

Uzbek Mathematical Journal
 2026, Volume 70, Issue 3, pp. 135–145
 DOI: 10.29229/uzmj.2026-3-14

A convex combination conjugate gradient algorithm with application in mode function

Mehamdia A. E., Chaib Y., Bouaziz T., Bechouat T.

Abstract. Conjugate gradient methods are a popular class of iterative methods for solving linear systems of equations and nonlinear optimization problems. In this paper, we suggest a new hybrid nonlinear conjugate gradient method, in which the conjugate gradient coefficient β_k is a convex combination of $\beta_k^{NVHS^*}$ and β_k^{DY} . With the strong Wolfe line search, the descent property and global convergence of the new hybrid method are proved. The numerical results also show that our method is robust and efficient. Furthermore, the proposed algorithm was extended to solve the problem of mode function.

Keywords: Hybrid conjugate gradient method, Line search, Sufficient descent condition, Global convergence, Numerical comparisons, Mode function, Kernel estimator.

MSC (2020): 90C30, 65K05, 62G05

1. INTRODUCTION

The optimization model is a needful mathematical problem since it has been connected to different fields such as economics, engineering and physics. In recent works, we have developed several efficient conjugate gradient methods with successful applications in image restoration, nonparametric estimation, and regression analysis [1, 2, 3, 4]. Today there are many optimization algorithms, such as Newton, quasi-Newton and bundle algorithms. Note that these algorithms fail to solve large-scale optimization problems because they need to store and calculate relevant matrices. In contrast, the conjugate gradient algorithm is successful due to its simplicity of iteration and low memory requirements. In this work, we study the nonlinear conjugate gradient method for the following unconstrained optimization problem:

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

where f is a smooth and nonlinear function. The conjugate gradient method generates a sequence $\{x_k\}_{k \geq 0}$ such that:

$$x_{k+1} = x_k + \alpha_k d_k, \quad (1.2)$$

where x_k is the current iteration point and $d_k \in \mathbb{R}^n$ is the search direction defined by the following formula:

$$d_{k+1} = -g_{k+1} + \beta_k d_k; \quad d_0 = -g_0, \quad (1.3)$$

where g_{k+1} the gradient of f at x_{k+1} and the parameter β_k is known as the conjugate gradient coefficient. The step length α_k is very important for the global convergence of conjugate gradient methods, one often requires the line search to satisfy the Wolfe conditions (WLS)

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k, \quad (1.4)$$

and

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k. \quad (1.5)$$

Also, the strong Wolfe (SWLS) conditions consist of (1.4) and

$$|g_{k+1}^T d_k| \leq -\sigma g_k^T d_k, \quad (1.6)$$

TABLE 1. List of famous conjugate gradient coefficient formulas

Formula	Reference	Property
$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{y_k^T d_k}$	Hestenes-Stiefel [5]	Good numerical results
$\beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\ g_k\ ^2}$	Polak-Ribière-Polyak [6, 7]	
$\beta_k^{LS} = \frac{g_{k+1}^T y_k}{-g_k^T d_k}$	Liu-Storey [8]	
$\beta_k^{FR} = \frac{\ g_{k+1}\ ^2}{\ g_k\ ^2}$	Fletcher-Reeves [9]	Strong convergence
$\beta_k^{CD} = \frac{\ g_{k+1}\ ^2}{-g_k^T d_k}$	Conjugate Descent [10]	
$\beta_k^{DY} = \frac{\ g_{k+1}\ ^2}{y_k^T d_k}$	Dai-Yuan [11]	

where $0 < \delta < \sigma < 1$. Now, we denote $y_k = g_{k+1} - g_k$, $\|\cdot\|$ the Euclidean norm. In Table 1, we will give some famous formulas of the parameter β_k :

Dai and Yuan [11] proved that the DY method always generate descent directions and converges globally with the standard Wolfe line conditions (1.4) and (1.5). On the other hand, the HS method is generally regarded to be one of the most efficient conjugate gradient methods, but their convergence property is not so good. In the latest years, based on the above six formulas and their hybridization, many works have devoted their time and effort to come up with new formulae in order to increase the efficiency and effectiveness of the HS method. Yao et al. [12] presented a nonlinear conjugate gradient algorithm, denoted by VHS, where the parameter β_k given by:

$$\beta_k^{VHS} = \frac{g_{k+1}^T \left(g_{k+1} - \frac{\|g_{k+1}\|}{\|g_k\|} g_k \right)}{y_k^T d_k}.$$

Yao et al. [12] proved that the VHS method also can produce sufficient descent direction and this method is globally convergent if strong Wolfe conditions (1.4) and (1.6) are used with $\sigma < \frac{1}{3}$. Zhang [13] gave the modified VHS method which we call NHS method. The β_k in NHS method is given by

$$\beta_k^{NHS} = \frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|}{\|g_k\|} |g_{k+1}^T g_k|}{d_k^T y_k}.$$

Zhang [13] proved that the NHS method satisfies the sufficient descent condition and is globally convergent under the strong Wolfe conditions (1.4) and (1.6) with $\sigma < \frac{1}{2}$. Du et al. [14] gave a new conjugate gradient method, called the NVHS* method. The β_k in the NVHS* method is defined as

$$\beta_k^{NVHS^*} = \frac{g_{k+1}^T \left(g_{k+1} - \frac{|g_{k+1}^T g_k|}{\|g_k\|^2} g_k \right)}{d_k^T y_k}.$$

