MJFAS MALAYSIAN JOURNAL OF FUNDAMENTAL AND APPLIED SCIENCES
PRINT ISSN: 2289-5981 | ONLINE ISSN: 2289-599X

RESEARCH ARTICLE

A combination of Broyden-Fletcher-Goldfarb-Shanno (BFGS) and *n*-th section method for solving small-scale unconstrained optimization

Atikah Ramli [a*], Ibrahim Jusoh [b], Mohd Rivaie Mohd Ali [c]

[a] Deparment of Mathematics,Faculty of Computer and Mathematical Sciences,Universiti Teknologi Mara Shah Alam,40450 Shah Alam, Selangor,Malaysia
[b,c] Deparment of Mathematics,Faculty of Computer and Mathematical Sciences,Universiti Teknologi Mara Terengganu,21300 Kuala Terengganu, Terengganu,Malaysia

* Corresponding author: atikahramli050593@gmail.com

Article history
Received 16 February 2017
Accepted 4 October 2017

Abstract

In this research, a new inexact line search method known as *n*-th section method is used to obtain the step size in BFGS method. The *n*-th section method is the modification of the original bisection method. As in bisection method, this simple *n*-th section method divides each interval section with an even number of interval which is greater than two. This new proposed algorithm is compared with the original bisection, newton and secant method in terms of number of iteration. Numerical results is obtained based on small scale functions .This research shows that the algorithm is more efficient than using the ordinary line search methods. Besides, this proposed algorithm also possessed global convergence properties.

Keywords: BFGS method, *n*-th section method, step size, global convergence

INTRODUCTION

An unconstrained optimization problem is defined as

$$\min f(x)\,,\quad x \in R^n,$$

where $f: R^n \to R$ is a continuously differentiable function.The function f is called the objective function and R^n is an n-dimensional Euclidean space. An iterative method is used to solve the problem which is

$$x_{k+1} = x_k + \alpha_k d_k\,, \qquad k = 0, 1, 2\ldots \tag{1}$$

where x_k is the approximate value of the current iteration and $\alpha_k > 0$ is step size obtained by line search. In Quasi-Newton method ,the search direction d_k is given by

$$d_k = -H_k g_k \qquad k = 0,1,2,\ldots \tag{2}$$

where H_k is the approximation of Hessian matrix of the objective function and g_k is gradient. The BFGS method equations for approximate Hessian matrix is given by

$$H_{k+1}^{BFGS} = H_k + \left(1 + \frac{y_k^T H_k y_k}{s_k^T y_k}\right)\frac{s_k s_k^T}{s_k^T y_k} - \frac{s_k^T y_k H_k + H_k y_k s_k^T}{s_k^T y_k}\,, \tag{3}$$

with $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$. In order to guarantee a decrease in objective function for small α, we have to ensure that $g_k^T H_k g_k > 0$. To fulfil this condition, H_k is required to be positive definite *nxn* matrix.

Line Search Method

Some researchers proposed several line search methods to calculate the step size,α_k. It is important to find good step size α_k that will provide the best reduction of the objective function along the search direction. These methods can be grouped into two categories which are exact and inexact line search. Exact line search gives the greatest possible reduction to the objective function along the search direction (Goh et al., 2012). The value of α_k is chosen such that

$$\alpha_k = argmin\,\{f(x_k + \alpha\, d_k); \alpha > 0\}. \tag{4}$$

However, in practical calculation, it is very expensive and at most times impossible to find the exact step size. Thus, it is preferable to use inexact line search at lesser cost compared to exact line search. (Adeleke et al., 2013). Previous researchers have presented some inexact line search method such as Goldstein Condition (Goldstein, 1965), Armijo line search (Armijo, 1966) and Wolfe (1969). However, these line search are complicated and considered costly because they need high processor of CPU.

Hence, this research focused on the improvement of BFGS method by applying the *n*-th section method as a line search which is a modification from classical Bisection method known as fourth and sixth section method that will lead to faster convergence of the root. Other than that, newton method and secant method are also applied in BFGS method as a comparison. The efficiency of the proposed method are analysed based on number of iterations and CPU times.

