Optimization is now considered a branch of computational science. This ethos seeks to answer the question «what is best?» by looking at problems where the quality of any answer can be expressed numerically. One of the most well-known methods for solving nonlinear, unrestricted optimization problems is the conjugate gradient (CG) method. The Hestenes and Stiefel (HS-CG) formula is one of the century's oldest and most effective formulas. When using an exact line search, the HS method achieves global convergence; however, this is not guaranteed when using an inexact line search (ILS). Furthermore, the HS method does not always satisfy the descent property. The goal of this work is to create a new (modified) formula by reformulating the classic parameter HS-CG and adding a new term to the classic HS-CG formula. It is critical that the proposed method generates sufficient descent property (SDP) search direction with Wolfe-Powell line (sWPLS) search at every iteration, and that global convergence property (GCP) for general non-convex functions can be guaranteed. Using the inexact sWPLS, the modified HS-CG (mHS-CG) method has SDP property regardless of line search type and guarantees GCP. When using an sWPLS, the modified formula has the advantage of keeping the modified scalar non-negative sWPLS. This paper is significant in that it quantifies how much better the new modification of the HS performance is when compared to standard HS methods. As a result, numerical experiments between the mHSCG method using the sWPL search and the standard HS optimization problem show that the CG method with the mHSCG conjugate parameter is more robust and effective than the CG method without the mHSCG parameter

Keywords: conjugate gradient method, descent direction, global property, strong Wolfe-Powell line search, unconstrained optimization

UDC 519
DOI: 10.15587/1729-4061.2022.254017

A NEW MODIFIED HS ALGORITHM WITH STRONG POWELL-WOLFE LINE SEARCH FOR UNCONSTRAINED OPTIMIZATION

Ghada Moayid Al-Naemi
Doctor of Mathematics,
Assistance Proffesor
Depatement of Mathematics
University of Mosul
Ninavah str., 17, 202, Mosul, Iraq, 41002
E-mail: drghadaalnaemi@uomosul.edu.iq

Received date 21.01.2022
Accepted date 01.03.2022
Published date 28.04.2022

How to Cite: *Al-Naemi, G. M. (2022). A new modified HS algorithm with strong Powell-Wolfe line search for unconstrained optimization. Eastern-European Journal of Enterprise Technologies, 2 (4 (116)), 14–21. doi: https://doi.org/10.15587/1729-4061.2022.254017*

1. Introduction

The CG method is one of the most popular methods for solving nonlinear unconstrained optimization problems. This method class has a wide range of applications, particularly in the design, construction, and maintenance of any engineering system. Engineers must make numerous technological and managerial decisions at various stages. The ultimate goal of all such decisions is to either minimize the amount of effort required or maximize the desired benefit. Because the effort required or the benefit desired in any practical situation can be expressed as a function of a decision variable, optimization can be defined as the process of determining the conditions under which the maximum or minimum value of a function can be found. The HSCG formula is widely regarded as one of the most effective methods devised during the twentieth century. If the descent condition is not met, the HS parameter may fail to satisfy the CGP of the CG method with the sWPLS. This method class has a wide range of applications, most notably in engineering.

2. Literature review and problem statement

In 1952, HS proposed a CG algorithm for solving symmetric positive definite systems of equations [1], while in 1964, FR [2] presented a nonlinear conjugate gradient method for solving unconstrained optimization problems based on the linear conjugate gradient method. Many researchers have expressed an interest in developing the HS formula to improve algorithm performance while ensuring that the proposed method meets the SDP and CGP. As a result, [3] developed a new formula to ensure that the conjugation condition is met. To demonstrate that the proposed formula is very efficient with any line search, [4] proposed a modified three-term HS-CG formula to show that the proposed formula is extremely efficient with any line search. Based on Armijo-type line search, [5] developed a new formula for HS [6] proposed another formula, different parameters have different iterative effects [7] suggested a new modification to HS. To improve the functionality of this algorithm, the researchers proposed the development of the HS formula [8], [9] proposed a hybrid of the HS formula and the PRP formula to improve the direction's performance.

Because of their low memory requirements and strong local and global convergence properties, CG methods are widely used. Our goal is to minimize a function of n-variables function by solving the optimization problem $\min\{f(x): x\in\mathcal{R}^n\}$, where $f:\mathcal{R}^n\to\mathcal{R}$ is a smooth and non-linear function. The CG method is iterative, the iterative scheme is defined by:

$$x_{j+1}=x_j+s_j,\ s_j=\mu_j d_j,\ \ j=0,1,2,\ldots. \tag{1}$$

The step length $\mu_j>0$ yields by some line search, and the direction d_j is generated by:

$$d_0=-g_0,\ d_{j+1}=-g_{j+1}+\beta_j s_j,\ \ j=0,1,2,\ldots, \tag{2}$$

where the gradient is defined by $g_j = \nabla f(x_j)$, the scalar $\beta_j \in \mathcal{R}$ is determined by various CG methods.

