Sains Malaysiana 43(10)(2014): 1591–1597

BFGS Method: A New Search Direction

(Kaedah BFGS: Arah Carian Baharu)

MOHD. ASRUL HERY BIN IBRAHIM, MUSTAFA MAMAT* & LEONG WAH JUNE

ABSTRACT

In this paper we present a new line search method known as the HBFGS method, which uses the search direction of the conjugate gradient method with the quasi-Newton updates. The Broyden-Fletcher-Goldfarb-Shanno (BFGS) update is used as approximation of the Hessian for the methods. The new algorithm is compared with the BFGS method in terms of iteration counts and CPU-time. Our numerical analysis provides strong evidence that the proposed HBFGS method is more efficient than the ordinary BFGS method. Besides, we also prove that the new algorithm is globally convergent.

Keywords: BFGS method; conjugate gradient method; globally convergent; HBFGS method

ABSTRAK

Dalam kertas ini kami berikan suatu kaedah carian yang baru dikenali sebagai kaedah HBFGS yang menggunakan arah carian kaedah kecerunan konjugat dengan kemaskini kuasi-Newton. Kemaskini Broyden-Flecther-Goldfarb-Shanno (BFGS) digunakan sebagai formula untuk penghampiran kepada Hessian bagi kedua-dua kaedah. Algoritma baru dibandingkan dengan kaedah kuasi-Newton dalam aspek bilangan lelaran dan juga masa CPU. Keputusan berangka menunjukkan bahawa kaedah HBFGS adalah lebih baik jika dibandingkan dengan kaedah BFGS yang asal. Selain itu, kami juga membuktikan bahawa algoritma baru ini adalah bertumpuan secara sejagat.

Kata kunci: Bertumpuan sejagat; kaedah BFGS; kaedah HBFGS; kaedah kecerunan konjugat

INTRODUCTION

Quasi-Newton methods are well-known methods in solving unconstrained optimization problems. These methods, which use the updating formulas for approximation of the Hessian, were introduced by Davidon in 1959, and later popularised by Fletcher and Powell in 1963 to give the Davidon-Fletcher-Powel (DFP) method. But the DFP method is rarely used nowadays. On the other hand, in 1970 Broyden, Fletcher, Goldfarb and Shanno developed the idea of a new updating formula, known as BFGS, which has become widely used and recently the subject of many modifications.

In general, the unconstrained optimization problems are described as follows:

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

where R^n is an n-dimensional Euclidean space and $f : R^n \to R$ is assumed to be continuously twice differentiable. The gradient and Hessian for (1) are denoted as g and G, respectively. In order to display the updated formula of BFGS, the step-vectors s_k and y_k are defined as:

$$\begin{aligned} s_k &\overset{def}{=} x_{k+1} - x_k \\ y_k &\overset{def}{=} g(x_{k+1}) - g(x_k) \\ &= g_{k+1} - g_k. \end{aligned} \tag{2}$$

In this paper, whenever quasi-Newton methods are concerned, we will focus on the BFGS method which has proved to be the most effective of all quasi-Newton methods. Hence, if B_k is denoted as an approximation of Hessian G at x_k, the updating formula for BFGS is,

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k}. \tag{3}$$

It's also well known that the matrix B_{k+1} is generated by (3) to satisfy the secant equation

$$B_{k+1}\, s_k = y_k, \tag{4}$$

which may be regarded as an approximate version of the Newton relation. Note that it is only possible to fulfill the secant equation if

$$s_k^T y_k > 0, \tag{5}$$

which is known as the curvature condition. It is also worth mentioning that having (5) holds would ensure that the BFGS updating matrix (3) is positive definite.

During the last few decades, the convergences of the quasi-Newton methods have received much study. Powell (1976) has proved the global convergence of the BFGS method with the practical Wolfe line search in the case when the f is convex. His results have been extended to the restricted Broyden's family, except the DFP method by Byrd et al. (1987). The question whether the BFGS method and other quasi-Newton methods are globally convergent for general objective functions has been open for several decades until quite recently, when Dai (2002) and Mascarenhas (2004) gave a negative answer by providing counter-examples independently.