METHODOLOGY

N-th section line search method

Bisection line search is simple line search method which is one of the first numerical methods that have been developed to find the root of a nonlinear equation, $f(x) = 0$. Single interval in bisection method is divided only into two sections and it leads to slow convergence. Alternatively, newton and secant methods are methods that converge faster than bisection method in term of number of iteration. However, in term of CPU time, these methods possess higher CPU time compared to bisection method. These methods have the disadvantage of being computationally expensive (Ding,2011).

Other than that, the newton and secant method is known not to converge when the initial guess of the root is far away from the exact root. Hence, a new inexact line search called as *n*-th section method is proposed (Nujma, 2015). This method used a higher division number such as four or six division that is applied for single interval. Theoretically, this will lead to faster convergence of the root if it is applied as line search in the BFGS method. Hopefully, this method will provide faster calculation for the root. Figure 1,2 and 3 show the schematic representation of Bisection and *n*-th section line search method.

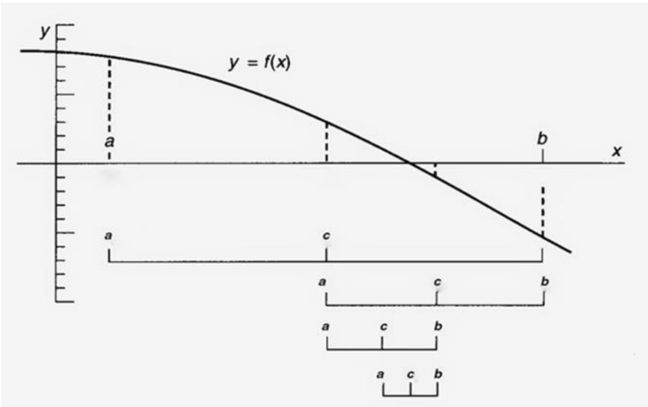

Fig. 1 The schematic representation of bisection method.

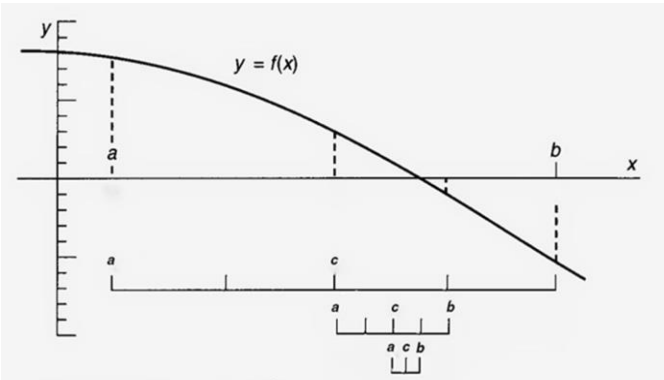

Fig. 2 The schematic representation of fourth section method.

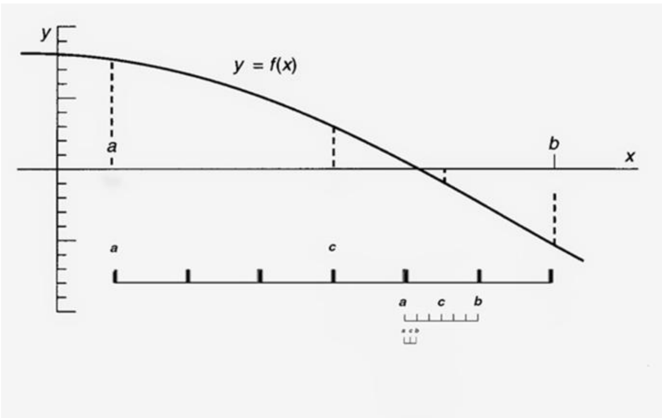

Fig. 3 The schematic representation of sixth section method.

The BFGS method

In this paper, the proposed method involves only the classical BFGS method which is the simplest version of BFGS method where its complexity of algorithm is relatively low when compared to the other version. In addition, small scale problems will be used in this research. As suggested by Andrei (2008), the stopping criteria that is appropriate for both large and small scale problems is $\|g_k\| \leq \varepsilon$ where $\varepsilon = 10^{-6}$ which applied in this research.The efficiency of this proposed method will be analyzed based on number of iterations. The standard test problems are tested using Maple16 software because the programming code is simple and easy to be applied. The CPU processor used was Corei5 processor 3337U with 4GB. Table 1,2,3,4 and 5 shows the algorithms that has been conducted in this research.

