Journal of Education and Science
Vol. 27, No.2, 2018, pp. 58-70
ISSN 1812-125X
http://edusj.mosuljournals.com

Development a Hybrid Conjugate Gradient Algorithm for Solving Unconstrained Minimization Problems

*** Sawsan S. Ismael** ****Basim A. Hassan**

** College of Education University of mosul*
**Department of Mathematics, College of Computers Sciences and Mathematics University of Mosul, IRAQ*

تاريخ القبول 2013/01/09 *تاريخ الاستلام 2012/09/18*

Abstract

In this paper, a new hybrid nonlinear conjugate gradient method are presented, which produce sufficient descent search direction at every iteration. This methods showed globally convergent under some assumptions. The numerical results show that all this new hybrid method are efficient for the given test problems.

الملخص

في هذا البحث تم اقتراح خوارزمية من خوارزميات التدرج المترافق الهجينية. وقد أثبتت الطريقة إن لها اتجاه بحث ذو انحدار كافي عند كل تكرار . وقد أظهرت هذه الطريقة تقارب شامل تحت بعض الفرضـــيات. وأظهرت النتائج العددية لهذه الطريقة كفاءتها وذلك بتطبيقها على دوال اختبار معطاة.

Introduction

Let us consider the unconstrained optimization problem

$$\min\{f(x) \mid x \in R^n\} \qquad ....(1)$$

where $f : R^n \to R$ is a continuously differentiable function, bounded from below. For solving this problem, starting from an initial guess $x_0 \in R^n$, a nonlinear conjugate gradient method, generates a sequence $\{x_k\}$ as :

$$x_{k+1} = x_k + \alpha_k d_k \quad \text{..........(2)}$$

where α_k is the step-size, and the direction d_k are generated as

$$d_{k+1} = -g_{k+1} + \beta_k d_k, \quad d_0 = -g_0 \quad \text{..........(3)}$$

where β_k is known as the conjugate gradient parameter, $v_k = x_{k+1} - x_k$ and $g_k = \nabla f(x_k)$ **[1]**. The step size α_k is chosen in such a way that $\alpha_k > 0$ and satisfies the strong Wolfe (SW) conditions

$$f(x_k + \alpha_k d_k) \le f(x_k) + \delta_1 \alpha_k d_k^T g_k \quad \text{..........(4)}$$

$$\left| g(x_k + \alpha_k d_k)^T d_k \right| \le -\delta_2 d_k^T g_k \quad \text{..........(5)}$$

with $0 < \delta_1 < \delta_2 < 1$, where $f_k = f(x_k)$, $g_k = g(x_k)$, g_k are the gradient of f evaluated at the current iterate x_k **[7]**. Where d_k is a descent direction. Different conjugate gradient algorithms correspond to different choices for the parameter β_k. For example Fletcher and Reeves (FR) **[**6**]**, Dai and Yuan (DY) **[**4**]** and Conjugate Descent (CD) **[**5**]** :

$$\beta_k^{FR} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k} \;, \beta_k^{DY} = \frac{g_{k+1}^T g_{k+1}}{y_k^T d_k} \;, \beta_k^{CD} = \frac{g_{k+1}^T g_{k+1}}{\left| g_k^T d_k \right|} \;, \quad \text{..........(6)}$$

They have strong convergence properties, but they may have modest practical performance due to jamming. On the other hand, the methods of Polak and Ribiere (PR) **[**9**]**, Hestenes and Stiefel (HS) **[**7**]**, or Liu and Storey, (LS) **[**8**]** :

$$\beta_k^{PR} = \frac{g_{k+1}^T y_k}{g_k^T g_k} \;, \beta_k^{HS} = \frac{g_{k+1}^T y_k}{y_k^T d_k} \;, \beta_k^{LS} = \frac{g_{k+1}^T y_k}{\left| g_k^T d_k \right|} \;, \quad \text{..........(7)}$$

in general, may not be convergent, but they often have better computational performances.

