Optimization
Iranian Journal of Optimization
Volume 12, Issue 1, 2020, 33-44
Research Paper
Online version is available on: www.ijo.iaurasht.ac.ir

Islamic Azad University
Rasht Branch
ISSN: 2588-5723
E-ISSN:2008-5427

A New Hybrid Conjugate Gradient Method Based on Secant Equation for Solving Large Scale Unconstrained Optimization Problems

Nasiru Salihu[1*], Mathew Remilekun Odekunle[2], Mohammed Yusuf Waziri[3] and Abubakar Sani Halilu[4]

[1] Department of Mathematics, School of Physical Science, Moddibo Adama University of Technology, Yola, Nigeria
[2] Department of Mathematics, School of Physical Sciences, Modibbo Adama University of Technology, Yola, Nigeria
[3] Department of Mathematical Sciences, Faculty of Sciences, Bayero University, Kano, Nigeria
[4] Department of Mathematics and Computer Science, Sule Lamido University, Ka n Hausa, Nigeria

Received: 09 August 2019
Accepted: 03 November 2019

Keywords:
Unconstrained optimization, conjugate gradient algorithm, large scale optimization problem, secant equation, global convergence

Abstract

There exist large varieties of conjugate gradient algorithms. In order to take advantage of the attractive features of Liu and Storey (LS) and Conjugate Descent (CD) conjugate gradient methods, we suggest hybridization of these methods in which the parameter β_k is computed as a convex combination of β_k^{LS} and β_k^{CD} respectively which the conjugate gradient (update) parameter was obtained from Secant equation. The algorithm generates descent direction and when the iterate jam, the direction satisfy sufficient descent condition. We report numerical results demonstrating the efficiency of our method. The hybrid computational scheme outperform or comparable with known conjugate gradient algorithms. We also show that our method converge globally using strong Wolfe condition.

***Correspondence E-mail**: nasirussalihu@gmail.com

INTRODUCTION

A Conjugate Gradient (CG) method is designed to solve a nonlinear unconstrained optimization problem,

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

where $f:\mathbb{R}^n\to\mathbb{R}$ is a smooth nonlinear function. There exist many different methods to solve (1) (Bartholomew-Biggs, 2005; Necedah & Wright, 2006). Here we are interested in CG method, which have low memory requirement and local and global convergence properties (Djordjevic, 2017).

The iterative formula of a CG method is given by

$$x_0 \in \mathbb{R}^n$$
$$x_{k+1}=x_k+s_k \quad \alpha_k d_k \ ,k=0,1,... \qquad (2)$$

where α_k is a steplength to be computed by line search procedure and d_k is the search direction defined by

$$d_0=-g_0$$
$$d_{k+1}=-g_{k+1}+\beta_k d_k \qquad (3)$$

where, $g_k=\nabla f(x_k)$ and β_k is a scalar called CG (update) parameter, often computed by performing some inner products (Babaie-Kafaki, 2011). Different CG schemes correspond to different values of the scalar parameter β_k.

In general, two classes of CG schemes exist; there are some strengths and weaknesses for the CG schemes in each class (Babaie-Kafaki, 2011; Babaie-Kafaki, 2013; Babaie-Kafaki,

Ghanbari & Mahdavi-Amiri, 2010). The schemes with common numerator $g_{k+1}^T y_k$ have better practical performance, but may not always be convergent (Babaie-Kafaki & Ghanbari, 2014). These schemes were initially proposed by Hestenes and Stiefel (HS) (1952), Polak, Ribie're and Polyak (PRP) (1967), Liu and Storey (LS) (1991) with the following CG parameters respectively:

$$\beta_k^{HS}=\frac{g_{k+1}^T y_k}{d_k^T y_k}, \qquad \beta_k^{PRP}=\frac{g_{k+1}^T y_k}{\|g_k\|^2},$$

$$\beta_k^{LS}=-\frac{g_{k+1}^T y_k}{d_k^T g_k}. \qquad (4)$$

Numerical experiments show that the CG schemes with common numerator $\|g_{k+1}\|^2$ have strong global convergence properties, but they may have modest practical performance due to jamming (Andrei, 2008c, 2008a). These schemes were earlier proposed by Fletcher and Revees (FR) (1964), Fletcher (Conjugate Descent (CD)) (1987), and Dai and Yuan (DY) (1991) with the following CG parameters respectively:

$$\beta_k^{FR}=\frac{\|g_{k+1}\|^2}{\|g_k\|^2}, \qquad \beta_k^{DY}=\frac{\|g_{k+1}\|^2}{d_k^T y_k},$$

$$\beta_k^{CD}=-\frac{\|g_{k+1}\|^2}{d_k^T g_k}. \qquad (5)$$

where $\|.\|$ denotes Euclidean norm and define $s_k=x_{k+1}-x\_k$ and $y_k=g_{k+1}-g_k$ (Dai & Yuan, 2001).

To improve the behavior of these schemes and to avoid jamming (Djordjevic, 2017), researchers were interested in combining CG schemes of the two groups (Babaie-Kafaki & Mahdavi-Amiri, 2013).