Realising the possible non-convergence for general objective functions, some authors have considered modifying quasi-Newton methods to enhance the convergence. For example, Li and Fukushima (2001) modify the BFGS method by skipping the update when certain conditions are not satisfied and prove the global convergence of the resulted BFGS method with a 'cautious update' (which is called the CBFGS method). However, their numerical tests showed that the CBFGS method did not perform better than the ordinary BFGS method. Then, Mustafa et al. (2009) proposed a new search direction for quasi-Newton methods in solving unconstrained optimization problems. Generally, the search direction focused on the hybridization of quasi-Newton methods with the steepest descent method. The search direction proposed by Mustafa et al. (2009) is $d_k = -\eta B_k^{-1} g_k - \delta g_k$, where $\eta > 0$ and $l > 0$. They realised that the hybrid method is more effective compared with the ordinary BFGS in terms of computational cost. Hence, the delicate relationships between the conjugate gradient method and the BFGS method have been explored in the past. Two competing algorithms of this type are the L-BFGS method described by Nocedal (1980) and the variable storage conjugate gradient (VSCG) method published by Buckley and LeNir (1983).

In this paper, motivated by the idea of conjugate gradient methods, we proposed a line search algorithm for solving (1), where the search direction of the quasi-Newton methods will be modified using the search direction of the conjugate gradient method approach. We prove that our algorithm with the Wolfe line search is globally convergent for general objective function. Then, we test the new approach on standard test problems, comparing the numerical results with the results of applying the quasi-Newton methods to the same set of test problems.

Methods

The iterative method is used to solve unconstrained optimization problems in order to get the minimal value of the function where the gradient is 0. Hence, the iterative formula for the quasi-Newton methods will be defined as

$$x_{k+1} = x_k + a_k d_k, \tag{6}$$

where the a_k and d_k denote the step size and the search direction, respectively. The step size must always have a positive value such that $f(x)$ is sufficiently reduced. The success of a line search depends on the effective choices of both the search direction d_k and the step size a_k. There are a lot of formulas in calculating the step size, which are divided into an exact line search and an inexact line search.

The ideal choice would be the exact line search formula, which is defined as a_k = arg min($f(x_k + \alpha_k d_k)$ α>0, but in general it is too expensive to identify this value. Generally, it requires too many evaluations of the objective function f and also its gradient g. The inexact line search has a few formulas which have been presented by previous researchers, such as the Armijo line search (Armijo 1966), Wolfe Condition (Wolfe 1969, 1971) and Goldstein Condition (Goldstein 1965). Shi (2006) claimed that among several well-known inexact line search procedures, the Armijo line search is the most useful and the easiest to implement in the computational calculation. It is also easy to implement it in programming like Matlab and Fortran. The Armijo line search is described as follows:

Given $s > 0, \lambda \in (0,1)$ $\sigma \in (0,1)$ and $\alpha_i = \max\{s, s\lambda, s\lambda^2, \ldots\}$ such that

$$f(x_k) - f(x_k + a_k d_k) \geq -\sigma \alpha_k g_k^T d_k, \tag{7}$$

$k = 0, 1, 2, 3, \ldots$. The reduction in f should be proportional to both the step size and directional derivative $g_k^T d_k$.

The search directions are also important in order to determine the value of f, which decreases along the direction. Moreover, the search direction of the quasi-Newton methods often has the form

$$d_k = -B_k^{-1} g_k, \tag{8}$$

where B_k is a symmetric and nonsingular matrix of approximation of the Hessian (3). Initial matrix B_0 is chosen by an identity matrix which subsequently is updated by an update formula. When d_1 is defined by (8) and B_k is a positive definite, we have $d_k^T = -g_k^T B_k^{-1} g_k < 0$, and therefore d_k is a descent direction. Hence, the algorithm for an iteration method of ordinary BFGS is described as follows:

Algorithm 1 (BFGS method)

Step 0. Given a starting point x_0 and $B_0 = I_n$. Choose values for s, β, and σ.
Step 1. Terminate if $\|g(x_{k+1})\| < 10^{-6}$.
Step 2. Calculate the search direction by (8).
Step 3. Calculate the step size α_1 by (7).
Step 4. Compute the difference $s_k = x_{k+1} - x_k$ and $y_k + g_{k+1} - g_k$.
Step 5. Update B_k by (3) to obtain B_{k+1}.
Step 6. Set $k = k + 1$ and go to Step 1.

