An Improved Affine-Scaling Interior Point Algorithm for Linear Programming

Douglas Kwasi Boah, Stephen Boakye Twum

Department of Mathematics, Faculty of Mathematical Sciences, University for Development Studies, Navrongo, Ghana
Email: boahdouglas@yahoo.com, stwum@uds.edu.gh

Abstract

In this paper, an Improved Affine-Scaling Interior Point Algorithm for Linear Programming has been proposed. Computational results of selected practical problems affirming the proposed algorithm have been provided. The proposed algorithm is accurate, faster and therefore reduces the number of iterations required to obtain an optimal solution of a given Linear Programming problem as compared to the already existing Affine-Scaling Interior Point Algorithm. The algorithm can be very useful for development of faster software packages for solving linear programming problems using the interior-point methods.

Keywords

Interior-Point Methods, Affine-Scaling Interior Point Algorithm, Optimal Solution, Linear Programming, Initial Feasible Trial Solution

1. Introduction

The Simplex Method (SM) remained a popular solution method of practical linear programming (LP) problems, until the development of interior point methods. [1] was the pioneer in the field and his Projective Scaling Method was able to compete with the SM as applied to realistic problems. As the name suggests, interior point methods approach an optimal point (which is known to be on the boundary of the feasible set) through a sequence of interior points [2]. Unlike the SM, iterates are calculated not on the boundary, but in the interior of the feasible region. Starting with an initial interior point, the method moves through the interior of the feasible set along an improving direction to another interior point. There, a new improving direction is found, along which a move is made to yet another interior point. This process is repeated, resulting in a sequence of interior points that converge to an optimal boundary point. Many different types

of interior-point methods for linear programming have been developed. Most of the methods fall under one of the three main categories: the projective and potential reduction method, affine-scaling method and path-following methods [3]. In this paper, an Improved Affine-Scaling Interior Point Algorithm of LP has been proposed with the view of increasing the efficiency of the original algorithm due to [4].

2. Materials and Methods

The Affine-Scaling Interior-Point Algorithm was first introduced by [4]. He subsequently published a convergence analysis in [5]. Dikin's work went largely unnoticed for many years until [6] [7] [8] and [9] rediscovered it as a simple variant of Karmarkar's algorithm. Here, the problem is rescaled in order to make the initial point stay some distance away from any boundary constraint and then restrict the step length, so that the next move will not reach the boundary. The algorithm is as follows.

Given an optimization problem in standard form:

Optimize $Z = c^\mathrm{T} x$

Subject to $Ax = b$

$$x \geq 0,$$

where c, A and b are the cost coefficients, technological coefficients and resource availability respectively, the Affine-Scaling Interior-Point Algorithm is summarized in the following steps:

Step 1: Given the initial trial solution, $x = (x_1, x_2, \cdots, x_n)^\mathrm{T}$, set

$$D = \begin{bmatrix} x_1 & 0 & 0 & \cdots & 0 \\ 0 & x_2 & 0 & \cdots & 0 \\ 0 & 0 & x_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & x_n \end{bmatrix}$$

Step 2: Calculate $\tilde{A} = AD$ and $\tilde{c} = Dc$.

Step 3: Calculate $P = I - \tilde{A}^\mathrm{T}\left(\tilde{A}\tilde{A}^\mathrm{T}\right)^{-1}\tilde{A}$ and $C_p = P\tilde{c}$ where P is a projection matrix and C_p is a projected gradient.

Step 4: Identify the negative component of C_p having the largest absolute value, and set v to this absolute value. Then Calculate

$$\tilde{x} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} + \frac{\theta}{v} C_p, \tag{1.0}$$

where $0 < \theta < 1$ [4].

Step 5: Calculate $x = D\tilde{x}$ as the trial solution for the next iteration starting from step 1.

(If this trial solution is virtually unchanged from the preceding one, then the algorithm has virtually converged to an optimal solution and the algorithm is

terminated) [10].

3. Results and Discussions

The Improved Affine-Scaling Interior-Point Algorithm

In the Affine-scaling interior-point algorithm [4] discussed above, the selected constant, θ in Equation 1.0 is required to be such that $0 < \theta < 1$. Thus, according to [4], the possible θ values should exclude 0 and 1. The selected constant, θ measures the fraction used of the distance that could be moved before the feasible region is exited [10]. [5] published convergence analysis of the method using $\theta = 0.5$. [11] used $\theta = 2/3$ in their convergence result. [10] used θ values of 0.5 and 0.9 in their Interactive Operations Research (IOR) software. In this study, an investigation into the consequence of θ value of one (1) on the algorithm has been undertaken. Subsequently, it has been observed that, θ value of one (1) gives the least number of iterations of a given LP problem. The observation has led to an improved Affine-scaling interior-point algorithm which is the same the Affine-scaling interior-point algorithm [4] but with θ values now given as $0 < \theta \leq 1$.

