Earthline Journal of Mathematical Sciences
E-ISSN: 2581-8147
Volume 13, Number 2, 2023, Pages 353-368
https://doi.org/10.34198/ejms.13223.353368

A Sufficient Descent Dai-Liao Type Conjugate Gradient Update Parameter

Oluwaseun B. Onuoha

Department of Mathematical Sciences, Adekunle Ajasin University, Akungba-Akoko, Nigeria
e-mail: oluwaseun.akinduko@aaua.edu.ng

Abstract

In recent years, conjugate gradient methods have gained popularity as efficient iterative techniques for unconstrained optimization problems without the need for matrix storage. Based on the Dai-Laio conjugacy condition, this article presents a new hybrid conjugate gradient method that combines features of the Dai-Yuan and Dai-Laio methods. The proposed method addresses the numerical instability and slow convergence of the Dai-Yuan method as well as the potential poor performance of the Dai-Laio method in highly non-linear optimization problems. The hybrid method solves optimization problems with faster convergence rates and greater stability by combining the advantages of both methods. The resulting algorithm is shown to be more effective and reliable, and theoretical analysis reveals that it has sufficient descent properties. The proposed method's competitive performance is shown through a number of benchmark tests and comparisons with other approaches, indicating that it has the potential to be an effective approach for complex, unconstrained optimization.

1 Introduction

Nonlinear conjugate gradient (CG) methods are ideal for handling large-scale problems because they have simple iterations and minimal memory demands.

Received: June 2, 2023; Accepted: July 8, 2023; Published: July 17, 2023
2020 Mathematics Subject Classification: 49M37, 90C30, 90C99.
Keywords and phrases: conjugacy condition, sufficient descent, unconstrained optimization, numerical experiment.

These methods are specifically developed to tackle optimization problems that are structured in the following manner:

$$\min f(x), x \in \mathbb{R}^n, \tag{1.1}$$

where the objective function f is continuously differentiable and $\mathbb{R}^n$ is an n-dimensional space.

Optimization problems given by (1.1) are prevalent in both applied fields like economics, social sciences, engineering, and theoretical fields since most optimization problems can be converted to unconstrained optimization problems [1,2]. The CG method is identified by a search direction that is defined as follows:

$$d_n = \begin{cases} -g_n & \text{if } n = 0, \\ -g_n + \beta_n d_{n-1} & \text{if } n \geq 1, \end{cases} \tag{1.2}$$

where g_n represents the gradient, denoted by ∇f_n and β_n is called the CG update parameter. The iterative scheme for solving (1.1) is generated recurrently by:

$$x_{n+1} = x_n + \omega_n d_n, \qquad n = 0, 1, 2, \cdots \tag{1.3}$$

where ω_n represents the step size, which is commonly determined by a search technique.

The choice of the update parameter β_n in CG methods is a crucial determinant of the algorithm's performance, and different formulae for β_n have been proposed over the years. These classical CG formulae are typically used to distinguish different variants of CG methods. The Hestenes and Stiefel (HS) [3], Fletcher and Reeves (FR) [4], Polak, Ribiere, and Polyak (PRP) [5,6], Conjugate Descent (CD) [7], Liu and Storey (LS) [8], and Hager and Zhang (HZ) [9] formulae are among the most well-known and defined as:

$$\beta_n^{HS} = \frac{g_n^T y_{n-1}}{d_{n-1}^T y_{n-1}}, \tag{1.4}$$

$$\beta_n^{FR} = \frac{\|g_n\|^2}{\|g_{n-1}\|^2}, \tag{1.5}$$

$$\beta_n^{PRP} = \frac{g_n^T y_{n-1}}{\|g_{n-1}\|^2}, \tag{1.6}$$

$$\beta_n^{CD} = \frac{\|g_n\|^2}{-d_{n-1}^T g_{n-1}}, \tag{1.7}$$

$$\beta_n^{LS} = \frac{g_n^T y_{n-1}}{-d_{n-1}^T g_{n-1}}, \tag{1.8}$$

and

$$\beta_n^{HZ} = \left(y_{n-1} - \frac{2d_{n-1}\|y_{n-1}\|^2}{d_{n-1}^T y_{n-1}} \right)^T \frac{g_n}{d_{n-1}^T y_{n-1}}. \tag{1.9}$$

