

Research Article

A HS-PRP-Type Hybrid Conjugate Gradient Method with Sufficient Descent Property

Xiaodi Wu,¹ Yihan Zhu,¹ and Jianghua Yin ²

¹College of Mathematics and Physics, Guangxi University for Nationalities, Nanning 530006, China

²College of Mathematics and Computer Science, Guangxi Science and Technology Normal University, Laibin 546199, China

Correspondence should be addressed to Jianghua Yin; jianghuayin1017@126.com

Received 5 September 2021; Revised 24 September 2021; Accepted 4 October 2021; Published 22 October 2021

Academic Editor: Ahmed Mostafa Khalil

Copyright © 2021 Xiaodi Wu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, based on the HS method and a modified version of the PRP method, a hybrid conjugate gradient (CG) method is proposed for solving large-scale unconstrained optimization problems. The CG parameter generated by the method is always nonnegative. Moreover, the search direction possesses the sufficient descent property independent of line search. Utilizing the standard Wolfe–Powell line search rule to yield the stepsize, the global convergence of the proposed method is shown under the common assumptions. Finally, numerical results show that the proposed method is promising compared with two existing methods.

1. Introduction

Consider the problem of minimizing f over R^n :

$$\min_{x \in R^n} f(x), \quad (1)$$

where $f: R^n \rightarrow R$ is continuously differentiable. Throughout, the gradient of f at x is denoted by $g(x)$, i.e., $g(x) := \nabla f(x)$. We know that conjugate gradient (CG) methods are very popular and effective for solving unconstrained optimization problems (1), especially for large-scale case by means of their simplicity and low memory requirements. These preferred features greatly promote their applications in various areas such as image deblurring and denoising, neural network, compressed sensing, and others. We refer the interested readers to some recent works [1–3] and references therein for more details. The numerical results reported in [1] reveal that the CG method has great potential in solving image restoration problems.

Generally, the iterative formula of the CG method for solving problem (1) can be read as

$$x_{k+1} = x_k + \alpha_k d_k, \quad (2)$$

where $\alpha_k > 0$ is called the stepsize computed by some line search. Here, d_k is commonly known as the search direction, which is defined as follows:

$$d_k = \begin{cases} -g_k, & \text{if } k = 1, \\ -g_k + \beta_k d_{k-1}, & \text{if } k \geq 2, \end{cases} \quad (3)$$

where $\beta_k \in R$ is the so-called CG parameter and g_k is the abbreviation of $g(x_k)$, i.e., $g_k := g(x_k)$. The two key factors that affect the numerical performance of the CG method are the stepsize and the CG parameter. First, we outline several well-known line search criteria in the literature.

- (a) The exact line search rule: calculate a stepsize α_k satisfying

$$f(x_k + \alpha_k d_k) = \min_{\alpha \geq 0} f(x_k + \alpha d_k). \quad (4)$$

- (b) The standard (weak) Wolfe–Powell (WWP) line search rule: calculate a stepsize α_k satisfying

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k, \quad (5)$$

and

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k, \quad (6)$$

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

(c) The strong Wolfe–Powell (SWP) line search rule: calculate a stepsize α_k satisfying (5) and

$$|g(x_k + \alpha_k d_k)^T d_k| \leq \sigma |g_k^T d_k|. \quad (7)$$

On the other hand, different CG methods are determined by different CG parameters. The well-known CG methods include the Fletcher–Reeves (FR) [4], Polak–Ribière–Polyak (PRP) [5, 6], Hestenes–Stiefel (HS) [7], Liu–Storey (LS) [8], Fletcher (CD) [9], and Dai–Yuan (DY) [10] methods, and their CG parameters β_k are, respectively, given by

$$\begin{aligned} \beta_k^{\text{FR}} &= \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \\ \beta_k^{\text{PRP}} &= \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \\ \beta_k^{\text{HS}} &= \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \\ \beta_k^{\text{LS}} &= -\frac{g_k^T y_{k-1}}{g_{k-1}^T d_{k-1}}, \\ \beta_k^{\text{CD}} &= -\frac{\|g_k\|^2}{d_{k-1}^T g_{k-1}}, \\ \beta_k^{\text{DY}} &= \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}, \end{aligned} \quad (8)$$

where $y_{k-1} := g_k - g_{k-1}$ and $\|\cdot\|$ stands for the Euclidean norm. The methods yielded by the above CG parameters are called the classical CG methods, and their convergence analysis and numerical performance have been extensively studied (see, e.g., [4–12]). It has been shown that the above formulas for the CG parameters are equivalent when $f(x)$ is convex quadratic and the stepsize α_k is obtained by carrying out the exact line search rule (4). However, their numerical performance strongly depends on the CG parameter β_k . The FR, CD, and DY methods possess good convergence, but the numerical performance for these methods is somewhat unsatisfactory for solving general unconstrained nonlinear optimization problems [12–14]. On the contrary, it has been shown that the convergence properties of PRP, HS, and LS methods are not so well, but they often possess better computational performance [12–14]. Therefore, in the past few decades, based on the above formulas, plenty of formulas for β_k are designed for CG methods that possess both good global convergence properties and promising numerical performance (see [12–16] and references therein).

