

Computer Oriented Interior Point Algorithm for Solving Linear Programming Problem with Application

Farhana Ahmed Simi* and Md. Ainul Islam

Department of Mathematics, Dhaka University, Dhaka-1000, Bangladesh

(Received: 17 April 2016; Accepted: 3 October 2016)

Abstract

In this paper, we study the interior point algorithm for solving linear programming (LP) problem developed by Narendra Karmarkar. As interior point algorithm for LP problem involves tremendous calculations, it is quite impossible to do so by hand calculation. To fulfill the requirement we develop computer code in MATLAB for LP which is based on this algorithm procedure. To illustrate the purpose, we formulate a real life sizeable large-scale linear program for diet problem and solve it using our computer code for interior point algorithm in MATLAB.

Keywords: LP, Interior point algorithm, MATLAB.

I. Introduction

An interior point algorithm is one that improves a feasible interior solution point of the linear program by steps through the interior, rather than one that improves by steps around the boundary of the feasible region, as the classical simplex algorithm does. For large scale problems the number of vertices is quite large, hence the simplex method would be prohibitively expensive in computer time if any substantial fraction of the vertices had to be evaluated. But by shooting through the interior of the feasible region, the interior point method tends to require a substantially smaller number of iterations^{1,2}.

The earliest interior point method is due to the famous mathematician John Von Neumann³. His method for finding a feasible solution to a linear program with a convexity constraint is notable for its simplicity and remarkable convergence properties. Since a general linear program combined with its dual can be reformulated into feasibility problem of this restricted form, Von Neumann's algorithm³ may be viewed as a method for solving the general linear program.

Now a day's interior point methods for linear programming eventually transformed the entire field of optimization. Despite its momentous impact on the field, Karmarkar⁴ proposed an interior point algorithm.

In this paper, we first briefly discuss the Karmarkar⁴ interior point algorithm and process of transforming LP problem into Karmarkar⁴ canonical form with an example in Section II and III respectively. Hence we formulate a sizable large-scale linear program for diet problem in Section IV and solve this problem by our computer oriented interior point algorithm in MATLAB in Section V. Finally a conclusion is drawn in favor of our work in Section VI.

II. Karmarkar Interior Point Algorithm

In this section, we present the Karmarkar's⁴ algorithm. This algorithm reduces the general LP problem

$$\begin{aligned} &\text{Minimize } f(x) = c^T x \\ &\text{subject to, } Ax \geq 0 \\ &\quad \quad \quad x \geq 0 \end{aligned}$$

into the following form:

$$\begin{aligned} &\text{Minimize } f(x) = c^T x \\ &\text{subject to,} \end{aligned}$$

$$\begin{aligned} Ax &= 0 \\ \sum x_i &= 1 \\ x &\geq 0 \end{aligned} \tag{1}$$

Here $x = (x_1, x_2 \dots x_n)^T \in R^n, c \in Z^n, A \in Z^{m \times n}$.

The LP problem (1) above in canonical form can be obtained from a standard form LP problem:

$$\begin{aligned} &\text{Minimize } c^T x \\ &\text{subject to,} \\ &\quad Ax = b \\ &\quad x \geq 0 \end{aligned} \tag{2}$$

Steps of the algorithm

The algorithm starts on the canonical form (2) and from the center of the simplex $a_0 = (\frac{1}{n}) e$, where $e = (1, 1 \dots 1)^T$ generates a sequence of iterates $x^{(0)}, x^{(1)} \dots x^{(k)}$ in the following steps:

Step 1. Set $x^{(0)}$ = center of the simplex

$$a_0 = \left(\frac{1}{n}\right) e$$

Step 2. To compute the next point, define the function $b = \varphi(a)$, where $b = x^{(k+1)}$ and $a = x^{(k)}$ and performing the following sequence of operations:

Substep 1. Let $D = \text{diag} \{a_1, a_2 \dots a_n\}$ be the diagonal matrix whose i th diagonal entry is a_i .

Substep 2. Let $B = \begin{bmatrix} AD \\ e^T \end{bmatrix}$, that is augment the matrix AD with a row of all $\mathbf{1}$'s.

