

Feasible Region Contraction Interior-Point Algorithm (FERCIPA) Solver for Multi-Objective Linear Programming Problems

Edwin F. Nsien¹, Ubon A. Abasiokwere^{2*}, Paul J. Udoh³

^{1,2}Department of Mathematics and Statistics, University of Uyo, Uyo, Nigeria

³Department of Mathematics, University of Michigan, Ann Arbor, USA

Abstract:- This paper presents FERCIPA solver for linear programming problems. The solver which can handle both single objective and multi-objective linear programming problems of large scales generates a sequence of interior feasible points that converge at the optimal solution for single objective linear programming problems and an optimal compromise solution for multi-objective linear programming problems. The solver is validated by its application to handle single objective linear programming problems and multi-objective linear programming problems involving up to six bounded variables and functional constraints. The solution obtained by FERCIPA solver is seen to compare favourably with those of other software like the Feasible Region Contraction Algorithm (FRCA) and MATLAB.

Keywords:- FERCIPA Solver, Multi-objective linear programming, Interior feasible point, Optimal compromise solution.

I. INTRODUCTION

The history of linear optimization goes back to 1827 when Fourier solved the problem of finding solution of a system of linear inequalities. His method was based on elimination of variables and he made an n^3 algorithm to find a feasible point or states that there is no feasible point when there is no feasible solution to the problem. This procedure was forgotten and rediscovered by Dines in 1918 and by Motzkin in 1936. This algorithm became Fourier-Motzkin algorithm which actually should be Fourier-Dines-Motzkin and it is similar to Gaussian elimination [12]. However, the algorithm is slow compared to interior point techniques. This method remained important even after the development of the simplex method since it is capable of stating the existence or nonexistence of a feasible point and also gives all the optimal solutions of a problem in integer linear programming [13].

Linear programming is, without doubt, the most popular tool used in an operations research study. This is attested to by the number of computer software that are available for solving linear programming problems. Some of these computer programs are based on the Simplex algorithm and its variants, e.g. EXCEL, MATLAB, LINGO, CPLEX, TORA, Optimizer in Corel Q Pro, etc. [1]

while others are based on the interior-point algorithms, e.g. MOSEX [2, 3]. Even though the simplex algorithm and its variants have the software based on them, it has enjoyed a general acceptance and usage in solving linear programming problems. They solve linear programming problem in exponential time. An algorithm that solves linear programming problem in polynomial time was the interior-point algorithm developed by Karmarkar [4]. Since then, there has been a growing interest in the interior-point method for solving linear programming problems [5-7].

The method of solving large-scale linear programming problems by the interior-point method under MATLAB environment was presented by Zhang [8]. The existing interior-point algorithms have some drawbacks such as extensive calculation requirements, a large number of iterations and large computer space requirements [9]. The methods of solving large multi-objective linear programming problems by interior-point method were presented by Pandian and Jayalakshmi and Stanimirovic *etal* [10, 11].

In this paper, we present a software called FERCIPA SOLVER capable of solving both single objective and multi-objective linear programming problems. This new software provides the computerized implementation of Feasible Region Contraction Interior-point Algorithm (FERCIPA).

The user interface for FERCIPA SOLVER application consists of a menu bar, an objective function data panel and a constraint equation data panel. The menu bar provides the user with access to the various application functions and commands. The objective function data panel displays the objective function(s) that have been specified for the current linear programming model. It provides command buttons that allow the addition of new objective function, the editing of a selected objective functions and the removal of a specified objective function. The constraints data panel displays the constraint equations that have been specified for the current LP model. It provides command buttons that allow the addition of new constraint equations, the editing of a selected constraint equation and the removal of a specified constraint. At the bottom of the user interface is a solve command button which the user can use to prompt the application to attempt the solving of the LP problem.

The FERCIPA solver has features that include: Solving of both single and multi-objective linear programming problems; automated generation of the corresponding weights attached to multi-objective functions for a given problem; loading and saving of data associated with the given linear programming problem to a storage device; modification of data values as required; load function that allows the loading of previously saved linear programming model data from a storage device; the specification of any number of decision variables for a given linear programming problem, saving of the solutions to a given linear programming problem, generating and printing of linear programming problem solutions.

II. PROCEDURE FOR IMPLEMENTATION

Here, we provide a step by step procedure for setting up and solving of single and mul-ti-objective linear programming problems using the FERCIPA solver. The solver is the computer software that provides the computerized implementation of FERCIPA.

