Comp. Appl. Math. (2018) 37:5113–5125
https://doi.org/10.1007/s40314-018-0620-8

A new modified BFGS method for unconstrained optimization problems

Razieh Dehghani[1] · **Narges Bidabadi**[1] · **Mohammad Mehdi Hosseini**[1]

Received: 6 June 2017 / Revised: 10 April 2018 / Accepted: 12 April 2018 /
Published online: 19 April 2018

Abstract Using chain rule, we propose a modified secant equation to get a more accurate approximation of the second curvature of the objective function. Then, based on this modified secant equation we present a new BFGS method for solving unconstrained optimization problems. The proposed method makes use of both gradient and function values, and utilizes information from two most recent steps, while the usual secant relation uses only the latest step information. Under appropriate conditions, we show that the proposed method is globally convergent without convexity assumption on the objective function. Comparative numerical results show computational efficiency of the proposed method in the sense of the Dolan–Moré performance profiles.

Keywords Unconstrained optimization · Nonlinear function · Two-step secant equation · Global convergence

Mathematics Subject Classification 90C53 · 65K05

1 Introduction

Consider the unconstrained nonlinear optimization problem

$$\min f(x), \quad x \in \mathbb{R}^n, \tag{1}$$

Communicated by Joerg Fliege.

✉ Narges Bidabadi
n_bidabadi@yazd.ac.ir

Razieh Dehghani
rdehghani@stu.yazd.ac.ir

Mohammad Mehdi Hosseini
hosse_m@yazd.ac.ir

1 Department of Mathematical Sciences, Yazd University, Yazd, Iran

where f is twice continuously differentiable. The quasi-Newton methods are popular iterative methods for solving (1), whose iterates are constructed as follows:

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

where α_k is a step size and d_k is a descent direction obtained by solving $B_k d_k = -g_k$, where $g_k = \nabla f(x_k)$ and B_k is an approximation of the Hessian matrix of f at x_k which satisfies the secant equation.

The standard secant equation can be established as follows (see Dennis and Schnabel 1983). We have

$$g_{k+1} - g_k = \int_0^1 \nabla^2 f(x_k + t s_k) \mathrm{d}t s_k, \tag{2}$$

where $s_k = x_{k+1} - x_k$. Since B_{k+1} is to approximate $G(x_{k+1}) = \nabla^2 f(x_{k+1})$, the secant equation is defined to be

$$B_{k+1} s_k = y_k, \tag{3}$$

where $y_k = g_{k+1} - g_k$. The relation (3) is sometimes called the standard secant equation.

A famous family of quasi-Newton methods is Broyden family Broyden (1965) in which the updates are defined by

$$B_{k+1} = B_k - \frac{B_k s_k s_k^{\mathrm{T}} B_k}{s_k^{\mathrm{T}} B_k s_k} + \frac{y_k y_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k} + \mu w_k w_k^{\mathrm{T}}, \quad w_k = (s_k^{\mathrm{T}} B_k s_k)^{1/2} \left[\frac{y_k}{s_k^{\mathrm{T}} y_k} - \frac{B_k s_k}{s_k^{\mathrm{T}} B_k s_k} \right], \tag{4}$$

where μ is a scale parameter. The BFGS, DFP and SR1 updates are obtained by setting $\mu = 0$, $\mu = 1$ and $\mu = 1/(1 - s_k^{\mathrm{T}} B_k s_k / s_k^{\mathrm{T}} y_k)$, respectively.

Among quasi-Newton methods, the most efficient method is the BFGS method Broyden (1970).

When f is convex, the global convergence of the BFGS method have been studied by some authors (see Byrd and Nocedal 1989; Byrd et al. 1987; Griewank 1991; Powell 1976; Toint 1986). However, the BFGS method is very efficient as regards numerical performance, but Dai (2003) have constructed an example to show that this method may fail for non-convex functions with inexact Wolfe line searches. In addition, Mascarenhas (2004) showed that the nonconvergence of the standard BFGS method even with exact line search.

Global convergence of the BFGS method for the general functions under Wolfe line search is still an open problem. Recently, Yuan et al. (2017, 2018) provided a positive answer, and proved the global convergence of BFGS method under a modified weak Wolfe–Powell line search technique for general functions.

To obtain better quasi-Newton methods, many modified methods have been presented (see Li and Fukushima 2001a, b; Wei et al. 2006; Yuan and Wei 2009, 2010; Yuan et al. 2017, 2018; Zhang et al. 1999; Zhang and Xu 2001).