Moreover, convergence analysis and implementation of the conjugate gradient algorithms, when α_k is one dimensional minimizer along d_k often requires line search α_k to be exact (Rao, 2009):

$$\alpha=arg\min_{\alpha}\ (x_k+\alpha d_k), \qquad (6)$$

or satisfy standard Wolfe conditions. However, in practice, an exact line search is not usually possible and any value of α_k satisfies certain conditions is accepted (Hager & Zhang, 2006; Nocedal & Wright, 2006; Touati-Ahmed & Storey, 1990):

$$f(x_k+\alpha_k d_k)\le f(x_k)+\ c_1\nabla f_k^T d_k, \qquad (7)$$

$$\nabla f(x_k+\alpha_k d_k)^T d_k \ge\ c_2\nabla f_k^T d_k, \qquad (8)$$

where $0<c_1<c_2<1$, d_k is a descent direction (Babaie-Kafaki & Ghanbari, 2014). On the other hand, strong Wolfe conditions consist of (7) and

$$|\nabla f(x_k + \alpha_k d_k)^T d_k| \leq -c_2 \nabla f_k^T d_k. \quad (9)$$

The difference between schemes with common numerator $\|g_{k+1}\|^2$ with other choices

for the update parameter in theory is that the global convergence theorems only require

the Lipchitz assumption, not the boundedness assumption (Hager & Zhang, 2006). The poor practical performance of FR method is related to jamming (Powell, 1984). If a bad direction and a tiny step from x_{k-1} and x_k are generated, then the next direction d_k and the next step α_k are also likely to be poor unless a restart along the gradient direction is made (Babaie-Kafaki, Fatemi & Mahdavi-Amiri, 2011). In spite of such a defect Zoutendijk (1970) proved that the FR method with exact line search is globally convergent on general functions; Al-Baali (1985) extended this result to an in-exact line search (Hager & Zhang, 2006). However, the schemes with the common numerator possess an automatic approximate restart feature which addresses jamming problem (Babaie-Kafaki, 2013). More exactly, when the step s_k is small, the factor y_k in the numerator tends to zero. Therefore, β_k becomes small and the new search direction d_{k+1} is approximately the steepest descent direction $-g_{k+1}$ (Andrei, 2008b, Andrei, 2009a). In general, the performance of this method is better than the performance of the methods with $\|g_{k+1}\|^2$ in the numerator of β_k but their convergence is uncertain (Hager & Zhang, 2006; Powell, 1984).

The CD scheme is closely related to FR scheme with exact line search, $\beta_k^{FR}=\beta_k^{CD}$. One important difference between FR and CD methods is that with CD, the sufficient descent holds for a strong Wolfe line condition (the constraint $c<1/2$ that stand up with FR, is not needed for CD). Moreover, for a line search that satisfies the generalized Wolfe conditions with $c_1<1$ and $c_2=0$, it can be shown that CD scheme is globally convergent (Hager & Zhang, 2006). Djordjevic (2017) pointed out that no much research has been done on the choice β_k^{LS}, except for the work of Liu and Storey (1991), but we expect that the techniques developed for the analysis of the PRP method should be applied to the LS method (Hager & Zhang, 2006). Similarly, for an exact line search, the LS scheme is also identical to PRP (Dai, 2001).

A large number of hybrid conjugate gradients techniques were proposed (Babaie-Kafaki, Ghanbari & Mahdavi-Amiri, 2010). These algorithms dynamically adjust the formula for β_k as the iteration evolves (Hager & Zhang, 2006). The idea is to use projections which are mainly proposed in order to avoid jamming (Andrei, 2008c, 2008a). Among them: (e.g. see also Andrei, 2008a; Andrei, 2009a; Babaie-Kafaki, Fatemi & Mahdavi-Amiri, 2011; Babaie-Kafaki & Mahdavi-Amiri; 2013; Dai, 2001, Dai & Yuan, 2001; Gilbert & Nocedal, 1992; Hu & Storey, 1991; Li & Fukushima, 2001; Liu & Storey, 1991; Sabiu & Waziri, 2017; Sabiu, Waziri & Idris, 2017; Touati-Ahmed & Storey, 1990; Yuan,1991). The excellent contributions of Andrei and Babaie-Kafaki on hybridization using convex combination and that of Djordjevic motivated us to extend their approaches to access and combine the strength of the LS and CD CG update parameters. This paper is organized as follows: Next section presents the proposed method. Convergence results are presented in Section 3. Some numerical results are reported in Section 4. Finally, conclusions are made in Section 5.

CONVEX COMBINATION HYBRID CG METHOD

We briefly discuss Hybrid Conjugate Gradient (HCG) of Babaie-Kafaki and Ghanbari (2014) in addition to Liu and Storey with Conjugate Descent Convex Combination (LSCDCC) of Djordjevic (2017) methods. HCG obtained two CG parameters from standard and scaled secant equations to avoid storing and computing Hessian matrix, where the parameters are computed as a convex combination of β_k^{HS} and β_k^{DY} while LSCDCC obtained the hybrid parameter from conjugacy condition which is globally convergent using strong Wolfe conditions. The hybrid parameter is computed as a convex combination of β_k^{LS} and β_k^{CD}. The hybrid parameters θ_k of these algorithms are computed as a proper convex combination. In order to achieve global convergence for general function, HCG adopted non-negative restriction of the CG parameter, while the other hybrid parameter is globally convergent for uniformly convex function. On the other hand, no much research has been done on the choice of β_k^{LS} CG parameter except for the work of Djordjevic (2017) and Liu & Storey (1991), and the fact that the most essential secant equation is the standard secant equation motivated this work.

In this section, we combine the CG update parameters proposed by Liu and Storey

(1991) with Fletcher (1987) conjugate descent as hybrid conjugate gradient method based on Convex Combination of LS and CD using Secant Equation (CLCS) as follows:

$$\beta_k^{CLCS} = (1-\theta_k)\,\beta_k^{LS} + \theta_k\,\beta_k^{CD}. \tag{10}$$

From equation (4) and (5)

$$\beta_k^{CLCS} = (1-\theta_k)\left(-\frac{g_{k+1}^T y_k}{g_k^T s_k}\right) + \theta_k\left(-\frac{g_{k+1}^T g_{k+1}}{g_k^T s_k}\right), \tag{11}$$

where θ_k is the hybridization scalar parameter satisfying $\theta_k \in [0,1]$. It is obvious that if $\theta_k \le 0$, set $\theta_k = 0$, then $\beta_k^{CLCS} = \beta_k^{LS}$ and if $\theta_k \ge 0$, set $\theta_k = 1$, then $\beta_k^{CLCS} = \beta_k^{CD}$. On the other hand, if $0 < \theta_k < 1$, then β_k^{CLCS} is a proper convex combination of β_k^{CD} and β_k^{LS}. Therefore, from relation (3) we obtain

$$d_{k+1} = -g_{k+1}\left((1-\theta_k)\left(-\frac{g_{k+1}^T y_k}{g_k^T s_k}\right) + \theta_k\left(-\frac{g_{k+1}^T g_{k+1}}{g_k^T s_k}\right)\right)s_k. \tag{12}$$