- Step 1: Set up a linear programming problem (single objective or multi-objective)
- Step 2: Click on the LP Model Menu item and select the New Option.
- Step 3: In the displayed dialog window,
 - input the number of decision variables in the model and click on the Ok but-ton
 - To create an objective function, click on the Add button associated with the objective function panel, input the coefficients for each decision variable in the objective function and click on the Ok button. The entered function will then be displayed in the objective function panel. Repeat the above steps to add other objective functions.
 - To create a constraint, click on the Add button associated with the constraint panel. In the displayed dialog, input the coefficients for the decision variable and the right-hand value. Then select the appropriate equality sign and Click on the Ok button. The entered equation will be displayed in the constraints panel. Repeat the above steps to add other constraints.
- Step 4:
 - *Edit objective function*
Select an objective function in the objective function panel by clicking on it, then click on the Edit button. A dialog box with the values of the selected objective function will be displayed. Edit as required and click on the Ok button to update the model.
 - *Edit constraints*
Select a constraint in the constraints panel by clicking on it, then click on the Edit button. The dialog box with the values of the selected constraint will be displayed. Edit as required and click on the Ok button to update the model.

➤ Step 5:

- *Remove an objective function.*

Select an objective function in the objective function panel by clicking on it, then click on the Remove button. Confirm the removal of the function by selecting Yes in the confirmation dialog box. Note that this operation cannot be undone.

- *Remove a constraint.*

Select a constraint in the constraints panel by clicking on it, then click on the Remove button. Confirm the removal of the constraint by selecting Yes in the confirmation dialog. Note that this operation cannot be undone.

➤ Step 6: Solving a model

To solve the LP model, click on the LP Model menu item and select the Solve option or click on the Solve button at the bottom of the screen. The application will attempt to solve the model and a report will be generated.

➤ Step 7: Saving a model

An LP model that has been created in the application can be saved for later use. To do this, click on the File menu item and select the Save option. In the displayed dialog box, specify a file name and, optionally, select a location for the file. Click on the save button.

➤ Step 8: Loading a model

To load a previously saved model, click on the File menu item and select the Load op-tion. In the displayed dialog box, browse to the folder containing the saved file, select the file, click on the Open button and the model will be loaded from the file.

III. APPLICATION OF THE FERCIPA SOLVER

Consider the following single objective linear programming problem:

$$\text{Maximize } Z = 25x_1 + 7x_2 + 24x_3$$

s. t.

$$3x_1 + x_2 + 5x_3 \leq 8000000$$

$$5x_1 + x_2 + 3x_3 \leq 5000000$$

$$x_1, x_2, x_3 \geq 0$$

The detailed report (output) produced by the FERCIPA solver for single objective linear programming problem is shown below and summarized in Table 1. The CPU time in this study was gotten from a Windows PC with 2GHz Intel processor and 2GB of RAM.

➤ LP Problem

$$\text{Maximize } Z=25X_1+7X_2+24X_3$$

$$\text{Subject to: } 3X_1+X_2+5X_3 \leq 8000000$$

$$5X_1+X_2+3X_3 \leq 5000000$$

$$X_i \geq 0, i = 1, 2, 3$$

➤ Iteration 1

	X₁	X₂	X₃	Z₁
Z₁ : X*	0	500000	1500000	39500000

$$x^* = [0, 500000, 1500000]$$

$$Z_1 = 39500000$$

➤ Time Taken
0.125 seconds

➤ Solution

Problem	Decision variable	Objective function value	Number of iterations	CPU time (seconds)
Linear Programming	(0, 500000, 1500000)	39500000	1	0.125

Table 1:- Summary of Results Using Fercipa on Single Objective Lp Problems.

Consider the following multi-objective linear programming problems:

$$\text{Molp 1 : Max } z_1 = 2x_1 + 3x_2$$

$$\text{Max } z_2 = 3x_1 - x_2$$

$$\text{Max } z_3 = 4x_1 + 3x_2$$

$$\text{s.t. } x_1 + x_2 \leq 20$$

$$2x_1 + x_2 \leq 30$$

$$x_2 \leq 12$$

$$x_i \geq 0$$

$$\text{Molp2 : Max } z_1 = 2x_1 + x_2 + 3x_3 + x_4$$

$$\text{Max } z_2 = x_1 + 3x_2 + 2x_3 + 1.5x_5$$

$$\text{Max } z_3 = 2.5x_1 + 4x_2 + 1.5x_3 + 3x_6$$

subject to :

$$x_1 + 2x_2 + 3x_3 + x_4 \leq 30$$

$$2x_1 + x_2 + x_3 + 3x_5 \leq 35$$

$$x_1 + x_2 + x_4 + x_6 \leq 16$$

$$x_i \geq 0, i = 1, 2, \dots, 6.$$

The detailed report (output) produced by the FERCIPA solver for multi-objective linear programming problem 1 is shown below and summarized in Table 2.