Li and Fukushima (2001a, b) made a modification on the standard BFGS method as follows:

$$B_{k+1} s_k = \overline{y}_k, \tag{5}$$

with

$$\overline{y}_k = y_k + r_k s_k, \quad r_k = C\|g_k\|^2 + \max\left(-\frac{y_k^{\mathrm{T}} s_k}{\|s_k\|^2}, 0\right), \tag{6}$$

where C is a positive constant.

They showed this method is globally convergent without a convexity assumption on the objective function f.

The usual secant equation employs only the gradients and the available function values are ignored. To get a higher order accuracy of approximating the Hessian matrix of the objective function, several researchers have modified the usual secant equation (3) to make full use of both the gradient and function values (see Wei et al. 2006; Yuan and Wei 2009, 2010; Yuan et al. 2017, 2018; Zhang et al. 1999; Zhang and Xu 2001).

Wei et al. (2006), using Taylor's series, modified (3) as follows:

$$B_{k+1}s_k = \tilde{y}_k, \tag{7}$$

where $\tilde{y}_k = y_k + \frac{\vartheta_k}{\|s_k\|^2}s_k$ and $\vartheta_k = 2(f_k - f_{k+1}) + (g_k + g_{k+1})^{\mathrm{T}}s_k$.

Recently, Yuan and Wei (2010) considered an extension of the modified secant equation (7) as follows:

$$B_{k+1}s_k = y_k + \frac{\max(\vartheta_k, 0)}{\|s_k\|^2}s_k. \tag{8}$$

Numerical results of Yuan and Wei (2010) showed that the modified BFGS method suggested by Yuan and Wei outperformed the CG methods proposed by Wei et al. (2006) and Li and Fukushima (2001b) and the standard BFGS method Broyden (1970).

Such modified secant equations make use of both the available gradient and function values only at the last two points. Here, we employ chain rule, and introduced a different secant relation utilizing information from three most recent points and using both the available gradient and function values. Then, we make use of the new secant equation in a BFGS updating formula.

This work is organized as follows: In Sect. 2, we first employ chain rule to derive an alternative secant equation and then we outline our proposed algorithm. In Sect. 3, we investigate the global convergence of the proposed method. Finally, in Sect. 4, we report some numerical results.

2 Two-step BFGS method

In this section, we obtain a new secant equation. Next, we use this secant equation and we give the algorithm.

2.1 Proposed modified secant equation

Here, we intend to make use of three iterates x_{k-1}, x_k and x_{k+1} generated by some quasi-Newton algorithm. Using chain rule to function $\nabla^2 f(X(t))$, we know

$$g(X(1)) - g(X(-1)) = \int_{-1}^{1} \nabla^2 f(X(t))\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t, \tag{9}$$

where $X(t)$ is a differentiable curve in $\mathbb{R}^n$.

Now, suppose that $X(t)$ is the interpolating curve so that

$$X(-1) = x_{k-1}, \qquad X(0) = x_k, \qquad X(1) = x_{k+1}. \tag{10}$$

Of course there are various choices for $X(t)$. Here, motivated by Ford and Saadallah (1987), we define a nonlinear interpolating with a free parameter as follows:

$$X(t) \equiv x(t,\theta) \equiv q(t)\frac{1}{1+t\theta}, \tag{11}$$

where $q(t) = a_0 + a_1 t + a_2 t^2$ ($\{a_i\}_{i=0}^2$ are constant vectors) and θ is a parameter to be chosen later.

Since $q(t) = x(t,\theta)(1+t\theta)$ is a second degree polynomial and hence, may be written in its Lagrangian form

$$q(t) = \sum_{j=0}^{2} L_j(t) q(t_j), \tag{12}$$

where $q(t_j) = x(t_j,\theta)(1+t_j\theta)$ and the $L_j(t)$ are the basic Lagrange polynomials:

$$L_j(t) = \prod_{i=0, i\neq j}^{2} \frac{t-t_i}{t_j-t_i}, \qquad j = 0, 1, 2. \tag{13}$$

After some algebraic manipulations, (12) can be written as

$$q(t) \equiv \left[\frac{t(t+1)(1+\theta)}{2} x_{k+1} + (1-t^2)x_k + \frac{t(t-1)(1-\theta)}{2} x_{k-1}\right]. \tag{14}$$

Therefore

$$x(t,\theta) \equiv \left[\frac{t(t+1)(1+\theta)}{2} x_{k+1} + (1-t^2)x_k + \frac{t(t-1)(1-\theta)}{2} x_{k-1}\right]\frac{1}{1+t\theta}. \tag{15}$$