Table 1(a) and Table 1(b) present some computational results of selected practical problems affirming the proposed algorithm. The tables specify the LP problems, selected θ values (with their corresponding number of iterations in brackets) and their optimal solutions using a developed Interior-Point Program based on the Affine-Scaling Interior Point Algorithm which was written in MATLAB. To obtain the optimal solution of any LP problem in standard form, the developed program requires the user to input the initial feasible trial solution (which gives the diagonal matrix), cost coefficients, the number of columns/rows of the identity matrix, technological coefficients and the selected constant.

Table 1. (a): Computational results of selected practical problems affirming the proposed algorithm; (b): Computational Results of selected practical problems affirming the proposed algorithm.

(a)

LP Problem	θ values and their corresponding number of iterations in brackets		Optimal Solution
Maximize $Z = 6X_1 + 8X_2$ Subject to $X_1 + 2X_2 \leq 12$ $X_1 + X_2 \leq 10$ $X_1, X_2 \geq 0$	$0.1 \rightarrow (79)$ $0.3 \rightarrow (24)$ $0.5 \rightarrow (12)$ $0.7 \rightarrow (8)$ $0.9 \rightarrow (6)$	$0.2 \rightarrow (38)$ $0.4 \rightarrow (17)$ $0.6 \rightarrow (10)$ $0.8 \rightarrow (7)$ $1.0 \rightarrow (2)$	$Z = 64$ $X_1 = 8$ $X_2 = 2$
Maximize $Z = 2X_1 + 3X_2$ Subject to $X_1 + X_2 \geq 350$ $2X_1 + X_2 \leq 600$ $X_1 \geq 125$ $X_1 X_2 \geq 0$	$0.1 \rightarrow (92)$ $0.3 \rightarrow (29)$ $0.5 \rightarrow (17)$ $0.7 \rightarrow (11)$ $0.9 \rightarrow (8)$	$0.2 \rightarrow (45)$ $0.4 \rightarrow (21)$ $0.6 \rightarrow (12)$ $0.8 \rightarrow (10)$ $1.0 \rightarrow (2)$	$Z = 1300$ $X_1 = 125$ $X_2 = 350$

Continued

Minimize $Z = 3X_1 + 2X_2$	$0.1 \rightarrow (90)$	$0.2 \rightarrow (43)$		
Subject to	$0.3 \rightarrow (27)$	$0.4 \rightarrow (20)$		$Z = 13$
$5X_1 + X_2 \geq 10$	$0.5 \rightarrow (15)$	$0.6 \rightarrow (11)$		$X_1 = 1$
$X_1 + X_2 \geq 6X_1 + 4X_2 \geq 12$	$0.7 \rightarrow (10)$	$0.8 \rightarrow (9)$		$X_2 = 5$
$X_1 X_2 \geq 0$	$0.9 \rightarrow (7)$	$1.0 \rightarrow (2)$		
Minimize $Z = 20X_1 + 10X_2$				
Subject to	$0.1 \rightarrow (90)$	$0.2 \rightarrow (43)$		
$X_1 + 2X_2 \leq 40$	$0.3 \rightarrow (27)$	$0.4 \rightarrow (20)$		$Z = 240$
$3X_1 + X_2 \geq 30$	$0.5 \rightarrow (15)$	$0.6 \rightarrow (11)$		$X_1 = 6$
$4X_1 + 3X_2 \geq 60$	$0.7 \rightarrow (10)$	$0.8 \rightarrow (9)$		$X_2 = 12$
$X_1 X_2 \geq 0$	$0.9 \rightarrow (7)$	$1.0 \rightarrow (2)$		
Maximize $Z = 16X_1 + 17X_2 + 10X_3$				
Subject to				
$X_1 + X_2 + 4X_3 \leq 2000$	$0.1 \rightarrow (122)$	$0.2 \rightarrow (58)$		$Z = 20625$
$2X_1 + X_2 + X_3 \leq 3600$	$0.3 \rightarrow (37)$	$0.4 \rightarrow (26)$		$X_1 = 30$
$X_1 + 2X_2 + 2X_3 \leq 2400$	$0.5 \rightarrow (19)$	$0.6 \rightarrow (15)$		$X_2 = 1185$
$X_1 \leq 30$	$0.7 \rightarrow (12)$	$0.8 \rightarrow (11)$		$X_3 = 0$
$X_1, X_2, X_3 \geq 0$	$0.9 \rightarrow (10)$	$1.0 \rightarrow (3)$		

(b)

