



Qi Tian¹, Xiaoliang Wang^{1,*}, Liping Pang¹, Mingkun Zhang¹ and Fanyun Meng²

- ¹ School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China; tiangi@mail.dlut.edu.cn (Q.T.); lppang@dlut.edu.cn (L.P.); mkzhang@mail.dlut.edu.cn (M.Z.)
- ² School of Information and Control Engineering, Qingdao University of Technology, Qingdao 266520, China; mengfanyun@qut.edu.cn
- * Correspondence: xliangwangdlut@mail.dlut.edu.cn

Abstract: Three-term conjugate gradient methods have attracted much attention for large-scale unconstrained problems in recent years, since they have attractive practical factors such as simple computation, low memory requirement, better descent property and strong global convergence property. In this paper, a hybrid three-term conjugate gradient algorithm is proposed and it owns a sufficient descent property, independent of any line search technique. Under some mild conditions, the proposed method is globally convergent for uniformly convex objective functions. Meanwhile, by using the modified secant equation, the proposed method is also global convergence without convexity assumption on the objective function. Numerical results also indicate that the proposed algorithm is more efficient and reliable than the other methods for the testing problems.

Keywords: three-term conjugate gradient method; sufficient descent property; secant equation; conjugate condition; global convergence; acceleration strategy

1. Introduction

In this paper, we consider the following unconstrained problem:

$$\min_{\boldsymbol{c}\in \boldsymbol{R}^n} f(\boldsymbol{x}),\tag{1}$$

where function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and bounded below. There are many methods for solving (1). such as the Levenberg–Marquardt methods [1], Newton methods [2] and quasi-Newton methods [3,4]. However, these methods are efficient for small and medium-sized problems and are not suitable for large scale problems in terms of the storage of a matrix for second order information or its approximation. Conjugate gradient (CG) methods [4–12] are much more effective for unconstrained problems, especially for large-scale cases by low memory requirements and strong convergence properties [6,8–11], etc. Meanwhile, CG methods have been applied to image restoration problems, optimal control problems and optimal problems in machine learning [13–15], etc. In this paper, we design a CG method for (1).

The nonlinear CG method was first proposed by Hestenes and Stiefel [16] for linear equations Ax = b. In 1964, Fletcher and Reeves [17] extended the CG method in [16] to unconstrained optimization problems. After that, many researchers proposed various CG methods [6–10,12,18]. In CG methods, a sequence of iterative point $\{x_k\}$ is generated by an initial point x_0 and:

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

where α_k is the step size which is determined by some line search technique and d_k is the search direction. In a traditional CG method, the direction is usually defined by



Citation: Tian, Q.; Wang, X.; Pang, L.; Zhang, M.; Meng, F. A New Hybrid Three-Term Conjugate Gradient Algorithm for Large-Scale Unconstrained Problems. *Mathematics* 2021, *9*, 1353. https://doi.org/ 10.3390/math9121353

Academic Editor: Ioannis K. Argyros

Received: 9 May 2021 Accepted: 8 June 2021 Published: 11 June 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/).



$$d_{k} = \begin{cases} -g_{k}, & \text{if } k = 0, \\ -g_{k} + \beta_{k} d_{k-1}, & \text{if } k \ge 1. \end{cases}$$
(3)

Different conjugate parameter β_k generates a different CG method which may be significantly different in theoretical properties and numerical performance. The Hestenes–Stiefel (HS) method [16] and Polak–Ribière–Polyak (PRP) method [19,20] have nice numerical performance and their conjugate parameters are:

$$\beta_{k+1}^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}, \quad \beta_{k+1}^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2},$$

where $g_k = \nabla f(x_k)$, $g_{k+1} = \nabla f(x_{k+1})$ and $y_k = g_{k+1} - g_k$. Note that the HS method automatically satisfies the conjugate condition $d_{k+1}^T y_k = 0$, independently of any line search technique. Data and Liao [18] extended the above conjugate condition to:

$$d_{k+1}^T y_k = -t g_{k+1}^T s_k, \quad \forall \ k \ge 1.$$
 (4)

where $t \ge 0$ and $s_k = x_{k+1} - x_k$. The new condition (4) gives a more accurate approximation for the Hessian matrix of the original objective function. Based on the condition (4), Dai and Liao [18] presented a new conjugate parameter:

$$\beta_{k+1}^{DL}(t) = \frac{g_{k+1}^T(y_k - ts_k)}{y_k^T d_k} = \frac{g_{k+1}^T y_k}{y_k^T d_k} - t \frac{g_{k+1}^T s_k}{y_k^T d_k}.$$
(5)

In order to have global convergence, they selected the non-negative conjugate parameter, meaning that:

$$\beta_{k+1}^{DL+}(t) = \max\{0, \beta_{k+1}^{DL}(t)\}.$$

Under some mild conditions, the global convergence was established. However, the selection of the parameter t strongly affects the numerical performance, thus many scholars have focused on the choices for the parameter t, as can be seen in [21–26] etc.