A New Search Direction

In this section, we will discuss the new search direction for the quasi-Newton methods, which will be proposed by using the concept of the conjugate gradient method. The search direction of conjugate gradient method is:

$$d_k = \begin{cases} -g_k & k = 0 \\ -g_k + \beta_k d_{k-1} & k \geq 1, \end{cases} \tag{9}$$

where β_1 is a coefficient of the conjugate gradient method. To incorporate more curvature information to the conjugate gradient direction, Birgin and Martinez (2001) proposed to scale the search direction (more precisely, on the steepest descent part $-g_k$) by some Rayleigh quotient of the local Hessian, which gives arise a new class of methods called spectral conjugate gradient methods. Surprisingly, the spectral conjugate gradient methods outperform sophisticated conjugate gradient algorithms in many problems. The numerical results in Birgin and Martinez (2001) and Zhang et al. (2006) suggested that spectral gradient and conjugate gradient ideas could be combined in order to obtain even more efficient algorithms. Motivated by this fact, we attempt to employ the well-known BFGS updating matrix, which would carry better spectral information so that the concept of the conjugate gradient method's search direction will be implemented into a new search, known as the HBFGS method and is given by,

$$d_k = \begin{cases} -B_k^{-1} g_k & k = 0 \\ -B_k^{-1} g_k + \lambda_k d_{k-1} & k \geq 1, \end{cases} \tag{10}$$

where B_k is the BFGS updating matrix and $\lambda_k = \eta g_k^T g_k / g_k^T d_{k-1}$ with $\eta \in (0,1]$ is chosen to ensure conjugacy. With these considerations in mind, we shall now propose the algorithm for the HBFGS method as follows:

Algorithm 2 (HBFGS method)

Step 0. Given a starting point x_0 and $B_0 = I_n$. Choose values for s, β and σ.
Step 1. Terminate if $\|g(x_{k+1})\| < 10^{-6}$.
Step 2. Calculate the search direction by (10).
Step 3. Calculate the step size α_k by (7).
Step 4. Compute the difference $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$.
Step 5. Update B_k by (3) to obtain B_{k+1}.
Step 6. Set $k = k + 1$ and go to Step 1.

Based on Algorithms 1 and 2, we assume that every search direction d_k satisfied the descent condition

$$g_k^T d_k < 0, \tag{11}$$

for all $k \geq 0$. If there exists a constant $c_1 > 0$ such that

$$g_k^T d_k \leq c_1 \|g_k\|^2 \tag{12}$$

for all $k \geq 0$, then the search directions satisfy the sufficient descent condition which can be proven in Theorem 3.2. Hence, we make a few assumptions based on the objective function.

Assumption 3.1

H_1: The objective function f is twice continuously differentiable.

H_2: The level set L is convex. Moreover, positive constants c_1 and c_2 exist, satisfying

$$c_1 \|z\|^2 \leq z^T F(x) z \leq c_2 \|z\|^2 , \tag{13}$$

for all $z \in R^n$ and $x \in L$, where $F(x)$ is the Hessian matrix for f.

H3: The Hessian matrix is Lipschitz continuous at the point x^*, that is, the positive constant c_3 exists, satisfying

$$\|G(x) - G(x^*)\| \leq c_3 \|x - x^*\| \tag{14}$$

for all x in a neighbourhood of x^*.

If the sequences $\{x_k\}$ are converging to a point x^*, it is to be expected that y_1 is approximately equal to $G(x^*)s_k$.

Theorem 3.1 (Byrd & Nocedal 1989)

Let $\{B_k\}$ e generated by the BFGS formula (3), where B_1 is symmetric and positive definite, and where $y_k^T s_k > 0$ for all k. Furthermore, assume that $\{s_k\}$ and $\{y_k\}$ are such that

$$\frac{\|(y_k - G^*) s_k\|}{\|s_k\|} \leq \varepsilon_k ,$$

for some symmetric and positive definite matrix $G(x^*)$ and for some sequence $\{\varepsilon_k\}$ with the property $\sum_{k=1}^{\infty} \varepsilon_k < \infty$. Then

$$\lim_{k \to \infty} \frac{\|(B_k - G_*) d_k\|}{\|d_k\|} = 0, \tag{15}$$

and the sequences $\{\|B_k\|\}$, or $\{\|B_k^{-1}\|\}$ are bounded.

