Int. J. Anal. Appl. (2023), 21:31

International Journal of Analysis and Applications

A Combined Conjugate Gradient Quasi-Newton Method with Modification BFGS Formula

Mardeen Sh. Taher[1,*], **Salah G. Shareef**[2]

[1] *Department of Mathematics, College of Science, Duhok University, Kurdistan Region, Iraq*

[2] *Department of Mathematics, College of Science, Zakho University, Kurdistan Region, Iraq*

[*] *Corresponding author:* mardinsh.tahir@uod.ac

Abstract. The conjugate gradient and Quasi-Newton methods have advantages and drawbacks, as although quasi-Newton algorithm has more rapid convergence than conjugate gradient, they require more storage compared to conjugate gradient algorithms. In 1976, Buckley designed a method that combines the CG method with QN updates, which is better than that observed for conjugate gradient algorithms but not as good as the quasi-Newton approach. This type of method is called the preconditioned conjugate gradient (PCG) method. In this paper, we introduce two new preconditioned conjugate gradient (PCG) methods that combine conjugate gradient with a new update of quasi-Newton methods. The new quasi-Newton method satisfied the positive define, and the direction of the new preconditioned conjugate gradient is descent direction. In numerical results, it is showing the new preconditioned conjugate gradient method is more effective on several high-dimension test problems than standard preconditioning.

1. Introduction

There are many types of numerical methods to find an optimum or near-optimum solution to one or more dimensional unconstrained optimization problems, which include the cubic interpolation, golden ratio gradient descent method, the Newton and quasi-Newton methods. The most widely used for solving large-scale problems in fields such as technology, sciences, and economics is the quasi- Newton (QN) or variable metric (VM) [4], methods because of its effectiveness and stability.

Consider the unconstrained minimization problem as follows:

Received: Feb. 12, 2023.

2020 *Mathematics Subject Classification.* 49M15.

Key words and phrases. unconstrained optimization; preconditioning conjugate gradient quasi-newton; rank-two update methods.

https://doi.org/10.28924/2291-8639-21-2023-31

ISSN: 2291-8639

$$min\{f(x) : x \in R^n\}, \tag{1.1}$$

where $f(x)$ is twice continuously differentiable function over R^n, the essential idea of quasi-Newton methods is to use an approximation of the inverse Hessian, and build up an approximation of the inverse Hessian is often used information about the gradient $\nabla f(x_k)$ from some or all of the previous iterates x_k. Quasi-Newton methods, instead of the true inverse Hessian, are observed as the most complicated for solving(1.1). Earliest quasi- Newton method was proposed by William C. Davidon in 1959 [3] and later developed by Fletcher and Powell (1963) [6].The updating formula of this method generates a symmetric positive matrix of the form $H_{k+1} = H_k + Q$, where Q is a correction matrix. Then a general quasi-Newton method is started with an initial point x_0 a first approximation of the minimum point, and a matrix H_0 (usually $H_0 = I$,I is a symmetric positive definite matrix), solving the following linear equation to compute search direction

$$d_k + H_{k+1} g_k = 0 \tag{1.2}$$

and find the next point x_{k+1} by searching along a decent direction d_k such that $d_k^T g_k \leq 0$, using the following equation:

$$x_{k+1} = x_k + \alpha_k d_k. \tag{1.3}$$

To find the step length α_k must apply an appropriate line search strategy along the search direction d_k , such that the following Wolfe–Powell [10] conditions are satisfied:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq c_1 \alpha_k \nabla f_k^T d_k, \tag{1.4}$$

$$\nabla f(x_k + \alpha_k d_k)^T d_k \geq c_2 \nabla f_k^T d_k, \tag{1.5}$$

with $0 < c_1 < c_2 < 1$. Select a new symmetric positive definite matrix H_{k+1} which satisfy the following original quasi-Newton equation, [5]

$$H_{k+1} y_k = v_k \tag{1.6}$$

x_k and x_{k+1} two points are given; describe $g_k = \nabla f(x_k)$ and $g_{k+1} = \nabla f(x_{k+1})$, so $v_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$.

Quasi-Newton methods use the correction matrix Q_k rank one or rank two matrix. In each update, the iterate matrix $H_{k+1} = H_{k+1} + Q_k$,should satisfy the quasi-Newton condition (1.6) Now substitute the correction matrix by

$$Q_k = auu^T + bww^T, \tag{1.7}$$

where a and$b$are scalars while $u$and w are vectors.

The quantities auu^T and bww^T are symmetric matrices; when $b = 0$ quasi-Newton methods are using rank-one updates, but if $b \neq 0$ then quasi-Newton methods are using rank two updates.

The general type of QN updates which was proposed by Broyden [4] and satisfy the ordinary quasi-Newton equation is:

$$H_{k+1} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{v_k v_k^T}{v_k^T y_k} + \varphi_k (y_k^T H_k y_k) R_k R_k^T, \tag{1.8}$$

where $R_k = \frac{v_k}{v_k^T y_k} - \frac{H_k y_k}{y_k^T H_k y_k}$, and $\varphi_k = \varphi(\theta_k) = \frac{1-\theta_k}{1+\theta_k \delta_k \mu_k}$ with $\delta_k = \frac{y_k^T H_k y_k}{v_k y_k^T}$ and $\mu_k = \frac{v_k^T H_k v_k}{v_k^T y_k}$.