Table 1 Algorithm of BFGS Method.

Step	Explanation
1	Given an initial point $x_0 \in R^n$, initial positive symmetric matrix $H_0 = R^n$, $\varepsilon \in R$ with the iteration number $k = 0,1,2 \ldots$
2	Computing search direction. $d_k = -H_k g_k$ as in (2) . If $g_k = 0$, then stop.
3	Computing step size, α_k with an employment of inexact line search.
4	Updating new point. $x_{k+1} = x_k + \alpha_k d_k$.
5	Updating Hessian matrix $H_{k+1}(x)$ as in eqt (3).
6	Convergent test and stopping criteria.If $f_{(x+1)} < f_{(x)}$ and $\|g_k\| \leq \varepsilon$,then stop. Otherwise go to Step 1 with $k = k + 1$.

Table 2 Algorithm of BFGS method applying Bisection method.

Step	Explanation
1	Given an initial point $x_0 \in R^n$, initial positive symmetric matrix $H_0 = R^n$, $\varepsilon \in R$ with the iteration number $k = 0,1,2 \ldots$
2	Computing search direction. $d_k = -H_k g_k$ as in (2). If $g_k = 0$, then stop.
3	Computing step size, α_k with an employment of bisection line search method : Step i: Identify two numbers *a* and *b* as an interval at which *f '(a)* has different signs. Step ii: Define sections, $m = \frac{a+b}{n}$ where *n* = 2 Step iii: Determine if i. *f '(a) . f '(a+m)*< 0 ,then *r* ϵ (*a, a+m*) ii. *f '(a+m) . f '(a+2m)*< 0 ,then *r* ϵ (*a+m,a+2m*) iii. *f '(a+2m) . f '(a+3m)*< 0 ,then *r* ϵ (*a+2m,a+3m*) . . . *f'(a+im) . f '(a+(i+1)m)*< 0 ,then *r* ϵ (*a+im, a+(i+1)m*) where $i = 0,1,2,3 \ldots$ Step iv : Define c_{mid} as midpoint, $c_{mid} = \frac{(a+im)+(a+(i+1)m)}{2}$ and $c_{mid} = \alpha_k$. Step v: Repeated until desired iteration/ accuracy until α_k has been determined. Then, continue to Step 4.
4	Updating new point after computing α_k, $x_{k+1} = x_k + \alpha_k d_k$.
5	Updating Hessian matrix $H_{k+1}(x)$ as in eqt (3).
6	Convergent test and stopping criteria.If $f_{(x+1)} < f_{(x)}$ and $\|g_k\| \leq \varepsilon$,then stop. Otherwise go to Step 1 with $k = k + 1$.

Table 3 Algorithm of BFGS method applying the *n*-th section method.

Step	Explanation
1	Given an initial point $x_0 \in R^n$, initial positive symmetric matrix $H_0 = R^n$, $\varepsilon \in R$ with the iteration number $k = 0,1,2 \ldots$
2	Computing search direction. $d_k = -H_k g_k$ as in (2). If $g_k = 0$, then stop.
3	Computing step size, α_k with an employment of *n*-th section line search method : Step i: Identify two numbers a and b as an interval at which *f '(a)* has different signs. Step ii: Define sections, $m = \frac{a+b}{n}$ where *n* = 4 and 6 Step iii: Determine if a. *f '(a) . f '(a+m)*< 0, then *r* ϵ (*a, a+m*) b. *f '(a+m) . f '(a+2m)*< 0 ,then *r* ϵ (*a+m,a+2m*) c. *f '(a+2m) . f '(a+3m)*< 0 ,then *r* ϵ (*a+2m,a+3m*) . . . *f'(a+im) . f '(a+(i+1)m)*< 0 ,then *r* ϵ (*a+im, a+(i+1)m*) where $i = 0,1,2,3 \ldots$ Step iv : Define c_{mid} as midpoint, $c_{mid} = \frac{(a+im)+(a+(i+1)m)}{2}$ and $c_{mid} = a_k$. Step v: Repeated until desired iteration/ accuracy until α_k has been determined. Then, continue to Step 4.
4	Updating new point after computing α_k, $x_{k+1} = x_k + \alpha_k d_k$.
5	Updating Hessian matrix $H_{k+1}(x)$ as in eqt (3).
6	Convergent test and stopping criteria.If $f_{(x+1)} < f_{(x)}$ and $\lVert g_k \rVert \leq \varepsilon$,then stop .Otherwise go to Step 1 with $k = k + 1$.

