Menemui Matematik (Discovering Mathematics)
Vol. 34, No. 1: 1– 6 (2011)

Efficient BFGS Algorithm for Solving Unconstrained Optimization (Algoritma BFGS yang Berkesan Bagi Menyelesaikan Pengoptimuman Tak Berkekangan)

Azfi Zaidi Mohammad Sofi[1] ,Mohd Asrul Hery Ibrahim[2] & Mustafa Mamat[3]

[1]*Department of Skills and Core Subjects, Faculty of Science and Information Technology, Kolej Universiti Islam Antarabangsa Selangor (KUIS), Malaysia.*

[2]*School of Applied Science and Foundation, Kuala Lumpur Infrastructure University College (KLIUC), Malaysia*

[3]*Department of Mathematics, Faculty of Science and Technology, Universiti Malaysia Terengganu (UMT), Malaysia*

[1]*azfizaidi@kuis.edu.my*, [2]*asrulhery@kliuc.edu.my*, [3] *mus@umt.edu.my*

ABSTRACT

Quasi-Newton method is one of the most efficient and well known method for solving unconstrained optimization problems. In Quasi-Newton method, BFGS update is the finest Hessian update to work with. In this paper, an alternative algorithm for the BFGS update is proposed by changing the condition for the step size selection and we conclude the result analysis at the end of this paper by the number of the iteration and by the number of the function evaluation. Proven here that our proposed BFGS algorithm is better.

Keywords: BFGS, Armijo line search, unconstrained optimization, step size, superlinear

INTRODUCTION

Consider the unconstrained optimization problem

$$\min_{x \in R^n} f(x) \tag{1}$$

where f is a twice continuously differentiable function from R^n to R The update BFGS formula is the iteration method whereby at the (k+1) th iteration, x_{k+1} is given by

$$x_{k+1} = x_k + \alpha_k d_k \tag{2}$$

where α_k denote the step size (Cauchy,1847; Curry 1944)

$$\alpha_k = \arg \min_{\alpha > 0} f(x + \alpha d_k) \tag{3}$$

and d_k denotes search direction

$$d_k = -B_k^{-1} g_k \tag{4}$$

and g_k is the gradient of f B_k in (4) is an update Hessian approximation formula defined by

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k}, \tag{5}$$

with $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$ and this Hessian approximation update formula must satisfy the equation $B_{k+1} s_k = y_k$.

In the Section 2 of this article, we discuss the determination of step size and followed by the details explanation about the hybrid BFGS search direction with the steepest descent search direction in the Section 3. Numerical results of the methods and their explanation will be shown in the Section 4 and we conclude the whole article in the Section 5.

STEP SIZE

In order to gain the superlinear convergence in Quasi-Newton method, the determination of the step size is very important. Most past researchers used only single formula to determine the step size such as Cauchy (1847), Han & Newmann (2003), Mustafa *et al*. (2004) and many more. The combination of step size using two formulas was started by Yuan (2006) and followed by Sofi *et al*. (2008[i] & 2008[ii]). Yuan (2006) had proposed the combination of the exact line search α_k (Cauchy, 1847) within the formula γ_k denoted by

$$\lambda_k := \frac{2}{\sqrt{\left(\frac{1}{\alpha_{k-1}^*} - \frac{1}{\alpha_k^*}\right)^2 + \frac{4\| g_k \|_2^2}{\| s_{k-1} \|_2^2} + \frac{1}{\alpha_{k-1}^*} + \frac{1}{\alpha_k^*}}} \tag{6}$$

Yuan (2006) used both of exact line search formula α_k at the even iteration and γ_k at the odd iteration in the classic steepest descent method. Iteration process (7) below show the process and details can be found in Yuan (2006).

$$\begin{aligned} x_2 &= x_1 + \alpha_1^* d_1 \\ x_3 &= x_2 + \lambda_2 d_2 \\ x_4 &= x_3 + \alpha_3^* d_3 \\ &M \\ x_{k+1}^* &= x_k + \alpha_k^* d_k \end{aligned} \tag{7}$$

From this step size combination concept in Yuan (2006), Sofi *et al.* (2008[i]) had applied this step size combination concept in the Quasi-Newton method with some modifications and details can be found in Sofi *et al.* (2008[i]).

