DOGLEG METHOD WITH BROYDEN CLASS UPDATING TECHNIQUE FOR SOLVING TRUST-REGION SUB-PROBLEMS OF SOME UNCONSTRAINED MULTIVARIATE NONLINEAR OPTIMIZATION PROBLEMS

E. C. Duru[1] and **B. I. Oruh**[2]

[1,2]*Department of Mathematics, Michael Okpara University of Agriculture, Umudike, Nigeria*

Abstract

In this paper, the Dogleg-type trust-region method that employed the Broyden Class updating techniques in generating the approximation matrices to the hessian of the objective function is presented where convergence is based on constructing two paths. The conditions on the paths were incorporated into the algorithm used in determining the optimum points of the smooth functions considered. Numerical computations on some test functions showed that this procedure is efficient and globally convergent. The result equally highlighted the effect of the Broyden class parameter on the convergence of the solution.

Keywords: *Trust-region methods, Dogleg method, Broyden class updating technique.*

Introduction

The basic unconstrained optimization problem can be expressed as

$$\min f(x), \qquad x \in \Re^n \qquad \text{Eq. (1)}$$

where $\mathbb{R}^n$ is an n-dimensional Euclidean space and $f: \mathbb{R}^n \to \mathbb{R}$ is a twice continuously differentiable function on $\mathbb{R}^n$ (i.e. f is smooth). Solving Eq. (1) by trust-region methods have become popular in recent times owing to the fact that trust-region methods converges faster than its' associated line search methods (Gertz, 2004). Another reason for this is the wide application of trust-region methods in many fields, such as science, engineering and economy because of its strong global convergence and robustness (Dai and Xu, 2003), (Dennis *et al*, 1997), (*Shultz et al*, 1985).

Given some unconstrained optimization problems of the form Eq. (1) whose associated trust-region sub-problem is of the form:

$$Min\, m_k(d) = f_k + \nabla f_k^T d + \frac{1}{2} d^T B_k d$$

$$s.t \quad \| d \| \le \Delta_k \qquad \text{Eq. (2)}$$

we seek a solution of the form

$$x_{k+1} = x_k + d_k \qquad \text{Eq. (3)}$$

where x_{k+1} is the current iterate, x_k is the previous iterate, B_k is the approximation matrix which will be generated at each k by a Broyden class updating technique, f_k is the gradient of f at each k, d_k is the search direction at each k, Δ_k is the trust-region radius, $\| d \|$ is the Euclidean norm on the search direction and m_k represents the trust-region sub-problem.

According to Oruh and Bamigbola (2013) the trust-region strategy works this way:

Given a bound Δ_k called the trust-region radius, and a current iterate $x_k \in \mathbb{R}^n$ to the solution of Eq. 2, define a model $m_k: \mathbb{R}^n \rightarrow \mathbb{R}$ of the objective function $f(x_k)$ in the region $\beta_k = \left\{ x \in R^n : \left\| x - x_k \right\| \leq \Delta_k \right\}$ called the trust-region, surrounding the current iterate x_k where the model is trusted to be an accurate representation of $f(x_k)$. This model is often assumed to be quadratic as shown in Eq. (2).

According to Gertz (2004), trust-region methods define each iterate as the approximate minimizer of a relatively simple model function within a region in which the algorithm trusts that the model function behaves like the objective function.

Trust-region methods have generated lots of attention in recent times with many publications on them because they offer faster convergence than their line search counterparts (Tong and Zhou, 2006), (Qu and Jiang, 2008), (Yuan *et al*., 2009), (Li *et al*., 2015) and (Burdakov *et al*., 2017).

The basic trust-region method given by Conn *et al.* (1987) is summarized by the following algorithm:

Algorithm 1:

Step 0: Initialization. An initial point x_0 and an initial trust-region radius Δ_0 are given. The constants η_1, η_2, γ_1 *and* γ_2 are also given and satisfy

$$0 < \eta_1 \leq \eta_2 < 1 \quad and \quad 0 < \gamma_1 \leq \gamma_2 < 1$$

Compute $f\left(x_0\right)$ and set $k = 0$.

Step 1: Model definition. Choose $\| d \|$ and define a model m_k in β_k.

