FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 8 No. 6, December, 2024, pp 306 - 312
DOI: https://doi.org/10.33003/fjs-2024-0806-2942

GLOBAL CONVERGENCE ANALYSIS OF A MODIFIED CONJUGATE GRADIENT METHOD FOR UNCONSTRAINED OPTIMIZATION PROBLEMS

***[1]Alilu Omachi Eleojo, [2]Ibrahim Abdullahi and [2]Usman Sani**

[1]Department of Mathematics, Federal College of Education, Jama'are, Bauchi State, Nigeria.
e-mail;
[2]Department of Mathematics, Federal University Dutse (FUD), P.M.B. 7156 Dutse, Jigawa State, Nigeria.

*Corresponding authors' email: alilueleojo@gmail.com

ABSTRACT

In this paper, the global convergence analysis of a modified conjugate gradient method for solving unconstrained optimization problems was considered. We proposed a modified conjugate gradient method for solving unconstrained optimization problems that incorporates an adaptive step size selection scheme. We analyze the method's global convergence properties theoretically, demonstrating that it satisfies the sufficient descent and global convergence conditions under various assumptions. And we provide numerical experiments to illustrate its effectiveness and efficiency in solving unconstrained optimization problems. We also compare the numerical performance of the proposed method against three existing methods namely, FR, HS and PR using MATLAB simulations. The proposed method was found to perform better than FR and HS, and in competition with PR with respect to computation time, number of iteration and function evaluation.

Keywords: Optimization, Coefficient, Algorithm, Descent, Convergence

INTRODUCTION

The Conjugate Gradient (CG) methods are preferably used for solving optimization problems because they comprise a class of unconstrained optimization algorithms which are characterized by low memory requirements and strong local and global convergence properties. These properties make the CG methods attractive to mathematicians and engineers for solving large-scale optimization problems (Lu *et al*, 2015).

Given the general formula for unconstrained optimization problem

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

where $f: \boldsymbol{R^n} \rightarrow \boldsymbol{R}$ is continuously differentiable. The iterative form of CG methods for solving (1) is given by

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

$$d_{k+1} = \begin{cases} g_k & if\ k=0 \\ -g_{k+1} + \beta_k d_k & if\ k \geq 1 \end{cases} \quad (3)$$

Where d_k is the search direction defined by (3), its gradient $g_k = \nabla f(x_k)$, is a column vector and $\beta_k \in \boldsymbol{R}$ is a scalar called the CG parameter or coefficient and $\alpha_k > 0$ is a step size computed using the exact line search

$$f(x_k + \alpha_k d_k) = min_{k \geq 0} f(x_k + \alpha d_k). \quad (4)$$

Some classical formulas for CG methods are shown in Table 1

Table 1: The classical formulae for CG parameters $\boldsymbol{\beta_k}$

S/No	Name	Coefficient
1	Fletcher-Reeves (FR)	$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}$
2	Conjugate Descent (CD)	$\beta_k^{CD} = \frac{\|g_{k+1}\|^2}{d_k^T g_k}$
3	Dai-Yuan (DY)	$\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}$
4	Polak-Ribiere-Polyak (PRP)	$\beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}$
5	Liu-Storey (LS)	$\beta_k^{LS} = \frac{g_{k+1}^T y_k}{d_k^T g_k}$
6	Hestenes-Stiefel (HS)	$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$

Methods such as Fletcher-Reeves (FR), Conjugate Descent (CD) and Dai-Yuan (DY) methods are known for their strong global convergence properties. But they have weak numerical strength (Powell, 1984). Methods such as Polak-Ribiere-Polyak (PRP), Liu-Storey (LS) and Hestenes-Stiefel (HS) methods may not always converge or slow to converge but have better numerical results than FR, CD and DY methods (Powell, 1984). $\|.\|$ denotes the Euclidean norm. The methods in Table 1 behave exactly the same for quadratic function problems when line search is exact.

Research works have shown several modifications carried out using these classical methods. Among them are a modified PRP by (Wei *et al*, 2006a) and (Wei *et al*, 2006b), a modified LS by (Liu & Du, 2012), modifications of the denominators of PRP, HS and LS and modifications of both numerators and denominators of PRP, HS and LS by (Rivaie & Mamat, 2012).

