Journal of Inequalities and Applications
a SpringerOpen Journal

RESEARCH Open Access

An efficient modification of the Hestenes-Stiefel nonlinear conjugate gradient method with restart property

Zabidin Salleh* and Ahmad Alhawarat

*Correspondence:
zabidin@umt.edu.my
School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu, Kuala Terengganu, Terengganu 21030, Malaysia

Abstract

The conjugate gradient (CG) method is one of the most popular methods to solve nonlinear unconstrained optimization problems. The Hestenes-Stiefel (HS) CG formula is considered one of the most efficient methods developed in this century. In addition, the HS coefficient is related to the conjugacy condition regardless of the line search method used. However, the HS parameter may not satisfy the global convergence properties of the CG method with the Wolfe-Powell line search if the descent condition is not satisfied. In this paper, we use the original HS CG formula with a mild condition to construct a CG method with restart using the negative gradient. The convergence and descent properties with the strong Wolfe-Powell (SWP) and weak Wolfe-Powell (WWP) line searches are established. Using this condition, we guarantee that the HS formula is non-negative, its value is restricted, and the number of restarts is not too high. Numerical computations with the SWP line search and some standard optimization problems demonstrate the robustness and efficiency of the new version of the CG parameter in comparison with the latest and classical CG formulas. An example is used to describe the benefit of using different initial points to obtain different solutions for multimodal optimization functions.

Keywords: conjugate gradient method; Wolfe-Powell line search; Hestenes-Stiefel formula; restart condition; performance profile

1 Introduction

Consider the following form for the unconstrained optimization problem:

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

where $f: \mathbb{R}^n \to \mathbb{R}$ is a smooth nonlinear function. To solve (1) using the conjugate gradient (CG) method, we normally use the following iterative method:

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

where the starting point $x_1 \in \mathbb{R}^n$ is arbitrary and $\alpha_k > 0$ is the step length, which is computed via a line search. The search direction d_k is defined by

$$d_k = -g_k + \beta_k d_{k-1}, \quad k = 2, 3, \ldots, \tag{3}$$

where $g_k = \nabla f(x_k)$, β_k is a scalar, and the steepest descent method is used as an initial search direction, *i.e.*,

$$d_1 = -g_1, \quad k = 1. \tag{4}$$

The most well-known CG formulas are the following:

Hestenes-Stiefel (HS) [1],

$$\beta_k^{\mathrm{HS}} = \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})}, \tag{5}$$

Fletcher-Reeves (FR) [2],

$$\beta_k^{\mathrm{FR}} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \tag{6}$$

and Polak-Ribiere-Polyak (PRP) [3],

$$\beta_k^{\mathrm{PRP}} = \frac{g_k^T(g_k - g_{k-1})}{\|g_{k-1}\|^2}. \tag{7}$$

Theoretically, equations (5), (6), and (7) are similar if we use an exact line search, *i.e.*,

$$f(x_k + \alpha_k d_k) = \min f(x_k + \alpha d_k), \quad \alpha > 0, \tag{8}$$

and quadratic functions, *i.e.*,

$$f(x) = \frac{1}{2}x^T Qx - b^T x,$$

where Q is a positive definite matrix and b is a vector. However, in numerical computations and convergence analyses, the three main CG formulas are different if we use non-quadratic functions.

To obtain the step length, we have two types of line searches: exact line search as given by (8), which is an expensive line search in terms of calculating the function and gradient evolutions, and inexact line search, which approximates the step length by reducing the function value and direction derivative. The inexact line search is inexpensive and inherits identical advantages as the exact line search. The most popular inexact line search is the Wolfe-Powell line search [4, 5], which is designed to approximate the suitable step length using the following equations:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k \tag{9}$$

and

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k, \tag{10}$$

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

The strong version of the weak Wolfe-Powell (WWP) line search is the strong Wolfe-Powell (SWP) line search, which is given by (9) and

$$\left|g(x_k+\alpha_k d_k)^T d_k\right| \leq \sigma \left|g_k^T d_k\right|. \tag{11}$$

The difference between the WWP and SWP line searches is that the former no longer searches for the step length when the current iteration in (2) is far from the stationary point.

An important rule in the CG method is the descent property, which is given by

$$g_k^T d_k < 0. \tag{12}$$

If the CG formula inherits (12), then the iterative formula in (2) absolutely reduces the function value in every iteration. It is clear from (9) that $f(x_k+\alpha_k d_k)-f(x_k) \leq \delta\alpha_k g_k^T d_k$. If the direction derivative (*i.e.*, $g_k^T d_k$) is negative, we obtain

$$f(x_k+\alpha_k d_k)=f(x_{k+1})<f(x_k).$$

Thus, (12) must be satisfied before using the Wolfe-Powell line search. If we extend (12) to the form

$$g_k^T d_k \leq -c\|g_k\|^2, \quad k \geq 1 \text{ and } c>0, \tag{13}$$

then (13) is called the sufficient descent condition.

The HS CG formula is related to the conjugacy condition regardless of the objective function and line search, *i.e.*,

$$d_k^T g_k - d_k^T g_{k-1} = 0. \tag{14}$$

If the CG formula inherits (14), the efficiency will be better than other CG parameters that do not inherit this property. Dai and Liao [6] proposed the following novel conjugacy condition for an inexact line search:

$$d_k^T g_k - d_k^T g_{k-1} = -t\alpha_{k-1} g_k^T d_{k-1}, \quad t>0. \tag{15}$$

Using the criteria of an exact line search, $g_k^T d_{k-1}=0$, (15) reduces to the original conjugacy condition (14).