The stopping criterion for the CG line search is often adopted on some version of the Wolfe equations [10]. The global convergence of the mHS method is proven under the sWPLS:

$$f(x_j + \mu_j d_j) \le f(x_j) + \sigma_1 \mu_j g_j^T d_j, \tag{3}$$

$$\left| g(x_j + \mu_j d_j)^T d_j \right| \le -\sigma_2 g_j^T d_j, \tag{4}$$

where $0<\sigma_1<0.5$ and $\sigma_1<\sigma_2<1$ [11] proved that the CG method is globally convergent when they generalized, the absolute value in (4) is replaced by:

$$\sigma_4 g_j^T d_j \le g(x_j + \mu_j d_j)^T d_j \le -\sigma_3 g_j^T d_j, \tag{5}$$

where $\sigma_4 \ge 0$, $0<\sigma_1<\sigma_4<1$, $0<\sigma_3<1$ and $\sigma_3 + \sigma_4 \le 1$. The special case, $\sigma_3=\sigma_4=\sigma_2$ corresponding to sWPLS [12]. These conditions guaranteed that:

$$s_j^T y_j = s_j^T (g_{j+1} - g_j) = s_j^T g_{j+1} - s_j^T g_j,$$

where $y_j=g_{j+1}-g_j$ and $s_j=x_{j+1}-x_j$. By using the special case of (5) for the above equation [13], we get:

$$0 < -(1-\sigma_2) s_j^T g_j \le s_j^T y_j \le -(1+\sigma_2) s_j^T g_j. \tag{6}$$

Furthermore, we have:

$$g_j^T d_j \le -\tau g_j^2, \ \tau > 0, \forall j \ge 0. \tag{7}$$

It is critical to use the inexact line search (ILS) to ensure the global convergence of the nonlinear CG method [13].

3. The aim and objectives of the study

The aim of this study is to propose a method to generate an SD direction with sWPLS at each iteration, which plays an important role to guranatee the GCP for general non-convex functions.

To achieve this aim, the following objectives are accomplished:

– to make sure that the proposed mHS formula is not negative;

– to ensure that the direction obtained by mHS is satisfying (7).

4. Materials and methods

In this paper, all codes are written in FORTRAN 77 double precision and compiled Visual (Fortran 6.6) (default compiler settings). Table 1 lists the names of the test functions that we used. We conduct numerical experiments on some nonlinear unconstrained test functions. These functions are discussed in CUTE [14], and their details are provided in the Appendix, see Andrei [15].

HS proposed a CG algorithm for solving symmetric positive definite equation systems [1], which are defined as follows:

$$\beta_j^{HS} = \frac{g_{j+1}^T y_j}{y_j^T d_j}.$$

We present an alternative to the mHSCG method. The generated directions are always descending, which is one of the method's appealing features. We proposed the following new mHS:

$$\beta_j^{mHS} = \frac{g_{j+1}^T y_j - \dfrac{\| g_{j+1} \|^2 g_{j+1}^T s_j}{\| s_j \|}}{y_j^T s_j}. \tag{8}$$

It is worth noting that if we use an exact line search in (8), we will get $\beta_j^{mHS} = \beta_j^{HS}$.

$$s_j^T g_{j+1} \le s_j \cdot g_{j+1},$$

then

$$\beta_j^{mHS} \le \frac{g_{j+1}^T y_j}{y_j^T s_j} \frac{g_{j+1}^3}{} \le \frac{g_{j+1}^T y_j}{y_j^T s_j} = \beta_j^{HS}.$$

From (1), we have $s_j = \mu_j d_j \Rightarrow d_j = s_j / \mu_j$. Powell's [16] restart criteria, as stated by:

$$\left| g_{j+1}^T g_j \right| \ge 0.2 \| g_{j+1} \|^2. \tag{9}$$

From (7):

$$g_j^T s_j \le -\mu_j \tau \| g_j \|^2 \Rightarrow -g_j^T s_j \ge \mu_j \tau \| g_j \|^2. \tag{10}$$

Since $g_{j+1}^T y_j = \| g_{j+1} \|^2 - g_{j+1}^T g_j$, we used one side of (9), i.e.

$$g_{j+1}^T g_j \ge 0.2 g_{j+1}^2 \Rightarrow g_{j+1}^T y_j = g_{j+1}^2 - 0.2 {g_{j+1}}^2 = 0.8 g_{j+1}^2. \tag{11}$$

Using (5), (10) and (11) in (8), we get:

$$\beta_j^{mHS} \ge \frac{0.8 \| g_{j+1} \|^2 + \dfrac{\| g_{j+1} \|^2}{\| s_j \|} \mu_j \sigma_2 \tau \| g_j \|^2}{y_j^T s_j} \ge 0.$$

This proves that the proposed formula in (8) is non-negative, hence we obtain $0 \le \beta_j^{mHS} \le \beta_j^{HS}$. The following algorithm describes the main steps that we used with sWPLS.

