*Mathematics*

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(2): 2473–2488.
DOI: 10.3934/math.2023128
Received: 05 September 2022
Revised: 05 October 2022
Accepted: 13 October 2022
Published: 04 November 2022

*Research article*

# A new three-term conjugate gradient algorithm with modified gradient-differences for solving unconstrained optimization problems

**Jie Guo**[1] **and Zhong Wan**[2,*]

[1] School of Mathematical Sciences, Changsha Normal University, Changsha 410081, China
[2] School of Mathematics and Statistics, Central South University, Changsha 410083, China

* **Correspondence:** Email: wanmath@163.com; Tel: +18373138549; Fax: +18373138549.

**Abstract:** Unconstrained optimization problems often arise from mining of big data and scientific computing. On the basis of a modified gradient-difference, this article aims to present a new three-term conjugate gradient algorithm to efficiently solve unconstrained optimization problems. Compared with the existing nonlinear conjugate gradient algorithms, the search directions in this algorithm are always sufficiently descent independent of any line search, as well as having conjugacy property. Using the standard Wolfe line search, global and local convergence of the proposed algorithm is proved under mild assumptions. Implementing the developed algorithm to solve 750 benchmark test problems available in the literature, it is shown that the numerical performance of this algorithm is remarkable, especially in comparison with that of the other similar efficient algorithms.

**Keywords:** optimization problems; conjugate gradient method; global convergence; line search; numerical simulation
**Mathematics Subject Classification:** 90C25, 90C30

## 1. Introduction

Since unconstrained optimization problems often arise from scientific computing and mining of big data [1–4], it is valuable to develop efficient numerical algorithms to solve these problems. However, it seems that there is no any algorithm available in the literature which is in commanding position when it is used to solve all the unconstrained optimization problems, compared with other similar algorithms [5–10]. For this reason, many researchers have been studying new numerical methods to solve the unconstrained optimization problems [1, 11].

Mathematically, a unconstrained optimization problem is written as

$$\min f(x), x \in R^n, \tag{1.1}$$

where $f : R^n \to R$ is continuously differentiable such that its gradient function $g$: $R^n \to R^n$ is available. By $g_k$ we denote the gradient vector of $g$ at $x_k$.

Owing to smaller capacity of computation and storage, conjugate gradient methods (CG) are usually used to solve problem (1.1). By CG, the iterative format to generate a sequence of approximate optimal solutions is

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \ldots, \tag{1.2}$$

where $x_0$ is an arbitrarily chosen initial solution, $d_k$ is a search direction to efficiently seek for an optimal solution of problem (1.1), and $\alpha_k > 0$ is a step size found by line search along $d_k$. In general, the search directions in the classical CG methods are given by

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{otherwise}, \end{cases} \tag{1.3}$$

where $\beta_k$ is the so called conjugate parameter, often being computed by the following classical methods [2, 12]:

$$\begin{aligned} \beta_k^{HS} &= \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \quad \beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \quad \beta_k^{PRP} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \\ \beta_k^{CD} &= \frac{\|g_k\|^2}{-d_{k-1}^T g_{k-1}}, \quad \beta_k^{LS} = \frac{g_k^T y_{k-1}}{-d_{k-1}^T g_{k-1}}, \quad \beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}. \end{aligned} \tag{1.4}$$

In (1.4), $y_{k-1} = g_k - g_{k-1}$ when $k \geq 1$. When $\alpha_k$ in (1.2) is the exact step size and problem (1.1) is a strictly convex quadratic minimization problem, the values of all $\beta_k$ in (1.4) are the same. However, for a generic nonlinear objective function, it is often difficult to find the exact step size. Thus, an inexact line search with lower computational cost is generally adopted. For instance, using the strong Wolfe inexact line search, Riahi and Qattan [13] established global convergence theory of the Fletcher-Reeves CG method and proved its property of local linear convergence. Unfortunately, in most cases, when the Armijo-type line search is used to find the step size $\alpha_k$, it is often difficult to establish global convergence of the classical CG methods, where the search direction $d_k$ is not necessarily descent. For this reason, many variants of CG methods have been proposed to overcome the above difficulty. For instance, using a modified Armijo-type line search, an improved spectral conjugate algorithm was developed in [6] and its global convergence was proved. Numerical tests also showed the advantages of this algorithm.