If x_{k+1} is close to x^*, then it is important to note that, the direction to follow is Newton's direction:

$$d_{k+1} = -\nabla^2 f(x_{k+1})^{-1} g_{k+1}. \tag{13}$$

Equating (12) and (13) and after some algebra we have

$$\theta_k = \frac{s_k^T g_{k+1} \quad s_k^T \nabla^2 f(x_{k+1}) g_{k+1} \quad \left(\frac{g_{k+1}^T y_k}{g_k^T s_k}\right) s_k^T \nabla^2 f(x_{k+1}) s_k}{\left(\frac{g_{k+1}^T g_k}{g_k^T s_k}\right) s_k^T \nabla^2 f(x_{k+1}) s_k}. \tag{14}$$

However, for large scale problems, choices for the update parameter that do not require evaluation of the Hessian matrix is often require (Ding, Lushi & Li, 2010). Therefore, in order to have an algorithm for solving large scale problems, we assume pair of (s_k, y_k) satisfies the standard secant equation $\nabla^2 f(x_{k+1})\, s_k = y_k$ (Sun & Yuan, 2006; Zhang & Xu, 2001); (14) becomes:

$$\theta_k = \frac{(s_k^T g_k)(s_k^T g_{k+1} - y_{k+1}^T g_{k+1}) - (g_{k+1}^T y_k)(s_k^T y_k)}{(g_{k+1}^T g_k)(s_k^T y_k)}. \tag{15}$$

Obviously, from (12) our direction can be justified as:

$$d_{k+1} = -g_{k+1} + (1-\theta_k)\left(-\frac{g_{k+1}^T y_k}{g_k^T s_k}\right)s_k + \theta_k\left(-\frac{g_{k+1}^T g_{k+1}}{g_k^T s_k}\right)s_k. \tag{16}$$

We can write (16) as:

$$d_{k+1} = -g_{k+1} + (1-\theta_k)\left(-\frac{s_k y_k^T}{g_k^T s_k}\right)g_{k+1} + \theta_k\left(-\frac{s_k g_{k+1}^T}{g_k^T s_k}\right)g_{k+1}. \tag{17}$$

It follows from (17), that

$$d_{k+1} = -\left[I + \left((1-\theta_k)\frac{s_k y_k^T}{g_k^T s_k} + \theta_k\frac{s_k g_{k+1}^T}{g_k^T s_k}\right)\right]g_{k+1}. \tag{18}$$

$$d_{k+1} = -Q_{k+1} g_{k+1}, \tag{19}$$

where, we have

$$Q_{k+1} = I + \left((1 \quad \theta_k)\frac{s_k y_k^T}{g_k^T s_k} + \theta_k\frac{s_k g_{k+1}^T}{g_k^T s_k}\right). \tag{20}$$

So, (19) can be considered as quasi-Newton direction in which the inverse Hessian matrix in each iteration is approximated by matrix Q_{k+1}. Therefore, the direction (19) is an approximation of the Newton direction.

CLCS algorithm

Step 1. Initialization. Select $x_0 \in R^n$ and parameter $0<c_1<c_2<1$. Compute $f(x_0)$ and g_0.

Consider $d_0=-g_0$ and set $\alpha_0=1$.

Step 2. Test for Continuation of Iterations. If $\|g_k\|_\infty \leq 10^{-4}$, then stop.

Step 3. Line Search. Compute $\alpha_k>0$ satisfying Wolfe conditions (7) and (9) and update the variables, $x_{k+1}=x_k+\alpha_k d_k$. Compute $f(x_{k+1})$, g_{k+1} and $s_k = x_{k+1}-x_k$, $y_k=g_{k+1}-g_k$.

Step 4. Computation of θ_k. If $(g_{k+1}^T g_k))(s_k^T y_k)=0$, then set $\theta_k=0$; otherwise, compute θ_k by (15).

Step 5. Computation of β_k^{CLCS}. If $0<\theta_k<1$, then compute β_k^{CLCS} by (10). If $\theta_k \geq 1$, then set $\beta k^{CLCS}=\beta_k^{CD}$. If $\theta_k \leq 0$, then set $\beta_k^{CLCS}=\beta k^{LS}$.

Step 6. Computation of Search Direction. Compute $d=-g_{k+1}+\beta k^{CLCS} s_k$. If restart criterion of Powell

$$|g_{k+1}^T g_k| > a\|g_{k+1}\|^2, \text{ where } a=0.2 \tag{21}$$

is satisfied, then set $d_{k+1}=-g_{k+1}$; otherwise, define $d_{k+1}=d$. Compute α_k, set $k=k+1$ and go to step 2.

CONVERGENCE ANALYSIS

To state the convergence result of hybrid CG method CLCS, the following definitions and basic assumptions are necessary:

Definition: Search direction satisfies descent directions (or equivalently, satisfy the decent condition) if an only if

$$d_k^T g_k<0, \tag{22}$$

and also satisfies sufficient descent condition if and only if

$$d_k^T g_k<-c\|g_k\|^2, \forall k\geq 0, \tag{23}$$

where c is positive constant.