Substep 3. Compute the orthogonal projection^{5,6} of Dc into the null space of B .

$$c_p = [I - B^T(BB^T)^{-1}B]Dc.$$

Substep 4. Normalize $c_p: \hat{c} = \frac{c_p}{|c_p|}$, that is \hat{c} is the unit vector in the direction of c_p .

Substep 5. $b' = a_0 - \alpha r \hat{c}$, that is take a step of length αr in the direction \hat{c} , where r is the radius of the largest inscribed sphere $r = \frac{1}{\sqrt{n(n-1)}}$ and $\alpha \in (0, 1)$.

Substep 6. To return b , applying the inverse projective transformation⁵ $b = \frac{Db'}{e^T Db'}$.

Step 3. The current iterate $x^{(k)}$ will be the optimal solution if $\frac{c^T x^{(k)}}{c^T x^{(0)}} \leq \epsilon$.

*Author for correspondence. e-mail:simi_96@du.ac.bd

III. Transformation of an LP Problem to the Canonical Form

In this section, we explain the steps of the transformation of LP problem into a canonical form with a simple example^{7,8}. The basic steps:

Consider a linear programming problem

$$\begin{aligned} &\text{Minimize } Z=c^T x \\ &\text{subject to, } Ax \geq b \\ &x \geq 0. \end{aligned}$$

Step 1. Combining primal and dual problems

$$\begin{aligned} Ax &\geq b \\ A^T u &\leq c \\ c^T x - b^T u &= 0 \\ x \geq 0, u &\geq 0. \end{aligned}$$

By duality theory, this combined problem is feasible if and only if the original problem has finite optimum solution.

Step 2. Introducing slack variables

$$\begin{aligned} Ax - y &= b \\ A^T u + v &= c \\ c^T x - b^T u &= 0 \\ x \geq 0, u \geq 0, v \geq 0, y \geq 0. \end{aligned}$$

Step 3. Introducing an artificial variable to create an interior starting point. Let x_0, y_0, u_0, v_0 be strictly interior points in the positive orthant.

$$\begin{aligned} &\text{Minimize } \lambda \\ &\text{subject to,} \\ &Ax - y + (b - Ax_0 + y_0)\lambda = b \\ &A^T u + v + (c - A^T u_0 - v_0)\lambda = c \\ &c^T x - b^T u + (-c^T x_0 + b^T v_0)\lambda = 0 \\ &x \geq 0, u \geq 0, v \geq 0, y \geq 0, \lambda \geq 0. \end{aligned}$$

Here $x = x_0, y = y_0, u = u_0, v = v_0, \lambda = 1$ is a strictly interior feasible solution which is a starting point. The minimum value of λ is zero if and only if the problem in Step 2 is feasible.

Step 4. Rewrite the problem in Step 3 as

$$\begin{aligned} &\text{Minimize } c^T x \\ &\text{Subject to, } Ax = b \\ &x \geq 0 \end{aligned}$$

$x=a$ is a known strictly interior starting point, and the target value of the objective function is zero.

Step 5. Consider the transformation $x' = T(x)$ where $x \in R^n, x' \in R^{n+1}$ are defined by

$$\begin{aligned} x_i' &= \frac{x_i/a_i}{\sum_j (x_j/a_j) + 1}, i = 1, 2 \dots n \\ x_{n+1}' &= 1 - \sum_{i=1}^n x_i' \end{aligned} \quad (3)$$

The inverse transformation is given by

$$x_i = \frac{a_i x_i'}{x_{n+1}'}, i = 1, 2 \dots n$$

Step 6. A_i denote i^{th} column of A . If

$$\sum_{i=1}^n A_i x_i = b$$

Then

$$\frac{\sum A_i a_i x_i'}{x_{n+1}'} = b$$

Or,

$$\sum_{i=1}^n A_i a_i x_i' - b x_{n+1}' = 0$$

Karmarkar⁴ defines the columns of a matrix A' by these equations:

$$A_i' = a_i A_i, i = 1, 2, \dots, n$$

$$A_{n+1}' = -b.$$

Then $\sum_{i=1}^{n+1} A_i' x_i' = 0$.