The NVHS* method has sufficient descent conditions and is globally convergent if the strong Wolfe line search is used and the parameter $\sigma < \frac{1}{3}$. The hybrid conjugate gradient method considered is to combine the standard conjugate gradient methods in a two distinct ways. The first class is based on projection concept. Recently, Touati-Ahmed and Storey [15] introduced one of the first hybrid conjugate gradient algorithms, where the parameter β_k is computed as:

$$\beta_k^{TaS} = \min \{ \beta_k^{FR}, \beta_k^{PRP} \}.$$

The authors proved that β_k^{TaS} has good convergence properties and numerically outperforms both the β_k^{FR} and β_k^{PRP} algorithms. Soon afterward, Hu and Storey [16], Gilbert and Nocedal [17] further studied other hybrid schemes about PRP and FR methods. Dai and Yuan [18] combined DY method with HS method, such that the CG coefficient is computed by:

$$\beta_k^{hDYz} = \max \{0, \min \{\beta_k^{HS}, \beta_k^{DY}\}\}.$$

Also, this author [18] has given a another modified conjugate gradient method, in which β_k is defined by:

$$\beta_k^{hDY} = \max \{-c\beta_k^{DY}, \min \{\beta_k^{HS}, \beta_k^{DY}\}\},$$

where $c = \frac{1-\sigma}{1+\sigma}$. For the standard Wolfe conditions (1.4) and (1.5), under the Lipschitz continuity of the gradient, Dai and Yuan [19] established the global convergence of these hybrid computational schemes. The second class of hybrid conjugate gradient methods is based on the convex combination of the standard methods. Recently, Andrei [20] introduced a new hybrid conjugate gradient method based on HS and DY method (denoted as HYBRID method) for solving unconstrained optimization problems (1.1), calculating the parameter β_k^c as a convex combination of β_k^{HS} and β_k^{DY} i.e.

$$\beta_k^c = (1 - \theta_k) \beta_k^{HS} + \theta_k \beta_k^{DY},$$

where θ_k is a scalar parameter satisfying $0 \leq \theta_k \leq 1$. Convergence with the standard Wolfe condition was established and numerical results show that this hybrid computational scheme outperforms the Hestenes-Stiefel and the Dai-Yuan conjugate gradient algorithms. This author also studied the global convergence of the CCOMB method [21] under strong Wolfe line search, such that the parameter β_k as a convex combination of β_k^{PRP} and β_k^{DY} i.e.

$$\beta_k^N = (1 - \theta_k) \beta_k^{PRP} + \theta_k \beta_k^{DY}.$$

Liu and Li [2] proposed another hybrid conjugate gradient as a convex combination of LS and DY method, (denoted as HLSDY method), where β_k in this method given by:

$$\beta_k^{HLSDY} = (1 - \theta_k) \beta_k^{LS} + \theta_k \beta_k^{DY}.$$

The global convergence was established under a strong Wolfe line search. Numerical results show that the method is efficient for the standard unconstrained problems in a CUTE library [22].

The most important and new thing in this work is the application of our method in nonparametric statistics, where we are the first to use and access in this field.

This paper aims to propose new hybrid conjugate gradient as a convex combination of NVHS* and DY conjugate gradient algorithms. We establish, under a strong Wolfe line search, convergence properties of the proposed conjugate gradient method. Numerical results show that the hHD method is efficient and robust and outperforms as six conjugate gradient methods famous. Finally, an application of our method in nonparametric mode estimator is also considered.

Now, we will organize our work as follows. In the next section, we consist of a new hybrid method and determine the parameter θ_k . Also, we present the specific algorithm and we prove the sufficient descent condition. In section 3, we prove the global convergence of the proposed method with a strong Wolfe line search. The numerical results are contained in section 4. In section 5, we focus an application of the hHD method in statistics nonparametric. Finally, we make a summary of our paper.

2. CONVEX COMBINATION METHOD

The parameter β_k in the presented method, denoted as β_k^{hHD} , is computed as a convex combination of $\beta_k^{NVHS^*}$ and β_k^{DY} , i.e:

$$\beta_k^{hHD} = (1 - \theta_k) \beta_k^{NVHS^*} + \theta_k \beta_k^{DY}, \quad (2.1)$$

where θ_k is a scalar parameter satisfying $0 \leq \theta_k \leq 1$, which follows to be determined. It is obvious that if $\theta_k = 0$, then $\beta_k^{hHD} = \beta_k^{NVHS^*}$ and if $\theta_k = 1$, then $\beta_k^{hHD} = \beta_k^{DY}$. On the other hand, if $0 < \theta_k < 1$, then β_k^{hHD} is a convex combination of $\beta_k^{NVHS^*}$ and β_k^{DY} parameters. The search direction d_k of our algorithm is computed by

$$d_0 = -g_0, \quad d_{k+1} = -g_{k+1} + \beta_k^{hHD} d_k. \quad (2.2)$$

2.1. The conjugate condition. The traditional conjugacy condition $d_{k+1}^T y_k = 0$, plays an important role in the convergence analyses and numerical calculation. The select the parameter θ_k , we consider the following Lemma.