Boundedness Assumptions: Assumption 3.1. The level set $S=\{x\in R: f(x)\leq f(x_0)\}$, with x_0 to be the starting point of CG methods (2) and (3) is bounded from below. That is, there exist a positive constant B such that

$$\|x\|\leq B, \forall x\in S. \tag{24}$$

Lipchitz Assumptions: Assumption 3.2. In a neighborhood N of S, the objective function f is continuously differentiable and its gradient ∇f is Lipchitz continuous on N that is, there exist a constant $L>0$ such that $\|\nabla f(x)-\nabla f(y)\| \leq L\|x-y\|$, for all $x,y\in N$.

Under Assumption 3.1 and Assumption 3.2 on f, there exist a constant Γ such that

$$\|\nabla f(x)\| \leq \Gamma, \tag{25}$$

for all $x\in S$ (Andrei, 2009b).

Lemma1. Let $f\in\subset(R^n)$. Let d_k be a descent direction in the point x_k, and suppose that the function f is bounded from below along direction $\{x_k+\alpha d_k | \alpha>0\}$. Then if $0<c_1<c_2<1$, there exist the intervals inside which the step size satisfies (7),(8) and (9) (Nocedal & Wright, 2006).

Theorem 1. Consider iteration of the form (2) and algorithm (9), assume that α_k satisfies (7) and (9). If $0<c_1<c_2<1$, then d_{k+1} given by (12) is a descent direction.

Theorem 2. Djordjevic (2017). Let Assumptions 3.1 and 3.2 hold. Let constant a in the algorithms CLCS be such that

$$0<a<1/c_1 -1 . \tag{26}$$

Then the algorithms CLCS is well defined d_k and satisfies the (23) for all k.

Proof: From Lemma 1, we know that Step 2 of the algorithms CLCS is well defined if d_k is a descent direction. We shall show that d_k satisfies the sufficient descent condition, and that will yield d_k as a descent direction. For k=0, it holds $d_0=-g_0$, so $g_0^T d_0=-\|g_k\|^2$, and that can be concluded that (23) holds for k=0.

Next is to show that it holds for k>0.

$$d_{k+1} = -g_{k+1} + \beta_k^{CLCS} s_k. \quad (27)$$

Obviously

$$d_{k+1} = -g_{k+1} + ((1-\theta_k)\ \beta_k^{LS} + \theta_k\ \beta_k^{CD}) s_k. \quad (28)$$

We can write

$$d_{k+1} = -(\theta_k g_{k+1}) + (1-\theta_k)\, g_{k+1} + ((1-\theta_k)\ \beta_k^{LS} + \theta_k\ \beta_k^{CD})) s_k. \quad (29)$$

It follows that

$$d_{k+1} = \theta_k(-g_{k+1} + \beta_k^{CD} s_k) + (1-\theta_k)(-g_{k+1}\ \beta_k^{LS} s_k.) \quad (30)$$

Where we have

$$d_{k+1} = \theta_k\ d_{k+1}^{CD}\ (1-\theta_k)\, d_{k+1}^{LS}. \quad (31)$$

Pre-multiply (31) by g_{k+1}^T, we get

$$g_{k+1}^T d_{k+1} = \theta_k\, g_{k+1}^T\ d_{k+1}^{CD} + (1-\theta_k)\, g_{k+1}^T d_{k+1}^{LS}. \quad (32)$$

Firstly, let $\theta_k=0$, then $d_{k+1}=d_{k+1}^{LS}$. Remember that

$$d_{k+1}^{LS} = -g_{k+1} + \beta_k^{LS} s_k.$$

$$\Rightarrow g_{k+1}^T d_{k+1} \leq -\|g_{k+1}\|^2 + (g_{k+1}^T y_k)(g_{k+1}^T s_k)/(-g_k^T s_k). \quad (33)$$

For (33) to satisfy sufficient descent condition, we have

$$\left|\frac{(g_{k+1}^T y_k)(g_{k+1}^T s_k)}{-g_k^T s_k}\right| \leq \mu\, \|g_{k+1}\|^2, \text{ where } 0<\mu<1.$$

So that

$$g_{k+1}^T d_{k+1}^{LS} \leq -\|g_{k+1}\|^2 + \mu\, \|g_{k+1}\|^2,$$

and

$$g_{k+1}^T d_{k+1}^{LS} \leq -(1-\mu)\|g_{k+1}\|^2.$$

We denote $K_1=(1-\mu)$: then we can write

$$g_{k+1}^T d_{k+1}^{LS} \leq -K_1\, \|g_{k+1}\|^2. \quad (34)$$

We are done with $\theta_k=0$.

Now, let $\theta_k=1$, then $d_{k+1}= d_{k+1}^{CD}$.

Further, we are going to prove that (23) holds for CD method in the presence of (7) and (9), and this fact is mentioned in [26].

For k = 0, the proof is a trivial one, having in view that $d_0^{CD}=-g0$, so $g_0^T d_0^{CD}=-\|g_k\|^2$, and that can be concluded that (23) holds for k=0.

Having in view that

$$d_{k+1}^{CD} = -g_{k+1} + \beta_k^{CD} s_k. \quad (35)$$

Pre-multiply (35) by g_{k+1}^T, we get

$$g_{k+1}^T d_{k+1}^{CD} = -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2}{-g_k^T s_k}(g_{k+1}^T s_k), \quad (36)$$

Where from (36)

$$g_{k+1}^T d_{k+1}^{CD} = -\|g_{k+1}\|^2\left(1 - \frac{g_{k+1}^T s_k}{-g_k^T s_k}\right)$$
$$= -\|g_{k+1}\|^2\left(\frac{-g_k^T s_k - g_{k+1}^T s_k}{-g_k^T s_k}\right).$$

Using (7) and (9), it is obvious that

$$\frac{-g_k^T s_k - g_{k+1}^T s_k}{-g_k^T s_k} \geq \frac{c_2 g_k^T s_k - g_k^T s_k}{-g_k^T s_k} = 1 - c_2 > 0.$$

Now we have
$g_{k+1}^T d_{k+1}^{CD} \leq -(1-c_2)\, \|g_{k+1}\|^2$.

