International Journal of Multidisciplinary Research and Growth Evaluation.

A New Three-Term Conjugate Gradient Algorithm Based on the Dai-Liao Nonlinear Conjugate Gradient and the Powell Symmetric Methods

Khalil K Abbo [1*], Aynur J Namik [2]
[1] Professor, Department of Mathematics, College of computers sciences and mathematics, Mosul University, Iraq
[2] Department of Mathematics, College of computers sciences and mathematics, Mosul University, Iraq

* Corresponding Author: **Khalil K Abbo**

Article Info

ISSN (online): 2582-7138
Volume: 06
Issue: 04
July - August 2025
Received: 06-05-2025
Accepted: 08-06-2025
Published: 12-07-2025
Page No: 1014-1021

Abstract
The development of the new three – term conjugate gradient method for solving large-scale unconstrained optimization problem based on the Dai-Laio and Powell symmetric methods. The suggested method satisfies both the descent condition and the conjugacy condition. For uniformly convex function, under standard assumption the global convergence of the algorithm is proved. Finally, some numerical results of the proposed method are given.

Keywords: Unconstrained Optimization, Descent Methods, Conjugate Gradient Methods

1. Introduction

We deal with the following unconstrained optimization problems:

$$\min f(x), x \in R^n \tag{1}$$

Where $f: R^n \to R$is continuously differentiable and its gradient $g = \nabla f$ is available. For solving (1), the iterative method is widely used and it ̕s from is given by

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

Where $x_k \in R^n$ in the kth approximation to a solution of (1). $\alpha_k \in R$ is a step-length usually chosen to satisfy certain line search conditions [14]. and $d_k \in R^n$ is the search direction and defined by

$$d_{k+1} = \begin{cases} -g_{k+1} & k = 0 \\ -g_{k+1} + \beta_k d_k k \geq 1 \end{cases} \tag{3}$$

Where $\beta_k \in R$ is a parameter which characterizes the conjugate gradient method?
For general nonlinear functions, different choices of β_k lead to different conjugate gradient methods. Well-known formulas for β_k are called the Fletcher-Reeves (FR) [6], Hestenes -Stiefel (HS) [7], and Polak-Ribiere (PR) [11]. are given by

$$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \; \beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k} \; \beta_k^{PR} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}$$

Where $y_k = g_{k+1} - g_k$ and $\| . \|$denotes to $\ell_2$norm.
The line search in conjugate gradient algorithms is often based on the standared Wolfe Conditions (WC) [15]:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \rho \alpha_k g_k^T d_k \ , \tag{4}$$

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k, \tag{5}$$

Where $d_k$is a descent direction and $0 < \rho \leq \sigma < 1$. However, for some conjugate gradient algorithms, a Stronger version of the Wolfe line search Conditions (SWC) given by (4) and

$$|g_{k+1}^T d_k| \leq -\sigma g_k^T d_k \ , \tag{6}$$

Is needed to ensure the convergence and to enhance the stability.
The pure conjugacy condition is represented by the form

$$d_{k+1}^T y_k = 0 \tag{7}$$

for nonlinear conjugate gradient methods. The extension of the conjugacy condition was studied by Perry [10]. He tried to accelerate the conjugate gradient method by incorporating the second-order information into it. Specifically, he used the secant condition

$$H_{k+1} y_k = s_k \tag{8}$$

of quasi-Newton methods, where a symmetric matrix H_{k+1} is an approximation to the inverse Hessian. For quasi-Newton methods, the search direction d_{k+1} can be calculated in the form

$$d_{k+1} = -H_{k+1} g_{k+1} \tag{9}$$

By (8) and (9), the relation

$$d_{k+1}^T y_k = -(H_{k+1} g_{k+1})^T y_k = -g_{k+1}^T (H_{k+1} y_k) = -g_{k+1}^T s_k$$