Theorem 3.2

Suppose that Assumption 3.1 and Theorem 3.1 hold. Then, condition (12) holds for all $k \geq 0$.

Proof:

From (10), we see that

$$\begin{aligned} g_k^T d_k &= -g_k^T B_k^{-1} g_k - \eta g_k^T g_k \\ &= -g_k^T B_k^{-1} - \eta \frac{g_k^T g_k}{g_k^T d_{k-1}} g_k^T d_{k-1}. \end{aligned}$$

and using the Cauchy inequality, we get

$$\begin{aligned} g_k^T d_k &\leq -g_k^T \delta_k g_k - \eta g_k^T g_k \\ &\leq -\delta_k \|g_k\|^2 - \eta \|g_k\|^2 \\ &\leq c_1 \|g_k\|^2 , \end{aligned}$$

where $c_1 = -(\delta_k + \eta)$ which is bounded away from zero. Hence, (12) holds and the proof is completed.

Lemma 3.1

Under Assumption 3.1, positive constants c_2 and ϖ_2 exist such that for any x_k and any d_k with $g_k^T d_k < 0$, the step size a_k, produced by Algorithm 2, will satisfy either,

$$f\left(x_k+\alpha_k d_k\right)-f_k \le c_4 \frac{\left(g_k^T d_k\right)^2}{\left\|d_k\right\|^2}, \qquad (16)$$

or

$$f\left(x_k+\alpha_k d_k\right)-f_k \le c_5 g_k^T d_k .$$

Proof:

Suppose that $a_k < 1$, which means that (7) failed for step size $a' \le a_k/\tau$:

$$f\left(x_k+\alpha_k' d_k\right)-f\left(x_k\right) \le \varpi a' g_k^T d_k . \qquad (17)$$

Then, using the mean value theorem we obtain

$$f\left(x_{k+1}\right)-f\left(x_k\right) \le \bar{g}^T\left(x_{k+1}-x_k\right),$$

where $\bar{g}=\nabla f(\bar{x})$, for some $\bar{x}\in\left(x_k,x_{k+1}\right)$. Now, by the Cauchy-Schwartz inequality, we get

$$\begin{aligned}\bar{g}^T\left(x_{k+1}-x_k\right) &= g^T\left(x_{k+1}-x_k\right)+\left(\bar{g}-g_k\right)^T\left(x_{k+1}-x_k\right)\\ &= g^T\left(x_{k+1}-x_k\right)+\left\|\bar{g}-g_k\right\|\left(x_{k+1}-x_k\right)\\ &\le g^T\left(x_{k+1}-x_k\right)+\mu\left\|x_{k+1}-x_k\right\|^2\\ &\le g^T\left(a'd_k\right)+\mu\left\|a'd\right\|^2\\ &\le g^T\left(a'd_k\right)+\mu\left(a'\|d\|\right)^2 .\end{aligned}$$

Thus from H3

$$(\varpi-1)a'g_k^T d_k < a'\left(\bar{g}-g_k\right)^T d_k \le M\left(a'\left\|d_k\right\|\right)^2,$$

which implies that

$$a_k \ge \tau a' > \tau\left(1-\varpi\right)\frac{-g_k^T d_k}{M\left(a'\left\|d_k\right\|\right)^2}.$$

Substituting this into (17), we have

$$f\left(x_k+\alpha_k' d_k\right)-f\left(x_k\right) \le c_4 \frac{-g_k^T d_k}{\left(a'\left[d_k\right]\right)^2},$$

where $c_6 = \tau(1-\varpi)/M$, which gives (16).

