Overcoming Computational Complexity in Nonlinear Optimization

Sadik Olaniyi Maliki
Ebonyi State University Abakaliki.
somaliki@yahoo.com

Adeola Olutayo Maliki
Ebonyi State University Abakaliki.
aomaliki@yahoo.com

Abstract
In this work our main objective is to show how to overcome computational complexity when dealing with nonlinear optimization problems. We consider in particular a nonlinear objective function involving five variables to be maximized subject to four equality constraints. The methodological procedure involves the application of Lagrange multipliers. The first order optimality conditions provide us with the critical values of the problem, while the second order condition is given by the so called Bordered Hessian Matrix. The results of our investigation involved solving highly nonlinear systems of equations involving the Lagrange multipliers, evaluating determinants of very large matrices and finally computing the roots of polynomials of order five. The above difficulties were easily overcome with the help of MathCAD software which proved very efficient in generating numerical solutions to the system of nonlinear equations, evaluating matrix determinants with its symbolic capabilities and computing roots of polynomials with ease. It is shown that for the given problem, only one critical point exists corresponding to a maximum.

Keywords: Lagrange multipliers, optimality conditions, Mathcad.

1 Introduction

Optimality criteria form the foundations of mathematical programming both theoretically and computationally. In general, these criteria can be classified as either necessary or sufficient. Naturally, one would like to have the same criterion be both necessary and sufficient; however, this occurs only under somewhat ideal conditions, which are rarely satisfied in practice. In the absence of convexity, one is never assured, in general, of the sufficiency of any such optimality criterion. We are then left with only the necessary optimality criterion to face the vast number of mathematical problems which are not convex.

In this work we develop necessary and sufficient conditions for determining constrained extrema. Such constraints may be in the form of equality or inequality. For the moment we will be interested in equality constraints, and for such problems the Lagrangian method is most suitable.

1.1 Problems involving the Lagrange Multipliers

The problem considered in this section is the following.

Problem A \[ \min f(x) \]
\[ \text{such that } g(x) = 0 \]
where \( x \in \mathbb{R}^n \), \( g(x) = (g_1(x), g_2(x), \ldots, g_m(x))^T \).

The functions \( f(x) \) and \( g_i(x), \ i = 1, 2, \ldots, m \) are assumed to be twice continuously differentiable. Many methods for solving the above problems have been published (Arrow & Solow, 1958; Bazaraa & Goode, 1972; Fletcher & Lill, 1971; Miele, el al, 1972). Generally, they are based on one of the two main procedures that follow.

1) The first is the Lagrange multiplier technique, where the constraints are adjoined to the function by means of multipliers \( \lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m) \) to form a new function:
Overcoming Computational Complexity in Nonlinear Optimization

\[ L(x, \lambda) = f(x) - \lambda \cdot g(x) \]

\[ = f(x) - \sum_{i=1}^{m} \lambda_i g_i(x) \]

Generally called the Lagrangian of the problem. The problem is then reduced to finding the saddle point of \( L(x, \lambda) \) in the \((x, \lambda)\) space, and thus the dimension of the problem is increased from \( n \) to \( n + m \).

2) The other basic approach is the penalty function method (Mc Kormick, 1967; Powell, 1972). In this case a function including the constraints in a proper manner is adjoined to the original function \( f(x) \), for example

\[ f(x) + c g^T(x) g(x) \]

where \( c \) is a positive real-valued parameter. Under very mild conditions, the solution of

\[ \min_x \{ f(x) + c g^T(x) g(x) \} \]

tends to the solution of

\[ \min_x \{ f(x) \} \]

such that \( g(x) = 0 \) as \( c \to \infty \).

It turns out that the penalty function method is not very attractive from a numerical point of view, since the functions created become very badly conditioned for numerical optimization.