Compared with the traditional two-term CG method, three-term CG methods [27–31] always have good numerical performance and nice theoretical properties, such as the sufficient descent property, independently of the accuracy of line search, i.e., it is always holds that:

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

where c > 0. Specifically, Zhang et al. [29] proposed a descent three-term PRP CG method in which the direction has the form:

$$d_0 = -g_0, \quad d_{k+1} = -g_{k+1} + \beta_{k+1}^{PRP} d_k + \delta_{k+1}^{PRP} y_k, \quad \delta_{k+1}^{PRP} = -\frac{g_{k+1}^T d_k}{\|g_k\|^2}, \quad k \ge 0.$$
(6)

and Zhang et al. [28] presented a descent three-term HSCG method in which the direction is:

$$d_0 = -g_0, \quad d_{k+1} = -g_{k+1} + \beta_{k+1}^{HS} d_k + \delta_{k+1}^{HS} y_k, \quad \delta_{k+1}^{HS} = -\frac{g_{k+1}^I d_k}{d_k^T y_k}, \quad k \ge 0.$$
(7)

and Babaie-Kafaki and Ghanbari [31] gave a modified three-term HS/DL method in which the direction owns the form:

$$d_0 = -g_0, \quad d_{k+1} = -g_{k+1} + \beta_{k+1}^{HS} d_k - t \frac{g_{k+1}^T s_k}{|y_k^T d_k|} d_k + \delta_{k+1}^{HS} y_k, \quad \delta_{k+1}^{HS} = -\frac{g_{k+1}^T d_k}{d_k^T y_k}, \quad k \ge 0.$$
(8)

For the above three directions, the sufficient descent property is always satisfied, i.e., $g_k^T d_k \leq -||g_k||^2$. Note that the sufficient descent property is stronger that the descent property and may greatly improve the numerical performance of the corresponding methods.

Motivated by the above discussions, in this paper, we propose a new descent hybrid three-term CG algorithm. Under some mild conditions, the direction in this descent hybrid three-term CG algorithm may reduce the directions in [28,29,31], respectively, and another new three-term direction which also satisfies the sufficient descent property with c = 1 (which is why we call our method as the hybrid three-term CG method). The new method owns the sufficient descent property independent of the accuracy of the line search technique. Under some mild conditions, the global convergence is established for uniformly convex objective functions. For general functions without convexity assumption, the global convergence is also established by using the modified secant condition in [32]. Numerical results indicate that the proposed algorithm is effective and reliable.

The paper is organized as follows. In Section 2, we firstly present the motivation for the hybrid three-term CG method and then propose the new hybrid three-term direction and prove some properties of the new direction and give the global convergence for uniformly convex objective functions at last. In Section 3, the global convergence for the general nonlinear functions is established with the help of the modified secant condition. Numerical tests are given in Section 4 to show the efficient and reliable nature of the proposed algorithm. Finally, the conclusions are presented in Section 5.

2. Motivation and Algorithm

In this section, we firstly present the motivation and give the form of the new direction. It should be noted that if the exact line search technique is adopted, which implies $g_{k+1}^T d_k = 0$, then it holds that:

$$\beta_{k+1}^{HS} = \beta_{k+1}^{PRP} = \beta_{k+1}^{DL}(t).$$

If the inexact line search technique is adopted, these three methods may be different in theoretical property and numerical performance and the HS method and DL method may be not well defined (the denominator $y_k^T d_k$ may be 0). Zhang [33] present a hybrid conjugate parameter β_{k+1}^{hybrid} for the traditional two-term Dai–Liao CG method:

$$eta_{k+1}^{hybrid} = rac{g_{k+1}^T(y_k - ts_k)}{\max\{y_k^T d_k, \|g_k\|^2\}},$$

The numerical results for general nonlinear equations show that the hybrid two-term conjugate residual method is effective and reliable.

Motivated by the above discussions and the nice properties of three-term CG methods, in the following, we propose a new hybrid descent three-term direction which has the following form: $d_0 = -g_0$ and:

$$d_{k+1}^{N} = -g_{k+1} + \beta_{k+1}^{N} \mathbf{s}_{\mathbf{k}} - \delta_{k+1}^{N} y_{k},$$
(9)

where:

$$\beta_{k+1}^{N} = \frac{g_{k+1}^{T}(y_{k} - ts_{k})}{\max\{\mathbf{y}_{\mathbf{k}}^{T}\mathbf{s}_{\mathbf{k}}, \|g_{k}\|^{2}\}}, \quad \delta_{k+1}^{N} = \frac{g_{k+1}^{T}s_{k}}{\max\{\mathbf{y}_{\mathbf{k}}^{T}\mathbf{s}_{\mathbf{k}}, \|g_{k}\|^{2}\}}.$$

Note that the direction d_{k+1}^N is well defined. In fact, if $y_k^T s_k = ||g_k||^2 = 0$ holds, the condition $||g_k||^2 = 0$ indicates that the method stops and the optimal solution (x_k) is obtained.

In the following, we give some remarks for the above direction. Note that if t = 0 and $y_k^T s_k \ge ||g_k||^2$ hold, the direction d_{k+1}^N reduces the direction d_{k+1}^{TTHS} in [28], and t = 0 and $y_k^T s_k \le ||g_k||^2$ hold, the direction d_{k+1}^N reduces to the direction d_{k+1}^{TTPRP} in [29]. Note also that if $y_k^T s_k \ge ||g_k||^2$ holds, then the parameter β_{k+1}^N reduces to the conjugate parameter

 β_{k+1}^{DL} and the direction d_{k+1}^N reduces to a modified vision of the direction d_{k+1}^{TDL} in [31]. If $y_k^T s_k < \|g_k\|^2$ holds, the direction d_{k+1}^N reduces to a new three-term direction which also satisfies the sufficient descent property with c = 1. Overall, we regard the direction d_{k+1}^N as the hybrid direction of the HS direction, the Dai–Liao direction and the PRP direction.

2.1. Algorithm for Uniformly Convex Functions

Now, based on the above analyses, we state the steps of our algorithm as follows:

| Algorithm 1: New hybrid three-term conjugate gradient method (HTTCG). | | | | |
|---|---|--|--|--|
| 1 | Step 0 . Select the initial point $x_0 \in \mathbb{R}^n$. Compute $g(x_0)$ and set $d_0 = -g_0$. | | | |
| | Let $k := 0$. | | | |
| 2 | Step 1 . If $ g_k \leq \varepsilon$, then stop, otherwise go to next step; | | | |
| 3 | Step 2 . Compute the step size α_k along the direction d_k by the line search strategy | | | |
| | technique; | | | |

- 4 **Step 3**. Let $x_{k+1} = x_k + \alpha_k d_k$;
- **5 Step 4**. Compute the search direction d_{k+1} by (9);
- **6 Step 5**. Set k := k + 1 and go to Step 1.