A variant of LS method by (Lu *et al*, 2015) is also of great importance to study.
Justin & Ohoriemu (2024), worked on solving complex optimization problems using hybrid strategy that integrate both mathematical modeling and evolutionary algorithms
In this paper, the performance of the proposed coefficient β_k is compared with some classical CG methods.

MATERIALS AND METHODS

We propose a new β_k defined by:

$$\beta_k^{EA} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\|d_{k-1}\|^2 - \alpha g_k^T d_{k-1}} \quad \alpha = 0.5 \qquad (5)$$

We will now describe the CG algorithm and show that our proposed formula possesses the descent properties.

Algorithm 1

1. Initialization. Select $x_\circ \in \boldsymbol{R^n},\ \varepsilon > 0\ set\ k = 0$
2. **Compute $\boldsymbol{\beta_k^{EA}}$** using (5)
3. Generate $d_k\ by$ (3). If $g_k = 0$, then stop.
4. **Compute $\boldsymbol{\alpha_k}$** based on (4)
5. Variable update, $x_{k+1} = x_k + \alpha_k d_k.compute\ f(x_{k+1}), g_{(k+1)}$
6. Test for convergence ans stopping criterion. If $\|g_k\| \le \varepsilon, then\ stop.$
   Otherwise, set k = k+1 and go to step 2.

Now, we will study the global convergence of β_k^{EA} beginning with the sufficient descent condition.
The sufficient descent condition is given by

$$g_k^T d_k \le -c\|g_k\|^2 \quad \forall\ k \ge 0, c > 0 \qquad (6)$$

Using exact line search in (4) we will use the lemma below to show that (5) satisfies the descent condition in (6)
Lemma1; let the sequences $\{x_k\}\ and\ \{d_k\}$ generated by the methods of (2) and (3), be determined by exact line search in (4), then $g_k^T d_k \le -\|g_k\|^2$ holds true.
Proof; using the principle of mathematical induction to obtain the conclusion, if $k = 1, g_1^T d_1 = -c\|g_1\|^2$. Hence, (6) holds true. Next, we show that it holds true for $k \ge 1$
Multiply (3) by g_{k+1} to get

$$d_{k+1}^T g_{k+1} = g_{k+1}\left(-g_{k+1} + \beta_k^{EA} d_k^T\right)$$
$$= -\|g_{k+1}\|^2 + \beta_k^{EA} d_k^T g_{k+1} \qquad (7)$$
$$= -\|g_{k+1}\|^2 + \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\|d_{k-1}\|^2 - g_k^T d_{k-1}} g_{k+1}^T d_k$$
$$= -\|g_{k+1}\|^2 + \frac{\|g_k\|^2 - \frac{\|g_k\|^2 \|g_{k-1}\|}{\|g_{k-1}\|}}{\|d_{k-1}\|^2 - \alpha g_k^T d_{k-1}} g_{k+1}^T d_k \qquad where\ \alpha = 0.5$$
$$= -\|g_{k+1}\|^2 + \frac{\|g_k\|^2 - \|g_k\|^2}{\|d_{k-1}\|^2 - 0.5 g_k^T d_{k-1}} g_{k+1}^T d_k \qquad (8)$$

For exact line search, we have $g_{k+1}^T d_{k+1} = 0, substitute$ that into (8) to have

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2 \qquad (9)$$

And this shows that the condition holds true for $k + 1$.
We make the following assumptions in order to prove and establish the global convergence of our proposed formula

Assumption A

(i) The level set M = $\{x \in R^n : f(x) \le f(x_0)\}$ is bounded.
(ii) In some neighborhood N of M, the function is continuously differentiable and its gradient is Lipschitz continuous, that is, there exist a constant $L > 0\ such\ that$

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|\ for\ all\ x.y \in N \quad (10)$$