1.2 Notation

For the purpose of notation, as usual the symbol \( \nabla \) is the differentiation operator with respect to the vector \( x = (x_1, x_2, \ldots, x_n) \) i.e.,

\[ \nabla f(x) = \left( \frac{\partial f(x)}{\partial x_1}, \ldots, \frac{\partial f(x)}{\partial x_n} \right)^T \]

The symbol \( \nabla^2 \) represents the operator applied twice, thus \( \nabla^2 f(x) \) is the \( n \times n \) matrix whose \( i, j \)th element is \( \frac{\partial^2 f(x)}{\partial x_i \partial x_j} \). Subsequently, \( f' \) will denote \( f(x') \), \( \nabla f' \) will denote \( \nabla f(x') \); for a parametrized function, \( \rho' \) will represent the derivative with respect to the parametrizing variable, i.e.

\[ f'(\omega) = \frac{df(x(\omega))}{d\omega} \]

By the term “differentiable” we mean continuously differentiable. The rank of a matrix \( A \) will be denoted by \( \rho(A) \).

1.3 Optimality Condition For Equality constraint problems

In this section, with reference to problem (A), we are interested in the conditions in which \( f \) and the constraint sets will be satisfied at an optimum.

The following theorem generalizes the concept of Lagrange multipliers.

1.4 The Theorem of Lagrange

The Theorem of Lagrange provides a powerful characterization of optima of equality-constrained optimization problems in terms of the behavior of the objective function and the constraint functions at these points.
1.5 Theorem (Lagrange)

Let \( f : \mathbb{R}^n \to \mathbb{R} \) and \( g_i : \mathbb{R}^n \to \mathbb{R} \) be \( C^1 \) functions, \( i = 1, \ldots, m \). Suppose \( x^* \) is a local maximum or minimum of \( f \) on the set \( K = \{ x \mid g_i(x) = 0, i = 1, \ldots, m \} \), where \( P \subset \mathbb{R}^n \) is open. Let \( \rho(\nabla g(x^*)) = m \), then there exists a vector \( \lambda^* = (\lambda_1^*, \lambda_2^*, \ldots, \lambda_m^*) \in \mathbb{R}^m \) such that

\[
\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(x^*) = 0
\]

where \( \lambda_i^* \)'s are called the Lagrangian multipliers associated with the local optimum \( x^* \). \( \rho(\nabla g(x^*)) \) is the rank of the gradient vector \( \nabla g_i(x) \) that is, the rank of

\[
\nabla g(x^*) = \frac{\partial g_i(x^*)}{\partial x_j}, \quad j = 1, \ldots, n, \quad i = 1, \ldots, m.
\]

We defined the Lagrangian function \( L : K \times \mathbb{R}^m \to \mathbb{R}, \forall x \in \mathbb{R}^n \) as follows:

\[
L(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i g_i(x), \quad \lambda \in \mathbb{R}^m
\]

The function \( L \) is called the Lagrangian associated with the problem (A). With this function, we get the first order necessary equations as follows:

\[
\frac{\partial L(x, \lambda)}{\partial x_j} = 0, \quad j = 1, \ldots, n, \quad \frac{\partial L(x, \lambda)}{\partial \lambda_i} = 0, \quad i = 1, \ldots, m.
\]

1.6 Definition.

Let \( x^* \in K \). We say that \( x^* \) is a regular point of \( K \) if the vectors \( \nabla g_i(x^*), \ldots, \nabla g_m(x^*) \) of \( \mathbb{R}^m \) are linear independent.

In matrix form, \( x^* \) is a regular point of \( K \) if the Jacobian matrix of \( g \) at \( x^* \), denoted by

\[
J(g, x^*) = \begin{bmatrix} \frac{\partial g_1(x^*)}{\partial x_1} \\ \vdots \\ \frac{\partial g_m(x^*)}{\partial x_n} \end{bmatrix}, \quad j = 1, \ldots, n, \quad i = 1, \ldots, m
\]

has rank \( m \), where \( g = (g_1, g_2, \ldots, g_m) \).