Several well known updates H_{k+1} defined in (1.8) by choosing different values of θ_k:- when; $\theta_k = \frac{v_k^T y_k}{v_k^T y_k - v_k^T H_k v_k}$, we get the symmetric rank-one formula(SR1):

$$H_{k+1}^{SR1} = H_k + \frac{(v_k - \ H_k y_k)(v_k - \ H_k y_k)^T}{(v_k - \ H_k y_k)^T y_k}. \tag{1.9}$$

For $\theta_k = 1$, we get the DFP formula due to Davidon, Fletcher, and Powell.

$$H_{k+1}^{DFP} = H_k - \frac{H_k y_k y_k^T H_k}{y_k^T H_k y_k} + \frac{v_k v_k^T}{v_k^T y_k}. \tag{1.10}$$

For $\theta_k = 0$, we get the BFGS formula due to Broyden, Fletcher, Goldfard, and Shanno.

$$H_{k+1}^{BFGS} = H_k + (1 + \frac{y_K^T H_k y_k}{v_k^T y_k})\frac{v_k v_k^T}{v_k^T y_k} - \frac{H_k y_k v_k^T + v_k y_k^T H_k}{y_K^T H_k y_k}. \tag{1.11}$$

In the next section, we propose a modified BFGS method and study the properties of it. we present a new update of H_{k+1} which satisfy the quasi-Newton condition.

2. A Modified BFGS Method (MBFGS)

In this section, a new class of quasi Newton updates for solving unconstrained non linear optimization problems is proposed. The idea of new updates is using the Powell equation [10] which is define as:

$$\widetilde{y}_k = (1-\theta) G v_k + \theta y_k \tag{2.1}$$

Where G is a hessian matrix which is a symmetric matrix of second partial derivatives of function and $\theta \ \ in(0,1)$. Now, we suppose that

$$G v_k = \frac{y_k}{\rho}. \tag{2.2}$$

Let, $\rho = \frac{2\sqrt{\omega}}{||v_k||}(1 + ||x_{k+1}||)$, ω is a machine accuracy, and $||.|| \geq 0$ is the Euclidean norm of vectors. so, we obtain

$$G v_k = ||v_k|| \frac{y_k}{2\sqrt{\omega}(1 + ||x_{k+1}||)}. \tag{2.3}$$

We replace $G v_k$ in (2.1) by (2.3), and getting the following

$$\widetilde{y}_k = (1-\theta)\Big(||v_k|| \frac{y_k}{2\sqrt{\omega}(1 + ||x_{k+1}||)}\Big) + \theta y_k. \tag{2.4}$$

For the new updated , we have investigated a new expression for the QN-condition via $\widetilde{y}_k$ put in (1.6) instead of y_k , and get

$$H_{k+1} \widetilde{y}_k = v_k. \tag{2.5}$$

A more flexible is gotten when the correction matrix $Q_k$is a rank two , hence the formula $H_{k+1} = H_k + Q$ can be written as,

$$H_{k+1} = H_k + auu^T + bww^T. \tag{2.6}$$

In 1970, Broyden, Fletcher, Goldfarb, and Shanno suggested an alternative method called the BFGS method, which is the most popular type of symmetric rank-two method for large-scale optimization and belongs to a group of quasi-Newton methods, It is a local search method. Now, we can drive a modified of H_{k+1}^{BFGS}depend on (2.5) to get H_{k+1}^{MBFGS} multiplying (2.6) by $\widetilde{y}_k$ to obtain

$$H_{k+1}\widetilde{y}_k = H_k\widetilde{y}_k + auu^T\widetilde{y}_k + bww^T\widetilde{y}_k. \tag{2.7}$$

The vectors u and v are no longer uniquely determined. In view of (2.7), it is adequate choose, $u = v_k$ and $w = H_k\widetilde{y}_k$. Then we obtain

$$H_{k+1}\widetilde{y}_k = H_k\widetilde{y}_k + av_kv_k^T\widetilde{y}_k + bH_k\widetilde{y}_k(H_ky_k)^T\widetilde{y}_k, \tag{2.8}$$

which implies, if $au^T\widetilde{y}_k = 1$ and $bw^T\widetilde{y}_k = -1$, thus determine a and b such that $a = \frac{1}{u^T\widetilde{y}_k} = \frac{1}{v_k^T\widetilde{y}_k}$ and $b = \frac{1}{\widetilde{y}_k^T H_k\widetilde{y}_k}$. Substituting the value of a, b, u and v to the updating formula (2.8), thus we get a new updated of QN-method is H_{k+1}^{MBFGS}

$$H_{k+1}^{MBFGS} = H_k + \left[1 + \frac{\widetilde{y}_k^T H_k\widetilde{y}_k}{v_k^T\widetilde{y}_k}\right]\frac{v_kv_k^T}{v_k^T\widetilde{y}_k} - \frac{H_k\widetilde{y}_kv_k^T + v_k\widetilde{y}_k^T H_k}{\widetilde{y}_k^T H_k\widetilde{y}_k}. \tag{2.9}$$

We can rewrite (2.9) as

$$H_{k+1}^{\mathrm{MBFGS}} = \left[I - \frac{v_k\widetilde{y}_k^T}{\widetilde{y}_k^T v_k}\right] H_k \left[I - \frac{\widetilde{y}_kv_k^T}{\widetilde{y}_k^T v_k}\right] + \frac{v_kv_k^T}{v_k^T\widetilde{y}_k}. \tag{2.10}$$