Because the PRP and HS formulas cannot satisfy the descent property when the SWP or WWP line searches are used, Gilbert and Nocedal [7] use Powell's [8] suggestion to solve the convergence problem of the PRP method as follows:

$$\beta_k^{\mathrm{PRP+}} = \max\left\{0, \beta_k^{\mathrm{PRP}}\right\},$$
$$\beta_k^{\mathrm{HS+}} = \max\left\{0, \beta_k^{\mathrm{HS}}\right\}.$$

However, $\beta_k^{\mathrm{PRP+}}$ cannot satisfy the descent property with the SWP line search. Thus, [7] uses the Moré and Thuente algorithm [9] if the descent property is not satisfied with the SWP line search.

Furthermore, [7] makes the following suggestion:

$$\beta_k = \begin{cases} -\beta_k^{\mathrm{FR}}, & \text{if } \beta_k^{\mathrm{PRP}} < -\beta_k^{\mathrm{FR}}, \\ \beta_k^{\mathrm{PRP}}, & \text{if } |\beta_k^{\mathrm{PRP}}| \le \beta_k^{\mathrm{FR}}, \\ \beta_k^{\mathrm{FR}}, & \text{if } \beta_k^{\mathrm{PRP}} > \beta_k^{\mathrm{FR}}. \end{cases}$$

Touati-Ahmed and Storey [10] suggest the following hybrid method:

$$\beta_k^{\mathrm{TS}} = \begin{cases} \beta_k^{\mathrm{PRP}}, & \text{if } 0 \le \beta_k^{\mathrm{PRP}} \le \beta_k^{\mathrm{FR}}, \\ \beta_k^{\mathrm{FR}}, & \text{otherwise.} \end{cases}$$

Several CG parameters that pertain to the PRP and HS formulas have been presented [11–13]. Here, we denote the following CG formulas using the WYL family:

$$\beta_k^{\mathrm{WYL}} = \frac{g_k^T(g_k - \frac{\|g_k\|}{\|g_{k-1}\|} g_{k-1})}{\|g_{k-1}\|^2}, \qquad \beta_k^{\mathrm{NPRP}} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g_k g_{k-1}|}{\|g_{k-1}\|^2},$$

$$\beta_k^{\mathrm{DPRP}} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} |g_k g_{k-1}|}{m|g_k^T d_{k-1}| + \|g_{k-1}\|^2}, \quad m \ge 0.$$

The CG formulas in the WYL family are clearly positive and satisfy the global convergence with descent properties. However, this family does not inherit the restart property. Thus, the convergence rate is linear. For more about the convergence rate, we refer the reader to [14].

Recently, Alhawarat *et al.* [15] constructed the following CG formula with the restart property as follows:

$$\beta_k^{\mathrm{HPRP}} = \begin{cases} \beta_k^{\mathrm{PRP}}, & \text{if } \|g_k\|^2 > |g_k^T g_{k-1}|, \\ \beta_k^{\mathrm{NPRP}}, & \text{otherwise.} \end{cases} \tag{16}$$

In addition, to learn about many versions of CG parameters related to classical CG methods and their convergence properties, we refer the reader to [16, 17].

2 Motivation and the new modification

The Hestenes-Stiefel (HS) CG formula is considered one of the most efficient methods developed in this century. In addition, the HS coefficient is related to the conjugacy condition regardless of whether the line search used. However, the HS parameter may not satisfy the global convergence properties of the CG method with the Wolfe-Powell line search if the descent condition is not satisfied. Thus, in this section, we present the following CG parameter:

$$\beta_k^{\mathrm{ZA}} = \begin{cases} \frac{\|g_k\|^2 - g_k^T g_{k-1}}{d_{k-1}^T g_k - d_{k-1}^T g_{k-1}}, & \text{if } \|g_k\|^2 > |g_k^T g_{k-1}|, \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

The non-zero term of (17) can clearly be written as follows:

$$\beta_k^{\mathrm{ZA}} \le \frac{\|g_k\|^2 + |g_k^T g_{k-1}|}{d_{k-1}^T g_k - d_{k-1}^T g_{k-1}} \le \frac{2\|g_k\|^2}{d_{k-1}^T g_k - d_{k-1}^T g_{k-1}} \tag{18}$$

and

$$\beta_k^{\mathrm{ZA}} > 0. \tag{19}$$

If $\beta_k^{\mathrm{ZA}} = 0$, the search direction becomes the steepest-descent method. In addition, if $g_k = 0$, the stationary point is found. Thus, in the following analysis, we always suppose that $\beta_k^{\mathrm{ZA}} > 0$ and $g_k \neq 0$ for all $k \geq 1$.

To prove that the number of restarts in (17) is not too many by using different non-standard initial points, we compare β_k^{ZA} with the following modified PRP CG parameter:

$$\beta_k^{\mathrm{PRP}*} = \begin{cases} \frac{g_k^T(g_k - g_{k-1})}{\|g_{k-1}\|^2}, & \text{if } \|g_k\|^2 > |g_k^T g_{k-1}|, \\ 0, & \text{otherwise}. \end{cases} \tag{20}$$

In the numerical results section, (20) restarted too many times by using non-standard different initial points. In other words, the CG parameter in (20) uses the steepest-descent method several times to reach the optimum solution. However, by using the standard initial points, PRP* becomes more efficient. Thus, in terms of the efficiency, (20) is not as efficient as (17) because the latter does not require as many times to restart.

The following algorithm describes the steps of using the CG method with (17) and SWP line search to obtain the solution for the optimization functions.

