INTERNATIONAL JOURNAL OF LATEST TECHNOLOGY IN ENGINEERING, MANAGEMENT & APPLIED SCIENCE (IJLTEMAS)

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

On Iterative Methods in Optimization

Henrietta Nkansah[1]* & Bismark Kwao Nkansah[2]

[1]Department of Mathematics, University of Cape Coast, Ghana

[2]Department of Statistics, University of Cape Coast, Ghana

***Corresponding Author**

DOI : https://doi.org/10.51583/IJLTEMAS.2024.1311014

Received: 18 November 2024; Accepted: 30 November 2024; Published: 16 December 2024

Abstract: The paper highlights a failure in the implementation of a recommendation for the modified Newton's method using a Rosenbrock type of functions that have slow convergence with two minimum points as test functions. The study finds that a recommended procedure, if the Hessian $\mathbf{H}(\mathbf{x}_k)$ at a point is not positive definite, may not lead to the desired optimal solution particularly when the initial point is not close enough to the expected solution. It has been demonstrated how to go round this problem. The results show that more than one technique may be required to determine all critical points of a given function.

Keywords: Optimization, Rosenbrock's function, Modified Newton's Method, Descent direction

I. Introduction

In some gradient-based unconstrained optimization techniques a non-positive definite Hessian matrix of the problem function possesses challenges for convergence of the solution. For the Newton's method in particular, various modifications have been suggested (e.g., Fiacco & McCormick, 1967; Marquardt, 1963) that incorporates the Steepest Descent (SD) method to obtain a new direction that probably will help to get to the minimum.

The convergence rate of an optimization problem may be influenced by a number of factors that includes the role of the underlying optimization method. Others involves the global or local nature of the convergence (Lewis & Nash, 2006). For studies on convergence, a typical test function of the type of the Rosenbrock's function (Emiola & Adem, 2021) comes handy for studying robustness of gradient-based optimization algorithms.

In Section 2, we provide a description of the illustrative problem functions and review how the upper bound on the convergence rate, expressed as a function of the level of ill-conditioning, poses a challenge for the optimization process. It is known that obtaining the desired critical point depends on the initial point. We demonstrate in this paper that with the appropriate search direction, it is possible to reach the desired optimal point even with a (reasonably) distant starting point and for highly ill-conditioned problem function for the Newton's method. In Section 3, the Newton's method as well as the Modified Newton's method is presented. In the process, the problem of interest of the study is highlighted that points out the failure in convergence in spite of known recommendations in the literature and prescribes a remedy. Throughout the implementations, we set a tolerance of $\varepsilon = 10^{-5}$ for which $\nabla f(\mathbf{x}_k) = [0; 0]$ for a function $f : \Re^p \to \Re$. In Section 4, the summary of the proposed procedure is presented and followed by the conclusion.

II. Illustrative Functions and Effect of Ill-Conditioning on Convergence

Two main types of functions are used in this paper. The functions are selected to illustrate the effect of ill-conditioning on the determination of optimal points.

Test Function I

We consider the problem of minimizing the function

$$f_1(\mathbf{x}) = x_1^4 + x_1 x_2 + (1 + x_2)^2 \quad (2.1)$$

The graph of $f_1(x)$ with almost flat base is given in Figure 1. The graph shows that locating a minimum of the function would pose a challenge.

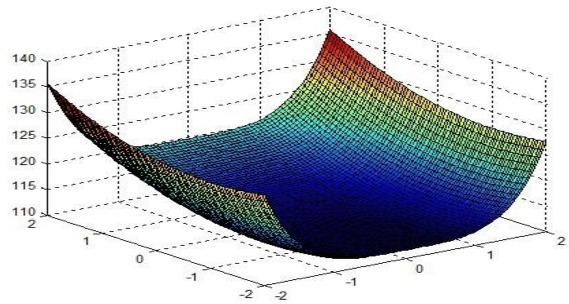

Figure 1: Graph of $f_1(\mathbf{x}) = x_1^4 + x_1x_2 + (1+x_2)^2$

Using the same starting point $\mathbf{x}_0 = [0;\ 0]$, the Newton's method and Conjugate Gradient method, respectively, lead to two different critical points given as $[-0.204128;\ 0.084945]$ and $[0.69588;\ -1.34794]$

The results show that $f_1(\mathbf{x})$ has multiple critical points which may be difficult to detect using a single method. The paper highlights the challenge in using the modified Newton's method to arrive at the same point as obtained by the Conjugate Gradient method.

