Available Online at http://www.recentscientific.com

CODEN: IJRSFP (USA)

International Journal of Recent Scientific Research
Vol. 8, Issue, 6, pp. 18000-18006, June, 2017

ISSN: 0976-3031

International Journal of Recent Scientific Research

DOI: 10.24327/IJRSR

Research Article

# NEW VERSION OF THE THREE-TERM CONJUGATE GRADIENT METHOD FOR SOLVE UNCONSTRAINED OPTIMIZATION

**Khalil K. Abbo[1] and Isam H. Albayaty[2]**

[1]Department of Mathematics, College of Computers Sciences and Mathematics, University of Mosul, Mosul, Iraq
[2]Department of Mathematics College of Sciences, University of Tikrit, Tikrit, Iraq

**DOI: http://dx.doi.org/10.24327/ijrsr.2017.0806.0434**

**ARTICLE INFO**

***Article History:***
Received 15th March, 2017
Received in revised form 25th April, 2017
Accepted 23rd May, 2017
Published online 28th June, 2017

**ABSTRACT**

In this paper, we Proposed a new three-term Conjugate Gradient (CG) method is suggested, the derivation of the method based on the descent property and conjugacy condition, the global convergence property is analyzed; numerical results indicate that the new proposed CG-method is well compared against other similar CG-methods in this field.



## INTRODUCTION

Consider the unconstrained optimization problem:

$$\min\left\{f(x) \;\middle|\; x \in R^n\right\} \tag{1}$$

where $f$ is a continuously differentiable function of $n$ variables. In order to introduce our new modified CG-method which is a generalization of three-term (Hestenes and Stiefel, 1952) (HS)-CG method. Let us simply recall the well-known BFCG quasi-Newton (QN) direction (Dennis and More *et al*., 1977). QN-methods for solving (1) often needed the new search direction $d_k$ at each iteration by:

$$x_{k+1} = x_k + \alpha_k d_k \tag{2}$$

where $g_k = \nabla f(x_k)$ is the gradient of $f$ evaluated at the current iterate $x_k$. One then computes the next iterate by

$$d_k = -H_k g_k \tag{3}$$

where the step size $\alpha_k$ satisfies the Wolfe-conditions

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

$$g(x_k + \alpha_k d_k)^T d_k \ge \delta_2 d_k^T g_k \tag{5}$$

where $0 \prec \delta_1 \prec 1/2$ and $\delta_1 \prec \delta_2 \prec 1$, and $H_{k+1}$ is an approximation to $\left\{\nabla^2 f(x_k)\right\}^{-1}$. The matrix $H_{k+1}$ satisfies the actual quasi-Newton condition

$$H_{k+1} y_k = \rho_k v_k \tag{6}$$

**Corresponding author:* **Khalil K. Abbo**
Department of Mathematics, College of Computers Sciences and Mathematics, University of Mosul, Mosul, Iraq

where $y_k = g_{k+1} - g_k$ , $v_k = x_{k+1} - x_k$ , $\rho_k$ is a scalar, for exact QN-condition $\rho_k = 1$.

For BFGS-update, where $H_{k+1}$ is obtained by the following BFGS formula:

$$H_{k+1} = H_K + \left(1 + \frac{y_k^T H_K y_k}{s_k^T y_k}\right) \frac{s_k s_K^T}{s_k^T y_k} - \frac{s_k y_k^T H_k + H_K y_k s_k^T}{s_k^T y_k} \tag{7}$$

If $H_k = I$ (where I is the identity matrix). Then the above BFGS method becomes the memory less BFGS method introduced by Shanno (Shanno, 1978). In this case the search direction $d_{k+1}$ can be defined as:

$$d_{k+1} = -g_{k+1} + \left(\frac{y_k^T g_{k+1}}{s_k^T y_k} - \left(1 + \frac{y_k^T y_k}{s_k^T y_k}\right) \frac{s_k^T g_{k+1}}{s_k^T y_k}\right) s_k + \frac{s_k^T g_{k+1}}{s_k^T y_k} y_k \tag{8}$$

which shows that $d_{k+1}$ possesses the following form:

$$d_{k+1} = -g_{K+1} + \beta_k s_k - \delta_k y_k \tag{9}$$

which is called the three-term CG-algorithm.

(Nazareth, 1977) proposed another CG- algorithm using a three -term recurrence formula:

$$d_{k+1} = -y_k + \frac{y_k^T y_k}{y_k^T d_k} d_k + \frac{y_{k-1}^T y_k}{y_{k-1}^T d_k} d_{k-1} \tag{10}$$

with $d_{-1} = 0$ , $d_0 = 0$.

