American Journal of Applied Mathematics
2014; 2(6): 221-226
Published online January 05, 2015 (http://www.sciencepublishinggroup.com/j/ajam)
doi: 10.11648/j.ajam.20140206.15
ISSN: 2330-0043 (Print); ISSN: 2330-006X (Online)

Numerical Experiments with the Lagrange Multiplier and Conjugate Gradient Methods (ILMCGM)

Samson Adebayo Olorunsola, Temitayo Emmanuel Olaosebikan, Kayode James Adebayo

Department of Mathematical Sciences, Ekiti State University, Ado Ekiti, Nigeria

Email address:
funmiladebayo@yahoo.com (S. A. Olorunsola), eolaosebikan@hotmail.com (T. E. Olaosebikan), akj7070@yahoo.com (K. J. Adebayo)

To cite this article:
Samson Adebayo Olorunsola, Temitayo Emmanuel Olaosebikan, Kayode James Adebayo. Numerical Experiments with the Lagrange Multiplier and Conjugate Gradient Methods (ILMCGM). *American Journal of Applied Mathematics.* Vol. 2, No. 6, 2014, pp. 221-226. doi: 10.11648/j.ajam.20140206.15

Abstract: In this paper, we imbed Langrage Multiplier Method (LMM) in Conjugate Gradient Method (CGM), which enables Conjugate Gradient Method (CGM) to be employed for solving constrained optimization problems of either equality, inequality constraint or both. In the past, Langrage Multiplier Method has been used extensively to solve constrained optimization problems. However, with some special features in CGM which makes it unique in solving unconstrained optimization problems, we see that this features we be advantageous to solve constrained optimization problems if we can add or subtract one or two things into the CGM. This, then call for the Numerical Experiments with the Lagrange Multiplier Conjugate Gradient Method (ILMCGM) that is aimed at taking care of any constrained optimization problems, either with equality or inequality constraint The authors of this paper desire that, with the construction of the Algorithm, one will circumvent the difficulties undergone using only LMM to solve constrained optimization problems and its application will further improve the result of the Conjugate Gradient Method in solving this class of optimization problem. We applied the new algorithm to some constrained optimization problems of two, three and four variables in which some of the problems are pertain to quadratic functions. Some of these functions are subject to linear, nonlinear, equality and inequality constraints.

Keywords: Lagrange Multiplier Method, Constrained Optimization Problem, Conjugate Gradient Method, Numerical Experiments of the Lagrange Multiplier Conjugate Gradient Method

1. Introduction

The general optimization problem to be considered is of the form described by [1] and [2] as:

Optimize:

$$f(x) \tag{1.1}$$

Subject to:

$$h_i(x) = 0 i = 1, 2, \ldots, m_1 \tag{1.2}$$

$$g_j(x) \geq 0 \, j = 1, 2, \ldots, m_2 \tag{1.3}$$

where $x \in R^n$, $h_i(x)$, an equality vector equations of dimension m_1 , and $g_j(x)$ is an inequality vector of dimension m_2 , such that the sum of the constraints $\mathrm{m} = (m_1 + m_2)$. The functions $f(x), h_i(x) and g_j(x)$ are differentiable functions. Methods for solving this model have been developed, tested and successfully applied to many important problems of scientific and economic interest. However, in spite of the proliferation of the methods, there is no universal method for solving all optimization problems which calls for application of ILMCGA to solve constrained optimization problems.

1.1. Conjugate Gradient Method

In 1952, Hestenes and Stiefel developed a Conjugate Gradient Method (CGM) algorithm for solving algebraic equations which was successfully applied to nonlinear equations with results reported by Fletcher and Reeves in 1964.

The CGM algorithm for iteratively locating the minimum x^* of $f(x)$ in $\mathcal{H}$ is described as follows:

Step 1: Guess the first element $x_0 \in \mathcal{H}$ and compute the remaining members of the sequence with the aid of the formulae in the steps 2 through 6.