Algorithm (mHS):

1. Choose an initial point $x_0 \in \mathcal{R}^n$, $\varepsilon > 0$ and $g_0 = -\nabla f(x_0)$, set $d_j = -g_j$, when $j=0$.
2. If $g_j \le \varepsilon$, $g_j \le \varepsilon$, stop; otherwise, go to 3.
3. Determine a step size μ_j by using SWPLS in (3) & (4).
4. Let $x_{j+1}=x_j+s_j$. If $\| g_{j+1} \| \le \varepsilon$, then stop.
5. Calculate the direction $d_{j+1} = -g_{j+1} + \beta_j^{mHS} s_j$.
6. If the restart criteria (9) are achieved, set $d_{j+1} = -g_{j+1}$; go to 2. Otherwise, continue.
7. Set $j=j+1$ and go to 3.

Now we have the following theorem, which shows that the mHS method, when combined with the sWPLS, can guarantee SDP satisfaction.

Consider the following theorem (1) to clarify the direction d_j obtained by mHS and the importance of satisfying (7) in the analysis of GCP [17]. We make the following basic assumptions about the objective function for our subsequent analysis.

Assumption (A):

1) $f(x)$ is restricted to the level defined by $\Lambda=\{x \in R^n, f(x) \le f(x_0)\}$, where x_0 is the starting point. There is, for example, $\eta>0$ that implies $\|x_j\| \le \eta \ \forall x \in \Lambda$ [18].

2) $f(x)$ is continuously differentiable in a specific neighborhood N of Λ, and its gradient is Lipschitz continuous, that is, there exists a constant $L>0$, s. t.

$$\|\nabla f(x)-\nabla f(y)\|\leq \mathcal{L}\|x-y\|,\ \forall x,y\in \mathbb{N}. \quad (12)$$

Now, using assumption (A), there exists a positive constant (ω&ρ), such that [19]:

$$0<\| g_{j+1}\|\leq \omega,\text{ and } 0<\| g_j\|\leq \rho,\ \forall x\in \Lambda.$$

Theorem (1). Impose that Assumptions (A) are true. If the methods (1) & (2) with β_j^{mHS} satisfy (8), and the step size μ_j satisfy sWPLS (3), (4), then there exists a constant $\Lambda>0$, s. t.

$$g_{j+1}^T d_{j+1}\leq -\lambda\| g_{j+1}\|^2,\ \lambda>0,\ \forall j\geq 0,$$

with

$$\lambda=\left[1-\frac{(1.2-\omega)\omega\eta}{\tau a(1-\sigma_2)b^2}\right]. \quad (13)$$

Proof. Firstly, for $j=0$, the proof is trivial, namely $d_0=g_0\Rightarrow$ $\Rightarrow g_0^T d_0=-g_0^2$.
Multiplying both sides of (2) by g_{j+1}^T:

$$g_{j+1}^T d_{j+1}=-g_{j+1}^T g_{j+1}+\frac{g_{j+1}^T y_j-\dfrac{\| g_{j+1}\|^2}{\| s_j\|}g_{j+1}^T s_j}{y_j^T s_j}g_{j+1}^T s_j. \quad (14)$$

By using another side of (9), we have $g_{j+1}^T g_j\leq -0.2\|g_{j+1}\|^2$, so:

$$g_{j+1}^T y_j\leq\| g_{j+1}\|^2+0.2\| g_{j+1}\|^2=1.2\| g_{j+1}\|^2. \quad (15)$$

Now, put (6), (7) & (15) in (14), we will get:

$$g_{j+1}^T d_{j+1}\leq\| g_{j+1}\|^2+$$
$$+\left[\frac{1.2\| g_{j+1}\|^2-\dfrac{\| g_{j+1}\|^2}{\| s_j\|}\| g_{j+1}\|\cdot\| s_j\|}{\tau a(1-\sigma_2)\| g_j\|^2}\right]\| g_{j+1}\|\cdot\| s_j\|\leq$$
$$\leq-\left[1-\frac{(1.2-\omega)\omega\eta}{\tau a(1-\sigma_2)b^2}\right]\cdot\| g_{j+1}\|^2\leq-\lambda\| g_{j+1}\|^2,$$

with $\lambda=\left[1-\dfrac{(1.2-\omega)\omega\eta}{\tau a(1-\sigma_2)b^2}\right]$.

In this section, we want to demonstrate the GCP of mHS under certain assumptions (A). The following lemmas are required, which are commonly used to prove GCP and are provided by Zoutendijk [10].