If $f$ is quadratic convex function, then for any step length $\alpha_k$ the search direction generated by (10) are conjugate subject to the Hessian of the nonlinear function $f$, even without exact line search. In the same context, (Zhang *et al*., 2007) proposed another descent modified HSCG method with three-term, say, ZTCG where its search direction was defined as:

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{s_k^T y_k} s_k - \frac{g_{k+1}^T s_k}{s_k^T y_k} y_k \tag{11}$$

Where $d_o = -g_0$ . A remarkable property of this method is that produce descent direction i.e.

$$d_k^T g_k = -\|g_{k+1}\|^2 \tag{12}$$

The convergent properties of (11) for a convex optimization are given in (Zhang *et al*., 2009).
(Laylani Y, 2016), has developed a new algorithm in the three–term CG Which search direction was defined by

$$d_{k+1} = -g_{k+1} + \beta^{AB} s_k - \beta^{AB} \frac{g_{k+1}^T s_k}{g_{k+1}^T y_k} y_k \tag{13}$$

$$\beta^{AB} = \frac{y_k^T y_k}{g_k^T g_k}$$

Where

There are many possibilities in choosing search directions in this type of methods and it must be said that there is no single choice that is superior to others in most situations. Below we will introduce a new formulated three term CG-method which its idea is based on two important properties, i .e. the descent property and conjugacy condition.

***A New Three-Term CG-Method (New)***

Consider the search direction which is suitable for any three-term CG-type methods is defined by the following formula:

$$d_{k+1} = -g_{k+1k} + \gamma_k s_k - (\gamma_k + \beta_k^{PRP}) y_k \tag{14}$$

Where $\gamma_k = \dfrac{g_{k+1}^T(s_k - y_k)}{y_k^T(s_k - y_k)}$.

***Some Remarks on the New Method***

1. If the line search is exact i.e. $g_{K+1}^T s_k = 0$, then the search direction in (14) reduces to the classical HSCG search direction .
2. If the objective function is quadratic convex and line search is exact, then $g_{k+1}^T g_k = 0$ and $s_k^T g_{k+1} = 0$, hence, the search direction defined in (14) will reduce to the classical Conjugate-Descent method since, $g_{k+1}^T s_k = 0$, and $(g_{k+1}^T y_k)^2 = (-g_{k+1}^T g_{k+1} - g_{k+1}^T g_{k+1})^2 = (g_{k+1}^T g_{k+1})^2$.

***Outlines of the New Algorithm (New)***

**Step1.** Given an initial point $x_1 \in R^n$ and $\varepsilon > 0$. Set $k = 0$

**Step2.** Set k=k+1 and calculate $g(x_k)$.

**Step3.** Check if $\|g_k\| \le \varepsilon$, then stop.

**Step4.** Calculate step length $\alpha_k$ using Wolfe line searches (4) and (5).

**Step5.** Set $x_{k+1} = x_k + \alpha_k d_k$.

**Step6.** Calculate $g_{k+1}$ and $f_{k+1}$.

**Step7.** Calculate The search direction $d_{k+1}$ defined in (14).

**Step8.** Go to Step2.

The Descent Property of the new formula To show that the search directions of (14) are descent directions:

***Proposition***

Suppose that the line search satisfies the Wolfe condition (4) and (5) then $d_{k+1}$ given by (14) is a descent direction.

**Proof**

The proof is by induction.

1. If k=1 then $g_1^T d_1 < 0$ $d_1 = -g_1 \rightarrow < 0$.
2. Let the relation $g_k^T d_k < 0$ for all $k$.

$$d_{k+1}^T g_{k+1} = -g_{k+1}^T g_{k+1} + \gamma_k s_k \; g_{k+1}^T - (\gamma_k + \beta_k^{PRP}) g_{k+1}^T y_k$$

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

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

*Let* a= $-g_{k+1}^T g_{k+1}$ and b= $\dfrac{\|g_{k+1}^T y_k\|^2}{\|g_k\|^2}$

$$d_{k+1}^T g_{k+1} = -a + \frac{g_{k+1}^T (s_k - y_k)}{y_k^T (s_k - y_k)} s_k g_{k+1}^T - \frac{g_{k+1}^T (s_k - y_k)}{y_k^T (s_k - y_k)} g_{k+1}^T y_k - b$$

$$d_{k+1}^T g_{k+1} = -a + \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - y_k^T y_k} s_k g_{k+1}^T - \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - y_k^T y_k} g_{k+1}^T y_k - b$$

$$d_{k+1}^T g_{k+1} = -a + \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - L y_k^T s_k} s_k g_{k+1}^T - \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - L y_k^T s_k} g_{k+1}^T y_k - b$$

