Int. J. Anal. Appl. (2025), 23:10

International Journal of Analysis and Applications

Another Updated Parameter for the Hestenes-Stiefel Conjugate Gradient Method

Osman Omer Osman Yousif[1], **Raouf Ziadi**[2], **Mohammed A. Saleh**[3,*], **Abdulgader Z. Almaymuni**[3]

[1]*Department of Mathematics, Faculty of Mathematical and Computer Sciences, University of Gezira, Wad Medani, Sudan*

[2]*Laboratory of Fundamental and Numerical Mathematics (LMFN), Department of mathematics, University Setif - 1 - Ferhat Abbas, Algeria*

[3]*Department of Cybersecurity, College of Computer, Qassim University, Saudi Arabia*

[*]*Corresponding author:* m.saleh@qu.edu.sa.

Abstract. The conjugate gradient (CG) methods are considered as one of the most popular methods for solving linear and non-linear unconstrained optimization problems, especially the problems of large-scale, that is because they are characterized by low memory requirements and strong local and global convergence properties. The method of Hestenes-Stiefel (HS) usually gives good numerical results in the practical computation. However, theoretically, its convergence properties are uncertain. To address the convergence failure of HS method, many choices for its update parameter have been proposed such as the choice of Gilbert and Nocedal in 1992, of Hager and Zhang in 2005, and of Yousif et al. in 2022. In this paper, motivated by these updated parameters, we propose another updated parameter for HS, and hence another CG method which inherits all the convergence properties of Gilbert and Nocedal, Hager and Zhang, and of Yousif et al. and has better numerical results. To show the efficiency and robustness of the new modified method in practice, a numerical experiment was done.

1. Introduction

Conjugate Gradient (CG) methods are iterative techniques used for solving unconstrained optimization problems in various fields, including science, engineering, and economics. The history of CG methods can be traced back to the contributions of Cornelius Lanczos and Magnus Hestenes at the Institute for Numerical Analysis, along with the independent work of Eduard Stiefel at the Swiss Federal Institute of Technology (ETH) Zürich.

CG methods are designed to solve optimization problems of the form:

Received: Oct. 25, 2024.

2020 *Mathematics Subject Classification.* 90C26, 64K05, 90C30.

Key words and phrases. optimization method; conjugate gradient method; sufficient descent property; strong Wolfe line search; global convergence.

https://doi.org/10.28924/2291-8639-23-2025-10
ISSN: 2291-8639

$$\min_{x\in\mathbb{R}^n} f(x) \tag{P}$$

where $f:\mathbb{R}^n\to\mathbb{R}$ is a continuously differentiable function, bounded from below.

These methods use the iterative formula:

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \ldots \tag{1.1}$$

where α_k is a positive real number called the *step length*, and d_k represents the search direction at the k-th iteration.

The search direction d_k is computed using the following rule:

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{if } k \geq 1, \end{cases} \tag{1.2}$$

where β_k is the update parameter, and g_k is the gradient vector of $f(x)$ at the point x_k.

Different formulas for β_k determine different CG methods. Some well-known choices for β_k are listed in the following table:

Table 1. The classical CG update parameters

Formula	**Name**	**Reference**
$\beta_k^{HS} = \frac{g_k^T y_k}{d_{k-1}^T y_k}$	Hestenes-Stiefel (HS)	Hestenes-Stiefel (1952) [10]
$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}$	Fletcher-Reeves (FR)	Fletcher-Reeves (1964) [21]
$\beta_k^{PRP} = \frac{g_k^T y_k}{\|g_{k-1}\|^2}$	Polak-Ribiere-Polyak (PRP)	Polak-Ribière-Polyak (1969) [1,3]
$\beta_k^{CD} = \frac{\|g_k\|^2}{-d_{k-1}^T g_{k-1}}$	Conjugate Descent (CD)	Fletcher (1987) [20]
$\beta_k^{LS} = \frac{g_k^T y_k}{-d_{k-1}^T g_{k-1}}$	Liu-Storey (LS)	Liu-Storey (1991) [23]
$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_k}$	Dai-Yuan (DY)	Dai-Yuan (1999) [24]

In Table 1, $\|.\|$ denotes the vector norm and $y_k = g_k - g_{k-1}$. For more formulas for the parameter β_k, see [4,7,12–16,25–28].

The step length α_k is chosen to obtain the exact minimum or approximate minimum of the sub-problem:

$$\min_{\alpha\geq 0} f(x_k + \alpha d_k).$$

Since $\alpha \geq 0$, the direction should satisfy the following descent condition:

$$g_k^T d_k < 0, \quad \forall k \geq 0,$$

which is called the sufficient descent condition if there exists a constant $c > 0$ such that:

$$g_k^T d_k \leq -C\|g_k\|^2, \quad \forall k \geq 0 \quad \text{and a constant} \quad C > 0. \tag{1.3}$$

The step length α_k is computed using exact or inexact methods called line searches. In exact line search, α_k is obtained in the direction d_k by the rule:

$$f(x_k + \alpha_k d_k) = \min_{\alpha \geq 0} f(x_k + \alpha d_k). \tag{1.4}$$

Since it is difficult in practice to compute α_k using formula (1.4), inexact line searches are introduced to compute approximate values for α_k. The Wolfe and strong Wolfe line searches are examples of inexact line searches and are often used in practice. In Wolfe line search [18, 19], α_k satisfies the following two conditions:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k, \tag{1.5}$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k,$$

whereas, in the strong Wolfe line search, α_k is chosen to satisfy condition (1.5) and:

$$|g(x_k + \alpha_k d_k)^T d_k| \leq -\sigma g_k^T d_k, \tag{1.6}$$

where $0 < \delta < \sigma < 1$.