Algorithm 1

```
Step 0. Initialization. Given $x_1$, set $k = 1$.
Step 1. If $\|g_k\| \leq \varepsilon$, then stop, where $0 < \varepsilon \ll 1$.
Step 2. Compute $\beta_k$ based on (17).
Step 3. Compute $d_k$ based on (3) and (4).
Step 4. Compute $\alpha_k$ based on (9) and (11).
Step 5. Update a new point based on (2).
Step 6. Convergent test and stopping criteria: if $\|g_k\| \leq \varepsilon$, then stop; otherwise, go to Step 2
        with $k = k + 1$.
```

3 Global convergence properties for the β_k^{ZA} method

Because we are interested in determining the stationary point for the nonlinear optimization functions that are bounded below and whose gradient is Lipschitz continuous, the following standard assumption is necessary.

Assumption 1

I. The level set $\Psi = \{x | f(x) \leq f(x_1)\}$ is bounded, *i.e.*, there is a positive constant M such that

$$\|x\| \leq M, \quad \forall x \in \Psi.$$

II. In some neighborhood N of Ψ, f is continuously differentiable, and its gradient is Lipschitz continuous, *i.e.*, for all $x, y \in N$, there is a constant $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\|.$$

This assumption implies that there is a positive constant B such that

$$\|g(u)\| \leq B, \quad \forall u \in N.$$

The following lemma is known as the Zoutendijk condition [8], which is normally used to prove the convergence properties of CG formulas with the standard CG method. The global convergence indicates that a stationary point is obtained.

Lemma 3.1 *Suppose that Assumption* 1 *holds. Consider the CG methods of forms* (2) *and* (3), *where the search direction satisfies the sufficient descent condition and the step length, which is computed using the standard WWP line search. Then*

$$\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty.$$

The following two theorems demonstrate that (17) satisfies the descent condition with SWP and WWP line searches.

Theorem 3.1 *Let the sequences* $\{g_k\}$ *and* $\{d_k\}$ *be generated by methods* (2), (3) *and* (17) *with step length* α_k, *which is computed using the SWP line searches* (9) *and* (11) *with* $\sigma < \frac{1}{3}$; *then the sufficient descent condition* (13) *holds for some* $c \in (0, 1)$.

Proof Multiplying (3) by g_k^T, we have

$$g_k^T d_k = g_k^T(-g_k + \beta_k d_{k-1}) = -\|g_k\|^2 + \beta_k g_k^T d_{k-1}. \tag{21}$$

Then we have the following two cases:

Case 1. If $g_k^T d_{k-1} \leq 0$, then using (17), we obtain

$$g_k^T d_k = -\|g_k\|^2 + \beta_k^{\mathrm{ZA}} g_k^T d_{k-1} < 0.$$

Case 2. If $g_k^T d_{k-1} > 0$, then divide both sides of (21) by $\|g_k\|^2$ and using (17), we obtain

$$\frac{g_k^T d_k}{\|g_k\|^2} = -1 + \beta_k^{\mathrm{ZA}} g_k^T d_{k-1}.$$

Using (11) with $\sigma < 1/3$, we obtain

$$\frac{g_k^T d_k}{\|g_k\|^2} \leq -1 - \frac{2\sigma g_{k-1}^T d_{k-1}}{(\sigma - 1) g_{k-1}^T d_{k-1}} = -1 + \frac{2\sigma}{(1-\sigma)} < 0.$$

Let $c = 1 - \frac{2\sigma}{(1-\sigma)}$, we obtain

$$g_k^T d_k \leq -c\|g_k\|^2.$$

The proof is complete. □

Theorem 3.2 *Assume the sequences* $\{g_k\}$ *and* $\{d_k\}$ *are generated using the methods* (2), (3) *and* (17) *with step length* α_k, *which is computed via the WWP line search given by* (9) *and* (10); *then the descent condition* (13) *holds.*

Proof If $g_k^T d_{k-1} \leq 0$, then the proof is similar to Case 1 in Theorem 3.1. If $g_k^T d_{k-1} > 0$, from (3), we have

$$\begin{aligned}
g_k^T d_k &= g_k^T(-g_k + \beta_k d_{k-1}) = -\|g_k\|^2 + \beta_k g_k^T d_{k-1} \\
&\leq -\|g_k\|^2 + \frac{2\|g_k\|^2}{d_{k-1}^T g_k - d_{k-1}^T g_{k-1}} g_k^T d_{k-1} \\
&= \frac{1}{d_{k-1}^T g_k - d_{k-1}^T g_{k-1}} \bigl(-\|g_k\|^2 d_{k-1}^T g_k + \|g_k\|^2 d_{k-1}^T g_{k-1} + 2\|g_k\|^2 g_k^T d_{k-1}\bigr) \\
&= \frac{1}{d_{k-1}^T(g_k - g_{k-1})} \bigl(\|g_k\|^2 d_{k-1}^T g_{k-1} + \|g_k\|^2 g_k^T d_{k-1}\bigr) \\
&= \frac{1}{d_{k-1}^T(g_k - g_{k-1})} \bigl(\|g_k\|^2 d_{k-1}^T g_{k-1} + \|g_k\|^2 d_{k-1}^T g_{k-1} - \|g_k\|^2 d_{k-1}^T g_{k-1} + \|g_k\|^2 g_k^T d_{k-1}\bigr) \\
&= \frac{1}{d_{k-1}^T(g_k - g_{k-1})} \bigl(\|g_k\|^2 d_{k-1}^T(g_k - g_{k-1}) + 2\|g_k\|^2 d_{k-1}^T g_{k-1}\bigr) \\
&\leq \|g_k\|^2 + \frac{2\|g_k\|^2 d_{k-1}^T g_{k-1}}{(\sigma - 1) d_{k-1}^T g_{k-1}} \\
&\leq \|g_k\|^2 + \frac{2\|g_k\|^2}{(\sigma - 1)}.
\end{aligned}$$

Dividing both sides by $\|g_k\|^2$, we obtain

$$\frac{g_k^T d_k}{\|g_k\|^2} \leq 1 + \frac{2}{(\sigma - 1)}.$$

Let $c = -1 + \frac{2}{(1-\sigma)}$; then we obtain

$$g_k^T d_k \leq -c\|g_k\|^2.$$

The proof is complete. □

Gilbert and Nocedal [7] presented a useful property to prove the global convergence properties for the methods that pertain to the PRP (HS) formula. The property is as follows.