Remark:

1) If a pair \( (x^*, \lambda^*) \) satisfies the twin conditions that \( g(x^*) = 0 \) and \( \nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(x^*) = 0 \), we will say that \( (x^*, \lambda^*) \) meets the first order conditions of the Theorem of Lagrange, or \( (x^*, \lambda^*) \) meets the first order necessary conditions in equality constrained optimization problems.

2) The Theorem of Lagrange only provides necessary conditions for local optima at \( x^* \), and that, only for those local optima \( x^* \) which also meet the condition that \( \rho(\nabla g(x^*)) = m \). These conditions are not asserted to be sufficient, meaning that, the theorem does not imply that if there exist such that \( g(x^*) = 0 \), and \( \nabla f(x^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(x^*) = 0 \), then \( x^* \) must either be a local maximum or a local minimum, even if \( x \) also meets the rank condition \( \rho(\nabla g(x^*)) = m \).
1.7 Sufficiency conditions for the Lagrangian

The sufficiency conditions for the Lagrangian method is stated as follows;

Define:

\[
H^B = \begin{bmatrix}
0 & P \\
P^T & Q_{(m+n)(m+n)}
\end{bmatrix}
\]

where:

\[
P = \begin{bmatrix}
\nabla g_1(x) \\
\vdots \\
\nabla g_m(x)
\end{bmatrix}_{m \times n}
\] and

\[
Q = \begin{bmatrix}
\frac{\partial^2 L(x, \lambda)}{\partial x_i \partial x_j}
\end{bmatrix}_{n \times n}
\]

The matrix \( H^B \) is called the Bordered Hessian matrix (BH-matrix) (Taha, 1972). Having computed the stationary point \((x^0, \lambda^0)\) for the Lagrangian function \(L(x, \lambda)\) and the BH-matrix \( H^B \) evaluated at \((x^0, \lambda^0)\), then \(x^0\) is;

(i) A maximum point if, starting with the principal major determinant of order \((2m+1)\), the last \((n-m)\) principal minor determinants of \( H^B \) form an alternating sign pattern with \((-1)^{m+1}\).

(ii) A minimum point if, starting with the principal minor determinant of order \((2m+1)\), the last \((n-m)\) principal minor determinants of \( H^B \) have the sign of \((-1)^m\).

Remark:

The above conditions are sufficient for identifying an extreme point, thus a stationary point may be an extreme point without satisfying these conditions.

Other conditions exist that are both necessary and sufficient for identifying extreme points. The major disadvantage is that the procedure may prove computationally complex. However with a spreadsheet such as MathCAD (Mathcad User’s guide), this difficulty is easily overcome. For this purpose we define a variant of the BH-matrix as;

\[
\Delta = \begin{bmatrix}
0 & P \\
P^T & Q - \mu I_{(m+n)(m+n)}
\end{bmatrix}
\]

Evaluated at the stationary point \((x^0, \lambda^0)\), where \(\mu\) is an unknown parameter. We now consider the determinant \(|\Delta|\); then each of the real \((m+n)\) roots \(\mu\) of the polynomial \(|\Delta| = 0\) must be

(i) Negative if \(x^0\) is a maximum point

(ii) Positive if \(x^0\) is a minimum point

2 An illustrative example

To illustrate the foregoing procedure we consider the following nonlinear optimization problem.
Max \( f(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 \)

Subject to

\[
g_1(x) = x_1^2 + x_2 + x_3 + 2x_4 + x_5 - 5 = 0
\]

\[
g_2(x) = x_1 + x_2^2 + x_3 + x_4 + 3x_5 - 10 = 0
\]

\[
g_3(x) = 2x_1 + x_2 + x_3^2 + 3x_4 + x_5 - 8 = 0
\]

\[
g_4(x) = 5x_1 + x_2 + x_3 + 3x_4 + x_5 - 2 = 0
\]

The Lagrangean function is given by

\[
L(x, \lambda) = \sum_{i=1}^{5} x_i^2 - \lambda_i (x_1^2 + x_2 + x_3 + 2x_4 + x_5 - 5) - \lambda_i (x_1 + x_2^2 + x_3 + x_4 + 3x_5 - 10)
\]

\[
- \lambda_i (2x_1 + x_2 + x_3^2 + 3x_4 + x_5 - 8) - \lambda_i (5x_1 + x_2 + x_3 + 3x_4 + x_5 - 2)
\]