The approach for the computation of the line search is important in determining the convergence speed of a CG method. The line search is usually computed either by an exact or inexact method. The exact method is costly, cumbersome, and prone to error; thus, the inexact approaches are preferable to researchers, as there is a need for a line search that can identify a step length that produces adequate reductions in the objective function's value at the lowest possible cost. Therefore, the fundamental goal of the inexact line search is to develop a yardstick that ensures ω_n is not too long or too short, to ensure that a suitable step size is chosen to kickstart the algorithm, and to design a sequence of updates such that the criterion generated is satisfied after every few steps.

One of the most popular inexact line searches is the strong Wolfe Powell (SWP) line search proposed by Wolfe [10] and given by:

$$f(x_n) - f(x_n + \omega_n d_n) \geq -\eta \omega_n g_n^T d_n, \tag{1.10}$$

and

$$|g(x_n + \omega_n d_n)^T d_n| \leq \sigma |g_n^T d_n|, \tag{1.11}$$

where $0 \leq \eta \leq \sigma < 1$.

Other inexact search methods include the Goldstein search rule proposed by Goldstein [11] as follows:

$$\delta_1 \omega_n g_n^T d_n \leq f(x_n + \omega_n d_n) - f_n \leq \delta_2 \omega_n g_n^T d_n, \tag{1.12}$$

where $0 < \delta_2 < \frac{1}{2} < \delta_1 < 1$, and Armijo line search rule proposed by Armijo [12], given by:

$$f(x_n + \omega_n d_n) \leq f(x_n) + \omega\epsilon\nabla f(x_n)^T d_n, \tag{1.13}$$

where $0 < \epsilon < 1$ and $\omega > 1$.

The need to improve the performance of classical CG methods brought about the introduction of hybrid CG methods by researchers. In practice, hybrid methods have been proven to be excellent because they exploit the advantages of the traditional methods involved in hybridization. Thus, several hybrid CG methods have been proposed by authors by combining two or more conventional methods, and the convergence characteristics of these new methods have been established using a wide range of inexact search criteria. For instance, the author in [13] presented two hybrid CG methods using the generalized Wolfe search method, where the resulting update parameters are given by:

$$\beta_n^{(1)} = \begin{cases} a_1\beta_n^{DY} + a_2\beta_n^{HS} & \text{if } \|g_n\|^2 > \left|g_n^T g_{n-1}\right|, \\ 0 & \text{else}, \end{cases}$$

and

$$\beta_n^{(2)} = \begin{cases} a_1\beta_n^{FR} + a_2\beta_n^{PRP} & \text{if } \|g_n\|^2 > \left|g_n^T g_{n-1}\right|, \\ 0 & \text{else}, \end{cases}$$

where a_1, a_2 are nonnegative parameters.

Djordjevic [14] proposed a new hybrid CG method by combining the LS and CD update parameters, resulting into the form:

$$\beta_n^{hyb} = (1 - \varpi_n)\beta_n^{LS} + \varpi_n\beta_n^{CD},$$

where the new parameter was made to satisfy the conjugacy condition. Dong *et al.* [15] presented a three-term search direction method by affine combination of variants of HS method. The new search direction is of the form:

$$d_n^{NHS} = (1 - \Theta_n)d_n^{HS3} + \Theta_n d_n^{HS2},$$

where

$$d_n^{HS2} = -g_n + \beta_n^{HS} d_{n-1} - \beta_n^{HS} \frac{g_n^T d_{n-1}}{g_n^T g_n} g_n,$$

and

$$d_n^{HS3} = -g_n + \beta_n^{HS} d_{n-1} - \frac{g_n^T d_{n-1}}{d_{n-1}^T y_{n-1}} y_{n-1}.$$