The efficiency of CG methods is measured by two things: their global convergence, that is:

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

and their numerical results in practice. Both measurements depend essentially on the well-chosen step length line search and the search direction d_k.

Under the exact line search, if the objective function f is a strongly convex quadratic, then all the methods with their parameters in Table 1 are equivalent. However, their performance differs for non-quadratic functions. Due to the jamming phenomenon, where small steps are taken without making significant progress, the performance of methods with $\|g_k\|^2$ in the numerator which are FR, CD, and DY in Table are less than the performance of methods with $g_k^T y_k$ in the numerator. This is because the PRP, HS, and LS methods, which share the common numerator, possess a built-in restart feature that addresses the jamming problem: when $x_k - x_{k-1} \to 0$, $y_k = g_k - g_{k-1}$ in the numerator tends to zero, prompting a restart with the steepest descent direction.

Considerable efforts have been made to prove the sufficient descent property and the global convergence of conjugate gradient methods or to modify them for better performance. Zoutendijk [5] proved the global convergence of the FR method under the exact line search, which was extended later by Al-Baali [8] under the strong Wolfe line search. As mentioned earlier, the PRP and HS methods show good numerical results compared to FR, CD, and DY, but their convergence properties are poor. Polak and Ribière [3] established the global convergence of the PRP method with the exact line search, and thus of HS, since $\beta_k^{PRP} = \beta_k^{HS}$ with the exact line search. However, Powell [9] later reported that for some non-convex problems, PRP does not converge globally. Moreover, their global convergence under the strong Wolfe line search has not been established.

Several efforts have been made to prove the global convergence of the original HS method via inexact line searches or to propose modified versions with better convergence properties.

Powell [9] suggested restricting the parameter β_k^{PRP} to be non-negative. Based on this suggestion, Gilbert and Nocedal [6] proposed the PRP+ method. Similarly, the HS+ method can be defined by the conjugate gradient (CG) update parameter:

$$\beta_k^{HS+} = \max(0, \beta_k^{HS}), \tag{1.7}$$

which is globally convergent under the Wolfe line search provided the sufficient descent condition is satisfied.

In addition to the built-in restart feature of the HS and HS+ methods, the conjugacy condition:

$$d_k^T y_k = 0,$$

is always satisfied, independently of the line search.

To address the poor convergence properties of the HS method, an interesting modified method, called CG-DESCENT, was proposed by Hager and Zhang [22] in 2005. The update parameter of CG-DESCENT is given by:

$$\beta_k^N = \frac{(y_k - 2d_{k-1}\frac{\|y_k\|^2}{d_{k-1}^T y_k})^T g_k}{d_{k-1}^T y_k}.$$

In addition to the good numerical results, the CG-DESCENT method possesses the sufficient descent property, independently of the line search.

Most recently, Yousif et al. [17] proposed a modified version of HS by restricting the parameter β_k^{HS} to lie within an interval depending on a parameter μ, where μ is a real number greater than 2. The modified version is called OHS, and its parameter is given by:

$$\beta_k^{OHS} = \begin{cases} \beta_k^{HS}, & \text{if } -\mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2} \le \beta_k^{HS} \le \mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \\ 0, & \text{otherwise.} \end{cases} \tag{1.8}$$

They proved the global convergence and the sufficient descent property of the OHS method under the strong Wolfe line search.

In this paper, motivated by the good performance of the HS method in practice and the convergence properties of the HS+ and OHS methods, we propose a new modified parameter for HS and, hence, a new modified algorithm in Section 2. In Section 3, the proof of the global convergence was established. To demonstrate the efficiency and robustness of the new modified algorithm, a numerical experiment is conducted in Section 4.

2. A Modified Formula and Algorithm

As mentioned in the Introduction section, to establish the global convergence of the HS method, the modified HS+ parameter is proposed by setting the negative values of the HS parameter to zero, as shown in formula (1.7). Additionally, the modified OHS parameter is proposed by restricting the values of the HS parameter to a certain interval that includes both positive and negative values of the HS parameter, as given in formula (1.8).

Motivated by the above, we propose another new modified parameter of HS by combining the negative values of OHS with the values of the HS+ parameter to obtain the following:

$$\beta_k^{OOHS} = \begin{cases} \beta_k^{HS}, & \text{if } \beta_k^{HS} \geq -\mu \frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \\ 0, & \text{otherwise.} \end{cases} \tag{2.1}$$

where $\mu > 2$ is a real number. With this new parameter, we derive a new modified method, called OOHS, which is defined in Algorithm 1 below.

Algorithm 1: The OOHS algorithm

Input: Choose the parameters δ and σ such that $0 < \delta < \sigma < 1$. Choose a scalar $\epsilon > 0$ sufficiently small to stop the algorithm.

// **Initialization**

- Set $k = 0$ and select a starting point $x_0 \in \mathbb{R}^n$.
- Set $g_0 = \nabla f(x_0)$ and $d_0 = -g_0$.

// **Main loop**

while $\|g_k\| \geq \epsilon$ **do**

- Compute the steplegth α_k that meets conditions (1.5) and (1.6).
- Evaluate d_k using (1.2).
- Set $x_{k+1} = x_k + \alpha_k d_k$ and evaluate g_{k+1} and β_k^{OOHS}.
- Compute the search direction $d_{k+1} = -g_{k+1} + \beta_k^{OOHS} d_k$.
- Set $k = k + 1$.

Note that, from (2.1), when $\mu \to \infty$, then $\beta_k^{OOHS} \to \beta_k^{HS}$. Therefore, for a sufficiently large value of μ, the OOHS method provides good approximations to the HS method. Similar to the HS method, we expect it to yield good numerical results in practical computations. The question now is: what about its convergence?