(1)

his yields the following necessary conditions;

\[
\frac{\partial L}{\partial x_1} = 2x_1 - 2\lambda_1 x_1 - \lambda_2 - 2\lambda_3 - 5\lambda_4 = 0
\]

(2)

\[
\frac{\partial L}{\partial x_2} = 2x_2 - \lambda_1 - 2\lambda_2 x_2 - \lambda_3 - \lambda_4 = 0
\]

(3)

\[
\frac{\partial L}{\partial x_3} = 2x_3 - \lambda_1 - \lambda_2 - 2\lambda_3 x_3 - \lambda_4 = 0
\]

(4)

\[
\frac{\partial L}{\partial x_4} = 2x_4 - 2\lambda_1 - \lambda_2 - 3\lambda_3 - 3\lambda_4 = 0
\]

(5)

\[
\frac{\partial L}{\partial \lambda_1} = -\left(x_1^2 + x_2 + x_3 + 2x_4 + x_5 - 5\right) = 0
\]

(6)

\[
\frac{\partial L}{\partial \lambda_2} = -\left(x_1 + x_2^2 + x_3 + x_4 + 3x_5 - 10\right) = 0
\]

(7)

\[
\frac{\partial L}{\partial \lambda_3} = -\left(2x_1 + x_2 + x_3^2 + 3x_4 + x_5 - 8\right) = 0
\]

(8)

\[
\frac{\partial L}{\partial \lambda_4} = -\left(5x_1 + x_2 + x_3 + 3x_4 + x_5 - 2\right) = 0
\]

(9)

From equations (2) to (11) we get

\[
\frac{\lambda_1 + 2\lambda_2 + 5\lambda_4}{2(1 - \lambda_4)} = x_1 = \frac{\lambda_1 + 2\lambda_2 + 5\lambda_4}{2(1 - \lambda_4)}
\]

(10)

\[
\frac{\lambda_1 + \lambda_2 + \lambda_4}{2(1 - \lambda_2)} = x_2 = \frac{\lambda_1 + \lambda_2 + \lambda_4}{2(1 - \lambda_2)}
\]

(11)

\[
\frac{\lambda_1 + \lambda_2 + \lambda_4}{2(1 - \lambda_3)} = x_3 = \frac{\lambda_1 + \lambda_2 + \lambda_4}{2(1 - \lambda_3)}
\]

(12)

\[
\frac{2\lambda_1 + 2\lambda_2 + 3\lambda_3 + 3\lambda_4}{2} = x_4 = \frac{2\lambda_1 + 2\lambda_2 + 3\lambda_3 + 3\lambda_4}{2}
\]

(13)

\[
\frac{\lambda_1 + 3\lambda_2 + \lambda_3 + \lambda_4}{2} = x_5 = \frac{\lambda_1 + 3\lambda_2 + \lambda_3 + \lambda_4}{2}
\]

(14)

\[
x_1^2 + x_2 + x_3 + 2x_4 + x_5 = 5
\]

(15)

\[
x_1 + x_2^2 + x_3 + x_4 + 3x_5 = 10
\]

(16)

\[
x_1 + 2x_2 + x_3 + 2x_4 + x_5 = 5
\]

(17)

\[
x_1 + x_2^2 + x_3 + x_4 + 3x_5 = 10
\]

(18)
Overcoming Computational Complexity in Nonlinear Optimization

2x_1 + x_2 + x_1^2 + 3x_4 + x_5 = 8 \hfill (19)
5x_1 + x_2 + x_3 + 3x_4 + x_5 = 2 \hfill (20)

Substituting the values of x_1, x_2, x_3, x_4 as a function of the \lambda’s in equations (17) to (20), and simultaneously applying the Mathcad (Find) algorithm format [8] for computing roots of nonlinear equations we get;

