Three-Term Conjugate Gradient Algorithm for Solve Unconstrained Optimization Problems

Khalil K. Abbo[1], Hassan H. Ibrahim[2], Isam H. Albayaty[2]

[1]Department of Mathematics,
Mosul University, Mosul, Iraq
prof_khalil@uomosul.edu.iq

[2]Department of Mathematics,
Tikrit University, Tikrit, Iraq
Hassan1962pl@yahoo.com, isamhalil@yahoo.com

Received: 11.11.2018, Accepted: 21.01.2019, Published: 21.01.2019

Abstract — *In this paper, we proposed a new three–term Conjugate Gradient (CG) method. 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.*

Keywords: Three-term conjugate gradient, Descent property, Conjugacy condition, Global convergence.
Mathematics Subject Classification: 80C50, 30A40, 90C26.

1 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 [6]. (HS)-CG method. Let us simply recall the well-known BFCG Quasi-Newton (QN) direction [4]. QN-methods for solving (1) often needed the new search direction d_k at each iteration by :

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

Cite as: *K. K. Abbo, H. H. Ibrahim, I. H. Albayaty, Three-Term Conjugate Gradient Algorithm for Solve Unconstrained Optimization Problems, Journal of Multidisciplinary Modeling and Optimization 1 (2) (2018), 82-90.*

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 α_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 $\{\nabla^2 f(x_k)\}^{-1}$. The matrix H_{k+1} satisfies the actual quasi-Newton condition

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

where $y_k = g_{k+1} - g_k$, $v_k = x_{k+1} - x_k$, ρ_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:

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

If $H_k = \mathrm{I}$ (where I is the identity matrix). Then the above BFGS method becomes the memoryless BFGS method introduced by Shanno [8]. In this case the search direction d_{k+1} can be defined as:

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

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.

In [7], it is 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 α_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, in [9] it is 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 [9].

2 A New Three-Term CG-Method

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

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

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

θ_1 , θ_2 Barzilai-Borwein (BB) methods to compute the step size [3]. We use it to search for less memory.

$$d_{k+1} = -\theta_1 g_{k+1k} + \theta_1\gamma_k s_k - (\gamma_k + \beta_k^{PRP}) y_k \tag{14}$$

$$\gamma_k = \frac{(s_k - \theta_1 y_k)^T g_{k+1}}{(s_k - \theta_1 y_k)^T y_k} \quad , \quad \theta_1 = \frac{s_k^T s_k}{s_k^T y_k}$$

$$d_{k+1} = -\theta_2 g_{k+1} + \theta_2\gamma_k s_{k+1} - (\beta_k^{PRP} + \gamma_k) y_k \tag{15}$$

$$\gamma_k = \frac{(s_k - \theta_2 y_k)^T g_{k+1}}{(s_k - \theta_2 y_k)^T y_k} \quad , \quad \theta_2 = \frac{s_k^T y_k}{y_k^T y_k}$$

Outlines of the New Algorithm:

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 α_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 (14,15).

Step8. Go to **Step2**.

3 The Descent Property of the new formula

To show that the search directions of (14, 15) are descent directions:

Proposition 3.1 *Suppose that the line search satisfies the Wolfe condition (4) and (5) then* d_{k+1} *given by (14,15) is a descent direction.*

Proof: See [1].

4 Convergence Analysis Property

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

Definition 4.1 *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:*

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

Definition 4.2 *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 $x \in s$:

$$\|y_k\| \le c_1 \tag{16}$$

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, 15) are always descent directions, to ensure convergence of the algorithm we need to constrain the choice of the step length α_k . Now, the following proposition shows that the Wolfe line search always gives a lower bound for the step length α_k .

Proposition 4.3 *Suppose that* d_K *is a descent direction and that the gradient* ∇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 (4) and (5), then:*

$$\alpha_k \ge \frac{(1-\sigma)\left|g_k^T d_k\right|}{L\|d_k\|^2}. \qquad (17)$$

Proof : See [2]. To prove the global convergence we need the following lemma [11].