(iii) Suppose k is sufficiently large,
then $0 < g_{k+1}^T g_k \le 2 g_{k+1}^T g_{k+1}$ (11)
Note (i) Suppose $\{f(x_k)\}$ is decreasing, then we are sure that the sequence $\{x_k\}$ generated by our algorithm is located in a bounded region (assumption 1(i)). As a result, assumption 1(ii) shows that there exist a constant $\gamma > 0\ such\ that$

$$\|\nabla f(x)\| \le \gamma \ \ \forall\ x \in M \qquad (12)$$

Thus, the sequence is bounded.
Note (ii) The first inequality of (11) requires that the angle between $g_{k+1}\ and\ g_k$ should be acute for k sufficiently large. Recall that g_{k+1} is an approximation of g_k, then it is trivial for the inequality to hold true. Also, in the second inequality from (11), it follows that
$g_{k+1}^T g_k \le 2 g_{k+1}^T g_{k+1}$ is equivalent to

$$\|g_{k+1}\| \ge \frac{1}{2}\|g_k\| \cos\theta \qquad (13)$$

Where θ is the angle between $g_{k+1}\ and\ g_k.$
Now, if k is large enough,
$\|g_{k+1}\| \in \left(\frac{1}{2}\|g_k\|, \|g_k\|\right)$, then (13) holds true.
Furthermore, if $\frac{\pi}{2} > \theta \ge \frac{\pi}{3}$ and
$\|g_{k+1}\| \in \left(\frac{1}{4}\|g_k\|, \|g_k\|\right)$, then assumption 1(iii) holds true.

Lemma 2: Suppose the sequences $\{x_k\}\ and\ \{d_k\}$ are generated by the algorithm and for β_k^{EA} in (5), then, $\beta_k^{EA} \ge 0$.
Proof; we show that β_k^{EA} is always not negative.
We can simplify β_k^{EA} using (11) to have

$$\beta_k^{EA} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\|d_{k-1}\|^2 - g_k^T d_{k-1}} \le 2\frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|^2 g_{k-1}^T}{\|g_{k-1}\|}}{\|d_{k-1}\|^2 - g_k^T d_{k-1}} \le \frac{2(\|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2 \|g_{k-1}\|}{\|g_{k-1}\|})}{\|d_{k-1}\|^2 - g_k^T d_{k-1}} \le \frac{2\|g_{k+1}\|^2 + 2\|g_{k+1}\|^2}{\|d_{k-1}\|^2 - g_k^T d_{k-1}} \le \frac{4\|g_{k+1}\|^2}{\|d_{k-1}\|^2 - g_k^T d_{k-1}} \qquad (14)$$

Therefore, $\beta_k^{EA} \ge 0$
Lemma 3; Supposed assumption 1 holds and $\{x_k\}$ is generated by the algorithm where $\{d_k\}$ satisfies (7) and α_k satisfies (4), then,

$$\sum_{k\ge1} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < +\infty \qquad (15)$$

From Lemma 3, we can have the following theorem
Theorem 1. Suppose assumption 1 holds true, let $\{x_k\}$ and $\{d_k\}$ be generated by (2) and (3), and the algorithm with β_k^{EA}, α_k is obtained by (4), then,

$$\lim_{k\to\infty} inf\|g_k\| = 0 \qquad (16)$$

Proof, (by contradiction)
Suppose theorem 1 is not true, then there exist a constant
$m > 0\ such\ that\ \|g_k\| \ge m\ \forall\ k \ge 0$ (17)
From (3), we have that $\left(\beta_k^{EA} d_k\right)^2 = (d_{k+1} + g_{k+1})^2$ (18)

$$\left(\beta_k^{EA}\right)^2 \|d_k\|^2 = \|d_{k+1}\|^2 + \|g_{k+1}\|^2 + 2 d_{k+1}^T g_{k+1} \qquad (19)$$

$$\|d_{k+1}\|^2 = \left(\beta_k^{EA}\right)^2 \|d_k\|^2 - 2d_{k+1}^T g_{k+1} - \|g_{k+1}\|^2 \qquad (20)$$

From (14) and (20), we have

$$\|d_{k+1}\|^2 = \left(\frac{4\|g_{k+1}\|^2}{\|d_{k-1}\|^2 - g_k^T d_{k-1}}\right)^2 \|\mathrm{d_k}\|^2 - 2\mathrm{d}_{\mathrm{k+1}}^{\mathrm{T}} \mathrm{g}_{\mathrm{k+1}} - \|\mathrm{g}_{\mathrm{k+1}}\|^2 \qquad (21)$$

As earlier proved, sufficient descent condition holds true, then from (6) and (21), we have

$$\frac{16\|g_{k+1}\|^4}{\|d_{k-1}\|^4 - 2g_k^T d_{k-1}\|d\|^2 + \|g_k\|^2\|d_{k-1}\|^2} \|d_k\|^2 + 2c\|g_{k+1}\|^2 - \|g_{k+1}\|^2$$

$$= \frac{16\|g_{k+1}\|^4}{\|d_{k-1}\|^4 - 2g_k^T d_{k-1}\|d_{k-1}\|^2 + \|g_k\|^2\|d_{k-1}\|^2} \|d_k\|^2 + (2c-1)\|g_{k+1}\|^2 \qquad (22)$$