Let $g_{k+1}^T (s_k - y_k) > 0$ and $L y_k^T s_k > y_k^T s_k$

$$d_{k+1}^T g_{k+1} = -a + \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - L y_k^T s_k} Max(g_{k+1}^T s_k, 0) - \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - L y_k^T s_k} Max(g_{k+1}^T y_k, 0) - b$$

Let $c = \frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - L y_k^T s_k} Max(g_{k+1}^T s_k, 0)$ and $d = -\frac{g_{k+1}^T (s_k - y_k)}{y_k^T s_k - L y_k^T s_k} Max(g_{k+1}^T y_k, 0)$

$$d_{k+1}^T g_{k+1} = -a - c + d - b < 0$$

$$d_{k+1}^T g_{k+1} < 0$$

***Convergence Analysis Property***

In this section, we have to prove the basic global convergence property of the (New) proposed algorithm under the following assumptions:

The level set $S = \{x \in R^n : f(x) \le f(x_1)\}$ is bounded, i.e. there exists a positive constant $B > 0$ such that, for all:

$$\|x\| \le B \quad , \quad \forall \ x \in s$$

$$\|s_k\| \le B_1 \quad , \quad \forall \ x \in s$$

In a neighborhood $N$ of S the function f is continuously differentiable and its gradient is Lipschitz continuous , i.e. there exists a constant $L > 0$ such that:

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|$$

Under these assumptions on f, there exists a constant $c \ge 0$ such that $\|\nabla f(x)\| \le c$, for all $x \in s$ :

$$\|y_k\| \le c_1 \tag{15}$$

Observe that in the above assumption, the function f is bounded below is weaker than the usual assumption that the level set is bounded. Although the search directions generated by (14) are always descent directions, to ensure convergence of the algorithm we need to constrain the choice of the step length $\alpha_k$. Now, the following proposition shows that the Wolfe line search always gives a lower bound for the step length $\alpha_k$.

***Proposition***

Suppose that $d_K$ is a descent direction and that the gradient $\nabla f$ satisfies the Lipschitz condition $\|\nabla f(x) - \nabla f(x_k)\| \le L\|x - x_k\|$ for all $x$ on the line segment connecting $x_k$ and $x_{k+1}$, where $L$ is a positive constant. If the line search satisfies the Wolfe conditions ( ) and ( ), then:

$$\alpha_k \ge \frac{(1-\sigma)|g_k^T d_k|}{L\|d_k\|^2} \tag{16}$$

***Proof:*** See (Andrei *et al*, 2013)

To prove the global convergence we need the following lemma (Zoutendijk, 1970).

***Lemma***

Suppose that $x_1$ is a starting point for which assumptions (5.1) and (5.2) hold. Let $x_k$ be generated by the descent algorithm (New) with $\alpha_k$ satisfies the Wolfe line search conditions ( ) and ( ) then we have:

$$\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty \tag{17}$$

It easy to get from Propositions (1) that (17) is equivalent to the following equation:

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty \tag{18}$$

***Theorem***

Suppose that assumptions (1) and (2) holds, and consider the new algorithm (New), where $\alpha_k$ is computed by the Wolfe line search conditions (4) and (5) then:

$$\liminf_{k\to\infty} \|g_k\| = 0 \tag{19}$$

***Proof***

The prove is by contradiction we suppose that the conclusion is not true. Then there exist a constant $\mathrm{r} > 0$ such that:

$$\|g_k\| > \mathrm{r} \qquad \forall\ k > 0 \tag{20}$$

since $\|g_k\| \ne 0$ and with **Proposition (4. 1)** it follows that $d_k \ne 0$. Consider the search direction defined by the equation (19):

$$d_{k+1} = -g_{k+1} - \alpha_k s_k + (\gamma_k + \beta_k^{PRP}) y_k$$

$$\|d_{k+1}\|^2 = \|-g_{k+1} - \alpha_k s_k + (\gamma_k + \beta_k^{PRP}) y_k\|^2$$

$$\|d_{k+1}\|^2 \le \|g_{k+}\|^2 + \gamma_k \|s_k\|^2 + (\gamma_k + \beta_k)\|y_{k+1}\|^2$$