To our knowledge, the first hybrid CG method in the literature was proposed by Touati–Ahmed and Storey [17] (TS method), where β_k is computed as

$$\beta_k^{\text{TS}} = \begin{cases} \beta_k^{\text{PRP}}, & \text{if } 0 \leq \beta_k^{\text{PRP}} \leq \beta_k^{\text{FR}}, \\ \beta_k^{\text{FR}}, & \text{otherwise.} \end{cases} \quad (9)$$

Apparently, the TS method has some good properties of FR and PRP methods since β_k^{TS} is a hybrid of β_k^{FR} and β_k^{PRP} . Combined with HS and DY methods, Dai and Yuan [18] proposed another hybrid CG method (hHD method), in which the hybrid CG parameter β_k is obtained by

$$\beta_k^{\text{hHD}} = \max\{0, \min\{\beta_k^{\text{HS}}, \beta_k^{\text{DY}}\}\}. \quad (10)$$

When the WWP line search rule is used to compute the stepsize, the resulting search direction in [18] is a descent one and the global convergence for the hHD method is proved. Moreover, the numerical experiments reported in [18] illustrated that the hHD method is competitive and practicable. For other closely related works, we refer the readers to [18, 19] and the references therein. It is worth noting that the CG parameters β_k defined in [17–19] are restricted to positive values. As explicated in [19], this restriction in turn results in global convergence of the algorithm. In recent years, many hybrid CG methods were proposed on the basis of the methodology of discrete combinations of several CG parameters (see, e.g., [1, 13, 20–23]). The combination parameter is computed by some secant equations [13, 20], the conjugacy condition [21, 22], or by minimizing the least-squares problem consisting of the unknown search direction and an existing one (see [23] and the references therein).

In 2016, Wei et al. [24] introduced a modified PRP method, usually called the WYL method, where the corresponding parameter β_k is yielded by

$$\beta_k^{\text{WYL}} = \frac{\|g_k\|^2 - \|g_k\| \|g_{k-1}\| |g_k^T g_{k-1}|}{\|g_{k-1}\|^2}. \quad (11)$$

Under the assumption that d_k generated by Wei et al. [24] satisfies the so-called sufficient descent condition

$$g_k^T d_k \leq -c \|g_k\|^2, \quad c > 0, \quad (12)$$

the WYL method is globally convergent under the WWP line search rule and possesses superior numerical performance. Subsequently, Dai and Wen [25] proposed two improved CG methods with sufficient descent property. The CG parameters β_k in [25] are defined as

$$\begin{aligned} \beta_k^{\text{DHS}} &= \frac{\|g_k\|^2 - \|g_k\| \|g_{k-1}\| |g_k^T g_{k-1}|}{d_{k-1}^T y_{k-1} + \mu |g_k^T d_{k-1}|}, \\ \beta_k^{\text{DPRP}} &= \frac{\|g_k\|^2 - \|g_k\| \|g_{k-1}\| |g_k^T g_{k-1}|}{\|g_{k-1}\|^2 + \mu |g_k^T d_{k-1}|}, \end{aligned} \quad (13)$$

Step 0: (initialization) given the initial point $x_1 \in R^n, \gamma > 2, \varepsilon > 0$. Set $d_1: = -g_1$ and $k: = 1$.
 Step 1: stop if $\|g_k\| \leq \varepsilon$.
 Step 2: compute the stepsize α_k by an appropriate line search rule.
 Step 3: generate the new iteration point by $x_{k+1} = x_k + \alpha_k d_k$ and compute $\beta_k: = \beta_k^{\text{hHPR}}$ according to (14).
 Step 4: compute the next direction d_{k+1} by (3), set $k: = k + 1$, and go to Step 1.

ALGORITHM 1: Algorithm framework.

where $\mu > 1$. Clearly, the search direction yielded by β_k^{DPRP} satisfies the sufficient descent condition without depending on any line search. However, the sufficient descent property associated with β_k^{DHS} relies on the WWP line search rule.

Based on the above observations, it is interesting to design a hybrid CG method such that the CG parameter is non-negative and the resulting search direction possesses the sufficient descent property independent of line search technique. Motivated by the methods in [24, 25] and considering that the HS method performs best among the classical CG methods, a new formula for the CG parameter β_k is given by

$$\beta_k^{\text{hHPR}} = \min \left\{ |\beta_k^{\text{HS}}|, \frac{\|g_k\|^2 - \|g_k\| \|g_{k-1}\| |g_k^T g_{k-1}|}{\|g_{k-1}\|^2 + \gamma |g_k^T d_{k-1}|} \right\}, \quad (14)$$

where $\gamma > 2$. It is not difficult to see that β_k^{hHPR} is a hybrid of $\beta_k^{\text{HS}}, \beta_k^{\text{WYL}}$, and β_k^{DPRP} . Interestingly, the above parameter β_k^{hHPR} is always nonnegative. To see this, let θ_k be the angle between g_k and g_{k-1} . Thus, we know from (14) that