Lemma 2.1. If the conjugacy condition $d_{k+1}^T y_k = 0$, is satisfied at every iteration, we get

$$\theta_k = 1 - \frac{\|g_k\|^2}{|g_{k+1}^T g_k|}. \quad (2.3)$$

Proof. We multiply the two sides of the relation (2.2) by y_k^T , we obtain

$$-y_k^T g_{k+1} + \left((1 - \theta_k) \beta_k^{NVHS^*} + \theta_k \beta_k^{DY} \right) y_k^T d_k = 0.$$

After simplification, we find

$$\theta_k = \frac{y_k^T g_{k+1} - \beta_k^{NVHS^*} y_k^T d_k}{(\beta_k^{DY} - \beta_k^{NVHS^*}) y_k^T d_k} = \frac{y_k^T g_{k+1} - \|g_{k+1}\|^2 + \frac{|g_{k+1}^T g_k|}{\|g_k\|^2} g_{k+1}^T g_k}{\frac{|g_{k+1}^T g_k|}{\|g_k\|^2} g_{k+1}^T g_k} = \frac{-g_{k+1}^T g_k + \frac{|g_{k+1}^T g_k|}{\|g_k\|^2} g_{k+1}^T g_k}{\frac{|g_{k+1}^T g_k|}{\|g_k\|^2} g_{k+1}^T g_k}.$$

Then, we get

$$\theta_k = \frac{\|g_k\|^2}{|g_{k+1}^T g_k|} - 1.$$

The result can be achieved. $\square$

2.2. hHD Algorithm. The detailed framework of the proposed algorithm is presented in Algorithm 1

Algorithm 1 hHD Conjugate Gradient Algorithm

Require: Initial point $x_0 \in \mathbb{R}^n$, parameters $\delta = 10^{-3}$, $\sigma = 10^{-1}$, $\varepsilon = 10^{-6}$, $max_iter = 2000$, $max_time = 500$ s

Ensure: Approximate minimizer x_k

- 1: Compute $f_0 = f(x_0)$, $g_0 = \nabla f(x_0)$
 - 2: Set $d_0 = -g_0$, $k = 0$
 - 3: **while** $\|g_k\|_\infty > \varepsilon$ and $k < max_iter$ and elapsed time $< max_time$ **do**
 - 4: Compute α_k satisfying strong Wolfe conditions (1.4) and (1.6) using Hager-Zhang line search
 - 5: $x_{k+1} = x_k + \alpha_k d_k$
 - 6: Compute $g_{k+1} = \nabla f(x_{k+1})$
 - 7: $y_k = g_{k+1} - g_k$
 - 8: **if** $|g_{k+1}^T g_k| < 10^{-12}$ **then**
 - 9: $\theta_k = 0$
 - 10: **else**
 - 11: $\theta_k = \frac{\|g_k\|^2}{|g_{k+1}^T g_k|} - 1$
 - 12: $\theta_k = \max(0, \min(1, \theta_k))$ $\triangleright$ Clamp to $[0, 1]$
 - 13: **end if**
 - 14: Compute $\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k}$
 - 15: Compute $\beta_k^{NVHS^*} = \frac{g_{k+1}^T \left(g_{k+1} - \frac{|g_{k+1}^T g_k|}{\|g_k\|^2} g_k \right)}{d_k^T y_k}$
 - 16: $\beta_k^{hHD} = (1 - \theta_k) \beta_k^{NVHS^*} + \theta_k \beta_k^{DY}$
 - 17: $d_{k+1} = -g_{k+1} + \beta_k^{hHD} d_k$
 - 18: $k = k + 1$
 - 19: **end while**
 - 20: **return** x_k
-

2.3. Sufficient descent condition. Now, we prove that the search direction d_k obtained by hHD method satisfies the sufficient descent condition.

Theorem 2.1. Let the sequence $\{g_k\}_{k \geq 0}$ and $\{d_k\}_{k \geq 0}$ be generated by hHD algorithm, then for positive constant c , we have

$$g_k^T d_k \leq -c \|g_k\|^2, \forall k \geq 0. \quad (2.4)$$

Proof. The following proof is by induction. For $k = 0$, $g_0^T d_0 = -\|g_0\|^2$, we conclude that the sufficient descent condition holds for $k = 0$. Now, we assume (2.4) holds for k and prove that for $k + 1$. From (2.1) and (2.2), we get

$$d_{k+1} = -g_{k+1} + \left((1 - \theta_k) \beta_k^{NVHS*} + \theta_k \beta_k^{DY} \right) d_k.$$

After a simple calculation, we have

$$d_{k+1} = \theta_k d_{k+1}^{DY} + (1 - \theta_k) d_{k+1}^{NVHS*}. \quad (2.5)$$

Firstly, let $\theta_k = 0$, then

$$d_{k+1} = d_{k+1}^{NVHS*} = -g_{k+1} + \beta_k^{NVHS*} d_k. \quad (2.6)$$

Multiplying (2.6) by g_{k+1}^T from the left, we get

$$g_{k+1}^T d_{k+1}^{NVHS*} \leq -\|g_{k+1}\|^2 + \beta_k^{NVHS*} |g_{k+1}^T d_k|. \quad (2.7)$$

Thus from (1.6), (2.4), we have

$$d_k^T y_k \geq (1 - \sigma) (-g_k^T d_k) \geq 0. \quad (2.8)$$

We have from the definition of β_k^{NVHS*} and the Cauchy Schwarz inequality, that

$$0 \leq \beta_k^{NVHS*} \leq 2 \frac{\|g_{k+1}\|^2}{d_k^T y_k}. \quad (2.9)$$

Using (1.6), (2.7), (2.8) and (2.9), we concluded

$$g_{k+1}^T d_{k+1}^{NVHS*} \leq -k_1 \|g_{k+1}\|^2, \quad (2.10)$$

where $k_1 = \frac{1-3\sigma}{1-\sigma}$, so there is a constant $k_1 > 0$, with $0 < \sigma < \frac{1}{3}$.