Table 4 Algorithm of BFGS method applying the Newton method.

Step	Explanation
1	Given an initial point $x_0 \in R^n$, initial positive symmetric matrix $H_0 = R^n$, $\varepsilon \in R$ with the iteration number $k = 0,1,2 \ldots$
2	Computing search direction. $d_k = -H_k g_k$ as in (2). If $g_k = 0$, then stop.
3	Computing step size, α_k with an employment of newton line search. Step i : Identify the function $f(a)$. Step ii : Compute the derivate $f'(a)$. Step iii : Identify the initial value of a_0. Step iv : Evaluate $f(a)$and $f'(a)$ at a_0. Step v : Apply Newton's formula for the next root estimate $a_{n+1} = a_n - \frac{f(a_n)}{f'(a_n)}$ Step vi : Repeated until desired iteration/ accuracy until α_k has been determined. Then, continue to Step 4.
4	Updating new point after computing α_k, $x_{k+1} = x_k + \alpha_k d_k$.
5	Updating Hessian matrix $H_{k+1}(x)$ as in eqt (3).
6	Convergent test and stopping criteria. If $f_{(x+1)} < f_{(x)}$ and $\lVert g_k \rVert \leq \varepsilon$,then stop .Otherwise go to Step 1 with $k = k + 1$.

Table 5 Algorithm of BFGS mnethod applying the Secant method.

Step	Explanation
1	Given an initial point $x_0 \in R^n$, initial positive symmetric matrix $H_0 = R^n$, $\varepsilon \in R$ with the iteration number $k = 0,1,2 \ldots$
2	Computing search direction. $d_k = -H_k g_k$ as in (2). If $g_k = 0$, then stop.
3	Computing step size, α_k with an employment of secant line search method. Step i : Identify the function $f'(a)$. Step ii : Identify the two initial values of a_0 and a_1. Step iii : Evaluate $f(a_0)$ and $f(a_1)$. Step iv : Apply Secant's formula for the next root estimate , $a_{n+1} = a_n - f(a_n)\left[\frac{a_n - a_{n-1}}{f(a_n) - f(a_{n-1})}\right]$ Step v : Repeated until desired iteration/ accuracy until α_k has been determined. Then, continue Step 4.
4	Updating new point after computing α_k, $x_{k+1} = x_k + \alpha_k d_k$.
5	Updating Hessian matrix $H_{k+1}(x)$ as in eqt (3).
6	Convergent test and stopping criteria. If $f_{(x+1)} < f_{(x)}$ and $\lVert g_k \rVert \leq \varepsilon$,then stop .Otherwise go to Step 1 with $k = k + 1$.

RESULTS AND DISCUSSION

In this section, the test problems considered by Andrei (2008) are used to analyze the modification of BFGS method. Each of the test problems is tested with dimensions 2 variables. Table 6 shows the result of number of iterations that has been tested.

Table 6 Result of number of iterations.

Initial point	Bisection	Fourth	Sixth	Newton	Secant
Booth					
1.25,1.25	4	4	3	2	F
5,5	3	3	3	2	F
10,10	3	3	3	2	F
100,100	3	3	3	2	F
Treccani					
1.25,1.25	5	4	7	4	F
5,5	6	11	4	F	F
10,10	6	6	6	F	F
100,100	4	13	13	F	F
Matyas					
1.25,1.25	2	2	2	2	F
5,5	2	2	2	2	F
10,10	2	2	2	2	F
100,100	2	2	2	2	F
Deschnb					
1.25,1.25	5	5	5	5	F
5,5	10	8	7	8	F
10,10	6	6	7	8	F
100,100	9	9	9	7	F

The results show that Bisection, fourth section and sixth section are succesful and achieved the global minimum point of the functions. For Newton method in Treccani function, there are some failure and for Secant method, the testing is totally fail. This is because both the Secant and Newton methods required two points for the start of each iterations. For these two points which are further apart or spread further from the exact root, will lead to the failure of these method which has been discussed earlier in methodology section.