Step 2: Compute the descent direction

$$p_0 = -g_0 \quad (1.4)$$

Step 3: Set

$$x_{i+1=}x_i + \alpha_i p_i \text{ ; where } \alpha_i = \frac{\langle g_i, g_i\rangle_{\mathcal{H}}}{\langle p_i, Gp_i\rangle_{\mathcal{H}}} \quad (1.5)$$

Step 4: Compute

$$g_{i+1} = g_i + \alpha_i G p_i \quad (1.6)$$

Step 5: Set

$$p_{i+1} = -g_{i+1} + \beta_i p_i \text{ ; } \beta_i = \frac{\langle g_{i+1}, g_{i+1}\rangle_{\mathcal{H}}}{\langle g_i, g_i\rangle_{\mathcal{H}}} \quad (1.7)$$

Step 6: If $g_i = 0$ for some i, then, terminate the sequence; else set $i = i+1$ and go to step 3.

In the iterative steps 2 through 6 above, p_i denotes the descent direction at ith step of the algorithm, α_i, is the step length of the descent sequence $\{x_i\}$ and g_i denotes the gradient of f at x_i. Steps 3, 4 and 5 of the algorithm reveal the crucial role of the linear operator G in determining the step length of the descent sequence and also in generating a conjugate direction of search.

Doctoral Thesis of [3] threw light on the theoretical applicability of the CGM, which was extended to optimal control problems by [4], [5] and [6].Applicability of the CGM algorithm thus depends solely on the explicit knowledge of the linear operator, G. Generally, for optimization problems, G is readily determined and such enjoys the beauty of the CGM as a computational scheme since the CGM exhibits quadratic convergence and requires only a little more computation per iteration.

1.2. Lagrange Multipliers Method

In mathematical optimization, the method of Lagrange multipliers (named after Joseph Louis Lagrange) provides a strategy for finding the maximum/minimum of a function subject to constraints. The Lagrange multiplier method was basically introduced to solve optimization problems with equality constraints of the form (1.1) and (1.2). In solving this, a new variable, λ, called the Lagrange multiplier introduced to append the constraint (1.2) into the objective function (1.1) to give a new function:

$$L(X,\lambda) = f(X) + \sum_{j=1}^{m} \lambda_j h_j(X) \quad (1.8)$$

the equivalent matrix form of (1.4) is of the form:

$$L(X,\lambda) = f(X) + \lambda^T h(X) \quad (1.9)$$

(1.8) and (1.9) are referred to as Lagrangian functions where λ is an $m \times 1$ vector of Lagrange multipliers.

In finding the minimum of the function *f(x),* generally we can set the partial derivatives of (1.8) or (1.9) to zero such as:

$$\frac{\partial L}{\partial x_i}\left(X^*, \lambda^*\right) = 0, i = 1,2,\ldots,n \quad (1.10)$$

$$\frac{\partial L}{\partial \lambda_j}\left(X^*, \lambda^*\right) = 0,\ j = 1,2,\ldots,m \quad (1.11)$$

where X^*, λ^* in (1.10) and (1.11 are the minimum solution and the set of associated Lagrange multipliers of (1.8) or (1.9). Also, (1.10) and (1.11) are referred to as Kuhn-Tucker necessary conditions for a local minimum of (1.8) or (1.9) while the second derivatives of the function f(x) given as:

$$\frac{\partial^2 L}{\partial^2 x_i}\left(X^*, \lambda^*\right) = 0, i = 1,2,\ldots,n \quad (1.12)$$

$$\frac{\partial^2 L}{\partial^2 \lambda_j}\left(X^*, \lambda^*\right) = 0,\ j = 1,2,\ldots,m \quad (1.13)$$

are referred to as sufficient conditions for a local minimum of (1.8) or (1.9).