Step 2: Step calculation. Compute a step d_k that sufficiently reduces the model m_k and such that $x_k + d_k \in \beta_k$.

Step 3: Acceptance of the trial point. Compute $f\left(x_k + d_k\right)$ and define

$$\rho_k = \frac{f\left(x_k\right) - f\left(x_k + d_k\right)}{m_k\left(d_k\right) - m_k\left(x_k + d_k\right)}$$

If $\rho_k \geq \eta_1$, then define $x_{k+1} = x_k + d_k$, otherwise define $x_{k+1} = x_k$

Step 4: Trust-region radius update. Set $\Delta_{k+1} = \begin{cases} \left[\Delta_k, \infty\right) & if\ \rho_k \geq \eta_2 \\ \left[\gamma_2 \Delta_k, \Delta_k\right] & if\ \rho_k \in \left[\eta_1, \eta_2\right] \\ \left[\gamma_1 \Delta_k, \gamma_2 \Delta_k\right] & if\ \rho_k < \eta_1 \end{cases}$

Increase k by 1 and go to step 1.

According to Conn *et al.* (1987), iterations for which $\rho_k \geq \eta_1$ and thus for which $x_{k+1} = x_k + d_k$ are called successful iterations and denoted by the set by $S = \{k \geq 0 \mid \rho_k \geq \eta_1\}$. Similarly, the set $\Upsilon = \{k \geq 0 \mid \rho_k \geq \eta_2\}$ represent the set of very successful iterations with $\Upsilon \subseteq S$.

Methods that find an approximate d_k that clearly solves Eq. (2) has been adapted over the years and Toint (1986) states that whatever method one chooses for computing d_k must perform at least better than the Cauchy point method.

According to Conn *et al.* (1987), the Cauchy point examines the behavior of the model along the steepest descent $-g_k$ within the trust-region β_k described in Fig. 1.

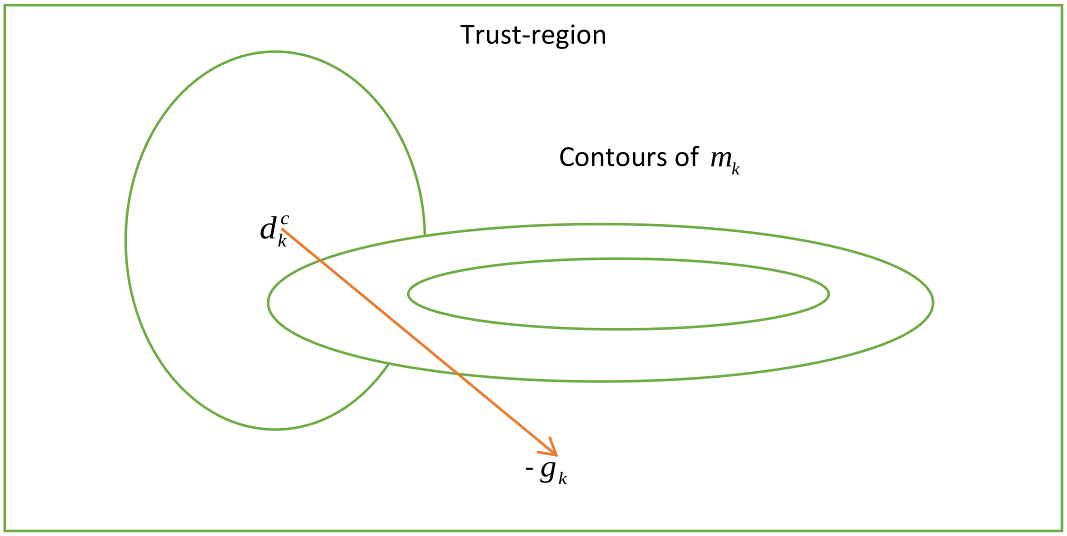


Fig. 1: The Cauchy Point Arc

The Cauchy point is given by: $$d_k^c = -\tau_k \frac{\Delta_k}{\|\nabla f_k\|} \nabla f_k \quad \text{Eq. (4)}$$

where $$\tau_k = \begin{cases} 1 & \text{if } \nabla f_k^T B_k \nabla f_k \le 0 \\ \min\left(\dfrac{\|\nabla f_k\|^3}{\Delta_k \nabla f_k^T B_k \nabla f_k}, 1 \right) & \text{otherwise} \end{cases} \quad \text{Eq. (5)}$$