Let $a = \gamma_k \|s_k\|^2 + (\gamma_k + \beta_k)\|y_{k+1}\|^2$

$$\|d_{k+1}\|^2 \le \|g_{k+}\|^2 + a$$

$$\|d_{k+1}\|^2 \le \overline{\gamma}^2 + a$$

$$\|d_{k+1}\|^2 \le \frac{1}{\bar{\gamma}^2}((\bar{\gamma}^2)^2 + \bar{\gamma}^2 a)$$

Let $b=((\bar{\gamma}^2)^2 + \bar{\gamma}^2 a)$

$$\|d_{k+1}\|^2 \le \frac{1}{\bar{\gamma}^2} b$$

$$\sum_{k=1}^{\infty} \frac{1}{\|d_{k+1}\|^2} \le \frac{1}{b}\bar{\gamma}^2 \sum_{k\ge 1} 1 = \infty$$

$$\lim_{k\to\infty} \|g_k\| = 0$$

***Numerical Results***

In this section, we compare the performance of new formal $KI1$ developed A New Three-Term CG-Method. We have selected (75) large scale unconstrained optimization problem, for each test problems taken from (Andrie, 2008). For each test function we have considered numerical experiments with the number of variables $n = 100, \ldots, 1000$. These two new versions are compared with well-known conjugate gradient algorithm, the YS algorithm. All these algorithms are implemented with standard Wolfe line search conditions (4) and (5) with. In all these cases, the stopping criteria is the $\|g_k\| = 10^{-6}$. All codes are written in doble precision FORTRAN Language with F77 default compiler settings. The test functions usually start point standard initially summary numerical results recorded in the figures (1),(2),(3). The performance profile by (Dolan and More´, 2002) is used to display the performance of the developed A New Three-Term CG- algorithm with YS algorithm. Define $p = 750$ as the whole set of $n_p$ test problems and $S = 2$ the set of the interested solvers. Let $l_{p,s}$ be the number of objective function evaluations required by solver $s$ for problem $p$. Define the performance ratio as

$$r_{p,s} = \frac{l_{p,s}}{l_p^*} \tag{21}$$

Where $l_p^* = \min\{l_{p,s} : s \in S\}$. It is obvious that $r_{p,s} \ge 1$ for all $p, s$. If a solver fails to solve a problem, the ratio $r_{p,s}$ is assigned to be a large number $M$. The performance profile for each solver $s$ is defined as the following cumulative distribution function for performance ratio $r_{p,s}$,

$$\rho_s(\tau) = \frac{size\{p \in P : r_{p,s} \le \tau\}}{n_p} \tag{22}$$

Obviously, $p_s(1)$ represents the percentage of problems for which solver $s$ is the best. See(Dolan and More´, 2002). for more details about the performance profile. The performance profile can also be used to analyze the number of iterations, the number of gradient evaluations and the cpu time. Besides, to get a clear observation, we give the horizontal coordinate a log-scale in the following figures.