Remark 1. Note that in Algorithm 1, the line search technique is not explicitly given. In fact, any line search technique is accepted.

In the following, we show that Algorithm 1 owns the sufficient descent property independent of any line search technique.

Lemma 1. For any line search technique, the sequence $\{d_k^N\}$ is generated by Algorithm 1, and it always holds that:

$$g_k^T d_k^N \le -\|g_k\|^2, \quad \forall k \ge 0.$$
 (10)

Proof. If k = 0, we have $d_0 = -g_0$, then it holds $g_0^T d_0 = -||g_0||^2$. For $k \ge 0$, by the definition of d_{k+1}^N , we have the following inequality:

$$g_{k+1}^{T}d_{k+1}^{N} = -\|g_{k+1}\|^{2} + \frac{g_{k+1}^{T}(y_{k}-ts_{k})}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}}g_{k+1}^{T}s_{k} - \frac{g_{k+1}^{T}s_{k}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}}g_{k+1}^{T}y_{k}$$

$$= -\|g_{k+1}\|^{2} + \frac{g_{k+1}^{T}s_{k}g_{k+1}^{T}y_{k}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}} - \frac{t(g_{k+1}^{T}s_{k})^{2}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}} - \frac{g_{k+1}^{T}s_{k}g_{k+1}^{T}y_{k}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}} - \frac{g_{k+1}^{T}s_{k}g_{k+1}^{T}y_{k}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}} - \frac{g_{k+1}^{T}s_{k}g_{k+1}^{T}y_{k}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}}$$

$$= -\|g_{k+1}\|^{2} - \frac{t(g_{k+1}^{T}s_{k})^{2}}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}}$$

$$\leq -\|g_{k+1}\|^{2},$$
(11)

where the last inequality holds by $t \ge 0$. Then, (10) holds. This completes the proof. \Box

Lemma 1 means that the new direction satisfies the sufficient descent property independent of the line search technique. A conjugate condition also plays an important role in numerical performance. In the HS method, it automatically satisfies the condition $d_{k+1}^T y_k = 0$; in Dai–Liao method, the modified condition $d_{k+1}^T y_k = -tg_{k+1}^T s_k$ is always satisfied. In our part, by the design of the direction d_{k+1}^N , we have:

From (12), it holds that the new direction d_{k+1}^N also satisfies the DL conjugate condition (4) in an extent form, i.e., $(d_{k+1}^N)^T y_k \leq -t_1 g_{k+1}^T s_k$ where $t_1 = \frac{ty_k^T s_k + ||y_k||^2}{\max\{y_k^T s_k, ||g_k||^2\}}$. In

fact, if we adopt the line search technique which results in $y_k^T s_k \ge 0$, then it holds that $t_1 = \frac{ty_k^T s_k + ||y_k||^2}{\max\{y_k^T s_k, ||g_k||^2\}} > 0.$

2.2. Convergence for Uniformly Convex Functions

In the following, we present the global convergence analysis of the HTTCG method under the following assumptions.

Assumption 1. The level set $\mathbb{T} := \{x \in \mathbb{R}^n : f(x) \le f(x_0)\}$ is bounded where x_0 is the starting point, namely, there exists a constant X > 0 such that:

$$\|x\| \le X, \quad \forall \ x \in \mathbb{T}.$$
(13)

Assumption 2. In some neighborhood \mathbb{N} of \mathbb{T} , the gradient of function f(x), g(x), is Lipschitz continuous, which means there exists a constant L > 0 such that:

$$|g(x) - g(y)|| \le L ||x - y||, \quad \forall x, y \in \mathbb{N}.$$
(14)

Note that based on Assumptions 1 and 2, there exists a positive constant *G* such that:

$$\|g(x)\| \le G, \quad \forall \ x \in \mathbb{T}.$$
(15)

In the following, we show that the sequence $\{d_k^N\}$ generated by Algorithm 1 is bounded.

Lemma 2. Assume $0 < t \le T$ and Assumptions 1 and 2 hold. For any line search technique, the sequence $\{d_k^N\}$ is generated by Algorithm 1. If the objective function f is uniformly convex on the set \mathbb{N} , then $\{d_k^N\}$ is bounded.

Proof. Since function *f* is uniformly convex on the set \mathbb{N} , then for any $x, y \in \mathbb{N}$, we have

$$(\nabla f(x) - \nabla f(y))^T (x - y) \ge \mu ||x - y||^2,$$

where $\mu > 0$ is the uniformly convex parameter. Especially, if we take $x = x_{k+1}$ and $y = x_k$, then it holds that:

$$y_k^T s_k \ge \mu \|s_k\|^2 > 0. \tag{16}$$

In the following, we prove the boundedness of parameters β_{k+1}^N and δ_{k+1}^N . In fact, by their definitions, we have:

$$\begin{aligned} |\beta_{k+1}^{N}| &= \left| \frac{g_{k+1}^{T}(y_{k}-ts_{k})}{\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}} \right| &\leq \frac{\|g_{k+1}\|(\|y_{k}\|+t\|s_{k}\|)}{|\max\{y_{k}^{T}s_{k},\|g_{k}\|^{2}\}|} \leq \frac{\|g_{k+1}\|(\|y_{k}\|+t\|s_{k}\|)}{y_{k}^{T}s_{k}} \\ &\leq \frac{(L+T)\|s_{k}\|}{\mu\|s_{k}\|^{2}} \|g_{k+1}\| = \frac{L+T}{\mu} \frac{\|g_{k+1}\|}{\|s_{k}\|}, \end{aligned}$$
(17)