Test Function II

We consider the problem of minimizing the Rosenbrock's function given as

$$f_2(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \tag{2.2}$$

using the steepest descent (SD) method. Table 1 gives some iterates in the application of the SD to the function using a starting point $\mathbf{x}_0 = [0;\ 0]$.

Table 1: Number of iterations and condition number of the Hessian of $f_2(x)$

Iterate (k)	**x**	$\kappa(\mathbf{H})$
0	[0; 0]	100.00
1	[0.15541; 0]	20.829
10	[0.31095; 0.08560]	60.578
100	[0.58593; 0.33975]	331.81
1000	[0.92741; 0.85909]	1653.2

From the table, there is slow rate of convergence of $f_2(\mathbf{x})$ which can be attributed to a large condition number, $\kappa(\mathbf{H})$. Even after 1000 iterations, the algorithm shows only slow convergence to the exact minimum $[1;\ 1]$. The graph of $f_2(\mathbf{x})$ within the interval

$[X, Y] = (-2:0.1:2, -2:0.1:2)$ is given in Figure 2.

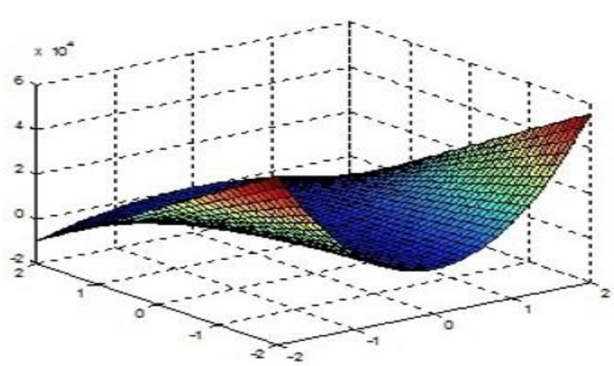

Figure 2: Graph of the Rosenbrock's Function

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

The rate of convergence of the function is explained by the long-valley base as shown in Figure 2. In spite of this, there is a single minimum point of the function.

The observations illustrate a major weakness of the SD method that it is not appropriate when the Hessian matrix is ill-conditioned. They also show that the determination of a minimum point of $f_1(\mathbf{x})$ could be more daunting than that of $f_2(\mathbf{x})$. It will be seen in the next section that the implementation of the SD for $f_1(\mathbf{x})$ converges after a few iterations rather to a point in the neighbourhood of the initial point but not to the desired optimum point.

Linear Convergence for the Case of a Quadratic Function

An algorithm exhibits linear convergence in the objective function values if there is a constant $\delta < 1$ such that for all k sufficiently large, the iterates $\mathbf{x}_k$ satisfy the expression

$$\frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} \le \delta$$

(2.3)

where $\mathbf{x}^*$ is some optimal point of the problem. By this statement, the optimality gap shrinks by at least δ at each iteration and speeds up the rate of convergence if δ is not close to 1. Consider the case in which the objective function $f(\mathbf{x})$ is a simple quadratic function of the form

$$f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x} \ ,$$

where **A** is a positive definite symmetric matrix, and suppose that the eigenvalues of **A** are $\lambda_1 \ge \lambda_2 \ge \ldots \ge \lambda_p > 0.$ The optimal solution of the problem is computed as

$$\mathbf{x}^* = -\mathbf{A}^{-1}\mathbf{b}$$

and by direct substitution, the optimal objective function value is

$$f(\mathbf{x}^*) = -\frac{1}{2}\mathbf{b}^T\mathbf{A}^{-1}\mathbf{b}$$

Let $\mathbf{x}_k$ denote the current point in the SD algorithm and let $\mathbf{d}_k$ denote the current direction given by

$$\mathbf{d}_k = -\nabla f(\mathbf{x}_k) = -\mathbf{A}\mathbf{x}_k - \mathbf{b}$$