As remarkable extensions of the classical CG methods, three-term CG methods have been attracting extensive research interest [8–10, 14–16]. The first three-term CG method was proposed in [14], which chooses the search directions by

$$d_{k+1} = -g_{k+1} + \beta_k d_k + \gamma_k d_t, \tag{1.5}$$

where $\beta_k = \beta_k^{HS}$ (or $\beta_k^{FR}, \beta_k^{DY}$ etc.), $d_t (t \leq k-1)$ was a restart direction, and

$$\gamma_k = \begin{cases} 0, & \text{if } t = k-1; \\ \dfrac{g_{k+1}^T y_t}{d_t^T y_t}, & \text{if } t < k-1. \end{cases} \tag{1.6}$$

By numerical tests, it was shown [14] that in the third term of (1.5), the automatical restarts of using the gradient information in (1.6) may improve convergence of the algorithm.

Nazareth [15] presented another method of choosing the directions, given by

$$d_{k+1} = -y_k + \frac{y_k^T y_k}{y_k^T d_k} d_k + \frac{y_{k-1}^T y_k}{y_{k-1}^T d_{k-1}} d_{k-1}, \tag{1.7}$$

where $d_{-1} = d_0 = 0$. It was proved that without requirement of the exact line search, the developed algorithm based on (1.7) can maintain finite termination as applied to solve convex quadratic minimization problems.

In [10, 17], two three-term conjugate gradient methods were given by

$$d_{k+1} = -g_{k+1} + \beta_k^{PRP} d_k - \frac{g_{k+1}^T d_k}{g_k^T g_k} y_k, \tag{1.8}$$

and

$$d_{k+1} = -g_{k+1} + \beta_k^{HS} d_k - \frac{g_{k+1}^T d_k}{d_k^T y_k} y_k. \tag{1.9}$$

respectively. Independent of any line search, it was proved that the directions in (1.8) and (1.9) are sufficiently descent. Since (1.8) and (1.9) can reduce to the standard PRP and HS conjugate gradient methods in (1.4) under the exact line search, respectively, they are regarded as two modified versions of the standard CG methods. It is noteworthy that the search directions in the standard PRP and HS conjugate gradient methods are not necessarily descent in general.

In [8, 9, 16], Andrei suggested three descent three-term CG methods, which computed the search directions by the following different formats:

$$d_{k+1} = -\frac{y_k^T s_k}{\|g_k\|^2} g_{k+1} + \frac{y_k^T g_{k+1}}{\|g_k\|^2} s_k - \frac{s_k^T g_{k+1}}{\|g_k\|^2} y_k, \tag{1.10}$$

$$d_{k+1} = -g_{k+1} - \left(\left(1 + \frac{\|y_k\|^2}{y_k^T s_k}\right) \frac{s_k^T g_{k+1}}{y_k^T s_k} - \frac{y_k^T g_{k+1}}{y_k^T s_k}\right) s_k - \frac{s_k^T g_{k+1}}{y_k^T s_k} y_k, \tag{1.11}$$

and

$$d_{k+1} = -g_{k+1} - \left(\left(1 + 2\frac{\|y_k\|^2}{y_k^T s_k}\right) \frac{s_k^T g_{k+1}}{y_k^T s_k} - \frac{y_k^T g_{k+1}}{y_k^T s_k}\right) s_k - \frac{s_k^T g_{k+1}}{y_k^T s_k} y_k. \tag{1.12}$$

All of them satisfy the conjugacy condition, and except for (1.10), the search directions in (1.11) and (1.12) are descent when the Wolfe line search are used. Numerical experiments indicated that the CG method in [8] outperforms the other six algorithms available in the literature.

Recently, Liu et al. [18] constructed two three-term CG methods, specified by:

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T y_k}{\|d_k\|^2} d_k - \frac{g_{k+1}^T d_k}{\|d_k\|^2} y_k, \tag{1.13}$$

and

$$d_{k+1} = -g_{k+1} + \frac{g_{k+1}^T (y_k - d_k)}{\|d_k\|^2} d_k - \frac{g_{k+1}^T d_k}{\|d_k\|^2} y_k, \tag{1.14}$$

respectively. A remarkable property of these search directions is that they were proved to be sufficiently descent under any line search. However, it is unclear whether these directions in (1.13) and (1.14) satisfy any conjugacy condition or not.

Motivated by a need to further improve numerical efficiency of algorithms, we intend to develop a novel three-term CG algorithm such that the search directions may simultaneously possess descent and conjugate properties. Then, using the standard Wolfe inexact line search, we attempt to prove its global and local convergence under appropriate assumptions, and test its numerical performance by solving benchmark test problems.

The remainder of this article is organized as follows. The new three-term CG algorithm is first developed in Section 2. In Section 3, global convergence of this algorithm is proved. Section 4 is devoted to testing of its numerical performance. Conclusions are drawn in the last section.

## 2. Development of a new algorithm

In this section, we state ideas to develop a new algorithm, and then present its framework of computer procedures.