Let Ω' denote this affine subspace, i.e.

$$\Omega' = \{x' \in R^{n+1} \mid A'x' = 0\}.$$

This is a system of homogeneous equations in the transformed space.

Step 7. Transforming the original space

$Z = \{x \in R^n \mid c^T x = 0\}$ into

$Z' = \{x' \in R^{n+1} \mid c'^T x' = 0\}$ by substituting

$$x_i = \frac{a_i x_i'}{x_{n+1}'}$$

That is, $\sum_{i=1}^{n+1} \frac{c_i a_i x_i'}{x_{n+1}'} = 0$.

Here

$$\begin{aligned} c_i' &= a_i c_i, \quad i = 1, 2 \dots n \\ c_{n+1}' &= 0 \end{aligned}$$

The transformed problem is

Minimize $c'^T x'$

Subject to,

$$\begin{aligned} A'x' &= 0 \\ \sum_{i=1}^{n+1} x_i' &= 1 \\ x' &\geq 0 \end{aligned}$$

This is the required canonical form.

Example

Consider the LP problem

$$\begin{aligned} &\text{Minimize } f(x) = x_1 + x_2 \\ &\text{subject to, } x_1 + 2x_2 \geq 2 \\ &x_1, x_2 \geq 0 \end{aligned}$$

Step 1. We write the dual of the given problem

$$\begin{aligned} &\text{Maximize } g(u) = 2u_1 \\ &\text{subject to,} \end{aligned}$$

$$\begin{aligned} u_1 &\leq 1 \\ 2u_1 &\leq 1 \\ u_1, u_2 &\geq 0 \end{aligned}$$

Combining the primal and dual problem

$$\begin{aligned} x_1 + 2x_2 &\geq 2 \\ u_1 &\leq 1 \\ 2u_1 &\leq 1 \\ x_1 + x_2 - 2u_1 &= 0 \\ x_1, x_2, u_1, u_2 &\geq 0 \end{aligned}$$

Step 2. Introducing slack and surplus variable

$$\begin{aligned} x_1 + 2x_2 - y_1 &= 2 \\ u_1 + v_1 &= 1 \\ 2u_1 + v_2 &= 1 \\ x_1 + x_2 - 2u_1 &= 0 \\ x_1, x_2, y_1, u_1, v_1, v_2 &\geq 0 \end{aligned}$$

Step 3. Introducing an artificial variable λ to create an interior starting point.

Minimize λ

subject to,

$$\begin{aligned} x_1 + 2x_2 - y_1 + (2 - 1 - 2 + 1)\lambda &= 2 \\ u_1 + v_1 + (1 - 1 - 1)\lambda &= 1 \\ 2u_1 + v_2 + (1 - 2 - 1)\lambda &= 1 \\ x_1 + x_2 - 2u_1 + (-1 - 1 + 2)\lambda &= 0 \\ x_1, x_2, y_1, u_1, v_1, v_2, \lambda &\geq 0 \end{aligned}$$

Or, Minimize λ

subject to,

$$\begin{aligned} x_1 + 2x_2 - y_1 &= 2 \\ u_1 + v_1 - \lambda &= 1 \\ 2u_1 + v_2 - 2\lambda &= 1 \\ x_1 + x_2 - 2u_1 &= 0 \\ x_1, x_2, y_1, u_1, v_1, v_2, \lambda &\geq 0 \end{aligned}$$

Step 4. We rewrite the problem in Step 3 as

Minimize x_7

subject to:

$$\begin{aligned} x_1 + 2x_2 - x_3 &= 2 \\ x_4 + x_5 - x_7 &= 1 \\ 2x_4 + x_6 - 2x_7 &= 1 \\ x_1 + x_2 - 2x_4 &= 0 \\ x_1, x_2, x_3, x_4, x_5, x_6, x_7 &\geq 0 \end{aligned}$$

Where,

$$y_1 = x_3, u_1 = x_4, v_1 = x_5, v_2 = x_6, \lambda = x_7.$$

Let $x = a = (a_1, a_2, a_3, a_4, a_5, a_6, a_7) = (1, 1, 1, 1, 1, 1, 1)^T$ is an interior starting point, and the target value of the objective function we are interested in is zero.