$$\begin{aligned} |\delta_{k+1}^{N}| &= \left| \frac{g_{k+1}^{T} s_{k}}{\max\{y_{k}^{T} s_{k} , \|g_{k}\|^{2}\}} \right| &\leq \frac{\|g_{k+1}\| \|s_{k}\|}{|\max\{y_{k}^{T} s_{k} , \|g_{k}\|^{2}\}|} \leq \frac{\|g_{k+1}\| \|s_{k}\|}{y_{k}^{T} s_{k}} \\ &\leq \frac{\|g_{k+1}\| \|s_{k}\|}{\mu \|s_{k}\|^{2}} = \frac{1}{\mu} \frac{\|g_{k+1}\|}{\|s_{k}\|}, \end{aligned}$$
(18)

By the definition of d_{k+1}^N , we have:

$$\begin{aligned} \|d_{k+1}^{N}\| &\leq \|g_{k+1}\| + |\beta_{k+1}^{N}| \|g_{k}\| + |\delta_{k+1}^{N}| \|y_{k}\| \\ &\leq \|g_{k+1}\| + \frac{L+T}{\mu} \frac{\|g_{k+1}\|}{\|g_{k}\|} \|g_{k}\| + \frac{1}{\mu} \frac{\|g_{k+1}\|}{\|g_{k}\|} \|y_{k}\| \\ &\leq \|g_{k+1}\| + \frac{L+T}{\mu} \|g_{k+1}\| + \frac{L}{\mu} \|g_{k+1}\| \\ &= \left(1 + \frac{2L+T}{\mu}\right) \|g_{k+1}\| \leq \left(1 + \frac{2L+T}{\mu}\right) G, \end{aligned}$$
(19)

where the last inequality holds by (15). This completes the proof. \Box

The standard Wolfe line search technique is often referred to in CG methods (see [6,7,11] etc.). It has the following form:

$$\begin{cases} f(x_k + \alpha_k d_k) \le f(x_k) + \sigma_1 \alpha_k g_k^T d_k, \\ g(x_k + \alpha_k d_k)^T d_k \ge \sigma_2 g_k^T d_k, \end{cases}$$
(20)

where $0 < \sigma_1 < \sigma_2 < 1$.

The following lemma plays an essential role for the global convergence theorem of our method. It can be seen in Lemma 3.1 in [34]. Hence, we only state it here and omit its proof.

Lemma 3. Suppose that Assumptions 1 and 2 hold. Consider any iterative method with the form (2) where d_k satisfies the sufficient descent property and α_k is computed by Wolfe line search technique (20). If the following relationship holds:

$$\sum_{k\geq 0} \frac{1}{\|d_k\|^2} = +\infty,$$
(21)

then the method globally converges in the sense that:

$$\liminf_{k \to +\infty} \|g_k\| = 0.$$
⁽²²⁾

Now, we establish the global convergence of Algorithm 1 for the uniformly convex objective functions.

Theorem 1. Suppose that Assumptions 1 and 2 hold. Consider Algorithm 1 in which step size α_k is computed by the line search technique (20). If the objective function f is uniformly convex on set \mathbb{N} , then Algorithm 1 globally converges in the sense that:

$$\lim_{k \to +\infty} \|g_k\| = 0.$$
⁽²³⁾

Proof. From Lemma 1, we have that the direction d_{k+1}^N satisfies the sufficient descent property with c = 1. By the first inequality in the line search technique (20), we have the sequence $\{f(x_k)\}_{k\geq 0}$ is non-increasing and $\{x_k\}_{k\geq 0} \subseteq \mathbb{N}$. By the boundedness of $\{d_k^N\}$ in Lemma 2, we have that (21) holds. Then, (22) holds. Since *f* is uniformly convex, then we have (23). This completes the proof. \Box

3. Convergence for General Nonlinear Functions

In order to achieve the global convergence without convexity assumption for the general function, we adopt the modified secant condition in [32] (similar modified secant conditions can also be founded in [35,36] etc.). Concretely, the modified secant condition is:

$$\nabla^2 f(x_{k+1}) s_k = z_k,$$

where p > 0 and C > 0 and:

$$z_{k} = y_{k} + h_{k} \|g_{k}\|^{p} s_{k}, \quad h_{k} = C + \max\left\{0, -\frac{y_{k}^{T} s_{k}}{\|s_{k}\|^{2}}\right\} \|g_{k}\|^{-p}.$$
(24)

Based on the modified secant condition, we present the direction: $d_0 = -g_0$ and:

$$d_{k+1}^{NN} = -g_{k+1} + \beta_{k+1}^{NN} s_k - \delta_{k+1}^{NN} z_k,$$
(25)

where:

$$\beta_{k+1}^{NN} = \frac{g_{k+1}^T(z_k - ts_k)}{\max\{z_k^T s_k, \|g_k\|^2\}}, \quad \delta_{k+1}^{NN} = \frac{g_{k+1}^T s_k}{\max\{z_k^T s_k, \|g_k\|^2\}}$$

Now, based on the above discussions, we state our algorithm as follows:

Algorithm 2: Hybrid three-term CG method using modified secant condition (HTTCGSC).

- 1 **Step 0**. Select $x_0 \in \mathbb{R}^n$, constants C > 0 and r > 0. Compute $g(x_0)$ and set $d_0 = -g_0$. Let k := 0.
- **2** Step 1. If $||g_k|| \le \varepsilon$, then stop, otherwise go to next step;
- 3 Step 2. Compute the step size *α_k* along the direction *d_k* by the line search technique;
- 4 **Step 3**. Let $x_{k+1} = x_k + \alpha_k d_k$;
- **5** Step 4. Compute the search direction d_{k+1} by (25);
- **6** Step 5. Set k := k + 1 and go to Step 1.