Combining the ideas in [6, 18], we construct the search direction by

$$d_{k+1} = \begin{cases} -g_{k+1}, & \text{if } w_k = 0, \\ -g_{k+1} + \dfrac{g_{k+1}^T(y_k - s_k)}{w_k} s_k - \dfrac{g_{k+1}^T s_k}{w_k} y_k, & \text{otherwise,} \end{cases} \tag{2.1}$$

where $w_k = \max\{\left|s_k^T \bar{y}_k\right|, s_k^T y_k\}$, $d_0 = -g_0$, and

$$\bar{y}_k = \left(I_n - \frac{g_{k+1} g_{k+1}^T}{\|g_{k+1}\|^2}\right) y_k$$

is defined as done in [6]. Clearly, $\bar{y}_k$ can be regarded as a modified difference of gradients. For this reason, we call the proposed CG method in this paper, where the search directions are defined by (2.1), a three-term CG method with modified gradient-differences. In essence, this three-term CG method is an extension of the two-term spectral conjugate gradient method in [6].

We first prove the following property of the search directions in (2.1).

**Proposition 1.** *Let $d_k$ be given by (2.1). Then, for any $k \geq 0$, the following inequality holds:*

$$g_k^T d_k \leq -\|g_k\|^2. \tag{2.2}$$

*Proof.* By definition, when $k = 0$, we have $g_0^T d_0 = -\|g_0\|^2$. When $k > 0$, we have $g_k^T d_k = -\|g_k\|^2$ if $w_{k-1} = 0$; Otherwise, it is true that

$$\begin{aligned} g_k^T d_k &= -g_k^T g_k + \frac{g_k^T (y_{k-1} - s_{k-1})}{w_{k-1}} g_k^T s_{k-1} - \frac{g_k^T s_{k-1}}{w_{k-1}} g_k^T y_{k-1} \\ &= -\|g_k\|^2 - \frac{(g_k^T s_{k-1})^2}{w_{k-1}} \\ &\leq -\|g_k\|^2. \end{aligned} \tag{2.3}$$

Consequently, for any $k \geq 0$,

$$g_k^T d_k \leq -\|g_k\|^2, \tag{2.4}$$

i.e., $d_k$ is always sufficiently descent. □

**Remark 1.** *As pointed out in [19–22], such a sufficiently descent condition like (2.4) plays a critical role in proving global convergence of CG methods.*

Based on the above nice property of the search directions (2.1) in Propositions 1, we come to state a framework of computer procedures for solving unconstrained optimization problems (1.1).

**Algorithm 1. (New three-term conjugate gradient algorithm (NTTCG))**
***Step 0.*** *Take an initial (approximate) solution* $x_0 \in R^n$ *and an initial search direction* $d_0 = -g_0$. *Choose the parameters* $0 < \rho < \sigma < 1$ *used in the line search. The tolerance error is* $\varepsilon \in (0, 1)$. *Set* $k := 0$.
***Step 1.*** *If* $\|g_k\|_\infty \leq \varepsilon$, *then the algorithm stops.*
***Step 2.*** *Determine the step size* $\alpha_k$ *by the following standard Wolfe line search:*

$$\begin{cases} f(x_k + \alpha_k d_k) - f(x_k) \leq \rho \alpha_k g_k^T d_k, \\ g_{k+1}^T d_k \geq \sigma g_k^T d_k. \end{cases} \tag{2.5}$$

***Step 3.*** *Update the solution by* $x_{k+1} := x_k + \alpha_k d_k$. *Compute* $g_{k+1}$, *and compute a search direction* $d_{k+1}$ *given in (2.1).*
***Step 4.*** *Set* $k := k + 1$. *Return to Step 1.*

**Remark 2.** *By Proposition 1, we know that* $d_{k+1}$ *in Step 3 of Algorithm 1 is a sufficiently descent direction at* $x_{k+1}$, *which ensures that the conducted line search in Step 2 of Algorithm 1 stops in finitely many steps [23].*

**Remark 3.** *It follows from (2.4) that the inequalities:*

$$\|g_k\|^2 \leq \|d_k^T g_k\| \leq \|d_k\| \|g_k\|$$

*hold for any* $k \geq 0$. *Thus,* $\|d_k\| \geq \|g_k\|$ *is true for any* $k \geq 0$.

## 3. Conjugacy properties and convergence analysis

In this section, we study conjugacy property of the search directions defined by (2.1), and establish global and local convergence theory of Algorithm 1.

### *3.1. Convex cases*

We first study the conjugacy property of the search directions generated by Algorithm 1 in the case that the objective function in problem (1.1) is convex quadratic.