Theorem 3.3 (Global convergence)

Suppose that Assumption 3.1 and Theorem 3.1 hold. Then

$$\lim_{k\to\infty}\left\|g_k\right\|^2 = 0.$$

Proof:

Combining the descent property (12) and Lemma 3.1 gives

$$\sum_{k=0}^{\infty}\frac{\left\|g_k\right\|^4}{\left\|d_k\right\|^2} < \infty. \qquad (18)$$

Hence, from Theorem 3.3 we can define that $\left\|d_k\right\| \le -c_1\left\|g_k\right\|$. Then (18) will be simplified as $\sum_{k=0}^{\infty}\left\|g_k\right\|^2 < \infty$. Therefore, the proof is completed.

NUMERICAL RESULT

In this section, we use a large number of test problem considered in Andrei (2008), Zbigniew (1996) and More et al. (1981) in Table 1 to analyse the performance of the HBFGS method with the BFGS method. The dimensions of the tests range between 2 and 1000. As suggested by Hillstrom (1977), for each of the test problems, three initial points are used, starting from a point that is closer to the solution point and moving to the one that is furtest from it. In doing so, it leads us to test the global convergence properties and the robustness of our method.

The comparison between Algorithm 1 (BFGS) and Algorithm 2 (HBFGS) based on the number of iterations and cpu-time (second). For the Armijo line search, we use $s = 1$. $\beta = 0.5$ and $\sigma = 0.1$. In our implementation, the programs are all written in Matlab using the Intel Pentium ® Dual Core of the processor. The stopping criteria that we used in both algorithms are $\left\|g\left(x_{i+1}\right)\right\| \le 10^{-6}$. The Euclidean norm is used in the convergence test to make these results comparable.

The performance results will be shown in Figures 1 and 2, respectively, using the performance profile introduced by Dolan and More (2002). The performance profile seeks to find how well the solvers perform relative