Nocedal and Wright (1999) said that implementing the Cauchy point is equivalent to implementing the steepest descent method with a particular choice of step length and it performs poorly even if an optimal step-length is used at each iteration. This necessitated the need for a better approximate solution. Hence, methods like Dogleg, Steihaug, Two-subspace minimization, etc. But, in this work we will consider the Dogleg method which is more suitable when the Hessian matrix is positive definite.

However, Algorithm 1 was modified in this work and is presented below in Algorithm 2.

Algorithm 2: (Oruh and Duru, 2019)

Step 0: Initialization. An initial point x_0 and an initial trust-region radius Δ_0 are given. The constants $\eta_1,\ \eta_2,\ \gamma_1$ *and* γ_2 are also given and satisfy

$$0<\eta_1\le\eta_2<1 \quad and \quad 0<\gamma_1\le\gamma_2<1$$

Compute $f(x_0)$ and set $k = 0$.

Step 1: Model definition. Choose $\|d\|$ and define a model m_k in β_k.

Step 2: Step calculation. Compute a step d_k that sufficiently reduces the model m_k and such that $x_k+d_k\in\beta_k$.

Step 3: Acceptance of the trial point. Compute $f(x_k+d_k)$ and define

$$\rho_k=\frac{f(x_k)-f(x_k+d_k)}{-m_k(d_k)}$$

If $\rho_k\ge\eta_1$, then define $x_{k+1}=x_k+d_k$, otherwise define $x_{k+1}=x_k$

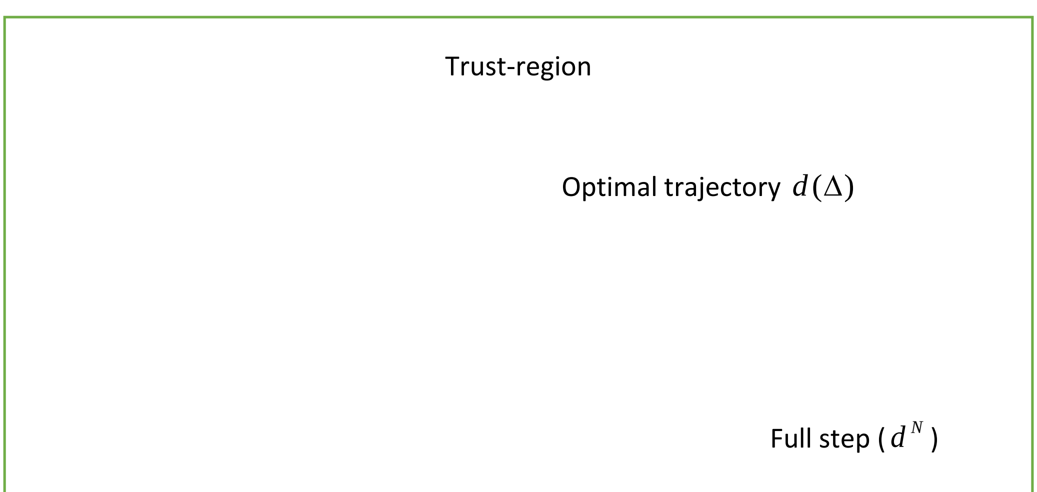
Step 4: Trust-region radius update. Set $\Delta_{k+1}=\begin{cases}[\Delta_k,\infty) & if\ \rho_k\ge\eta_2\\ [\gamma_2\Delta_k,\Delta_k] & if\ \rho_k\in[\eta_1,\eta_2]\\ [\gamma_1\Delta_k,\gamma_2\Delta_k] & if\ \rho_k<\eta_1\end{cases}$

Increase k by 1 and go to step 1.