3. Convergence Analysis

One of the most important issues when studying a CG method is its convergence properties, so we devote this section to the analysis of the convergence of OOHS algorithm. Firstly, we note that, since the OOHS parameter combines both the parameters of HS+ and OHS, it is clear that the OOHS method inherits the two attractive features of both methods, that are, self-restart feature and the conjugacy condition, independent of line search. Now, throughout this section, we assume that

(i)- $g_k \neq 0$, for all $k \geq 0$,

(ii)- The level set, $\Omega = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ is bounded, where x_0 is the starting point.

(iii)- In some neighbourhood N of Ω, the objective function is continuously differentiable, and its gradient is Lipschitz continuous, namely, there exists a constant $l > 0$ such that

$$\|g(x) - g(y)\| \leq l\|x - y\|, \quad \forall x, y \in N.$$

Note that assumptions (ii) and (iii) imply that there exists a positive constant $\bar{\gamma}$ such that

$$\|g(x)\| \leq \bar{\gamma}, \quad \text{for all } x \in N.$$

Assumptions (ii) and (iii) above on the objective function and the Zoutendijk condition [5] are usually used to analyze the global convergence of conjugate gradient methods.

Lemma 3.1. *Suppose that assumptions (ii) and (iii) hold. Consider any CG method of the form* (1.1)-(1.3), *where the step length α_k is computed by the Wolfe line search. Then, the following condition, known as the Zoutendijk condition, is satisfied:*

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

For good convergence results, we need to require that β_k be small when the step length is small. This property is called **Property*** and is defined by Gilbert and Nocedal [6].

Property*: Consider a method of the form (1.1) and (1.2), and suppose that for all $k \geq 0$,

$$0 < \gamma \leq \|g_k\| \leq \bar{\gamma},$$

where γ and $\bar{\gamma}$ are two positive constants. Then the method has Property* if there exist two constants $b > 1$ and $\lambda > 0$ such that for all k:

$$|\beta_k| \leq b,$$

and

$$\|x_k - x_{k-1}\| \leq \lambda \implies |\beta_k| \leq \frac{1}{2b}.$$

Theorem 3.1. *The OOHS method possesses Property (*).*

Proof. The proof follows straightforwardly from the fact that HS has Property (*) and OOHS is a restriction of HS. □

Now, based on Theorem 3.1, we show the global convergence of the OOHS algorithm, which essentially depends on the following two results:

Result 1. *In [17], Yousif et al. have proven that if a CG method with a parameter β_k satisfying the condition*

$$|\beta_k| \leq \mu \frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \quad \textit{for } k \geq 1 \textit{ and a real number } \mu \geq 1,$$

is applied under the strong Wolfe line search with $0 < \sigma < \frac{1}{4\mu}$ for solving an unconstrained optimization problem, then:

(a) $\frac{\|g_k\|}{\|d_k\|} < 2$, *for all $k \geq 0$,*
(b) *the sufficient descent condition is satisfied,*
(c) *under assumptions (ii) and (iii) , the method is globally convergent.*

Result 2. *In [6], Gilbert and Nocedal have proven that any CG method under Wolfe line search, so under the strong Wolfe line search with* $0 < \sigma < \frac{1}{4\mu}$*, satisfies the following:*

(a) $\beta_k \geq 0$*, for all* k*,*
(b) *the sufficient descent condition holds,*
(c) *Property* holds,*
(d) *assumptions (ii) and (iii) hold,*

then it is globally convergent.

Theorem 3.2. *Suppose that assumptions (ii) and (iii) hold. Then the OOHS algorithm is globally convergent when it is applied under the strong Wolfe line search with* $0 < \sigma < \frac{1}{4\mu}$*.*

Proof. By rewriting β_k^{OOHS} as follows:

$$\beta_k^{\text{OOHS}} = \begin{cases} \beta_k^{\text{HS}}, & \text{if } -\mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2} \leq \beta_k^{\text{HS}} < \mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \\ \beta_k^{\text{HS}}, & \text{if } \beta_k^{\text{HS}} \geq \mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2}, \\ 0, & \text{otherwise.} \end{cases}$$

We can now consider the following two cases:

- Case I: $-\mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2} \leq \beta_k^{\text{OOHS}} < \mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2}$. The proof follows straightforwardly from Result 1.

- Case II: $\beta_k^{\text{OOHS}} \geq \mu\frac{\|g_k\|^2}{\|d_{k-1}\|^2}$. Since OOHS satisfies Property (*), the proof follows straightforwardly from Result 2.

4. Numerical Experiment

In this section, we compare the OOHS method with the HS, HS+, CG-DESCENT, and OHS methods. In the comparison:

- Most of the test problems are from [11].
- To show robustness, test problems are implemented under low, medium, and high dimensions, namely, 2, 3, 4, 10, 50, 100, 500, 1000, 5000, and 10000. Also, for each dimension, two different initial points are used: one is the initial point suggested by Andrei [11], and the other point is chosen arbitrarily.
- All methods are applied under the strong Wolfe line search with the parameters $\delta = 10^{-4}$ and $\sigma = 10^{-1}$.
- For all methods, the termination condition is set to $\|g_k\| \leq 10^{-6}$.
- The parameter μ is set to 10.

The numerical results are presented in Tables 2 and 3 where:

Dim: the dimension of the test problem,
NI: the number of iterations,
CPU: the time required by the computer to solve a test problem,
NF: the number of function evaluations,
NG: the number of gradient evaluations,

FAIL: indicates the method failed to solve the test problem.