(2.4)

To obtain the next iterate of the SD algorithm, we compute the step size, α, by considering

$$\begin{aligned} f(\mathbf{x}_k + \alpha\mathbf{d}_k) &= \frac{1}{2}(\mathbf{x}_k + \alpha\mathbf{d}_k)^T\mathbf{A}(\mathbf{x}_k + \alpha\mathbf{d}_k) + \mathbf{b}^T(\mathbf{x}_k + \alpha\mathbf{d}_k) \\ &= f(\mathbf{x}_k) - \alpha\mathbf{d}_k^T\mathbf{d}_k + \frac{1}{2}\alpha^2\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k \end{aligned}$$

Optimizing the value of α therefore yields

$$\alpha = \frac{\mathbf{d}_k^T\mathbf{d}_k}{\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k}.$$

and the next iterate is given as

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \frac{\mathbf{d}_k^T\mathbf{d}_k}{\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k}\mathbf{d}_k$$

and

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

$$f(\mathbf{x}_{k+1}) = f(\mathbf{x}_k) - \frac{1}{2}\frac{\left(\mathbf{d}_k^T\mathbf{d}_k\right)^2}{\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k}$$

From the results above, we obtain

$$\begin{aligned} f(\mathbf{x}) - f(\mathbf{x}^*) &= \frac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x}_k + \mathbf{b}^T\mathbf{x}_k + \frac{1}{2}\mathbf{b}^T\mathbf{A}^{-1}\mathbf{b} \\ &= \frac{1}{2}(\mathbf{A}\mathbf{x}_k + \mathbf{b})^T\mathbf{x}_k + \frac{1}{2}\mathbf{b}^T\mathbf{x}_k + \frac{1}{2}\mathbf{b}^T\mathbf{A}^{-1}\mathbf{b} \\ &= \frac{1}{2}(\mathbf{A}\mathbf{x}_k + \mathbf{b})^T\mathbf{x}_k + \frac{1}{2}\mathbf{b}^T\mathbf{A}^{-1}(\mathbf{A}\mathbf{x}_k + \mathbf{b}) \\ &= \frac{1}{2}(\mathbf{A}\mathbf{x}_k + \mathbf{b})^T(\mathbf{x}_k + \mathbf{b}^T\mathbf{A}^{-1}) \\ &= \frac{1}{2}(\mathbf{A}\mathbf{x}_k + \mathbf{b})^T\mathbf{A}^{-1}(\mathbf{A}\mathbf{x}_k + \mathbf{b}) \\ &= \frac{1}{2}\mathbf{d}_k^T\mathbf{A}^{-1}\mathbf{d}_k \end{aligned}$$

Thus,

$$\begin{aligned} \frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} &= \frac{f(\mathbf{x}_k) - \frac{1}{2}\frac{(\mathbf{d}_k^T\mathbf{d}_k)^2}{\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k} - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} \\ &= 1 - \frac{(\mathbf{d}_k^T\mathbf{d}_k)^2}{(\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k)(\mathbf{d}_k^T\mathbf{A}^{-1}\mathbf{d}_k)} \\ &= 1 - \frac{1}{\beta} \end{aligned} \tag{2.5}$$

where

$$\beta = \frac{(\mathbf{d}_k^T\mathbf{A}\mathbf{d}_k)(\mathbf{d}_k^T\mathbf{A}^{-1}\mathbf{d}_k)}{(\mathbf{d}_k^T\mathbf{d}_k)^2} \tag{2.6}$$