Also, under mild assumptions on the objective function, DY method is shown to be globally convergent under a variety of line search conditions. These advantages motivated us to study the hybridizations of HS and DY methods following the effective approach proposed in **[2,3** and **11]**. The formula β_k in **[2,3],** namely β_k^C, is obtained by a convex combination of β_k^{HS} and β_k^{DY}. That is,

$$\beta_k^C = (1-\theta_k)\beta_k^{HS} + \theta_k \beta_k^{DY} = (1-\theta_k)\frac{g_{k+1}^T y_k}{y_k^T d_k} + \theta_k \frac{g_{k+1}^T g_{k+1}}{y_k^T d_k} \quad \text{..........(8)}$$

where θ_k , namely the hybridization parameter, is a scalar parameter satisfying $0 \le \theta_k \le 1$. Therefore, Substituting (8) into (3), we get :

$$d_{k+1} = -g_{k+1} + (1-\theta_k)\frac{g_{k+1}^T y_k}{y_k^T d_k} d_k + \theta_k \frac{g_{k+1}^T g_{k+1}}{y_k^T d_k} d_k, \qquad ....(9)$$

Or equivalently,

$$d_{k+1} = -g_{k+1} + (1-\theta_k)\frac{g_{k+1}^T y_k}{y_k^T v_k} v_k + \theta_k \frac{g_{k+1}^T g_{k+1}}{y_k^T v_k} v_k. \qquad ....(10)$$

As known, if the point x_{k+1} is close enough to a local minimizer x^*, then a good direction to follow is the Newton direction, that is,

$$d_{k+1} = -G_{k+1}^{-1} g_{k+1}. \qquad ....(11)$$

Motivated by this, Andrei [2,3] rewrite (10) as follows :

$$-G_{k+1}^{-1} g_{k+1} = -g_{k+1} + (1-\theta_k)\frac{g_{k+1}^T y_k}{y_k^T v_k} v_k + \theta_k \frac{g_{k+1}^T g_{k+1}}{y_k^T v_k} v_k. \qquad ....(12)$$

After some algebraic manipulations one obtains :

$$\theta_k = \frac{v_k^T G_{k+1} g_{k+1} - v_k^T g_{k+1} - \dfrac{g_{k+1}^T y_k}{y_k^T v_k} v_k^T G_{k+1} v_k}{\dfrac{g_{k+1}^T g_k}{y_k^T v_k} v_k^T G_{k+1} v_k}. \qquad ....(13)$$

In quasi-Newton methods, an approximation matrix B_k for the Hessian G_k is used and updated so that the new matrix B_{k+1} satisfies a version of the secant equation. In [2], B_{k+1} is determined to satisfy the standard secant equation, that is, $B_{k+1} v_k = y_k$. Therefore, θ_k is computed by :

$$\theta_k = \frac{v_k^T g_{k+1}}{g_{k+1}^T g_k}. \qquad ....(14)$$

In [3], B_{k+1} is determined to satisfy the modified secant equation proposed by Li et al. [10],

$$B_{k+1} v_k = y_k + \eta_k v_k / \|v_k\|^2, \qquad ....(15)$$

where

$$\eta_k = 2(f_k - f_{k+1}) + (g_{k+1} + g_k)^T v_k, \qquad ....(16)$$

and so θ_k is computed by

$$\theta_k = \frac{\left[\frac{\eta_k}{\|v_k\|} - 1\right] v_k^T g_{k+1} - \frac{g_{k+1}^T y_k}{y_k^T d_k}\eta_k}{g_{k+1}^T g_k + \frac{g_{k+1}^T g_k}{y_k^T v_k}\eta_k}. \qquad \text{..........(17)}$$

Now, using (17) in (10) we get :

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{y_k^T v_k + \eta_k} v_k - \left[1 - \frac{\eta_k}{\|v_k\|^2}\right] \frac{v_k^T g_{k+1}}{y_k^T v_k + \eta_k} v_k . \qquad \text{..........(18)}$$

However, using exact line searches ($v_k^T g_{k+1} = 0$) in (18), the direction d_{k+1} reduced to

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{y_k^T v_k + \eta_k} v_k . \qquad \text{..........(19)}$$

It was shown in [3] that the hybrid CG method with θ_k as in (17) incorporated with an acceleration scheme is more efficient than the HS and DY method, and the hybrid CG methods proposed Andrei [2].

The structure of the paper is as follows. In section 2, we present the new hybrid conjugate gradient algorithm. Section 3 presents a new Algorithm and Convergence analysis. Section 4 numerical results are presented and In section 5 discuss the we give brief conclusions and discussions.