Osinuga and Olofin [16] proposed a hybrid search direction in the form:

$$d_n = \begin{cases} -H_n g_n & n = 0, \\ -H_n g_n + \eta(-g_n + \beta_n^{PRP} d_{n-1}) - \vartheta_n y_n, & n \geq 1, \end{cases}$$

where $\eta > 0, \vartheta_n = \frac{g_n^T d_{n-1}}{\|g_n - 1\|^2}$.

The global convergence of their method was proved under the Armijo search method. In [17], Djordjevic developed another hybrid CG method based on LS and FR update parameters, and it is given by:

$$\beta_n^{hyb} = (1 - \varpi_n)\beta_n^{LS} + \varpi_n \beta_n^{FR},$$

where ϖ_n is a scalar parameter. Salihu *et al.* [18] proposed a hybrid CG method of Dai-Liao type based on a convex combination of HS and FR methods, resulting in the following β_n formula:

$$\beta_n^{DHF} = (1 - \lambda_n)\,\beta_n^{HS} + \lambda_n \beta_n^{FR},$$

where

$$\lambda_n = \frac{-s_n^T g_{n+1}\,|g_n\|^2}{(y_n^T s_n)\,|g_{n+1}\|^2 - \left(g_{n+1}^T y_n\right)|g_n\|^2}.$$

The authors in [19] designed a new β_n formula for the denominators of PR, HS, and LS methods while retaining the original numerator. The resulting new β_n method, namely Rivaie-Mustapha-Ismail-Leong (RMIL) method is given by:

$$\beta_n^{RMIL} = \frac{g_{n+1}^T y_n}{\|d_n\|^2}. \tag{1.14}$$

Similarly in [20], the authors modified the denominators of the FR, CD, and DY methods while retaining the original numerator to produce a new β_n coefficient, namely the Mandara-Mamat-Waziri-Usman (MMWU) method, given by:

$$\beta_n^{MMWU} = \frac{\|g_{n+1}\|^2}{\|d_n\|^2}. \tag{1.15}$$

Recently, new hybrid CG methods have been proposed by authors in [21–25].

Inspired by earlier works [19, 20, 26, 27], this paper presents a new hybrid algorithm by combining the Dai-Laio and Dai-Yuan update parameters. The remainder of this article is organized as follows: The new β_n algorithm is described in Section 2, its sufficient descent property is established in Section 3, numerical results and a discussion are provided in Section 4, and the conclusion is provided in Section 5.

2 The New β_n Formula

In [26], Dai and Laio proposed a new conjugacy condition, resulting in the following formula for β_n:

$$\beta_n^{DL} = \frac{g_n^T\left(y_{n-1} - ts_{n-1}\right)}{d_{n-1}^T y_{n-1}}, \tag{2.1}$$

where $t \geq 0$. It is evident from (2.1) that:

$$\beta_n^{DL} = \beta_n^{HS} - t\frac{g_n^T s_{n-1})}{d_{n-1}^T y_{n-1}}, \tag{2.2}$$

where β_n^{HS} is given by (1.4). By replacing β_n^{HS} in the above with β_n^{DY}, proposed by Dai and Yuan in [27], a new hybrid CG method of Dai-Liao type is proposed as follows:

$$\beta_n^{HDYDL} = \beta_n^{DY} - t\frac{g_n^T s_{n-1})}{d_{n-1}^T y_{n-1}}, \tag{2.3}$$

where

$$\beta_n^{DY} = \frac{\|g_n\|^2}{d_{n-1}^T y_{n-1}}.$$

In this case, $t > 0$, $||\ .\ ||$ denotes the Euclidean norm, $y_{n-1} = g_n - g_{n-1}$, and $s_{n-1} = x_n - x_{n-1}$. By simplifying (2.3), the following is obtained:

$$\beta_n^{HDYDL} = \frac{g_n^T (g_n - ts_{n-1})}{d_{n-1}^T y_{n-1}}. \tag{2.4}$$

The algorithm below is used to implement the new β_n formula.