\lambda_1 = 0 \lambda_2 := 0 \lambda_3 := 0 \lambda_4 := 0 \hfill \text{(starting value for the iterative procedure)}

**Given**

\[
\begin{align*}
\frac{\lambda_1 + 2\lambda_2 + 5\lambda_4}{2(1-\lambda_1)} + \frac{\lambda_1 + \lambda_2 + \lambda_4}{2(1-\lambda_1)} + 2 \left( \frac{\lambda_1 + \lambda_2 + 3\lambda_4}{2} \right) + \frac{\lambda_1 + 3\lambda_2 + \lambda_3 + \lambda_4}{2} &= 5 \\
\frac{\lambda_2 + 2\lambda_3 + 5\lambda_4}{2(1-\lambda_2)} + \frac{\lambda_2 + \lambda_3 + \lambda_4}{2(1-\lambda_2)} + 2 \left( \frac{\lambda_2 + \lambda_3 + 3\lambda_4}{2} \right) + \frac{\lambda_2 + 3\lambda_3 + \lambda_1 + \lambda_4}{2} &= 10 \\
\frac{\lambda_3 + 2\lambda_4 + 5\lambda_3}{2(1-\lambda_3)} + \frac{\lambda_3 + \lambda_3 + \lambda_3}{2(1-\lambda_3)} + 2 \left( \frac{\lambda_3 + \lambda_3 + 3\lambda_4}{2} \right) + \frac{\lambda_3 + 3\lambda_3 + \lambda_1 + \lambda_2}{2} &= 8 \\
\frac{\lambda_4 + 2\lambda_1 + 5\lambda_1}{2(1-\lambda_4)} + \frac{\lambda_4 + \lambda_1 + \lambda_1}{2(1-\lambda_4)} + 2 \left( \frac{\lambda_4 + \lambda_1 + 3\lambda_1}{2} \right) + \frac{\lambda_4 + 3\lambda_1 + \lambda_2 + \lambda_3}{2} &= 2 \\
\end{align*}
\]

Find (\lambda_1, \lambda_2, \lambda_3, \lambda_4) = \begin{bmatrix} -0.105 \\ 2.509 \\ 1.344 \\ -1.358 \end{bmatrix}

We have a have three other roots with their respective initial values. The results are summarized in the table below.

**Table 1.0**

<table>
<thead>
<tr>
<th>\lambda_01</th>
<th>\lambda_02</th>
<th>\lambda_03</th>
<th>\lambda_04</th>
<th>\lambda_05</th>
</tr>
</thead>
<tbody>
<tr>
<td>\lambda_1</td>
<td>0</td>
<td>-0.105</td>
<td>0.01</td>
<td>-14.906</td>
</tr>
<tr>
<td>\lambda_2</td>
<td>0</td>
<td>2.509</td>
<td>0</td>
<td>2.118</td>
</tr>
<tr>
<td>\lambda_3</td>
<td>0</td>
<td>1.344</td>
<td>0</td>
<td>4.252</td>
</tr>
<tr>
<td>\lambda_4</td>
<td>0</td>
<td>-1.358</td>
<td>0</td>
<td>4.109</td>
</tr>
</tbody>
</table>

The accuracy of the above solutions can be determined from the error values \(\varepsilon_i = |(\text{LHS} - \text{RHS})_i|\) for the given equations. These are tabulated below.

**Table 2.0**

<table>
<thead>
<tr>
<th>Eqn no</th>
<th>\lambda_1 \varepsilon_i \lambda_2 \varepsilon_i</th>
<th>\lambda_3 \varepsilon_i</th>
<th>\lambda_4 \varepsilon_i</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>3.452x10^{-4}</td>
<td>3.518x10^{-3}</td>
<td>8.973x10^{-3}</td>
</tr>
<tr>
<td>22</td>
<td>2.914x10^{-4}</td>
<td>1.200x10^{-2}</td>
<td>1.000x10^{-2}</td>
</tr>
<tr>
<td>23</td>
<td>1.238x10^{-3}</td>
<td>1.400x10^{-2}</td>
<td>1.200x10^{-2}</td>
</tr>
<tr>
<td>24</td>
<td>4.509x10^{-3}</td>
<td>6.989x10^{-5}</td>
<td>3.700x10^{-2}</td>
</tr>
</tbody>
</table>

The critical points can now be computed.