Note also that in Algorithm 2, the line search technique is not explicitly given. Similar to Lemma 1, we also have the following lemma. Here, we omit its proof.

Lemma 4. For any line search technique, the sequence $\{d_k^{NN}\}$ is generated by Algorithm 2, then it always holds that:

$$g_k^T d_k^{NN} \le -\|g_k\|^2, \quad \forall \, k \ge 0.$$
 (26)

Lemma 5. The sequences $\{z_k\}$ and $\{s_k\}$ are generated by Algorithm 2, then for any line search technique, we have:

$$z_k^I s_k \ge C \|g_k\|^p \|s_k\|^2, \tag{27}$$

and:

$$z_k \| \le (2L + CG^p) \| s_k \|.$$
(28)

Proof. In fact, for any line search technique, we consider two cases:

Case (i) $y_k^T s_k \ge 0$ holds. In this case, we have $h_k = C$ and $z_k = y_k + C ||g_k||^p s_k$. Then, we have:

$$z_k^T s_k = y_k^T s_k + C \|g_k\|^p \|s_k\|^2 \ge C \|g_k\|^p \|s_k\|^2.$$
⁽²⁹⁾

Case (ii) $y_k^T s_k < 0$ holds. In this case, it holds that $h_k = C - \frac{y_k^T s_k}{\|s_k\|^2} \|g_k\|^{-p}$ and $z_k = y_k + C \|g_k\|^p s_k - \frac{y_k^T s_k}{\|s_k\|^2} s_k$. Then, we have:

$$z_k^T s_k = y_k^T s_k + C \|g_k\|^p \|s_k\|^2 - y_k^T s_k \ge C \|g_k\|^p \|s_k\|^2.$$
(30)

Based on the above discussions, we have that for any line search technique, (27) always holds.

By the definition of z_k , we have:

$$|z_k\| \le \|y_k\| + |h_k| \|g_k\|^p \|s_k\| \le L \|s_k\| + \left(C + \frac{|y_k^T s_k|}{\|s_k\|^2} \|g_k\|^{-p}\right) \|g_k\|^p \|s_k\| \le (2L + CG^p) \|s_k\|,$$

where the last inequality holds by $||g_k|| \le G$. Then, (28) holds. This completes the proof. \Box

In the following, we assume that the limit (22) does not hold, otherwise Algorithm 2 converges. This means that there exists a positive constant η such that:

$$\|g_k\| \ge \eta, \quad \forall \, k \ge 0. \tag{31}$$

Now, we give the global convergence of Algorithm 2 for general nonlinear problems without convex assumption.

Theorem 2. Suppose that Assumptions 1 and 2 hold. Consider Algorithm 2 in which step size α_k is computed by line search technique (20). Then, Algorithm 2 globally converges in the sense that (22) holds.

Proof. We proceed by contradiction. Suppose that (22) is not true. Then, the inequality (31) holds.

In the following, we firstly prove that the sequence $\{d_k^{NN}\}_{k\geq 0}$ is bounded. Similarly, due to the analyses in (17) and (18), we have that:

$$\begin{aligned} |\beta_{k+1}^{NN}| &= \left| \frac{g_{k+1}^T(z_k - ts_k)}{\max\{z_k^T s_k, \|g_k\|^2\}} \right| &\leq \frac{\|g_{k+1}\|(\|z_k\| + t\|s_k\|)}{z_k^T s_k} \\ &\leq \frac{(2L + CG^p + T)\|s_k\|}{C\|g_k\|^p\|s_k\|^2} \|g_{k+1}\| &\leq \frac{2L + CG^p + T}{C\eta^p} \frac{\|g_{k+1}\|}{\|s_k\|}, \end{aligned}$$
(32)

where the third inequality holds by Assumption 1 and (28), the last inequality holds by $||g_k|| \ge \eta$.

$$\begin{aligned} |\delta_{k+1}^{NN}| &= \left| \frac{g_{k+1}^T s_k}{\max\{z_k^T s_k, \|g_k\|^2\}} \right| &\leq \frac{\|g_{k+1}\| \|s_k\|}{\max\{z_k^T s_k, \|g_k\|^2\}|} \leq \frac{\|g_{k+1}\| \|s_k\|}{z_k^T s_k} \\ &\leq \frac{\|g_{k+1}\| \|s_k\|}{C\|g_k\|^p \|s_k\|^2} \leq \frac{1}{C\eta^p} \frac{\|g_{k+1}\|}{\|s_k\|}, \end{aligned}$$
(33)

where the last inequality holds by $||g_k|| \ge \eta$. Then, by (25), it holds that:

$$\begin{aligned} \|d_{k+1}^{NN}\| &\leq \|g_{k+1}\| + |\beta_{k+1}^{NN}| \|s_{k}\| + |\delta_{k+1}^{NN}| \|z_{k}\| \\ &\leq \|g_{k+1}\| + \frac{2L + CG^{p} + T}{C\eta^{p}} \frac{\|g_{k+1}\|}{\|s_{k}\|} \|s_{k}\| + \frac{1}{C\eta^{p}} \frac{\|g_{k+1}\|}{\|s_{k}\|} \|z_{k}\| \\ &\leq \|g_{k+1}\| + \frac{2L + CG^{p} + T}{C\eta^{p}} \|g_{k+1}\| + \frac{2L + CG^{p}}{C\eta^{p}} \|g_{k+1}\| \\ &= \left(1 + \frac{4L + 2CG^{p} + T}{C\eta^{p}}\right) \|g_{k+1}\| \leq \left(1 + \frac{4L + 2CG^{p} + T}{C\eta^{p}}\right) G. \end{aligned}$$
(34)

That means the sequence $\{d_k^{NN}\}_{k\geq 0}$ is bounded. Hence, (21) holds. From Lemmas 3 and 4, we have $\liminf_{k\to+\infty} ||g_k|| = 0$, which contradicts (31). Then, (22) holds. This completes the proof. \Box

4. Numerical Results

In this section, we firstly present the numerical performance of Algorithm 2 and compare it with other methods in [28,31]. Then, an accelerated technique is applied in our method.