Specifically, a problem of convex quadratic minimization is written as:

$$\min f(x) = \frac{1}{2} x^T Q x + q^T x + c, \tag{3.1}$$

where $Q \in R^{n \times n}$ is a given positive definite matrix and $q \in R^n$ is a given vector. When Algorithm 1 is applied to solve problem (3.1), we can prove that the search directions in (2.1) have the following property.

**Proposition 2.** *For problem (3.1), let* $d_k$ *be chosen by (2.1). Then, by the exact line search,* $d_{k+1}$ *and* $d_k$ *are conjugate with respect to* $Q$ *for any* $k \geq 0$.

*Proof.* With the exact line search, we have

$$g_{k+1}^T s_k = 0, \tag{3.2}$$

and

$$s_k \bar{y}_k = s_k^T y_k = s_k^T (g_{k+1} - g_k) = s_k^T Q s_k. \tag{3.3}$$

Consequently,

$$\begin{aligned} d_{k+1} &= -g_{k+1} + \frac{g_{k+1}^T (y_k - s_k)}{\max\{\left|s_k^T \bar{y}_k\right|, s_k^T y_k\}} s_k - \frac{g_{k+1}^T s_k}{\max\{\left|s_k^T \bar{y}_k\right|, s_k^T y_k\}} y_k \\ &= -g_{k+1} + \frac{g_{k+1}^T y_k}{\left|s_k^T \bar{y}_k\right|} s_k \\ &= -g_{k+1} + \frac{g_{k+1}^T Q s_k}{s_k^T Q s_k} s_k \\ &= -g_{k+1} + \frac{g_{k+1}^T Q d_k}{d_k^T Q d_k} d_k. \end{aligned} \tag{3.4}$$

Thus, for $k = 0$,

$$d_1^T Q d_0 = g_1^T Q g_0 - \frac{g_1^T Q g_0}{g_0^T Q g_0} g_0^T Q g_0 = 0. \tag{3.5}$$

For $k > 0$, we have

$$d_{k+1}^T Q d_k = -g_{k+1}^T Q d_k + \frac{g_{k+1}^T Q d_k}{d_k^T Q d_k} d_k^T Q d_k = 0. \tag{3.6}$$

In other words, $d_{k+1}$ and $d_k$ are conjugate with respect to $Q$ for any $k \geq 0$. □

**Remark 4.** *The basic idea to derive the search directions (2.1) is to guarantee their sufficiently descent property by appropriately modifying the steepest descent direction $-g_{k+1}$ (see Proposition 1). It is well known that the conjugate directions in the classic conjugate direction method are not necessarily descent. Proposition 2 demonstrates that our search directions also satisfy the so-called conjugacy condition in the case that the objective function is convex quadratic, although it is not true that any two directions (for example, the two directions $d_{k+2}$ and $d_k$) are conjugate with respect to the matrix $Q$, as in the classic conjugate direction method. In one word, compared with majority of the existing three-term conjugate gradient methods in the literature, the search directions (2.1) have an advantage of simultaneously possessing descent and conjugate properties.*

The following result further states global convergence of Algorithm 1 when it is implemented to solve a uniformly convex optimization problem, an extention of the convex quadratic minimization problem (3.1).

**Theorem 1.** *Let $f : R^n \to R$ be twice continuously differentiable and uniformly convex on a level set $\Omega = \{x \in R^n | f(x) \leq f(x_0)\}$, i.e., there exists a positive constant $\mu$ such that for all $x, y \in \Omega$,*

$$(g(x) - g(y))^T (x - y) \geq \mu \|x - y\|^2$$

*holds. Let $\{g_k\}$ be the gradient sequence generated by Algorithm 1. Then,*

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{3.7}$$

*Proof.* For the sake of contradiction, we suppose that there exists a constant $\varepsilon > 0$ such that $\|g_k\| \geq \varepsilon$ for all $k \in N$.

Since the step size satisfies (2.5), the sequence $\{f(x_k)\}$ generated by Algorithm 1 is decreasing, and all the iterative points $x_k$ are in the level set $\Omega$. Since $f : R^n \to R$ is twice continuously differentiable and uniformly convex on $\Omega$, it follows from Steps 2 and 3 in Algorithm 1 that the level set $\Omega$ is a bounded closed convex set, i.e., there exists a positive constant $B > 0$ such that

$$\|x\| \leq B, \forall x \in \Omega. \tag{3.8}$$

In addition, the gradient of $f$ is also Lipschitz continuous on $\Omega$, i.e., there exists a constant $L > 0$ such that for all $x, y \in \Omega$, the following inequality holds:

$$\|g(x) - g(y)\| \leq L\|x - y\|. \tag{3.9}$$

Furthermore, we can prove boundedness of the sequence $\{d_k\}$. Actually, from (3.9) and the uniformly convex property of $f$, we have $\|y_k\| \leq L\|s_k\|$ and $s_k^T y_k \geq \mu\|s_k\|^2$. Thus, by the Cauchy-Schwarz inequality, we have