$$\begin{aligned} \beta_k^{\text{hHPR}} &\leq \frac{\|g_k\|^2 - \|g_k\| \|g_{k-1}\| |g_k^T g_{k-1}|}{\|g_{k-1}\|^2 + \gamma |g_k^T d_{k-1}|} = \frac{\|g_k\|^2 (1 - \cos \theta_k)}{\|g_{k-1}\|^2 + \gamma |g_k^T d_{k-1}|} \\ &\leq \frac{2\|g_k\|^2}{\|g_{k-1}\|^2 + \gamma |g_k^T d_{k-1}|}, \end{aligned} \quad (15)$$

which further implies

$$0 \leq \beta_k^{\text{hHPR}} \leq \frac{2\|g_k\|^2}{\|g_{k-1}\|^2}. \quad (16)$$

Moreover, plugging the CG parameter $\beta_k: = \beta_k^{\text{hHPR}}$ into (3), we can show that the resulting search direction possesses the sufficient descent property independent of line search technique (see Lemma 1 below).

The structure of this paper is organized as follows. In Section 2, our algorithm framework is presented, and the sufficient descent property with respect to the resulting search direction is discussed in detail. Section 3 is devoted to establishing the convergence of the proposed method with the WWP line search rule. In the last section, some preliminary numerical results are reported to verify the efficiency of the presented method.

2. The Algorithm

In this section, we first propose the algorithm framework for solving problem (1), in which we do not specify which line search rule generates the stepsize. Subsequently, we analyze the sufficient descent property for the search direction. By inserting the WWP line search rule into the algorithm framework, our hybrid CG method is proposed.

The following lemma shows that the direction sequence $\{d_k\}$ generated by Algorithm 1 possesses the sufficient descent property independent of any line search.

Lemma 1. *Let $\{d_k\}$ be a sequence generated by Algorithm 1. Then, for some constant $M \in (0, 1)$, it holds that*

$$g_k^T d_k \leq -M \|g_k\|^2, \quad \forall k \geq 1. \quad (17)$$

Proof. When $k = 1$, it follows from the definition of d_k in (3) that $g_1^T d_1 = -\|g_1\|^2 \leq -M \|g_1\|^2$. So, the relation in (17) holds when $k = 1$. Now, consider the case $k \geq 2$. If $g_k^T d_{k-1} = 0$, it follows from (3) that

$$g_k^T d_k = -\|g_k\|^2 \leq -M \|g_k\|^2. \quad (18)$$

Suppose that $g_k^T d_{k-1} \neq 0$ for all $k \geq 2$. It then follows from (3), (15), and (16) that

$$\begin{aligned} g_k^T d_k &= -\|g_k\|^2 + \beta_k^{\text{hHPR}} g_k^T d_{k-1} \\ &\leq -\|g_k\|^2 + \frac{2\|g_k\|^2}{\|g_{k-1}\|^2 + \gamma |g_k^T d_{k-1}|} |g_k^T d_{k-1}| \\ &\leq -\|g_k\|^2 + \frac{2\|g_k\|^2}{\gamma |g_k^T d_{k-1}|} |g_k^T d_{k-1}| \\ &\leq -\left(1 - \frac{2}{\gamma}\right) \|g_k\|^2 =: -M \|g_k\|^2, \end{aligned} \quad (19)$$

which completes the proof. \square

For convenience, in the following statements, we call the method generated by Algorithm 1 with the WWP line search rule as the hHPR CG method.

3. Convergence

In this section, we analyze the convergence for the hHPR CG method. For this goal, the following common assumptions are necessary.

Assumption 1

- (i) The level set $\Omega = \{x \in R^n | f(x) \leq f(x_1)\}$ is bounded. Here, x_1 is the given initial point.
- (ii) In some neighborhood N of the level set Ω , the objective function $f(x)$ is continuously differentiable, and its gradient $g(x)$ is Lipschitz continuous, i.e., there exists a constant $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\|, \quad \forall x, y \in N. \quad (20)$$

The following lemma provides the convergence for the PRP-type CG method, which was originally introduced in [19].

Lemma 2. Consider the general CG method (2) and (3) with the following three properties:

- (i) The CG parameter is always nonnegative, i.e., $\beta_k \geq 0$ for all $k \geq 1$.
- (ii) The line search satisfies (5) and (6) and the sufficient descent condition.
- (iii) Property (*) holds. Then,

$$\liminf_{k \rightarrow \infty} \|g_k\| = 0. \quad (21)$$

Property 1. (*) Consider a method of forms (2) and (3). Suppose that

$$0 < \gamma \leq \|g_k\| \leq \bar{\gamma}, \quad \forall k \geq 1. \quad (22)$$

We say that the method has property (*), if for all $k \geq 1$, there exist constants $b > 1$ and $\lambda > 0$ such that $|\beta_k| \leq b$, and if $\|s_{k-1}\| \leq \lambda$ where $s_{k-1} = x_k - x_{k-1}$, then we have $|\beta_k| \leq 1/2b$.