Step 5. Consider the transformation $x' = T(x)$, where $x \in R^7, x' \in R^8$ are defined by $x_i' = \frac{x_i}{\sum_{i=1}^7 x_i + 1}$ $i = 1, 2 \dots 7$

$$x_8' = 1 - \sum_{i=1}^7 x_i' = \frac{1}{\sum_{i=1}^7 x_i + 1} \quad [\text{From (3)}]$$

Thus we have, $\sum_{i=1}^8 x_i' = 1$.

And the inverse transformation is

$$x_i = \frac{a_i x_i'}{x_8'} \quad i = 1, 2 \dots 7$$

Using this transformation into the above problem, we get,

Minimize x_7'

subject to,

$$\begin{aligned} x_1' + 2x_2' - x_3' - 2x_8' &= 0 \\ x_4' + x_5' - x_7' - x_8' &= 0 \\ 2x_4' + x_6' - 2x_7' - x_8' &= 0 \\ x_1' + x_2' - 2x_4' &= 0 \\ x_i' &\geq 0, \quad i = 1, 2 \dots 8 \end{aligned}$$

The transformed problem is now

Minimize x_7'

subject to,

$$\begin{aligned} x_1' + 2x_2' - x_3' - 2x_8' &= 0 \\ x_4' + x_5' - x_7' - x_8' &= 0 \\ 2x_4' + x_6' - 2x_7' - x_8' &= 0 \\ x_1' + x_2' - 2x_4' &= 0 \\ x_1' + x_2' + x_3' + x_4' + x_5' + \\ & \quad x_6' + x_7' + x_8' = 1 \\ x_i' &\geq 0, \quad i = 1, 2 \dots 8 \end{aligned}$$

which is the canonical form of the given LP.

IV. Diet Problem

One of the classic applications of linear programming is diet problem. The main goal is to select a set of foods that meets certain daily nutritional requirements and preferences at minimum cost. Consider the following list of foods with their nutritional profile:

Table 1. Nutritional requirement of foods

Food	Portion size	Energy Kcal	Proteins gram	Calcium mg	Price (Tk/ portion)	Limit (portion/ day)
Oats	28g	110	4	2	40	4
Chicken	100g	200	32	12	100	3
Eggs	2 big ones	160	13	54	20	2
Milk	237 cc	160	8	280	50	8
Fish	170g	450	4	22	200	2
Beans	260g	260	14	80	20	2
Donut	60g	239	4	27.6	90	2
Oat Bran	57g	145	6	0.8	40	2
Yogurt	125g	119	6	190	40	3
Chili	216g	190	14	80	80	2
Broccoli	140g	49	3	56	75	3
Apple	182g	95	0	10.9	65	4
Orange	140g	69	1	60.2	70	4
Lentils	100g	116	9	19	115	2
Carrots	28g	10	0	9	35	6
Brussels	78g	28	2	16.8	90	6
Blueberries	28g	16	0	1.7	85	3
Spinach	180g	41	5	245	65	4
Banana	118g	105	1	5.9	30	3
Rice	100g	130	2.4	2	35	3
White bread	3 pieces	65	5	0	40	3
Corn	100g	72	3	2	45	2
Brownies	60g	400	35	28	150	2
Cheese cake	50g	500	40	35	150	2
Ice Cream	200 cc	200	20	15	100	1

We wish to propose a diet containing at least 2,000 (Kcal), at least 55 grams of protein and 800 (mg) of calcium. In addition, to provide some variety in the diet, some limits are set for the daily portions of food. The problem is to find the lowest cost associated with meeting the above requirements.