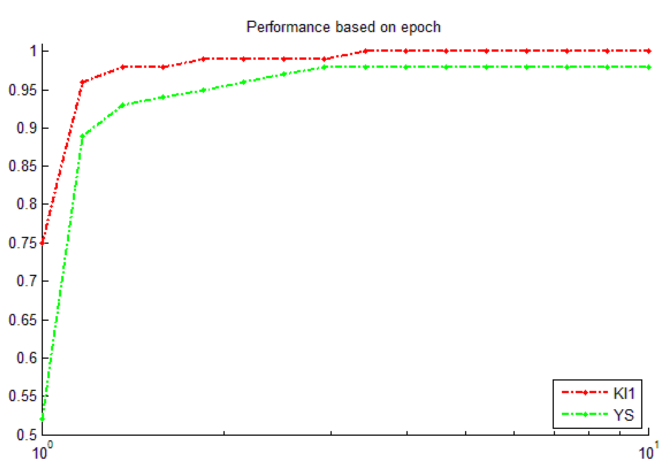


Figure 1

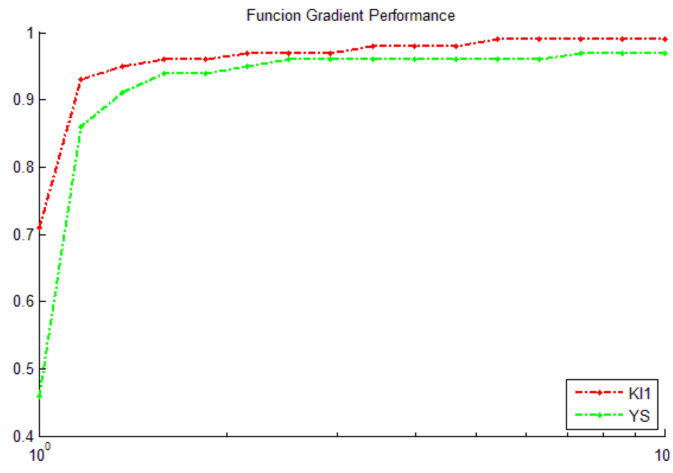


Figure 2

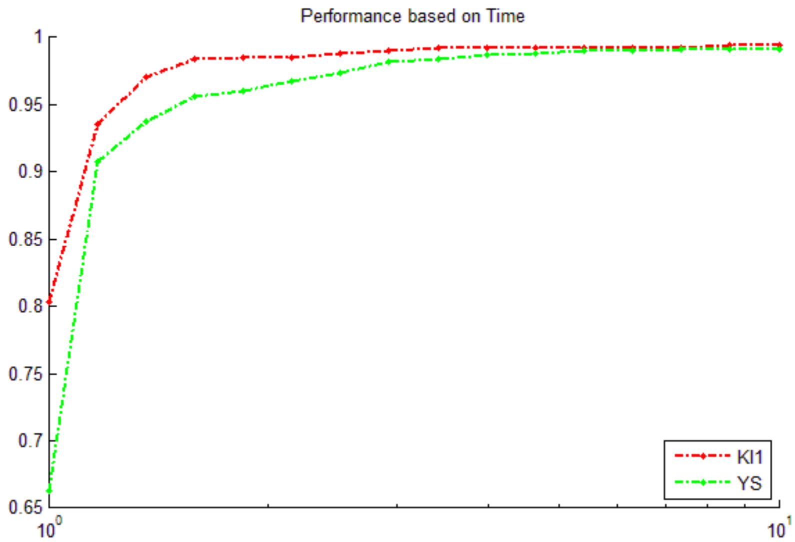


Figure 3

## References


1. Andrei N. (2008), "An Unconstrained Optimization test function collection". Adv. Model. Optimization. 10. pp.147-161.
2. Andrei, N. (2013), A simple three-term conjugate gradient algorithm for unconstrained optimization, *Journal of Computational and Applied Mathematics*, 241, 19-29.
3. Bongartz, I.; Conn, A.; Gold, N. and Toint, P. (1995), CUTE: constrained and unconstrained testing environment, ACM Trans., Math. Software, 21.
4. Dennis, J. and More, J. (1977), Quasi-Newton methods, motivation and theory, SIAM Review, 19, 46-89.
5. Dolan. E. D and Mor´e. J. J, "Benchmarking optimization software with performance profiles", *Math. Programming*, 91 (2002), pp. 201-213.
6. Fletcher, R. (1987), Practical Methods of Optimization (second edition)", John Wiley and Sons, New York.
7. Hestenes, M. and Stiefel, E. (1952), Method of conjugate gradients for solving linear systems", *J. Research Nat. Standards*, 49, 409-436.
8. Laylane, Y, A (2016). Developing New Three-Term Conjugate Gradient Algorithms with Applications in Multilayer Networks. Ph. D.Thesis, College of Computer Sciences and Mathematics, University of Mosul.
9. Nazareth, L. (1977), A conjugate direction algorithm without line search. *Journal of Optimization Theory and Applications*, 23, 373-387.
10. Shanno, D. (1978), Conjugate gradient methods with inexact searches, Math. of operation Research, 3, 244-256.
11. Zhang, J.; Xiao, Y. and Wei, Z. (2009), Nonlinear conjugate gradient methods with sufficient descent condition for large-scale unconstrained optimization. *Math. Prog. Eng.*, Article ID 243290, 16. DOI: 10.1155/2009/243290.
12. Zhang, L. Zhou, Y. (2012), A note on the convergence properties of the original three-term Hestenes-Stiefel method, AMO-Advanced Modeling and Optimization, 14, 159-163.
13. Zhang, L. Zhou, W. and Li, D. (2007), Some descent three-term conjugate gradient methods and their global convergence. Optimization Methods and Software, 22, 697- 711.