Taking the derivative from both sides of (15), we obtain:

$$\begin{aligned}\frac{\mathrm{d}x(t,\theta)}{\mathrm{d}t} \simeq &\left(\frac{(1+2t)(1+\theta)}{2} x_{k+1} - 2t x_k + \frac{(2t-1)(1-\theta)}{2} x_{k-1}\right)\frac{1}{1+t\theta}\\ &+\left(\frac{t(t+1)(1+\theta)}{2} x_{k+1} + (1-t^2)x_k + \frac{t(t-1)(1-\theta)}{2} x_{k-1}\right)\frac{-\theta}{(1+t\theta)^2}.\end{aligned} \tag{16}$$

On the other hand, using Lagrange interpolation we have

$$\nabla^2 f(x(t,\theta)) \simeq \sum_{j=0}^{2} L_j(t)\, \nabla^2 f(x_{k+j-1}). \tag{17}$$

Substituting relation (17) into (9), we obtain:

$$\begin{aligned} g(x_{k+1}) - g(x_{k-1}) &= \int_{-1}^{1} \nabla^2 f(X(t)) \frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t\\ &= \sum_{j=0}^{2}\int_{-1}^{1} L_j(t)\, \nabla^2 f(x_{k+j-1})\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t, \end{aligned} \tag{18}$$

where $\frac{\mathrm{d}X(t)}{\mathrm{d}t} \equiv \frac{\mathrm{d}x(t,\theta)}{\mathrm{d}t}$ given by (16).

Now, by considering B_{k+1} as a new approximation of $\nabla^2 f(x_{k+1})$, (18) leads to

$$B_{k+1}\int_{-1}^{1} L_2(t)\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t = g(x_{k+1}) - g(x_{k-1}) - B_k\int_{-1}^{1} L_1(t)\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t - B_{k-1}\int_{-1}^{1} L_0(t)\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t, \quad (19)$$

where B_{k-1} and B_k approximate $\nabla^2 f(x_{k-1})$ and $\nabla^2 f(x_k)$, respectively.

Equation (19) provides a new modified secant relation as follows:

$$B_{k+1}s_k^* = y_k^*, \quad (20)$$

where y_k^* and s_k^* are given by

$$s_k^* = \int_{-1}^{1} L_2(t)\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t, \quad (21)$$

and

$$y_k^* = y_k + y_{k-1} - B_k\int_{-1}^{1} L_1(t)\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t - B_{k-1}\int_{-1}^{1} L_0(t)\frac{\mathrm{d}X(t)}{\mathrm{d}t}\mathrm{d}t, \quad (22)$$

with $\frac{\mathrm{d}X(t)}{\mathrm{d}t} \equiv \frac{\mathrm{d}x(t,\theta)}{\mathrm{d}t}$ given by (16).

Now, the issue is choosing a strategy to determine a numerical value for θ. Define

$$\varphi(t,\theta) = f(x(t,\theta)). \quad (23)$$

Clearly we have

$$\int_{-1}^{1} \varphi'(t,\theta) = f_{k+1} - f_{k-1}. \quad (24)$$

On the other hand, a reasonable estimate of the integral would be given by

$$\int_{-1}^{1} \varphi'(t,\theta) \simeq 2\varphi'(0,\theta) = 2x'(0,\theta)g_k. \quad (25)$$

Remark A In constructing this estimate of the integral, we are using advantage of the fact that $t = 0$ is an interior point of the interval of integration $[-1, 1]$.

On the other hand, from Eq. (16) we have

$$x'(0,\theta) \simeq \frac{1}{2}\left[s_k + s_{k-1} + \theta s_k - \theta s_{k-1}\right]. \quad (26)$$

From (24), (25) and (26), we obtain

$$\theta \equiv \frac{f_{k+1} - f_{k-1} - s_k^{\mathrm{T}}g_k - s_{k-1}^{\mathrm{T}}g_k}{s_k^{\mathrm{T}}g_k - s_{k-1}^{\mathrm{T}}g_k}. \quad (27)$$

Obviously (27) is a good estimation of θ and it dose not require expensive computations.

Also, it is easy to see that if the denominator, $(1+t\theta)$ in (11), becomes zero over the interval $[-1, 1]$, then the interpolating curve $x(t,\theta)$ is undesirable. To overcome this difficulty, since the denominator $(1 + t\theta)$ is positive at $t = 0$, we impose the two conditions as follows:

$$1 + \theta > 0 \qquad 1 - \theta > 0$$

that is,

$$-1 < \theta < 1. \tag{28}$$

In implementing the new algorithm, if the condition (28) dose not hold, then we set $\theta = 0$.