2. A new hybrid conjugate gradient algorithm

We develop the secant equation based on the modified BFGS method proposed by Li et al. [10]. For this purpose, in order to unify both approaches, we consider a slight modification of the modified secant condition (15) as

$B_{k+1} v_k = z_k$ where $z_k = u_k y_k + (1-u_k)\eta_k v_k / \|v_k\|^2$, ...(20)

This leads us to development a hybrid conjugate gradient algorithm (10) where

$$\theta_k = \frac{\left[\frac{\eta_k}{\|v_k\|} - 1\right] v_k^T g_{k+1} - \frac{g_{k+1}^T y_k}{u_k y_k^T v_k}(1-u_k)\eta_k}{g_{k+1}^T g_k + \frac{g_{k+1}^T g_k}{u_k y_k^T v_k}(1-u_k)\eta_k}. \qquad \text{..........(21)}$$

Now, using (21) in (10) we get :

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} v_k - \left[1 - \frac{\eta_k}{\|v_k\|^2}\right] \frac{v_k^T g_{k+1}}{u_k y_k^T v_k + (1-u_k)\eta_k} v_k . \qquad ....(22)$$

Using exact line searches ($v_k^T g_{k+1} = 0$) in (22), the direction d_{k+1} defined in (22) reduced to

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} v_k . \qquad ....(23)$$

Where $u \in [0\ ,\ 1]$ is a constant. Our motivation to get a good algorithm for solving (1) is to choose the parameter u in (12) in such a way so that for every $k \geq 1$ the direction d_{k+1} given by (23) is the Newton direction. This is motivated by the fact that when the initial point x_0 is near the solution of (1) and the Hessian is a nonsingular matrix then the Newton direction is the best line search direction. Therefore, from the equation

$$-G^{-1} g_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} v_k \ . \qquad ....(24)$$

Multiplying (24) by y_k^T, we have

$$-G^{-1} y_k^T g_{k+1} = -y_k^T g_{k+1} + \frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} y_k^T v_k \qquad ....(25)$$

Since $G^{-1} y_k = v_k$ then we have

$$-v_k^T g_{k+1} = -y_k^T g_{k+1} + \frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} y_k^T v_k \qquad ....(26)$$

from (26) we get :

$$\frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} = \frac{-v_k^T g_{k+1} + y_k^T g_{k+1}}{y_k^T v_k} \qquad .....(27)$$

$$(y_k^T g_{k+1}) y_k^T v_k = (-v_k^T g_{k+1} + y_k^T g_{k+1}) \quad ((1-u_k)\eta_k + u_k\ y_k^T v_k)$$
$$(y_k^T g_{k+1}) y_k^T v = (-v_k^T g_{k+1})((1-u_k)\eta_k) + (-v_k^T g_{k+1})(u_k y_k^T v_k) + (y_k^T g_{k+1})((1-u_k)\eta_k) + (y_k^T g_{k+1})(u_k y_k^T v_k) \qquad .(28)$$

$$(y_k^T g_{k+1}) y_k^T v = \eta_k(-v_k^T g_{k+1}) - u_k \eta_k(-v_k^T g_{k+1}) +$$
$$(-v_k^T g_{k+1})(u_k y_k^T v_k) + \eta_k(y_k^T g_{k+1}) - u_k \eta_k(y_k^T g_{k+1}) + (y_k^T g_{k+1})(u_k y_k^T v_k) \qquad ......(29)$$

$$(y_k^T g_{k+1}) y_k^T v + \eta_k(v_k^T g_{k+1}) - \eta_k(y_k^T g_{k+1}) =$$
$$u_k[-\eta_k(-v_k^T g_{k+1}) + (-v_k^T g_{k+1})(y_k^T v_k) - \eta_k(y_k^T g_{k+1}) + (y_k^T g_{k+1})(y_k^T v_k)] \qquad ......(30)$$

and from (30) we get :

$$u_k = \frac{(y_k^T g_{k+1}) y_k^T v + \eta_k(v_k^T g_{k+1}) - \eta_k(y_k^T g_{k+1})}{-\eta_k(-v_k^T g_{k+1}) + (-v_k^T g_{k+1})(y_k^T v_k) - \eta_k(y_k^T g_{k+1}) + (y_k^T g_{k+1})(y_k^T v_k)} \qquad ....(31)$$

3. New Algorithm and Convergence

We analyze the convergence property of the hybrid CG-method using our newly proposed formal as in (23). Throughout this section, we assume $g_{k+1} \neq 0$, for $k \geq 1$, otherwise, a stationary point is at hand. We make the following basic assumptions on the objective function.