Theorem (2). Assume Assumption (A) is correct. Assume any iteration method (1) and (2), and μ_j obtained by the sWPLS (3), (4). If

$$\sum_{j\geq 1}\frac{1}{\| d_{j+1}\|^2}=\infty, \quad (16)$$

Then

$$\liminf_{j\to\infty}\| g_{j+1}\|=0. \quad (17)$$

Theorem (3). Consider that assumption (A) is established. Suppose that the algorithm mHS, and μ_j is obtained by the sWPLS and d_{j+1} is the descent direction. Then

$$\liminf_{j\to\infty}\| g_{j+1}\|=0.$$

Proof. We have $d_{j+1}\neq 0$ because the descending property holds. As a result, lemma (1) is sufficient to show that $\|d_{j+1}\|$ is bounded above. Derived from (2), (8), and taking the norm of both sides of the above equation, we get:

$$\| d_{j+1}\|=\left\|-g_{j+1}+\frac{g_{j+1}^T y_j-\dfrac{\| g_{j+1}\|^2}{\| s_j\|}s_j^T g_{j+1}}{y_j^T s_j}s_j\right\|.$$

We are aware of this:

$$g_{j+1}^T s_j=g_{j+1}^T s_j-g_j^T s_j+g_j^T s_j=y_j^T s_j+g_j^T s_j\leq y_j^T s_j.$$

Now from (6), (9), (12), and (15):

$$\tau a(1-\sigma_2)\| g_j\|^2\leq y_j^T s_j\leq \mathcal{L}\|s_j\|^2. \quad (18)$$

We employ the above-mentioned relationships:

$$\|d_{j+1}\|\leq\left\|-g_{j+1}+\frac{1.2g_{j+1}^2-\mathcal{L}g_{j+1}\cdot\|s_j\|}{\tau a(1-\sigma_2)g_j^{\,2}}s_j\right\|\leq$$
$$\leq\left[1+\frac{1.2g_{j+1}+\mathcal{L}\|s_j\|}{\tau a(1-\sigma_2)b^2}\|s_j\|\right]\|g_{j+1}\|\leq$$
$$\leq\left[1+\frac{1.2\omega+\mathcal{L}\eta}{\tau a(1-\sigma_2)b^2}\eta\right]\leq D\cdot\omega=\varphi.$$

The last inequality implies:

$$\sum_{j\geq 1}\frac{1}{\| d_j\|^2}\geq\frac{1}{\varphi^2}\sum_{j\geq 1}1=\infty.$$

The right-hand side of this is infinite, which contradicts the result of Theorem (1). Hence the conclusion (17) holds.

Lemma (1). Assume that Assumption (A) is correct. Let $\{x_j\}$ and $\{d_j\}$ be generated by the mHS algorithm, and μ_j be obtained by the sWPLS (3) & (4), and

$$\sum_{j\geq 1}v_{j+1}-v_j\leq\infty,\text{ where } v_j=\frac{d_j}{d_j}.$$

Proof. To begin, note that $d_j\neq 0$, because otherwise (7) would imply $g_0=0$. As a result, v_j is well defined. Let us now define:

$$w_{j+1}=\frac{-g_{j+1}}{d_{j+1}},\ \alpha_{j+1}=\beta_j^{mHS}\frac{d_j}{d_{j+1}}\geq 0$$

and

$$u_{j+1}=\frac{w_j+1}{d_{j+1}}.$$

As a result, we have:

$$v_{j+1}=\frac{d_{j+1}}{d_{j+1}}=\frac{-g_{j+1}+\beta_j^{mHS}s_j}{d_{j+1}}=w_{j+1}+\alpha_{j+1}\mu_j w_j. \quad (19)$$

We now have $v_{j+1}=v_j=1$,

$$u_{j+1}^2=v_{j+1}-\alpha_{j+1}\mu_j v_j^2=\alpha_{j+1}\mu_j v_{j+1}-v_j^2.$$

Since $\alpha_{j+1}\geq 0$, the triangle inequality and (19) provide us with:

$$v_{j+1}-v_j\leq\left(1+\alpha_{j+1}\mu_j\right)\left(v_{j+1}-v_j\right)\leq$$
$$\leq v_{j+1}-\alpha_{j+1}\mu_j v_j+\alpha_{j+1}\mu_j v_{j+1}-v_j=2v_{j+1}. \quad (20)$$

As a result of the definition of $w_{(j+1)}$, it follows that:

$$w_{j+1}\cdot d_{j+1}=g_{j+1}\leq\omega. \quad (21)$$

As a result of (20) and (21), we have

$$v_{j+1}-v_j^2\leq\left(2v_{j+1}\right)^2\leq 4\omega^2/d_{j+1}^2.$$

Now, we can solve the above inequality by adding the sum of both sides and squaring it:

$$\sum_{j\geq 1}v_{j+1}-v_j^2=\sum_{j\geq 1}\frac{4\omega^2}{d_{j+1}^2}<+\infty.$$

When the mHS algorithm generates a smaller step-size, the next search direction will automatically approach $d_j=-g_j$. Furthermore, small step-size is not continuously genetized. This is primarily due to the following property: β_j^{mHS}. Because the step-size is small enough, mHS tends to zero. Property (*) was introduced for the first time by Gillebert and Nocedal [17].

Property (*). Assuming $\bar{\omega}\leq\| g_{j+1}\|\leq\omega$, we say that the algorithm has property (*), if for all j, there exist constants $b>1$ and $\eta>0$ s.t.

$$\left|\beta_j\right|\leq b \quad (22)$$

and we have:

$$s_j\leq\eta\Rightarrow\left|\beta_j\right|\leq\frac{1}{2b}. \quad (23)$$

Lemma (2). Consider the case where assumption (A) holds and algorithm (mHS) generates the sequences $\{g_j\}$ and $\{d_j\}$. The new formula β_j^{mHS} then has the property (*).