HDYDL Algorithm

Step 1: Given that $x_0 \in \mathbb{R}^n$, set $n = 0$ and $\epsilon \geq 0$.
Step 2: Stop if $\|g_n\| \leq \epsilon$.
Step 3: determine d_n by (1.2).
Step 4: Determine the stepsize ω_n by (1.13).
Step 5: Determine x_n by (1.3), $g_n = g(x_n)$.
Step 6: Calculate β_n by (2.4).
Step 7: Set $n := n + 1$, and return to step 2.

3 Sufficient Descent Property of the New CG Formula

For a CG method to globally converge, a sufficient descent property must be satisfied. The following definition and lemma will be useful for establishing the sufficient descent property of the HDYDL method.

Definition 3.1. A CG method fulfills the sufficient descent condition if $\zeta > 0$ is a constant such that:

$$g_n^T d_n \leq -\zeta \|g_n\|^2, \quad 0 < \zeta \leq 1. \tag{3.1}$$

Lemma 3.1. *The HDYDL method meets the sufficient descent requirement (3.1) with* $\zeta = \frac{1}{1-\sigma}$.

Proof. By pre-multiplying (1.2) by g_n^T and using (2.4),

$$
\begin{aligned}
g_n^T d_n &= -g_n^T g_n + \beta_n^{HDYDL} \left(g_n^T d_{n-1} \right), \\
&= -g_n^T g_n + \frac{g_n^T \left(g_n - t s_{n-1} \right)}{d_{n-1}^T y_{n-1}} \left(g_n^T d_{n-1} \right).
\end{aligned}
$$

By SWP line search (1.11) we have that:

$$
d_{n-1}^T g_n \leq \sigma d_{n-1}^T g_{n-1}, \quad \sigma \in (0, 1].
$$

With $y_{n-1} = g_n - g_{n-1}$, the denominator gives:

$$
\begin{aligned}
d_{n-1}^T y_{n-1} &= d_{n-1}^T g_n - d_{n-1}^T g_{n-1}, \\
&\leq \sigma d_{n-1}^T g_{n-1} - d_{n-1}^T g_{n-1}. \\
&\leq d_{n-1}^T g_{n-1} \left(\sigma - 1 \right).
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
g_n^T d_n &\leq -\left\| g_n \right\|^2 + \frac{g_n^T \left(g_n - t s_{n-1} \right)}{d_{n-1}^T g_{n-1} \left(\sigma - 1 \right)} \left(\sigma d_{n-1}^T g_{n-1} \right), \\
&\leq -\left\| g_n \right\|^2 + \frac{\sigma d_{n-1}^T g_{n-1} \left(g_n^T g_n - t g_n^T s_{n-1} \right)}{d_{n-1}^T g_{n-1} \left(\sigma - 1 \right)}, \\
&\leq -\left\| g_n \right\|^2 + \frac{\sigma}{\sigma - 1} \left\| g_n \right\|^2, \\
&= \left\| g_n \right\|^2 \left(-1 + \frac{\sigma}{\sigma - 1} \right), \\
&= \left\| g_n \right\|^2 \left(\frac{1}{\sigma - 1} \right), \\
&= -\left(\frac{1}{1 - \sigma} \right) \left\| g_n \right\|^2.
\end{aligned}
$$

Therefore,

$$
g_n^T d_n \leq -\left(\frac{1}{1 - \sigma} \right) \left\| g_n \right\|^2.
$$

Hence the proof □

4 Numerical Results

This section provides a report that compares the performance of the proposed new method with that of three existing methods.