From (16) and Lemmas 1 and 2, to obtain the global convergence of the hHPR CG method, we only prove that our method owns property (*).

Lemma 3. Consider the method of forms (2) and (3) in which $\beta_k = \beta_k^{\text{hHPR}}$. If Assumption 1 holds, then β_k^{hHPR} satisfies property (*).

Proof. Considering the method of forms (2) and (3) and using the constants γ and $\bar{\gamma}$ in (22), we have from (16) that

$$0 \leq \beta_k^{\text{hHPR}} \leq \frac{2\|g_k\|^2}{\|g_{k-1}\|^2} \leq \frac{2\bar{\gamma}^2}{\gamma^2}. \quad (23)$$

Let $b = 2\bar{\gamma}^2/\gamma^2 \geq 2$ and $\lambda = \gamma^4/8L\bar{\gamma}^3 > 0$. If $\|s_{k-1}\| \leq \lambda$, we obtain from Assumption 1(ii) and (15) that

$$\begin{aligned} \beta_k^{\text{hHPR}} &\leq \frac{g_k^T (g_k - \|g_k\|/ \|g_{k-1}\| g_{k-1})}{\|g_{k-1}\|^2} \\ &\leq \frac{\|g_k\| \cdot \|g_k - g_{k-1} + g_{k-1} - \|g_k\|/ \|g_{k-1}\| g_{k-1}\|}{\|g_{k-1}\|^2} \\ &\leq \frac{\|g_k\| \cdot \|g_k - g_{k-1}\| + \|g_k\| \cdot \|\|g_{k-1}\| - \|g_k\|\|}{\|g_{k-1}\|^2} \quad (24) \\ &\leq \frac{2\|g_k\| \|g_k - g_{k-1}\|}{\|g_{k-1}\|^2} \leq \frac{2L\|s_{k-1}\| \|g_k\|}{\|g_{k-1}\|^2} \\ &\leq \frac{2L\lambda\bar{\gamma}}{\gamma^2} = \frac{1}{2b}. \end{aligned}$$

Therefore, the proof is completed. \square

With (16) and Lemmas 1–3 at hand, one can establish the global convergence of the hHPR CG method.

Theorem 1. Let $\{x_k\}$ be a sequence generated by the hHPR CG method. If Assumption 1 holds, then $\liminf_{k \rightarrow \infty} \|g_k\| = 0$.

4. Numerical Experiments

In this section, we verify the efficiency and robustness of the hHPR CG method (hHPR for short) by solving some classical tested problems and compare it with two well-known CG methods: DHS and DPRP in [25].

For the tested problems, some of them are from the well-known CUTE library in [26] and the others come from [27]. Moreover, their dimensions range from 2 to 1000000. All codes were written in MATLAB R2016a, and the numerical experiments were conducted on a Dell PC with Intel Core CPU 3.00 GHz and 16.00 GB RAM. For the aforementioned methods, we reset the search direction by taking $d_k = -g_k$ once an ascent direction occurs. For the sake of fairness, all the stepsizes α_k are yielded by the WWP line search rule following a bisection algorithm proposed in [28], and the corresponding parameters are set to $\delta = 0.01$ and $\sigma = 0.1$. Moreover, we adopt the strategy described in [29] to compute the initial stepsize.

Let $\gamma = 3$ for hHPR, and let $\mu = 2$ for DHS and DPRP. Denote the iteration numbers, the CPU time in seconds, and the final value of $\|g_k\|$ by Itr, Tcpu, and $\|g_*$, respectively. If $\|g_k\| \leq 10^{-6}$ or Itr > 2000, we stop the program. If the latter requirement holds, i.e., Itr > 2000, we use “-” to denote Itr, Tcpu, and $\|g_*$.

The numerical results are listed in Tables 1 and 2, where “TP” denotes the tested problems used in numerical experiments and “Dim” stands for the dimension of the tested problems.

As we all know, the performance profile introduced in [30] is very useful in measuring the performance of numerical algorithms. Figures 1 and 2 plot the performance

TABLE 1: Numerical results for the three tested methods.