2.2 Proposed BFGS algorithm

Here, we apply the modified secant equation given in the previous subsection then we propose new modified BFGS method such that B_{k+1} update by

$$B_{k+1} = B_k - \frac{B_k s_k^* s_k^{*\mathrm{T}} B_k}{s_k^{*\mathrm{T}} B_k s_k^*} + \frac{y_k^* y_k^{*\mathrm{T}}}{s_k^{*\mathrm{T}} y_k^*}, \tag{29}$$

where y_k^* and s_k^* are given by

$$s_k^* = \int_{-1}^{1} L_2(t) \frac{\mathrm{d}X(t)}{\mathrm{d}t} \mathrm{d}t, \tag{30}$$

and

$$y_k^* = y_k + y_{k-1} - B_k \int_{-1}^{1} L_1(t) \frac{\mathrm{d}X(t)}{\mathrm{d}t} \mathrm{d}t - B_{k-1} \int_{-1}^{1} L_0(t) \frac{\mathrm{d}X(t)}{\mathrm{d}t} \mathrm{d}t, \tag{31}$$

with $\frac{\mathrm{d}X(t)}{\mathrm{d}t}$ and θ are given by (16) and (27) respectively.

We note that this new modified BFGS contains information from the three most recent points where the usual BFGS method and modified BFGS method introduced by Li and Fukushima (2001b), Wei et al. (2006) and Yuan and Wei (2010), make use of the information merely at the two latest points. In addition, both the available gradient and function values are being utilized.

We know, $s_k^{*\mathrm{T}} y_k^* > 0$, is sufficient to ensure B_{k+1} to be positive definite (see Nocedal and Wright 2006) and consequently, the generated search directions are descent directions. However, for a general function f, $s_k^{*\mathrm{T}} y_k^*$ may not be positive for all $k \geq 0$, and consequently B_{k+1} may not be positive definite.

For preserving positive definiteness of the updates, we set

$$B_{k+1} = \begin{cases} B_k - \frac{B_k s_k^* s_k^{*\mathrm{T}} B_k}{s_k^{*\mathrm{T}} B_k s_k^*} + \frac{y_k^* y_k^{*\mathrm{T}}}{s_k^{*\mathrm{T}} y_k^*}, & \frac{s_k^{*\mathrm{T}} y_k^*}{\|s_k^*\|^2} \geq \delta, \\ B_k, & \text{otherwise}. \end{cases} \tag{32}$$

where δ is a positive constant.

Remark B From (32), it is easy to see that $s_k^{*\mathrm{T}} y_k^* > 0$ therefore the matrix B_{k+1} generated by (32), is symmetric and positive definite for all k.

We can now give a new BFGS algorithm using new secant relation (20), as Algorithm 1.

Algorithm 1: The new modified BFGS method.

Step 1: Give ε as a tolerance for convergence, $\sigma_1 \in (0, 1)$, $\sigma_2 \in (\sigma_1, 1)$, a starting point $x_0 \in \mathbb{R}^n$, and a positive definite matrix B_0. Set $k = 0$.

Step 2: **If** $\|g_k\| < \varepsilon$ **then stop**.

Step 3: Compute a search direction d_k: Solve $B_k d_k = -g_k$.

Step 4: Compute the step length α_k satisfying the following Wolfe conditions:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \sigma_1 \alpha_k g_k^{\mathrm{T}} d_k, \tag{33}$$

$$g(x_k + \alpha_k d_k)^{\mathrm{T}} d_k \geq \sigma_2 g(x_k)^{\mathrm{T}} d_k. \tag{34}$$

Step 5: Set $x_{k+1} = x_k + \alpha_k d_k$. Compute s_k^* and y_k^* by (30) and (31) respectively, with θ given by (27). If $s_k^{*\mathrm{T}} y_k^* < 10^{-4}\|s_k^*\|\|y_k^*\|$ then set $s_k^* = s_k$ and $y_k^* = y_k$.
Step 6: Update B_{k+1} by (32).
Step 7: Set $k = k + 1$ and **go to** Step 2.

Next, we will investigate the global convergence of Algorithm 1.

3 Convergence analysis

To establish the global convergence of the Algorithm 1, we need some commonly used assumptions.

Assumption A (i) The level set $D = \{x \mid f(x) \leq f(x_0)\}$ is bounded, where x_0 is the starting point of Algorithm 1.

(ii) The function f is twice continuously differentiable and there is constant $L > 0$, such that

$$\|G(x) - G(y)\| \leq L\|x - y\|, \quad \forall x, y \in D.$$

It is clear that Assumption A implies

$$\|G(x)\| \leq m, \quad \forall x \in D, \tag{35}$$

where m is a positive constant

Since B_k is a approximate $G(x)$ at x_k, similar to Yuan and Wei (2009) and Zhu (2005) we give the following assumption.