Holds. By taking this relation into account, Perry replaced the conjugacy condition (7) by the condition

$$d_{k+1}^T y_k = -g_{k+1}^T s_k. \tag{10}$$

Dai and Liao [4] generalized the condition (10) to the following

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

where $t \geq 0$is a scalar. The case $t = 0$, (11) reduces to the usual conjugacy condition (7). On the other hand, the case $t = 1$, (11) becomes Perry's condition (10). To ensure that the search direction $d_k$satisfies condition (11), by substituting $d_{k+1} = -g_{k+1} + \beta_k d_k$ into (11), they had

$$-g_{k+1}^T y_k + \beta_{k+1} d_k^T y_k = -t g_{k+1}^T s_k.$$

This gives the Dai-Liao formula

$$\beta_k^{DL} = \frac{g_{k+1}^T (y_k - t s_k)}{d_k^T y_k}. \tag{12}$$

We note that the case $t = 1$ reduces to the Perry formula

$$\beta_k^{P} = \frac{g_{k+1}^T (y_k - s_k)}{d_k^T y_k}. \tag{13}$$

Furthermore, if $t = 0$, then β^{DL} reduces to the β^{HS}. The approach of Dai and Liao (DL) has been paid special attention to by many researches. In several efforts, modified secant equations have been applied to make modifications on the DL method. It is remarkable that numerical performance of the DL method is very dependent on the parameter t for which there is no any optimal choice [2]. This paper is organized as follows. In section 2 we briefly review the Three-term conjugate gradient methods. In section 3, the proposed algorithm is stated. The properties and convergent results of the new method are given in Section 4. Numerical results and one conclusion are presented in Section 5 and in Section 6, respectively.

2. Three-term Conjugate Gradient (CG) methods

Recently many researchers have been studied three- term conjugate gradient methods. For example Narushima, Yab and Ford [9] have proposed a wider class of three term conjugate gradient methods (called 3TCG) which always satisfy the sufficient descent condition. Shanno in [13] used the well-known BFGS quasi-Newton method to obtain the following three-term CG method.

$$d_{k+1} = -g_{k+1} + \left[\frac{g_{k+1}^T y_k}{s_k^T y_k} - \left(1 + \frac{\|y_k\|^2}{s_k^T y_k}\right)\frac{g_{k+1}^T s_k}{s_k^T y_k}\right] s_k + \frac{g_{k+1}^T s_k}{s_k^T y_k} y_k \quad (14)$$

Furthermore, Liu and Xu in [8] was generalized the Perry conjugate gradient algorithm (13), the search directions were formulated as follows

$$d_{k+1}^{ps} = -g_{k+1} + \left[\frac{g_{k+1}^T y_k}{s_k^T y_k} - \left(\tau_k + \frac{\|y_k\|^2}{s_k^T y_k}\right)\frac{g_{k+1}^T s_k}{s_k^T y_k}\right] s_k + \frac{g_{k+1}^T s_k}{s_k^T y_k} y_k \quad (15)$$

Where τ_k is parameter, which is symmetric Perry three-term conjugate gradient methods. When $\tau_k s_k^T y_k > 0$, the search directions defined by (15) satisfy the descent property

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

Or the sufficient descent property

$$d_{k+1}^T g_{k+1} \leq -c_0 \|g_{k+1}\|^2, \; c_0 > 0 \quad (16)$$

Notice that if $\tau_k = 1$, then (15) reduces to the (14). It is remarkable that there is no any optimal choice for τ_k, However different values used for τ_k in [3], for example

$$\tau_k = 1, \; \tau_k = c_1 \frac{y_k^T y_k}{s_k^T y_k}, \ldots$$

3. A new three-term conjugate gradient (CG) method

The aim of this section is to derive a new three-term conjugate gradient method Aynur and Khalil (AK4 say) by using Powell Symmetric (PS) method (15) and Dai and Liao (DL) CG method (3) and (12). consider the search direction given by Dai and Liao

$$d_{k+1}^{DL} = -g_{k+1} + \frac{y_k^T g_{k+1}}{s_k^T y_k} s_k - t\frac{s_k^T g_{k+1}}{s_k^T y_k} s_k \, , \quad (17)$$