Proof. By (8), sWPLS (4)&(7):

$$\left|\beta_j^{mHS}\right|=\left|\frac{g_{j+1}^T y_j-\dfrac{g_{j+1}^2}{s_j}s_j^T g_{j+1}}{y_j^T s_j}\right|\leq\frac{g_{j+1}y_j+g_{j+1}^2}{\tau a\left(1-\sigma_2\right)g_j^2}.$$

Since $y_j=g_{j+1}-g_j\leq g_{j+1}+g_j$:

$$\left|\beta_j^{mHS}\right|\leq\frac{2g_{j+1}^2+g_{j+1}g_j}{\tau a\left(1-\sigma_2\right)g_j^2}\leq\frac{2\omega^2+\omega\rho}{\tau a\left(1-\sigma_2\right)\bar{\rho}^2}=b. \quad (24)$$

We have, $s_j=x_{j+1}-x_j\leq x_{j+1}+x_j\leq 2\eta$.

Furthermore, from (6), (7), (12) and the above inequality, we get:

$$\left|\beta_j^{mHS}\right|=\left|\frac{g_{j+1}^T y_j-\dfrac{\| g_{j+1}\|^2}{\| s_j\|}s_j^T g_{j+1}}{y_j^T s_j}\right|\leq$$
$$\leq\frac{\mathcal{L}\left\|g_{j+1}\right\|\cdot\| s_j\|^2-\| g_{j+1}\|^2\cdot\| s_j\|}{\tau a\left(1-\sigma_2\right)\| g_j\|^2}\leq\left[\frac{4\mathcal{L}\omega\eta-\omega^2}{\tau a\left(1-\sigma_2\right)\bar{\rho}^2}\right]\| s_j\|. \quad (25)$$

Now, if we allow (24) and (25), with

$$b=\frac{2\omega^2+\partial\rho}{\tau a\left(1-\sigma_2\right)\bar{\rho}^2},\text{ and }\eta=\frac{\tau a\left(1-\sigma_2\right)\bar{\rho}^2}{2b\left(\mathcal{L}\omega\eta-\partial^2\right)}.$$

As a result, (22) and (23) are correct. Thus, the algorithm has the property (*).

5. Results of the study

5. 1. Non-negative formula

It has been demonstrated that the new formula remains non-negative, which is critical in ensuring the method's regression, and it turns out that $0\leq\beta_j^{mHS}\leq\beta_j^{HS}$.

5. 2. Global convergence

The proposed method has demonstrated that it achieves the GCP by utilizing the sWPLS, with the help of which the method achieves SDP, and this resulted in the method achieving its desired goal, as demonstrated in theory 3.

The comparison of the proposed mHS (modified Hestenes & Stiefel) and HS (classical Hestenes & Stiefel) algorithms by using (40) well-known test functions is done in Table 1. Our comparisons are based on the number of iterations (*Ni*); the number of restarts (*Nr*), the number of test functions and gradient evaluations (*Nfg*); finally, the total time (in seconds) required to complete the evaluation process. These algorithms implement the sWPLS with $\sigma_1=0.01$ and $\sigma_2=0.1$, and the initial step size μ_j is computed from:

$$\mu=\begin{cases}1,\text{ if } j=1,\\ \mu_{j-1}\cdot\dfrac{d_{j-1}}{d_j},\text{ if } j\geq 2.\end{cases} \quad (26)$$

Each function is tested 10 times to gradually increase the number of variables: $n=1000$, 2000, ..., 10000. The stopping criterion used in this algorithm is $g_{j+1}\leq 10^{-6}$, denoting the Euclidean norm. The unconstrained optimization functions with the given initial points.

Table 2 shows that the percentage performance of the mHS method is slightly better than that of the classical HS methods. We see that the mHS method gave (*Ni* 30.4 %), (*Nr* 7.7 %), (*Nfg* 30.6 %) and finally (Time 8.5 %) compared with the classical HS method. This behavior could be explained by making a small change to the HS method so that the generated direction always satisfies the sufficient descent condition, whereas search directions generated by classical HS methods do not guarantee to satisfy the descent property for some problems.