Multiply both sides of (21) by $\frac{\|g_{k+1}\|^2}{\|d_{k+1}\|^2}$ to have

$$\|d_{k+1}\|^2 \frac{\|g_{k+1}\|^2}{\|d_{k+1}\|^2} = \frac{\|g_{k+1}\|^2}{\|d_{k+1}\|^2}\left(\frac{16\|g_{k+1}\|^4}{\|d_{k-1}\|^4 - 2g_k^T d_{k-1}\|g_k\|^2 + \|g_k\|^2\|d_{k-1}\|^2}\right)\|d_k\|^2 + \frac{\|g_{k+1}\|^2}{\|d_{k+1}\|^2}(2c-1)\|g_{k+1}\|^2$$

$$\|d_{k+1}\|^2 \frac{\|g_{k+1}\|^2}{\|d_{k+1}\|^2} = \frac{\|g_{k+1}\|^4}{\|d_{k+1}\|^2}\left(\frac{16\|g_{k+1}\|^2}{\|d_{k-1}\|^4 - 2g_k^T d_{k-1}\|g_k\|^2 + \|g_k\|^2\|d_{k-1}\|^2}\|d_k\|^2 + (2c-1)\right)$$

$$\|g_{k+1}\|^2 \le \frac{\|g_{k+1}\|^4}{\|d_{k+1}\|^2} \qquad (23)$$

$$\frac{\|d_{k+1}\|^2}{\|g_{k+1}\|^4} \le \frac{1}{\|g_{k+1}\|^2} \qquad (24)$$

Hence,

$$\frac{\|d_{k+1}\|^2}{\left(g_{k+1}^T d_{k+1}\right)^2} \le \sum_{i=1}^{k} \frac{1}{\|g_i\|^2} \le \frac{k}{m^2}.$$

Furthermore, $\frac{\left(g_{k+1}^T d_{k+1}\right)^2}{\|d_{k+1}\|^2} \ge m^2 \sum_{k\ge 0} \frac{1}{k} = +\infty$

This contradicts the Zoutendijk condition (Zoutendijk, 1970) in (15). Thus, proof is completed.
The proof of this theorem shows that our proposed coefficient in (5) converges globally.

RESULTS AND DISCUSSION

The presentation of the simulation results on the test problems for our proposed method where $\beta_k = \beta_k^{EA}$ against some existing methods in the literature are done here. We consider some test problems from (Andrei, 2004) to validate the numerical strength of our proposed method versus some existing methods, using exact line search conditions in (4) as shown in Table 2.

Table 2: A list of test functions

S/NO	Function	Dimensions	Initial points
1	Extended Penalty	50, 100	9, 12
2	Diagonal 2	2	(20, 20, 20, 20)
3	Extended Tridiagonal 1	2, 200	2, 8
4	Generalized Tridiagonal 2	4, 12, 120	-1, -1, 10
5	Diagonal 4	2, 4, 200	1,1,..1, 7
6	Extended Cliff	2, 4, 12	0,2,…,6,3
7	Extended Hiebert	2	21
8	Extended Tridiagonal 2	4, 10, 200	1, 1, 1
9	TRIDIA	4, 8	4, 5
10	NONDQUAR	5	3
11	DQDRTICF	5, 10, 50, 500	1,2,…,3, 5, …1,2
12	DIXMAANA	4, 12, 32	13, 13, 13
13	DIXMAANB	12, 32	8, 13
14	DIXMAANC	4, 12, 32, 400	2, 2, 3, 8
15	DIXMAAND	12, 32, 400	3, 13, 13
16	DIXMAANL	4, 12, 32, 400	8, 3, 2, 8
17	Partial Perturbed Quadratic	4, 12, 120	0.5, 1.5, 0.5
18	Broyden Tridiagonal	4, 12, 400	-3, -3, 3
19	Almost Perturbed Quadratic	10, 20	0.5, 4.5
20	Tridiagonal Perturbed Quadratic	6, 12, 18	2.5, 6.5, 0.5
21	HIMMELBHA	4	(0,2,0,2)
22	STAIRCASE	4, 32	4, 1
23	Dixon 3DQ	4	1
24	DenschnB	400	4
25	BIGGSB 1	4, 32, 40, 400	3, 3, 3, 5

The parameters such as number of iterations (it), number of function evaluations (nf) and CPU time (t) were considered to evaluate the computational capability of the proposed method β_k^{EA} as compared with FR, HS and PR. For each test problem, the stopping criterion is taken as $\|g_k\| \le \varepsilon$, where $\varepsilon = 10^{-5}$.