In this research, we choose one of the famous update Hessian approximation formula in the Quasi-Newton method and we combine the step size formula between the exact line search formula and Armijo (1966) step size formula with some modifications to the algorithm. We applied the modification made by Sofi *et al.* (2008) into this algorithm. Sofi *et al.* (2008) had found when ($\alpha_k < 0.1$)), the descent of the function is not great and state that we must used another formula of step size instead of the exact line search.

Armijo line search can be summarize as followed.

Given $s < 0, \beta \in (0,1), \sigma \in (0,1)$ and $\lambda_k = \max\{s, s\beta, s\beta^2, \ldots\}$ when

$$f(x) - f(x_k + \lambda_k d_k) \geq -\sigma \lambda_k g_k^T d_k \tag{8}$$

where k = 0,1,2,…. However, in the proposed algorithm, we only use Armijo line search when k=2. More details explanation about Armijo line search can be seen in Armijo (1966) and Shi (2006).

ALGORITHM

Here, we shown the Algorithm 1 and Algorithm 2. Both of the algorithms shown the standard 'general BFGS' and 'conditional general BFGS' consequently. Algorithm 2 is the algorithm that we proposed.

Algoritma1: General BFGS

Data $f \in C^2$, $x \in D$,

1. $k := 1$
2. $x_k := x$
3. $g_k := \nabla f(x_k)$
4. $B_k := I_n$
5. $d_k := -B_k^{-1} g_k$
6. Calculate α_k^* based on (4)
7. $x_{k+1} := x_k + \alpha_k^* d_k$
8. If $\| g_{k+1} \| \leq \varepsilon$, then stop
9. $s_k := \alpha_k^* d_k$
10. $y_k := g_{k+1} - g_k$
11. Calculate B_{k+1}^{-1} based on (5)
12. $k := k+1$

Algoritma 2: Conditional General BFGS

Data $f \in C^2$, $x \in D$,

1. $k := 1$
2. $x_k := x$
3. $g_k := \nabla f(x_k)$
4. $B_k := I_n$
5. $d_k := -B_k^{-1} g_k$
6. Calculate α_k^* based on (4)
7. $x_{k+1} := x_k + \alpha_k^* d_k$
8. If $\| g_{k+1} \| \leq \varepsilon$, then stop
9. Calculate B_{k+1}^{-1} based on (5)
10. $k := k+1$
11. For k >2 and if and only if $\alpha_{k-1}^* < 0.1$ replace α_k^* with γ_k
    11.1 γ_k calculated by (8)
    11.2. $x_{k+1} := x_k + \lambda_k d_k$

The Algorithm 3.1 use the exact line line search as the step size. However, the Algorithm 3.2 use the combination of exact line search and Armijo line search with some conditional implementation to the algorithm as we can see at the step 11th in the Algorithm 3.2. The efficiency of all algorithms are tested using some standard problems and further discussion can be found details in the next section.

NUMERICAL RESULTS

In this section, all numerical results for all algorithms are shown. Seven standard unconstrained optimization testing problems are tested on the Algorithm 3.1 and Algorithm 3.2. Listed below are the testing problems with their initial points and minimizer points.

Problem 1. (Cube Function, n=2)

$$f(x_1,x_2) = 100(x_2 - x_1^3)^2 + (1 - x_1)^2.$$

Initial point: (-1.2, 1.6)
Minimizer: (1,1)

Problem 2. (Rosenbrock Function, n=2)

$$f(x_1,x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

Initial point: (x_1=1.6, x_2=-2.5)
Minimizer: (1,1)

Problem 3. (Rosenbrock Function, n=4)

$$f(x_1,x_2,x_3,x_4) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 100(x_4 - x_3^2)^2 + (1 - x_3)^2.$$

Initial point: (-32, -3.3, -3.2, -3.3)
Minimizer:

Problem 4. (Shalow Function, n=2)

$$f(x_1,x_2) = (x_2 - x_1^2)^2 + (1 - x_1)^2.$$

Initial point: (50, 100)
Minimizer: (1,1)

Problem 5. (Shalow Function, n=4)

$$f(x_1,x_2,x_3,x_4) = (x_2 - x_1^2)^2 + (1 - x_1)^2 + (x_4 - x_3^2)^2 + (1 - x_3)^2.$$