Table 1

Comparison between mHS and classical HS algorithms

No.	β_j^{mHS}				β_j^{HS}			
	Ni	*Nr*	*Nfg*	Time	*Ni*	*Nr*	*Nfg*	Time
1	702	391	1.093	2.90	688	384	1.088	2.88
2	1.088	2.141	11.930	6.17	8.119	2.225	12.719	6.75
3	19.436	16.638	473.948	1144.17	20.010	16.958	481.803	1334.03
4	4.504	1.459	7.471	15.81	4.756	1.465	7.795	16.26
5	20.010	17.615	539.243	835.24	20.010	17.609	539.371	837.28
6	4.590	1.389	7.499	15.54	4.550	1.361	7.481	15.77
7	19.436	16.638	473.948	1154.95	20.010	17.257	507.993	1207.28
9	172	114	314	0.15	170	118	308	0.15
10	7.530	2.141	11.930	6.15	8.815	2.406	13.829	7.30
11	90	50	108	0.06	99	59	198	0.08
12	96	74	310	0.12	94	73	306	0.09
13	598	351	1.009	2.12	596	348	998	2.15
14	196	99	416	0.15	200	100	428	0.16
15	259	138	587	0.21	262	141	598	0.24
16	179	119	349	0.16	179	119	349	0.16
17	179	119	349	0.14	179	119	349	0.15
18	40	40	90	0.34	40	40	90	0.36
19	102	57	222	0.08	105	58	228	0.09
20	151	88	289	0.14	140	80	270	0.13
21	681	682	1.078	1.14	678	671	1064	1.13
22	118	68	218	0.09	207	109	339	0.17
23	769	238	1.456	0.52	791	211	1.483	0.54
24	357	195	822	0.29	350	190	814	0.28
25	248	207	3.312	0.76	269	230	4.156	1.33
26	357	231	634	1.42	363	239	638	1.49
27	117	62	243	0.11	113	61	234	0.08
28	40	40	90	0.14	40	40	90	0.14
29	612	533	13.673	53.12	946	878	25.218	98.84
30	88	66	594	1.23	122	88	698	11.71
31	310	164	643	0.27	308	168	639	0.25
32	593	227	956	0.55	634	255	1002	0.67
33	13.704	13.295	427.553	899.47	15.946	15.520	502.857	1009.53
34	7.530	2.141	11.930	6.19	8.119	2.225	12.719	6.73
35	4.525	1.448	7.534	15.78	4.660	1.433	7.701	15.86
36	103	53	236	0.08	106	56	242	0.09
37	246	143	501	0.18	250	145	508	0.19
38	40	40	90	0.12	40	40	90	0.12
39	7.830	7.647	227.874	75.58	8.885	8.649	260.459	76.37
40	926	866	24.247	94.75	2.528	2.464	78.851	314.03

Appendix: 1 – ARWHEAD (CUTE); 2 – A Quadratic QF2; 3 – Broyden Tridiagonal; 4 – DENSCHNB (CUTE); 5 – Diagonal1; 6 – Diagonal2; 7 – Diagonal3; 9 – DQDRTIC; 10 – DIXMAANA (CUTE); 11 – DIXON3DQ (CUTE); 12 – DIXMAANF (CUTE); 13 – DIXMAANG (CUTE); 14 – DIXMAANH (CUTE); 15 – DIXMAANI (CUTE); 16 – (CUTE); 17 – DIXMAANJ (CUTE); 18 – Diagonal5; 19 – EDENSCH (CUTE); 20 – Extended Beal U63 (Matrix Rom); 21 – Extended Block Diagonal BD1; 22 – Extended Himmelblau; 23 – Extended Powell; 24 – Extended Rosebrock; 25 – Extended Penalty U52 (Matrix Rom); 26 – Extended Trigonometric; 27 – Extended Tridiagonal-1; 28 – Extended Tridiagonal-2; 29 – Extended Three Exponential Terms; 30 – Extended PSC1; 31 – Extended White & Holst; 32 – Generalized tridiagonal-2; 33 – LIARWHD (CUTE); 34 – Perturbed Quadratic; 35 – Partial Perturbed Quadratic; 36 – Partial perturbed Quadratic PPQ2; 37 – NONDIA (Shanoo-78 CUTE); 38 – Raydan2; 39 – TRIDIA (CUTE); 40 – VARDIM (CUTE)

Table 2

Performance of the proposed methods shown in percentage

Measures	β_i^{HS}	β_i^{mHS}
Ni	100 %	69.6 %
Nr	100 %	92.3 %
Nfg	100 %	69.4 %
Time	100 %	91.5 %

Fig. 1–4 show the method's efficiency in terms of the number of iterations evaluated, the number of restarts, the number of functions and derivatives evaluated, and finally the time required for the two methods, in comparison to the proposed β_i^{mHS} and classical β_i^{HS}, as demonstrated below.

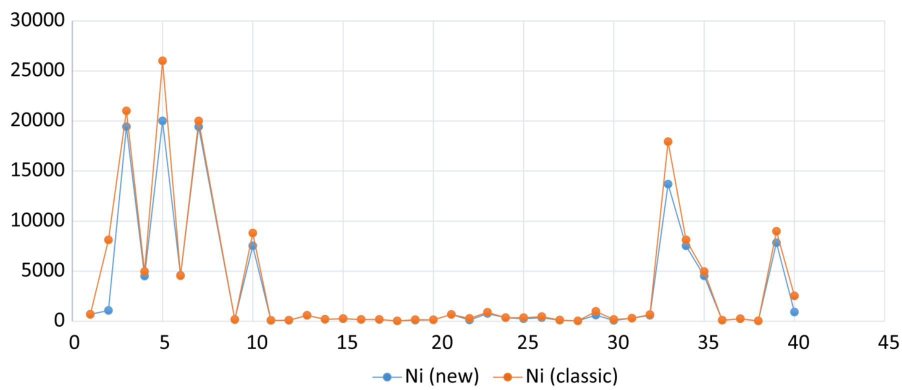

Fig. 1. The difference between the number of evaluation iterations for β_i^{mHS} and β_i^{HS}