3. Algorithm of Modified BFGS

- **Step 0:** Start with initial point of solution $x_0 \in R^n$, $k = 0$, set $\epsilon > 0, n \in Z$,and select a real symmetric positive definite$H_0 = I$, I is an $n \times n$ identity matrix.
- **Step 1:** Test if $||g_k|| < \epsilon$ then stop, else $d_k = -H_kg_k = -H_k\nabla f(x_k)$ and go to step (2)
- **Step 2:** Using line search procedure to determine the size step $\alpha_k = argminf(x_k + \alpha_kd_k)$ such that rules (1.4) and (1.5) are satisfied
- **Step 3:** Calculate $x_{k+1} = x_k + \alpha_kd_k$.
- **Step 4:** check , if $||g_{k+1}|| < \epsilon$ then stop and x_{k+1} is optimal point.otherwise calculate $d_{k+1} = -H_{k+1}g_{k+1}$, H_{k+1} is defined in (2.10) and go to step (5).
- **Step 5:** set $k = k + 1$. and go to step 1.

4. Hereditary Property and Positive Definiteness of The MBFGS-Method

In this section, we prove that the new modification of BFGS is satisfied both properties the hereditary Property (secant condition) and preserves positive definite H_{k+1} matrices.

Theorem 4.1. *If the new method is applied to minimize a quadratic function with positive definite Hessian $G = G^T$, then the (1.6) is hold i.e $H_{k+1}^{MBFGS}\widetilde{y}_k = v_k$ for all $0 \leq k$.*

Proof. Multiply both sides of (2.10) by $\widetilde{y}_k$ from right , so we have

$$H_{k+1}^{\text{MBFGS}}\ \widetilde{y}_k = (\left[I - \frac{v_k\widetilde{y}_k^T}{\widetilde{y}_k^T v_k}\right] H_k \left[I - \frac{\widetilde{y}_k v_k^T}{\widetilde{y}_k^T v_k}\right] + \frac{v_k v_k^T}{v_k^T \widetilde{y}_k})\widetilde{y}_k, \tag{4.1}$$

by using a basic algebraic operations we found the (4.1) becomes in following form:

$$H_{k+1}^{\text{MBFGS}}\widetilde{y}_k = H_k\widetilde{y} - \frac{H_k\widetilde{y}_k v_k^T\widetilde{y}}{\widetilde{y}_k^T v_k} - \frac{v_k\widetilde{y}_k^T H_k\widetilde{y}_k}{\widetilde{y}_k^T v_k} + \frac{v_k\widetilde{y}_k^T H_k\widetilde{y} v_k^T\widetilde{y}_k}{(\widetilde{y}_k^T v_k)^2} + \frac{v_k v_k^T}{v_k^T\widetilde{y}_k}\widetilde{y}_k. \tag{4.2}$$

It is knowing that the $v_k^T\widetilde{y}_k$ is scalar and $\widetilde{y}_k^T v_k = v_k^T\widetilde{y}_k$, so (4.2) becomes

$$H_{k+1}^{MBFGS}\widetilde{y}_k = v_k. \tag{4.3}$$

□

Theorem 4.2. *We first demonstrate that if $H_k^{MBFGS}$is positive definite, then H_{k+1}^{MBFGS} is also positive definite.*

Proof. To ensure positive-definiteness ofH_{k+1}^{MBFGS} assumingH_k^{MBFGS} is positive definite. Typically the algorithm starts with $H_0^{MBFGS} = I$ or a similar diagonal positive-definite matrix. We only need to check that $w^T H_{k+1}^{MBFGS} w > 0$, for any $w \neq 0$ and$w \in R^n$, we have

$$w^T H_{k+1}^{\text{MBFGS}}\ w = w^T(\left[I - \frac{v_k\widetilde{y}_k^T}{\widetilde{y}_k^T v_k}\right] H_k \left[I - \frac{\widetilde{y}_k v_k^T}{\widetilde{y}_k^T v_k}\right] + \frac{v_k v_k^T}{v_k^T\widetilde{y}_k})w, \tag{4.4}$$

$$w^T H_{k+1}^{\text{MBFGS}}\ w = w^T \left[I - \frac{v_k\widetilde{y}_k^T}{\widetilde{y}_k^T v_k}\right] H_k \left[I - \frac{\widetilde{y}_k v_k^T}{\widetilde{y}_k^T v_k}\right] w + w^T\frac{v_k v_k^T}{v_k^T\widetilde{y}_k}w. \tag{4.5}$$

Let $z_k = w(I - \frac{v_k\widetilde{y}_k^T}{\widetilde{y}_k^T v_k})$ and $z_k \neq 0$, so rewrite (4.5) as:

$$w^T H_{k+1}^{\text{MBFGS}}\ w = z^T H_k z + \frac{(w^T v_k)^2}{v_k^T\widetilde{y}_k}. \tag{4.6}$$

It is clear the first term of (4.6) $z^T H_k z > 0$, because H_k is positive defined. The second $\frac{(w^T v_k)^2}{v_k^T\widetilde{y}_k}$, $(w^T v_k)^2 > 0$, now we need to prove $v_k^T\widetilde{y}_k > 0$, whenever

$$v_k^T\widetilde{y}_k = v_k^T((1-\theta)\Big(||v_k||\frac{y_k}{2\sqrt{\omega}(1+||x_{k+1}||)}\Big) + \theta y_k). \tag{4.7}$$