The methods were implemented in Matlab and carried out on a computer running Windows 10 Pro with 4 GB of RAM and a processor speed of 2.16 GHz. The iterations were stopped when either the norm of the gradient vector was less than or equal to 10^{-6} or when the number of iterations exceeded 2000. The study tested 25 unconstrained problems from the CUTEr library [28] and Andrei [29], each of dimensions 5000 and 10,000. Table 1 lists the problems solved and their sources, while Table 2 provides details of the numerical results for the problems listed in Table 1. The computational details include the number of iterations (NOI) and the computational time (CPUT). For failed iterations, the notation "I_f" is used. The Armijo search technique was used for the computations.

The study compared the efficiency of four methods, namely HDYDL, RMIL [19], DL [26], and MMWU [20], using the profile of [30]. The comparison was based on the number of iterations and computational time. The results are displayed in Figures 1 and 2, which show the performance of each method in terms of computational time (in seconds), and number of iterations. The vertical axis to the left of the curves represents the proportion of test problems solved successfully by each method, with the top curve denoting the fastest method. The HDYDL method was found to be the most effective, with a success rate of 90%, followed by MMWU with 76%, DL with 68%, and RMIL with 64%. It is worth noting that the DL method used t = 0.1, while t = 0.01 was used in the HDYDL method.

The table of results shows that while the DY, RMIL, and MMWU methods solved some problems faster than the others, the HDYDL method solved most of the problems with fewer iterations and less computational time than the three existing methods. Therefore, the HDYDL method is more efficient and robust than the existing methods.

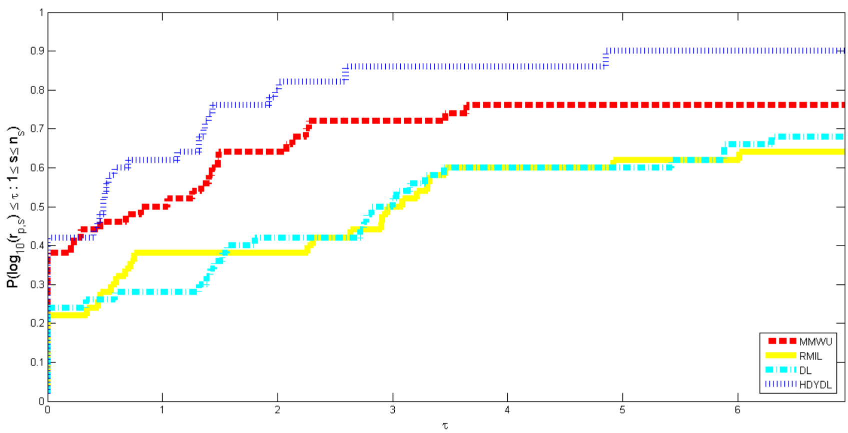

Figure 1: Iteration profile comparing the HDYDL method with RMIL, DL, and MMWU methods.

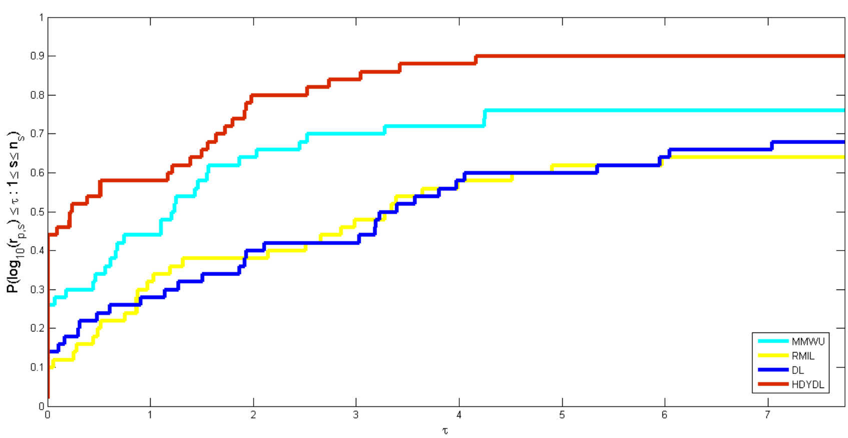

Figure 2: CPU profile comparing the HDYDL method with the RMIL, DL, and MMWU methods.