Table 2. Numerical results

Problem number	Test Problem	Dim	initial vector	HS NI/CPU/NF/NG	HS+ NI/CPU/NF/NG	CG-DESCENT NI/CPU/NF/NG	OHS NI/CPU/NF/NG	OOHS NI/CPU/NF/NG
1	THREE-HUMP	2	(2,2)	30/0.09/931/135	11/0.05/313/84	15/0.06/290/107	30/0.09/931/135	30/0.09/931/135
			(5,5)	26/0.07/774/120	10/0.03/234/80	Fail	Fail	24/0.06/668/294
2	GENERALIZED	2	(0,0)	16/0.02/111/53	16/0.02/114/57	16/0.02/109/53	18/0.02/112/62	16/0.02/114/57
			(10,10)	49/0.07/582/210	30/0.04/287/117	37/0.06/413/157	50/0.06/432/195	30/0.04/287/117
3	SIX-HUMP	2	(1.1)	5/0.01/22/14	5/0.01/22/14	8/0.02/31/20	5/0.01/22/14	5/0.01/22/14
			(10,10)	9/0.02/53/24	10/0.03/62/31	8/0.02/53/20	12/0.03/77/35	9/0.03/53/24
4	TRECANNI	2	(1,1)	5/0.01/19/13	5/0.01/20/13	5/0.01/20/13	5/0.01/19/13	5/0.01/19/13
			(10,10)	5/0.01/23/15	8/0.02/35/21	7/0.2/32/18	9/0.02/38/23	8/0.02/35/21
5	ZETTLE	2	(1,1)	10/0.02/40/30	10/ 0.02/42/31	10/0.02/40/30	10/0.02/45/35	10/0.02/40/30
			(10,10)	11/0.02/45/30	10/0.02/44/29	10/0.03/47/34	10/0.02/46/32	10/0.02/44/29
6	BOOTH	2	(1,1)	2/0.01/6/4	2/0.01/6/4	2/0.01/6/4	2/0.01/6/4	2/0.01/6/4
			(10,10)	2/0.01/6/4	2/0.01/6/4	2/0.01/6/4	2/0.01/6/4	2/0.01/6/4
7	LEON	2	(0,0)	16/0.03/111/53	16/0.03/114/57	16/0.03/109/53	18/0.04/112/62	16/0.03/114/57
			(10,10)	49/0.07/582/210	30/0.05/287/117	37/0.06/413/157	50/0.07/432/195	30/0.05/287/117
8	DIXON PRICE	3	(1,1,1)	10/0.01/40/27	10/ 0.01/40/27	13/0.02/49/33	10/ 0.01/40/27	10/ 0.01/40/27
			(10,10,10)	24/0.03/100/60	23/0.03/111/62	50/0.04/200/120	21/0.02/105/58	21/0.02/105/58
9	CUBE	3	(-1.2,1.-1,2)	391.0.21/1589/908	278/0.16/1222/664	449/.28/1897/1053	78/0.07/465/257	278/0.16/1222/664
			(0,0,0)	474/0.28/1953/1117	1946/.97/6896/4412	1336/.71/4771/2990	82/0.07/474/270	1946/0.97/6896/4412
10	NONDIA	3	(-1,-1,-1)	557/0.27/1981/1261	557/0.27/1981/1261	389/0.24/1264/802	49/0.17/264/135	557/0.27/1981/1261
			(10,10,10)	374/0.23/1661/905	895/0.50/3426/2025	139/0.11/724/353	103/0.09/547/308	895/0.50/3426/2025
11	EXTENDED WOOD	4	(0,0,0,0)	123/0.07/481/282	123/0.07/481/282	120/0.06/466/274	51/0.04/225/128	123/0.07/481/282
			(5,5,5,5)	167/0.10/736/401	125/0.08/567/306	240/0.14/975/561	74/0.06/397/200	70/0.05/383/187
12	LIARWHD	4	(0,0,0,0)	1/0.01/3/2	1/0.01/3/2	1/0.01/3/2	1/0.01/3/2	1/0.01/3/2
			(0,5,0,5)	FAIL	21/0.05/383/86	92/0.31/2760/462	26/0.09/744/239	21/0.05/434/94
13	COLVILLE	4	(0,0,0,0)	123/0.06/481/282	123/0.06/481/282	120/0.06/466/274	51/0.03/225/128	123/0.06/481/282
			(10,10,10,10)	91/0.06/537/245	182/0.10/847/451	191/0.10/728/430	73/0.05/408/202	177/0.09/848/440
14	EXTENDED POWELL	4	(1,1,1,1)	70/0.04/279/172	1303/.49/3957/2635	256/0.11/835/557	56/0.03/259/178	35/0.03/173/105
			(10,10,10,10)	1290/.48/3964/2623	1290/.48/3964/2623	301/0.12/988/643	59/0.04/314/212	1290/0.48/3964/2623
15	ENGVAL1	4	(2,2,2,2)	23/0.02/79/52	24/0.02/84/56	21/0.02/77/51	23/0.02/79/52	23/0.02/79/52
			(10,10,10,10)	22/0.02/85/53	18/0.02/74/46	21/0.02/84/52	22/0.02/85/53	22/0.02/85/53
16	BIGGSB1	10	(0,0,. . .)	5/0.01/17/12	5/0.01/17/12	5/0.01/17/12	5/0.01/17/12	5/0.01/17/12
			(10,10,. . .)	5/0.01/17/12	5/0.01/17/12	5/0.01/17/12	5/0.01/17/12	5/0.01/17/12
17	GENERALIZED TRIDIAGONAL 2	10	(1,1,. . .)	12/0.02/40/28	12/0.02/40/28	12/0.02/41/29	12/0.02/40/28	12/0.02/40/28
			(10,10,. . .)	35/0.03/133/91	40/0.04/145/98	38/0.04/136/100	37/0.03/132/93	35/0.03/133/91
18	GENERALIZED TRIDIAGONAL 1	10	(2,2,. . .)	23/0.02/76/50	23/0.02/76/50	23/0.02/76/50	23/0.02/76/50	23/0.02/76/50
			(10,10,. . .)	27/0.03/112/68	27/0.03/112/68	27/0.03/111/67	27/0.03/112/68	27/0.03/112/68
19	NONSCOMP	10	(3,3,. . .)	2123/0.85/7282/4821	2123/.85/7282/4821	3056/1.31/9959/6624	FAIL	2123/0.85/7282/4821
			(10,10,. . .)	3247/1.23/10496/7198	3247/1.23/10496/7198	FAIL	FAIL	3247/1.23/10496/7198
20	SUM SQUARE	10	(-1,-1,. . .)	10/0.01/30/20	10/0.01/30/20	10/0.01/30/20	10/0.01/30/20	10/0.01/30/20
			(10,10,. . .)	10/0.01/30/20	10/0.01/30/20	10/0.01/30/20	10/0.01/30/20	10/0.01/30/20
21	POWER	50	(1,1,. . .)	65/0.04/195/130	65/0.04/195/130	66/0.05/198/132	1478/.84/4434/2965	65/0.04/195/130
			(10,10,. . .)	67/0.05/201/134	67/0.05/201/134	67/0.05/201/134	1781/1.0/5343/3562	67/0.05/201/134
22	HAGER	50	(1,1,. . .)	19/0.02/59/40	19/0.02/59/40	19/0.02/59/40	19/0.02/59/40	19/0.02/59/40
			(5,5,. . .)	21/0.02/76/49	21/0.02/76/49	21/0.02/76/49	21/0.02/76/49	21/0.02/76/49
23	EDENSCH	50	(0,0,. . .)	23/0.02/80/52	274/0.21/1319/664	AIL	24/0.03/90/52	23/0.02/80/52
			(10,10,. . .)	25/0.03/107/64	27/0.03/109/64	FAIL	27/0.03/125/67	25/0.03/107/64
24	RAYDAN1	50	(1,1,. . .)	47/0.03/144/137	47/0.03/144/137	47/0.03/144/138	47/0.03/144/137	47/0.03/144/137
			(5,5,. . .)	90/0.06/342/278	90/0.06/342/278	91/0.07/361/276	135/0.09/522/400	90/0.06/342/278
25	ARWHEAD	50	(1,1,. . .)	5/0.02/23/13	5/0.02/23/13	5/0.02/22/12	5/0.02/23/13	5/0.02/23/13
			(10,10,. . .)	9/0.03/61/25	9/0.03/61/25	10/0.03/68/28	10/0.03/68/28	9/0.03/61/25