Secondly, let $\theta_k = 1$, then

$$d_{k+1} = d_{k+1}^{DY} = -g_{k+1} + \beta_k^{DY} d_k. \quad (2.11)$$

Multiplying (2.11) by g_{k+1}^T from the left, we get

$$g_{k+1}^T d_{k+1}^{DY} \leq -\|g_{k+1}\|^2 + \|g_{k+1}\|^2 \frac{|g_{k+1}^T d_k|}{y_k^T d_k}. \quad (2.12)$$

Using the line search condition (1.6), (2.8) and (2.12), we obtain

$$g_{k+1}^T d_{k+1}^{DY} \leq -k_2 \|g_{k+1}\|^2, \quad (2.13)$$

where $k_2 = \frac{1-2\sigma}{1-\sigma}$. Since $\theta_k \in (0, 1)$ and the algorithm does not stagnate, we can define $a_1 = \inf_k \theta_k > 0$ and $a_2 = \sup_k \theta_k < 1$. Finally, we assume that $0 < \theta_k < 1$, i.e. there exists two positive constants a_1 and a_2 positives such that: $0 < a_1 \leq \theta_k \leq a_2 < 1$. From the relations (2.5), (2.10) and (2.13), we get

$$g_{k+1}^T d_{k+1} \leq -c \|g_{k+1}\|^2,$$

where $c = a_1 k_1 + (1 - a_2) k_2$. Hence, Theorem 2.1 is proved. $\square$

3. GLOBAL CONVERGENCE

- To establish the global convergence of our method, we need the following basic assumptions on the objective function.

Assumption 3.1. The level set

$$S = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\},$$

is bounded.

Assumption 3.2. In some open convex neighborhood $\mathcal{N}$ of S , the function f is continuously differentiable and its gradient is Lipschitz continuous, namely, there exists a constant $L > 0$, such that:

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\| \quad \forall x, y \in \mathcal{N}. \quad (3.1)$$

From Assumption 3.2, we can deduce that for all $x \in \mathcal{N}$, there exists a positive constant $\Gamma \geq 0$, such that

$$\|\nabla f(x)\| \leq \Gamma, \text{ for all } x \in \mathcal{N}. \quad (3.2)$$

- It follows from Dai et al. [23] proved the sufficient condition for the convergence of conjugate gradient methods with strong Wolfe line search.

Lemma 3.1. Let Assumptions 3.1 and 3.2 hold. Consider the method (1.2) and (1.3), where d_k is a descent direction, and α_k is obtained by the strong Wolfe line search. If

$$\sum_{k \geq 0} \frac{1}{\|d_k\|^2} = \infty, \quad (3.3)$$

then

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0.$$

Now, we need also this lemma to prove the convergence of our method.

Lemma 3.2. Let Assumptions 3.1 and 3.2 hold. If d_k is a descent direction and α_k satisfies the WLS (1.4) and (1.5). Then

$$\alpha_k \geq \frac{(1 - \sigma) |g_k^T d_k|}{L \|d_k\|^2}. \quad (3.4)$$

Proof. See the proof of Lemma 3.2 in Liu and Li [2]. $\square$

Remark 3.1. From (1.6), (2.4), the step size α_k obtained in the hHD algorithm satisfies (3.3). This indicates, the step size α_k obtained in our method is not equal to zero, i.e., there exists a constant $\lambda > 0$, such that

$$\alpha_k \geq \lambda, \forall k \geq 0. \quad (3.5)$$

Theorem 3.1. Suppose that Assumptions 3.1 and 3.2 hold. Consider any conjugate gradient method in the form (1.2) and (1.3), with the parameter $\beta_k = \beta_k^{hHD}$, in which the step length α_k is determined to satisfy the strong Wolfe line search condition (1.4) and (1.6), where d_k is a descent search direction. Then, this method converges in the sense that

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0. \quad (3.6)$$

Proof. To prove Theorem 3.1, we use contradiction. That is, if equation (3.6) is not true. Then we can find a positive constant γ , such that

$$\|g_k\| \geq \gamma, \quad \text{for all } k. \quad (3.7)$$

From (2.1) and (2.9), we have

$$\beta_k^{hHD} \leq \beta_k^{DY} + \beta_k^{NVHS^*} \leq 3\beta_k^{DY}. \quad (3.8)$$

By using (2.8), (2.10) and (3.7), we obtain

$$y_k^T d_k \geq k_2(1 - \sigma)\gamma^2. \quad (3.9)$$

From (3.2), (3.8) and (3.9), we have

$$\beta_k^{hHD} \leq \frac{3\Gamma^2}{k_2(1 - \sigma)\gamma^2} = E. \quad (3.10)$$

Thus, it follows from (2.2), (3.2), (3.5) and (3.10), that

$$\|d_{k+1}\| \leq \|g_{k+1}\| + \beta_k^{hHD} \frac{\|x_{k+1} - x_k\|}{\alpha_k} \leq M, \quad (3.11)$$

where

$$M = \Gamma + E \frac{D}{\lambda}, \quad \text{and} \quad D = \max \{\|y - z\| : y, z \in \mathcal{N}\}.$$

By taking the summation $k \geq 0$,

$$\sum_{k \geq 0} \frac{1}{\|d_k\|^2} = \infty.$$

So, applying Lemma 3.1, we conclude that (3.6) is true. This is a contradiction with (3.7), so we have proved (3.6). $\square$