4.1. Numerical Performance of Algorithm 2

In this subsection, we focus on the numerical performance of Algorithm 2 and compare it with the MTTDLCG method in [31] and the MTTHSCG method in [28]. For the MTTDLCG method and the MTTHSCG method, we take their settings for parameters. For the value of parameter *t*, authors in [37] point out that $t = \frac{\|y_k\|^2}{y_k^T s_k}$ is a good choice for the Dai–Liao method and the authors in [18] suggest t = 0.1 is a good choice. In this paper, we take $t = \max\{0.1, \frac{\|z_k\|^2}{\max\{z_k^T s_k, \|g_k\|^2\}}\}$.

We execute the tests on a personal computer with the Windows 10 operating system, AMD CPU with @2.1 GHz, 16.00 GB of RAM. Meanwhile, the corresponding codes are written in MATLAB R2016b. The parameters referred are that: $\sigma_1 = 0.20$, $\sigma_2 = 0.85$, C = 0.1, r = 1 if $||s_k||^2 < 1$, r = 3 otherwise. In the following, we present the stopping rules:

Stopping rules (Himmeblau rule): During the testing, if $|f(x_k)| > \epsilon_1$, set $\Delta = \frac{|f(x_k) - f(x_{k+1})|}{|f(x_k)|}$, or let $\Delta = |f(x_k) - f(x_{k+1})|$. Testing stops if $||g(x)|| < \epsilon$ or Tem $< \epsilon_2$ holds.

The values of these parameters are $\varepsilon_1 = \varepsilon_2 = 10^{-5}$ and $\epsilon = 10^{-6}$. Meanwhile, the testing also stops if the total iteration number is larger than 10,000.

For the step size, α_k will be chosen when the search number of the WWP line search is more than 6. Testing problems with initial points considered here are from [38], which are listed in Table 1. Meanwhile, for each problem, ten large-scale dimensions with 1500, 3000, 60000, 7500, 9000, 15,000, 30,000, 60,000, 75,000 and 90,000 variables are considered.

| N0 | Problem | N0 | Problem |
|-----------|---|-----------|---------------------------|
| 1 | Extended Trigonometric Function | 26 | BDQRTIC (CUTE) |
| 2 | Extended Rosenbrock Function | 27 | ARWHEAD (CUTE) |
| 3 | Extended White and Holst Function | 28 | NONDIA (Shanno-78) (CUTE) |
| 4 | Extended Beale Function U63 (MatrixRom) | 29 | DQDRTIC (CUTE) |
| 5 | Extended Penalty Function | 30 | EG2 (CUTE) |
| 6 | Raydan 1 Function | 31 | DIXMAANA (CUTE) |
| 7 | Raydan 2 Function | 32 | DIXMAANB (CUTE) |
| 8 | Diagonal 3 Function | 33 | DIXMAANC (CUTE) |
| 9 | Generalized Tridiagonal-1 Function | 34 | DIXMAANE (CUTE) |
| 10 | Extended Tridiagonal-1 Function | 35 | Broyden Tridiagonal |
| 11 | Extended Three Exponential Terms | 36 | EDENSCH Function (CUTE) |
| 12 | Generalized Tridiagonal-2 Function | 37 | VARDIM Function (CUTE) |
| 13 | Diagonal 4 Function | 38 | DIAGONAL 6 |
| 14 | Diagonal 5 Function (MatrixRom) | 39 | DIXMAANF (CUTE) |
| 15 | Extended Himmelblau Function | 40 | DIXMAANG (CUTE) |
| 16 | Generalized PSC1 Function | 41 | DIXMAANH (CUTE) |
| 17 | Extended PSC1 Function | 42 | DIXMAANI (CUTE) |
| 18 | Extended Maratos Function | 43 | DIXMAANJ (CUTE) |
| 19 | Extended Cliff Function | 44 | DIXMAANK (CUTE) |
| 20 | Extended Wood Function | 45 | DIXMAANL (CUTE) |
| 21 | Extended Quadratic Penalty QP1 Function | 46 | DIXMAAND (CUTE) |
| 22 | Extended Quadratic Penalty QP2 Function | 47 | ENGVAL1 (CUTE) |
| 23 | A Quadratic Function QF2 | 48 | COSINE (CUTE) |
| 24 | Extended EP1 Function | 49 | Extended DENSCHNB (CUTE) |
| 25 | Extended Tridiagonal-2 Function | 50 | Extended DENSCHNF (CUTE) |

Table 1. The testing problems.

To approximately assess the corresponding numerical performances, the performance profile introduced by Dolan and Moré in [39] is adopted. That is, for each method, we plot the fraction *P* of the testing problems for which the method is within a factor τ of the best time. The left side of the figure presents the percentage of the test problems for which a method is the fastest; the right side gives the percentage of the test problems that are successfully solved by each of the methods. Figure 1 presents the performance profile of the three methods in iterations.

From Figure 1, we have that Algorithm 2 (the HTTCG method) in 53%, the MTTDLCG method in 35% and the MTTHSCG method in 41% solve the test problems with the least iteration number. This indicates that Algorithm 2 performs best. Figure 2 presents the performance profile of the three methods in the function-gradient number case:

From Figure 2, we have that Algorithm 2 at 69%, the MTTDLCG method at 14% and the MTTHSCG method at 22% solve the test problems with the least number of computing functions and gradients. This also indicates that Algorithm 2 performs best. Figure 3 presents the performance profile of the three methods in CPU time consumed.