We denotes $(1-c_2)=K_2>0$.

$$g_{k+1}^T d_{k+1}^{CD} \leq -K_2\, \|g_{k+1}\|^2. \quad (37)$$

Now suppose that $0<a_1<\theta_k<a_2<1$. From (32), we conclude that

$$g_{k+1}^T d_{k+1} \leq a_1 g_{k+1}^T d_{k+1}^{CD} + (1-a_2) g_{k+1}^T d_{k+1}^{LS}. \quad (38)$$

Denote $K = a_1 K_1 (1-a_2) K_2$; then we finally get

$$g_{k+1}^T d_{k+1} \leq -K \|g_{k+1}\|^2 . \quad (39)$$

Global convergence analysis

For any conjugate gradient method with strong Wolfe line search, the convergence holds. But, for general function, only weak form of the Zoutendijk condition is needed (Dai and Liao, 2001):

Lemma 2. Let Assumptions 3.1 and 3.2 hold. Consider the method (2), (3) where d_k is a descent direction and α_k satisfies (7) and (9). If

$$\sum_{k\geq 1} \frac{1}{\|d_k\|^2} = \infty, \quad (40)$$

then

$$\lim_{k\to\infty} inf \|g_k\| = 0. \quad (41)$$

A CG method converges globally if $g_k=0$ for some k or (41) holds.

Theorem 3. Consider the iterative method, defined by CLCS algorithms . Let d_{k+1} be a descent direction, then either $g_k=0$, for some k, or

$$\lim_{k\to\infty} inf \|g_k\| = 0. \quad (42)$$

The proof is using contradiction, that theorem (3) is not true.

Proof: Let $g_k \neq 0$, for all k. Suppose on the contrary, that (42) does not hold, which means the gradient is bounded away from zero. Then there exist a constant $c>0$, such that

$$\|g_k\| \geq c. \quad (43)$$

Let D be the diameter of the level set S. So, from (10) we have

$$\| \beta_k^{CLCS} \| \leq | \beta_k^{LS} | + |\beta_k^{CD}|, \quad (44)$$

but from (25), it holds that

$$\left| \beta_k^{LS} \right| = \left| \frac{g_{k+1}^T y_k}{-g_k^T s_k} \right| \leq \frac{\|g_{k+1}\| \|y_k\|}{|-g_k^T s_k|} \leq$$

$$\frac{\Gamma \|y_k\|}{|-g_k^T s_k|} \leq \frac{\Gamma L \|y_k\|}{|-g_k^T s_k|} .$$

Because $\|s_k\| \leq D$, it means that

$$\left| \beta_k^{LS} \right| \leq \frac{\Gamma L D}{|-g_k^T s_k|} . \quad (45)$$

Using theorem (2), we know that for *LS* method, the sufficient descent condition holds, so it is possible to satisfy (7) and (9).

Now we are to prove that there exist $\alpha_* > 0$, such that $\alpha_k \geq \alpha_* > 0$, for all k.

Suppose, on the contrary, that there do not exist any α_* , such that $\alpha_k \geq \alpha_* > 0$. Then there exist an infinite sub-sequence $\alpha_* = \beta^{jk}$, $k \in K_1$ such that

$$\lim_{k\in K_1} \alpha_k = 0, \quad (46)$$

then

$$\lim_{k\in K_1} \beta^{jk-1} = 0.$$

That is,

$$\lim_{k\in K_1} (jk-1) = \infty.$$

But, from Armijo line search, we get

$$f(x_k + \beta^{jk} d_k) - f(x_k) \leq c_1 \beta^{jk} g_k^T d_k, \quad (47)$$

$$f(x_k + \beta^{jk-1} d_k) - f(x_k) > c_1 \beta^{jk-1} g_k^T d_k. \quad (48)$$

Remember that $c_1 < 1$. From (48), we have

$$\frac{f(x_k + \beta^{jk-1} d_k) - f(x_k)}{\beta^{jk-1}} > c_1 g_k^T d_k. \quad (49)$$

But, using relations (46) and from (49), we conclude that

$$g_k^T d_k \geq c_1 g_k^T d_k. \quad (50)$$

But, LS method satisfies (23), so $g_k^T d_k \leq 0$. Also, $c_1 < 0$. So, the relation (50) is true only if $g_k^T d_k = 0$. Then, from (8), we get $g_{k+1}^T d_k = 0$ which is an exact line search, a contradiction.

Now, we can write

$|-g_k^T s_k| = |-\alpha_k g_k^T s_k| \geq |-\alpha_* g_k^T s_k|$.

From (45), using (23) in $|-g_k^T s_k|$, we get

$$|\beta_k^{LS}| \leq \frac{\Gamma LD}{K'\|g_{k+1}\|^2}, K' > 0. \tag{51}$$

From (43), we get

$$|\beta_k^{LS}| \leq \frac{\Gamma LD}{K'c^2}. \tag{52}$$

Since

$$d_{k+1}^{LS} = -g_{k+1} + \beta_k^{LS} s_k.$$

we get

$$\|d_{k+1}^{LS}\| \leq \|g_{k+1}\| + |\beta_k^{LS}| \, \|s_k\|. \tag{53}$$

Using (25), (52) and $\|s_k\| \leq D$, we get

$$\|d_{k+1}^{LS}\| \leq \|g_{k+1}\| + \frac{\Gamma LD}{K'c^2} D \leq \Gamma + \frac{\Gamma LD^2}{K'c^2}. \tag{54}$$

Also using (25) in (35) we get

$$\|d_{k+1}^{CD}\| \leq \|g_{k+1}\| + |\beta_k^{CD}| \, \|s_k\| \leq \Gamma + |\beta_k^{CD}| D. \tag{55}$$