Table 1: List of Problems solved and their references.

S/N	Problem Names	Sources
1	Qf1	[29]
2	Extended Block Diadonal	[29]
3	Qf2	[29]
4	Extended Powell	[29]
5	Diagonal 5	[29]
6	Diagonal 4	[29]
7	Staircase1	[29]
8	Staircase2	[29]
9	Extended Beale	[29]
10	RMODF COSINE	[28]
11	MDF EXPLIN 1	[28]
12	MODF SINE	[28]
13	RMODF SINE	[28]
14	RMDF GENHUMPS	[28]
15	Extended Three Exponential Terms	[29]
16	Partial Perturbed Quadratic	[29]
17	QUARTC	[28]
18	Extended DENSCHNB	[28]
19	Generalized Quartic	[29]
20	Diagonal 7	[29]
21	Diagonal 8	[29]
22	Full Hessian FH3	[29]
23	SINCOS	[29]
24	HIMMELBG	[28]
25	Extended Tridiagonal-1	[29]

Table 2: The Numerical Results.

	MMWU Method	RMIL Method	DL Method	HDYDL Method
S/N	NOI/CPUT	NOI/CPUT	NOI/CPUT	NOI/CPUT
1	I_f/I_f	I_f/I_f	146/4.175	207/9.672
	I_f/I_f	I_f/I_f	I_f/I_f	270/15.038
2	62/1.441	30/1.044	82/1.954	114/3.237
	67/2.179	38/2.184	48/1.299	83/3.397
3	15/0.527	I_f/I_f	164/4.778	21/0.611
	15/0.415	I_f/I_f	I_f/I_f	21/0.593
4	342/10.441	I_f/I_f	291/6.837	I_f/I_f
	I_f/I_f	I_f/I_f	I_f/I_f	I_f/I_f
5	22/0.560	22/0.722	118/3.569	2/0.058
	25/0.682	17/0.570	118/4.732	2/0.036
6	28/0.638	38/1.161	I_f/I_f	74/2.429
	29/1.023	40/1.392	I_f/I_f	75/3.846
7	1/0.032	1/0.028	1/0.086	1/0.020
	I_f/I_f	I_f/I_f	I_f/I_f	I_f/I_f
8	29/1.094	45/1.451	258/8.116	6/0.201
	I_f/I_f	I_f/I_f	I_f/I_f	I_f/I_f
9	67/3.213	107/6.296	I_f/I_f	109/3.427
	77/6.051	113/11.781	I_f/I_f	110/5.769
10	1/0.058	1/0.028	1/0.029	1/0.027
	1/0.026	1/0.028	1/0.035	1/0.023
11	48/1.500	94/2.927	156/6.099	19/0.514
	48/1.527	96/3.127	160/6.465	20/0.710
12	42/1.145	I_f/I_f	40/1.179	15/0.416
	45/1.504	I_f/I_f	40/1.124	16/0.510
13	8/0.218	79/2.211	28/0.792	20/0.641
	8/0.190	80/1.196	23/0.716	20/0.537
14	70/7.387	124/4.048	113/4.072	16/0.388
	72/2.714	128/14.119	116/6.594	17/0.473
15	983/26.274	I_f/I_f	I_f/I_f	853/37.560
	746/17.299	I_f/I_f	I_f/I_f	I_f/I_f
16	19/0.606	37/1.170	315/9.736	4/0.148
	953/34.328	1572/63.064	1428/38.546	1308/77.193
17	1/0.038	1/0.040	1/0.039	1/0.028
	1/0.031	1/0.025	1/0.026	1/0.021
18	53/1.777	I_f/I_f	20/0.920	27/0.748
	54/1.822	I_f/I_f	20/0.775	29/0.912
19	I_f/I_f	25/0.797	I_f/I_f	33/0.927
	I_f/I_f	22/0.801	65/3.053	33/1.048
20	23/0.582	1486/36.459	152/5.441	91/3.335
	23/0.578	I_f/I_f	155/4.711	93/3.848
21	22/1.524	485/15.128	156/10.374	16/0.663
	22/1.483	112/5.324	160/9.110	18/0.549
22	30/1.686	38/1.947	I_f/I_f	30/1.068
	I_f/I_f	I_f/I_f	I_f/I_f	26/1.554
23	62/2.801	94/6.398	162/6.753	374/9.728
	93/10.085	98/6.913	58/2.777	350/22.885
24	I_f/I_f	I_f/I_f	168/5.776	431/22.771
	I_f/I_f	I_f/I_f	172/4.669	466/15.440
25	I_f/I_f	3/0.172	I_f/I_f	87/3.068
	I_f/I_f	3/0.306	I_f/I_f	87/3.273