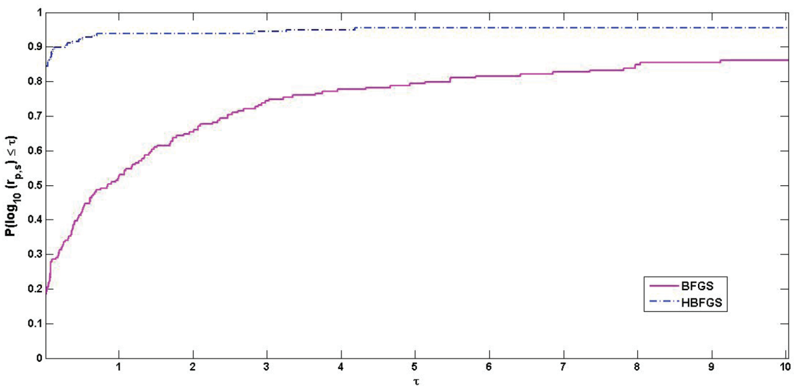

FIGURE 1. Performance profile in a $\log_{10}$ scaled based on iteration

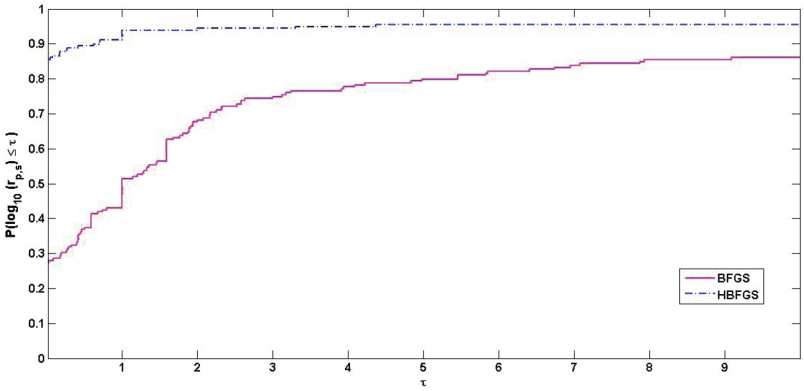

FIGURE 2. Performance profile in a $\log_{10}$ scaled based on CPU time

TABLE 1. A list of problem functions

Test problem	***n*–dimensional**	Initial points
Powell badly scaled	2	(10,10),(100,100),(1000,1000)
Beale	2	(2,2),(30,30),(700,700)
Biggs Exp6	6	(30,…,30),(50,…,50),(2,…,2)
Chebyquad	4, 6	(10,…,10),(100,…,100),(1000,…,1000)
Colville polynomial	4	(10,…,10),(200,…,200),(500,…,500),
Variably dimensioned	4, 8	(10,…,10),(100,…,100),(700,…,700),(1000,…,1000)
Freudenstein and Roth	2	(2,2),(10,10),(200,200)
Goldstein price polynomial	2	(10,10),(100,100),(1000,1000)
Himmelblau	2	(200,200),(500,500),(1000,1000)
Penalty 1	2, 4	(10,…,10),(100,…,100),(1000,…,1000)
Extended powell singular	4, 8	(2,…,2),(20,…,20),(150,…,150),(90,…,90)
Rosenbrock	2, 10, 100, 200, 500, 1000	(5,5),(50,50),(1000,1000),(10,…,10),(100,…,100),(800,…,800),(15,…,15),(125,…,125),(150,…,150),(210,…,210)
Trigonometric	6, 10, 100, 200, 500, 1000	(10,…,10),(75,…,75),(500,…,500),(100,…,100),(1000,…,1000),(200,…,200)
Watson	4, 8	(5,…,5),(20,…,20),(200,…,200), (70,…,70)
Six-Hump camel back polynomial	2	(15,15),(100,100),(1000,1000)
Extended shallow	2,4,10,100,200, 500, 1000	(20,20),(70,70),(200,200), (500,…,500),(50,…,50),(100,…,100), (350,…,350),(1000,…,1000),(900,…,900), (150,…,150)
Extended strait	2,4,10,100,200, 500, 1000	(100,100),(300,300),(500,500),(4,…,4),(70,…,70),(900,…,900),(50,…,50),(400,…,400),(1000,…,1000),(200,…,200)
Scale	2	(2,2),(30,30),(150,150),
Raydan 1	2	(20,20),(50,50),(200,200),
Raydan 2	2,4,10,100,200, 500, 1000	(20,20),(70,70),(200,200),(15,…,15),(50,…,50),(200,…,20),(100,…,100)
Diagonal 3	2	(50,50),(100,100),(200,200)
Cube	2,10,100	(4,4),(40,40),(100,100),(20,…,20),(50,…,50),(80,…,80),(6,…,6),(55,…,55),(300,…,300),(150,…,150)
De Jong F2	2	(5,5),(50,50),(1000,1000)
PSC 1	2	(5,5),(100,100),(500,500)

TABLE 2. Global characteristics of BFGS and HBFGS method

Global characteristic	BFGS	HBFGS
Total number of iterations	213484	74752
Total cpu time (s)	5580.3	1742.6

to the other solvers on a set of problems. In general, $P(\tau)$ is the fraction of problems with performance ratio τ, thus, a solver with high values of $P(\tau)$ or one that is located at the top right of the figure is preferable.

Figures 1 and 2 show that the HBFGS method has the best performance since it can solve 95.53% of the test problems compared with the BFGS (86.63%). Moreover, we can also say that the HBFGS method is the fastest solver on approximately 84.36% of the test problems for iteration and 85.47% of CPU time. Table 2 shows the global characteristics corresponding to these test problems in Table 1.

We see that the HBFGS is better than BFGS method in both characteristics. From Table 2, we can see that the total number of iterations is reduced by 65% and the total cpu time is 68.8% for HBFGS method. So, we can conclude that the HBFGS is much better compare to BFGS method.

CONCLUSION

We have presented a new hybrid method for solving unconstrained optimization problems. The numerical results for a small dimension of test problems show that the HBFGS method is efficient and robust in solving unconstrained optimization problems. The numerical results and figures from the programming are reported and analysed to show the characters of the proposed method. Our further interest is to try the HBFGS method with the coefficient of the conjugate gradient methods Fletcher-Reeves, Hestenes-Steifel and the Liu-Storey coefficient for λ.

ACKNOWLEDGMENTS

This research was supported by IUKL's Internal Funding (Int.Fund.1). We would also like to thank Dr. Mohd Rivaie and the reviewer on the valuable comments and advices.