We implemented the method using MATLAB R2015b(8.6.0.267246 on CP computer. Tables 3, 4 and 5 show the simulation results of the proposed method against some existing methods (FR, HS and PR). Where (-) implies failure in numerical computation.

Table 3: Numerical results of EA, PR, HS and FR for Problems 1-11

Prob.	Dim.	EA			PR			HS			FR		
		it	nf	t	it	nf	t	it	nf	t	it	nf	t
1	50	2	3	**0.05**	2	3	0.06	-	-	-	2	3	0.06
1	100	2	3	**0.05**	2	3	0.08	-	-	-	2	3	0.06
2	2	2	4	0.05	2	4	0.03	2	4	0.05	2	4	0.08
3	2	10	17	0.28	10	22	0.22	10	22	0.17	24	43	0.59
3	200	2	3	**0.11**	2	3	0.11	2	3	0.13	2	3	0.11
4	4	14	15	0.37	15	16	0.33	15	16	0.36	15	16	0.34
4	12	23	24	**0.47**	24	25	0.56	24	25	0.61	23	24	0.56
4	120	17	18	0.55	19	20	0.48	18	19	0.52	18	19	0.55
5	2	2	3	0.03	2	3	0.02	2	3	0.03	2	3	0.05
5	4	2	3	**0.03**	2	3	0.03	2	3	0.06	2	3	0.06
5	200	2	3	**0.02**	2	3	0.05	2	3	0.05	2	3	0.06
6	2	3	18	**0.08**	3	18	0.09	3	18	0.09	3	18	0.11
6	4	3	18	0.09	3	18	0.08	3	18	0.06	3	18	0.14
6	12	3	18	**0.09**	3	18	0.13	3	18	0.09	3	18	0.09
7	2	55	62	1.25	50	56	1.09	45	53	1.05	69	71	1.41
8	4	3	5	**0.03**	5	7	0.08	5	7	0.11	3	5	0.06
8	10	3	5	**0.09**	3	5	0.09	3	5	0.09	3	5	0.09
8	200	3	5	**0.06**	3	5	0.14	3	5	0.16	3	5	0.16
9	4	4	5	0.06	4	5	0.05	4	5	0.03	4	5	0.09
9	8	8	9	**0.13**	8	9	0.13	8	9	0.13	8	9	0.13
10	5	53	89	1.20	52	75	1.17	41	60	0.91	53	114	1.17
11	5	5	6	0.11	5	6	0.03	5	6	0.06	5	6	0.08
11	10	5	6	0.11	5	6	0.09	5	6	0.08	5	6	0.09
11	50	5	6	**0.08**	5	6	0.14	5	6	0.11	5	6	0.11
11	500	5	6	**0.03**	5	6	0.06	5	6	0.09	5	6	0.13