By the Kantorovich inequality, an upper bound on the value of β is given (Huang & Zhou, 2005) as

$$\beta \leq \frac{(\lambda_1 + \lambda_p)^2}{4\lambda_1\lambda_p}.$$

Now, applying this inequality in Equation (2.3) gives (Freund, 2004)

$$\begin{aligned} \frac{f(\mathbf{x}_{k+1}) - f(\mathbf{x}^*)}{f(\mathbf{x}_k) - f(\mathbf{x}^*)} &\leq 1 - \frac{4\lambda_1\lambda_p}{(\lambda_1 + \lambda_p)^2} \\ &= \frac{(\lambda_1 - \lambda_p)^2}{(\lambda_1 + \lambda_p)^2} \\ &= \left(\frac{\frac{\lambda_1}{\lambda_p} - 1}{\frac{\lambda_1}{\lambda_p} + 1}\right)^2 \\ &=: \delta \end{aligned}$$

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

Since by definition $\kappa(\mathbf{H}) = \frac{\lambda_1}{\lambda_p} \geq 1$, if the ratio is large, then the convergence constant δ will be close to (slightly smaller than) 1, which slows down the convergence rate. Thus, a much smaller value than 1 of δ is desired.

Example

Consider $f_2(x)$ in Equation (2.2). With a starting point of $\boldsymbol{x}_0 = [0; 0]$, the descent direction $\mathbf{d} = -\nabla f(\mathbf{x}_0) = [2;\ 0]$. Table 2 gives the sensitivity of SD convergence rate to the eigenvalue ratio.

Table 2: Sensitivity of SD Convergence Rate to the Eigenvalue Ratio

λ_1	λ_p	Upper Bound on δ	k	$\kappa(\mathbf{H})$
237.21622	7.97875	0.87407	2	29.731
279.17358	4.60848	0.93609	10	60.578
476.63669	1.43646	0.98800	100	331.81
744.95842	0.66666	0.99634	500	1117.4
889.92511	0.53830	0.99757	1000	1653.2

From Table 2, $\kappa(\mathbf{H})$ keeps increasing as we increase the number of iterations, *k*. It also shows the effect of the relationship between $k(\boldsymbol{H})$ and the upper bound on δ on the convergence rate. We see that the convergence constant ranges from 0.87407 to 0.99757, implying that for functions of the type $f_2(\boldsymbol{x})$, the convergence rate of SD could be extremely slow.

III. Methods and Illustrative Problem

Newton's Method

The multi-dimensional Newton's method finds a stationary point of $f : \Re^p \to \Re$ by solving the non-linear system of equations $\nabla f(\mathbf{x}) = 0$. Alternatively, using the Taylor series for a function of several variables gives

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T(\mathbf{x}-\mathbf{x}_0) + \frac{1}{2}(\mathbf{x}-\mathbf{x}_0)^T\mathbf{H}(\mathbf{x}_0)(\mathbf{x}-\mathbf{x}_0) + \cdots$$

(3.1)

Where $\mathbf{H}(\mathbf{x})$ (also denoted by $\nabla^2 f(\boldsymbol{x})$) is the Hessian matrix, if we take $\mathbf{x}_0$ to be $\mathbf{x}^*$, and since $\nabla f(\boldsymbol{x}^*)$ is zero, we have

$$f(\mathbf{x}) = f(\mathbf{x}^*) + \nabla f(\mathbf{x}^*)^T(\mathbf{x}-\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T\mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) + \cdots$$

$$= f(\mathbf{x}^*) + \frac{1}{2}(\mathbf{x}-\mathbf{x}^*)^T\mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) + \cdots$$

Since $f(\boldsymbol{x}^*)$ is the local minimum value of f, it must follow that

$$(\mathbf{x}-\mathbf{x}^*)^T\mathbf{H}(\mathbf{x}^*)(\mathbf{x}-\mathbf{x}^*) \geq 0$$

at least for $\mathbf{x}$ near $\mathbf{x}^*$. Using Newton's method to find the roots of $\nabla f(x) = 0$, we arrive at the minimization method (assuming that $\mathbf{H}(\mathbf{x}^*)$ is positive definite)

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$$

Hence,

$$\mathbf{H}(\mathbf{x}_k)(\mathbf{x}_{k+1} - \mathbf{x}_k) = -\nabla f(\mathbf{x}_k)$$

(3.2)

We seek now to:

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

1. solve for the step $\delta_k = \mathbf{x}_{k+1} - \mathbf{x}_k$;
2. compute $\mathbf{x}_{k+1}$ from $\mathbf{x}_{k+1} = \mathbf{x}_k + \delta_k$.