Other than that, the performance results are shown in Figure 4 using the performance profile introduced by Dolan and More (2002). The performance profile seeks to find how well the solvers perform relative to the other solvers on the problems. In this performance profile, they introduced the notion of a means to evaluate and compare the performance of the set solver S on a test set P. Assume that s_n solvers and p_n problems exists, for each problem p and solver s, define that

$t_{p,s}$ = computing time (the number of iterations or CPU time or others) required to solve problems p by solver s .

The performance of solver s on any given problem might be of interest, but because want to obtain an overall assessment of the performance of the solver, then it was defined

$$p_s(t) = \frac{1}{n_p} size\{p \in P : r_{p,s} \leq t\}$$

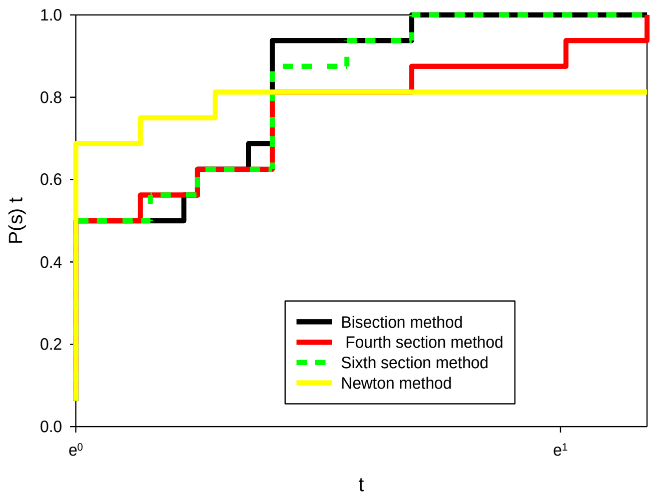

Fig. 4 Performance profile in a log_{10} scale based on number of iterations.

Based on the performance profile, the best method is determined based on the right side of figure which has the highest curve value. In this case, the best line search are Bisection, fourth section and sixth section method with the value 1. These mean that these line search can solve all the given problems succesfully. The fastest method is determined based on the left side which is Newton method. However, Newton method could not solve all the given problems. The Secant method is the worst method which could not solve all the problems.Therefore, it can be concluded that the *n*-th sections method could be used as the line search method.

.

CONCLUSION

In this study, a combination of BFGS and *n*-th section method for solving unconstrained optimization problem is investigated. This method is tested using standard test problems in terms of number of iterations. As a conclusion, it shows that the *n*-th section method is efficient when applied as a line search method if in BFGS method.

ACKNOWLEDGEMENT

This work is supported by Ministry of Higher Education Malaysia and conducted under the fundamental research grant scheme (FRGS) grant number 600-RMI/FRGS 5/3 (64/2015).

REFERENCES

Adeleke O. J., Aderemi O. A., Omoregbe N. I. and Adekunle, R. A. 2013. Numerical comparison of line search criteria in nonlinear conjugate gradient algorithms, *International Journal of Mathematics and Statistics Studies*, *2*, 24.

Andrei, N. 2008. An unconstrained optimization test functions collection. *Advanced Modelling and Optimization*, *10,* 147-161. Retrieved from https://www.researchgate.net/publication/228737339_An_unconstrained_optimization_test_functions_collection.

Armijo, L. 1966. Minimization of functions having Lipshitz continuous first partial derivatives, *Pacific Journal Mathematics*, *16*, 1-3.

Ding, Y., Lushi, E., & Li, Q. 2011. *Investigation of quasi-Newton methods for unconstrained optimization*. Simon Fraser University, Canada.

Dolan, E.D. & More, J.J. (2002).Benchmarking optimization software with performance profile. *Mathematics Programming, 91*, 201-213.

Goh, K.W., Mamat, M., Mohd, I. and Dasril, Y. (2012), A Novel of Step Size Selection Procedures for Steepest Descent Method, *Applied Mathematical Sciences*, 6, 2507 – 2518

Goldstein, A. A. (1965), On steepest descent, *SIAM Journal Control*, *3*, 147-151.

Nujma Hayati (2015). The N-th section method : A Modification of Bisection Method, *The 3rd International Innovation, Design And Articulation (i-Idea 2016)*, Malaysia

Wolfe, P. (1969). Convergence conditions for ascent method. *SIAM Journal Control*, *11*, 226-235.