Formulation

The three basic steps in constructing a LP model as follows:

Step 1. (Identify the decision variables)

For this problem the unknown variables are the set of foods to include in the daily diet. Let

- x_1 = Portions of Oats to eat during the day.
- x_2 = Portions of Chicken to eat during the day.
- x_3 = Portions of Eggs to eat during the day.
- x_4 = Portions of Milk to eat during the day.
- x_5 = Portions of Fish to eat during the day.
- x_6 = Portions of Beans to eat during the day.
- x_7 = Portions of Donut to eat during the day.
- x_8 = Portions of Oat Bran to eat during the day.
- x_9 = Portions of Yogurt to eat during the day.
- x_{10} = Portions of Chili to eat during the day.
- x_{11} = Portions of Broccoli to eat during the day.
- x_{12} = Portions of Apple to eat during the day.
- x_{13} = Portions of Orange to eat during the day.
- x_{14} = Portions of Lentils to eat during the day.
- x_{15} = Portions of Carrots to eat during the day.
- x_{16} = Portions of Brussels to eat during the day.
- x_{17} = Portions of Blueberries to eat during the day.
- x_{18} = Portions of Spinach to eat during the day.
- x_{19} = Portions of Banana to eat during the day.
- x_{20} = Portions of Rice to eat during the day.
- x_{21} = Portions of White Bread to eat during the day.
- x_{22} = Portions of Corn to eat during the day.
- x_{23} = Portions of Brownies to eat during the day.
- x_{24} = Portions of Cheese Cake to eat during the day.
- x_{25} = Portions of Ice Cream to eat during the day.

Step 2. (Identify the constraints)

In this problem constraints are the daily nutritional requirements and preferences for foods.

1. Minimum Calories (Kcal):

$$100x_1 + 200x_2 + 160x_3 + 160x_4 + 450x_5 + 260x_6 + 239x_7 + 145x_8 + 119x_9 + 190x_{10} + 49x_{11} + 95x_{12} + 69x_{13} + 116x_{14} + 10x_{15} + 28x_{16} + 16x_{17} + 41x_{18} + 105x_{19} + 130x_{20} + 65x_{21} + 72x_{22} + 400x_{23} + 500x_{24} + 200x_{25} \geq 2000$$

2. Minimum Proteins (gram):

$$4x_1 + 32x_2 + 13x_3 + 8x_4 + 4x_5 + 14x_6 + 4x_7 + 6x_8 + 6x_9 + 14x_{10} + 3x_{11} + 0x_{12} + x_{13} + 9x_{14} + 0x_{15} + 2x_{16} + 0x_{17} + 5x_{18} + x_{19} + 2.4x_{20} + 5x_{21} + 3x_{22} + 35x_{23} + 40x_{24} + 20x_{25} \geq 55$$

3. Minimum Calcium (mg):

$$2x_1 + 12x_2 + 54x_3 + 280x_4 + 22x_5 + 80x_6 + 27.6x_7 + 0.8x_8 + 190x_9 + 80x_{10} + 56x_{11} + 10.9x_{12} + 60.2x_{13} + 19x_{14} + 9x_{15} + 16.8x_{16} + 1.7x_{17} + 245x_{18} + 5.9x_{19} + 2x_{20} + 0x_{21} + 2x_{22} + 28x_{23} + 35x_{24} + 15x_{25} \geq 800$$

4. Diet variety:

$$\begin{aligned} x_1 &\leq 4 \\ x_2 &\leq 3 \\ x_3 &\leq 2 \\ x_4 &\leq 8 \\ x_5 &\leq 2 \\ x_6 &\leq 2 \\ x_7 &\leq 2 \\ x_8 &\leq 2 \\ x_9 &\leq 3 \\ x_{10} &\leq 2 \\ x_{11} &\leq 3 \\ x_{12} &\leq 4 \\ x_{13} &\leq 4 \\ x_{14} &\leq 2 \\ x_{15} &\leq 6 \\ x_{16} &\leq 6 \\ x_{17} &\leq 3 \\ x_{18} &\leq 4 \\ x_{19} &\leq 3 \\ x_{20} &\leq 3 \\ x_{21} &\leq 3 \\ x_{22} &\leq 2 \\ x_{23} &\leq 2 \\ x_{24} &\leq 2 \\ x_{25} &\leq 1 \end{aligned}$$

5. Non negativity:

$$x_i \geq 0, i = 1, 2, \dots, 25$$

Step 3. (Identify the objective function)