Property $*$ Consider a method of the form given by (2) and (3) and suppose that

$$0 < \gamma \leq \|g_k\| \leq \bar{\gamma}. \tag{22}$$

We say that the method has Property $*$ if there are constants $b > 1$ and $\lambda > 0$ such that for all $k \geq 1$, we obtain $|\beta_k| \leq b$; if $\|x_k - x_{k-1}\| \leq \lambda$, then

$$|\beta_k| \leq \frac{1}{2b}.$$

The following lemma is similar to that presented in [7].

Lemma 3.2 *Consider the CG method as defined in* (2), (3), *and* (17) *and the step length computed using the WWP line search. If the equation in* (22) *and Assumption* 1 *hold, then* β_k^{ZA} *satisfies Property* $*$.

Proof Assume that $b = \frac{2\bar{\gamma}^2}{(1-\sigma)c\gamma^2}$ and $\lambda = \frac{(1-\sigma)c\gamma^2}{2L\bar{\gamma}b}$. Then $b > 1$ and $\lambda > 0$. Using (10) and Theorem 3.2, we obtain

$$d_{k-1}^T(g_k - g_{k-1}) \geq (\sigma - 1)g_{k-1}^T d_{k-1} \geq c(1-\sigma)\|g_{k-1}\|^2.$$

Using (17) and Assumption 1, we obtain

$$\left|\beta_k^{\mathrm{ZA}}\right| = \left|\frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})}\right| \leq \frac{\|g_k\|^2 + |g_k^T g_{k-1}|}{c(1-\sigma)\|g_{k-1}\|^2} \leq \frac{2\bar{\gamma}^2}{c(1-\sigma)\gamma^2} = b.$$

If $\|x_k - x_{k-1}\| \leq \lambda$, then

$$\left|\beta_k^{\mathrm{ZA}}\right| = \left|\frac{\|g_k\|^2 - g_k^T g_{k-1}}{d_{k-1}^T(g_k - g_{k-1})}\right| \leq \frac{\|g_k\|\|g_k - g_{k-1}\|}{c(1-\sigma)\|g_{k-1}\|^2} \leq \frac{L\lambda\bar{\gamma}}{c(1-\sigma)\gamma^2} = \frac{1}{2b}.$$

The proof is complete. □

The proof of the forthcoming lemmas and Theorem 3.3 originally can be found in [7]. However, we present it here for readability. The following lemma shows that if the CG formula satisfies Property $*$, then the fraction of steps cannot be too small.

Lemma 3.3 *Assume that Assumption* 1 *holds. Assume that the sequences* $\{g_k\}$ *and* $\{d_k\}$ *are generated by Algorithm* 1, *where* α_k *is computed using the WWP line search, in which the sufficient descent condition* (13) *holds, and assume that the method has Property* $*$. *Suppose also* $\|g_k\| \geq \gamma$ *for some* $\lambda > 0$. *Then there exists* $\lambda > 0$ *such that for any* $\Delta \in \mathbb{N}$ *and any index* k_0, *there is an index* $k > k_0$ *that satisfies*

$$\left|\kappa_{k,\Delta}^{\lambda}\right| > \frac{\lambda}{2},$$

where $\kappa_{k,\Delta}^{\lambda} = \{i \in \mathbb{N} : k \leq i \leq k + \Delta - 1, \|s_i\| > \lambda\}$, $\mathbb{N}$ *denotes the set of positive integers, and* $|\kappa_{k,\Delta}^{\lambda}|$ *denotes the number of elements in* $\kappa_{k,\Delta}^{\lambda}$.

Lemma 3.4 *Suppose that Assumption* 1 *holds. Assume that the sequences* $\{g_k\}$ *and* $\{d_k\}$ *are generated by Algorithm* 1, *where* α_k *is computed using the WWP line search, and that the sufficient descent condition* (13) *holds. If* $\beta_k \geq 0$ *and* (22) *holds, then* $d_k \neq 0$ *and*

$$\sum_{k=0}^{\infty} \|u_{k+1} - u_k\|^2 < \infty, \quad \text{where } u_k = \frac{d_k}{\|d_k\|}.$$

Theorem 3.3 *Suppose that Assumption* 1 *holds. Assume that the sequences* $\{g_k\}$ *and* $\{d_k\}$ *are generated by Algorithm* 1, *where* α_k *is computed using the WWP line search, and that the sufficient descent condition* (13) *holds. In addition, suppose that Property* $*$ *holds. Then we have* $\lim_{k\to\infty} \inf \|g_k\| = 0$.

Proof Based on Lemma 3.4, we prove the theorem by contradiction. Define $u_i := \frac{d_i}{\|d_i\|}$. For any two indices l, k with $l \geq k$, we have

$$x_l - x_{k-1} = \sum_{i=k}^{l} \|s_{i-1}\| u_{i-1} = \sum_{i=k}^{l} \|s_{i-1}\| u_{k-1} + \sum_{i=k}^{l} \|s_{i-1}\| (u_{i-1} - u_{k-1}),$$

where $s_{i-1} = x_i - x_{i-1}$.