$$\begin{aligned}
\|d_{k+1}\| &\leq \|g_{k+1}\| + \frac{\|g_{k+1}\|(\|y_k\| + \|s_k\|)\|s_k\|}{s_k^T y_k} + \frac{\|g_{k+1}\|\|s_k\|\|y_k\|}{s_k^T y_k} \\
&\leq \|g_{k+1}\| + \frac{\|g_{k+1}\|(2\|y_k\| + \|s_k\|)\|s_k\|}{\mu\|s_k\|^2} \\
&\leq \|g_{k+1}\| + \frac{2L\|g_{k+1}\|\|s_k\|}{\mu\|s_k\|} + \frac{\|g_{k+1}\|}{\mu} \\
&= \left(1 + \frac{2L+1}{\mu}\right)\|g_{k+1}\|.
\end{aligned} \tag{3.10}$$

Take $M = 1 + \frac{2L+1}{\mu}$. From (3.10) and $\|g_k\| \geq \varepsilon$, it follows that

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} \geq \sum_{k=0}^{\infty} \frac{\varepsilon^2}{M^2} = +\infty. \tag{3.11}$$

Using (3.9) and the second inequality in (2.5), it yields

$$(\sigma - 1)g_k^T d_k \leq (g_{k+1} - g_k)^T d_k \leq \|g_{k+1} - g_k\|\|d_k\| \leq \alpha_k L\|d_k\|^2. \tag{3.12}$$

Consequently,

$$f(x_k) - f(x_{k+1}) \geq -\rho\alpha_k g_k^T d_k \geq \rho\alpha_k\|g_k\|^2 \geq \frac{\rho(1-\sigma)\|g_k\|^4}{L\|d_k\|^2}, \tag{3.13}$$

hence,

$$f(x_0) - \lim_{k\to\infty} f(x_{k+1}) \geq \frac{\rho(1-\sigma)}{L} \sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2}. \tag{3.14}$$

From (3.14) and the boundedness of $f$ on $\Omega$, we know that

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} \leq \frac{L(f(x_0) - \lim_{k\to\infty} f(x_{k+1}))}{\rho(1-\sigma)} < +\infty, \tag{3.15}$$

which contradicts (3.11). The proof is completed. □

We can also prove that Algorithm 1 is R-linearly convergent when the objective function is uniformly convex.

**Theorem 2.** *Suppose that $f : R^n \to R$ is twice continuously differentiable and uniformly convex on the level set $\Omega$, and that the sequence $\{x_k\}$ generated by Algorithm 1 converges to the unique optimal solution $x^*$. Then for all $k > 0$, there exist constants $a > 0$ and $b \in (0, 1)$ such that*

$$\|f(x_k) - f(x^*)\| \leq ab^k. \tag{3.16}$$

*Proof.* Since $f$ is twice continuously differentiable and uniformly convex on $\Omega$, it follows from (3.2)-(3.4) and (3.12) in [24] that there exist constants $\hat{\lambda} > \lambda > 0$, $\hat{\zeta} > \zeta > 0$ such that for all $x \in \Omega$, the following inequalities hold:

$$\zeta\|x - x^*\|^2 \leq \lambda\|g(x)\|^2 \leq f(x) - f(x^*) \leq \hat{\lambda}\|g(x)\|^2 \leq \hat{\zeta}\|x - x^*\|^2. \tag{3.17}$$

Thus, from the first inequality in (2.5), we have

$$\begin{aligned} f(x_{k+1}) - f(x^*) &\leq (f(x_k) - f(x^*)) + \rho\alpha_k g_k^T d_k \\ &\leq (f(x_k) - f(x^*)) - \rho\frac{(1-\sigma)\,\|g_k\|^2}{L\|d_k\|^2}\|g_k\|^2 \\ &\leq (f(x_k) - f(x^*)) - \rho\frac{1-\sigma}{LM^2}\|g_k\|^2 \\ &\leq (f(x_k) - f(x^*)) - \rho\frac{1-\sigma}{\hat{\lambda}LM^2}(f(x_k) - f(x^*)) \\ &= \left(1 - \rho\frac{1-\sigma}{\hat{\lambda}LM^2}\right)(f(x_k) - f(x^*)), \end{aligned} \tag{3.18}$$

where the second inequality follows from (3.12) and (2.2), the third inequality follows from (3.10), and the last inequality follow from (3.17). Consequently,

$$f(x_{k+1}) - f(x^*) \leq \left(1 - \rho\frac{1-\sigma}{\hat{\lambda}LM^2}\right)^{k+1}(f(x_0) - f(x^*)). \tag{3.19}$$

Taking $a = f(x_0) - f(x^*)$ and $b = 1 - \rho\dfrac{1-\sigma}{\hat{\lambda}LM^2}$, the desired result (3.16) has been proved. □

*3.2. Non-convex cases*

For non-convex minimization problems, we can prove that the search directions in (2.1) satisfy an approximate Dai-Liao conjugate condition.