Table 4: Numerical results of EA, PR, HS and FR for Problems 12-19

Prob.	Dim.	EA			PR			HS			FR		
		it	nf	t	it	nf	t	it	nf	t	it	nf	t
12	4	11	**12**	0.30	8	9	0.20	8	9	0.16	6	7	0.13
12	12	11	**12**	0.28	8	9	0.19	8	9	0.19	7	8	0.16
12	32	11	**12**	0.33	8	9	0.23	8	9	0.28	7	8	0.22
13	12	8	9	**0.13**	7	8	0.22	7	8	0.16	9	10	0.19
13	32	9	10	0.30	9	10	0.31	9	10	0.28	9	10	0.22
14	4	5	6	0.14	5	6	0.13	5	6	0.19	5	6	0.14
14	12	6	7	**0.14**	6	7	0.17	6	7	0.16	6	7	0.16
14	32	7	8	**0.22**	7	8	0.28	7	8	0.22	8	9	0.23
14	400	12	13	**0.97**	12	13	1.05	12	13	1.05	21	22	1.34
15	12	8	9	0.20	8	9	0.27	8	9	0.22	7	8	0.19
15	32	14	15	0.44	13	14	0.42	13	14	0.39	11	12	0.38
15	400	22	23	2.05	22	23	2.02	22	23	2.05	27	28	2.22
16	4	13	14	0.34	12	13	0.38	12	13	0.28	-	-	-
16	12	8	9	0.22	7	8	0.19	7	8	0.19	7	8	0.19
16	32	6	7	**0.14**	5	6	0.16	5	6	0.20	9	10	0.23
16	400	13	14	**1.04**	13	14	1.19	13	14	1.27	23	24	1.91
17	4	4	5	**0.06**	4	5	0.09	4	5	0.09	4	5	0.09
17	12	12	13	0.23	12	13	0.25	12	13	0.22	12	13	0.25
17	120	40	41	**1.00**	40	41	1.05	40	41	1.13	40	41	1.02
18	4	17	18	0.33	15	16	0.36	15	16	0.31	17	18	0.36
18	12	26	27	0.56	26	27	0.61	26	27	0.59	21	22	0.42
18	400	23	24	0.72	23	23	0.75	23	24	0.67	26	27	0.83
19	10	10	11	**0.16**	10	11	0.25	10	11	0.20	-	-	-
19	20	20	21	0.45	20	21	0.45	20	21	0.42	-	-	-

Table 5: Numerical results of EA, PR, HS and FR for Problems 20-29

Prob.	Dim.	EA			PR			HS			FR		
		it	nf	t	it	nf	t	it	nf	t	it	nf	t
20	6	6	7	0.13	6	7	0.14	6	7	0.11	6	7	0.13
20	12	12	13	**0.17**	12	13	0.31	12	13	0.30	12	13	0.30
20	18	18	19	0.39	18	19	0.38	18	19	0.36	18	19	0.38
21	4	8	12	0.23	5	10	0.08	5	10	0.11	-	-	-
22	4	3	4	**0.05**	3	4	0.06	3	4	0.05	3	4	0.05
22	32	86	87	**1.67**	86	87	1.79	86	87	1.69	89	90	1.77
23	4	14	15	0.30	11	12	0.23	11	12	0.30	15	16	0.28
24	400	6	7	0.25	6	7	0.20	6	7	0.31	5	6	0.20
25	4	2	3	**0.02**	2	3	0.02	2	3	0.03	2	3	0.02
25	32	16	20	**0.30**	16	20	0.33	16	20	0.34	16	20	0.31
25	40	20	25	**0.33**	20	25	0.33	20	25	0.38	20	25	0.34
25	400	981	992	**20.67**	1221	1230	24.28	1448	1449	30.59	921	927	18.30
26	4	8	9	**0.16**	8	9	0.19	8	9	0.17	8	9	0.16
26	40	7	8	0.14	7	8	0.11	7	8	0.16	7	8	0.14
26	400	6	7	0.14	6	7	0.18	6	7	0.13	6	7	0.13
27	4	11	12	**0.22**	11	12	0.22	11	12	0.25	10	11	0.22
27	32	13	14	**0.23**	13	14	0.23	13	14	0.27	13	14	0.27
27	40	25	26	0.55	24	25	0.45	24	25	0.48	31	32	0.64
27	400	15	16	0.36	15	16	0.31	15	16	0.36	15	16	0.33
28	4	169	170	3.69	170	171	3.77	167	195	3.63	171	172	3.73
28	40	33	34	0.73	33	34	0.72	32	33	0.81	34	35	0.73
28	400	343	**344**	5.59	245	246	5.72	186	187	4.39	206	207	4.63
29	4	10	11	**0.16**	10	11	0.25	10	11	0.20	10	11	0.19
29	32	126	127	2.67	125	126	2.52	126	127	2.81	162	163	3.36
29	40	156	157	**3.29**	175	176	3.86	177	178	3.77	161	162	3.34
29	400	321	322	7.38	321	322	6.93	1154	1155	25.44	487	488	10.36

Graphically, performance of the proposed algorithm versus PR, HS and FR methods are produced in Figures 1 to 3 based on the number of function evaluation (nf), number of iterations (it) and the computation time (t) using performance profile by (Dolan & More, 2002).