So that

$$|\beta_k^{CD}| = \frac{\|g_{k+1}\|^2}{|-g_k^T s_k|} \leq \frac{\Gamma^2}{|-g_k^T s_k|}. \tag{56}$$

We conclude that (22) and (23) hold for CD method too, so, analogically, we can get

$$|\beta_k^{CD}| = \frac{\Gamma^2}{K''\|g_k\|^2} \leq \frac{\Gamma^2}{K''c^2}, \; K'' > 0. \tag{57}$$

So,

$$\|d_{k+1}^{CD}\| \leq \Gamma + \frac{\Gamma^2 D}{K''c^2}. \tag{58}$$

Applying (52) and (58) on (31), we find that

$$\|d_{k+1}\| \leq \Gamma + \frac{\Gamma LD^2}{K'c^2} + \Gamma + \frac{\Gamma^2 D}{K''c^2}. \tag{59}$$

Therefore,

$$\sum_{k\geq 1} \frac{1}{\|d_k\|^2} = \infty. \tag{60}$$

So, applying Lemma (1), we conclude that

$$\lim_{k\to\infty} inf \|g_k\| = 0. \tag{61}$$

This is a contradiction of (43), so we have proved (42).

NUMERICAL RESULTS

In this section, we present the computational performance of CLCS and compare with that of LSCDCC of Djordjevic (2017) and HCG method of Babaie-Kafaki and Ghanbari (2014). To implement the hybridize CG parameters, the codes were written in Matlab 8.3 (R2014a) and run on a personal computer 2.20 GHz CPU processor and 3.0 GB RAM memory and tested on a set of 250 unconstrained optimization problems. The test problems are the unconstrained problems in (Andrei, 2008b) and (Gould, Orban & Toint, 2003). Since CG schemes are mainly designed to solve large-scale unconstrained optimization, we select 25 problems in extended or generalized form. Each problem is tested 10 times for a gradually increasing number of variables: 100, 200, 500, 1000, 2000, 5000, 10000, 20000, 50000 and 100000 with summary of the numerical results has shown in Table 1. All the algorithms were implemented on a strong Wolfe line search conditions with $c_1=0.0001$ and $c_2= 0.001$ and the step length is computed with initial trail value $\alpha_k=1$. The same stopping criterion $\|g_k\|_\infty \leq [\![10]\!]^{-4}$ is used. All the test functions were minimizing from standard starting points.

Numerical results were compared based on number of iterations and CPU time. In some cases, the computation stopped due to failure of the line search to find the positive step size, and thus it was considered a failure. In addition, we considered a failure if the number of iterations exceeds 10000 or CPU time exceeds 500 (Secs). fig. 1-2 show the performance of these methods using the profiles of Dolan and Mor´e (2002). The $P(\tau)$ is the fraction of problems with performance ration τ, thus, a solver with high values $P(\tau)$ or at the top right of the figures are preferable. That is, for each method, we plot the fraction or percentage $P(\tau)$ of the problems for which the method is within a factor versus time τ, the best time for each algorithm. The left side gives the percentage of the test problems of the method that is fastest. The right side gives the percentage of the test problems that are successfully solved by each method. The interpretation of fig. 1 shows that the probability of CLCS method is the winner on a given problem is 62% while LSCDCC and HCG methods win 44% and 15% percentages respectively, when the factor τ is chosen within the interval $0 < \tau < 0.5$. Clearly, CLCS method has the most wins, because it has the highest probability of being closer to the optimal solution. However, if we extend our τ of interest to $\tau \geq 0.5$, CLCS and HCG algorithms solved the test functions in a given time and reach 88% respectively, while LSCDCC method is 85% to. It is easy to see that the performance of CLCS and HCG algorithms are comparable and computationally efficient than LSCDCC scheme.

Table 1: Summary of Numerical Results of CLCS, LSCDCC and HCG Methods

	CLCS	HCG	LSCDCC	
SUCCESSFUL	101	20	33	CPU TIME
	197	177	178	NO. OF ITERATION
NOT SUCCESSFUL	119	200	180	CPU TIME
	23	36	35	NO. OF ITERATION
FAILURE	30	30	37	CPU TIME
	30	37	36	NO. OF ITERATION
TOTAL	250	250	250	CPU TIME
	250	250	250	NO. OF ITERATION

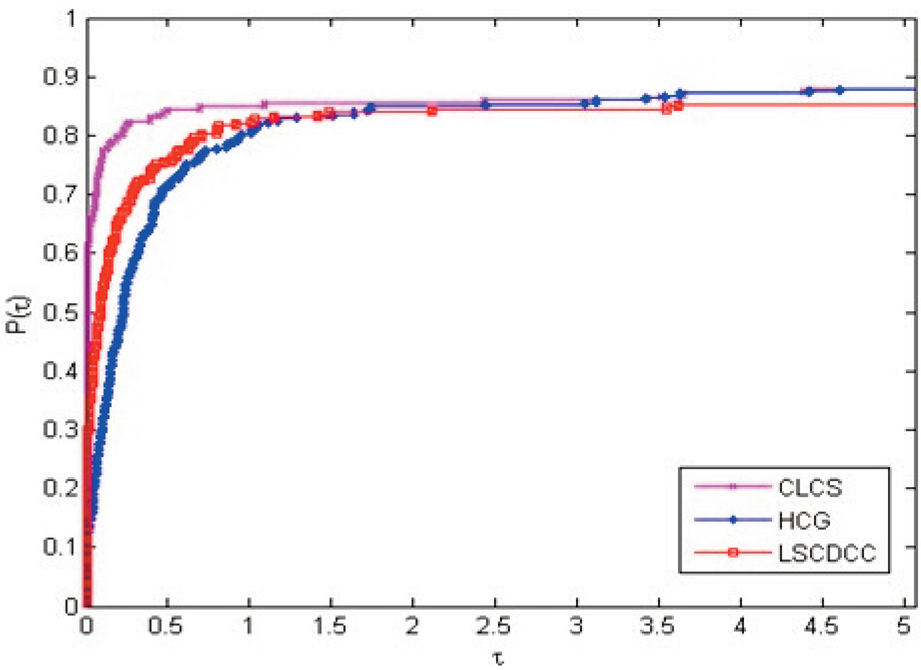

Fig. 1. CPU time performance profile for CLCS, HCG and LSCDCC schemes.