Lemma 4.4 *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* α_k *satisfies the Wolfe line search conditions (4) and (5) then we have:*

$$\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty. \qquad (18)$$

It easy to get from Proposition 4.1 that (18) is equivalent to the following equation:

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \qquad (19)$$

Theorem 4. 5 *Suppose that assumptions (5.1) and (5.2) holds, and consider the new algorithm (New), where* α_k *is computed by the Wolfe line search conditions (4) and (5) then:*

$$\liminf_{k\to\infty} \|g_k\| = 0 \qquad (20)$$

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

$$\|g_k\| > r \qquad \forall\ k > 0 \qquad (21)$$

since $\|g_k\| \neq 0$ and with Proposition (4.1) it follows that $d_k \neq 0$. Consider the search direction defined by the equation (20):

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

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

$$\|d_{k+1}\|^2 \le \theta_{1,2} \|g_{k+1}\|^2 + \theta_{1,2}\gamma_k \|s_k\|^2 + (\gamma_k + \beta_k^{PRP}) \|y_k\|$$

$$a = \theta_{1,2}\gamma_k \|s_k\|^2 + (\gamma_k + \beta_k^{PRP}) \|y_k\|$$

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

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

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

Let b=$(\theta_{1,2} (\overline{\gamma}^{2})^2 + \overline{\gamma}^{2} a)$

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

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

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

5 Numerical Results

In this section, we compare the performance of new formal $KI2,\ KI3$ developed A New Three-Term CG-Method. we have selected (75) large scale unconstrained optimization problem, for each test problems taken from [1]. For each test function we have considered numerical experiments with the number of variables $n = 100$,..., 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 double precision FORTRAN Language with F77 default compiler settings. The test functions usually start point standard initially, summary numerical results are recorded in the Figs. 1, 2 and 3. The performance profile by [5] is used to display the performance of the developed A New Three-Term CG- algorithm with KI1 algorithm. Define $p = 750$ as the whole set of n_p test problems and $S = 3$, the set of the inter-

ested 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 $r_{p,s} \geq 1$ 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 $r_{p,s}$,

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

Obviously, $p_s(1)$ represents the percentage of problems for which solver s is the best.

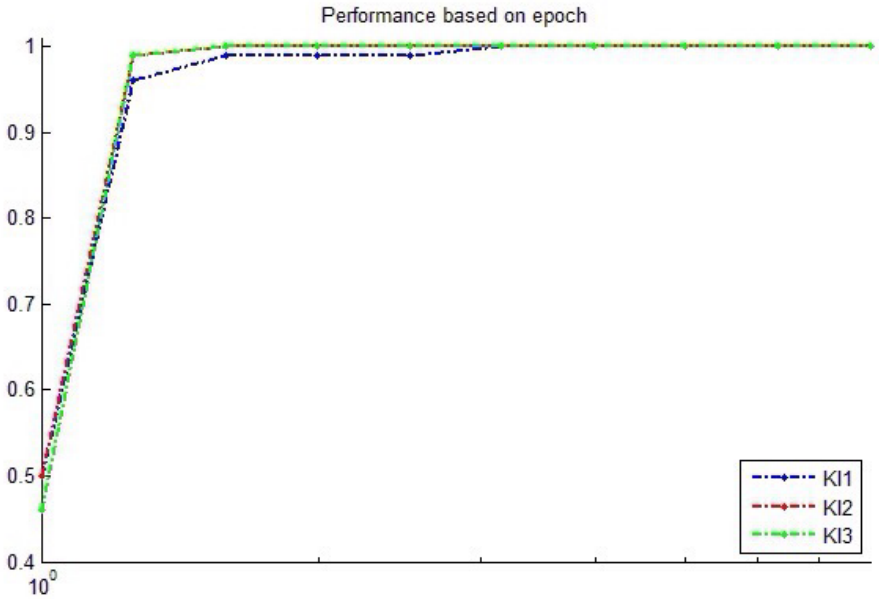

Figure 1.The performance based on the epoch

See [5] 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 Figs. 1, 2 and 3.