Problems	hHPR	DHS	DPRP
TP/Dim	Itr/Tcpu/ $\ g_*\ $	Itr/Tcpu/ $\ g_*\ $	Itr/Tcpu/ $\ g_*\ $
bdexp/50000	2/0.157/3.51e-89	2/0.136/3.51e-89	2/0.138/3.51e-89
bdexp/100000	2/0.223/4.58e-106	2/0.223/4.58e-106	2/0.220/4.58e-106
bdexp/1000000	2/2.857/1.42e-170	2/2.829/1.42e-170	2/2.848/1.42e-170
exdenschnf/50000	38/0.260/1.53e-07	30/0.230/9.80e-07	30/0.227/2.39e-07
exdenschnf/100000	42/0.496/6.67e-07	25/0.425/8.47e-07	24/0.412/2.50e-07
exdenschnb/5000	24/0.020/6.69e-07	16/0.011/3.36e-08	19/0.011/3.61e-07
exdenschnb/20000	25/0.047/2.79e-07	17/0.043/1.77e-07	22/0.057/4.59e-07
exdenschnb/100000	17/0.215/5.21e-07	17/0.182/1.48e-07	17/0.183/1.03e-07
himmelbg/20000	2/0.029/6.91e-28	2/0.022/6.91e-28	2/0.021/6.91e-28
himmelbg/100000	2/0.095/1.46e-28	2/0.094/1.46e-28	2/0.095/1.46e-28
genquartic/20000	21/0.061/3.91e-07	18/0.052/5.09e-07	13/0.045/5.23e-07
genquartic/100000	17/0.231/6.08e-07	18/0.230/4.49e-07	17/0.228/6.68e-07
genquartic/1000000	28/3.189/1.74e-07	18/2.818/5.34e-07	16/2.469/4.73e-07
biggsb1/200	1218/0.064/9.68e-07	1607/0.103/9.61e-07	1742/0.114/1.00e-06
biggsb1/400	—	—	—
sine/100	95/0.013/9.33e-07	29/0.003/5.63e-07	27/0.003/2.45e-07
sinqquad/3	79/0.015/6.74e-07	359/0.022/9.39e-07	224/0.012/5.27e-07
fletcbv3/20	101/0.014/4.42e-07	110/0.006/9.01e-07	147/0.008/5.29e-07
fletcbv3/40	409/0.024/4.63e-07	485/0.026/7.67e-07	504/0.025/7.84e-07
eg2/100	428/0.074/9.04e-07	—	1428/0.115/7.75e-07
eg2/170	1353/0.395/2.61e-07	—	—
nonscomp/10000	—	—	—
nonscomp/20000	50/0.077/7.79e-07	49/0.076/2.55e-07	46/0.073/8.37e-07
nonscomp/50000	74/0.251/6.30e-07	69/0.239/6.07e-07	80/0.266/8.04e-07
cosine/1000	51/0.020/9.66e-07	20/0.009/5.27e-07	21/0.009/6.42e-07
cosine/10000	51/0.090/5.31e-07	33/0.072/4.32e-07	21/0.057/1.04e-07
dixmaana/3000	20/0.132/2.60e-07	22/0.113/4.60e-07	17/0.107/5.22e-07
dixmaanb/3000	18/0.129/8.63e-07	20/0.107/1.96e-07	14/0.100/3.01e-07
dixmaanc/3000	17/0.130/3.91e-07	17/0.098/2.62e-07	16/0.126/6.68e-07
dixmaand/3000	23/0.146/2.23e-07	17/0.106/4.12e-07	17/0.116/3.92e-07
dixmaane/3000	428/0.915/8.90e-07	612/1.731/8.10e-07	584/1.676/9.45e-07
dixmaanf/3000	287/0.633/9.86e-07	540/1.527/9.46e-07	477/1.364/9.29e-07
dixmaang/3000	431/0.951/7.93e-07	553/1.587/9.31e-07	598/1.759/8.21e-07
dixmaanb/3000	538/1.292/5.87e-07	960/3.049/7.59e-07	918/2.984/8.35e-07
dixmaanb/3000	—	—	—
dixmaank/3000	156/0.387/6.90e-07	—	—
dixmaanl/3000	—	—	1199/3.390/9.95e-07
dixon3dq/80	876/0.043/8.55e-07	912/0.050/8.66e-07	1097/0.062/9.83e-07
dixon3dq/160	—	—	—
dqdrtic/10000	67/0.073/4.83e-07	296/0.189/7.98e-07	243/0.160/5.41e-07
dqdrtic/100000	85/0.468/9.32e-07	256/1.248/5.90e-07	153/0.819/2.43e-07
dqdrtic/1000000	70/5.007/6.70e-07	265/14.500/8.11e-07	273/14.436/7.49e-07
dqrtic/200	27/0.014/2.68e-07	24/0.008/5.68e-07	26/0.009/5.17e-08
dqrtic/500	33/0.021/5.05e-07	34/0.023/6.73e-07	29/0.020/5.84e-07
edensch/1000	40/0.051/7.37e-07	41/0.056/7.02e-07	36/0.046/8.63e-07
edensch/4000	73/0.493/8.66e-07	46/0.133/3.62e-07	43/0.204/8.20e-07
edensch/8000	44/0.620/8.57e-07	68/1.022/5.23e-07	39/0.456/6.70e-07
engval1/6	30/0.009/8.33e-07	37/0.002/2.36e-07	38/0.002/8.34e-07
errinros/3	314/0.028/8.62e-07	—	—
fletchr/100	84/0.022/9.11e-07	82/0.005/4.67e-07	67/0.005/7.99e-07
fletchr/300	45/0.005/8.17e-07	110/0.008/7.13e-07	117/0.007/8.66e-07
freuroth/50	263/0.047/8.81e-07	788/0.107/8.23e-07	664/0.054/9.91e-07
genrose/5000	—	—	—
genrose/10000	177/0.152/6.78e-07	499/0.433/9.48e-07	776/0.659/8.02e-07
genrose/20000	—	—	—

profiles of hHPR, DHS, and DPRP in terms of Itr and Tcpu, respectively. Based on the left side of Figures 1 and 2, the proposed method is clearly above the other two curves, and this in turn shows that compared with DHS and DPRP, our

proposed method is efficient and encouraging. On the other hand, based on the right side of Figures 1 and 2, our proposed method can successfully solve about 90% of the tested problems and clearly outperforms the other two methods.