Table 3. Numerical results

Problem number	Test Problem	Dim	initial vector	HS	HS+	CG-DESCENT	OHS	OOHS
				NI/CPU/NF/NG	NI/CPU/NF/NG	NI/CPU/NF/NG	NI/CPU/NF/NG	NI/CPU/NF/NG
26	FLETCHER	50	(0,0,. . .)	359/ 0.39/1832/867	359/ 0.39/1832/867	366/ 0.42/1859/886	1861/ 1.53/6380/3942	359/ 0.39/1832/867
			(10,10,. . .)	329/0.28/1220/747	323/0.28/1211/741	345/0.32/1300/800	1846/1.42/5811/3806	323/0.28/1211/741
27	GENERALIZED ROSENBROCK	100	(-1.2,1,. . .)	855/1.13/4845/2239	852/ 1.11/4793/2225	834/1.13/4713/2170	2748/2.88/11673/6561	55/1.13/4845/2239
			(5,5,. . .)	863/1.15/4919/2262	853/1.13/4881/2228	846/1.16/4847/2222	3017/3.10/12521/7131	863/1.15/4919/2262
28	HIMMELH	100	(0,0,. . .)	5/0.02/15/10	6/0.02/22/12	5/0.02/15/10	6/0.02/25/13	6/0.02/29/12
			(0.5,0.5,. . .)	5/0.02/15/10	5/0.02/15/10	5/0.02/15/10	5/0.02/15/10	5/0.02/15/10
29	GENERALIZED QUARTIC	100	(1,1,. . .)	10/0.04/149/73	7/0,03/114/62	9/0.05/225/74	9/ 0.04/175/97	7/0.03/114/62
			(2,2,. . .)	8/0.02/63/40	6/0.02/56/35	10/0.04/200/53	5/0.02/18/13	6/0.02/56/36
30	EXTENDED MARATOS	100	(1.1,0.1,. . .)	FAIL	38/0.07/309/117	30/0.05/209/100	49/0.07/312/154	35/0.06/265/118
			(2,2,. . .)	15/0.03/108/49	15/0.02/82/45	13/0.02/86/41	23/0.04/138/72	15/0.02/82/45
31	EXTENDED PENALTY	500	(1,2,3,. . .)	FAIL	119/0.30/497/276	FAIL	26/0.09/146/68	119/0.30/497/276
			(2,2,. . .)	20/0.09/145/67	20/0.09/150/70	FAIL	10/0.05/79/28	23/0.11/178/74
32	TRIDIA	500	(1,1,. . .)	239/0.45/717/478	239/0.45/717/478	240/0.48/720/480	FAIL	239/0.45/717/478
			(10,10,. . .)	251/0.52/753/502	251/0.49/753/502	252/0.51/756/504	FAIL	251/0.49/753/502
33	QF2	500	(0.5,0.5,. . .)	253/0.54/897/563	253/0.54/897/563	254/0.56/900/565	207/0.47/767/479	253/0.54/897/563
			(10,10,. . .)	225/0.53/885/515	225/0.53/885/515	252/0.60/976/577	225/0.53/885/515	225/0.53/885/515
34	QP2	500	(1,1,. . .)	39/0.31/469/136	39/ 0.30/456/132	38/0.32/482/124	57/0.40/613/190	39/0.30/460/131
			(10,10,. . .)	FAIL	40/0.31/457/130	41/0.34/516/137	57/0.40/610/188	39/0.30/451/130
35	QF1	500	(1,1,. . .)	131/0.25/393/262	131/0.25/393/262	131/0.25/393/262	131/0.25/393/262	131/0.25/393/262
			(10,10,. . .)	140/0.27/420/280	140/0.27/420/280	140/0.27/420/280	140/0.27/420/280	140/0.27/420/280
36	QP1	1000	(0,5,. . .)	12/0.10/100/37	13/0.09/86/38	FAIL	12/0.10/106/31	13/0.09/86/38
			(3,3,. . .)	14/0.12/123/45	9/0.06/57/21	FAIL	9/0.06/50/21	9/0.06/57/21
37	PERTURBED QUADRATIC	1000	(0,5,. . .)	187/0.61/561/374	187/0.61/561/374	187/0.61/561/374	523/1.71/1569/1046	187/0.61/561/374
			(10,10,. . .)	203/0.67/609/406	203/0.67/609/406	203/0.67/609/406	629/2.12/1887/1258	203/0.67/609/406