LP Problem	θ values and their corresponding number of iterations in brackets		Optimal Solution
Minimize $Z = 1.06X_1 + 0.56X_2 + 300X_3$ $+ 2703.50X_4 + 4368.23X_5$ Subject to $1.06X_1 + 0.015X_2 \geq 729824.87$ $0.56X_2 + 0.649X_3 \geq 1522188.03$ $3.00X_3 \geq 5040.16$ $2703.50X_4 \geq 162210.06$ $4368.23X_5 \geq 17472.92$ $X_1, X_2, X_3, X_4, X_5 \geq 0$	$0.1 \rightarrow (130)$ $0.3 \rightarrow (42)$ $0.5 \rightarrow (22)$ $0.7 \rightarrow (15)$ $0.9 \rightarrow (13)$	$0.2 \rightarrow (62)$ $0.4 \rightarrow (29)$ $0.6 \rightarrow (16)$ $0.8 \rightarrow (14)$ $1.0 \rightarrow (5)$	$Z = 2435620.485$ $X_1 = 688490.254$ $X_2 = 2716245.849$ $X_3 = 1680.053$ $X_4 = 60.000$ $X_5 = 4.000$
Minimize $Z = 2.03X_1 + 0.56X_2 + 2.93X_3$ $+ 1543.85X_4 + 1494.14X_5$ Subject to $2.03X_1 + 0.015X_3 \geq 3604.90$ $0.56X_2 + 0.633X_3 \geq 430264.03$ $2.93X_3 \geq 750.50$ $1543.85X_4 \geq 26245.39$ $1494.14X_5 \geq 5976.56$ $X_1, X_2, X_3, X_4, X_5 \geq 0$	$0.1 \rightarrow (130)$ $0.3 \rightarrow (42)$ $0.5 \rightarrow (22)$ $0.7 \rightarrow (15)$ $0.9 \rightarrow (13)$	$0.2 \rightarrow (62)$ $0.4 \rightarrow (29)$ $0.6 \rightarrow (16)$ $0.8 \rightarrow (14)$ $1.0 \rightarrow (5)$	$Z = 466675.399$ $X_1 = 1773.920$ $X_2 = 768039.091$ $X_3 = 256.143$ $X_4 = 17.000$ $X_5 = 4.000$

It is seen from Table 1(a) and Table 1(b) that, the number of iterations decreases as θ values increase, and that $\theta = 1$ gives the least number of iterations. Since θ value of one (1) gives the least number of iterations, it should be included in the algorithm as proposed above to increase efficiency of the algo-

rithm.

4. Conclusion

An Improved Affine-Scaling Interior Point Algorithm for Linear Programming has been proposed. Computational results of selected practical problems affirming the proposed algorithm have been provided. The proposed algorithm is accurate, faster and therefore reduces the number of iterations required to obtain an optimal solution of a given Linear Programming (LP) problem as compared to the already existing Affine-Scaling Interior Point Algorithm. The algorithm can be very useful for development of faster software packages for solving linear programming problems using the interior-point methods.

Future Work

In this paper, computational results of selected practical problems affirming the proposed algorithm have been provided. We hope to provide a rigorous proof of the algorithm in the near future.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Karmarkar, N. (1984) Polynomial-Time Algorithm for Linear Programming. *Combinatorica*, **4**, 373-395. https://doi.org/10.1145/800057.808695

[2] Eiselt, H.A. and Sandblom, C.L. (2007) Linear Programming and Its Applications. Springer-Verlag, Berlin Heidelberg, 280.

[3] Singh, J.N. and Singh, D. (2002) Interior-Point Methods for Linear Programming: A Review. *International Journal of Mathematical Education in Science and Technology*, **33**, 405-423. https://doi.org/10.1080/00207390110119934

[4] Dikin, I.I. (1967) Iterative Solution of Problems of Linear and Quadratic Programming. *Soviet Mathematics Doklady*, **8**, 674-675.

[5] Dikin, I.I. (1974) On the Speed of an Iterative Process. *Upravlyaemye Sistemi*, **12**, 54-60.

[6] Karmarkar, N. and Ramakrishnan, U. (1985) Further Developments in the New Polynomial-Time Algorithm for Linear Programming. Talk Given at ORSA/TIMS National Meeting, Boston.

[7] Barnes, E.R. (1986) A Variation on Karmarkar's Algorithm for Solving Linear Programming Problems. *Mathematical Programming*, **36**, 174-182. https://doi.org/10.1007/BF02592024

[8] Vanderbei, R.J., Meketon, M.S. and Freedman, B.A. (1986) A Modification of Karmarkar's Linear Programming Algorithm. *Algorithmica*, **1**, 395-407. https://doi.org/10.1007/BF01840454

[9] Adler, I., Resende, M., Veiga, G. and Karmarkar, N. (1989) An Implementation of Karmarkar's Algorithm for Linear Programming. *Mathematical Programming*, **44**,

297-335. https://doi.org/10.1007/BF01587095

[10] Hillier, F.S. and Lieberman, G.J. (2010) Introduction to Operations Research. Ninth Edition, McGraw-Hill Higher Education, New York, 195-259.

[11] Tsuchiya, T. and Muramatsu, M. (1995) Global Convergence of a Long-Step Affine Scaling Algorithm for Degenerate Linear Programming Problems. *SIAM Journal on Optimization*, **5**, 525-551. https://doi.org/10.1137/0805027