**Proposition 3.** *Suppose that $s_k^T y_k > \left|s_k^T \bar{y}_k\right|$. Then, $d_{k+1}$ in (2.1) satisfies the following approximate Dai-Liao conjugate condition:*

$$d_{k+1}^T y_k = -t_k g_{k+1}^T s_k. \tag{3.20}$$

*Proof.* When $s_k^T y_k > \left|s_k^T \bar{y}_k\right|$, it holds that

$$\begin{aligned} d_{k+1}^T y_k &= -g_{k+1}^T y_k + \frac{g_{k+1}^T (y_k - s_k)}{s_k^T y_k} s_k^T y_k - \frac{g_{k+1}^T s_k}{s_k^T y_k} \|y_k\|^2 \\ &= -\left(1 + \frac{\|y_k\|^2}{s_k^T y_k}\right) g_{k+1}^T s_k \\ &= -t_k g_{k+1}^T s_k, \end{aligned} \tag{3.21}$$

where $t_k = 1 + \dfrac{\|y_k\|^2}{s_k^T y_k} > 0$. The result (3.20) has been proved. □

**Remark 5.** *Although the condition (3.20) in Proposition 3 does not always hold at any iteration, it does not affect global convergence of Algorithm 1. By a simple example, we can show that this condition is often satisfied (see Table 1). In addition, our numerical tests will also show advantages of the search directions given by (2.1).*

**Example 1.** *For the Rosenbrock problem:*

$$\min f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

*Initial point* $x_0 = (-1.2, 1)$.

We implement Algorithm 1 to solve the Rosenbrock problem, and partly present the obtained values of $s_k^T y_k$ and $\left|s_k^T \bar{y}_k\right|$ in Table 1.

**Table 1.** Values of $s_k^T y_k$ and $\left|s_k^T \bar{y}_k\right|$ in different iterations.

| $k$ | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|
| $s_k^T y_k$ | 0.246690 | 0.391931 | 0.320335 | 0.225285 | 0.192523 | 0.555195 |
| $\left|s_k^T \bar{y}_k\right|$ | 0.245609 | 0.391931 | 0.302341 | 0.230323 | 0.192523 | 0.257161 |

In Table 1, it is easy to see that the directions at the 10th, 12th and 15th iterations, the inequality (3.20) holds.

Before stating global convergence of Algorithm 1 in the non-convex case, we first make the following mild assumptions.

**Assumption 1.** *The level set* $\Omega = \{x \in R^n | f(x) \leq f(x_0)\}$ *is bounded, i.e., there exists a positive constant* $B > 0$ *such that (3.8) holds for all* $x \in \Omega$.

**Assumption 2.** *In some neighborhood* $N$ *of* $\Omega$, $f$ *is continuously differentiable and its gradient is Lipschitz continuous. That is to say, there exists a constant* $L > 0$ *such that (3.9) holds for all* $x, y \in N$.

**Theorem 3.** *Let* $\{g_k\}$ *be a gradient sequence generated by Algorithm 1. Suppose that there exists a constant* $\tau > 0$ *such that* $s_k^T y_k \geq \tau$ *for any* $k \geq 1$. *Under Assumptions 1 and 2, it is true that*

$$\liminf_{k \to \infty} \|g_k\| = 0. \tag{3.22}$$

*Proof.* From Assumptions 1 and 2, we have

$$\begin{aligned} \|d_{k+1}\| &\leq \|g_{k+1}\| + \frac{\left|g_{k+1}^T y_k\right|}{s_k^T y_k}\|s_k\| + \frac{\left|g_{k+1}^T s_k\right|}{s_k^T y_k}\|s_k\| + \frac{\left|g_{k+1}^T s_k\right|}{s_k^T y_k}\|y_k\| \\ &\leq \|g_{k+1}\| + \frac{\|g_{k+1}\|\|y_k\|}{\tau}\|s_k\| + \frac{\|g_{k+1}\|\|s_k\|}{\tau}\|s_k\| + \frac{\|g_{k+1}\|\|s_k\|}{\tau}\|y_k\| \\ &\leq \|g_{k+1}\| + \frac{4LB^2\|g_{k+1}\|}{\tau} + \frac{4B^2\|g_{k+1}\|}{\tau} + \frac{4LB^2\|g_{k+1}\|}{\tau} \\ &= \left(1 + \frac{8L+4}{\tau}B^2\right)\|g_{k+1}\|, \end{aligned} \tag{3.23}$$

where the first and second inequalities follow from the Cauchy-Schwarz inequality and $s_k^T y_k \geq \tau$, and the last inequality follows from (3.8) and (3.9).