The objective is to minimize the cost of daily diet. So we have

$$\begin{aligned} \text{Minimize } f(x) &= 40x_1 + 1005x_2 + 20x_3 + 50x_4 + 200x_5 + 20x_6 + 90x_7 + 405x_8 + 40x_9 + 80x_{10} + 75x_{11} + 65x_{12} + 70x_{13} + 115x_{14} + 35x_{15} + 90x_{16} + 85x_{17} + 65x_{18} + 30x_{19} + 35x_{20} + 40x_{21} + 45x_{22} + 150x_{23} + 150x_{24} + 100x_{25} \end{aligned}$$

Hence we have the following LP model:

$$\begin{aligned} \text{Minimize } f(x) &= 40x_1 + 1005x_2 + 20x_3 + 50x_4 + 200x_5 + 20x_6 + 90x_7 + 405x_8 + 40x_9 + 80x_{10} + 75x_{11} + 65x_{12} + 70x_{13} + 115x_{14} + 35x_{15} + 90x_{16} + 85x_{17} + 65x_{18} + 30x_{19} + 35x_{20} + 40x_{21} + 45x_{22} + 150x_{23} + 150x_{24} + 100x_{25} \end{aligned}$$

subject to,

$$100x_1 + 200x_2 + 160x_3 + 160x_4 + 450x_5 + 260x_6 + 239x_7 + 145x_8 + 119x_9 + 190x_{10} + 49x_{11} + 95x_{12} + 69x_{13} + 116x_{14} + 10x_{15} + 28x_{16} + 16x_{17} + 41x_{18} + 105x_{19} + 130x_{20} + 65x_{21} + 72x_{22} + 400x_{23} + 500x_{24} + 200x_{25} \geq 2000$$

$$4x_1 + 32x_2 + 13x_3 + 8x_4 + 4x_5 + 14x_6 + 4x_7 + 6x_8 + 6x_9 + 14x_{10} + 3x_{11} + 0x_{12} + x_{13} + 9x_{14} + 0x_{15} + 2x_{16} + 0x_{17} + 5x_{18} + x_{19} + 2.4x_{20} + 5x_{21} + 3x_{22} + 35x_{23} + 40x_{24} + 20x_{25} \geq 55$$

$$2x_1 + 12x_2 + 54x_3 + 280x_4 + 22x_5 + 80x_6 + 27.6x_7 + 0.8x_8 + 190x_9 + 80x_{10} + 56x_{11} + 10.9x_{12} + 60.2x_{13} +$$

$$19x_{14} + 9x_{15} + 16.8x_{16} + 1.7x_{17} + 245x_{18} + 5.9x_{19} + 2x_{20} + 0x_{21} + 2x_{22} + 28x_{23} + 35x_{24} + 15x_{25} \geq 800$$

$$x_1 \leq 4$$

$$x_2 \leq 3$$

$$x_3 \leq 2$$

$$x_4 \leq 8$$

$$x_5 \leq 2$$

$$x_6 \leq 2$$

$$x_7 \leq 2$$

$$x_8 \leq 2$$

$$x_9 \leq 3$$

$$x_{10} \leq 2$$

$$x_{11} \leq 3$$

$$x_{12} \leq 4$$

$$x_{13} \leq 4$$

$$x_{14} \leq 2$$

$$x_{15} \leq 6$$

$$x_{16} \leq 6$$

$$x_{17} \leq 3$$

$$x_{18} \leq 4$$

$$x_{19} \leq 3$$

$$x_{20} \leq 3$$

$$x_{21} \leq 3$$

$$x_{22} \leq 2$$

$$x_{23} \leq 2$$

$$x_{24} \leq 2$$

$$x_{25} \leq 1$$

$$x_i \geq 0, i = 1, 2 \dots 25$$

The problem consists of 25 decision variables and 28 constraints. So if we want to solve this problem by Karmarkar's⁴ algorithm by hand calculation, it will be quite difficult for us to solve it. Since we have to first transform the problem into canonical form, it involves 108 decision variables and 54 constraints which cannot be solved by hand calculation. To overcome these difficulties, we develop computer oriented interior point algorithm in MATLAB. We solve this problem by our computer code in MATLAB stated as follows:

V. Computer Code for Interior Point Algorithm

function f=interior(A, b, c)

% Solve the linear programming problem

% min F(x)=c*x

% subject to: A x (<=, =, >=) b, x >= 0

% using the interior point algorithm.