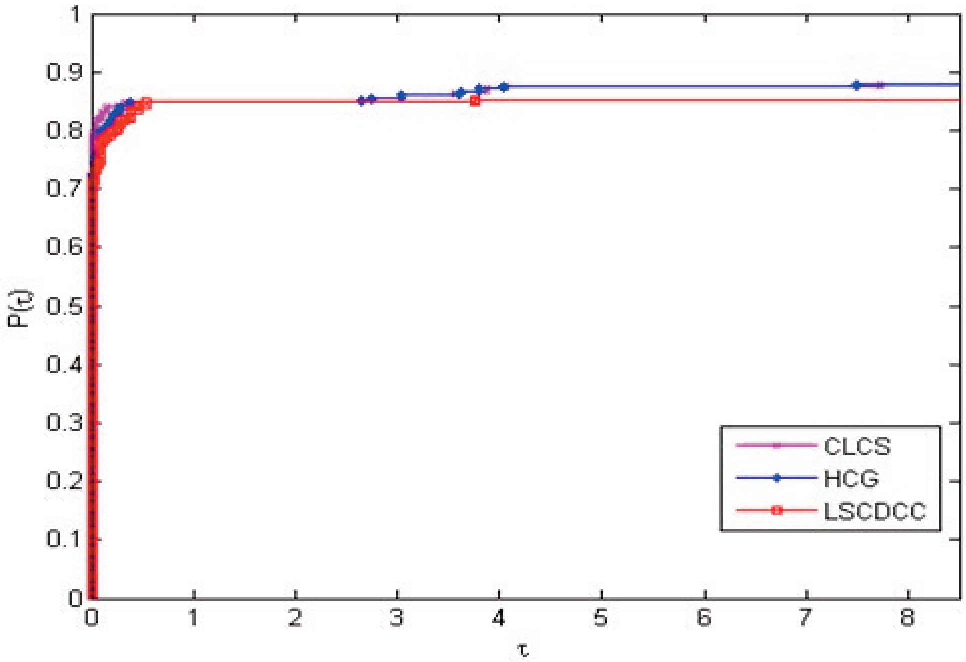

Fig. 2. Number of iterations performance profile for CLCS, HCG and LSCDCC schemes.

Since the CPU time is often affected by the environment of computer such as the busy or free task status and the operating system, we further make a comparison among the three methods with the number of iterations. fig 2, shows that the fraction of CLCS method is the winner on a given problem with 82% while LSCDCC and HCG methods win 79% and 72% percentages respectively, when the factor τ is chosen within the interval $0<\tau<0.5$. Clearly, CLCS method has the most wins, because it has the highest probability of being closer to the optimal solution. However, if we extend our τ of interest to $\tau \geq 0.5$, CLCS and HCG algorithms solved the test functions in a given number of iterations and reach 88% respectively, while LSCDCC method is 85%, it is easy to see that the performance of CLCS and HCG algorithms are computationally efficient than LSCDCC scheme.

CONCLUSION

Numerous studies of CG methods led to new-varieties of conjugate gradient algorithms. However, we have presented new hybrid conj gate hybrid algorithms in which the parameter β_k is computed as a convex combination of β_k^{LS} and β_k^{CD}. The hybrid parameter was obtained based on secant equation and compared with LSCDCC conjugate gradient method proposed by Djordjevic and HCG proposed by Babaie-Kafaki and Ghanbari. Numerical results show that our scheme and HCG algorithm are comparable and outperform LSCDCC scheme. The algorithm converge globally using strong Wolfe condition.

REFERENCE

Al-Baali, M. (1985). Descent property and global convergence of the Fletcher Reeves method with inexact line search. *IMA Journal of Numerical Analysis, 5*, 121-124.

Andrei, N. (2008a). A hybrid conjugate gradient algorithm for unconstrained optimization as convex combination of Hestenes- Stiefel and Dai-Yuan. *Study Information Control, 17*, 55-70.

Andrei, N. (2008b). An unconstrained optimization test function. *Advanced Modeling and Optimization. An Electronic International Journal, 10*, 172-182.

Andrei, N. (2008c). Another hybrid conjugate gradient algorithm for unconstrained optimization. *Numerical Algorithm, 47*, 143-156.

Andrei, N. (2009a). Hybrid conjugate gradient algorithm for unconstrained optimization. *Journal of Optimization Theory Application, 141,* 249-264.

Andrei, N. (2009b). New hybrid conjugate gradient algorithm for unconstrainedoptimization. *Encyclopedia of Optimization*, 2560-2571.

Babaie-Kafaki S., Ghanbari R., & Mahdavi-Amiri, N (2010). Two new conjugate gradient methods based on modified secant equations. *J. Comput. Appl. Math.*, *234*, 1374-1386.

Babaie-Kafaki, S. & Ghanbari, R. (2014). Two hybrid nonlinear conjugate gradient methods based on a modified secant equation. *Journal of Mathematical Programming and Operation Research, 63*(7), 1027-1042.

Babaie-Kafaki, S. & Mahdavi-Amiri, N. (2013). Two hybrid conjugate gradient methods based on hybrid secant equation. *Mathematical Modelling and Analysis, 18*(1): 32-52.

Babaie-Kafaki, S. (2011). A modified BFGS algorithm based on a hybrid secant equation. *Science China Mathematics, 58,* 315-331.