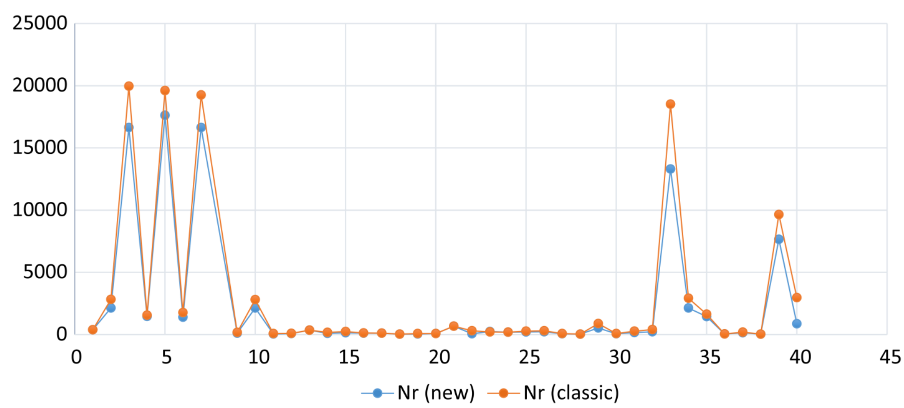

Fig. 2. The difference between the number of restart evaluations for β_i^{mHS} and β_i^{HS}

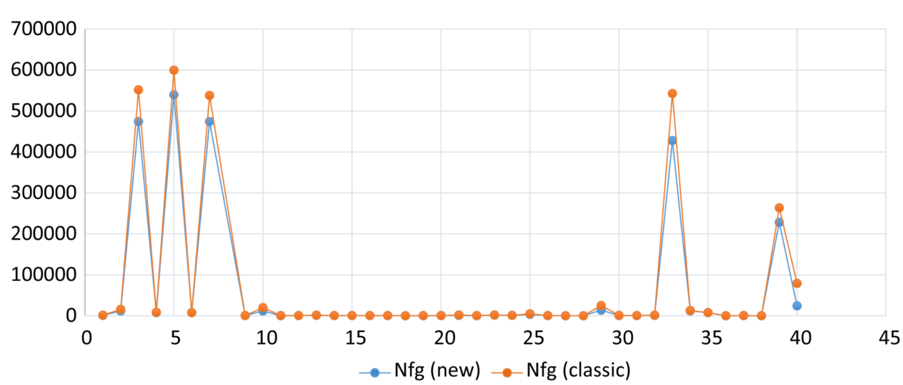

Fig. 3. The difference between the number of functions and the gradient evaluation for β_i^{mHS} and β_i^{HS}

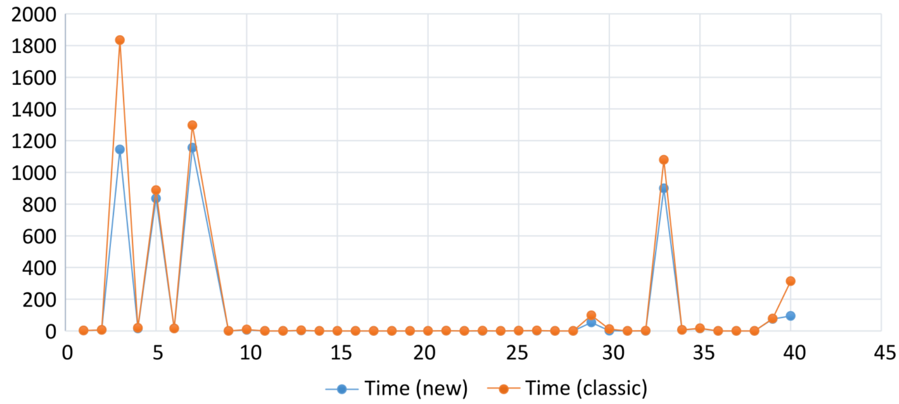

Fig. 4. The time difference for β_i^{mHS} and β_i^{HS}

6. Discussion of experimental results

The numerical results demonstrated that the proposed mHS method is more effective and faster than the original HS formula. The proposed mHS method was faster than the HS method, as evidenced by the percentages in Table 2. The proposed algorithm is sensitive to changing any of the parameter values (σ_1 and σ_2) or the line search (3) and (4), any change that leads to either improving the method, increasing its efficiency, and obtaining the best results, or failing the method and obtaining poor and inefficient results.

Using universal test functions, which are always used to validate the proposed methods, the numerical results demonstrated that the proposed mHS method is more effective and faster than the original HS formula. Its goal is to evaluate the efficacy of the proposed method in the context of other lines of research, as well as the extent to which it influences numerical results. These methods are extremely sensitive to changing the $\sigma_1=0.01$ and $\sigma_2=0.1$ values in sWPLS, and any change in these values will result in either better results and completion of the tasks, or worse results and failure to complete the tasks.