% [op, ns]= interior(A, b, c)

% c: coefficients of the objective function

% x: initial point (must be feasible)

% A: constraint matrix

% b: constraint vector

% mexit: maximum number of iterations (default = 100)

% op: optimal point

% ns: number of iterations

tic

[m, n] = size(A);

% dual of A

t=A';

% combining primal and dual problem

I1=eye(m);

fori=1:size(A)

k(i,:)= [A(i,:) -I1(i,:) 0*ones(1,m) 0*ones(1,n) b(i,:)-
sum(A(i,:))+sum(I1(i,:)) -b(i,:)];

end

I2=eye(n);

fori=1:size(t)

l(i,:)= [0*ones(1,n) 0*ones(1,m) t(i,:) I2(i,:) c(:,i)-sum(t(i,:))-
sum(I2(i,:)) -c(:,i)];

end

g=[c 0*ones(1,m) -b' 0*ones(1,n) -sum(c)+sum(b)' 0];

% canonical form of given LPP

ca=[k;l;g];

[s,q]=size(ca);

nc=[0*ones(1,q-2) 1 0];

% Algorithm

erro=norm(nc);

I=eye(q);

r=1/sqrt(q*(q-1));

alfa=1;

ns=0;

tol=1e-6;

mexit=100;

e=ones(1,q);

x0=e'/q;

D=diag(x0);

whileerro>tol& ns <mexit,

ns=ns+1;

B=[ca*D;e];

P=pinv(B*B')*B;

cp=(I-B*P)*D*nc';

dp=norm(cp);

uc=cp/dp;

y=x0-(alfa*r*uc);

x=D*y./(e*D*y);

D=diag(x);

Brownies=0 portions,
Cheese Cake=0 portions,
Ice Cream=0 portions.
And the minimum cost of the daily diet is
TK 410.8599.

VI. Conclusions

In this paper, we present Karmarkar⁴ interior point algorithm and developed its computer oriented code in MATLAB for solving LP problem. We also formulate a real life sizable large scale diet problem, which involves a numerous amount of data, constraints and variables. In Karmarkar⁴ interior point algorithm, first we have to transform a LP problem into canonical form. So when we convert the LP problem into canonical form the number of variables and constraints increases. Then it becomes difficult to solve this problem by hand calculation. But by using our computer code we can easily get our desired solution in a short time. In this paper, we solved the large scale diet problem by our computer code which required a very few seconds. So we conclude that the interior point algorithm and required computer code in MATLAB developed by us is a powerful method for solving all types of LP problems. Also in future, we can extend this algorithm for linear fractional programming problems.

References

1. Yingn, Ye., 1996. Interior Point Algorithm: Theory and Analysis.
2. Dantzig, G. B., 1963. Linear programming and Extension, Princeton University Press, Princeton, NJ.
3. Dantzig, G. B. and Mukund, N. Thapa, 2003. Linear programming, 2: Theory and Extensions, Springer series in Operations Research.
4. Karmarkar, Narendra, 1984. A new polynomial time algorithm for linear programming. *Combinatorica*, **4**, 373-395.
5. Veblen, O. and J. W. Young, 1938. Projective Geometry. Blaisdell, New York.
6. Tantawy, S. F., 2007. A New Method to Find All Alternative Extreme Optimal Points for Linear Programming Problem. *Australian Journal of Basic and Applied Sciences*, **1**, 38-44.
7. Vanderbei, R. J., M. S. Meketon, and B. A. Freedman, 1986. A Modification of karmarkar's linear Programming Algorithm. *Algorithmica*, **1**, 395-407.
8. Adler, Resende, I. M. G. C., G. Veiga, and N. Karmarkar, 1989. An implementation of Karmarkar's algorithm for linear programming. *Math program*, **44**, 297-335.