Initial point: $(x_1 = 20.,x_2 = 40.,x_3 = 20.,x_4 = 40.)$
Minimizer: (1,1,1,1)

Problem 6. (Strait Function, n=2)

$$f(x_1,x_2) = (x_2 - x_1^2)^2 + 100(1 - x_1)^2.$$

Initial point: (30., 50)
Minimizer: (1,1)

***Problem** 7.* (Strait Function, n=4)

$$f(x_1,x_2,x_3,x_4) = \left(x_2 - x_1^2\right)^2 + 100(1-x_1)^2 + \left(x_4 - x_3^2\right)^2 + 100(1-x_3)^2.$$

Initial point: (30., 50., 30., 50.)
Minimizer: (1,1,1,1)

Table 1: Numerical Results for Algorithm 3.1 and Algorithm 3.2

Tested Problems	Algorithm 3.1		Algorithm 3.2	
	Number of iteration,	Number of function evaluation,	Number of iteration,	Number of function evaluation,
1	17	103	**10**	**56**
2	19	77	**17**	**63**
3	23	93	**21**	**72**
4	17	69	17	69
5	14	57	14	57
6	9	37	9	37
7	10	41	10	41

Table 1 shows the numbers of iteration and the numbers of function evaluation n_f for Algorithm 3.1 and Algorithm 3.2 respectively with seven optimization problems tested on both of them.

The bold numbers in Table 1 show the least number of iteration n_i and the least number of the function evaluation n_f. We can see from Table 1 that Algorithm 3.2 are better for solving unconstrained optimization problem compared to Algorithm 3.1. The proposed algorithm works on the tested problem 1, 2 and 3.

CONCLUSION

In this research, we proposed alternative algorithms where the modification of the step size bring the good impact to the BFGS update. Results show that alternative algorithms (Algorithm 3.2) that had been proposed are efficient for solving the unconstrained optimization problems tested. We can look clearly if we compare the results in Table 1, our proposed method (Algorithm 3.2) are are more effective compared to the standard algorithm (Algorithm 3.1) for the several problems tested. For that reason, we believe that there many more methods after this will be conducted and constructed in order to find the best solution using the combination of the step size. It is proven helps the origin condition of the standard BFGS update. This proposed conditional method based on BFGS update class also work on Broyden update class and we can see the effectiveness of the method in Ibrahim *et al.* (2010).

REFERENCES

Armijo, L. (1966). Minimization functions having Lipshitz continuous first partial derivatives, *PacificJ. Mathematics* **16**:1-3.

Cauchy, A.(1847). Méthodes générales pour la résolution des systèmes d'équations simultanées, *C. R. Acad. Sci. Paris* **25**:536-538.

Curry, H. B. (1944). The method of steepest descent for nonlinear minimization problems, *Quart. Appl. Math.,* **2**: 258-261.

Han, L. & Newman, M. (2003). Combining quasi-Newton and Cauchy directions, *Int. Journal of Applied Mathematics* **12(2)**:167-171.

Ibrahim, M.A.H., Mamat, M., Sofi, A.Z.M., Mohd, I. & Ahmad, W.M.A.W. (2010). Alternative Algorithms of Broyden FAMILY^{AMI} for Unconstrained Optimization: *ICMS International Conference on Mathematical Science, AIP Conference Proceedings,* **1309**: 670-680.

Mustafa Mamat, Yosza Dasril & Ismail Mohd (2004). Kaedah quasi-Newton untuk pengoptimuman tak berkekangan, *Prosiding Simposium Kebangsaan Sains Matematik ke 12, UIA, Gombak Selangor*: 23-25.

Shi, Z.J. (2006). Convergence of quasi-Newton method with new inexact line search, *J. Math. Anal. Appl.* **315**: 120-131.

Sofi, A.Z.M., Mamat, M., Mohd, I. & Dasril, Y. (2008[i]). An alternative hybrid search direction for unconstrained optimization, *Journal of Interdisciplinary Mathematics*, **11(5)**: 731-739.

Sofi, A.Z.M., Mamat, M., Mohd, I. & Dasril, Y. (2008[ii]). An efficient step size for BFGS, *Journal of Ultra Scientist of Physical Sciences*, **20(3)**.

Yuan, Y.X. (2006). A new stepsize for the stepest descent method, *Journal of Computational Mathematics*, **24(2)**: 149-156.