Similar to the proof of Theorem 1, we can also prove that (3.15) holds under Assumptions 1 and 2. Together with (3.23), it is concluded that (3.22) holds. □

In order to present R-linear convergence of Algorithm 1 in the non-convex case, we need the following assumption:

**Assumption 3.** *(1) $f : R^n \to R$ is twice continuously differentiable.*
*(2) The sequence $\{x_k\}$ generated by Algorithm 1 satisfies $x_k \to x^*$, where $\nabla f(x^*) = 0$ and $\nabla^2 f(x^*)$ is positive definite.*
*(3) There exists a positive constant $\tau$ such that $s_k^T y_k > \tau$ holds for all the sufficiently large $k > 0$.*

**Theorem 4.** *Let $\{x_k\}$ be the sequence generated by Algorithm 1, which converges to a solution $x^*$ satisfying (2) in Assumption 3. Suppose that Assumptions 1 and 2 hold. Then, for the sufficiently large $k > 0$, there exist constants $a > 0$ and $b \in (0, 1)$ such that*

$$\|f(x_k) - f(x^*)\| \leq ab^k. \tag{3.24}$$

*Proof.* From (3) in Assumption 3, we can know that $\|d_k\|$ is bounded for all $k > 0$. Moreover, from (4.1)–(4.3) in [18], it is clear that there exists a neighborhood of $x^*$, denoted by $U(x^*)$, such that (3.17) holds for all $x \in U(x^*)$. The rest of the proof is similar to that of Theorem 2, we omit it here. □

## 4. Numerical tests

In this section, by numerical experiments, we study effectiveness and robustness of Algorithm 1 when it is employed to solve unconstrained optimization problems.

Algorithm 1 (NTTCG) is tested through solution of the 75 benchmark test problems with variable dimensions from 1000 to 10000. These problems are from [25] or CUTE [26]. Its computer codes are written using the language of Fortran 77, and run on a personal computer with a 2.2GHZ CPU processor, 8GB memory and Windows 10 operation system.

To show advantages of our algorithm (NTTCG), we compare it with the other four similar algorithms, including TMRMIL in [18], ISCG in [6], CG_DESCENT in [7] and THREECG in [8]. For all the compared algorithms, the termination condition is $\varepsilon = 10^{-6}$ or the number of iterations exceeds 10,000. In Algorithm 1, $\rho = 0.0001$, $\sigma = 0.01$, and the parameters not mentioned here are consistent with the corresponding literature. We show the numerical performance differences among these five algorithms by the Dolan and Moré performance profiles [27]. Let $S$ be a set of all methods, $P$ be a set of test problems, $n_p$ be the size of the set $P$ and $t_{p,s}$ be the number of iterations or the CPU time needed to solve problem $p \in P$ by method $s \in S$. Then, the performance ratio is computed by $r_{p,s} = \dfrac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}$, and the overall performance of Algorithm $s$ is given by $\rho_s(\tau) = \frac{1}{n_p} size\{p \in P : r_{p,s} \leq \tau\}$. In fact, $\rho_s(\tau)$ is the probability for Algorithm $s$ that a performance ratio $r_{p,s}$ is within a factor $\tau \in R$ of the best possible ratio. The function $\rho_s(\tau)$ is the distribution function for the performance ratio $r_{p,s}$. We report the numerical results in Figures 1 and 2. From Figures 1 and 2 , we can known that our algorithm (NTTCG) performs the best among the five algorithms, either with respect to the number of iterations, or with respect to the elapsed CPU time.