1.3. Lagrange Multipliers Method Algorithm

In order to maximize or minimize the function (1.1) subject to the constraint (1.2), the following procedures are taken:

Step 1: first create the Lagrange Function.

$$L(X,\lambda) = f(X) + \sum_{j=1}^{m} \lambda_j h_j(X) \quad (1.14)$$

Step 2: Compute the partial derivatives with respect to X and the Lagrange multiplier λ of the function (1.14)

Step 3: Set each of the partial derivatives of (1.14) equal to zero to get:

$$\frac{\partial L}{\partial X} = 0 \quad (1.15)$$

$$\frac{\partial L}{\partial \lambda} = 0 \quad (1.16)$$

using (1.14) proceed to solve for X in term of λ. Now substitute the solutions for X so that (1.15) is in terms of λ only. Then, solve for λ and use this value to find the optimal values X .

2. Imbedded Lagrange Multiplier Conjugate Gradient Method (ILMCGM) Algorithm

Haven investigated the two methods; we now draw out the following steps which will be used to solve some constrained optimization problems. The steps are as follows:

Step 1: Equate the constraint to zero (in case of equation is of the form: $AX = b$)

Step 2: Append the new equation in step1 (i.e.$AX - b = 0$) into the performance index using Lagrange Multiplier λ to form Lagrangian or Augmented Lagrangian function

$$\text{[i.e. } L(x,\lambda) = f(x) + \lambda(AX - b) \text{]}$$

Step 3: Guess the initial elements $x_0, \lambda > 0$

Step 4: Compute the initial gradient, g_0, as well as the initial descent direction, $p_0 = -g_0$

Step 5: Compute the Hessian Matrix, H, in step 2

Step 6: Set

$$x_{i+1} = x_i + \alpha_i p_i, where \alpha_i = \frac{g^T{}_i g_i}{p^T{}_i H p_i}, i = 1, 2, \ldots, n$$

Step 7: Update the gradient using:

$$g_{i+1} = g_i + \alpha_i H p_i, i = 1, 2, \ldots, n$$

Step 8: Update the descent direction using:

$$p_{i+1} = -g_i + \beta_i p_i, where \beta_i = \frac{g^T{}_{i+1} g_{i+1}}{g^T{}_i g_i}, i = 1, 2, \ldots, n$$

Step 9: If $g_i = 0$ stop, else, set $i = i + 1$ and return to step 6.

NOTE: $f(x)$ and $L(x,\lambda)$ are the performance index and Lagrangian function respectively which are differentiable.

3. Computational Procedure of the ILMCGA Algorithm

Considering (1.1) and (1.2), there exists a Lagrange Multiplier λ which imbed (1.2) into (1.1) to give a Lagrangian function such as:

$$L(X,\lambda) = f(X) + \sum_{i=1}^{n} \lambda_i h_i(X) \tag{3.1}$$

Let the initial guess be:

$$x_0 = \begin{pmatrix} x_{1(0)} \\ x_{2(0)} \\ . \\ . \\ . \\ x_{n(0)} \end{pmatrix} \tag{3.2}$$

$$\lambda_0 = \begin{pmatrix} \lambda_{1(0)} \\ \lambda_{2(0)} \\ . \\ . \\ . \\ \lambda_{n(0)} \end{pmatrix} \tag{3.3}$$

Putting (3.2) and (3.3) in (1.1) and (3.1) respectively gives the initial functions values i.e. $f(x_0)$ and $L(x_0, \lambda_0)$.

Computing the gradient of (3.1) with respect to $(x_1, x_2, \ldots, x_n)^T$ we have:

$$\begin{pmatrix} \frac{\partial}{\partial x_1} L(X,\lambda) = \frac{\partial}{\partial x_1} f(X) + \lambda_i \frac{\partial}{\partial x_1} \sum_{i=1}^{n} h_i(X) \\ \frac{\partial}{\partial x_2} L(X,\lambda) = \frac{\partial}{\partial x_2} f(X) + \lambda_i \frac{\partial}{\partial x_2} \sum_{i=1}^{n} h_i(X) \\ . \\ . \\ . \\ \frac{\partial}{\partial x_n} L(X,\lambda) = \frac{\partial}{\partial x_n} f(X) + \lambda_i \frac{\partial}{\partial x_n} \sum_{i=1}^{n} h_i(X) \end{pmatrix} \tag{3.4}$$