Taking the norms,

$$\sum_{i=k}^{l} \|s_{i-1}\| \leq \|x_l\| + \|x_{k-1}\| + \sum_{i=k}^{l} \|s_{i-1}\| \|u_{i-1} - u_{k-1}\|.$$

Using Assumption 1, we know that sequence $\{x_k\}$ is bounded, and there is a positive constant η such that $\|x_k\| \leq \eta$ for all $k \geq 1$. Thus,

$$\|x_l\| + \|x_{k-1}\| \leq 2\eta,$$

which implies that

$$\sum_{i=k}^{l} \|s_{i-1}\| \leq 2\eta + \sum_{i=k}^{l} \|s_{i-1}\| \|u_{i-1} - u_{k-1}\|. \tag{23}$$

Assume that $\lambda > 0$ is given by Lemma 3.3. Following the notation of this lemma, we define

$$\Delta := \left\lceil \frac{8\eta}{\lambda} \right\rceil.$$

From Lemma 3.3, we can find an index k_0 such that

$$\sum_{k \geq k_0}^{\infty} \|u_i - u_{i-1}\|^2 < \frac{1}{4\Delta}. \tag{24}$$

With this Δ and k_0, Lemma 3.4 gives an index $k \geq k_0$ such that

$$\left|\kappa_{k,\Delta}^{\lambda}\right| > \frac{\Delta}{2}. \tag{25}$$

Next, according to the Cauchy-Schwarz inequality and (24), we see that for any index $i \in [k, k + \Delta - 1]$,

$$\begin{aligned} \|u_{i-1} - u_{k-1}\| &\leq \sum_{j=k}^{i-1} \|u_j - u_{j-1}\| \\ &\leq (i-k)^{1/2} \left(\sum_{j=k}^{i-1} \|u_j - u_{j-1}\|^2 \right)^{1/2} \\ &\leq \Delta^{1/2} \left(\frac{1}{4\Delta} \right)^{1/2} = \frac{1}{2}. \end{aligned}$$

By this relation, (23) and (25), with $l = k + \Delta - 1$, we have

$$2\eta \geq \frac{1}{2}\sum_{i=k}^{k+\Delta-1} \|s_{i-1}\| > \frac{\lambda}{2}\left|\kappa_{k,\Delta}^{\lambda}\right| > \frac{\lambda\Delta}{4}.$$

Thus, $\Delta < 8\eta/\lambda$, which contradicts the definition of Δ. The proof is complete. □

Using Lemmas 3.2, 3.3, and 3.4 and Theorem 3.3, the global convergence of Algorithm 1 with the Wolfe-Powell line search is similarly established to that in Theorem 4.3 in [7]. Therefore, the proof of the following theorem is omitted, and we present the following theorems without proof.

Theorem 3.4 *Suppose that Assumption* 1 *holds. Consider the CG method of forms* (2), (3) *and* (11), *where* α_k *is computed using the WWP line search*; *then* $\lim_{k\to\infty} \inf \|g_k\| = 0$.

Theorem 3.5 *Suppose that Assumption* 1 *holds. Consider the CG method of forms* (2), (3) *and* (11), *where* α_k *is computed using the SWP line search with* $0 < \sigma < 1/3$; *then* $\lim_{k\to\infty} \inf \|g_k\| = 0$.

4 Numerical results and discussion

To test the efficiency and robustness of the new method (17), some standard test functions were selected from CUTE [18] and Andrei [19], as summarized in Table 1. We performed a comparison with other CG methods, which included the WYL family, PRP*, HPRP, PRP, HS+, FR, and ZA parameters. The stopping criterion was $\|g_k\| \leq 10^{-6}$ for all algorithms.

The initial point $x_1 \in \mathbb{R}^n$ is arbitrary. As shown in Table 1, we used different initial points based on the original standard points. We notice from the numerical results that different initial points almost had different stationary points for the multimodal functions. In addition, the efficiency of the algorithm depended on the initial points for every function. For example, the efficiency of the FR algorithm with the extended Rosenbrock function and the initial point $(-1.2, 1, -1.2, 1, \ldots, -1.2, 1)$ is different from that with $(5, 5, \ldots, 5)$ or $(10, 10, \ldots, 10)$ as the initial point. Moreover, the initial point determines the value of the CG formula based on Powell [20]; for example, the PRP or HS parameter fails to obtain the solution if its value is negative. In contrast, if we use another initial point, the value of PRP is non-negative and satisfies the descent property. This result motivated us to further study the initial points. Moreover, different dimensions were used for every function, and the dimension range was $[2, 10{,}000]$.

We present Himmelblau's function (Figure 1), which is a multimodal function to test the efficiency of the optimization algorithms. The function is defined as follows:

$$f(x, y) = \left(x^2 + y - 11\right)^2 + \left(x + y^2 - 7\right)^2.$$

In Table 2, we used different initial points with Algorithm 1 and Himmelblau's function. Every initial point gave a different solution point, as indicated in Table 2. We used a MATLAB 7.9 subroutine with an Intel (R) Core (TM) i5 CPU, 4 GB DDR2 RAM and the SWP line search under cubic interpolation. We used the Sigma Plot 10 program to graph the data based on multiple horizontal steps, and the graphs are shown in Figures 2 and 3. The selected values of δ and σ are 0.01 and 0.1, respectively.