Let $\xi = \left(\frac{||v_k||}{2\sqrt{\omega}(1+||x_{k+1}||)}\right)$, and $\xi > 0$.
So,

$$v_k^T\widetilde{y}_k = v_k^T((1-\theta)(\xi y_k + \theta y_k)) \tag{4.8}$$

Suppose $\mu = (1-\theta)(\xi+\theta)$, $\mu > 0$, therefore

$$v_k^T\widetilde{y}_k = (\mu v_k^T y_k), \tag{4.9}$$

$$v_k^T \widetilde{y}_k = \mu(v_k^T g_{k+1} - v_k^T g_k). \tag{4.10}$$

In case of, exact line search then we have $v_k^T g_{k+1} = 0$, and $v_k^T g_k = -\alpha_k g_k^T H_k g_k$, then (4.10) becomes

$$v_k^T \widetilde{y}_k = \mu \alpha_k g_k^T H_k g_k. \tag{4.11}$$

Since H_k is positive, means $g_k^T H_k g_k > 0$. There fore (4.11) is positive. In case, inexact line search, $v_k^T g_{k+1} \neq 0$, we rewrite(4.10) as:

$$v_k^T \widetilde{y}_k = \mu v_k^T y_k = \mu \alpha_k d_k^T y_k. \tag{4.12}$$

Noteworthy that, from second Wolf's condition we get, $d_k^T y_k = d_k^T (g_{k+1} - g_k) > (\ c_2 - 1) d_k^T g_k$ and $(\ c_2 - 1)\ d_k^T g_k > 0$, so $d_k^T y_k > 0$, it is clear $\mu \alpha_k > 0$, thus, we see $v_k^T \widetilde{y}_k > 0$. Since

$$w^T H_{k+1}^{MBFGS} w > 0 \ w \neq 0, \tag{4.13}$$

therefor, H_{k+1}^{MBFGS} is positive definite. □

Theorem 4.3. *Let x_{k+1} andd_{k+1} are two sequences generated by new algorithm 3, with line search under Wolf's conditions (1.4) and (1.5), then the new direction d_{k+1} is satisfied the sufficient descent condition.*

$$d_{k+1}^T g_{k+1} \leq 0. \tag{4.14}$$

Proof. Form (1.2) and(2.10) we have,

$$d_{k+1} = -H_{k+1} g_{k+1}, \tag{4.15}$$

$$H_{k+1}^{\mathrm{MBFGS}} = \left[I - \frac{v_k \widetilde{y}_k^T}{\widetilde{y}_k^T v_k} \right] H_k \left[I - \frac{\widetilde{y}_k v_k^T}{\widetilde{y}_k^T v_k} \right] + \frac{v_k v_k^T}{v_k^T \widetilde{y}_k}. \tag{4.16}$$

Multiply both sides of (4.15) by g_{k+1}^T

$$g_{k+1}^T d_{k+1} = -g_{k+1}^T (\left[I - \frac{v_k \widetilde{y}_k^T}{\widetilde{y}_k^T v_k} \right] H_k \left[I - \frac{\widetilde{y}_k v_k^T}{\widetilde{y}_k^T v_k} \right] + \frac{v_k v_k^T}{v_k^T \widetilde{y}_k}) g_{k+1}. \tag{4.17}$$

It is clear $H_{k+1}^{MBFGS}$is positive defined and $g_{k+1}$is a vector, therefore

$$g_{k+1}^T (H_{k+1}) g_{k+1} \ > 0. \tag{4.18}$$

This implies

$$d_{k+1}^T g_{k+1} \leq 0. \tag{4.19}$$

□

5. New Preconditioned Conjugate Gradient PCG-Method

The Conjugate Gradient (CG) method is a attractive method for minimizing a large unconstrained nonlinear problems, because it is using the first derivative information to generate search directions, and the QN- method faster than CG- method but need more area computer store for the reason that the QN- methods generated a symmetric positive defined matrix in each iteration which needed a $(n(n+1)/2)$ location of store. So in 1978, Buckley suggested a method combining the conjugate gradient with QN-method called (PCG-method)the aim of this suggestion is accelerating the convergence of conjugate gradient and reduce amount of storage in QN-method. The idea of PCG-method is based on combining by using the matrix of QN in the conjugate gradient algorithm which is corresponding to solve a problem in the transformed space.

5.1. **A New PCG Method:** let H_{k+1}^{MBFGS} is a preconditioned matrix and it is a symmetric positive definite, by using Cholesky decomposition of H_{k+1}^{MBFGS}, i.e there exists a lower triangular matrix $\widetilde{L}$ such that $H_{k+1}^{MBFGS} = \widetilde{L}\widetilde{L}^T$. Assume $f(x)$ be a strictly convex quadratic function and $f(x)$ can be written as:

$$f(x) = \frac{1}{2}x^T Gx + x^T b + c, \tag{5.1}$$

such that, gradient of $f(x)$ is

$$\nabla f(x) = g(x) = Gx + b, \tag{5.2}$$

$$f(\widetilde{L}z) = \frac{1}{2}(\widetilde{L}z)^T G(\widetilde{L}z) + (\widetilde{L}z)^T b + c. \tag{5.3}$$

Let $f(\widetilde{L}z) = h(z)$, so,

$$h(z) = \frac{1}{2}(\widetilde{L}z)^T G(\widetilde{L}z) + (\widetilde{L}z)^T b + c. \tag{5.4}$$