Putting (3.2) and (3.3) for X and λ respectively in (3.4) gives us the initial gradient as:

$$g_0 = \begin{pmatrix} \frac{\partial}{\partial x_1} L(x_0, \lambda_0) \\ \frac{\partial}{\partial x_2} L(x_0, \lambda_0) \\ . \\ . \\ . \\ \frac{\partial}{\partial x_n} L(x_0, \lambda_0) \end{pmatrix} \tag{3.5}$$

Multiplying (3.5) by negative gives the decent direction as:

$$p_0 = -g_0 = \begin{pmatrix} -\frac{\partial}{\partial x_1} L(x_0, \lambda_0) \\ -\frac{\partial}{\partial x_2} L(x_0, \lambda_0) \\ . \\ . \\ . \\ -\frac{\partial}{\partial x_n} L(x_0, \lambda_0) \end{pmatrix} \tag{3.6}$$

Computing the Hessian Matrix of (3.1) using (3.4) gives:

$$H = \begin{pmatrix} \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2{}_1} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}} \cdots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}} \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2{}_2} \cdots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}} \\ . & . & . \\ . & . & . \\ . & . & . \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}} \cdots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}} \end{pmatrix} \tag{3.7}$$

On transposing (3.5) and (3.6) respectively, we have:

$$g_0^T = \left(\frac{\partial}{\partial x_1}L(x_0,\lambda_0)\frac{\partial}{\partial x_2}L(x_0,\lambda_0)\dots\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right)^T \quad (3.8)$$

and

$$p_0^T = \left(-\frac{\partial}{\partial x_1}L(x_0,\lambda_0)-\frac{\partial}{\partial x_2}L(x_0,\lambda_0)\dots-\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right)^T \quad (3.9)$$

Multiplying (3.5) and (3.8) gives us a scalar, k .i.e.

$$k = g_0^T g_0$$

$$\text{k} = \left(\frac{\partial}{\partial x_1}L(x_0,\lambda_0) \quad \frac{\partial}{\partial x_2}L(x_0,\lambda_0) \quad \dots \quad \frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right)^T \begin{pmatrix} \frac{\partial}{\partial x_1}L(x_0,\lambda_0) \\ \frac{\partial}{\partial x_2}L(x_0,\lambda_0) \\ . \\ . \\ . \\ \frac{\partial}{\partial x_n}L(x_0,\lambda_0) \end{pmatrix}$$

$$\text{k} = \left(\left(\frac{\partial}{\partial x_1}L(x_0,\lambda_0)\right)^2 + \left(\frac{\partial}{\partial x_2}L(x_0,\lambda_0)\right)^2 + \dots + \left(\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right)^2\right) \quad (3.10)$$

Similarly, multiplying (3.9), (3.7) and (3.6) gives a scalar, z .i.e.

$$z = p_0^T H p_0$$

$$z = \left(-\frac{\partial}{\partial x_1}L(x_0,\lambda_0) \quad -\frac{\partial}{\partial x_2}L(x_0,\lambda_0) \quad \dots \quad -\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right)^T \begin{pmatrix} \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_1} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}} \dots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}} \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_2} \dots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}} \\ . & . & . \\ . & . & . \\ . & . & . \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}} \dots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}} \end{pmatrix} \begin{pmatrix} -\frac{\partial}{\partial x_1}L(x_0,\lambda_0) \\ -\frac{\partial}{\partial x_2}L(x_0,\lambda_0) \\ . \\ . \\ . \\ -\frac{\partial}{\partial x_n}L(x_0,\lambda_0) \end{pmatrix} \quad (3.11)$$