The Illustrative Problem

We consider the minimization problem

$$\min: \ f_1(\mathbf{x}) = x_1^4 + x_1x_2 + (1+x_2)^2$$

Using an initial guess of $\mathbf{x}_0 = [0.75; -1.25]$ we obtain $\nabla f(\mathbf{x}_0) = [0.43750; 0.2500]$ and

$\mathbf{H}(\mathbf{x}_0) = \begin{pmatrix} 6.74999 & 1 \\ 1 & 2 \end{pmatrix}$. By repeated use of Equation (3.2), we obtain the results in Table 3.

Table 3: Iterations for minimizing $f_1(\boldsymbol{x})$ using the Newton's method

k	$\mathbf{x}_k$	$\nabla f(\mathbf{x}_k)$
0	[0.75; -1.25]	[0.43750; 0.25000]
1	[0.70; -1.35]	$[2.2\times10^{-2}; -5.5511\times10^{-12}]$
2	[0.69591; -1.34796]	$[1.3104\times10^{-4}; -1.0\times10^{-5}]$
3	[0.69588; -1.34794]	$[-2.3294\times10^{-5}; 5.5511\times10^{-12}]$

Thus, from Table 3, $\nabla f(\mathbf{x}_3) = [0; 0]$ and $\mathbf{H}(\mathbf{x}_3) = \begin{pmatrix} 5.81097 & 1 \\ 1 & 1.99999 \end{pmatrix}$ is positive definite.

Thus, $\mathbf{x}_3 = [0.69588; -1.34794]$ is a minimum point of $f(\mathbf{x})$. As noted in Section 2, the minimum point of this function under Conjugate Gradient method is the same as $\boldsymbol{x}_3$ and is attained in just one iteration with the starting point $\mathbf{x}_0 = [0; 0]$ which is farther away from $\mathbf{x}_3$ than the starting point $\mathbf{x}_0 = [0.75; -1.25]$ used in this case. This presents the problem of interest in this paper.

Now, suppose that $\mathbf{x}_0 = [0; 0]$ in this example. Then, $\nabla f(\mathbf{x}_0) = [0; 2]$ and

$\mathbf{H}(\mathbf{x}_0) = \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix}$. which is indefinite and hence a saddle point at $\mathbf{x}_0 = [0; 0]$. This means that the expression $\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$ in the Newton's method fails to be a descent direction.

To ensure a descent direction, we make use of an alternative direction $\boldsymbol{v}_k$ which is the eigenvector corresponding to the negative eigenvalue of the Hessian matrix at $\mathbf{x}_k$ in the SD method. That is,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k v_k \quad (3.3)$$

Using this method, we compute the various iterates for given values of the step size α_k and v_k. These iterations are given in Table 4.

Table 4: Optimization of $f_1(\mathbf{x})$ for various step sizes under SD method

k	α_k	v_k.	$\mathbf{x}_k$
0	0.21571	[-0.92388; 0.38268]	[0; 0]
1	0.0051142	[-0.89608; 0.44388]	[-0.199292; 0.082549]

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

2	0.00024236	[-0.89451; 0.44704]	[-0.203875;	0.084819]
3	0.000040420	[-0.89444; 0.44719]	[-0.204092;	0.084927]
4	0.000040420	[-0.89443 ; 0.44722]	[-0.204128;	0.084945]
5	0.000040420	[-0.89443 ; 0.44722]	[-0.204128;	0.084945]

We see from Table 4 that the iterations converge after four iterations to the critical point $[-0.204128; 0.084945]$. It is interesting to note that this minimum point is different from $[0.69588; -1.34794]$ obtained under the Conjugate Gradient method after only one iteration, and under the Newton's method after three iterations using the initial point $[0.75; -1.25]$.