4. NUMERICAL EXPERIMENTS

In this section, we present some numerical experiments obtained with the new proposed conjugate gradient method. The test problems have been taken to the CUTE library [22, 24]. All the algorithms have been coded in MATLAB 2013 and compiler settings on the PC machine (2.5 GHz, 3.8 GB RAM) with Windows XP operating system. We compare the computational results of hHD method against the DY [11], NVHS* [14], CCOMB [21], hDYz [18], HYBRID [20] and HLSDY [2] methods. In this numerical result, all algorithms implement the SWLS condition with $\delta = 10^{-3}$ and $\sigma = 10^{-1}$. The iteration is terminated if one of the following conditions is satisfied (i) $\|g_k\|_\infty < 10^{-6}$, where $\|\cdot\|_\infty$ is the maximum absolute component of a vector, (ii) The number of iterations exceeded 2000, (iii) The computing time is more than 500 s. We show the performance difference clearly between our method and six conjugate gradient algorithms. We choose the performance profile introduced by Dolan and Moré [25] to compare the performance according to number iteration and CPU time to rule as follows. Let S is the set of methods and P is the set of the test problems with n_p , n_s is the number of the test problems and the number of the methods, respectively. For each problem $p \in P$ and solver $s \in S$, denote $\tau_{p,s}$ be the number of iterations or CPU time required to solve problems $p \in P$ by solver $s \in S$. Then comparison between different solvers based on the performance ratio is given by

$$r_{p,s} = \frac{\tau_{p,s}}{\min \{\tau_{p,i}, 1 \leq i \leq n_s\}}.$$

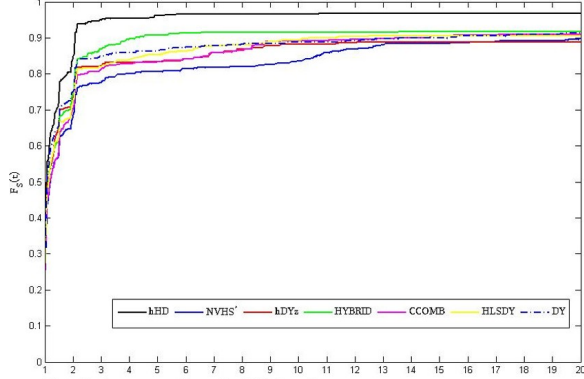
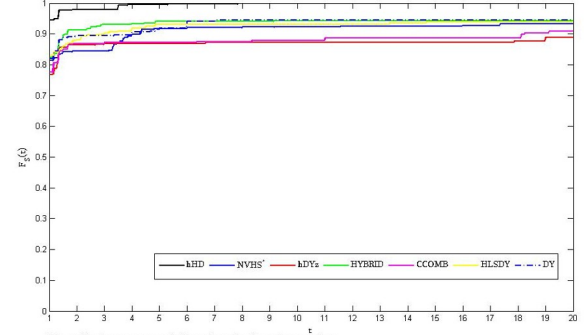
Suppose that a parameter $r_M \geq r_{p,s}$ for all problems and solvers chosen, and $r_M = r_{p,s}$ if and only if solver S does not solve problem p . The overall evaluation of the performance of the solvers is then given by the performance profile function given by

$$F_s(t) = \frac{\text{size} \{p : 1 \leq p \leq n_p, r_{p,s} \leq t\}}{n_p},$$

where $t \geq 1$ and $\text{size} \{p : 1 \leq p \leq n_p, r_{p,s} \leq t\}$ is the number of elements in the set

$\{p : 1 \leq p \leq n_p, r_{p,s} \leq t\}$. This function $F_s : [1, \infty[\rightarrow [0, 1]$ is the distribution function for the performance ratio. The value of $F_s(1)$ is the probability that the solver will win the rest of the solvers. In this numerical study, Dim denotes the dimension of the problem, ITR denotes the number of iterations, TIME denotes the CPU time and Inf denotes the algorithm failed to yield a solution for the problem.

Table 2, Fig. 1 and Fig. 2 give a performance comparison of the hHD method with those for the CPU time and the number of iterations. From these Figs. and Table 2, we can conclude that the hHD method is competitive and is superior to DY [11], NVHS* [14], CCOMB [21], hDYz [18], HYBRID [20] and HLSDY [2] conjugate gradient methods for the given test problems. These obtained preliminary results are indeed encouraging.

Fig. 1. Performance profile on the CPU time. t Fig. 2. Performance profile based on the iteration number. t