Table 1 A list of test problem functions

No.	Function	Dimension/s	Initial points
1	Extended White & Holst function	2, 500, 1,000, 5,000, 10,000	$(-1.2,1,-1.2,1,\ldots,-1.2,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
2	Extended Rosenbrock function	2, 500, 1,000, 5,000, 10,000	$(-1.2,1,-1.2,1,\ldots,-1.2,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
3	Six hump function	2	$(1,1)$, $(5,5)$, $(10,10)$, $(15,15)$
4	Extended Beale function	2, 500, 1,000, 5,000, 10,000	$(-1,-1,\ldots,-1)$, $(0.5,0.5,\ldots,0.5)$, $(10,10,\ldots,10)$, $(1,1,\ldots,1)$
5	Three hump function	2	$(1,1)$, $(5,5)$, $(10,10)$, $(15,15)$
6	Extended Himmelblau function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
7	Diagonal 2 function	2, 500, 1,000, 5,000, 10,000	$(0.2,0.2,\ldots,0.2)$, $(0.25,0.25,\ldots,0.25)$, $(0.5,0.5,\ldots,0.5)$, $(1,1,\ldots,1)$
8	NONSCOMP function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(-1,-1,\ldots,-1)$, $(-2,-2,\ldots,-2)$, $(-5,-5,\ldots,-5)$
9	Extended DENSCHNB function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
10	Shallow function	2, 500, 1,000, 5,000, 10,000	$(-2,-2,\ldots,-2)$, $(2,2,\ldots,2)$, $(5,5\ldots,5)$, $(10,10,\ldots,10)$
11	Booth function	2	$(1,1)$, $(5,5)$, $(10,10)$, $(15,15)$
12	Extended quadratic penalty function QP2	2, 500, 1,000, 5,000, 10,000	$(2,2,\ldots,2)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
13	DIXMAANA function	300, 6,000, 9,000, 12,000	$(1,1,\ldots,1)$, $(2,2,\ldots,2)$, $(3,3,\ldots,3)$, $(5,5,\ldots,5)$,
14	DIXMAANB function	300, 6,000, 9,000, 12,000	$(1,1,\ldots,1)$, $(2,2,\ldots,2)$, $(3,3,\ldots,3)$, $(5,5,\ldots,5)$,
15	NONDIA function	2, 500, 1,000, 5,000, 10,000	$(-2,-2,\ldots,-2)$, $(-1,-1,\ldots,-1)$, $(0,0,\ldots,0)$, $(1,1,\ldots,1)$
16	Extended tridiagonal 1 function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
17	DQDRTIC function	2, 500, 1,000, 5,000, 10,000	$(-1,-1,\ldots,-1)$, $(1,1,\ldots,1)$, $(2,2,\ldots,2)$, $(3,3,\ldots,3)$
18	Diagonal 4 function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
19	Extended Cliff function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
20	Shallow function	2, 500, 1,000, 5,000, 10,000	$(-2,-2,\ldots,-2)$, $(2,2,\ldots,2)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$
21	NONDIA (Shanno-78)	2, 500, 1,000, 5,000, 10,000	$(-2,-2,\ldots,-2)$, $(-1,-1,\ldots,-1)$, $(0,0,\ldots,0)$, $(1,1,\ldots,1)$
22	Raydan 2 Function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
23	Extended block diagonal BD1 function	2, 500, 1,000, 5,000, 10,000	$(0.1,0.1,\ldots,0.1)$, $(0.2,0.2,\ldots,0.2)$, $(0.3,0.3,\ldots,0.3)$, $(0.4,0.4,\ldots,0.4)$
24	Generalized tridiagonal 1 function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
25	Diagonal 4 function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
26	Extended Powell function	2, 500, 1,000, 5,000, 10,000	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$
27	Perturbed quadratic function	2, 500, 1,000, 5,000, 10,000	$(0.5,0.5,\ldots,0.5)$, $(2,2,\ldots,2)$, $(1,1,\ldots,1)$, $(10,10,\ldots,10)$
28	A quadratic function QF2	2, 500, 1,000, 5,000, 10,000	$(0.5,0.5,\ldots,0.5)$, $(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$
29	Sum squares function	2, 500, 1,000, 5,000, 10,000	$(-5,-5,\ldots,-5)$, $(-1,-1,\ldots,-1)$, $(1,1,\ldots,1)$, $(5,5,\ldots,5)$
30	Zettl function	2	$(1,1,\ldots,1)$, $(5,5,\ldots,5)$, $(10,10,\ldots,10)$, $(15,15,\ldots,15)$