$$Hp_0 = \begin{pmatrix} \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_1} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}} \dots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}} \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_2} \dots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}} \\ . & . & . \\ . & . & . \\ . & . & . \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}} & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}} \dots & \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}} \end{pmatrix} \begin{pmatrix} -\frac{\partial}{\partial x_1}L(x_0,\lambda_0) \\ -\frac{\partial}{\partial x_1}L(x_0,\lambda_0) \\ . \\ . \\ . \\ -\frac{\partial}{\partial x_n}L(x_0,\lambda_0) \end{pmatrix}$$

$$= \begin{pmatrix} \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_1}\left(-\frac{\partial}{\partial x_1}L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}}\left(-\frac{\partial}{\partial x_2}L(x_0,\lambda_0)\right) + \dots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}}\left(-\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right) \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}}\left(-\frac{\partial}{\partial x_1}L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_2}\left(-\frac{\partial}{\partial x_2}L(x_0,\lambda_0)\right) + \dots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}}\left(-\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right) \\ . \qquad . \qquad . \\ . \qquad . \qquad . \\ . \qquad . \qquad . \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}}\left(-\frac{\partial}{\partial x_1}L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}}\left(-\frac{\partial}{\partial x_2}L(x_0,\lambda_0)\right) + \dots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}}\left(-\frac{\partial}{\partial x_n}L(x_0,\lambda_0)\right) \end{pmatrix} \quad (3.12)$$

putting (3.12) into (3.11), we have:

$$z = \left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0) \quad -\frac{\partial}{\partial x_2} L(x_0,\lambda_0) \quad \ldots \quad -\frac{\partial}{\partial x_n} L(x_0,\lambda_0) \right)^T \begin{pmatrix} \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_1}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right) \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_2}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right) \\ \vdots \\ \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right) \end{pmatrix} \quad (3.13)$$

With matrix multiplication, (3.13) becomes:

$$z = \begin{pmatrix} -\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\left(\frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_1}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)\right) \\ -\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\left(\frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_2}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)\right) \\ \vdots \\ -\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\left(\frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)\right) \end{pmatrix} \quad (3.14)$$

dividing (3.10) and (3.14) .i.e.:

$$\alpha_0 = \frac{\left(\left(\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right)^2 + \left(\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right)^2 + \ldots + \left(\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)^2\right)}{\begin{pmatrix} -\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\left(\frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_1}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{12}}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{1n}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)\right) \\ -\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\left(\frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{21}}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x^2_2}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{2n}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)\right) \\ \vdots \\ -\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\left(\frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m1}}\left(-\frac{\partial}{\partial x_1} L(x_0,\lambda_0)\right) + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{m2}}\left(-\frac{\partial}{\partial x_2} L(x_0,\lambda_0)\right) + \ldots + \frac{\partial^2 L(x_0,\lambda_0)}{\partial x_{mn}}\left(-\frac{\partial}{\partial x_n} L(x_0,\lambda_0)\right)\right) \end{pmatrix}} \quad (3.15)$$

(3.15) is the step length. Now set $x_{i+1} = x_i + \alpha_i p_i, i = 0,1,2,\ldots,n$

4. Computational Results

The following problems were evaluated using the ILMCGM algorithm thus:

Problem 1:

$$Minimize f(x) = \frac{1}{2}x^2_1 + x^2_2 + 2x^2_3 + 4x^2_4$$

Subject to: $x_1 + x_2 + x_3 + x_4 = 1$

Problem 2:

$$Minimize\ f(x) = x^2_1 + x^2_2$$

$$Subject\ to: (x_1 - 2)^2 + (x_2 - 3)^2 \le 4$$

$$x^2_1 = 4x_2$$

Problem 3:

$$Minimize f(x) = 3x^2_1 - \frac{1}{2}x^2_2 - \frac{1}{2}x^2_3 + x_1x_2 - x_1x_3 + 2x_2x_3$$

Subject to: $2x_1 - x_2 + x_3 = 2$

Problem 4:

$$Minimize f(x) = x^2_1 + 2x^2_2 + 2x_1x_2 + 2x_1 + 3x_2$$

Subject to: $x^2_1 + x_2 = 1$

Table 1. *Table of result for problem 1, at* $\lambda = 1$.