In step 3, algorithm 2 we used $\rho_k=\dfrac{f(x_k)-f(x_k+d_k)}{-m_k(d_k)}$ instead of $\rho_k=\dfrac{f(x_k)-f(x_k+d_k)}{m_k(d_k)-m_k(x_k+d_k)}$ because the later failed in our work. $f(x_k)-f(x_k+d_k)$ is called the actual reduction while $m_k(d_k)-m_k(x_k+d_k)$ is called the predicted reduction. $m_k(d_k)=g_k^Td_k+\frac{1}{2}d_k^TB_kd_k$ and $m_k(x_k+d_k)=g_k^T(x_k+d_k)+\frac{1}{2}(x_k+d_k)^TB_k(x_k+d_k)$. This means that our predicted reduction is evaluated at the current search direction d_k at each k and not on the straight-line connecting d_k and (x_k+d_k) within the trust-region.

2. The Dogleg Method

According to Oruh and Bamigbola (2013), the dogleg method for solving the trust-region sub-problem originated from Dennis and Schnabel (1983) and they said that the Dogleg methods find an approximate solution for the trust-region sub-problem by replacing the curved trajectory of the Cauchy point with a path consisting of two line segments as shown in Fig. 2.



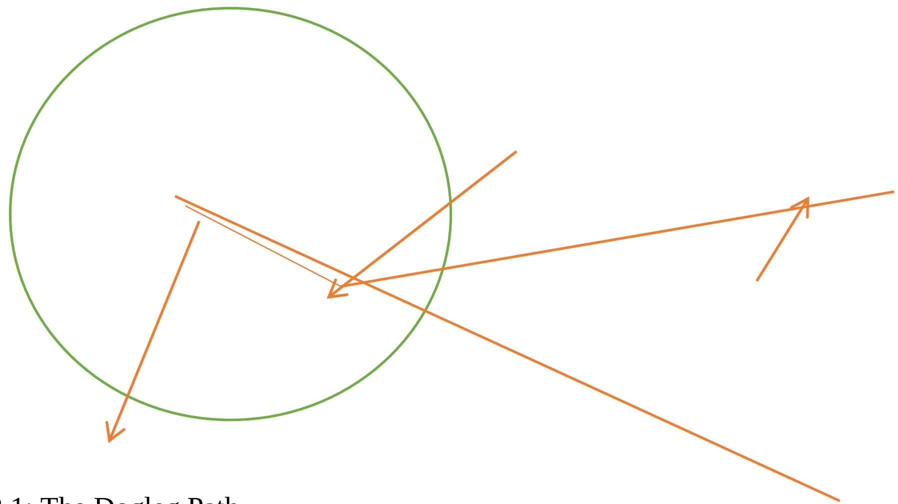


Fig. 2.1: The Dogleg Path

Fig. 2: The Dogleg Method

The first line segment runs from the origin to the unconstrained minimizer along the steepest descent direction defined by:

$$d_k^c = -\frac{g_k^T g_k}{g_k^T B_k g_k} g \qquad \text{Eq. (6)}$$

While the second line segment runs from d_k^c to d_k^N and the trajectory for d_k is given by:

$$d_k = \begin{cases} \tau d_k^c & 0 \le \tau \le 1 \\ d_k^c + (\tau - 1)(d_k^N - d_k^c) & 1 \le \tau \le 2 \end{cases} \qquad \text{Eq. (7)}$$

where $d_k^N = -B^{-1} g_k$ and $\tau \in [0, 2]$ is a scalar parameter that satisfies the trust-region bound. Over the years, many Dogleg methods have been adapted and used by different authors such as Powell (1970), Bertolazzi (2011), Oruh and Bamigbola (2013), etc.

Theorem 2.1: The vector d is a global solution of the trust-region problem

$$Min\, m(d) = f(x) + g^T d + \frac{1}{2} d^T B d$$

$$s.t \|d\| \le \Delta \qquad \text{Eq. (8)}$$

If and only if d is feasible and there is a scalar $\tau \ge 0$ such that the following conditions hold:

$$(B + \tau I)d = -g \qquad \text{Eq. (9)}$$

$$\tau(\Delta - \| d \|) = 0 \qquad \text{Eq. (10)}$$

$(B + \tau I)$ is positive semi-definite Eq. (11)