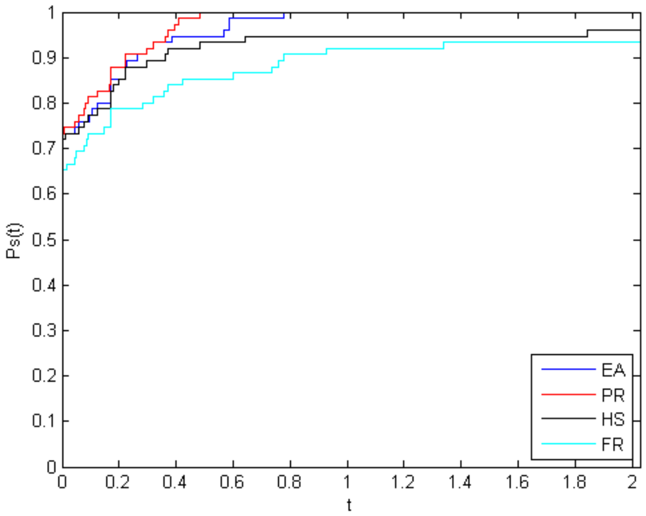

Figure 1: performance profile based on function evaluation of EA Versus PR, HS and FR

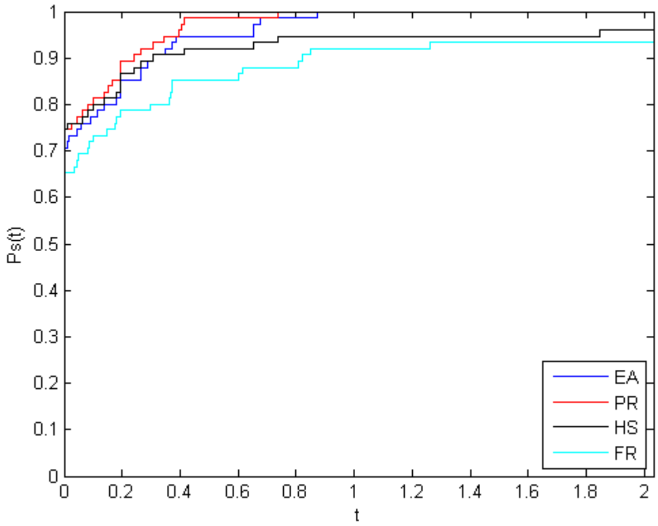

Figure 2: Performance profile based on number of iteration (it) of EA versus PR, HS and FR

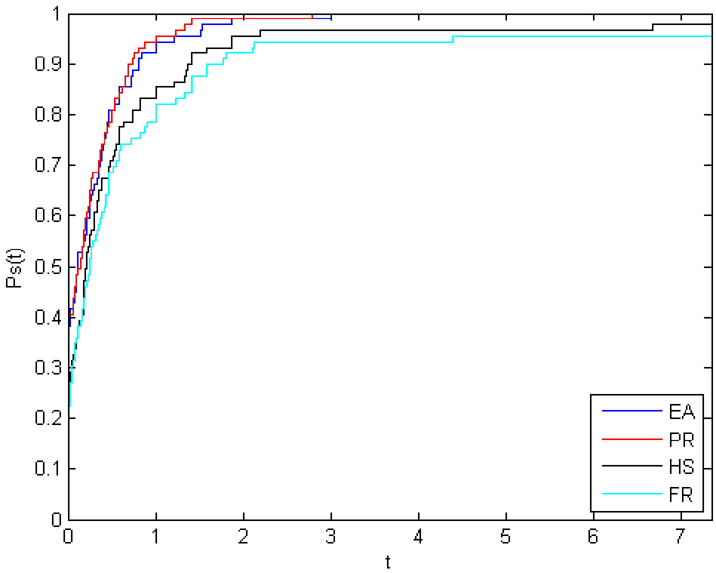

Figure 3: performance profile base on CPU time (t) of EA versus PR, HS and FR

From Tables 3, 4 and 5, experimentation of our proposed algorithm was done against some existing algorithms with respect to number of iteration (it), number of function evaluation (nf) and CPU time (t). The bold figures indicate that either our proposed algorithm EA converge faster against other CG coefficients under consideration or converge at the same time. The symbol (-) on the Tables 3, 4 and 5 means that numerical computation failed there.