The situation encountered so far is that the Hessian at a point is indefinite and therefore could not continue with the Newton's method. At this point, the steps taken to obtain a positive definite Hessian basically combines the Newton's method with the SD method. It should be noticed that even though this combined approach (McMormick & Fiacco, 1967) produces a solution, this solution is not what is intended (see Table 3). To achieve the desired result, we first review the modified Newton's method.

Modified Newton's Method

In the Newton's method,

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$$

the expression

$$s_k = -\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$$
(3.4)

is a descent direction if $\mathbf{H}^{-1}(\mathbf{x}_k)$ is positive definite. The algorithm for damped Newton's method is

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$$
(3.5)

for some damping sequence

$$\{\alpha_k\}_{k=0}^{\infty}, \quad 0 < \alpha_k \le 1, \text{ and } \quad \alpha_k \to 1 \text{ as } \quad k \to \infty .$$

When $\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$ is not a descent direction, a modified Newton's method substitutes a descent direction s_k for $\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$ From Equation (3.5), the modified Newton's method becomes the SD method

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \nabla f(\boldsymbol{x}_k) , \tag{3.6}$$

by taking $\mathbf{H}^{-1}(\mathbf{x}_k) = \mathbf{I}$, the identity matrix. In most cases, it is still possible to compute s_k from

Equation (3.4) and to search along $\pm s_k$, the sign chosen to ensure a descent direction.

A theoretical argument against the generalized Newton's method is that if $\mathbf{H}(\mathbf{x}_k)$ is not a positive definite matrix, a move in the direction given by Equation (3.5) where $\alpha_k > 0$ is chosen to minimize f along s_k in Equation (3.4) starting from x_k, may result in an increase rather than a decrease in $f(x)$, yielding $\alpha_k = 0$, which terminate the process at x_k. Another objection is that $\mathbf{H}(\mathbf{x}_k)$ may not have an inverse, even if $f(x)$ is convex. The modified second-order method takes into account these two objections. The direction vector s_k is generated according to two rules (McMormick & Fiacco, 1967). In both cases,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k s_k$$

where α_k is chosen to be the smallest value of $\alpha \ge 0$ for which $\boldsymbol{x}_k + \alpha_k s_k$ gives a local minimum of $f(x)$. The rules are as follows:

1. If $\mathbf{H}(\mathbf{x}_k)$ has a negative eigenvalue, let s_k be a vector where

$$s_k^T \mathbf{H}(\mathbf{x}_k) s_k < 0 \text{ and } s_k^T \nabla f(\mathbf{x}_k) \le 0$$

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

2. If $\mathbf{H}(x_k)$ has all eigenvalues greater than or equal to zero, choose $\boldsymbol{s}_k$ such that either

$$\mathbf{H}(x_k)s_k = 0 \text{ , and } s_k^T \nabla f(\mathbf{x}_k) < 0$$

or

$$\mathbf{H}(\mathbf{x}_k)s_k = -\nabla f(\mathbf{x}_k)$$

(3.7)

The rationale for Rule 1 is that if the second partial derivative matrix has a negative eigenvalue there are certain directions along which the function $f(x)$ decreases and along which the *rate* of decrease also decreases. Now, to obtain s_k, we use the factorization

$$\mathbf{H}(\mathbf{x}_k) = \mathbf{LDL}^T$$

(3.8)

where **L** is a non-singular lower triangular matrix and $\mathbf{D} = \mathrm{diag}(d_1, d_2, \ldots, d_p)$. The conditions for the factorization are as follows:

1. If **D** has all positive diagonal elements, solve for $s_k = -\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$ ..
2. If **D** has all non-negative diagonal elements, and at least one is zero, the vector $\boldsymbol{s}_k$ is generated according to the second rule above.
3. If **D** has some diagonal elements that are negative, solve $\mathbf{L}^T\mathbf{v} = \mathbf{a}_k$, where

$$\mathbf{a}_k = \begin{cases} 1, & \text{if } d_k < 0 \\ 0, & \text{otherwise} \end{cases}$$

Let

$$s_k = \begin{cases} \mathbf{v}, & \text{if } \mathbf{v}^T\nabla f(\mathbf{x}_k) \le 0 \\ -\mathbf{v}, & \text{otherwise} \end{cases}$$