**Table 3.0**

<table>
<thead>
<tr>
<th>x = \frac{\lambda_0 + 2\lambda_1 + 5\lambda_4}{2(1-\lambda_1)}</th>
<th>x(\lambda_1)</th>
<th>x(\lambda_2)</th>
<th>x(\lambda_3)</th>
<th>x(\lambda_4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.721</td>
<td>0.98</td>
<td>-0.868</td>
<td>3.591</td>
<td></td>
</tr>
</tbody>
</table>
The determinant of the above matrix will now be evaluated at the critical points. At the point \( x=\left(\lambda_1^*\right) \),

\[
D_1(\mu) = \begin{bmatrix}
0 & 0 & 0 & -1.442 & 1 & 1 & 2 & 1 \\
0 & 0 & 0 & 1 & 0.078 & 1 & 1 & 3 \\
0 & 0 & 0 & 2 & 2 & -3.04 & 3 & 1 \\
0 & 0 & 0 & 5 & 1 & 1 & 3 & 1 \\
-1.442 & 1 & 2 & 5 & 2.21 - \mu & 0 & 0 & 0 \\
1 & 0.078 & 2 & 1 & 0 & -3.018 - \mu & 0 & 0 \\
1 & 1 & -3.04 & 1 & 0 & -0.888 - \mu & 0 & 0 \\
2 & 1 & 3 & 3 & 0 & 0 & 2 - \mu & 0 \\
1 & 2 & 1 & 3 & 0 & 0 & 0 & 2 - \mu \\
\end{bmatrix}
\]

\[
= -5.833 \times 10^{-16} \mu^5 + 7.083 \times 10^{-15} \mu^4 - 2.958 \times 10^{-14} \mu^3 + 4.942 \times 10^{-14} \mu^2 - 25159.953 \mu - 42272.936
\]

At the point \( x=\left(\lambda_2^*\right) \),

\[
D_2(\mu) = \begin{bmatrix}
0 & 0 & 0 & 0 & 1.96 & 1 & 1 & 2 & 1 \\
0 & 0 & 0 & 1 & 5.854 & 1 & 1 & 3 \\
0 & 0 & 0 & 2 & 2 & 2.668 & 3 & 1 \\
0 & 0 & 0 & 5 & 1 & 1 & 3 & 1 \\
1.96 & 1 & 2 & 5 & 31.812 - \mu & 0 & 0 & 0 \\
1 & 5.854 & 2 & 1 & 0 & -2.236 - \mu & 0 & 0 \\
1 & 1 & 2.668 & 1 & 0 & -6.504 - \mu & 0 & 0 \\
2 & 1 & 3 & 3 & 0 & 0 & 2 - \mu & 0 \\
1 & 2 & 3 & 1 & 0 & 0 & 0 & 2 - \mu \\
\end{bmatrix}
\]

\[
= 0
\]
Overcoming Computational Complexity in Nonlinear Optimization

At the point \( x(\lambda_i^*) \),

\[
D_3(\mu) = \begin{bmatrix}
0 & 0 & 0 & 0 & -1.736 & 1 & 1 & 2 & 1 \\
0 & 0 & 0 & 1 & -2.77 & 1 & 1 & 3 & 0 \\
0 & 0 & 0 & 2 & 2 & -2.806 & 3 & 1 & 0 \\
0 & 0 & 0 & 5 & 1 & 1 & 3 & 1 & 0
\end{bmatrix}
\]

\[
= -1.0 \cdot 10^{-17} \cdot \mu^5 + 1.208 \cdot 10^{-16} \cdot \mu^4 - 5.25 \cdot 10^{-16} \cdot \mu^3 + 9.792 \cdot 10^{-16} \cdot \mu^2 + 213.112 \cdot \mu + 3201.528
\]