38	DIXON3DQ	1000	(-1,-1,. . .)	500/1.35/1508/1009	500/1.35/1508/1009	500/1.35/1508/1009	FAIL	500/1.35/1508/1009
			(10,10,. . .)	500/1.35/1508/1009	500/1.35/1508/1009	500/1.35/1508/1009	FAIL	500/1.35/1508/1009
39	DQDRTIC	1000	(3,3,. . .)	13/0.04/39/26	13/0.04/39/26	13/0.04/39/26	10/0.03/30/20	13/0.04/39/26
			(10,10,. . .)	18/0.06/54/36	16/0.06/48/32	13/0.05/39/26	11/0.04/33/22	18/0.06/54/36
40	EXTENDED DENSCHNF	1000	(2,2,. . .)	14/0.43/447/266	16/0.51/544/308	27/0.71/745/318	11/0.32/330/221	11/0.32/330/221
			(10,10,. . .)	FAIL	20/0.34/343/118	27/0.62/645/189	20/0.50/520/238	20/0.50/520/238
41	FREUDENSTEIN & ROTH	5000	(0.5,-2,. . .)	FAIL	7/0.15/35/18	FAIL	7/0.15/35/18	7/0.15/35/18
			(5,5,. . .)	10/0.21/52/25	8/0.21/48/22	FAIL	FAIL	8/0.21/48/22
42	EXTENDED TRIDIAGONAL1	5000	(2,2,. . .)	14/0.42/71/60	14/0.42/71/60	14/0.44/75/62	12/0.37/61/51	14/0.42/71/60
			(10,10,. . .)	7/ 0.24/42/35	17/0.45/75/59	11/0.39/69/59	8/0.27/47/38	7/0.24/42/35
43	DIAGONAL 4	5000	(1,1,. . .)	2/0.03/6/5	2/0.03/6/5	2/0.03/6/5	2/0.03/6/5	2/0.03/6/5
			(10,10,. . .)	2/0.03/6/5	2/0.03/6/5	2/0.03/6/5	2/0.03/6/5	2/0.03/6/5
44	EXTENDED DENSCHNB	5000	(1,1,. . .)	5/0.08/19/14	5/0.08/19/14	5/0.08/19/14	5/0.08/19/14	5/0.08/19/14
			(10,10,. . .)	8/0.14/34/21	10/0.17/45/29	9/0.16/41/27	9/0.15/37/23	8/0.14/34/21
45	EXTENDED ROSENBROCK	5000	(-1.2,1,. . .)	19/0.42/120/58	20/0.43/123/69	18/0.42/121/60	28/0.59/168/90	22/ 0.49/141/71
			(10,10,. . .)	23/0.59/176/70	31/0.74/220/102	22/0.58/172/71	33/0.75/219/102	32/0.75/222/104
46	EXTENDED HIMMELBLAU	10000	(1,1,. . .)	7/0.23/29/17	8/0.25/32/19	7/0.23/30/18	8/0.25/32/19	8/0.25/32/19
			(10,10,. . .)	7/ 0.24/32/17	8/0.26/34/18	6/0.22/28/15	8/0.27/35/19	8/0.27/35/19
47	STRAIT	10000	(0,0,. . .)	18/0.62/90/51	18/0.62/90/51	13/0.50/73/40	15/0.55/80/44	18/0.62/90/51
			(5,5,. . .)	20/0.85/129/59	20/1.00/156/63	19/0.93/144/64	19/1.16/184/74	21/1.20/190/78
48	SHALLOW	10000	(0,0,. . .)	7/0.20/27/21	7/ 0.21/28/21	7/0.20/27/21	8/0.23/31/24	7/ 0.19/27/20
			(10,10,. . .)	13/0.46/63/42	13/0.45/64/40	13/0.46/66/39	12/0.43/62/39	11/0.38/53/34
49	EXTENDED BEALE	10000	(1,0.8, . . .)	10/0.70/48/31	10/0.73/49/32	11/0.73/48/29	13/0.84/58/37	10/0.72/49/32
			(3,3,. . .)	11/0.78/53/29	11/0.81/56/34	11/ 0.81/57/30	13/0.94/66/40	12/0.83/58/34
50	EXTENDED WHITE & HOLST	10000	(-1.2,1,. . .)	15/1.18/113/47	14/1.16/112/49	15/1.18/104/45	15/1.02/95/50	14/1.16/112/49
			(10,10,. . .)	50/5.89/580/212	31/3.05/293/125	38/4.27/416/159	51/4.55/435/197	30/3.02/293/125

To show the method of the best performance, we used the technique that was introduced by Dolan and More [2]. The results are in Figures 1-4.