7. Conclusions

1. It was mathematically proven that the proposed formula is non-negative, resulting in the method's efficiency.

2. The algorithm meets the condition of sufficient regression, which is critical for obtaining comprehensive convergence, as demonstrated by theorem (1), and this was demonstrated through the proof of theorem (3). The numerical results represent using cumulative by comparing the CG method with the proposed CG parameter with the CG parameter HS. Fig. 1–4 depict the performance profile for the number of iterations evaluated, the number of restarts, the number of functions and derivatives evaluated, and the CPU time. As a result, according to these figures, the CG method with the proposed parameter is always on top of the curve in the performance profile when compared to the other.

Acknowledgments

The author would like to express gratitude to the University of Mosul's College of Computer Sciences and Mathematics for their encouragement and support.

References

1. Hestenes, M. R., Stiefel, E. (1952). Methods of conjugate gradients for solving linear systems. Journal of Research of the National Bureau of Standards, 49 (6), 409. doi: https://doi.org/10.6028/jres.049.044
2. Fletcher, R. (1964). Function minimization by conjugate gradients. The Computer Journal, 7 (2), 149–154. doi: https://doi.org/10.1093/comjnl/7.2.149
3. Dai, Y.-H. (2001). New Conjugacy Conditions and Related Nonlinear Conjugate Gradient Methods. Applied Mathematics and Optimization, 43 (1), 87–101. doi: https://doi.org/10.1007/s002450010019
4. Al-Naemi, G. M. (2014). A Modified Hestenes-Stiefel Conjugate Gradient Method and its Global convergence for unconstrained optimization. Iraqi Journal of Science, 55 (1), 202–217. Available at: https://iasj.net/iasj/download/9be4a3f4393e9e31
5. Li, Y., Du, S. (2019). Modified HS conjugate gradient method for solving generalized absolute value equations. Journal of Inequalities and Applications, 2019 (1). doi: https://doi.org/10.1186/s13660-019-2018-6
6. Wang, G., Shan, R., Huang, W., Liu, W., Zhao, J. (2017). Two new spectral conjugate gradient algorithms based on Hestenes-Stiefel. Journal of Algorithms & Computational Technology, 11 (4), 345–352. doi: https://doi.org/10.1177/1748301817725296
7. Salleh, Z., Alhawarat, A. (2016). An efficient modification of the Hestenes-Stiefel nonlinear conjugate gradient method with restart property. Journal of Inequalities and Applications, 2016 (1). doi: https://doi.org/10.1186/s13660-016-1049-5
8. Japri, N. A., Basri, S., Mamat, M. (2021). New modification of the Hestenes-Stiefel with strong Wolfe line search. AIP Conference Proceedings. doi: https://doi.org/10.1063/5.0053211
9. Wu, X., Zhu, Y., Yin, J. (2021). A HS-PRP-Type Hybrid Conjugate Gradient Method with Sufficient Descent Property. Computational Intelligence and Neuroscience, 2021, 1–8. doi: https://doi.org/10.1155/2021/2087438

10. Zoutendij, G. (1970). Nonlinear programming, computational methods. Integer and nonlinear programming, 143, 37–86.
11. Malik, M., Mamat, S., Abas, S., Sulaiman, I. M., Sukono (2020). A new coefficient of the conjugate gradient method with the sufficient descent condition and global convergence properties. Engineering Letters, 28 (3), 704–714.
12. Hager, W., Zhang, H. (2006). A survey of non-linear conjugate gradient methods. Pacific Journal of Optimization, 2, 35–58.
13. Gilbert, J. C., Nocedal, J. (1992). Global Convergence Properties of Conjugate Gradient Methods for Optimization. SIAM Journal on Optimization, 2 (1), 21–42. doi: https://doi.org/10.1137/0802003
14. Bongartz, I., Conn, A. R., Gould, N., Toint, P. L. (1995). CUTE. ACM Transactions on Mathematical Software, 21 (1), 123–160. doi: https://doi.org/10.1145/200979.201043
15. Andrei, N. (2013). Test functions for unconstrained optimization. ICI Tecchnical Report, 3–5.
16. Powell, M. J. D. (1977). Restart procedures for the conjugate gradient method. Mathematical Programming, 12 (1), 241–254. doi: https://doi.org/10.1007/bf01593790
17. Al-Namat, F. N., Al-Naemi, G. M. (2020). Global Convergence Property with Inexact Line Search for a New Hybrid Conjugate Gradient Method. OALib, 07 (02), 1–14. doi: https://doi.org/10.4236/oalib.1106048
18. Jardow, F. N., Al-Naemi, G. M. (2021). A new hybrid conjugate gradient algorithm as a convex combination of MMWU and RMIL nonlinear problems. Journal of Interdisciplinary Mathematics, 24 (3), 637–655. doi: https://doi.org/10.1080/09720502.2020.1815346