3. The New Dogleg Method

In the traditional dogleg method, the piecewise line connecting the Cauchy point d_k^c and the Newton point d_k^N could leave the trust-region through two points instead of one point. Oruh and Bamigbola (2013) considered these two paths with one path involving d_k^c and d_k^N while the other paths involves three points d_k^c, ψd_k^N and d_k^N. Thus, the curve can be approximated either by two straight line segments or three line segments with the same endpoint. They defined the path by:

$$d_k(t) = \begin{cases} d_k^c + t(d_k^N - d_k^c) & 0 \le \text{t} \le 1, \ \psi = 1 \\ d_k^c + t(\psi d_k^N - d_k^c) & 0 \le \text{t} \le 1, \ \psi < 1 \end{cases} \qquad \text{Eq. (12)}$$

This method due to Oruh and Bamigbola (2013) which is the basis of this work used in solving the trust-region sub-problem is summarized by the following algorithm:

Algorithm 3

At iteration k,

1. Compute $d_k^c = -\dfrac{g_k^T g_k}{g_k^T B_k g_k} g_k$

If $\left\| d_k^c \right\| \ge \Delta_k$, then stop with $d_k = -\dfrac{\Delta_k}{\left\| g_k \right\|} g_k$

Else

2. Compute $d_k^N = -B_k^{-1} g_k$

If $\left\| d_k^N \right\| \le \Delta_k$, then stop with $d_k = d_k^N$

Else

3. Find t^* such that $\left\| d_k^c + t(d_k^N - d_k^c) \right\| = \Delta_k$

Where $t = \dfrac{-a_2 + \sqrt{D}}{a_1}$ and $a_1 = \left\| d_k^C - d_k^N \right\|^2$, $a_2 = d_k^N (d_k^C - d_k^N)$, $\text{c} = \left\| d_k^N \right\|^2 - \Delta_k^2$,

$D = a_2^2 - a_1 c$

Lemma 3.1: Let *m* be the quadratic function defined by

$$Min\, m(d) = g^T d + \frac{1}{2} d^T B d \qquad \text{Eq. (13)}$$

where *B* is any symmetric matrix. Then

(i) *m* attains a minimum if and only if *B* is positive semi-definite and *g* is in the range of *B*;

(ii) *m* has a unique minimizer if and only if *B* is positive definite;

(iii) if *B* is positive semi-definite, then every d satisfying $Bd = -g$ is a global minimizer of *m*.

Proof

(i) Let's assume g is in the range of B then there exist a d such that $Bd = -g$.

For all $x \in$, we have that from (13)

$$m(x+d) = g^T(x+d) + \frac{1}{2}(x+d)^T B(x+d) \qquad \text{Eq. (14)}$$

Thus, expanding the R.H.S of (14) gives

$$= g^T x + g^T d + \frac{1}{2}x^T Bx + \frac{1}{2}(Bd)^T x + \frac{1}{2}(Bd)^T x + \frac{1}{2}d^T Bd$$

$$= g^T x + g^T d + \frac{1}{2}x^T Bx + (Bd)^T x + \frac{1}{2}d^T Bd$$

Since $Bd = -g$, then substituting it into the R.H.S gives

$$= (g^T d + \frac{1}{2}d^T Bd) + \frac{1}{2}x^T Bx - g^T x + g^T x$$

From (13), substituting $m(d) = g^T d + \frac{1}{2}d^T Bd$ we will have

$$m(x+d) = m(d) + \frac{1}{2}x^T Bx$$

But, $x^T Bx \geq 0$ since B is positive semi – definite

This implies that $m(x+d) \geq m(d)$ and since B is positive semi-definite, then d is a minimum of m.

Conversely, let d be a minimizer of m then, taking the first derivative of m w.r.t d gives

$$m'(d) = Bd + g = 0$$

Hence, $Bd = -g$ and $m''(d) = B \geq 0$ thus satisfying the result.

(ii) Let's assume that B is positive definite, then from (i) $m(x+d) = m(d) + \frac{1}{2}x^T Bx$

$x^T Bx > 0$ whenever $x \neq 0$ implies $m(x+d) = m(d) + \textit{some positive number}$.

Hence, $m(x+d) \geq m(d)$ and d is a unique minimizer of m.