Assumptions

i- The level set $l = \{x \in R^n | f(x) \leq f(x_0)\}$ is bounded, there exists a constant $B > 0$ such that

$$\|x\| \leq B, \ \forall x \in l . \qquad ....(32)$$

ii- In some neighborhood U of L, $f(x)$ is continuously differentiable and its gradient is Lipschitz continuous, namely, there exists a constant $\mu > 0$ such that

$$\|g(x_{k+1}) - g(x_k)\| \leq L\|x_{k+1} - x_k\|, \ \forall x_{k+1}, x_k \in U . \qquad ....(33)$$

3.1. The Algorithm has the Following Steps :

Step 0 : Given parameters $\varepsilon = 1*10^{-5}$, $\delta_1 \in (0,1)$, $\delta_2 \in (0,1/2)$

choose initial point $x_0 \in R^n$, $k = 1$, $d_k = -g_k$.

Step 1 : Computing g_k ; if $\|g_k\| \leq \varepsilon$ then stop ; else continue .

Step 2 : Set $x_{k+1} = x_k + \alpha_k d_k$, (Use strong Wolfe line search technique to compute the parameter α_k).

Step 3 : Compute u_k is defined by (31). If $u_k < 0$ then $u_k = 0$ and if $u_k > 1$ then $u_k = 1$

Step 4 : Set $\beta_k = \dfrac{g_{k+1}^T y_k}{u_k y_k^T d_k + (1 - u_k)\eta_k}$.

Step 5 : Compute $d_{k+1} = -g_{k+1} + \beta_k d_k$,

Step 6 : If $k = n$ go to **Step 1** with new values of x_{k+1} and g_{k+1}.

If not continue.

Theorem (3.1)

Assume that f is a convex function and α_k in algorithm (2) and (23), where θ_k is given by (31), is determined by the strong Wolfe conditions $(4)-(5)$. If $0<\theta_k<1$ then the direction d_{k+1} given by (23) is a sufficient descent direction**.**

Proof.

Since $d_0=-g_0$, we have $g_0^T d_0 \le -\|g_0\|^2<0$. Assume by induction that

$$g_k^T d_k \le -c\|g_k\|^2<0 \text{ where } 0<c\le 1 \qquad \text{..........}(33)$$

which is a sufficient descent direction. To complete the proof, we have to show that the theorem is true for $k+1$

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2 + \beta_k g_{k+1}^T v_k$$

$$= -\|g_{k+1}\|^2 + \frac{g_{k+1}^T y_k}{u_k y_k^T v_k + (1-u_k)\eta_k} g_{k+1}^T v_k \qquad \text{..........}(34)$$

The second term in (34) can be written

$$\frac{g_{k+1}^T y_k (g_{k+1}^T v_k)}{u_k y_k^T v_k + (1-u_k)\eta_k} = \frac{(g_{k+1}^T y_k)(u_k y_k^T v_k + (1-u_k)\eta_k)(g_{k+1}^T v_k)}{(u_k y_k^T v_k + (1-u_k)\eta_k)^2}$$

$$= \frac{\left[(u_k y_k^T v_k + (1-u_k)\eta_k) g_{k+1}\right]^T \left[(g_{k+1}^T v_k) y_k\right]}{(u_k y_k^T v_k + (1-u_k)\eta_k)^2}$$

$$\le \frac{(u_k y_k^T v_k + (1-u_k)\eta_k)^2 \|g_{k+1}\|^2 + (g_{k+1}^T v_k)^2 \|y_k\|^2}{2\,(u_k y_k^T v_k + (1-u_k)\eta_k)^2} \qquad \text{....}(35)$$

$$= \frac{1}{2}\|g_{k+1}\|^2 + \frac{(g_{k+1}^T v_k)^2 \|y_k\|^2}{2\,(u_k y_k^T v_k + (1-u_k)\eta_k)^2}.$$

Now, using (35) in (34) we get

$$g_{k+1}^T d_{k+1} \le -\|g_{k+1}\|^2 + \frac{1}{2}\|g_{k+1}\|^2 + \frac{(g_{k+1}^T v_k)^2 \|y_k\|^2}{2\,(u_k y_k^T v_k + (1-u_k)\eta_k)^2} \qquad \text{..........}(36)$$

Observe that the last term in (36) tends to zero very fast. Therefore, the direction d_{k+1} satisfies the sufficient descent condition :

$$\frac{g_{k+1}^T d_{k+1}}{\|g_{k+1}\|^2} \le -c \qquad \text{..........}(37)$$

where c is a positive constant, and $c \approx 1/2$.