In this performance profile:

- $t_{p,m}$ is the result (may be NI, CPU, NF, or NG in our experiment) when a method m is applied to solve problem p.
- $t = \frac{t_{p,m}}{\min\{t_{p,m} : m \in M\}}$.
- $P_m(t)$ is the probability that a method m has a performance ratio t.

Therefore, based on this performance profile, the left side shows the best performance (having minimum NI, CPU time, NF, and NG), meaning that the highest curve corresponds to the best method. Also, the right side measures the percentage of the total number of test problems that are successfully solved by the corresponding method.

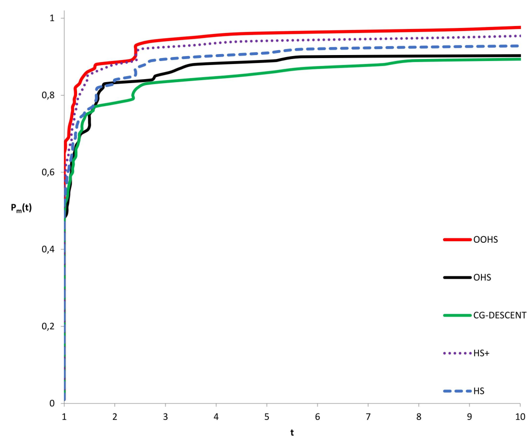

FIGURE 1. The performance in terms of NI

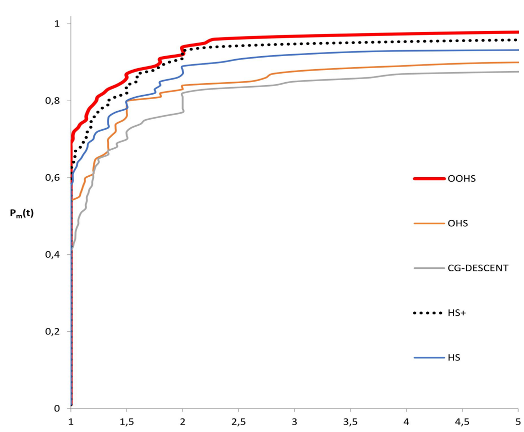

FIGURE 2. The performance in terms of CPU

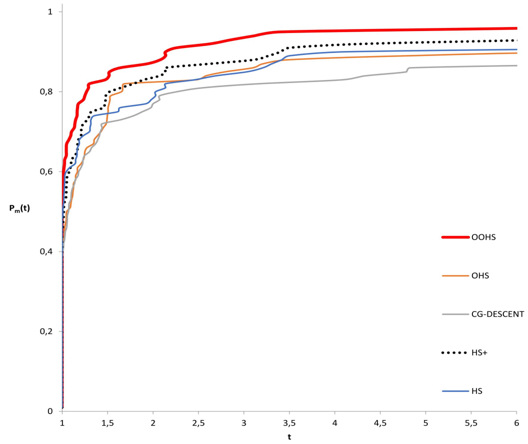

FIGURE 3. The performance in terms of NF

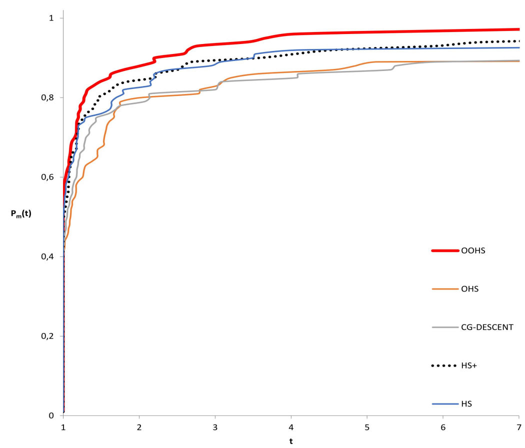

FIGURE 4. The performance in terms of NF

The left side of Figures 1-4 show that although in some cases OOHS seem to be similar to HS+ method but it has the minimum number of NI, CPU, NF, and NG. Also, the right side of all figures show that the OOHS method solve the entire given test problems. Therefore, we can conclude that OOHS has the best performance.

5. Conclusion

In this paper, motivated by HS+ and OHS parameters, a new conjugate gradient update parameter of HS was proposed, hence, another modified method of HS. The new method inherits the convergence properties of HS+ and OHS methods. To show the efficiency and robustness of the modified method in practical computation, it was compared under the strong Wolfe line search with HS, HS+, CG-DESCENT, and OHS methods. The comparison was based on four items; the number of iterations, CPU time, the number of function and number of gradient evaluations. It was reported that the new modified method performs better than the others.

Acknowledgement: The Researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support (QU-APC-2025).

Conflicts of Interest: The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] B.T. Polyak, The Conjugate Gradient Method in Extremal Problems, USSR Comput. Math. Math. Phys. 9 (1969), 94–112. https://doi.org/10.1016/0041-5553(69)90035-4.

[2] E.D. Dolan, J.J. Moré, Benchmarking Optimization Software with Performance Profiles, Math. Program. 91 (2002), 201–213. https://doi.org/10.1007/s101070100263.