Functions	NVHS*		hHD		HLSDY		hDYz		HYBRID		CCOMB		DY		Dim
	ITR	TIME	ITR	TIME	ITR	TIME	ITR	TIME	ITR	TIME	ITR	TIME	ITR	TIME	
Hager	49	1.6870	7	0.1560	51	1.7970	8	0.2970	12	0.3070	50	1.6880	68	2.0160	1900
Griewank	43	5.3900	23	2.6280	23	2.7780	65	14.3720	20	2.6310	19	2.6200	22	2.7400	3000
Dixon	47	0.3310	42	0.3010	54	0.4820	56	0.4910	49	0.3640	48	0.3610	51	0.4520	800
Diagonal 4	23	3.7960	8	0.4580	7	2.1570	9	0.3910	7	0.4530	8	0.4530	7	0.4530	3000
Chung	5	2.1410	4	2.1200	5	2.1560	6	2.1410	5	2.1410	5	2.1410	5	2.1400	500
Branin	9	0.1620	5	0.0980	29	0.1720	8	0.1470	8	0.1470	29	0.1360	29	0.1560	2
Beale	21	0.0150	14	0.0150	21	0.0160	22	0.0160	22	0.0160	18	0.0320	21	0.0150	7
Tridiagonal	7	0.3980	5	0.3280	12	0.6250	9	0.4810	8	0.4530	8	0.4580	7	0.4030	1470
Alpine 2	9	2.9060	6	2.3450	9	2.8750	12	3.1010	8	2.3910	24	4.0840	10	2.9630	2600
Staircas S3	7	0.0310	7	0.0320	7	0.0320	7	0.0310	7	0.0310	7	0.0470	7	0.0310	400
Staircas S2	5	0.0610	3	0.0310	4	0.0470	4	0.0470	5	0.0630	4	0.0630	3	0.0320	470
Nonscomp	2	0.0160	2	0.0150	2	0.0310	2	0.0150	2	0.0150	2	0.0320	2	0.0310	1470
Perturbed	3	0.0310	3	0.0150	3	0.0310	3	0.0320	3	0.0160	3	0.0150	3	0.0310	500
Diagonal 1	12	0.3270	4	0.0460	6	0.0780	870	29.4060	5	0.0670	192	5.1710	12	0.3280	2800
Liarwhd	51	1.6810	42	1.3470	48	1.6280	48	1.6270	44	1.4840	46	1.6230	44	1.4810	1000
Schwefel223	10	0.1870	11	0.1970	10	0.1880	12	0.2040	10	0.1880	10	0.1880	12	0.2040	470
Cube	421	6.3280	224	3.3410	668	8.8470	Diverged	Diverged	298	3.8320	371	5.2350	267	3.8310	350
Rosenbrock	21	1.3440	8	0.5470	18	1.2480	119	5.8280	17	1.1090	79	3.1100	22	1.1930	2700
Power	1080	29.0750	700	25.1100	1190	27.1890	1297	31.9320	Diverged	Diverged	1385	32.8570	871	26.1480	1300
Perturbed	62	2.8320	57	2.6110	65	2.9220	104	3.5470	88	3.0880	65	2.8340	60	2.7440	1800
Lion	3	0.0310	3	0.0150	3	0.0310	3	0.0160	3	0.0150	3	0.0310	3	0.0160	2
Rayden1	7	0.4030	6	0.3210	9	0.4530	11	0.7030	10	0.6720	29	1.4770	15	0.8280	900
Zakharov	7	0.4030	9	0.3210	9	0.4530	11	0.7030	10	0.6720	29	1.4770	15	0.8280	1200
Rasting	30	1.9850	6	0.1020	Diverged	Diverged	8	0.1690	11	0.3720	53	2.1700	161	9.7480	1200
Quadratic	15	0.4130	22	0.5830	21	0.5190	17	0.4230	17	0.4350	18	0.4500	15	0.4180	3100
Staircas S1	56	2.9440	53	2.8800	102	4.9400	58	2.9460	55	2.9420	74	3.1250	78	3.1090	1470
Diagonal 2	Diverged	Diverged	7	0.0160	Diverged	Diverged	176	0.1250	Diverged	Diverged	180	0.1410	Diverged	Diverged	63
Almost Perturbed Quadratic	8	15.773	9	3.110	8	4.482	7	5.830	3	4.651	7	4.226	8	4.027	2000
England 1	7	58.871	2	0.390	8	0.346	8	0.418	7	0.416	6	0.631	8	0.442	3000
Almost Perturbed Quartic	23	128.456	5	0.411	2	2.246	4	1.859	9	1.437	4	1.542	8	2.628	1200
Nondia	Diverged	Diverged	8	0.043	Diverged	0.078	Diverged	Diverged	19	0.062	Diverged	Diverged	7	0.078	1800
Quartem	4	3.469	3	3.347	3	2.628	2	2.633	3	1.484	2	2.588	3	3.481	1800
Linear Perturbed	8	5.403	1	0.343	0.125	1.281	Diverged	Diverged	39	40.100	Diverged	Diverged	4	1.414	1000
England 8	Diverged	Diverged	5	13.746	Diverged	13.852	Diverged	Diverged	5	14.623	Diverged	Diverged	7	14.623	2000
Feletcher	28	42.191	2	3.845	1	1.875	2	7.162	2	2.391	2	2.086	2	2.921	2600
Dixon 3 dq	6	6.666	6	0.047	6	0.042	6	0.047	6	0.046	6	0.062	6	0.511	500
Tridia	12	8.470	0	5.118	11	6.189	3	5.932	Diverged	Diverged	85	2.857	6	5.148	12600
Double Border Arrow Up	8	5.667	8	0.416	2	0.284	0	0.381	8	0.375	0	0.382	2	0.402	4000
DiagonalDiag-Aup1	3	3.333	3	0.072	3	0.063	3	0.062	3	0.062	3	0.076	3	0.072	10000

TABLE 2. The simulation results of hHD, NVHS*, HLSDY, hDYz, HYBRID, CCOMB and DY methods

Note: In this table, “Diverged” indicates that the algorithm failed to converge. The possible failure modes are: (i) the number of iterations exceeded the maximum limit of 2000, (ii) the CPU time exceeded 500 seconds, or (iii) the gradient norm did not satisfy the stopping criterion $\|g_k\|_\infty < 10^{-6}$ within the allowed iterations.

5. APPLICATION IN MODE FUNCTION

Nonparametric estimation has received a great deal of attention in both theoretical and applied statistics literature. For the historical and mathematical survey, we refer to the reader to Sager [26]. In

statistics, it is always interesting to study the central tendency of the data, that is usually quantified using the location parameters (mean, mode, median). The problem of estimating the mode function of a probability density function (p.d.f.) has taken considerable attention in the past for both independent and dependent data, and a number of distinguished papers deal with this topic. For example, Parzen [27] and Eddy [28] for estimation of the unconditional mode in the independent and identically distributed (i.i.d.) case.