REFERENCES

Andrei, N. 2008. An unconstrained optimization test functions collection. *Advanced Modelling and Optimization* 10(1): 147-161.

Armijo, L. 1966. Minimization of functions having Lipschitz continuous partial derivatives. *Pacific Journal of Mathematics* 16(1): 1-3.

Birgin, E. & Martínez, J.M. 2001. A spectral conjugate gradient method for unconstrained optimization. *Applied Mathematics and Optimization* 43: 117-128.

Buckley, A. & Lenir, A. 1983. QN-like variable storage conjugate gradients. *Mathematical Programming* 27(2): 155-175.

Byrd, R.H. & Nocedal, J. 1989. A tool for the analysis of Quasi-Newton methods with application to unconstrained minimization. *SIAM Journal on Numerical Analysis* 26(3): 727-739.

Byrd, R.H., Nocedal, J. & Yuan, Y-X. 1987. Global convergence of a class of Quasi-Newton methods on convex problems. *SIAM Journal on Numerical Analysis* 24 (5): 1171-1191.

Dai, Y-H. 2002. Convergence properties of the BFGS algoritm. *SIAM Journal on Optimization* 13(3): 693-702.

Dolan, E.D. & Moré. J.J. 2002. Benchmarking optimization software with performance profiles. *Mathematical Programming* 91(2): 201-213.

Goldstein, A. 1965. On steepest descent. *Journal of the Society for Industrial and Applied Mathematics Series A Control* 3(1): 147-151.

Hillstrom, E.K. 1977. A simulation test approach to the evaluation of nonlinear optimization algorithm. *Journal ACM Trans. Mathematics Software* 3(4): 305-315.

Li, D-H. & Fukushima, M. 2001. A modified BFGS method and its global convergence in nonconvex minimization. *Journal of Computational and Applied Mathematics* 129: 15-24.

Mamat Mustafa, Ismail Mohd, Leong Wah June & Yosza Dasril. 2009. Hybrid Broyden method for unconstrained optimization. *International Journal of Numerical Methods and Applications* 1(2): 121-130.

Mascarenhas, W.F. 2004. The BFGS method with exact line searches fails for non-convex objective functions. *Mathematical Programming* 99(1): 49-61.

More, J.J., Garbow, B.S. & Hillstrom, K.E. 1981. Testing unconstrained optimization software. *ACM Trans. Math. Softw.* 7(1): 17-41.

Nocedal, J. 1980. Updating Quasi-Newton matrices with limited storage. *Mathematics of Computation* 35(151): 773-782.

Powell, M.J.D. 1976. Some global convergence properties of variable metric algorithm for minimization without exact line searches in nonlinear programming. In *SSIAM-AMS Proc IX AMS Providence,* edited by Cottle, R.W. & Lemke, C.E. RI: 53-72.

Shi, Z-J. 2006. Convergence of quasi-Newton method with new inexact line search. *Journal of Mathematical Analysis and Applications* 315(1): 120-131.

Wolfe, P. 1971. Convergence conditions for ASCENT methods. II: Some corrections. *SIAM Review* 13(2): 185-188.

Wolfe, P. 1969. Convergence conditions for ASCENT methods. *SIAM Review* 11(2): 226-235.

Zbigniew M. 1996. *Genetic Algorithms + Data Structures = Evolution Programs*. New York: Springer Verlag.

Zhang, L., Zhou, W.J. & Li, D.H. 2006. A descent modified Polak-Ribiére-Polyak conjugate gradient method and its global convergence. *IMA Journal of Numerical Analysis* 26: 629-640.

Mohd Asrul Hery Ibrahim
Fakulti Keusahawanan dan Perniagaan
Universiti Malaysia Kelantan
16100 Pengakalan Chepa, Kelantan
Malaysia

Mustafa Mamat*
Fakulti Informatik dan Komputeran
Universiti Sultan Zainal Abidin
21030 Kuala Terengganu, Terengganu
Malaysia

Leong Wah June
Jabatan Matematik, Fakulti Sains
Universiti Putra Malaysia
43400 Serdang, Selangor, D.E.
Malaysia

*Corresponding author; email: must@unisza.edu.my

Received: 25 April 2013
Accepted: 13 February 2014