➤ MOLP Problem 1

$$\text{Maximize } Z_1 = 2X_1 + 3X_2$$

$$\text{Maximize } Z_2 = 3X_1 - X_2$$

$$\text{Maximize } Z_3 = 4X_1 + 3X_2$$

Subject to:

$$X_1 + X_2 \leq 20$$

$$2X_1 + X_2 \leq 30$$

$$X_2 \leq 12$$

$$X_i \geq 0, i = 1, 2$$

➤ Iteration 1

	X₁	X₂	Z₁	Z₂	Z₃
Z₁ : X*	8	12	52	12	68
Z₂ : X*	15	0	30	45	60
Z₃ : X*	10	10	50	20	70

Weights

W₁	W₂	W₃
0.4333	0.2	0.3667

$$x^* = [10.1333, 8.8667]$$

➤ Iteration 2

	X₁	X₂	Z₁	Z₂	Z₃
Z₁ : X*	10.3067	9.3867	48.7733	21.5333	69.3867
Z₂ : X*	15	0	30	45	60
Z₃ : X*	10.3067	9.3867	48.7733	21.5333	69.3867

➤ Solution

$$x^* = [10.3067, 9.3867]$$

$$Z_1 = 48.7733$$

$$Z_2 = 21.5333$$

$$Z_3 = 69.3867$$

➤ Time Taken
0.124 seconds

The detailed report (output) produced by the FERCIPA solver for multi-objective linear programming problem 2 is shown below and summarized in Table 2.

➤ MOLP Problem 2

$$\text{Maximize } Z_1 = 2X_1 + X_2 + 3X_3 + X_4$$

$$\text{Maximize } Z_2 = X_1 + 3X_2 + 2X_3 + 1.5X_5$$

$$\text{Maximize } Z_3 = 2.5X_1 + 4X_2 + 1.5X_3 + 3X_6$$

Subject to:

$$X_i \geq 0, i = 1, 2, \dots, 6$$

$$X_1 + 2X_2 + 3X_3 + X_4 \leq 30$$

$$2X_1 + X_2 + X_3 + 3X_5 \leq 35$$

$$X_1 + X_2 + X_4 + X_6 \leq 16$$

➤ Iteration 1

	X₁	X₂	X₃	X₄	X₅	X₆	Z₁	Z₂	Z₃
Z₁ : X*	15.1667	0	4.6667	0.8333	0	0	45.1667	24.5	44.9167
Z₂ : X*	0	15	0	0	6.6667	1	15	55	63
Z₃ : X*	0	15	0	0	0	1	15	45	63

Weights

W₁	W₂	W₃
0.3748	0	0.6252

$$x^* = [5.6842, 9.3782, 1.749, 0.3123, 0, 0.6252]$$

➤ Iteration 2

	X₁	X₂	X₃	X₄	X₅	X₆	Z₁	Z₂	Z₃
Z₁ : X*	13.4604	7.6981	0.3812	0	0	-5.1585	35.7623	37.3169	49.5396
Z₂ : X*	6.153	9.847	1.3843	0	3.8209	0	26.306	44.194	56.847
Z₃ : X*	0	3.694	7.5373	0	7.4402	12.306	26.306	37.3169	63

Weights

W₁	W₂	W₃
0.429	0.2696	0.3014

➤ Solution

$$x^* = [7.4334, 7.0708, 2.8083, 0, 3.2725, 1.4958]$$

$$Z_1 = 30.3626$$

$$Z_2 = 39.1713$$

$$Z_3 = 55.5666$$

➤ Time Taken

0.125 seconds

Problem	Decision variable	Objective function value			Number of iterations	CPU time (seconds)
		Z1	Z3	Z2		
MOLP 1	(10.3067,9.3867)	48.7733	21.5333	69.3867	2	0.124
MOLP 2	(7.43,7.07,2.81,0,3.27,1.50)	30.3626	39.1713	55.5666	2	0.250

Table 2:- Summary of Results Using Fercipa on Multi-Objective Lp Problems

IV. COMPARISON OF FERCIPA WITH FRCA AND MATLAB

The performance of FERCIPA software is compared with the popular interior-point algorithm operating under FRCA and MATLAB environment. The modes of

comparison are the objective function value, values of decision variables and number of iterations. Table 3 shows the comparison of the three methods using the single-objective and multi-objective linear programming problems presented in this work.