From Figure 3, we have that Algorithm 2 at 12%, the MTTDLCG method at 31% and the MTTHSCG method at 46% solve the test problems with the least CPU time consumed. This indicates that the MTTHSCG method performs best in terms of CPU time consumed. From

Figures 1–3, we have that Algorithm 2 is effective and comparable with the MTTDLCG method and the MTTHSCG method for the test problems in Table 1.



Figure 1. Performance profiles of the numerical results for these methods in number of iterations case.



Figure 2. Performance profiles of the numerical results for these methods in a number of function and gradient evaluations.



Figure 3. Performance profiles of the numerical results for these methods in CPU time consumed case.

4.2. Accelerated Strategy for Algorithm 2

In order to improve the numerical performances of Algorithm 2, in this subsection, we utilize the acceleration strategy in [40], which modifies the step in a multiplicative manner along iterations. Concretely, the iterative form (2) reduces to:

$$x_{k+1} = x_k + \rho_k \alpha_k d_k, \tag{35}$$

We also test the problems in Table 1 and compare Algorithm 2 with the acceleration strategy with Algorithm 2. The corresponding parameters remain unchanged. The performance profiles can be founded in Figures 4–6.



Figure 4. Performance profiles of Algorithm 2 and the acceleration form in NI case.



Figure 5. Performance profiles of Algorithm 2 and the acceleration form in NFG case.



Figure 6. Performance profiles of Algorithm 2 and the acceleration form in CPU Time case.

From Figure 4, we find that the Algorithm 2 with the acceleration strategy (HTTCG-A method) in 73% and HTTCG method in 36% solve the testing problems with the least iteration number. From Figure 5, we see that the HTTCG-A method in 72% and HTTCG method in 35% solve the testing problems with the least number of computing functions and gradient. From Figure 6, we see that the HTTCG-A method in 59% and HTTCG method in 28% solve the testing problems with the least CPU time consumed. These all

indicate that the acceleration strategy works and can reduce the number of iterations, the number of computing functions and gradients and the time consumed.

5. Conclusions

Unconstrained smooth optimization problems can be found in many problems such as optimal control problems and machine learning problems, etc. In this paper, a hybrid three-term descent conjugate gradient algorithm is proposed. This hybrid three-term conjugate gradient algorithm owns the sufficient descent property independent of any line search technique. Meanwhile, it also satisfies the extent Dai–Liao conjugate condition. Under some mild conditions, this algorithm is globally convergent for the uniformly convex functions. For general nonlinear function, the hybrid method is also globally convergent by using some modified secant conditions. Numerical results indicate that the hybrid method is effective and reliable. Meanwhile, an acceleration strategy is adopted to improve the numerical performances. In the future, we will apply our conjugate gradient methods to some non-smooth problems by smoothing strategy and Moreau–Yosida regularization technique and to image restoration problems.

Author Contributions: Conceptualization, Q.T., X.W., L.P. and M.Z.; methodology, Q.T. and X.W.; software, X.W.; validation, X.W., L.P., M.Z. and F.M.; formal analysis, X.W., Q.T. and F.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by the Natural Science Foundation of Shandong Province with No. ZR2019BA014.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data can be found in the manuscript.

Acknowledgments: The authors would like to thank Gonglin Yuan of School of Mathematics and Information Science at Guangxi University for their assistance in the numerical experiment. The authors are grateful to the referees and the editor for their constructive comments and helpful suggestions which much improve the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

- 1. Fan, J.; Zeng, J. A Levenberg–Marquardt algorithm with correction for singular system of nonlinear equations. *Appl. Math. Comput.* **2013**, 219, 9438–9446. [CrossRef]
- Yerina, M.; Izmailov, A. The Gauss–Newton method for finding singular solutions to systems of nonlinear equations. *Comp. Math. Math. Phys.* 2007, 47, 748–759. [CrossRef]
- 3. Yuan, G.; Wei, Z.; Wang, Z. Gradient trust region algorithm with limited memory BFGS update for nonsmooth convex minimization. *Comput. Optim. Appl.* **2013**, *54*, 45–64. [CrossRef]
- Yuan, G.; Wei, Z.; Lu, X. Global convergence of BFGS and PRP methods under a modified weak Wolfe–Powell line search. *Appl. Math. Model.* 2017, 47, 811–825. [CrossRef]
- 5. Yuan, G.; Lu, J.; Wang, Z. The PRP conjugate gradient algorithm with a modified WWP line search and its application in the image restoration problems. *Appl. Numer. Math.* **2020**, *152*, 1–11. [CrossRef]
- 6. Dai, Y.; Han, J.; Liu, G.; Sun, D.; Yin, H.; Yuan, Y. Convergence properties of nonlinear conjugate gradient methods. *SIAM J. Optim.* **2000**, *10*, 345–358. [CrossRef]
- Yuan, G.; Wang, X.; Sheng, Z. Family weak conjugate gradient algorithms and their convergence analysis for nonconvex functions. *Numer. Algorithms* 2020, *84*, 935–956. [CrossRef]
- 8. Hager, W.; Zhang, H. A survey of nonlinear conjugate gradient methods. Pac. J. Optim. 2006, 2, 35–58.
- 9. Andrei, N. Numerical comparison of conjugate gradient algorithms for unconstrained optimization. *Stud. Inform. Control.* **2007**, *16*, 333–352.
- 10. Li, X.; Wang, X.; Sheng, Z.; Duan, X. A modified conjugate gradient algorithm with backtracking line search technique for large-scale nonlinear equations. *Int. J. Comput. Math.* **2018**, *95*, 382–395. [CrossRef]
- 11. Wang, X.; Hu, W.; Yuan, G. A Modified Wei-Yao-Liu Conjugate Gradient Algorithm for Two Type Minimization Optimization Models; Springer: Cham, Switzerland, 2018.