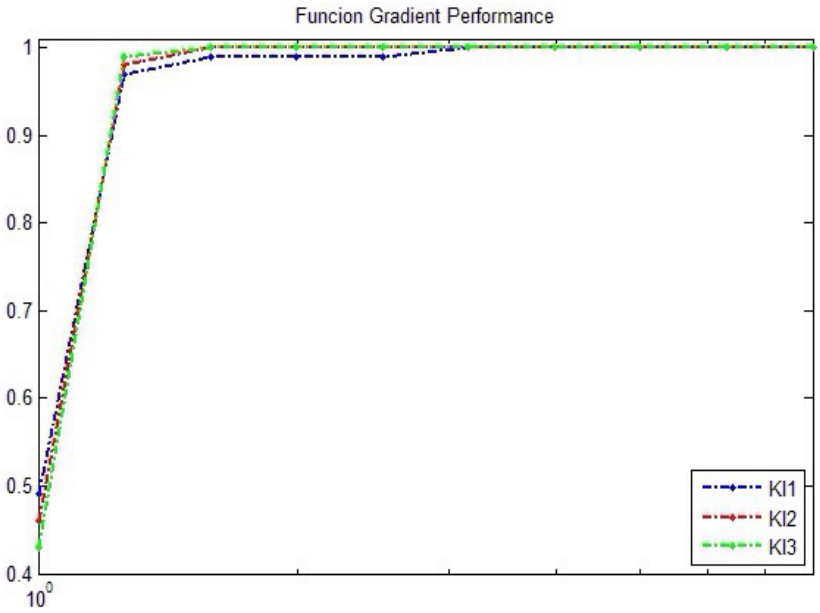

Figure 2. The graph function gradient performance.

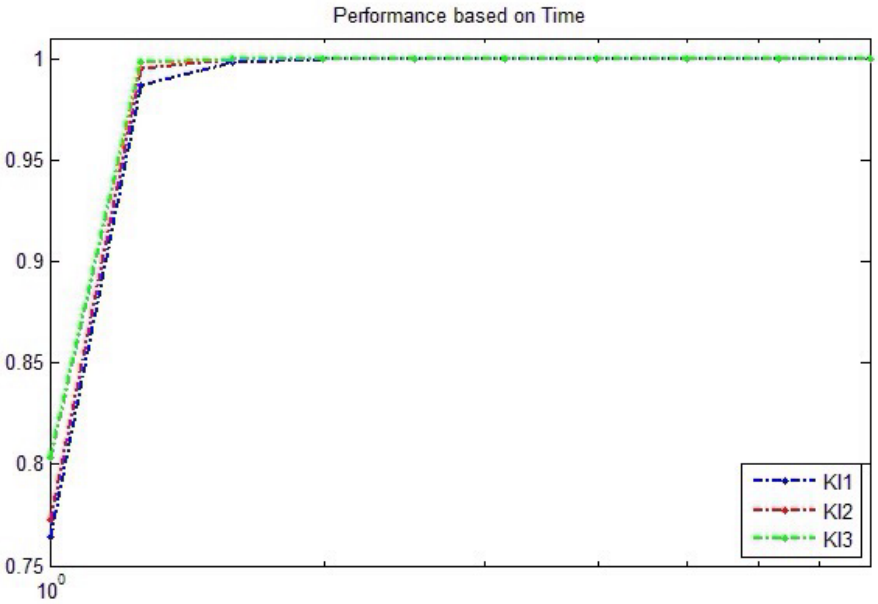

Figure 3. The graph of performance of the algorithm based on time.

References

[1] N. Andrei, An unconstrained optimization test function collection, Adv. Model. Optim., 10 (1) 2008, 147-161.

[2] N. Andrei, A simple three-term conjugate gradient algorithm for unconstrained optimization, J. Comput. Appl. Math., (241) 2013, 19–29.

[3] J. Barzilai and J.M.Borwein, Two point step size gradient methods, IMA Journal of Numerical Analysis 8(1988), 48-58.

[4] J. Dennis and J. More, Quasi-Newton methods, motivation and theory, SIAM Rev., 19(1) 1977, 46-89.

[5] D. Dolan and J. Moré, Benchmarking optimization software with performance profiles, Math. Program., 91(2) 2002, 201-213.

[6]M. Hestenes and E. Stiefel, Methods of conjugate Gradients For solving linear systems, J. Res. Nat. Bur. Stand., 49(1) 1952.

[7] L. Nazareth, A conjugate direction algorithm without line search, J. Optimiz. Theory App., 23(3) 1977, 373-387.

[8] D. Shanno, Conjugate gradient methods with inexact searches, Math. Oper. Res., 3(3) 1978, 244-256.

[9] L. Zhang and Y. Zhou, A note on the convergence properties of the original three-term Hestenes–Stiefel method, AMO-Advanced Modeling and Optimization, (14) 2012, 159–163.

[10] L. Zhang, W. Zhou and D. Li, Some descent three-term conjugate gradient methods and their global convergence, Optim. Method Softw., 22(4) 2007, 697- 711.

[11] G. Zoutendijk, Nonlinear programming, computational methods, In Integer and Nonlinear Programming, J. Abadie(Ed), North- Holand 1970.

Khalil K. Abbo, ORCID: https://orcid.org/0000-0001-5858-625X
Isam H. Albayaty, ORCID: https://orcid.org/0000-0001-5699-7741