5 Conclusion

In this paper, a new hybrid CG method has been proposed by combining the DY and DL update parameters based on Dai-Liao conjugacy condition. The proposed method is easy to implement and does not require any additional parameters. The new method has been theoretically shown to possess sufficient descent property under the SWP line search. Numerical experiments conducted on a set of standard optimization problems demonstrate that the new method outperforms existing methods in terms of convergence and efficiency. Therefore, the proposed hybrid CG method is a promising approach for solving unconstrained optimization problems and can be extended to other optimization problems. Future research will focus on global convergence.

References

[1] D.C. Luenberger, *Linear and Nonlinear Programming*, 2nd ed., Addition-Wesley, Reading MA, 1989.

[2] O.L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New-York, 1969.

[3] M.R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, *J. Res. Natl. Bur. Stand.* 49 (1952), 409-436. https://doi.org/10.6028/jres.049.044

[4] R. Fletcher and C.M. Reeves, Function minimization by conjugate gradients, *Comput. J.* 7 (1964), 149-154. https://doi.org/10.1093/COMJNL/7.2.149

[5] E. Polak and G. Ribiere, Note Sur la Convergence de directions conjugees, *ESAIM: Math. Model. Numer. Anal.* 3 (1969), 35-43. https://doi.org/10.1051/M2AN/196903R100351

[6] B.T. Polyak, The conjugate gradient method in extreme problems, *Comput. Math. Math. Phys.* 9 (1969), 94-112. https://doi.org/10.1016/0041-5553(69)90035-4

[7] R. Fletcher, *Practical Method of Optimization*, 2nd ed., John Wiley, New York, 1987. https://doi.org/10.1002/9781118723203

[8] Y. Liu, C. Storey, Efficient generalized conjugate gradient algorithms part 1, theory, *J. Optim. Theory Appl.* 69 (1991), 322-340. https://doi.org/10.1007/BF00940464

[9] W.W. Hager, H. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, *SIAM J. Optim.* 16 (1) (2005), 170-192. https://doi.org/10.1137/030601880

[10] P. Wolfe, Convergence conditions for ascent methods, *SIAM J. Optim.* (1969), 226-235. https://doi.org/10.1137/1011036

[11] A.A. Goldstein, On steepest descent, *J. Soc. Ind. Appl. Math. Ser. A Control* 3 (1) (1965), 147-151.

[12] L. Armijo, Minimization of function having Lipschitz continuous first partial derivative, *Pacific J. Math.* 16 (1966), 1-3. https://doi.org/10.2140/pjm.1966.16.1

[13] X. Xu, F. Kong, New hybrid conjugate gradient method with the generalized wolfe line search, *SpringerPlus* 5 (2016), 1-10. https://doi.org/10.1186/s40064-016-2522-9

[14] S.S. Djordjevic, New hybrid conjugate gradient method as a convex combination of LS and CD methods, *Filomat* 31 (6) (2017), 1813-1825. https://doi.org/10.2298/fil1706813d

[15] X. Dong, D. Han, R. Ghanbari, X. Li, Z. Dai, Some new three-term Hestenes-Stiefel conjugate gradient methods with affine combination, *Optim.* 66 (5) (2017), 1-18. https://doi.org/10.1080/02331934.2017.1295242

[16] I.A. Osinuga, I.O. Olofin, Extended hybrid conjugate gradient method for unconstrained optimization, *J. Comput. Sci. Appl.* 25 (1) (2018), 25-33.