TABLE 2: Numerical results for the three tested methods (continued).

Problems	hHPR	DHS	DPRP
TP/Dim	Itr/Tcpu/ $\ g_*\ $	Itr/Tcpu/ $\ g_*\ $	Itr/Tcpu/ $\ g_*\ $
liarwhd/5000	94/0.066/7.37e-07	—	—
liarwhd/10000	138/0.186/8.02e-07	—	—
liarwhd/20000	125/0.388/7.88e-07	—	—
nondquar/30	534/0.072/8.10e-07	760/0.085/9.19e-07	—
penalty1/1000	30/0.418/2.55e-07	29/0.411/2.65e-07	29/0.390/9.90e-07
penalty1/5000	117/39.785/2.62e-07	278/103.383/4.27e-07	203/72.649/1.01e-07
power1/50	595/0.029/9.03e-07	743/0.039/9.46e-07	917/0.050/2.82e-07
power1/100	1468/0.068/9.91e-07	1666/0.101/6.89e-07	—
quartc/100	23/0.010/6.45e-08	21/0.004/3.28e-07	27/0.006/1.06e-07
quartc/560	32/0.023/7.15e-07	29/0.020/7.08e-07	33/0.022/5.70e-07
tridia/100	409/0.026/6.30e-07	477/0.030/3.94e-07	681/0.038/9.21e-07
tridia/1000	1564/0.153/9.39e-07	—	—
raydan1/100	77/0.008/9.73e-07	127/0.010/8.52e-07	106/0.005/9.59e-07
raydan1/500	210/0.015/9.14e-07	281/0.023/5.80e-07	268/0.022/7.43e-07
raydan2/5000	12/0.027/9.72e-07	12/0.021/3.54e-07	12/0.023/3.54e-07
raydan2/10000	12/0.051/3.75e-07	13/0.052/7.31e-08	13/0.050/7.31e-08
raydan2/50000	16/0.239/2.43e-08	16/0.217/8.05e-07	17/0.272/7.70e-07
diagonal1/40	61/0.010/7.58e-07	81/0.005/7.52e-07	63/0.004/8.67e-07
diagonal2/10000	830/1.082/9.48e-07	1398/2.017/9.75e-07	1134/1.660/6.18e-07
diagonal2/20000	1241/3.050/8.03e-07	1387/4.025/9.98e-07	1879/5.192/8.08e-07
diagonal3/10	39/0.007/7.69e-07	37/0.002/4.92e-07	39/0.002/4.59e-07
diagonal3/90	91/0.008/7.31e-07	144/0.010/9.04e-07	166/0.016/6.14e-07
bv/1000	138/0.447/8.56e-07	117/0.513/8.14e-07	127/0.550/9.09e-07
bv/2000	105/1.174/9.03e-07	107/1.476/8.99e-07	130/1.701/9.46e-07
ie/50	16/0.051/9.24e-07	11/0.039/2.20e-07	15/0.044/1.83e-07
ie/10	14/0.162/7.41e-07	12/0.156/2.72e-07	13/0.167/2.10e-07
singx/100	277/0.051/3.49e-07	—	—
singx/1000	565/2.397/7.35e-07	—	453/1.711/5.18e-07
woods/10000	177/0.175/2.74e-07	589/0.492/2.25e-07	814/0.631/8.15e-07
band/3	19/0.012/2.13e-07	15/0.002/6.71e-07	18/0.002/9.14e-07
bard/3	101/0.028/3.00e-07	698/0.097/4.95e-07	811/0.112/8.16e-07
beale/2	48/0.010/6.38e-07	131/0.007/2.30e-07	96/0.007/8.34e-07
biggs/6	—	—	—
box/3	74/0.014/5.48e-07	203/0.017/1.45e-07	185/0.015/5.45e-07
froth/2	92/0.014/6.85e-07	338/0.025/9.02e-07	381/0.026/8.13e-07
gauss/3	13/0.010/3.75e-07	24/0.004/4.71e-07	21/0.003/3.27e-07
helix/3	102/0.019/4.65e-07	361/0.032/9.10e-07	418/0.036/9.20e-07
jensam/2	39/0.009/2.79e-07	148/0.009/9.70e-07	149/0.009/8.40e-07
kowosb/4	224/0.029/9.98e-07	1029/0.075/3.25e-07	797/0.056/9.64e-07
lin/100	13/0.062/8.86e-07	13/0.054/8.86e-07	13/0.054/8.86e-07
lin/500	18/0.418/9.57e-07	18/0.420/9.38e-07	18/0.421/9.38e-07
osb2/11	734/0.113/7.42e-07	1513/0.259/4.38e-07	1174/0.198/7.47e-07
pen1/60	78/0.025/4.74e-07	390/0.056/8.20e-07	668/0.079/8.95e-07
pen2/100	151/0.074/7.94e-07	162/0.045/5.89e-07	235/0.081/8.49e-07
rose/2	79/0.011/9.71e-07	584/0.037/9.06e-07	858/0.053/2.86e-07
rosex/100	104/0.022/4.68e-07	1190/0.132/7.73e-07	705/0.080/8.08e-07
rosex/1000	106/0.899/2.49e-07	1040/6.318/8.60e-07	1374/8.180/2.95e-07
sing/4	213/0.023/6.12e-07	—	1111/0.070/9.15e-07
trid/100	80/0.021/4.41e-07	112/0.022/4.22e-07	128/0.024/6.31e-07
trid/200	37/0.015/6.81e-07	46/0.016/7.20e-07	42/0.017/6.19e-07
vardim/8	28/0.011/4.13e-07	26/0.004/6.51e-07	31/0.004/9.75e-07
watson/6	1811/0.500/7.74e-07	—	—
wood/4	173/0.020/6.93e-07	622/0.047/5.42e-07	966/0.069/9.26e-07