The first derivative of $h(z)$ is

$$\nabla h(z) = \widetilde{L}z^T G\widetilde{L}z + \widetilde{L}^T b, \tag{5.5}$$

$$\nabla h(z) = \widetilde{L}^T (G\widetilde{L}z + b), \tag{5.6}$$

$$g^z = \widetilde{L}^T g^x. \tag{5.7}$$

Now, we set

$$z_{k+1} = z_k + \alpha_k d_k^z. \tag{5.8}$$

Multiplication both sides of(5.8) by $\widetilde{L}$, we get

$$\widetilde{L}z_{k+1} = \widetilde{L}z_k + \alpha_k \widetilde{L}d_k^z. \tag{5.9}$$

We have $x = \widetilde{L}z$, so (5.9) becomes:

$$x_{k+1} = x_k + \alpha_k \widetilde{L} d_k^z. \tag{5.10}$$

From(5.10), we noted $\widetilde{L} d_k^z = d_k^x$, since $d_k^z = \widetilde{L}^{-1} d_k^x$.

Set

$$y_k^z = g_k^{z+1} - g_k^z, \tag{5.11}$$

where, y_k^{z+1} and y_k^z are the gradients of $h(z)$ at points z_{k+1} and z_k respectively, since from (5.7), (5.11) becomes as follows:

$$y_k^z = \widetilde{L}^T g_k^{k+1} - \widetilde{L}^T g_k^x. \tag{5.12}$$

Now consider applying the modification of Parry conjugate gradient method $\beta_k^{MPrrey} = \frac{g_{k+1}^T(\tilde{y}_k - v_k)}{d_k^T \tilde{y}_k}$ [9], to the objective function $h(z)$,

$$d_{k+1}^z = -g_{k+1}^z + \frac{g_{k+1}^{z^T}(\tilde{y}_k^z - v_k^z)}{d_k^{z^T} \tilde{y}_k^z} d_k^z. \tag{5.13}$$

Using (5.7), (5.11) and (5.12) in (5.13) and multiply by $\widetilde{L}$, we get:

$$\widetilde{L}\widetilde{L}^{-1} d_{k+1}^x = -\widetilde{L}\widetilde{L}^T g_{k+1}^x + \frac{g_{k+1}^{x^T} \widetilde{L}\widetilde{L}^T (\tilde{y}_k^x - v_k^x)}{d_k^{x^T} \tilde{y}^x} \widetilde{L}\widetilde{L}^{-1} d_k^x, \tag{5.14}$$

$$d_{k+1}^x = -H_{k+1}^{MBFGS} g_{k+1}^x + \frac{g_{k+1}^{x^T} H_{k+1}^{MBFGS} (\tilde{y}_k^x - v_k^x)}{d_k^{x^T} \tilde{y}^x} d_k^x. \tag{5.15}$$

(5.15) is our preconditioned conjugate method which is require less storage and computation time and has a quadratic termination property.

5.2. **Algorithm of the New PCG-Method.**

- **Step 0:** let $k = 0$, $x_0 \ in \ R^n$ is an initial point of solution, set $\epsilon > 0, n \in Z$,and select a real symmetric positive definite matrix $H_0 = I$, I is an $n \times n$ identity matrix and ω is accuracy of computer.
- **Step 1:** test a criterion for stopping, if $||g_k|| < \epsilon$ then stop else go to step 2.
- **Step 2:** $d_k = -H_k \nabla f(x_k) = -H_k g_k$ and continuous.
- **Step 3:**using line search procedure to determine the size stepα_k , $\alpha_k = argminf\,(x_k + \alpha_k d_k)$ such that rules (1.4) and (1.5) are satisfied.
- **Step 4:** calculate $x_{k+1} = x_k + \alpha_k d_k$,and go to next step.
- **Step 5:** check, if $||g_{k+1}|| < \epsilon$ then stop and x_{k+1} is optimal point, otherwise calculate $v_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$ and find $\tilde{y}$ by $\tilde{y} = (1-\theta)\, \|v_k\| \, \frac{y_k}{2\sqrt{\omega}(1+\|x_{k+1}\|)} + \theta y_k$, $\theta \in (0,1)$, ω is error of machine and go.
- **Step 6:** find $d_{k+1} = -H_{k+1}^{MBFGS} g_{k+1} + \frac{g_{k+1}^T H_{k+1}^{MBFGS}(\tilde{y}_k - v_k)}{d_k^T \tilde{y}} d_k$, and H_{k+1}^{MBFGS} is defined as $H_{k+1}^{MBFGS} = H_k + \left[1 + \frac{\tilde{y}_k^T H_k \tilde{y}_k}{v_k^T \tilde{y}_k}\right] \frac{v_k v_k^T}{v_k^T \tilde{y}_k} - \frac{H_k \tilde{y}_k v_k^T + v_k \tilde{y}_k^T H_k}{\tilde{y}_k^T H_k \tilde{y}_k}$ then go to step (7).

- **Step 7:** If $g_{k+1}^T g_{k+1} \leq -0.8 d_{k+1}^T g_{k+1}$, then go to step 2.
  else $k = k+1$ and go to step 3.