Conversely, let d be a unique minimizer of m then, taking the first derivative of m w.r.t d gives

$$m'(d) = Bd + g = 0$$

Hence, $Bd = -g$ and $m''(d) = B \geq 0$ thus B is positive semi-definite.

Now, if B is not positive definite, then there exist a vector $x \neq 0$ such that $Bx = 0$. Hence, $m(x+d) = m(d) + \frac{1}{2}x^T Bx$ becomes $m(x+d) = m(d)$ which is a contradiction that m has a unique minimizer.

(iii) Let B be positive semi-definite, then $m'(d) = Bd + g = 0$ implies that $Bd = -g$ and d is a minimizer of m.

4. Updating Techniques for B_k

There are various methods of updating B_k which is an approximation to the Hessian matrix of the objective function. Our choice of B_k is to avoid computing the second derivative of the function which may be costly and difficult to compute for some multivariate functions. The procedure will begin with $B_0 = identity\,matrix$ and proceed with an updating technique to find the next iteration matrix. In this paper, we adopted the Broyden class updating technique which is a linear combination of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) and (Davidon-Fletcher-Powell) DFP methods. The Broyden class is given by:

$$B_{k+1} = B_k - \frac{B_k d_k d_k^T B_k}{d_k^T B_k d_k} + \frac{y_k y_k^T}{y_k^T d_k} + Q_k \left(d_k^T B_k d_k v_k v_k^T \right) \qquad \text{Eq. (15)}$$

$$\text{With } v_k = \left[\frac{y_k}{y_k^T d_k} - \frac{B_k d_k}{d_k^T B_k d_k} \right]$$

Where Q_k is a scalar parameter and $y_k = \nabla f_{k+1} - \nabla f_k$. A method that chooses $0 \le Q_k \le 1$ is called the Restricted Broyden Class. It is important to note that setting $Q_k = 0$ reduces the Broyden Class to BFGS method and setting $Q_k = 1$ reduces the Broyden Class to DFP method. Many people have used the Broyden class recently in different ways and for further understanding and study see [17], [18], [19].

Specifically for this work, we shall choose the values of the scalar parameters as 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.

5. Numerical Examples

We tested our algorithm on a set of test problems from CUTE collections established by (Bongartz et al., 2003). In running the programme, we used these general values for the scalar parameter for our problems;

$\gamma_1 = 0.5, \quad \gamma_2 = 0.5, \quad \eta_1 = 0.01, \quad \eta_2 = 0.75, \quad B_0 = I, \quad Q_k = [0,1]$

and $x*$ represents the critical points for each problem.

Problem 1: Rosenbrock function

$$f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2, \qquad x_0 = [2,1], \qquad \Delta_0 = 0.1 \;\; x* = (1,1)$$

Problem 2

$$f(x_1, x_2, x_3) = 100(x_2 - x_1)^2 + (1 - x_1)^2 + 100(x_3 - x_2^2)^2 + (1 - x_2)^2, \qquad x_0 = [0,\ 1,\ 1.7], \qquad \Delta_0 = 0.5$$

$x* = (1,1,1)$

Problem 3: Booth function

$$f(x_1, x_2) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2 \qquad x_0 = [0,0], \qquad \Delta_0 = 1 \qquad x* = (1,3)$$

Problem 4: Wood function

$$f\left(x_1, x_2, x_3, x_4\right) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + \left(1 - x_3\right)^2 + 10.1\left[\left(x_2 - 1\right)^2 + \left(x_4 - 1\right)^2\right] + 19.8\left(x_2 - 1\right)\left(x_4 - 1\right),$$

$x_0 = [0, 0, 0, 0]$, $\Delta_0 = 0.1$

$x* = (1,1,1,1)$

The stopping criterion for our problem will be $\|g_k\| \leq 10^{-6}$

6. Results

The numerical results obtained from solving our various problems are presented in a table format showing the number of iterations involved (i), the values of our decision variables at the critical points $(x_1,\ x_2)$ and the values of the scalar parameter Q_k used at each point. In the results presented below, we will see that when compared with the critical points of each problem given in Oruh and Bamigbola (2013) that some converged while others could not converge at the different values of the scalar parameter Q_k

Table 6.1: Summary of results obtained from solving Rosenbrock Function at different Q_k