Assumption B Assume that B_k is a good approximation to $G(x)$ at x_k., i.e.,

$$\|B_k - G(x_k)\| \leq \varepsilon_k, \tag{36}$$

where $\varepsilon_k \in (0, 1)$ are suitable quantities.

On the other hand, we have

$$\|B_k\| - \|G(x_k)\| \leq \|B_k - G(x_k)\| \leq \varepsilon_k,$$

Hence, we can give

$$\|B_k\| \leq \gamma, \quad \forall k \geq 0, \tag{37}$$

where $\gamma = \varepsilon_k + m$.

Using Assumption A and the Wolfe conditions, $\{f(x_k)\}$ is a nonincreasing sequence, which ensures $\{x_k\} \subset D$ and the existence of x^* such that

$$\lim_{k\to\infty} f(x_k) = f(x^*). \tag{38}$$

To establish the global convergence of Algorithm 1, we present the following useful Lemmas.

Lemma 3.1 *Let f satisfies assumptions A and B, and $\{x_k\}$ be generated by Algorithm 1 and there exist constants a_1 and a_2 such that*

$$\|B_k s_k\| \leq a_1\|s_k\|, \qquad s_k^T B_k s_k \geq a_2\|s_k\|^2, \tag{39}$$

for infinitely many k. Then, we have

$$\liminf_{k\to\infty} g(x_k) = 0. \tag{40}$$

Proof Since $s_k = \alpha_k d_k$, it is clear that (39) holds true if s_k is replaced by d_k. From (39) and the relation $g_k = -B_k d_k$, we have

$$d_k^{\mathrm{T}} B_k d_k \geq a_2 \|d_k\|^2, \qquad a_2\|d_k\| \leq \|g_k\| \leq a_1\|d_k\|. \tag{41}$$

Let Λ be the set of indices k for which (39) hold. Using (34) and Assumption asB, we have

$$L\alpha_k\|d_k\|^2 \geq (g_{k+1} - g_k)^{\mathrm{T}} d_k \geq -(1-\sigma_2) g_k^{\mathrm{T}} d_k. \tag{42}$$

This implies that, for any $k \in \Lambda$,

$$\alpha_k \geq \frac{-(1-\sigma_2) g_k^{\mathrm{T}} d_k}{L\|d_k\|^2} = \frac{(1-\sigma_2) d_k^{\mathrm{T}} B_k d_k}{L\|d_k\|^2} \geq \frac{(1-\sigma_2)a_2}{L}. \tag{43}$$

Moreover, by (38), we obtain

$$\sum_{k=1}^{\infty}(f_k - f_{k+1}) = \lim_{N\to\infty}\sum_{k=1}^{N}(f_k - f_{k+1}) = \lim_{N\to\infty} f(x_1) - f(x_N)) = f(x_1) - f(x^*),$$

which yields

$$\sum_{k=1}^{\infty}(f_k - f_{k+1}) < \infty.$$

Using (33), we get

$$\sum_{k=1}^{\infty}\alpha_k g_k^{\mathrm{T}} d_k < \infty,$$

which ensures

$$\lim_{k\to\infty} \alpha_k g_k^{\mathrm{T}} d_k = 0.$$

This together with (43) lead to

$$\lim_{k\in\Lambda, k\to\infty} d_k^{\mathrm{T}} B_k d_k = \lim_{k\in\Lambda, k\to\infty} -g_k^{\mathrm{T}} d_k = 0.$$

which a long with (41), yields (40). □

Now, we prove the global convergence of Algorithm 1.

Theorem 3.1 *Let f satisfy the assumptions A and B and $\{x_k\}$ be generated by Algorithm 1. Then, we have*

$$\liminf_{k\to\infty} g(x_k) = 0. \tag{44}$$

Proof Using Lemma 3.1, it is sufficient to show relation (39) holds for infinitely many k. Using (37), we have

$$\|B_k s_k\| \leq \|B_k\|\|s_k\| \leq \gamma\|\|s_k\|. \tag{45}$$

Since B_k, in Algorithm 1 is symmetric and positive definite then there exists a_2 such that