Theorem 5.1. *Let the sequences of x_k and d_k are generated by algorithm of the New PCG-Method 5.2 then the descent property of a new PCG- method is descent condition:*

$$d_{k+1}^T g_{k+1} < 0. \tag{5.16}$$

Proof. We prove by induction, at $k = 0$, $d_0 = -H_0 g_0$, so we have

$$d_0^T g_0 \leq -g_0^T H_0 g_0, \tag{5.17}$$

where $H_0 = I$, $g_0 \neq 0$, and $-g_0^T H_0 g_0 < 0$.

Now we assume that the conclusion (5.16) holds for $k \geq 0$, means, $g_k^T d_k \leq \gamma \|g_k\|^2$, need to prove it is true at $k+1$.

Let

$$d_{k+1} = -H_{k+1}^{MBFGS} g_{k+1} + \frac{g_{k+1}^T H_{k+1}^{MBFGS}(\widetilde{y}_{k\,k} - v_k)}{d_k^T \widetilde{y}_k} d_k. \tag{5.18}$$

Multiply both sides of (5.18) by g_{k+1} we get,

$$d_{k+1}^T g_{k+1} = -g_{k+1}^T H_{k+1}^{MBFGS} g_{k+1} + \frac{g_{k+1}^T H_{k+1}^{MBFGS}(\widetilde{y}_k - v_k)}{d_k^T \widetilde{y}_k} d_k^T g_{k+1}. \tag{5.19}$$

Thus,

$$d_{k+1}^T g_{k+1} = -g_{k+1}^T H_{k+1}^{MBFGS} g_{k+1} + \frac{g_{k+1}^T H_{k+1}^{MBFGS}(\widetilde{y}_k)}{d_k^T \widetilde{y}_k} d_k^T g_{k+1} - \frac{g_{k+1}^T H_{k+1}^{MBFGS}(v_k)}{d_k^T \widetilde{y}_k} d_k^T g_{k+1}. \tag{5.20}$$

We notice that if we use an exact line search then, we have $d_k^T g_{k+1} = 0$ and also H_{k+1}^{MBFGS} is positive symmetric definite from theorem 4.2, $g_{k+1}^T H_{k+1}^{MBFGS} g_{k+1} \geq 0$ for all, $g_{k+1} \neq 0$, therefore $d_{k+1}^T g_{k+1} < 0$. In case inexact line search $d_k^T g_{k+1} \neq 0$, from (5.19) , we get

$$d_{k+1}^T g_{k+1} = -\frac{g_{k+1}^T H_{k+1}^{MBFGS} v_k}{d_k^T \widetilde{y}} d_k^T g_{k+1}, \tag{5.21}$$

$\widetilde{y} = (1-\theta)\|v_k\| \frac{y_k}{2\sqrt{\omega}(1+\|x_{k+1}\|)} + \theta y_k$.

We need to show $d_k^T \widetilde{y} > 0$, means $d_k^T \widetilde{y} = d_k^T((1-\theta)\|v_k\| \frac{y_k}{2\sqrt{\omega}(1+\|x_{k+1}\|)} + \theta y_k) > 0$. It is noted that the $d_k^T y_k = d_k^T(g_{k+1} - g_k) > (\delta_2 - 1) d_k^T g_k$,$d_k^T v_k = \|d_k\|^2 \alpha_k$, and $\theta \in (0,1)$

$$d_{k+1}^T g_{k+1} = -\frac{g_{k+1}^T H_{k+1}^{MBFGS} g_{k+1} \|d_k\|^2 \alpha_k}{d_k^T \widetilde{y}}. \tag{5.22}$$

Let $\tau = \frac{g_{k+1}^T H_{k+1}^{MBFGS} g_{k+1} \|d_k\|^2 \alpha_k}{d_k^T \widetilde{y}}$, we see τ is positive, then (5.22) becomes:

$$d_{k+1}^T g_{k+1} < 0. \tag{5.23}$$

□

6. Numerical Experiments and Discussions

It is clearly that the theoretical evidence is not sufficient to demonstrate the effectiveness or robustness of any iterative methods. Therefore, researchers turn to study the numerical results of methods by evaluate the performance method on a group of test problems and evaluation the number of iterations or Computation time (CPU-time).

In this section, we present the results of numerical experiments for our new suggestion to solve different nonlinear test problems of large size. In practice, the construction sequence of preconditions is based on well-known suggestion method modified BFGS techniques in order to keep under control the amount of memory. We use FORTRAN95 LANGUAGE to write all codes and the run is stopping when this inequality $||g_{k+1}|| < 10^{-5}$ is satisfied. For compare, we used the well-known nonlinear problems with dimension ranging between 4 to 5000, [1]. All algorithms use exactly the same method (cubic fit method) to find the step length α_k the same implementation of the Wolfe line search conditions (1.4) and (1.5) with $c_1 = 0.001$ and$c_2 = 0.1$.

According to the Table1, it is not difficult to show that the performance of new PCG -method is better than stander PCG method when using Hestain and Stiefen formula ($\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$) [12], and the restart $g_{k+1}^T g_{k+1} \leq -0.8 d_{k+1}^T g_{k+1}$.

Table 2 results illustrating the behavior of new PCG -method and standard PCG methods when taken the Perry suggestion $\beta_k^{perry} = \frac{g_{k+1}^T (y_k - v_k)}{d_k^T y_k}$)for coefficient of conjugate gradient method [9] under the restart $|g_{k+1}^T g_k| > 0.2 g_{k+1}^T g_{k+1}$, for more analyse of the numerical result we use performance profile proposed by Dolan and More [14].