In this section, we consider the problem of estimating the mode of a multivariate unimodal probability density f with support in $\mathbb{R}^n$ from i.i.d. standard normal random variables $X_1, \dots, X_n$ with common probability density function f . This problem has been investigated in numerous papers. To quote a few of them, Konakov [19] and Samanta [15]. We assume that density f has a unique mode denoted by θ and defined by

$$f(\theta) = \max_{x \in \mathbb{R}^n} f(x). \quad (5.1)$$

A kernel estimator of the mode θ is defined as the random variable $\hat{\theta}$ which maximizes the kernel estimator $f_n(x)$ of $f(x)$, that is

$$f_n(\hat{\theta}) = \max_{x \in \mathbb{R}^n} f_n(x), \quad (5.2)$$

where

$$f_n(x) = \frac{1}{nh_n^n} \sum_{i=1}^n K\left(\frac{x - X_i}{h_n}\right). \quad (5.3)$$

The bandwidth (h_n) is a sequence of positive real numbers which goes to zero as n goes to infinity and the kernel K is a p.d.f. on $\mathbb{R}^n$. The kernel function K controls the smoothness of the estimated density. The Gaussian kernel produces infinitely smooth estimates, while the Epanechnikov kernel has compact support and is optimal in terms of asymptotic mean integrated squared error. The bandwidth h_n plays a crucial role in balancing bias and variance: a small bandwidth leads to undersmoothing (high variance), while a large bandwidth leads to oversmoothing (high bias). The cross-validation method selects the optimal bandwidth by minimizing an estimate of the prediction error, which provides a data-driven criterion for bandwidth selection.

In this simulation, we choose between two different types of kernel: the standard Gaussian kernel defined by

$$K(x) = \frac{1}{(2\pi)^{\frac{n}{2}}} \exp\left(-\frac{1}{2} \sum_{j=1}^n x_j^2\right),$$

and the Epanechnikov kernel given by

$$K(x) = \left(\frac{3}{4}\right)^n \prod_{j=1}^n (1 - x_j^2).$$

The selection of the bandwidth h is an important and basic problem in kernel smoothing techniques. In this simulation, we choose the optimal bandwidth by the cross-validation method.

In this context, we employ the hHD method to solve the problem (5.2) under the strong Wolfe line search technique and compare it with NVHS* [25] and DY [23] methods. We choose some initial points and obtain the results in Table 3. Additionally, we compare the hHD method with COMB method, and the results are shown in Table 4.

According to Table 3, it is clear that the hHD method is more efficient than NVHS* [25], COMB [21], and DY [23] methods based on the number of iterations and CPU time for solving the problem (5.2).

Conclusion

We have presented in this paper a new hybrid conjugate gradient algorithm by using the convex combination of NVHS* and DY methods. The global convergence properties and the sufficient descent condition of the proposed method have been established, under the strong Wolfe line search conditions.

TABLE 3. The simulation results of hHD, CCOMB, NVHS* and DY methods for solving problem (5.2)

Model	Initial Points	Dim	hHD		CCOMB		NVHS*		DY	
			ITR	TIME	ITR	TIME	ITR	TIME	ITR	TIME
Linear	0.2	26	80	6.5000	464	37.3410	975	67.5470	850	60.1230
		32	35	3.5000	59	6.7480	86	8.9830	75	7.8910
		37	90	10.5000	110	13.8640	57	7.7770	60	8.1020
		40	65	8.2000	84	12.3570	85	12.1750	82	11.9560
		44	160	28.0000	84	17.5780	84	17.6570	80	16.5280
		48	30	6.5000	54	12.1710	54	12.1710	50	11.0820
		58	6	2.1000	50	16.5830	43	15.0470	45	14.2000
		60	25	8.0000	14	4.3040	13	4.2660	15	5.4720
Nonlinear	0.5	160	70	180.00	22	36.8900	7	17.6860	10	20.8430
		170	26	78.0000	43	60.5780	63	80.5010	55	70.3570
		190	16	55.0000	15	45.3470	15	47.1470	18	50.0080
		180	15	48.0000	18	65.6030	18	65.5320	20	68.0140
		270	22	160.0000	47	341.400	48	350.620	45	330.3450
		290	10	85.0000	12	102.340	18	151.870	15	130.5320
		320	12	125.0000	16	157.340	28	289.170	25	250.0000
		340	8	90.0000	18	245.220	16	236.390	14	220.0230
Nonlinear	-0.3	54	32	7.5000	19	4.5742	16	3.9841	18	5.0000
		56	225	70.0000	41	12.6510	41	12.9020	45	15.0160
		66	34	13.5000	53	23.3400	53	23.6040	50	22.4560
		68	14	7.0000	35	14.6710	27	12.7020	30	13.9640
		78	390	185.0000	62	28.1250	124	51.6980	100	45.0030
		70	37	18.0000	56	27.8930	53	27.4250	55	26.0340
		72	22	11.5000	50	26.6290	52	27.3590	48	25.8730
		74	16	8.5000	20	10.3221	21	11.0090	22	12.2460

Numerical results also show that hHD method is robust and effective. The practical applicability of our method is also explored in the nonparametric estimation of the mode function.

Acknowledgments

The authors are very grateful to the anonymous referees for their valuable comments and useful suggestions, which improved the quality of this paper.