Letting $t = \frac{s_k^T y_k}{\|y_k\|^2}$ in equation (17) we get

$$d_{k+1} = -g_{k+1} + \frac{y_k^T g_{k+1}}{s_k^T y_k} s_k - \frac{s_k^T g_{k+1}}{\|y_k\|^2} s_k \quad (18)$$

Now equating the equations (15) and (18) i.e

$$d_{k+1} = d_{k+1}^{PS} \, .$$

With simple algebra and with the change signal of the last term in d_{k+1}^{PS} we get

$$\tau_k = \frac{s_k^T y_k}{\|y_k\|^2} - \frac{\|y_k\|^2}{s_k^T y_k} \quad (19)$$

Substitute (19) in the equation (15) to obtain the new search direction

$$d_{k+1}^{AK4} = -g_{k+1} + \left[\frac{g_{k+1}^T y_k}{s_k^T y_k} - \frac{g_{k+1}^T s_k}{\|y_k\|^2}\right] s_k - \frac{g_{k+1}^T s_k}{s_k^T y_k} y_k \quad (20)$$

Note that, if line search is exact i.e $g_{k+1}^T s_k = 0$ then the search direction d^{AK4} reduces to the well-known Hestenes and Stiefel β^{HS}, furthermore if $g_{k+1}^T s_k = 0$ and successive gradients are orthogonal i.e $g_{k+1}^T g_k = 0$ then d^{AK4} reduces to the CD-Fletcher method defined by

$$\beta_k^{CD} = \frac{g_{k+1}^T g_{k+1}}{s_k^T g_k}$$

In the following we summarize the our AK4 algorithm.

Algorithm (AK4)

Step (1): Select a starting point $x_1 \in dom f$ and $\varepsilon > 0$, compute $f_1 = f(x_1)$ and $g_1 = \nabla f(x_1)$. Select some positive values for ρ and σ. Set $d_1 = -g_1$ and $k = 1$.

Step (2): Test for convergence. If $\|g_k\|_\infty \leq \varepsilon$, then stop x_k is optimal; otherwise go to step (3).

Step (3): Determine the step length α_k, by using the Wolfe line search conditions (4)-(5).

Step (4): Update the variables as: $x_{k+1} = x_k + \alpha_k d_k$. Compute f_{k+1} , g_{k+1}, $y_k = g_{k+1} - g_k$ and $s_k = x_{k+1} - x_k$.

Step (5): Compute the search direction as: If $y_k^T s_k \neq 0$ then $d_{k+1} = d_{k+1}^{AK4}$ else $d_{k+1} = -g_{k+1}$.

Step (6): Set $k = k + 1$ and go to step 2.

4. Convergence analysis

In this section. We investigate the global convergence property of the algorithm (AK4). For this purpose, we make the following assumptions:

- The level set $S = \{x \in R^n : f(x) \leq f(x_0)\}$ is bounded, i.e. there exists positive constant $B > 0$ such that, for all $x \in S$, $\|x\| \leq B$.
- 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 $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$, for all $x, y \in N$.

Under these assumptions on f, there exists a constant $\Gamma \geq 0$ such that $\|\nabla f(x)\| \leq \Gamma$, for all $x \in S$. Observe that the assumption that the function f is bounded below is weaker than the usual assumption that the level set is bounded. Although the search directions generated by (20) are always descent directions, to ensure convergence of the algorithm we need to constrain the choice of the step length α_k. The following proposition shows that the Wolfe line search always gives a lower bound for the step length α_k. Based on the above assumptions we shall show that our method satisfies the conjugacy condition, the sufficient descent condition, and globally convergent with Wolfe line search conditions.

Theorem (1): Suppose that the step-size α_k satisfies the standard Wolfe conditions, consider the search directions d_k generated from (20) then the search directions d_{k+1} are conjugate for all k that is.

$$d_{k+1}^T y_k = -c_0 g_{k+1}^T s_k$$

Where c_0 positive constant.