At the point \( x(\lambda_4^*) \),

\[
D_4(\mu) = \begin{bmatrix}
0 & 0 & 0 & 0 & 7.182 & 1 & 1 & 2 & 1 \\
0 & 0 & 0 & 1 & 5.278 & 1 & 1 & 3 & 0 \\
0 & 0 & 0 & 2 & 2 & 9.26 & 3 & 1 & 0 \\
0 & 0 & 0 & 5 & 1 & 1 & 3 & 1 & 0
\end{bmatrix}
\]

\[
= 1.417 \cdot 10^{-15} \cdot \mu^5 - 1.667 \cdot 10^{-14} \cdot \mu^4 + 6.625 \cdot 10^{-14} \cdot \mu^3 - 1.003 \cdot 10^{-13} \cdot \mu^2 - 84200.269 \cdot \mu + 108348.739
\]

The roots of the polynomials \( D_1(\mu), D_2(\mu), D_3(\mu) \) and \( D_4(\mu) \) will determine the nature of the respective critical points. The roots are computed with the help of the MathCAD polyroots algorithm. First we form a vector of the coefficients staring with the constant term.

The roots are then given by

\[
\text{polyroots(v1)} = \begin{bmatrix}
-5.73 \cdot 10^4 + 5.73 \cdot 10^4 i \\
-5.73 \cdot 10^4 - 5.73 \cdot 10^4 i \\
-1.68
\end{bmatrix}
\]

\[
\text{polyroots(v2)} = \begin{bmatrix}
-6.794 \cdot 10^4 \\
6.776 - 6.794 \cdot 10^4 i \\
6.776 + 6.794 \cdot 10^4 i
\end{bmatrix}
\]

\[
\text{polyroots(v3)} = \begin{bmatrix}
-108348.739 \\
-84200.269 \\
1.417 \cdot 10^{-15}
\end{bmatrix}
\]

\[
\text{polyroots(v4)} = \begin{bmatrix}
-423487.543 \\
-271702.261 \\
2.75 \cdot 10^{-15}
\end{bmatrix}
\]
From the above we observe that of the computed polynomials $D_1(\mu), D_2(\mu), D_3(\mu), D_4(\mu)$, only $D_1(\mu)$ possesses a single negative real root $\mu = -1.69$. The other polynomials possess real roots; however they are of mixed signs and therefore cannot be used to determine the nature of the critical points.

We conclude therefore that for the given problem only one critical point exists corresponding to a maximum, i.e. $x^* = (-0.721, 0.039, -1.52, 1.128, 3.704)$ corresponds to a maximum value of the objective function given by $f(x^*) = (-0.721)^2 + (0.039)^2 + (-1.52)^2 + (1.128)^2 + (3.704)^2 = 17.824$.

3 Conclusion

In this paper we have demonstrated the possibility of solving a complex nonlinear optimization problem involving five variables and four equality constraints. The difficulties encountered here are three fold, namely:

(i) Solving highly nonlinear systems of equations involving the Lagrange multipliers.
(ii) Evaluating the determinant of very large matrices.
(iii) Computing the roots of polynomials of order five.

The above difficulties were easily overcome with the help MathCAD software which proved very efficient in generating numerical solutions to the system of nonlinear equations, evaluating matrix determinants with its symbolic capabilities and computing roots of polynomials with ease.

A similar problem of higher dimension could easily be solved using this approach. The advantage of MathCAD over other applications such as MATHEMATICA or MAPLE is that mathematical expressions and equations can be entered into the spreadsheet directly without the need for conversion into a special format.

It is our conviction that the above research and methods are important and relevant to mathematical modeling and can help students and interested researchers in better understanding nonlinear optimization, especially when the problem is multidimensional.

Furthermore, we have shown that with powerful software such as MathCAD it is possible to overcome computational complexities in nonlinear optimization.

References