REFERENCES

- [1] Chaib Y., Mehamdia A. E., Global convergence of hybrid conjugate gradient method and its application to nonparametric estimation. Mathematical Foundations of Computing, (2025) Vol. 8, P. 296–309.
- [2] Liu J. K., Li S. J., New hybrid conjugate gradient method for unconstrained optimization. Appl. Math. Comput. (2014) Vol. 245, P. 36–43.
- [3] Mehamdia A. E., Yacine C., Khudhur H. M., Ziadi R., Two modified conjugate gradient methods and their application to image restoration problems. Ural mathematical journal, (2025) Vol. 11,
- [4] Mehamdia A. E., Chaib Y., Bechouat T., A new hybrid conjugate gradient method as a convex combination methods. Bulletin of the Transilvania University of Brasov Series III Mathematics and Computer Science, (2025) Vol. 5,
- [5] Hestenes M. R., Stiefel E. L., Methods of conjugate gradients for solving linear systems. J. Res. Natl. Bur. Stand. (1952) Vol. 49, P. 409–436.
- [6] Polak E., Ribière G., Note sur la convergence de directions conjuguée. Rev. Française Inform. Rech. Opér. (1969) Vol. 16, P. 35–43.
- [7] Polyak B. T., The conjugate gradient method in extreme problems. USSR Comput. Math. Math. Phys. (1969) Vol. 9, P. 94–112.
- [8] Liu Y., Storey C., Efficient generalized conjugate gradient algorithms. J. Optim. Theory Appl. (1991) Vol. 69, P. 129–137.
- [9] Fletcher R., Reeves C., Function minimization by conjugate gradients. Comput. J. (1964) Vol. 7, P. 149–154.

- [10] Fletcher R., Practical Methods of Optimization. Wiley, New York, (1987).
- [11] Dai Y. H., Yuan Y., A nonlinear conjugate gradient method with a strong global convergence property. SIAM J. Optim. (1999) Vol. 10, P. 177–182.
- [12] Yao S. W., Wei Z. X., Huang H., A note about WYL's conjugate gradient method and its applications. Appl. Math. Comput. (2007) Vol. 191, P. 381–388.
- [13] Zhang L., An improved Wei-Yao-Liu nonlinear conjugate gradient method for optimization computation. Appl. Math. Comput. (2009) Vol. 6, P. 2269–2274.
- [14] Du X. W., Zhang P., Ma W., Some modified conjugate gradient methods for unconstrained optimization. J. Comput. Appl. Math. (2016) Vol. 305, P. 92–114.
- [15] Touati-Ahmed D., Storey C., Efficient hybrid conjugate gradient techniques. J. Optim. Theory Appl. (1990) Vol. 64, P. 379–397.
- [16] Hu Y. F., Storey C., Global convergence result for conjugate gradient methods. J. Optim. Theory Appl. (1991) Vol. 71, P. 399–405.
- [17] Gilbert J. C., Nocedal J., Global convergence properties of conjugate gradient methods for optimization. SIAM J. Optim. (1992) Vol. 2, P. 21–42.
- [18] Dai Y. H., Yuan Y. X., An efficient hybrid conjugate gradient method for unconstrained optimization. Ann. Oper. Res. (2001) Vol. 103, P. 33–47.
- [19] Konakov V. D., On the asymptotic normality of the mode of multidimensional distributions. Theory Probab. Appl. (1974) Vol. 19, P. 794–799.
- [20] Andrei N., A hybrid conjugate gradient algorithm for unconstrained optimization as a convex combination of Hestenes-Stiefel and Dai-Yuan. Stud. Inform. Control, (2008) Vol. 17, P. 55–70.
- [21] Andrei N., New hybrid conjugate gradient algorithms for unconstrained optimization. J. Optim. Theory Appl. (2009) Vol. 141, P. 249–264.
- [22] Andrei N., An unconstrained optimization test functions. Adv. Model. Optim. (2008) Vol. 10, P. 147–161.
- [23] Dai Y. H., Han J. Y., Liu G. H., Sun D. F., Yin X., Yuan Y., Convergence properties of nonlinear conjugate gradient methods. SIAM J. Optim. (1999) Vol. 10, P. 348–358.
- [24] Bongartz I., Conn A. R., Gould N., Toint P. L., Constrained and unconstrained testing environment. ACM Trans. Math. Softw. (1995) Vol. 21, P. 123–160.
- [25] Dolan E. D., Moré J. J., Benchmarking optimization software with performance profiles. Math. Program. (2002) Vol. 91, P. 201–213.
- [26] Sager T. W., An iterative method for estimating a multivariate mode and isopleth. J. Am. Statist. Assoc. (1975) Vol. 74, P. 329–339.
- [27] Parzen E., On estimating probability density function and mode. Ann. Math. Statist. (1962) Vol. 33, P. 1065–1076.
- [28] Eddy W. F., Optimum kernel estimators of the mode. Ann. Statist. (1980) Vol. 8, P. 870–882.

Mehamdia A. E.,
 Laboratory Informatics and Mathematics (LIM),
 Mohamed Cherif Messaadia University, Souk Ahras, Algeria
 e-mail: a.mehamdia@univ-soukahras.dz

Chaib Y.,
 Laboratory Informatics and Mathematics (LIM),
 Mohamed Cherif Messaadia University, Souk Ahras, Algeria
 e-mail: y.chaib@univ-soukahras.dz

Bechouat T.,
 Mohamed Cherif Messaadia University, Souk Ahras, Algeria
 e-mail: bachoite.t@gmail.com