Dai et al. [4] proved that for any conjugate gradient method with strong Wolfe line search the following general result holds :

Lemma (3.1)

Suppose that the assumptions (i) and (ii) hold and consider any conjugate gradient method (2) and (3), where d_{k+1} is a descent direction and α_k is obtained by the strong Wolfe line search (3) and (4). If

$$\sum_{k\geq 0}\frac{1}{\|d_{k+1}\|^2}=\infty\ , \qquad ....(38)$$

then

$$\liminf_{k\to\infty}\ \|g_{k+1}\|=0\ . \qquad ....(39)$$

As we know, if f is a uniformly convex functions, then there exists a constant $\mu>0$ such that

$$(\nabla f(x)-\nabla f(y))^T(x-y)\geq \mu\|x-y\|^2 \quad \text{, for any } x,y\in S. \qquad ....(40)$$

Equivalently, this can be expressed as

$$f(x)\geq f(y)+\nabla f(y)^T(x-y)+\frac{\mu}{2}\|x-y\|^2 \quad , \quad \text{for any } x,y\in S. \qquad ....(41)$$

From (40) and (41) it follows that

$$y_k^Tv_k\geq \mu\|v_k\|^2,$$
$$f_k-f_{k+1}\geq -g_{k+1}^Tv_k+\frac{\mu}{2}\|v_k\|^2. \qquad ....(42)$$

Obviously, from (38) we get :

$$\mu\|v_k\|^2\leq y_k^Tv_k\leq L\|v_k\|^2,\ \ i.e.\ \mu\leq L\ . \qquad ....(43)$$

More details can be found in **[2,3]**.

Using the above relations (42) and (43) we have

$$\begin{aligned}\lambda_k y_k^Tv_k+(1-\lambda_k)\theta_k &= \lambda_k y_k^Tv_k+(1-\lambda_k)(2(f_k-f_{k+1})+(g_{k+1}+g_k)^Tv_k)\\ &\geq \lambda_k y_k^Tv_k+(1-\lambda_k)(2(-g_{k+1}^Tv_k+\frac{\mu}{2}\|v_k\|^2)+(g_{k+1}+g_k)^Tv_k)\\ &= \lambda_k y_k^Tv_k+(1-\lambda_{kk})(-2g_{k+1}^Tv_k+2\mu\|v_k\|^2+g_{k+1}^Tv_k+g_k^Tv_k)\\ &= (\lambda_k+\lambda_k-1)y_k^Tv_k+(1-\lambda_k)(\mu\|v_k\|^2)\end{aligned} \qquad ....(44)$$

$$\lambda_k y_k^T v_k + (1-\lambda_k)\theta_k \ge (2\lambda_k - 1) y_k^T v_k + (1-\lambda_k)\frac{\mu}{L} y_k^T v_k$$
$$= (2\lambda_k - 1 + (1-\lambda_k)\frac{\mu}{L}) y_k^T v_k \qquad \text{..........(45)}$$
$$= m y_k^T v_k$$

Now, if $L \ge \mu$, then the right hand side of (45) is positive, that

$$u_k y_k^T v_k + (1-u_k)\theta_k \ge m y_k^T v_k \quad \text{where} \quad m = 2u_k - 1 + (1-u_k)\frac{\mu}{L} \qquad \text{......(46)}$$

Theorem (3.2)

Suppose that the assumptions (i) and (ii) hold and descent condition (37) hold. Consider the hybrid CG method in the form of (23) with u_k defined by (31), where α_k is computed using the strong Wolfe line search (3) and (4). If the objective function f is uniformly convex

$$\liminf_{k\to\infty} \|g_{k+1}\| = 0 . \qquad \text{..........(47)}$$

Proof :

Because the descent condition holds, we have $d_{k+1} \ne 0$. So, using Lemma 1, it is sufficient to prove that $\|d_{k+1}\|$ is bounded above. From (23), we have

$$\|d_{k+1}\| = \left\| -g_{k+1} + \frac{g_{k+1}^T y_k}{u_k y_k^T d_k + (1-u_k)\eta_k} v_k \right\|$$
$$\le \|g_{k+1}\| + \frac{|g_{k+1}^T y_k|}{|u_k y_k^T d_k + (1-u_k)\eta_k|}\|v_k\| \qquad \text{..........(48)}$$