Q_k	I	x_1	x_2
0	19	0.999999494515	0.999999019425
0.1	21	1.000000076322	1.000000170603
0.2	50	2.070602467281	4.293734297812
0.3	50	1.487066833880	2.214563456547
0.4	25	1.000000091521	1.000000171564
0.5	24	1.000000873852	1.000001704331
0.6	24	1.000000794970	1.000001551289
0.7	28	1.000001187506	1.000002448714
0.8	50	2.142690810877	4.592403260206
0.9	50	1.879982162648	3.540114202658
1.0	50	2.290668227116	4.907678259148

Table 6.2: Summary of results obtained from solving Problem 2 at different Q_k

Q_k	i	x_1	x_2	x_3
0	50	1.589281	1.592435	2.540120
0.1	16	1.000000	1.000000	1.000000
0.2	50	2.278162	2.287717	5.244737
0.3	50	1.822148	1.830219	3.355829
0.4	50	1.819554	1.825733	3.334510
0.5	50	0.964804	0.964614	0.930200
0.6	50	1.045938	1.046272	1.094998
0.7	50	1.783509	1.790909	3.206966
0.8	50	1.083339	1.084152	1.175561
0.9	50	1.083357	1.083988	1.175471
1.0	50	1.372763	1.375324	1.893652

Table 6.3: Summary of results obtained from solving Problem 3 at different Q_k

Q_k	I	x_1	x_2
0	9	0.999997	3.000004
0.1	9	0.999997	3.000005
0.2	9	0.999996	3.000006
0.3	9	0.999995	3.000007
0.4	9	0.999994	3.000009
0.5	10	1.000000	3.000000
0.6	10	1.000000	3.000000
0.7	10	1.000000	3.000000
0.8	10	1.000000	3.000000
0.9	10	1.000000	3.000000
1.0	10	1.000000	3.000000

Table 6.4: Summary of results obtained from solving Problem 4 at different Q_k

Q_k	I	x_1	x_2	x_3	x_4
0	300	0.874060	0.769341	1.000147	1.005430
0.1	44	0.999630	0.995903	1.000574	1.004196
0.2	47	0.999668	0.995510	1.000543	1.004584
0.3	47	0.999695	0.995360	1.000540	1.004722
0.4	47	0.999698	0.995283	1.000534	1.004800
0.5	62	0.954614	0.914189	0.971994	0.947966
0.6	Failed results due to closeness of the Hessian matrices to singularity				
0.7	146	0.992141	0.984630	0.999868	1.000060
0.8	269	0.923218	0.848180	0.999868	1.000301
0.9	139	0.990720	0.981828	1.000998	1.002359
1.0	122	0.996371	0.992866	1.000517	1.001172

Table 6.5: Comparison of our results and other known Dogleg methods

Problem	**New Dogleg by Duru and Oruh**		**New Dogleg by Oruh & Bamigbola**		**Standard Dogleg Method**		**Exact Solution**
	No. of Itn	x^*	**No. of Itn**	x^*	**No. of Itn**	x^*	

1	19	$\begin{pmatrix}1.000000\\1.000000\end{pmatrix}$	22	$\begin{pmatrix}1.00000\\1.00000\end{pmatrix}$	85	$\begin{pmatrix}1.00000\\1.00000\end{pmatrix}$	$\begin{pmatrix}1\\1\end{pmatrix}$
2	16	$\begin{pmatrix}1.000000\\1.000000\\1.000000\end{pmatrix}$	69	$\begin{pmatrix}0.999697\\0.999997\\0.999993\end{pmatrix}$	434	$\begin{pmatrix}0.999619\\0.999630\\0.999253\end{pmatrix}$	$\begin{pmatrix}1\\1\\1\end{pmatrix}$
3	10	$\begin{pmatrix}1.000000\\3.000000\end{pmatrix}$	Not available		Not available		$\begin{pmatrix}1\\3\end{pmatrix}$
4	44	$\begin{pmatrix}0.999630\\0.995903\\1.000574\\1.004196\end{pmatrix}$	36	$\begin{pmatrix}1.000000\\1.000000\\1.000000\\1.000000\end{pmatrix}$	Not available		$\begin{pmatrix}1\\1\\1\\1\end{pmatrix}$

7. Conclusions

Examining the effects of the Broyden class parameter on the convergence of the solutions of the problems considered in this work, we discovered that the value of $Q_k = 0.1$ performed relatively better than other values within the interval $[0, 1]$ considered unlike in many literatures where the value $Q_k = 0$ which corresponds to the (BFGS) method is said to be the best. However, we recommend that this method should be tested on more complex and higher dimensional problems to ascertain the consistency of the method over any range in $\Re^n$.