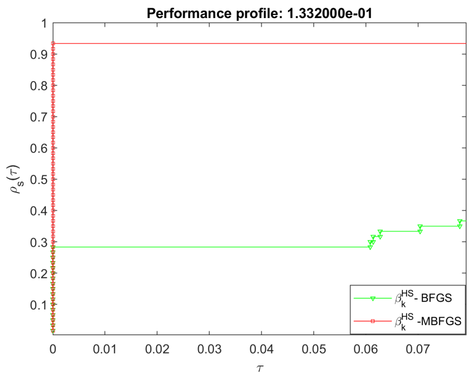
According to the rule of this performance profile, we describe the performance curves based on Table 1 and Table 2 as in Figures 1–4. Based on the four figures, we see that the new PCG method is superior to the standard PCG method under the unconstrained problems in Tables 1 and 2.

Table 1. Comparing performance of $\beta_k^{HS} - BFGS$ and $\beta_k^{HS} - MBFGS$

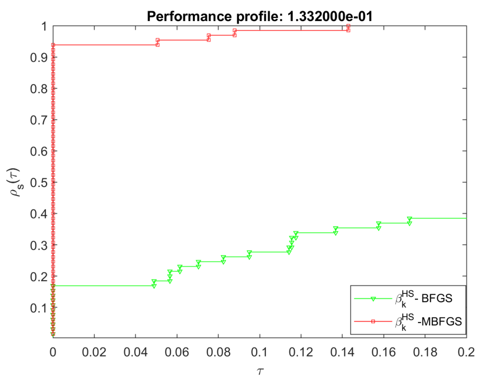
		$\beta_k^{HS} - BFGS$	$\beta_k^{HS} - MBFGS$
Test Function	N	NOI-NOF	NOI-NOF
Powell (-3,-1,0,1;...)	4	21-86	20-72
	100	67-179	39-109
	500	48-142	44-121
	1000	47-142	45-126
	3000	51-144	37-123
	5000	40-119	40-119
Miele (1,2,2,2;...)	4	25-87	18-68
	100	31-99	18-71
	500	31-103	22-86
	1000	38-117	23-88
	3000	34-106	23-91
	5000	35-104	29-89
Cantral (1,2,2,2;...)	4	36-269	12-79
	100	42-341	23-167
	500	48-416	36-311
	1000	51-451	26-217
	3000	55-506	41-404
	5000	57-532	32-281
Wolf (-1;...)	4	7-18	8-17
	100	72-145	58-117
	500	82-165	63-127
	1000	96-194	66-133
	3000	181-388	108-227
	5000	182-382	107-236
cubic (-1.2,1;...)	4	19-58	18-52
	100	70-167	36-91
	500	53-124	42-101
	1000	71-167	48-112
	3000	71-168	49-177
	5000	67-163	55-127
NON-DIAGONAL (-1;...)	4	24-73	23-61
	100	74-177	56-134
	500	82-205	63-153
	1000	85-218	61-147
	3000	111-335	69-174
	5000	100-275	77-194
Shallow (-2,-2;...)	4	8-26	8-24
	100	8-26	8-24
	500	8-26	8-25
	1000	8-26	8-25
	3000	9-28	10-29
	5000	10-30	10-29
Rosen (-1.2,1;...)	4	32-92	34-85
	100	235-6827	220-538
	500	578-1537	462-1133
	1000	864-2150	747-2014
	3000	1061-2667	910-2434
	5000	1428-3577	891-2357
Beal (0,0;..)	4	9-22	9-22
	100	10-25	10-25
	500	10-25	10-25
	1000	10-25	10-25
	3000	10-25	10-25
	5000	10-25	10-25
Dixon (-1;..)	4	9-24	9-23
	100	218-537	209-495
	500	210-509	193-541
	1000	199-490	225-541
	3000	252-590	195-461
	5000	204-529	175-423

Table 2. Comparing performance profiles of $\{\beta_k^{perry} - MBFGS\}$ and $\{\beta_k^{perry} - BFGS\}$

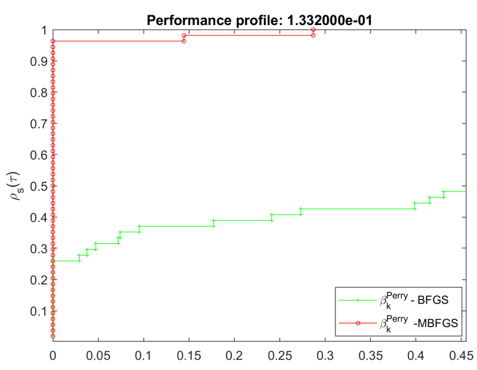
		$\beta_k^{perry} - BFGS$	$\beta_k^{perry} - MBFGS$
Test Function	N	NOI-NOF	NOI-NOF
Powell (-3,-1,0,1;...)	4	124-294	36-102
	100	529-1152	40-121
	500	119-285	40-121
	1000	289-1684	40-121
	3000	152-379	40-121
	5000	150-381	40-121
Cantral (1,2,2,2;...)	4	36-1515	19-85
	100	274-2216	23-137
	500	365-2657	24-152
	1000	564-2854	24-152
	3000	610-3200	28-213
	5000	330-3350	32-281
OSP (-1;...)	4	14-59	8-45
	100	157-519	52-190
	500	417-1170	115-352
	1000	523-1421	174-532
	3000	1006-2692	303-965
	5000	1356-3637	380-1221
Wood (-3,-1,-3,-1;...)	4	26-61	22-91
	100	26-61	23-55
	500	36-81	23-55
	1000	36-81	23-55
	3000	36-81	23-55
	5000	36-81	23-55
NON-DIAGONAL (-1;...)	4	31-80	30-75
	100	47-113	44-107
	500	49-119	49-112
	1000	50-122	61-120
	3000	50-122	49-119
	5000	50-122	50-120
Rosen (-1;...)	4	39-104	39-102
	100	41-109	39-104
	500	38-103	38-103
	1000	39-105	38-103
	3000	40-105	38-105
	5000	40-104	40-103
Sum (2;...)	4	3-11	3-11
	100	20-107	14-81
	500	19- 92	21-115
	1000	31-172	23-117
	3000	63-351	32-179
	5000	79-425	42-222
cubic (-1.2,1;...)	4	22-60	14-46
	100	29-76	16-51
	500	29-75	22-62
	1000	31-82	22-64
	3000	29-74	24-68
	5000	32-83	24-69
Edger (-1;..)	4	5-14	5-14
	100	6-16	6-16
	500	6-16	6-16
	1000	6-16	6-16
	3000	6-16	6-16
	5000	6-16	6-16