Proof:

$$y_k^T d_{k+1}^{AKTCG} = -y_k^T g_{k+1} + \left[\frac{g_{k+1}^T y_k}{s_k^T y_k} - \frac{g_{k+1}^T s_k}{\|y_k\|^2}\right] y_k^T s_k - \frac{g_{k+1}^T s_k}{s_k^T y_k} y_k^T y_k$$

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

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

By Wolfe condition $s_k^T y_k > 0$ we have

$$\therefore \frac{s_k^T y_k}{\|y_k\|^2} + \frac{\|y_k\|^2}{s_k^T y_k} = c_0 > 0$$

Therefore

$$d_{k+1}^T y_k = -c_0 g_{k+1}^T s_k \ .$$

Theorem (2): Suppose that the step-size α_k satisfies the standard Wolfe conditions (WC), consider the search directions d_k generated from (20) then the search directions d_{k+1} satisfies the sufficient descent condition $d_k^T g_k \leq -c\|g_k\|^2$, for all k.

Proof: The proof is by induction.

$$\text{If } k = 1 \Rightarrow d_1 = -g_1 \ , \therefore d_1^T g_1 = -\|g_1\|^2$$

know let $s_k^T g_k < -c\|g_k\|$ to proof for $k+1$, multiply (20) by g_{k+1}^T to get

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

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

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

By Couchy-Shwartiz inequality and Lipschitz condition we get

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

Therefore $d_{k+1}^T g_{k+1} = -c\|g_{k+1}\|^2$ Where $c = 1 + \frac{1}{L^2} > 0$.

Proposition 1 ([15,16]). 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

$$\alpha_k \ge \frac{(1-\sigma)|g_k^T d_k|}{L\|d_k\|^2}. \tag{21}$$

Proposition 2 ([12]). Suppose that assumptions (1) and (2) hold. Consider the algorithm (2) and (20), where d_k is a descent direction and α_k is computed by the general Wolfe line search (4) and (5). Then

$$\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < +\infty . \tag{22}$$

Proposition 3 ([16]). Suppose that assumptions (1) and (2) hold, and consider any conjugate gradient algorithm (2), where d_k is a descent direction and α_k is obtained by the Strong Wolfe line search (4) and (6). If

$$\sum_{k\ge 1} \frac{1}{\|d_k\|^2} = \infty, \tag{23}$$

$$\text{Then } \lim_{k\to\infty} \inf \|g_k\| = 0. \tag{24}$$

For uniformly convex functions, we can prove that the norm of the direction d_{k+1} generated by (20) is bounded above. Therefore, by Proposition 3, we can prove the following result.

Theorem (3): Suppose that assumptions (1) and (2) hold, and consider the algorithm (2) and (20), where d_k is a descent direction and α_k is computed by the strong Wolfe line search (4) and (6). Suppose that f is a uniformly convex function on S, i.e. there exists a constant $\mu > 0$ such that

$$\left(\nabla f(x) - \nabla f(y)\right)^T (x - y) \ge \mu\|x - y\|^2 \text{ for all } x, y \in N; \text{ then } \lim_{k\to\infty} \inf \|g_k\| = 0.$$

Proof: The prove is by Contradiction.

$$|\beta_k| = \left|\frac{g_{k+1}^T y_k}{s_k^T y_k} - \frac{g_{k+1}^T s_k}{\|y_k\|^2}\right| \le \frac{|g_{k+1}^T y_k|}{|s_k^T y_k|} + \frac{|g_{k+1}^T s_k|}{\|y_k\|^2}$$

Since f is uniformly convex then $s_k^T y_k \ge \mu\|s_k\|^2$where $\mu > 0$.

$$\therefore |\beta_k| \le \frac{\|g_{k+1}\| \ \ \|y_k\|}{\mu\|s_k\|^2} + \frac{\|s_k\| \ \ \|g_{k+1}\|}{\|y_k\|^2}$$