From Tables 3, 4 and 5, we can deduce that our coefficient converged in about 22 functions faster than the other algorithms and converged at the same time with them in about 11 functions.

Clearly, from these tables, we can see that in many places our proposed coefficient converges faster than the other algorithms. In problem 12, our algorithm solved more functions than the other algorithms. Also in problem 28 (dimension 400), our algorithm solved more functions than the others. But in some cases, the other methods solved more number of functions than our algorithm.

Also, at some points, our algorithm EA and the other methods located their minima at the same time or almost at the same time.

In Fig.1, our proposed CG coefficient (on blue path) reached the minima only after PR with a probability of about a 100%. The other two coefficients HS and FR, are below 0.95 (94%). This clearly shows that our coefficient outperformed HS and FR with respect to number of function evaluation. And it is also in competition with PR.

In Fig.2, the proposed method EA (on blue path), wins just like PR (on red path) with the probability of about 100% compared to HS and FR with about 96% in terms of number of iterations..

In Fig.3, our proposed CG coefficient outperformed HS and FR and in competition with PR to reach 1. With respect to CPU time (t), the probability that our CG coefficient EA and PR are the winner is about 100% as displayed by the performance profile as against 95% of HS and FR.

CONCLUSION

In this paper, we propose a new conjugate gradient coefficient β_K^{EA} that is a modification of HS for solving unconstrained optimization problems. This proposed coefficient possesses the descent properties with the exact line search condition. We established the global convergence of the method using the Zoutendijk condition in (Zoutendijk, 1970). We experiment our formula on some test functions and results obtained showed that our algorithm EA is efficient and effective.

REFERENCES

Andrei, N. (2004). Test functions for unconstrained optimization. *Research Institute for Informatics, Centre for Advanced Modeling and Optimization, 1-15*

Dai, Y. H., & Yuan, Y. (1999); A Nonlinear Conjugate Gradient Method with a strong Global Convergence Property. *SIAM journal on optimization, 10(1), 177482, http;//dx.doi.org*

Dolan, E. D. & More, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical programming.* 91(2), 201-213.

Fletcher, R. & Reeves, C. M. (1964); Function Minimization by Conjugate Gradients, 2(149-154), http://dx.doi.org/10.1098/comin/7.2.149

Ibrahim, A. & Rohanin, A. (2015). Convergence analysis of a new conjugate gradient method for unconstrained optimization. *Applied Mathematical Sciences.* 140(9), 6969-6984.

Ibrahim, A., Rohanin, A. (2016). Global Convergence Analysis of a new Hybrid Conjugate Gradient method for unconstrained optimization problems, *Malaysian Journal of Fundamental and Applied Sciences 13(2), 40-48*

Justin, O. O. & Ohoriemu, B. O. (2024). A hybrid approach to solving complex optimization problems using evolutionary algorithms and mathematical modeling. *FUDMA Journal of Sciences.* 8(3), 443-449.

Liu, J. & Du, X. (2012); Global Convergence of a Modified Ls Method, *mathematical problems in Engineering, 910303/10.1156/910303*

Lu, Y., Li, W., Zhang, C. & Yang, Y. (2015). A class of new hybrid conjugate gradient method for unconstrained optimization. *Journal of Information and Computational science.* 12(5), 1941-1949.

Powell, M. J. (1984). Nonconvex minimization calculations and the conjugate gradient method. *Springer*

Rivaie, M., Fauzi, M. & Mamat, M. (2012). New modifications of conjugate gradient coefficient with global convergence properties. *Humanities, Science and Engineering Research (SHUSER), 2012 IEEE Symposium on. IEEE, 625-629*

Wei, Z., Li, G. & Qi, L. (2006a). New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems. *Applied Mathematics and Computation.* 179(2), 407-430

Wei, Z., Yao, S. & Liu, L. (2006b). The convergence properties of some new conjugate gradient methods. *Applied Mathematics and Computation.* 183(2), 226-235.

Yuan, G., Meng, Z. & Li, Y. (2016). A modified Hestenes and Stiefel conjugate gradient algorithm for large scale nonsmooth minimizations and nonlinear equations. *Journal of Optimization Theory and Applications.* 168(1), 129-152.

Zoutendijk, G. (1970). Nonlinear programming, computational methods. *Integer and nonlinear programming.* 143(1), 37-86.