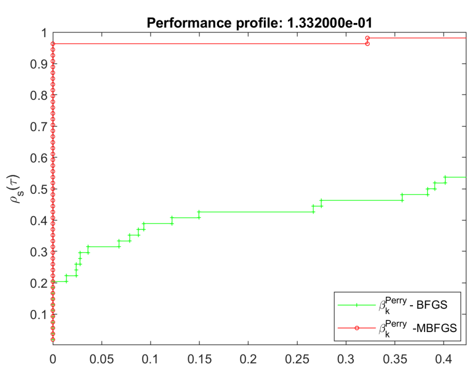
(Figure1: performance profiles of two methods based on (NOI))



(Figure2: Performance profiles of two methods based on(NOF&∇F))



(Figure3: performance profiles of two methods based on (NOI))



(Figure4: Performance profiles of two methods based on(NOf & ∇ f))

7. Conclusion

The nonlinear quasi-Newton method is widely used in unconstrained optimization. In this paper, we suggest new updates to the quasi-Newton method for solving unconstrained optimization problems. We use this new quasi-Newton to introduce the new PCG method. The analysis and implementation of the descent property with the Wolfe line search of the modified method are studied. The numerical results show that the proposed formula for the combined quasi-Newton conjugate gradient method is very encouraging for general, unconstrained optimizations.

Conflicts of Interest: The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] N. Andrei, An Unconstrained Optimization Test Functions Collection, Adv. Model. Optim. 10 (2008), 147-161.
[2] M. Al-Baali, Descent Property and Global Convergence of the Fletcher-Reeves Method With Inexact Line Search, IMA J. Numer. Anal. 5 (1985), 121-124. https://doi.org/10.1093/imanum/5.1.121.
[3] C.g. Broyden, the Convergence of a Class of Double-Rank Minimization Algorithms 1. General Considerations, IMA J. Appl. Math. 6 (1970), 76-90. https://doi.org/10.1093/imamat/6.1.76.
[4] C.G. Broyden, J.E. Dennis Jr., J.J. More, On the Local and Superlinear Convergence of Quasi-Newton Methods, IMA J. Appl. Math. 12 (1973), 223-245. https://doi.org/10.1093/imamat/12.3.223.
[5] W.C. Davidon, Variable-Metric Method for Minimization, Technical Report, ANL-5990, (1959). https://doi.org/10.2172/4252678.
[6] R. Fletcher, Practical Methods of Optimization, John Wiley & Sons, (2013).
[7] R. Fletcher, A New Approach to Variable Metric Algorithms, Computer J. 13 (1970), 317-322. https://doi.org/10.1093/comjnl/13.3.317.
[8] J.D. Pearson, Variable Metric Methods of Minimisation, Computer J. 12 (1969), 171-178. https://doi.org/10.1093/comjnl/12.2.171.
[9] M.Sh. Taher, S.G. Shareef, A Modified Perry's Conjugate Gradient Method Based on Powell's Equation for Solving Large-Scale Unconstrained Optimization, Math. Stat. 9 (2021), 882-888. https://doi.org/10.13189/ms.2021.090603.
[10] M.J.D. Powell, A Fast Algorithm for Nonlinearly Constrained Optimization Calculations, in: G.A. Watson (Ed.), Numerical Analysis, Springer Berlin Heidelberg, Berlin, Heidelberg, 1978: pp. 144–157. https://doi.org/10.1007/BFb0067703.
[11] P. Wolfe, Convergence Conditions for Ascent Methods, SIAM Rev. 11 (1969), 226-235. https://doi.org/10.1137/1011036.
[12] M.R. Hestenes, E. Stiefel, Methods of Conjugate Gradients for Solving Linear Systems, J. Res. Nat. Bureau Standards. 49 (1952), 409-436.
[13] G. Zoutendijk, Nonlinear Programming Computational Methods, Integer Nonlinear program. (1970), 37-86. https://cir.nii.ac.jp/crid/1571980075701600256.
[14] E.D. Dolan, J.J. More, Benchmarking Optimization Software With Performance Profiles, Math. Program. 91 (2002), 201-213. https://doi.org/10.1007/s101070100263.