No. of Iterations	x_1	x_2	x_3	x_4	Function values	Gradient Norms
0	2	4	6	0	101	26.75817632
1	1.197309416	1.5919228248	-0.6890882	-0.267563528	6.319642463	5.168980366
2	0.120061882	-0.516334808	-0.304956703	0.247065426	-0.750192335	3.330241158
3	-0.214370162	-0.812905518	-0.091077714	-0.158164124	-1.476070995	1.04720774
4	-0.818231751	-0.519110945	0.083430343	-0.14742187	-1.696252376	0.608823127

Table 2. *Table of result for problem 2, at* $\lambda_1 = 0, \lambda_2 = 1$ *the slack variable* $\theta = 1$.

No. of iterations	x_1	x_2	Function values	Gradient Norms
0	1	2	-5	2.828427125
1	0.6	2.4	-3.5	0.565685424
2	1.1283333333	2.500000000	5.0329388888	0.0000000001

Table3. *Table of result for problem 3, at* $\lambda = 1$.

No. of Iterations	x_1	x_2	x_3	Function values	Gradient Norms
0	2	4	6	34	15.03329638
1	-0.685148512	1.986138616	5.776237624	8.714851489	9.827776829
2	-1.2084530093	10.26682388	5.99125423	42.94077035	16.7817968
3	-0.224824648	-0.40097582	0.409314579	-1.79795779	0.129166442

Table 4. *Table of result for problem 4, at* $\lambda = 0$.

No. of iterations	x_1	x_2	Function values	Gradient Norms
0	2	3	47	15.000000000
1	-0.291981844	-0.628971253	-1.227118003	0.186985068
2	-0.500000858	-0.499999476	-1.250000000	0.000042076

5. Conclusion

Computationally, the resulting algorithm from the Lagrange Multiplier Method imbedded in Conjugate Gradient Method was tested on some constrained optimization problems of two, three and four variables. The problems are pertained to quadratic functions. Some of these functions are subject to linear and nonlinear constraints with varying Lagrange parameter, λ, between 0 and 1. While the slack variable parameter, θ, is 1.

Suppose we take the function value as the terminating criterion, Problem 2 and 3 with the numerical results 5.03293888888 and -1.79795779 when compare with the analytical results which are: 5 and -1.21093750446 respectively, it invariably establishes the relevance of the new algorithm for solving constrained optimization problems. Problem1 and 4 decreases monotonically establishing the convergence of the constrained Optimization Problems. On using the Gradient Norm as the stopping criterion, the Gradient Norm of Problems 1, 2, 3 and 4 tends to zero which show the convergence of the problems. On using the Gradient Norm as the stopping criterion, the Gradient Norm of Problems 1, 2, 3 and 4 tends to zero which show the convergence of the problems. All these points to the fact that, the constructed ILMCGM algorithm efficiently solve the problems as supposed.

References

[1] RAO, S. S., (1978), Optimization Theory and Applications, Willy and Sons.

[2] THOMAS, F.E., and DAVID, M.H., (2001), Optimization of Chemical Processes, McGraw Hill

[3] IGOR, G., STEPHEN, G. N. and ARIELA, S., (2009), Linear and Nonlinear Optimization, George Mason University, Fairfax, Virginia, SIAM, Philadelphia.

[4] David, G. Hull, (2003), Optimal Control Theory for Applications, Mechanical Engineering Series, Springer-Verlag, New York, Inc., 175 Fifth Avenue, New York, NY 10010.

[5] Bersekas, D. P, (1982), Constrained Optimization and Lagrange Multipliers Method, Academic Press, Inc.

[6] Rockafellar, R. T., (2005), Multiplier Method of Hestenes and Powell applied to convex Programming, Journal of Optimization Theory and Applications, Vol. 4, No. 4.

[7] Triphath S. S and Narenda K. S, (1972), Constrained Optimization Problems Using Multiplier Methods, Journal of Optimization Theory and Applications: Vol. 9, No. 1.