By assumption (2) and Lipschitz continuity, we have $\|y_k\| \le L\|s_k\|$.we get

$$|\beta_k| \leq \frac{\Gamma L}{\mu\|s_k\|} + \frac{\Gamma}{L^2\|s_k\|} = \Gamma\left(\frac{L}{\mu} + \frac{1}{L^2}\right)\frac{1}{\|s_k\|}$$

$$|\eta_k| = \left|\frac{g_{k+1}^T s_k}{s_k^T y_k}\right| = \frac{|g_{k+1}^T s_k|}{|s_k^T y_k|} \leq \frac{\|g_{k+1}\| \quad \|s_k\|}{\mu\|s_k\|^2} \leq \frac{\Gamma}{\mu\|s_k\|}$$

$$\therefore \|d_{k+1}\| \leq \|g_{k+1}\| + |\beta_k| \quad \|s_k\| + |\eta_k| \quad \|y_k\|$$

$$\leq \Gamma + \Gamma\left(\frac{L}{\mu} + \frac{1}{L^2}\right)\frac{1}{\|s_k\|} \quad \|s_k\| + \left(\frac{\Gamma}{\mu\|s_k\|}\right) \quad L\|s_k\|$$

$$\leq \Gamma + \Gamma\left(\frac{L}{\mu} + \frac{1}{L^2}\right) + \frac{\Gamma L}{\mu}$$

$$\leq \Gamma\left(1 + \frac{2L}{\mu} + \frac{1}{L^2}\right)$$

$$\|d_{k+1}\| \leq \Gamma b$$

Where $b = \left(1 + \frac{2L}{\mu} + \frac{1}{L^2}\right)$

$$\therefore \frac{1}{\|d_{k+1}\|} \geq \frac{1}{\Gamma b}$$

Taking the sum for both sides and considering $\|d_1\| = \|g_1\|^2 \geq \Gamma$

$$\sum_{k=0}^{\infty} \frac{1}{\|d_{k+1}\|^2} = \Gamma + \sum_{k=0}^{\infty} \frac{1}{\Gamma b} = \Gamma + \frac{1}{\Gamma b}\sum_{k=0}^{\infty} 1 = \infty$$

∴ Contradiction we have $\liminf_{k\to\infty} \|g_k\| = 0$

5. Numerical results and comparison

In this section, we report some numerical results obtained with an implementation of the AK4 algorithm. The code of the AK4 Algorithm is written in Fortran and compiled with f77 (default compiler settings), taken from N. Andrei web page. We selected 80. Large-scale unconstrained optimization test functions in the generalized or extended form presented in [1]. For each test function, we undertook ten numerical experiments with the number of variables increasing as n=100, 200,…, 1000.

The algorithm implements the Wolfe line search conditions with $\rho = 0.0001, \sigma = 0.9$ and the same stopping criterion $\|g_k\|_\infty \leq 10^{-6}$, where $\|\cdot\|_\infty$ is the maximum absolute component of a vector. In all

The algorithms we considered in this numerical study the maximum number of iterations is limited to 1000.

The comparisons of algorithms are given in the following context. Let f_i^{ALG1} and f_i^{ALG2} be the optimal values found by ALG1 and ALG2, for problem i = 1, . . ., 800, respectively. We say that, in the particular problem i, the performance of ALG1 was better than the performance of ALG2 if

$$\left|f_i^{ALG1} - f_i^{ALG2}\right| < 10^{-3}$$

and the number of iterations (iter), or the number of function-gradient evaluations (fg) or the CPU time of ALG1 was less than The number of iterations, or the number of function-gradient evaluations, or the CPU time corresponding to ALG2 respectively.

Figures (1), (2) and (3) shows the Dolan and Moré [5] (iterations (iter), function-gradient evaluations(fg) and CPU time) performance

profile of AK4 versus Dai-Liao(DL) and Powell symmetric (PS) and Fletcher(CD) and Hestenes-Stiefel(HS) conjugate gradient algorithms. In a performance profile plot, the top curve corresponds to the method that solved the most problems in a(iter) or (fg) or CPU time that was within a given factor of the best((iter) or (fg) or CPU time). The percentage of the test problems for which a method is the fastest is given on the left axis of the plot. The right side of the plot gives the percentage of the test problems that were successfully solved by these algorithms, respectively. The right is a measure of the robustness of an algorithm. When comparing AK4 with the DL and PS subject (iter, fg, CPU) as in figures (1), (2) and (3) we see that AK4 is the top performer.