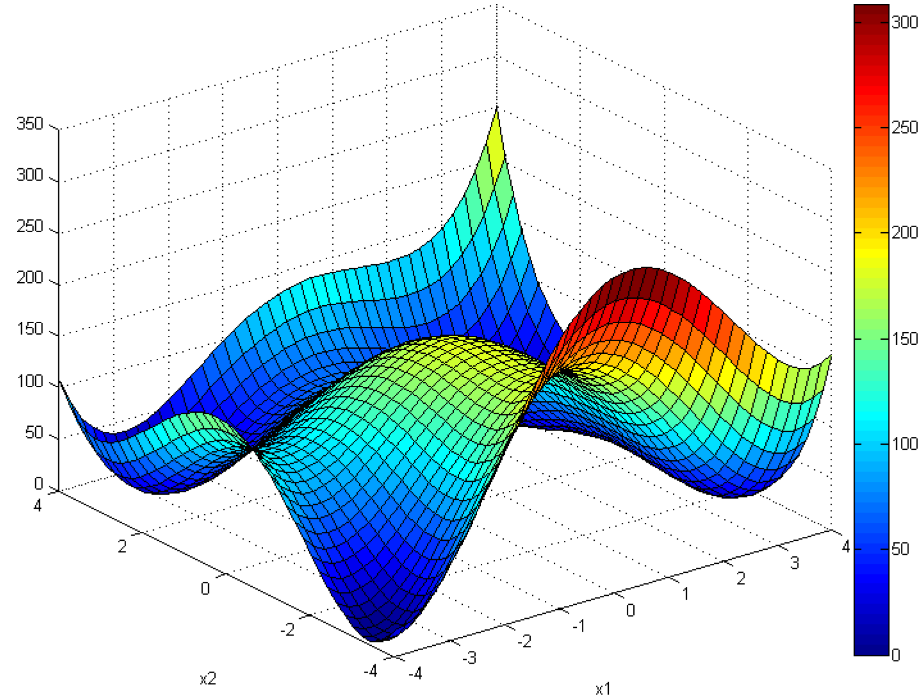

Figure 1 **Himmelblau's function.**

Table 2 **The initial points corresponding the optimal points with the Himmelblau function**

Initial point	The optimal solution	The function value
$(1, 1)$	$(3, 2)$	$f(3, 2) = 0$
$(-1, -1)$	$(3.5844, -1.8481)$	$f(3.5844, -1.8481) = 0$
$(10, 10)$	$(-3.7793, -3.2832)$	$f(-3.7793, -3.2832) = 0$
$(-5, -5)$	$(-2.8051, 3.1313)$	$f(-2.8051, 3.1313) = 0$

The performance results are shown in Figures 2 and 3 with a performance profile introduced by Dolan and Moré [21].

This performance measure was introduced to compare a set of solvers S on a set of problems F. Assuming that there are n_s solvers and n_f problems in S and F, respectively. Then the measure $t_{f,s}$ is defined as the required number of iterations or CPU time to solve problem f using solver s. To create a baseline for comparison, the performance of solver s on problem f is scaled by the best performance of any solver in S on the problem using the ratio

$$r_{f,s} = \frac{t_{f,s}}{\min\{t_{f,s} : s \in S\}}.$$

Suppose that a parameter $r_M \geq r_{f,s}$ for all f, s is selected. $r_{f,s} = r_M$ if and only if solver s does not solve problem f.

Because we would like to obtain an overall assessment of the performance of a solver, we defined the measure

$$P_s(t) = \frac{1}{n_f} \operatorname{size}\{f \in F : \log r_{f,s} \leq t\}.$$

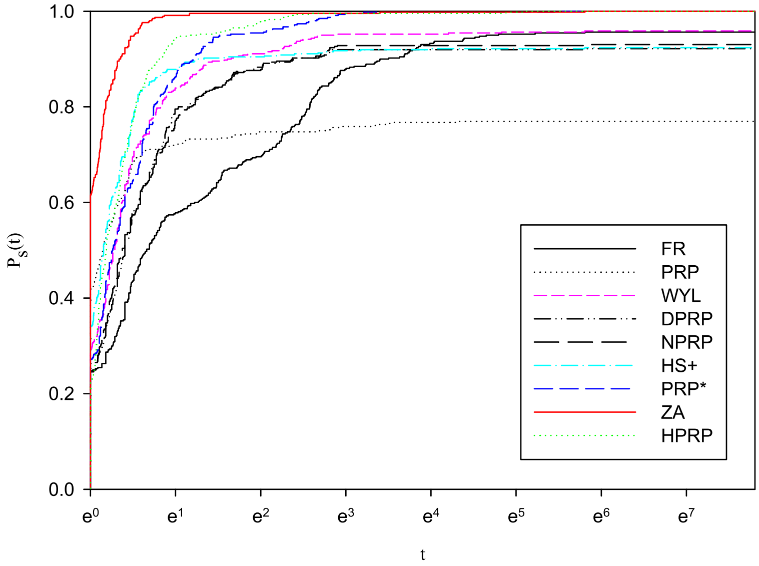

Figure 2 Performance profile based on the number of iteration.

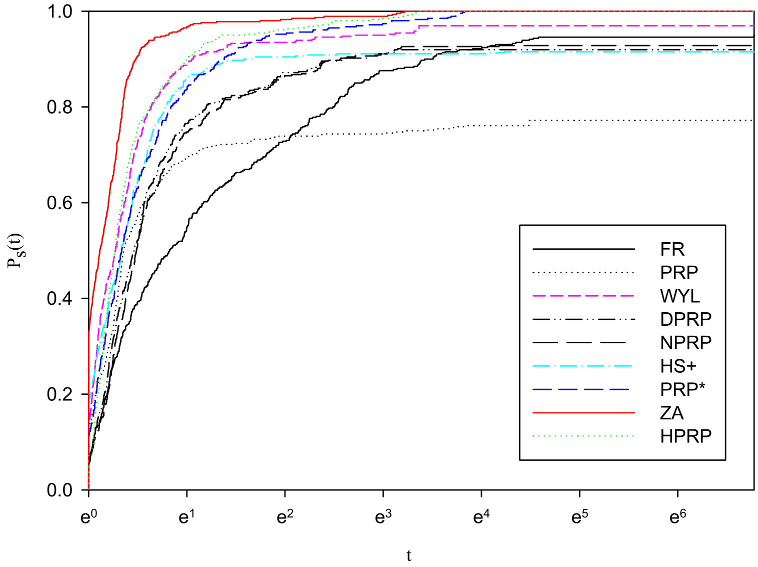

Figure 3 Performance profile based on the CPU time.