Other forms of modification are to:

1. Modify the Newton's search direction by giving it a bias towards the steepest descent vector $-\nabla f(\boldsymbol{x}_k)$ (Levenberg, 1944; Marquardt, 1963). This is achieved by adding a scalar multiple of the unit matrix to $\mathbf{H}(x_k)$ and solving the system

$$(\mathbf{H}(\mathbf{x}_k) + \nu\,\mathbf{I})s_k = -\nabla f(\mathbf{x}_k)$$

to obtain **v** so that the matrix $\mathbf{H}(\mathbf{x}_k) + \nu\,\mathbf{I}$ is positive definite.

2. Modify the Hessian matrix in the form (Murray, 1972; Hebden, 1973) as

$$\mathbf{H}(\mathbf{x}_k) + \mathbf{D}$$

where **D** is diagonal and it is used to determine the search direction. The modification occurs as the matrix is being factorized.

The Illustrative Problem Continued

For $f_1(\boldsymbol{x})$ with a starting point of $\boldsymbol{x}_0 = [0; 0]$, it has already been found that $\boldsymbol{H}(x_0)$ is indefinite and that $s_k = -\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$ is not a descent direction. Therefore, we look for a new direction, **v**. The Newton's method then becomes $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{v}_k$.

To obtain **v**, we factorize the Hessian matrix $\mathbf{H}(\mathbf{x}_0)$ as in Equation (3.8) to obtain

$$\mathbf{D} = \begin{pmatrix} 2 & 0 \\ 0 & -0.5 \end{pmatrix}.$$

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XIII, Issue XI, November 2024

Now, because a diagonal entry is negative, we use the third condition to obtain $\boldsymbol{a} = [0; 1]$, and hence $\mathbf{v} = [1; -0.5]$ which is the required descent direction. Using the SD algorithm to compute the first iterate, we obtain $\mathbf{x}_1 = [-0.20363; 0.10181]$ with the Hessian

$$\mathbf{H}(\mathbf{x}_1) = \begin{pmatrix} 0.49758 & 1 \\ 1 & 1.99999 \end{pmatrix}$$

which is not positive definite. The process therefore cannot continue. Thus, the recommendation of Fiacco and McCormick (1967) seizes to work.

In order to overcome this problem, we need to search along a different direction which is in the same direction as $\mathbf{v}$. We observe that if we choose $\mathbf{x}_2 = \nabla f(\mathbf{x}_1) = [-0.23656; -0.86608]$, the Hessian given as

$$\mathbf{H}(\mathbf{x}_2) = \begin{pmatrix} 0.67150 & 1 \\ 1 & 2 \end{pmatrix}$$

is positive definite. We then switch back to the Newton's method to continue with the process. Table 5 shows the iterations of the process.

Table 5: Optimization process for $f_1(\boldsymbol{x})$ using Modified Newton's method

Method	k	$\boldsymbol{x}_k$	$\nabla f(\boldsymbol{x}_k)$	$f(\boldsymbol{x}_k)$
	0	[0; 0]	[0; 2]	1
Steepest	1	[-0.20363; 0.10181]	[0.068041; 2]	1.1950
Newton	2	[-0.23656; -0.86608]	[-0.919032; 0.031278]	0.22594
Newton	3	[5.2134; -3.6067]	$[563.17; -8; 3901 \times 10^{-6}]$	726.70
Newton	4	[3.4840; -2.7415]	$[166.41; 9.8662 \times 10^{-4}]$	140.81
Newton	5	[2.3375; -2.1688]	$[48.921; 1.9252 \times 10^{-6}]$	26.153
Newton	6	[1.5857; -1.7928]	$[14.156; 7.9298 \times 10^{-6}]$	4.1081
Newton	7	[1.1086; -1.5543]	$[3.8962; -2.6482 \times 10^{-7}]$	0.09475
Newton	8	[0.83521; -1.4176]	$[0.91289; 2.2768 \times 10^{-7}]$	-0.52299
Newton	9	[0.71923; -1.35961]	$[0.12858; 3.8630 \times 10^{-7}]$	-0.58096
Newton	10	[0.69670; -1.34835]	$[0.0043342; 6.8773 \times 10^{-8}]$	-0.58244
Newton	11	[0.69589; -1.34794]	$[5.5268 \times 10^{-6}; 1.8097 \times 10^{-9}]$	-0.58245
Newton	12	[0.69588; -1.34794]	$[-5.5511 \times 10^{-12}; 5.5511 \times 10^{-12}]$	-0.58245