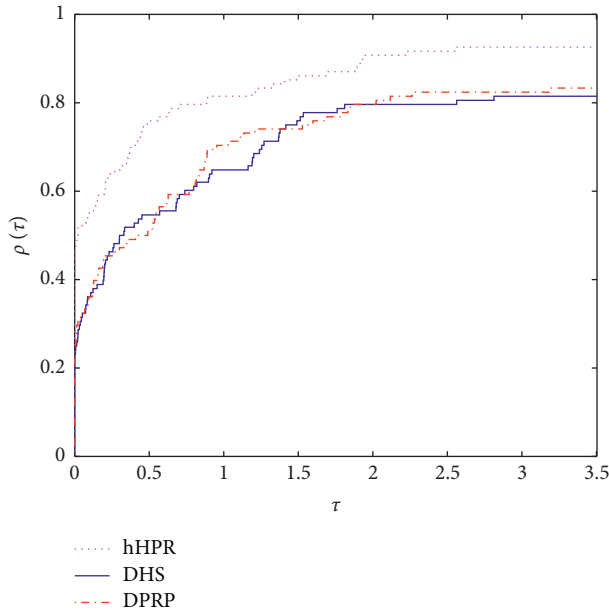


FIGURE 1: Performance profile on Tcpu.

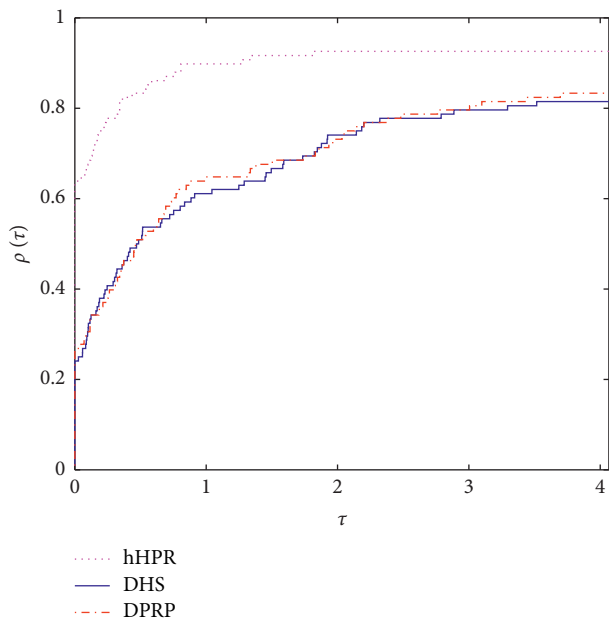


FIGURE 2: Performance profile on Itr.

Data Availability

All the datasets used in this paper are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The corresponding author acknowledges the Natural Science Foundation of Guangxi Province (grant no. 2021GXNSFAA075001).