Babaie-Kafaki, S. (2013). A hybrid conjugate gradient method based on quadratic relaxation of Dai-Yuan hybrid conjugate gradient parameter. *Journal of Mathematical Programming and Operation Research, 62*(7), 929-941.

Babaie-Kafaki, S., Fatemi, M., & Mahdavi-Amiri, N. (2011). Two effective hybrid conjugate gradient algorithms based on modified BFGS updates. *Science China Mathematic, 54*(9), 2019-2036.

Bartholomew-Biggs, M. (2005). *Nonlinear optimization with financial applications*. Springer Science & Business Media.

Dai, Y. H. (2001). New properties of nonlinear conjugate gradient method. *Numerical Mathematic, 89*, 83-98.

Dai, Y.H & Liao, L.Z. (2001). New conjugacy conditions and related nonlinear conjugate gradient methods, *Appl. Math. Optim*., *43*, 87-101.

Dai, Y.H. & Yuan, Y. (1999). A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim*., *10*, 177-182.

Dai, Y.H. & Yuan, Y. (2001). An efficient hybrid conjugates gradient method for unconstrained optimization. *Annals of Operations Research, 103*, 33-47.

Ding, Y., Lushi, E. & Li, Q. (2010). Investigation of quasi-Newton methods for unconstrained optimization. *International Journal of Computer Application, 29*, 48-58.

Djordjevic, S.S. (2017). New hybrid conjugate gradient methods as a convex combination of LS and CD methods. *Published by faculty of science and mathematics. University of Nis, Serbia, 31*(6), 1813-1825.

Dolan, E.D. & More´, J.J. (2002). Benchmarking optimization software with performance profiles. *Journal of Math. Program*, *91*(2), 201-213.

Fletcher, R. (1987). *Practical methods of optimization,* (vol. 1): *Unconstrained optimization*. New York: John Wiley & Sons.

Fletcher, R., & Reeves, C. (1964). Function minimization by conjugate gradients. *Computational Journal*, *7*, 149-154.

Gilbert, J.C. & Nocedal, J. (1992). Global convergence properties of conjugate gradient methods for optimization. *SIAM J. Optim*., *2*, 21-42.

Gould, N.I.M., Orban, D. & Toint, P.L. (2003). CUTEr: a constrained and unconstrained testing environment, revisited. *ACM Transactions on Mathematical Software, 29*(4), 373-394.

Hager, W.W. & Zhang, H. (2006). A survey of nonlinear conjugate gradient methods. *Pacific J. Optim.*, *2*, 35-58.

Hestenes, M.R., & Stiefel, E.L. (1952). Methods of conjugate gradients for solving linear systems. *Journal Res. Nat. Bur. Stand.*, *49*, 409-436

Hu, Y. F., & Storey, C. (1991). Global convergence result for conjugate gradient Methods. *Journal of Optimization Theory Application, 71*, 399-405.

Li, D.H. and Fukushima, M. (2001). A modified BFGS method and its global convergence in nonconvex minimization. *Journal of Computational Applied. Mathematics, 129*, 15-35.

Liu, Y., & Storey, C. (1991). Efficient generalized conjugate gradient algorithms, part 1: Theory. *Journal of optimization theory and applications*. *69*, 129-137.

Nocedal, J., & Wright, S.J. (2006). *Numerical optimization*. Springer Science & Business Media.

Polyak, B.T. (1967). A general method of solving extremal problems. *Soviet Math. Doklady, 8*, 14-29.

Powell, M.J.D. (1984). Nonconvex minimization calculations and the conjugate gradient method, *Numerical Analysis (Dundee, 1983), D.F. Griffiths, ed., Lecture Notes in Mathematics*, Vol. 1066, Springer, Berlin. 122-141.

Rao, S.S. (2009). *Engineering optimization theory and practice* (4th Edition). New Jersey: John Wiley & Sons, Inc.

Sabiu, J., & Waziri, M.Y. (2017). Affective modified hybrid conjugate gradient method for large-scale symmetric nonlinear equation. *Application and Applied Mathematics, 12*, 1036-1056.

Sabiu, J., Waziri, M.Y., & Idris, A. (2017). A new hybrid Dai-Yuan and Hestenes Stiefel conjugate gradient parameter for solving system of equation. *Mayfeb Journal of Mathematics, 1*, 44-55.

Sun, W., & Yuan, Y.X. (2006). *Optimization theory and methods: Nonlinear programming.* New York: Springer.

Touati-Ahmed, D., & Storey, C. (1990). Efficient hybrid conjugate gradient techniques. *Journal of Optimization Theory Application, 64*, 379-397.

Yuan, Y. X. (1991). A modified BFGS algorithm for unconstrained optimization. *IMA Journal of Numerical Analysis, 11*, 325-332.

Zhang, J.Z. & Xu, C.X. (2001). Properties and numerical performance of quasi- Newton methods with modified quasi-Newton equation. *J. Comput. Appl. Math., 137*, 269-278.

Zoutendijk, G. (1970). Nonlinear programming computational methods, in integer and nonlinear programming. *J. Abadie ed.*, North-Holland, Amsterdam 37-86.

Appendix

List of Test Functions

S/No	Functions
1	Extended White & Holst
2	Extended Rosenbrock
3	Extended Freudenstein & Roth
4	Extended Beale
5	Raydan 1
6	Extended Tridiagonal 1
7	Diagonal 4
8	Extended Himmelblau
9	Extended Powel 1
10	Fletcher Function (Cute)
11	Extended Powel
12	Nonscomp Function (Cute)
13	Extended Denschnb Function (Cute)
14	Extended Quadratic Penelty Qp1
15	Hager
16	Extended Maratos
17	Shallo
18	Generalized Quartic
19	Quardratic Qf2
20	Generalized Tridiagonal 1
21	Generalized Tridiagonal 2
22	Power
23	Quadratic Qf1
24	Extended Quadratic Penelty
25	Extended Penalty