References

Al-Towaiq, M. and Abu-Hour, Y. S. (2017). Two Improved Classes of Broyden's Methods for Solving Nonlinear Systems of Equations. *Journal of Mathematics and Computer Science,* 17, 22-31.

Bertolazzi, E. (2011). Trust-Region Methods. Lectures for Ph.D Course on Numerical Optimization, 1-36.

Burdakov, O., Gong, L., Zikrin, S. and Yuan, Y. (2017). On Efficiently Combining Limited-Memory and Trust-region Techniques, *Journal of Mathematical Programming Computation.* 9(1).

Conn, A. R., Gould, N. I. M. and Toint, P. L. (1987). Trust-Region Methods. *SIAM Journal of Applied Mathematics*. Philadelphia, 980.

Dai, Y.H. and Xu, D.C. (2003). A new family of trust region algorithm for unconstrained optimization, *Journal of Computational Mathematics,* 221-228.

Deguchy, O., Erway, J. B. and Marcia, R. F. (2017). Compact Representation of the Full Broyden Class of Quasi-Newton Updates. Technical report 2017, Wake Forest University, 1-28.

Dennis, J.E. (Jr.), El-Alem, M. and Maciel, M.C. (1997). A global convergence theory for general trust-region based algorithms for equality constrained optimization, *SIAM Journal on Optimization*, 7(1), 177-207.

Dennis, J. E. (Jr.) and Schnabel, R. B. (1983). Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Prentice-Hall, Inc. Englewood Cliffs, New Jersey.

Gertz, M. (2004). A Quasi-Newton Trust-region Methods. *Journal of Mathematical Programming*, 100(3), 447-470.

Huang, W., Gallivan, K. A. and Absil, P. A. (2015). A Broyden Class of Quasi-Newton Methods for Riemannian Optimization, Technical Report 15-04, Mathematics Department, Florida State University.

Li, C., Zhou, Q. and Wu, X. (2015). A Non-Monotone Trust-region Method with Non-Monotone Wolfe-Type Line Search Strategy for Unconstrained Optimization. *Journal of Applied Mathematics and Physics,* 3(6), 300-306.

Nocedal, J. and Wright, S. J. (1999). *Numerical Optimization*, Springer-Verlag, New York.

Oruh, B. I. and Bamigbola, O. M. (2013). A New Dogleg Method for Solving the Trust-region Sub-problem*.* *IOSR Journal of Mathematics*, 8(2), 41-48.

Powell, M. J. D. (1970). A New Algorithm For Unconstrained Optimization. In: Nonlinear Programming. J. B. Rosen, O. L. Mangasarian and K. Ritter, (Eds). Academic Press, New York, 2, 1-27.

Qu, S. and Jiang, S. (2008). A Trust-region Method with a Conic Model for Unconstrained Optimization. *Mathematical Methods in the Applied Sciences,* 31(15), 1780-1808.

Shultz G.A., Schnabel R.B. and Byrd, R.H. (1985). A family of trust-region-based algorithms for unconstrained minimization with strong global convergence, *SIAM Journal of Numerical Analysis*. 22, 47-67.

Toint, Ph.L. (1986). Global convergence of a class of trust region methods for nonconvex minimization in Hilbert space, *IMA Journal of Numerical Analysis*, 8, 231-252.

Tong, X. and Zhou, S. (2006). Combining Trust-region and Line Search Methods for Equality Contrained Optimization. *Journal of Numerical Functional Analysis and Optimization,* 143-162.

Yuan, G., Meng, S. and Wei, Z. (2009). A Trust-region-based BFGS Method with a Line Search Technique for Symmetric Non-linear Equations. In: Advances in Operations Research.