$${s_k}^{\mathrm{T}} B_k s_k \geq a_2\|s_k\|^2.$$

Then, Lemma 3.1 completes the proof. □

Table 1 Test problems taken from CUTEr library

No	Test function	Dim	No	Test function	Dim	No	Test function	Dim
1	AIRCRFTB	8	41	GULF	3	81	ARWHEAD	500
2	ALLINITU	4	42	HATFLDD	3	82	BOX	100
3	ARGLINA	200	43	HEART8LS	10	83	BRKMCC	2
4	BARD	3	44	HELIX	3	84	BROYDN7D	1000
5	BIGGS3	6	45	HILBERTA	2	85	CHAINWOO	1000
6	BIGGS5	6	46	HILBERTB	10	86	COSINE	1000
7	BIGGS6	6	47	HILBERTF	5	87	CUBE	2
8	BOX2	3	48	HILBERTG	2	88	CURLY10	1000
9	BOX3	3	49	HILBERTH	2	89	CURLY20	1000
10	BROWNAL	200	50	JENSMP	2	90	CURLY30	1000
11	BROWNBS	200	51	KOWOSB	5	91	DENSCHNE	3
12	BRYBND	5000	52	LIARWHD	5000	92	DENSCHNF	2
13	CHNROSNB	50	53	LIARWHD	1000	93	EDENSCH	36
14	DECONVU	60	54	test LIARWHD	100	94	EG2	1000
15	DENSCHNA	2	55	MANCINO	100	95	ENGVAL1	100
16	DENSCHNB	2	56	MOREBV	5000	96	FLETCBV2	5000
17	DENSCHNC	12	57	NLMSURF	5000	97	FLETCHCR	1000
18	DIXMAANA	3000	58	NONDIA	5000	98	FMINSURF	961
19	DIXMAANB	3000	59	NONDQUAR	5000	99	GENHUMPS	5000
20	DIXMAANC	3000	60	OSBORNEB	11	100	GENROSE	500
21	DIXMAAND	3000	61	PALMER5C	6	101	HAIRY	2
22	DIXMAANE	3000	62	POWELLSG	1000	102	HATFLDFL	3
23	DIXMAANF	3000	63	QUARTC	1000	103	HUMPS	2
24	DIXMAANG	3000	64	ROSENBR	2	104	JIMACK	3549
25	DIXMAANH	3000	65	S308	2	105	MARATOSB	2
26	DIXMAANI	3000	66	SCHMVETT	3	106	MSQRTALS	529
27	DIXMAANJ	3000	67	SISSER	2	107	MSQRTBLS	529
28	DIXMAANK	3000	68	SNAIL	2	108	NCB20	110
29	DIXMAANL	3000	69	SPARSQUR	1000	109	NCB20B	21
30	DIXON3DQ	1000	70	SPMSRTLS	1000	110	NONCVXU2	1000
31	DQDRTIC	1000	71	SROSENBR	1000	111	PENALTY1	50
32	DQRTIC	1000	72	TESTQUAD	5000	112	PENALTY2	50
33	EIGENALS	110	73	TOINTTGSS	5000	113	PENALTY3	100
34	EIGENBLS	110	74	TQUARTIC	5000	114	POWER	100
35	EIGENCLS	110	75	TRIDIA	5000	115	SINEVAL	2
36	ENGVAL2	3	76	VAREIGVL	50	116	SINQUAD	5
37	EXPFIT	2	77	WATSON	12	117	SPARSINE	5000
38	EXTROSNB	1000	78	WOODS	200	118	TOINTQOR	50
39	FMINSRF2	32	79	YFITU	5	119	VARDIM	10
40	GROWTHLS	3	80	ZANGWIL2	2	120	VIBRBEAM	8

4 Numerical results

We compare the performance of the following four methods on some unconstrained optimization problems:

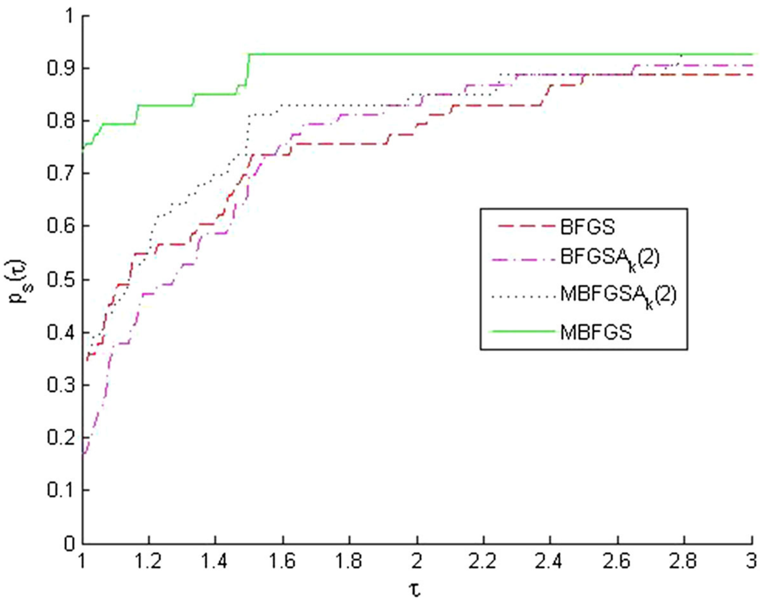

Fig. 1 The Dolan–More performance profiles using number of function evaluations

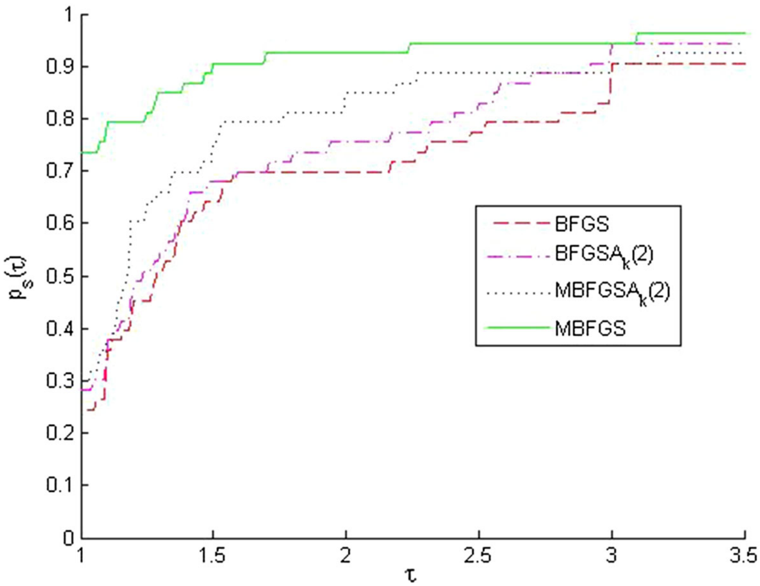

Fig. 2 The Dolan–More performance profiles using number of iterations

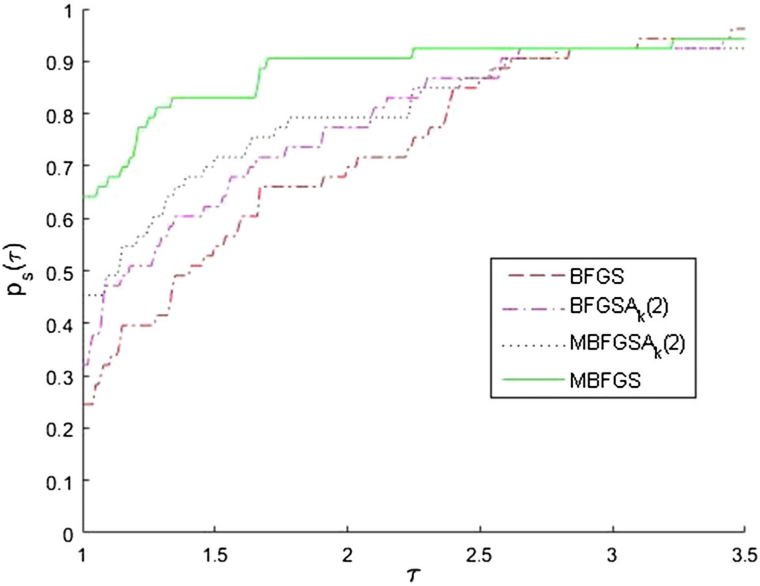

Fig. 3 The Dolan–More performance profiles using CPU times

MBFGS: proposed method (Algorithm 1).
BFGS: the usual BFGS method using (3) [2].
BFGS$A_k(2)$: the modified BFGS method of Wei et al. using (5) Wei et al. (2006).
MBFGS$A_k(2)$: the modified BFGS of Yuan and Wei using (7) Yuan and Wei (2010).

We have tested all the considered algorithms on 120 test problems from CUTEr library Gould et al. (2003). A summary of these problems are given in Table 1. All the codes were written in Matlab 7.14.0.739 (2012a) and run on PC with CPU Intel(R) Core(TM) i5-4200 3.6 GHz, 4 GB of RAM memory and Centos 6.2 server Linux operating system. In the four algorithms, the initial matrix is set to be the identity matrix and $\varepsilon = 106$. In Algorithm 1 we set $\sigma_1 = 0.01$, and $\sigma_2 = 0.9$ and $\delta = 10^{-6}$.

We used the performance profiles of Dolan and More (2002) to evaluate performance of these four algorithms with respect to CPU time, the number of iterations and the total number of function and gradient evaluations computed as $N_f + nN_g$ where N_f and N_g, respectively, denote the number of function and gradient evaluations (note that to account for the higher cost of N_g, as compared to N_f the former is multiplied by n).

Figures 1, 2 and 3 demonstrate the results of the comparisons of the four methods. From these figures, it is clear that Algorithm 1 (MBFGS) is the most efficient in solving these 120 test problems.