Thus, $P_s(t)$ is the probability for solver $s \in S$ that the performance ratio $r_{f,s}$ is within a factor $t \in \mathbb{R}$ of the best possible ratio. If we define function p_s as the cumulative distribution function for the performance ratio, then the performance measure $f_s : \mathbb{R} \to [0,1]$ for a solver is non-decreasing and piecewise continuous from the right. The value of $f_s(1)$ is the probability that the solver has the best performance of all solvers. In general, a solver with high values of $f(t)$, which appears in the upper right corner of the figure, is preferable.

Based on the left side of Figures 2 and 3, ZA is clearly above the other curves. As we previously mentioned, PRP* seems to be better than HPRP because the latter restarted too many times by using the negative gradient. Furthermore, WYL is better than NPRP, DPRP.

Although the PRP and HS methods are efficient, both of them have theoretical problems; thus, the number of solved function using the PRP formula does not exceed 75%. The HS+ formula also has theoretical problem when the direction derivative is positive; hence, it may not satisfy the descent property with the SWP line search. Thus, the percentage value of solved functions using the HS+ formula is approximately 90%. The FR formula satisfies the descent property and convergence property, but we terminated the program several times because it is cyclic without reaching the solution. For all algorithms, the time limit to obtain the solution was 500 seconds.

5 Conclusion

In this paper, we used the HS CG formula with the restart. The global convergence and descent properties were established with WWP and SWP line searches. The numerical results demonstrate that the new modification is better than other CG parameters.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors contributed equally to the writing of this paper. All authors read and approved the final version of this paper.

Acknowledgements

The authors are grateful to the editor and the anonymous reviewers for their valuable comments and suggestions, which have substantially improved this paper. In addition, we acknowledge the Ministry of Higher Education Malaysia and Universiti Malaysia Terengganu; this study was partially supported under the Fundamental Research Grant Scheme (FRGS) Vote no. 59347.

Received: 8 November 2015 Accepted: 18 March 2016 Published online: 06 April 2016

References

1. Hestenes, MR, Stiefel, E: Methods of conjugate gradients for solving linear systems. J. Res. Natl. Bur. Stand. **49**(6), 409-436 (1952)
2. Fletcher, R, Reeves, CM: Function minimization by conjugate gradients. Comput. J. **7**(2), 149-154 (1964)
3. Polak, E, Ribiere, G: Note sur la convergence de méthodes de directions conjuguées. ESAIM: Math. Model. Numer. Anal. **3**(R1), 35-43 (1969)
4. Wolfe, P: Convergence conditions for ascent methods. SIAM Rev. **11**, 226-235 (1968)
5. Wolfe, P: Convergence conditions for ascent methods. II: some corrections. SIAM Rev. **13**, 185-188 (1971)
6. Dai, Y-H, Liao, L-Z: New conjugacy conditions and related nonlinear conjugate gradient methods. Appl. Math. Optim. **43**(1), 87-101 (2001)
7. Gilbert, JC, Nocedal, J: Global convergence properties of conjugate gradient methods for optimization. SIAM J. Optim. **2**(1), 21-42 (1992)
8. Zoutendijk, G: Nonlinear programming, computational methods. Integer Nonlinear Program. **143**(1), 37-86 (1970)
9. Moré, JJ, Thuente, DJ: On line search algorithms with guaranteed sufficient decrease. Mathematics and Computer Science Division Preprint MCS-P153-0590, Argonne National Laboratory, Argonne, IL (1990)
10. Touati-Ahmed, D, Storey, C: Efficient hybrid conjugate gradient techniques. J. Optim. Theory Appl. **64**, 379-397 (1990)
11. Wei, Z, Yao, S, Liu, L: The convergence properties of some new conjugate gradient methods. Appl. Math. Comput. **183**(2), 1341-1350 (2006)
12. Zhang, L: An improved Wei-Yao-Liu nonlinear conjugate gradient method for optimization computation. Appl. Math. Comput. **215**(6), 2269-2274 (2009)
13. Dai, Z, Wen, F: Another improved Wei-Yao-Liu nonlinear conjugate gradient method with sufficient descent property. Appl. Math. Comput. **218**(14), 7421-7430 (2012)
14. Sun, W, Yuan, YX: Optimization Theory and Methods: Nonlinear Programming, vol. 1. Springer, Berlin (2006)
15. Alhawarat, A, Mamat, M, Rivaie, M, Salleh, Z: An efficient hybrid conjugate gradient method with the strong Wolfe-Powell line search. Math. Probl. Eng. **2015**, Article ID 103517 (2015)
16. Yuan, G, Meng, Z, Li, Y: A modified Hestenes and Stiefel conjugate gradient algorithm for large-scale nonsmooth minimizations and nonlinear equations. J. Optim. Theory Appl. **168**(1), 129-152 (2016)
17. Alhawarat, A, Mamat, M, Rivaie, M, Mohd, I: A new modification of nonlinear conjugate gradient coefficients with global convergence properties. Int. J. Math. Comput. Stat. Nat. Phys. Eng. **8**(1), 54-60 (2014)
18. Bongartz, I, Conn, AR, Gould, N, Toint, PL: CUTE: constrained and unconstrained testing environment. ACM Trans. Math. Softw. **21**(1), 123-160 (1995)
19. Andrei, N: An unconstrained optimization test functions collection. Adv. Model. Optim. **10**(1), 147-161 (2008)
20. Powell, MJD: Nonconvex Minimization Calculations and the Conjugate Gradient Method. Springer, Berlin (1984)
21. Dolan, ED, Moré, JJ: Benchmarking optimization software with performance profiles. Math. Program. **91**(2), 201-213 (2002)