[3] E. Polak, G. Ribiere, Note Sur La Convergence de Méthodes de Directions Conjuguées, Rev. Fr. Inform. Rech. Oper. Ser. Rouge. 3 (1969), 35–43. https://doi.org/10.1051/m2an/196903R100351.

[4] G. Yuan, X. Lu, A Modified PRP Conjugate Gradient Method, Ann. Oper. Res. 166 (2009), 73–90. https://doi.org/10.1007/s10479-008-0420-4.

[5] G. Zoutendijk, Nonlinear Programming Computational Methods, in: J. Abadie (Ed.), Integer and Nonlinear Programming, North-Holland, Amsterdam, pp. 37–86, 1970.

[6] J.C. Gilbert, J. Nocedal, Global Convergence Properties of Conjugate Gradient Methods for Optimization, SIAM J. Optim. 2 (1992), 21–42. https://doi.org/10.1137/0802003.

[7] L. Zhang, An Improved Wei–Yao–Liu Nonlinear Conjugate Gradient Method for Optimization Computation, Appl. Math. Comput. 215 (2009), 2269–2274. https://doi.org/10.1016/j.amc.2009.08.016.

[8] M. Al-Baali, Descent Property and Global Convergence of the Fletcher—Reeves Method with Inexact Line Search, IMA J. Numer. Anal. 5 (1985), 121–124. https://doi.org/10.1093/imanum/5.1.121.

[9] M.J.D. Powell, Convergence Properties of Algorithms for Nonlinear Optimization, SIAM Rev. 28 (1986), 487–500. https://doi.org/10.1137/1028154.

[10] M.R. Hestenes, E. Steifel, Method of conjugate gradient for solving linear equations, J. Res. Natl. Bur. Stand. 49 (1952), 409–436.

[11] N. Andrei, An Unconstrained Optimization Test Functions Collection, Adv. Model. Optim. 10 (2008), 147–161.

[12] O. Omer, M. Mamat, A. Abashar, M. Rivaie, The Global Convergence Properties of a Conjugate Gradient Method, in: Kuala Lumpur, Malaysia, 2014: pp. 286–295. https://doi.org/10.1063/1.4882501.

[13] O. Omer, M. Rivaie, M. Mamat, Z. Amani, A New Conjugate Gradient Method with Sufficient Descent without Any Line Search for Unconstrained Optimization, AIP Conf. Proc. 1643 (2015), 602–608. https://doi.org/10.1063/1.4907500.

[14] O. Omer, M. Rivaie, M. Mamat, A. Abdalla, A New Conjugate Gradient Method and Its Global Convergence under the Exact Line Search, AIP Conf. Proc. 1635 (2014), 639–646. https://doi.org/10.1063/1.4903649.

[15] O.O.O. Yousif, A. Abdelrahman, M. Mohammed, M.A. Saleh, A Sufficient Condition for the Global Convergence of Conjugate Gradient Methods for Solving Unconstrained Optimisation Problems, Sci. J. King Faisal Univ. 23 (2022), 106-112. https://doi.org/10.37575/b/sci/220013.

[16] O.O.O. Yousif, The Convergence Properties of RMIL+ Conjugate Gradient Method under the Strong Wolfe Line Search, Appl. Math. Comput. 367 (2020), 124777. https://doi.org/10.1016/j.amc.2019.124777.

[17] O.O.O. Yousif, M.A.Y. Mohammed, M.A. Saleh, M.K. Elbashir, A Criterion for the Global Convergence of Conjugate Gradient Methods under Strong Wolfe Line Search, J. King Saud Univ. Sci. 34 (2022), 102281. https://doi.org/10.1016/j.jksus.2022.102281.

[18] P. Wolfe, Convergence Conditions for Ascent Methods, SIAM Rev. 11 (1969), 226–235. https://doi.org/10.1137/1011036.

[19] P. Wolfe, Convergence Conditions for Ascent Methods. II: Some Corrections, SIAM Rev. 13 (1971), 185–188. https://doi.org/10.1137/1013035.

[20] R. Fletcher, Practical Method of Optimization, Wiley, New York, 1997.

[21] R. Fletcher, Function Minimization by Conjugate Gradients, Comput. J. 7 (1964), 149–154. https://doi.org/10.1093/comjnl/7.2.149.

[22] W.W. Hager, H. Zhang, A New Conjugate Gradient Method with Guaranteed Descent and an Efficient Line Search, SIAM J. Optim. 16 (2005), 170–192. https://doi.org/10.1137/030601880.

[23] Y. Liu, C. Storey, Efficient Generalized Conjugate Gradient Algorithms, Part 1: Theory, J. Optim. Theory Appl. 69 (1991), 129–137. https://doi.org/10.1007/BF00940464.

[24] 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.

[25] Y. Yuan, W. Sun, Theory and Methods of Optimization, Science Press, Beijing, 1999.

[26] Z. Dai, F. Wen, A Modified CG-DESCENT Method for Unconstrained Optimization, J. Comput. Appl. Math. 235 (2011), 3332–3341. https://doi.org/10.1016/j.cam.2011.01.046.

[27] Z. Wei, G. Li, L. Qi, New Nonlinear Conjugate Gradient Formulas for Large-Scale Unconstrained Optimization Problems, Appl. Math. Comput. 179 (2006), 407–430. https://doi.org/10.1016/j.amc.2005.11.150.

[28] Z. Wei, S. Yao, L. Liu, The Convergence Properties of Some New Conjugate Gradient Methods, Appl. Math. Comput. 183 (2006), 1341–1350. https://doi.org/10.1016/j.amc.2006.05.150.