Now, from (43) and (46) we have :

$$\|d_{k+1}\| \le \|g_{k+1}\| + \frac{L\|v_k\|\|g_{k+1}\|}{mL\|v_k\|^2}\|v_k\|$$
$$\le \varepsilon + \frac{\varepsilon L\|v_k\|}{mL\|v_k\|^2}\|v_k\| \qquad \text{..........(49)}$$
$$\le \varepsilon \left[1 + \frac{1}{m}\right]$$

This relation implies

$$\sum_{k\geq 1}\frac{1}{\|d_{k+1}\|^2}\geq\left[\frac{\mu}{\mu+L}\right]^2\frac{1}{\varepsilon^2}\sum_{k\geq 1}1=\infty \qquad ....(49)$$

Therefore, from Lemma (3.1) we have $\liminf_{k\to\infty}\|g_{k+1}\|=0$, which for uniformly convex function is equivalent to $\lim_{k\to\infty}\|g_{k+1}\|=0$.

4. Numerical Results

In this section, we reported some numerical results obtained with the implementation of the new algorithm on a set of unconstrained optimization test problems. We have selected (10) large scale unconstrained optimization problems in extended or generalized form, for each test function, we have considered numerical experiment with the number of variable n=100-1000. Using the standard Wolfe line search conditions (4) and (5) with $\delta_1=0.0001$ and $\delta_2=0.1$ In the all these cases, the stopping criteria is the $\|g_k\|\leq 10^{-5}$. The programs were written in Fortran 90. The test functions were commonly used for unconstrained test problems with standard starting points and a summary of the results of these test functions was given in Tables (3.1) and (3.2) . We tabulate for comparison of these algorithms, the number of function evaluations (NOF) and the number of iterations (NOI) .

Table (4.1)

No.	n	DY-algorithm	New-algorithm
		NOF (NOI)	**NOF (NOI)**
1	**100**	252 (33)	160 (24)
	1000	336 (39)	199 (26)
2	**100**	299 (121)	263 (110)
	1000	2492 (1001)	2250 (1001)
3	**100**	218 (105)	246 (105)
	1000	2025 (1005)	278 (120)
4	**100**	277 (94)	60 (21)
	1000	279	61

		(95)	(21)
5	**100**	216 (103)	40 (16)
	1000	554 (73)	39 (16)
6	**100**	209 (102)	22 (8)
	1000	561 (278)	22 (8)
7	**100**	97 (37)	63 (23)
	1000	116 (47)	72 (27)
8	**100**	115 (57)	101 (50)
	1000	273 (130)	141 (70)
9	**100**	77 (37)	27 (11)
	1000	85 (41)	27 (11)
10	**100**	79 (14)	84 (14)
	1000	163 (31)	134 (25)
	Total	**8723 (3443)**	**4289 (1707)**

Conclusions and Discussions

In this paper, we have proposed a new hybrid method for solving unconstrained minimization problems. The computational experiments show that the new approaches given in this paper are successful.

Table (4.2) gives a comparison between the new hybrid -algorithm and the DY-algorithm for convex optimization , this table indicates that the new algorithm saves (49.57)% NOI and (49.16)% NOF, overall against the standard DY-algorithm, especially for our selected test problems.

Relative Efficiency of the Different Methods Discussed in the Paper.

Tools	NOI	NOF
DY-CG	100 %	100 %
New	50.43 %	50.84 %

Appendix

1. Cantrell *function*:

$$f(x)=\sum_{i=1}^{n/4}[\exp(x_{4i-3})-x_{4i-2}]^4+100(x_{4i-2}-x_{4i-1})^6+[\tan^{-1}(x_{4i-1}-x_{4i})]^4+x_{4i-3}^8$$

Starting point: $(1,\ 2,\ 2,\ 2,\ldots\ldots)^T$

2. *Miele function*:

$$f(x)=\sum_{i=1}^{n/4}[\exp(x_{4i-3})-x_{4i-2}]^2+100(x_{4i-2}-x_{4i-1})^6+[\tan(x_{4i-1}-x_{4i})]^4+x_{4i-3}^8+(x_{4i}-1)^2$$