5 Conclusion

We introduced a modified BFGS (MBFGS) method using a new secant equation. An interesting feature of the proposed method was taking both the gradient and function values into account. Another important property of the MBFGS method was the utilization of information from the two most recent steps instead of the last step alone. Under suitable assumptions, we established the global convergence of the proposed method. Numerical results on the collection of problems from the CUTEr library showed the proposed method to be more efficient as compared to several proposed BFGS methods in the literature.

Acknowledgements We would like to thank the referees and the associate editor, whose very helpful suggestions led to much improvement of this paper.

References

Broyden CG (1965) A class of methods for solving nonlinear simultaneous equations. Math Comput 19:577–593

Broyden CG (1970) The convergence of a class of double rank minimization algorithms-Part 2: the new algorithm. J Inst Math Appl 6:222–231

Byrd R, Nocedal J (1989) A tool for the analysis of quasi-Newton methods with application to unconstrained minimization. SIAM J Numer Anal 26:727–739

Byrd R, Nocedal J, Yuan Y (1987) Global convergence of a class of quasi-Newton methods on convex problems. SIAM J Numer Anal 24:1171–1189

Dai Y (2003) Convergence properties of the BFGS algorithm. SIAM J Optim 13:693–701

Dennis JE, More JJ (1974) A characterization of superlinear convergence and its application to quasi-Newton methods. Math Comput 28:549–560

Dennis JE, Schnabel RB (1983) Numerical methods for unconstrained optimization and nonlinear equations. Prentice-Hall, Englewood Cliffs

Dolan E, More JJ (2002) Benchmarking optimization software with performance profiles. Math Program 91:201–213

Fletcher R (1970) A new approach to variable metric algorithms. Comput J 13:317–322

Ford JA, Saadallah AF (1987) On the construction of minimisation methods of quasi-Newton type. J Comput Appl Math 20:239–246

Goldfarb D (1970) A family of variable metric methods derived by variational means. Math Comput 24:23–26

Gould NIM, Orban D, Toint PhL (2003) CUTEr, a constrained and unconstrained testing environment, revisited. ACM Trans Math Softw 29:373–394

Griewank A (1991) The global convergence of partitioned BFGS on problems with convex decompositions and Lipschitzian gradients. Math Program 50:141–175

Li DH, Fukushima M (2001) On the global convergence of BFGS method for nonconvex unconstrained optimization problems. SIAM J Optim 11:1054–1064

Li DH, Fukushima M (2001) A modified BFGS method and its global convergence in nonconvex minimization. J Comput Appl Math 129:15–35

Mascarenhas WF (2004) The BFGS method with exact line searches fails for non-convex objective functions. Math Program 99:49–61

Nocedal J, Wright SJ (2006) Numerical optimization. Springer, New York

Pearson JD (1969) Variable metric methods of minimization. Comput J 12:171–178

Powell MJD (1976) Some global convergence properties of a variable metric algorithm for minimization without exact line searches. In: Cottle RW, Lemke CE (eds) Nonlinear programming. SIAMAMS proceedings, vol IX. SIAM, Philadelphia

Shanno DF (1970) Conditioning of quasi-Newton methods for function minimization. Math Comput 24:647–656

Toint PL (1986) Global convergence of the partioned BFGS algorithm for convex partially separable opertimization. Math Program 36:290–306

Wei Z, Li G, Qi L (2006) New quasi-Newton methods for unconstrained optimization problems. Appl Math Comput 175:1156–1188

Yuan G, Wei Z (2010) Convergence analysis of a modified BFGS method on convex minimizations. Comput Optim Appl 47:237–255

Yuan G, Sheng Z, Wang B, Hu W, Li C (2018) The global convergence of a modified BFGS method for nonconvex functions. J Comput Appl Math 327:274–294

Yuan G, Wei Z (2009) BFGS trust-region method for symmetric nonlinear equations. J Comput Appl Math 230:44–58

Yuan G, Wei Z, Lu X (2017) Global convergence of BFGS and PRP methods under a modified weak Wolfe–Powell line search. Appl Math Model 47:811–825

Zhang JZ, Deng NY, Chen LH (1999) New quasi-Newton equation and related methods for unconstrained optimization. J Optim Theory Appl 102:147–167

Zhang JZ, Xu CX (2001) Properties and numerical performance of quasi-Newton methods with modified quasi-Newton equation. J Comput Appl Math 137:269–278

Zhu DT (2005) Nonmonotone backtracking inexact quasi-Newton algorithms for solving smooth nonlinear equations. Appl Math Comput 161:875–895