References

- [1] A. Alhawarat, Z. Salleh, and I. A. Masmali, "A convex combination between two different search directions of conjugate gradient method and application in image restoration," *Mathematical Problems in Engineering*, vol. 2021, Article ID 9941757, 15 pages, 2021.
- [2] A. Alhawarat, G. Alhamzi, I. Masmali, and Z. Salleh, "A descent four-term conjugate gradient method with global convergence properties for large-scale unconstrained optimisation problems," *Mathematical Problems in Engineering*, vol. 2021, Article ID 6219062, 14 pages, 2021.
- [3] I. A. Masmali, Z. Salleh, Z. Salleh, and A. Alhawarat, "A decent three term conjugate gradient method with global convergence properties for large scale unconstrained optimization problems," *AIMS Mathematics*, vol. 6, no. 10, pp. 10742–10764, 2021.
- [4] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, no. 2, pp. 149–154, 1964.
- [5] E. Polak and G. Ribière, "Note sur la convergence de méthodes de directions conjuguées," *Revue française d'informatique et de recherche opérationnelle. Série rouge*, vol. 3, no. 16, pp. 35–43, 1969.
- [6] B. T. Polyak, "The conjugate gradient method in extremal problems," *USSR Computational Mathematics and Mathematical Physics*, vol. 9, no. 4, pp. 94–112, 1969.
- [7] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards*, vol. 49, no. 6, pp. 409–436, 1952.
- [8] Y. Liu and C. Storey, "Efficient generalized conjugate gradient algorithms, part 1: Theory," *Journal of Optimization Theory and Applications*, vol. 69, no. 1, pp. 129–137, 1991.
- [9] R. Fletcher, "Practical methods of optimization," *Unconstrained Optimization*, John Wiley & Sons, New York, NY, USA, 1987.
- [10] Y. H. Dai and Y. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM Journal on Optimization*, vol. 10, no. 1, pp. 177–182, 1999.
- [11] Y. H. Dai and Y. Yuan, *Nonlinear Conjugate Gradient Methods (In Chinese)*, Shanghai Scientific and Technical Publishers, Shanghai, China, 2000.
- [12] J. K. Liu and S. J. Li, "New hybrid conjugate gradient method for unconstrained optimization," *Applied Mathematics and Computation*, vol. 245, pp. 36–43, 2014.
- [13] S. Babaie-Kafaki and R. Ghanbari, "Two hybrid nonlinear conjugate gradient methods based on a modified secant equation," *Optimization*, vol. 63, no. 7, pp. 1027–1042, 2014.
- [14] J. Jian, L. Han, and X. Jiang, "A hybrid conjugate gradient method with descent property for unconstrained optimization," *Applied Mathematical Modelling*, vol. 39, no. 3-4, pp. 1281–1290, 2015.
- [15] M. Sun and J. Liu, "Three modified Polak-Ribière-Polyak conjugate gradient methods with sufficient descent property," *Journal of Inequalities and Applications*, vol. 2015, no. 1, pp. 125–138, 2015.
- [16] I. Arzuka, M. R. Abu Bakar, and W. J. Leong, "A scaled three-term conjugate gradient method for unconstrained optimization," *Journal of Inequalities and Applications*, vol. 2016, no. 1, pp. 325–340, 2016.
- [17] D. Touati-Ahmed and C. Storey, "Efficient hybrid conjugate gradient techniques," *Journal of Optimization Theory and Applications*, vol. 64, no. 2, pp. 379–397, 1990.

- [18] Y. H. Dai and Y. Yuan, "An efficient hybrid conjugate gradient method for unconstrained optimization," *Annals of Operations Research*, vol. 103, pp. 33–47, 2001.
- [19] J. C. Gilbert and J. Nocedal, "Global convergence properties of conjugate gradient methods for optimization," *SIAM Journal on Optimization*, vol. 2, no. 1, pp. 21–42, 1992.
- [20] N. Andrei, "Another hybrid conjugate gradient algorithm for unconstrained optimization," *Numerical Algorithms*, vol. 47, no. 2, pp. 143–156, 2008.
- [21] N. Andrei, "Hybrid conjugate gradient algorithm for unconstrained optimization," *Journal of Optimization Theory and Applications*, vol. 141, no. 2, pp. 249–264, 2009.
- [22] S. S. Djordjević, "New hybrid conjugate gradient method as a convex combination of LS and FR methods," *Acta Mathematica Scientia*, vol. 39, pp. 214–228, 2019.
- [23] S. Babaie-Kafaki and R. Ghanbari, "A hybridization of the Hestenes-Stiefel and Dai-Yuan conjugate gradient methods based on a least-squares approach," *Optimization Methods and Software*, vol. 30, no. 4, pp. 673–681, 2015.
- [24] Z. Wei, S. Yao, and L. Liu, "The convergence properties of some new conjugate gradient methods," *Applied Mathematics and Computation*, vol. 183, no. 2, pp. 1341–1350, 2006.
- [25] Z. Dai and F. Wen, "Another improved Wei-Yao-Liu nonlinear conjugate gradient method with sufficient descent property," *Applied Mathematics and Computation*, vol. 218, no. 14, pp. 7421–7430, 2012.
- [26] I. Bongartz, A. R. Conn, N. Gould, and P. L. Toint, "Cute," *ACM Transactions on Mathematical Software*, vol. 21, no. 1, pp. 123–160, 1995.
- [27] J. J. Moré, B. S. Garbow, and K. E. Hillstom, "Testing unconstrained optimization software," *ACM Transactions on Mathematical Software*, vol. 7, no. 1, pp. 17–41, 1981.
- [28] J. V. Burke and A. Engle, "Line search methods for convex-composite optimization," 2018, <https://arxiv.org/abs/1806.05218>.
- [29] B. Sellami, Y. Laskri, and R. Benzine, "A new two-parameter family of nonlinear conjugate gradient methods," *Optimization*, vol. 64, no. 4, pp. 993–1009, 2015.
- [30] E. D. Dolan and J. J. Moré, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, no. 2, pp. 201–213, 2002.