[17] S.S. Djordjevic, New hybrid conjugate gradient method as a convex combination of LS and FR methods, *Acta Math. Sci.* 39 (1) (2019), 214-228. https://doi.org/10.1007/s10473-019-0117-6

[18] N. Salihu, M.R. Odekunle, A.M. Saleh, S. Salihu, A Dai-Liao hybrid Hestenes-Stiefel and Fletcher-Reeves methods for unconstrained optimization, *Int. J. Ind. Optim.* 2 (1) (2021), 33-50. https://doi.org/10.12928/IJIO.V2I1.3054

[19] M. Rivaie, M. Mamat, W.J. Leong, M. Ismail, A new class of nonlinear conjugate gradient coefficients with global convergence properties, *Appl. Math. Comput.* 218 (2012), 11323-11332.

[20] A.V. Mandara, M. Mamat, M.Y. Waziri, M.A. Mohammed, U.A. Yakubu, A new conjugate gradient coefficient with exact line search for unconstrained optimization, *Far East J. Math. Sci.* 105 (2) (2018), 193-206. https://doi.org/10.17654/ms105020193

[21] A.B. Abubakar, M. Malik, P. Kumam, H. Mohammad, M. Sun, A.H. Ibrahim, A.I. Kiri, A Liu-Storey-type conjugate gradient method for unconstrained minimization problem with application in motion control, *J. King Saud Univ.-Sci.* 34 (2022), 1-11. https://doi.org/10.1016/j.jksus.2022.101923

[22] M. Fang, M. Wang, M. Sun, R. Chen, A modified hybrid conjugate gradient method for unconstrained optimization, *Hindawi J. Math.*, Article ID 5597863 (2021), 1-9. https://doi.org/10.1155/2021/5597863

[23] J. Sabí́u, K. Muangchoo, A. Shah, A.B. Abubakah, K.O. Aremu, An inexact optimal hybrid conjugate gradient method for solving symmetric nonlinear equations, *Symmetry* 13 (2021), 1829. https://doi.org/10.3390/sym13101829

[24] O.B. Akinduko, A new conjugate gradient method with sufficient descent property, *Earthline J. Math. Sci.* 6 (2021), 163-174. https://doi.org/10.34198/EJMS.6121.163174

[25] T. Diphofu, P. Kaelo, A.R. Tufa, A convergent hybrid three-term conjugate gradient method with sufficient descent property for unconstrained optimization, *Topol. Algebra Appl.* 10 (2022), 47-60. https://doi.org/10.1515/taa-2022-0112

[26] Y.H. Dai, L.Z. Liao, New conjugacy conditions and related nonlinear conjugate gradient method, *Appl. Math. Optim.* 43 (2001), 87-101. https://doi.org/10.1007/s002450010019

[27] Y.H. Dai, Y. Yuan, A nonlinear conjugate gradient method with a strong global convergence property, *SIAM J. Optim.* 10 (1999), 177-182. https://doi.org/10.1137/s1052623497318992

[28] I. Bongartz, A.R. Conn, N.I.M. Gould, P.L. Toint, CUTE: Constrained and Unconstrained Testing Environments, *ACM Trans. Math. Softw.* 21 (1995), 123-160. https://doi.org/10.1145/200979.201043

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

[30] E.D. Dolan and J.J. More, Benchmarking optimization software with performance profiles, *Math. Program.* 91 (2002), 201-213.