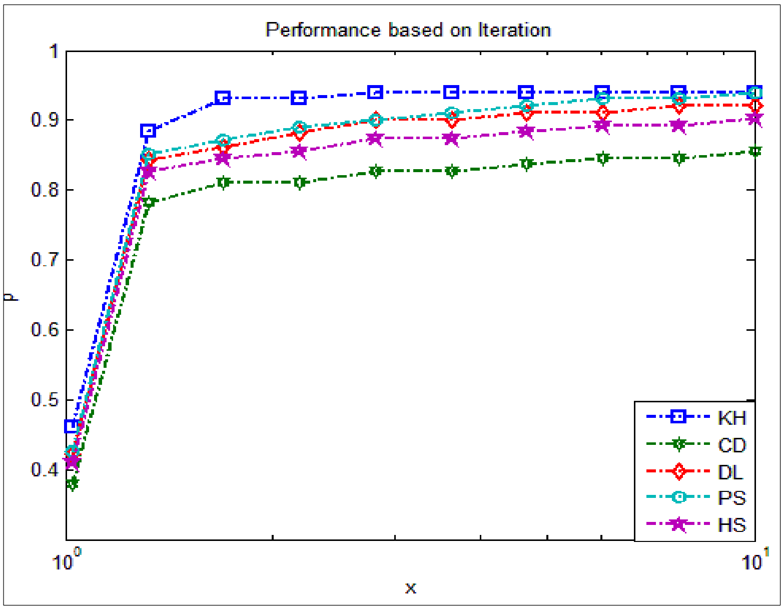

Fig 1: Performance based on iteration

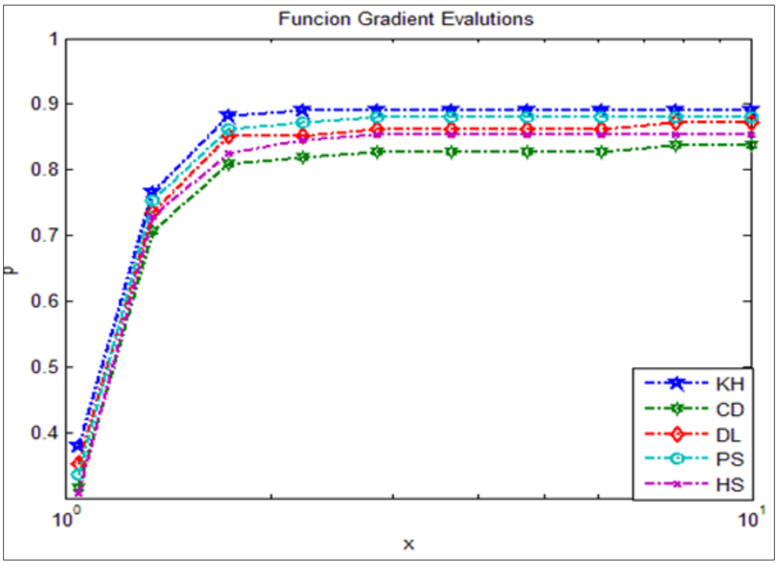

Fig 2: Performance based on Function gradient evaluation

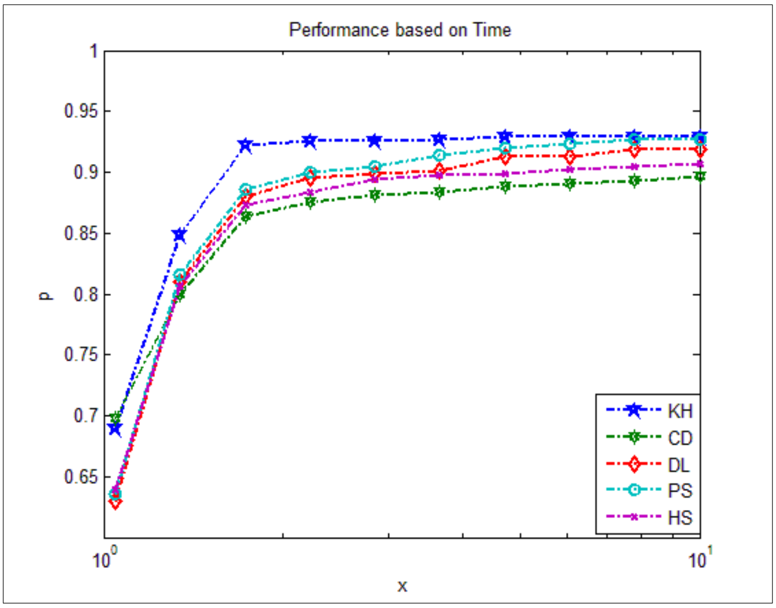

Fig 3: Performance based on Time

6. Conclusion

In this paper, a new three –term conjugate gradient algorithm, as a modification of the DL and PS methods which generates sufficient descent and conjugate directions. Under suitable assumptions our method have been shown to converge globally. In numerical experiments, we have confirmed the effectiveness of the proposed method by using performance profile.

7. References

1. Andrei N. An unconstrained optimization test function collection. Adv Model Optim. 2008;10:147-61.
2. Andrei N. Open problems in nonlinear conjugate gradient algorithms for unconstrained optimization. Bull Malays Math Sci Soc. 2011;34:319-30.
3. Dai YH, Kou CX. A nonlinear conjugate gradient algorithm with an optimal property and an improved Wolfe line search. SIAM J Optim. 2013;23(1):296-320.
4. Dai YH, Liao LZ. New conjugacy conditions and related nonlinear conjugate gradient methods. Appl Math Optim. 2001;43:87-101.
5. Dolan ED, Moré JJ. Benchmarking optimization software with performance profiles. Math Program. 2002;91:201-13.
6. Hassan BA, Jabbar HN, Laylani YA, Moghrabi IAR, Alissa AJ. An enhanced Fletcher-Reeves-like conjugate gradient methods for image restoration. Int J Electr Comput Eng. 2023;13:6268-76.
7. Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. J Res Natl Bur Stand. 1952;49(5):409-36.
8. Liu D, Xu G. Symmetric Perry conjugate gradient method. Comput Optim Appl. 2013;56:317-41.
9. Narushima Y, Yabe H, Ford JA. A three-term conjugate gradient method with sufficient descent property for unconstrained optimization. SIAM J Optim. 2011;21:212-30.
10. Perry A. A modified conjugate gradient algorithm. Oper Res. 1978;26:1073-8.
11. Polak E, Ribiére G. Note sur la convergence de directions conjuguée. Rev Fr Inform Rech Oper. 1969;16:35-43.
12. Powell MJD. Nonconvex minimization calculations and the conjugate gradient method. In: Numerical Analysis (Dundee, 1983). Lecture Notes in Mathematics. Vol. 1066. Berlin: Springer-Verlag; 1984. p. 122-41.
13. Shanno DF. Conjugate gradient methods with inexact searches. Math Oper Res. 1978;3:244-56.
14. Sun W, Yuan Y. Optimization theory and methods: nonlinear programming. New York: Springer Science, Business Media, LLC; 2006.
15. Laylani Y, Khudhur HM, Nori EM, Abbo KK. A hybridization of the Hestenes-Stiefel and Dai-Yuan conjugate gradient methods. Eur J Pure Appl Math. 2023;16(2):1059-67.
16. Zoutendijk G. Nonlinear programming, computational methods. In: Abadie J, editor. Integer and nonlinear programming. Amsterdam: North-Holland; 1970. p. 38-86.