It can be observed from the table that after twelve iterations, the solution converges to $\mathbf{x}_{12} = [0.69588; -1.34794]$. This solution is the same as the one obtained with a closer initial point $\mathbf{x}_0 = [0.75; -1.25]$ using the Newton's method.

IV. Summary and Conclusion

In this section, we present the summary and conclusion of the study. In the summary, the conceptualization of the procedure for correcting the failure in the convergence of the Newton's method is provided.

Summary

Suppose that Hessian $\mathbf{H}(\mathbf{x}_k)$ is not positive definite. Then $s_k = -\mathbf{H}^{-1}(\mathbf{x}_k)\nabla f(\mathbf{x}_k)$ is not a descent direction. By the factorization $\mathbf{H}(\mathbf{x}_k) = \mathbf{LDL}^T$ where $\mathbf{L}$ is a non-singular lower triangular matrix, if some elements of $\mathbf{D} = \mathrm{diag}(d_1, d_2, \ldots, d_p)$ are negative, and if for the solution of $\mathbf{L}^T\mathbf{v} = \mathbf{a}_k$, where

$$\mathbf{a}_k = \begin{cases} 1, & \text{if } d_k < 0 \\ 0, & \text{otherwise} \end{cases}$$

$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \nu_k$ does not provide a positive definite Hessian $\boldsymbol{H}(x_{k+1})$, then $\nabla f(\mathbf{x}_k - \alpha_k \nu_k)$ could provide a descent direction for the Newton's method.

Conclusion

The study has examined a failure in the implementation of some of the procedures and algorithms that are used in unconstrained optimization techniques. These methods are examined in the light of known highly ill-conditioned functions.

The study finds that a recommended modification to the Newton's method, if the Hessian $\mathbf{H}(\mathbf{x}_k)$ is not positive definite, may not lead to the desired optimal solution particularly when the initial point is not close enough to the expected solution. It has therefore been demonstrated how to go round the problem. The results show that an optimal solution of a function should be confirmed by using multiple initial points that are not too close.

References

1. Emiola, I., & Adem, R. (2021). Comparison of Minimization Methods for Rosenbrock Functions. arXiv:2101.10546 [math.OC].
2. Freund, R. M. (2004). The Steepest Descent Algorithm for Unconstrained Optimization and a Bisection Line Search Method. Massachusetts Institute of Technology.
3. Hebden, M. D. (1973). An Algorithm for Minimization using Exact Second Derivatives. AERE Harwell Report, TP515.
4. Huang, J. & Zhou, J. (2005). A direct proof and a generalization for a Kantorovich type inequality. Linear Algebra and its Applications. **397**, 185-192.
5. Levenberg, K. (1944). A Method for the Solution of Certain Nonlinear Problems in Least Squares. Quart. Appl. Math., **2**. 164-168.
6. Lewis R.M. & Nash S.G. (2006) Factors Affecting the Performance of Optimization-based Multigrid Methods. In: Hager W.W., Huang SJ., Pardalos P.M., Prokopyev O.A. (eds) Multiscale Optimization Methods and Applications. Nonconvex Optimization and Its Applications, **82**. Springer, Boston, MA.
7. Marquardt, D. W. (1963). An Algorithm for Least Squares estimation of non-linear Parameters. SIAM J., **11**, 431-441.
8. McCormick, G. P., & Fiacco, A. V. (1967). Nonlinear programming. John Wiley, New York.
9. Murray, W. (1972). Second Derivative Methods in Numerical Methods for Unconstrained Optimization. Academic Press, London.