- 12. 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.* **2016**, *168*, 129–152. [CrossRef]
- 13. Cao, J.; Wu, J. A conjugate gradient algorithm and its applications in image restoration. *Appl. Numer. Math.* **2020**, 152, 243–252. [CrossRef]
- 14. Møller, F. A scaled conjugate gradient algorithm for fast supervised learning. Neural Netw. 1993, 6, 525–533.
- 15. Zhou, Y.; Wu, Y.; Li, X. A New Hybrid PRPFR Conjugate Gradient Method for Solving Nonlinear Monotone Equations and Image Restoration Problems. *Math. Probl. Eng.* 2020, 2020, 1–13. [CrossRef]
- 16. Hestenes, R.; Stiefel, L. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.* **1952**, *49*, 409–436. [CrossRef]
- 17. Fletcher, R.; Reeves, C.M. Function minimization by conjugate gradients. Comput. J. 1964, 2, 149–154. [CrossRef]
- 18. Dai, Y.; Liao, L. New conjugacy conditions and related nonlinear conjugate gradient methods. *Appl. Math. Opt.* **2001**, *43*, 87–101. [CrossRef]
- 19. Polak, E. The conjugate gradient method in extreme problems. USSR Comput. Math. Math. Phys. 1969, 9, 94–112. [CrossRef]
- 20. Polak, E.; Ribière, G. Note sur la convergence de directions conjugees. Rev. Fr. Informat Rech. Opertionelle 1969, 16, 35–43.
- Zhang, K.; Liu, H.; Liu, Z. A new Dai-Liao conjugate gradient method with optimal parameter choice. *Numer. Funct. Anal. Optim.* 2019, 40, 194–215. [CrossRef]
- 22. Andrei, N. A Dai-Liao conjugate gradient algorithm with clustering of eigenvalues. *Numer. Algorithms* **2018**, *77*, 1273–1282. [CrossRef]
- 23. Babaie-kafaki, S.; Ghanbari, R. The Dai-Liao nonlinear conjugate gradient method with optimal parameter choices. *Eur. J. Oper. Res.* **2014**, 234, 625–630. [CrossRef]
- 24. Zheng, Y.; Zheng, B. Two new Dai-Liao-type conjugate gradient methods for unconstrained optimization problems. *J. Optim. Theory Appl.* **2017**, *175*, 502–509. [CrossRef]
- 25. Peyghami, M.; Ahmadzadeh, H.; Fazli, A. A new class of efficient and globally convergent conjugate gradient methods in the Dai-Liao family. *Optim. Method. Softw.* **2015**, *30*, 843–863. [CrossRef]
- 26. Yuan, G.; Wang, X.; Sheng, Z. The projection technique for two open problems of unconstrained optimization problems. *J. Optim. Theory Appl.* **2020**, *186*, 590–619. [CrossRef]
- Narushima, Y.; Yabe, H.; Ford, J. A three-term conjugate gradient method with sufficient descent property for unconstrained optimization. SIAM J. Optim. 2011, 21, 212–230. [CrossRef]
- 28. Zhang, L.; Zhou, W.; Li, D. Some descent three-term conjugate gradient methods and their global convergence. *Optim. Method. Softw.* **2007**, *22*, 697–711. [CrossRef]
- 29. Zhang, L.; Zhou, W.; Li, D. A descent modified Polak-Ribière-Polyak conjugate gradient method and its global convergence. *IMA J. Numer. Anal.* 2006, 26, 629–640. [CrossRef]
- 30. Andrei, N. A simple three-term conjugate gradient algorithm for unconstrained optimization. *J. Comput. Appl. Math.* 2013, 241, 19–29. [CrossRef]
- 31. Babaie-Kafaki, S.; Ghanbari, R. Two modified three-term conjugate gradient methods with sufficient descent property. *Optim. Lett.* **2014**, *8*, 2285–2297. [CrossRef]
- Li, D.; Fukushima, M. A modified BFGS method and its global convergence in nonconvex minimization. J. Comput. Appl. Math. 2001, 129, 15–35. [CrossRef]
- 33. Zhang, L. A derivative-free conjugate residual method using secant condition for general large-scale nonlinear equations. *Numer. Algorithms* **2020**, *83*, 1277–1293. [CrossRef]
- 34. Sugiki, K.; Narushima, Y.; Yabe, H. Globally convergent three-term conjugate gradient methods that use secant conditions and generate descent search directions for unconstrained optimization. *J. Optim. Theory Appl.* **2012**, *153*, 733–757. [CrossRef]
- 35. Wei, Z.; Li, G.; Qi, L. New quasi-Newton methods for unconstrained optimization problems. *Appl. Math. Comput.* **2006**, 175, 1156–1188. [CrossRef]
- 36. Yuan, G.; Wei, Z. Convergence analysis of a modified BFGS method on convex minimizations. *Comput. Optim. Appl.* **2010**, 47, 237–255. [CrossRef]
- 37. Dai, Y.; Kou, C. A nonlinear conjugate gradient algorithm with an optimal property and an improved wolfe line search. *SIAM J. Optim.* **2013**, *23*, 296–320. [CrossRef]
- 38. Andrei, N. An unconstrained optimization test functions collection. Adv. Model Optim. 2008, 10, 147–161.
- 39. Dolan, E.; Moré, J. Benchmarking optimization software with performance profiles. Math. Program 2002, 91, 201–213. [CrossRef]
- 40. Andrei, N. An acceleration of gradient descent algorithm with backtracking for unconstrained optimization. *Numer. Algorithms* **2006**, *42*, 63–73. [CrossRef]