean integration representation of generalized fuzzy number", Journal of Chinese Fuzzy Systems, vol.5, no.2, 1999, pp,1-7.

[11] Dheyab, A. N," Finding the optimal solution for fractional linear programming problems with fuzzy numbers", Journal of Kerbala University, vol.10, no.3, 2012, pp, 105-110.

[12] 12.Kar Yau tam and Melody Y. Kiang,"Managerial

[13] Applicationsof neural networks: The case of bank

[14] Failure productions", Management science, vol.38,

[15] No.7, 1992, pp, 926-947.

[16] 13.A. Cichocki, R. Unbehauen, K. Weinsierl and R.

[17] Holzel, "A new neural network for solving linear

[18] programming problems", En.ropean J. o/

[19] Opernt*onaf ftesenrcñ, vol. 93, 1996, pp,244—256.

[20] 14. W.S. McCulloch and W.A. Pitts, "A logical calculus of

[21] the ideas immanent in nervous activity", Bul learn of

[22] Motfiernntics rind Biophpsics, vol. 5, 1943, pp,

[23] 115-133.

[24] 15. Jun Wang, "A neural network approach to modeling

[25] Fuzzy preference relations for multiple criteria

[26] Decision making", computers and operations research,

[27] Vol.21, No.9, 1994, pp, 991-1000.

[28] 16. B. Shirazi and S. Yih, "Critical analysis of applying

[29] Hopfield neural net model to optimization problems"

[30] IEEE International conference on systems, Man and

[31] Gybereneli's, 14-17, vol.1, 1989, pp, 210-215.

[32] S.H. Zak, V. Upatising and S. Hui, "Solving linear programming problems with neural networks", A comparative study, IEEE Wansactions on Neural Networks 6 , vol.1, 1995, pp, 94—104.

[33] M. Sun, A. Stam and R.E. Steuer," Solving multiple objective programming problems using feed-forward artificial neural networks". The interactive FFANN procedure, Management Science 42, vol.6, 1996, pp, 835—849.

[34] M. Gen, K. Ida and R. Kobuchi," Neural network technique for fuzzy multi-objective linear programming", Computers arid industrial Engineering 35, vol. (3—4), 1998, pp, 543—546.

[35] E.S. Lee and R.J. Li," Fuzzy multiple objective programming and compromise programming with Pareto optimum", Fuzzy Sets and Systems vol.53, 1993, pp, 275—288

[36] (Thesis or Dissertation style)
[37] Guzel, N, "A proposal to the solution of multiobjective linear fractional programming problem", Abstract and Applied Analysis (Vol. 2013), Hindawi Publishing Corporation
[38] Bajalinov, E. B, "Linear-fractional programming: theory, methods, applications and software", Kluwer Academic Publishers, Boston, 2003.
[39] Steuer R.E.,"Multiple criteria optimization", Theory computations and applications, Wiley, New York, 1986.