Starting point: $(1,\ 2,\ 2,\ 2,\ldots\ldots)^T$

3. *Generalized powell function*:

$$f(x)=\sum_{i=1}^{n/4}(x_{4i-3}-10x_{4i-2})^2+5(x_{4i-1}-x_{4i})^2+(x_{4i-1}-2x_{4i})^2+10(x_{4i-9}-x_{4i})^4+(x_{4i-2}-2x_{4i-1}-x_{4i})^2)$$

Starting point: $(3,-1,0,1,\ldots\ldots)^T$

4. *Rosenbrock function*:

$$f(x)=\sum_{i=1}^{n/2}(100(x_{2i}-x_{2i-1}^2)^2+(1-x_{2i-1})^2)$$

Starting point: $(-1.2,1,-1.2,1,\ldots\ldots)^T$

5. *Cubic function*:

$$f(x)=\sum_{i=1}^{n/2}(100(x_{2i}-x_{2i-1}^3)^2+(1-x_{2i-1})^2)$$

Starting point: $(-1.2,1,-1.2,1,\ldots\ldots)^T$

6. *Penalty2 function*:

$$f(x)=e^{(x(i)-1)^2}+(x(i)^2-0.25)^2$$

Starting point: $(1,2,\ \ldots\ldots)^T$

7. *Non-diagonal function*:

$$f(x)=\sum_{i=1}^{n/2}(100(x_i-x_i^3)^2+(1-x_i)^2)$$

Starting point: $(-1,\ldots\ldots)^T$

8. *Welfe function*:

$$f(x)=(-x_1(3-x_1/2)+2x_2-1)^2+\sum_{i=1}^{n-1}(x_{i-1}-x_i(3-x_i(3-x_i/2)+2x_{i+1}-1)^2+(x_{n+1}-x_n(3x_n/2-1)^2$$

Starting point: $(-1,\ \ldots\ldots)^T$

9. *Shallow function*:

$$f(x)=\sum_{i=1}^{n/2}((x_{2i-1}^2-x_{2i}x_{2i-1}^3)^2+(1-x_{2i-1})^2)$$

Starting point: $(-2,\ldots\ldots)^T$

10. *Sum of Quartics function*:

$$f(x)=\sum_{i=1}^{n}(x_i-1)^4$$

$$Starting\ point: (2, \ldots\ldots\ldots\ldots\ldots)^T$$

References

[1] Andrie, N. (2008) ' An Unconstrained Optimization Test functions collection ' Advanced Modeling and optimization. 10, pp.147-161.

[2] Andrei, N. (2008) ' Another hybrid nonlinear conjugate gradient algorithm for unconstrained optimization, Numer, Algor. 47, pp. 143-156.

[3] Andrei, N. (2010) ' Accelerated conjugate gradient algorithm with modified secant condition for un constrained optimization, Numer, Algor. 54, pp. 23-46.

[4] Dai, Y. and Yuan Y. (1999) ' A nonlinear conjugate gradients method with a strong global convergence property' SIAM J. Optimization 10, pp.177-182

[5] Fletcher, R. (1987) ' Practical Methods of Optimization (second edition). John Wiley and Sons, New York.

[6] Fletcher, R. and Reeves C. (1964) ' Function minimization by conjugate gradients ' Computer J. 7, pp.149-154.

[7] Hestenes, M. R. and Stiefel E. L. (1952) ' Method of conjugate gradients for solving linear systems' J. Research Nat. Standards 49, pp.409-436.

[8] Liu Y. and Storey C. (1991) ' Efficient generalized conjugate gradients algorithms ' Part 1 : Theory. J. Optimization Theory and Applications 69, pp.129-137.

[9] Polak, E. and Ribiere, G. (1969) ' Note for Convergence Direction Conjugate, Revue Francaise Informant, Reserche. Opertionelle, 16, 35-43.

[10] Li, G. Tang C. and Wei, Z., (2007) ' New conjugate gradient and related new conjugate gradient methods for unconstrained optimization, J. Comput. Appl. Math. 2, pp. 523-539.

[11] Saman, B. Kafaki, M., Masoud F. and Nezam, A. (2011) ' Two effective hybrid conjugate gradient algorithm for based on modified BFGS updates, unconstrained optimization, Numer, Algor. 58, pp. 315-331.