Mode	FERCIPA	FRCA	MATLAB
Single Objective Linear Programming			
Decision variable	(0, 500000, 1500000)	(0.0011, 499999, 1500000)	
Objective function	$z = 39500000$	$z = 39499999.99$	
No. of iterations	1	12	
Multi-objective-MOLP 1			
Decision variable	(10.3067, 9.3867)		As known from experiment performed
Objective function	$Z_1 = 48.77, Z_2=21.53, Z_3=69.38$		
No. of iterations	2		
Multi-objective-MOLP 2			
Decision variable	(7.43,7.07,2.81,0,3.27,1.5)		As known from experiment performed
Objective function	$Z_1 = 30.36, Z_2=39.17, Z_3=55.56$		
No. of iterations	2		

Table 3:- Comparison of the Three Software

V. CONCLUSION

Terlaky and Boggs [9] showed the interior point algorithm are more efficient than the simplex algorithm when applied to a large-scale linear programming problem, but less efficient when applied to a small-scale linear programming problem. As it can be seen in the above table, FERCIPA performs better than FRCA for single objective linear programming problem and better than MATLAB for multi-objective linear programming problems. Hence FERCIPA by implication performs better in single, multi-objective and large-scale linear programming problems.

The FERCIPA solver presented in this work is an efficient software capable of solving both single and multi-objective large-scale linear programming problems with certain restrictions. It is programmed to solve linear programming problems only. The solver, in finding the optimal solution, generates a sequence of interior-points which converge at the optimal solution for the single-objective linear programming problem and generates a sequence of weights attached to multi-objective linear programming problem.

REFERENCES

- [1]. H.A. Taha, *Operations Research. An Introduction*. New Delhi, Prentice Hall of India, 2006.
- [2]. R. Fourer, *Survey of linear programming software*. OR/MS Today, 42(3) 2001; 58-68
- [3]. E.O. Effanga and I.O. Isaac, A Feasible Region Contraction Algorithm (FRCA) for Solving Linear Programming Problems. *Journal of Mathematics Research*, 3(3) (2011), 159-166.
- [4]. N. Karmarkar, A New Polynomial-Time Algorithm for Linear Programming, *Combinatorica*, 4, (1984), 373-395.
- [5]. T. Kosaki, An Interior-point method for minimizing the sum of piecewise-linear convex functions. *Department of Industrial and Management, Tokyo Institute of Technology*, (2010), 1-9.
- [6]. M. Ehrgott, L. Shao and A. Schobel, An Approximation algorithm for convex

- multiobjective programming problems. *Journal of Global Optimization*, 50 (2011), 397-416.
- [7]. M.L. Bournol, J.H. Dula and P. Rouse, Interior-point methods in DEA to determine non-zero multiplier weight. *Computer and Operations Research*, 39 (2012), 698-708.
- [8]. Y. Zhang, Solving large scale linear programming by interior point methods under MATLAB environment, *Technical Report, Mathematics. Department of Maryland Baltimore Country*, 1996.
- [9]. T. Terlaky and P.T. Boggs, Interior-point methods. In: *Encyclopedia of Operations Research and Management Science*. Springer, Boston, MA, 2001.
- [10]. P. Pandian and M. Jayalakshmi, Determining Efficient Solution to Multiple Objective Linear Programming Problems, *Applied Mathematical Sciences*, 7(26) (2013), 1275-1282.
- [11]. I.P. Stanimirovic, M.L. Zlatanovic and M.D. Petkovic, On the linear weighted sum method for multi-objective optimization, *Facta Acta University*, 26 (2011), 49-63.
- [12]. P. Parrilo and S. Lall, Fourier-motzkin elimination, Stanford University, (2003), 1–29.
- [13]. G. B. Dantzig and B. C. Eaves, Fourier–motzkin elimination and its dual, *Journal of Combinatorial Theory*, (A) (1973), 288–297.