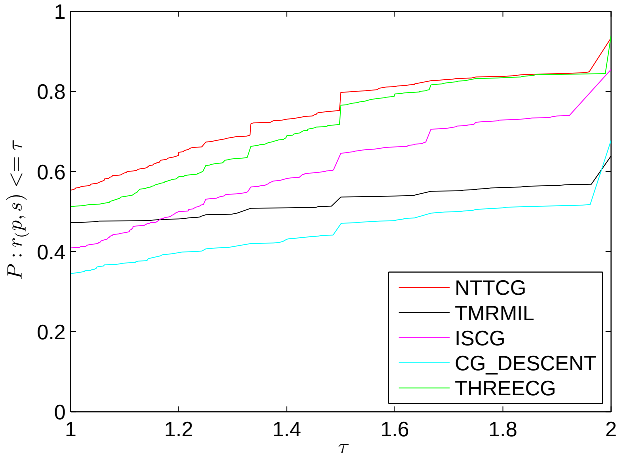


**Figure 1.** Performance profile for the consumed CPU time.

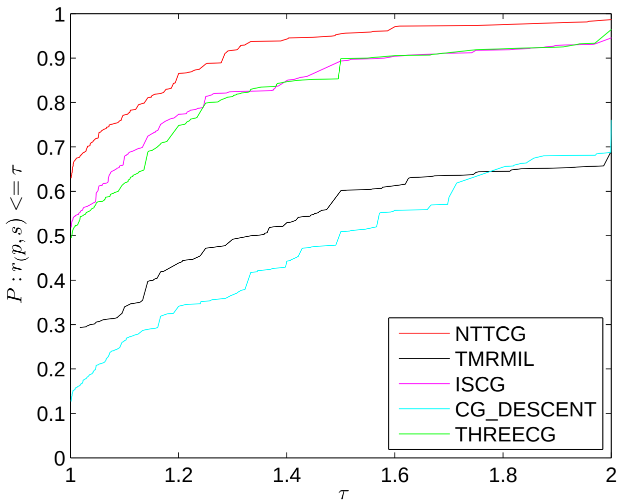


**Figure 2.** Performance profile for the number of iterations.

We also underline good numerical results in Table 3. In Table 3, P, Ni and CPU stand for the number of problems in Table 2, the number of iterations and the consumed CPU time in centisecond (cs), respectively. From the underlined results in Table 3, we know that our algorithm (NTTCG) performs very well in some test problems.

**Table 2.** Benchmark test problems.

| No. | Problem | Dim | No. | Problem | Dim |
|---|---|---|---|---|---|
| 1 | Extended Trigonometric | 7000 | 15 | Extended Quadratic Penalty QP1 3000 | 2000 |
| 2 | Extended Rosenbrock | 10000 | 16 | Extended Tridiagonal 2 | 9000 |
| 3 | Extended White & Holst | 9000 | 17 | BDQRTIC (CUTE) | 3000 |
| 4 | Diagonal 3 | 6000 | 18 | TRIDIA (CUTE) | 8000 |
| 5 | Raydan 1 | 10000 | 19 | NONDIA (CUTE) | 6000 |
| 6 | Diagonal 1 | 9000 | 20 | DQDRTIC (CUTE) | 10000 |
| 7 | Diagonal 2 | 1000 | 21 | DIXMAANC (CUTE) | 10000 |
| 8 | Diagonal 3 | 1000 | 22 | LIARWHD (CUTE) | 9000 |
| 9 | Extended Himmelblau | 8000 | 23 | DIXMAANG (CUTE) | 3000 |
| 10 | Extended Powell | 10000 | 24 | DIXMAANJ (CUTE) | 3000 |
| 11 | Extended Block-Diagonal BD1 | 6000 | 25 | DIXMAANL (CUTE) | 9000 |
| 12 | Extended Maratos | 8000 | 26 | SINQUAD (CUTE) | 9000 |
| 13 | Extended Cliff | 6000 | 27 | BIGGSB1 (CUTE) | 7000 |
| 14 | Quadratic QF1 | 10000 | 28 | Scaled Quadratic SQ2 | 9000 |

**Table 3.** Advantages of Algorithm 1 in numerical tests.

| Problem | NTTCG | TMRMIL | ISCG | CG_DESCENT | THREECG |
|---|---|---|---|---|---|
| | Ni/CPU(cs) | Ni/CPU(cs) | Ni/CPU(cs) | Ni/CPU(cs) | Ni/CPU(cs) |
| 1 | 50/46 | 58/34 | 79/73 | 85/51 | 52/43 |
| 2 | 16/7 | 26/14 | 22/13 | 35/21 | 28/8 |
| 3 | 10/2 | 13/3 | 11/4 | 17/6 | 12/3 |
| 4 | 8/1 | 10/2 | 10/2 | 21/22 | 10/2 |
| 5 | 2/1 | 3/1 | 3/2 | 4/1 | 3/2 |
| 6 | 431/98 | 5835/1899 | 473/204 | F/F | 477/171 |
| 7 | 229/20 | 1372/307 | 230/25 | 231/12 | 231/28 |
| 8 | 43/2 | 69/5 | 45/3 | 44/2 | 45/2 |
| 9 | 198/118 | 5161/1800 | 215/163 | 731/486 | 312/140 |
| 10 | 11/5 | 15/3 | 12/5 | 17/6 | 12/6 |
| 11 | 57/6 | 60/6 | 61/12 | 59/6 | 75/13 |
| 12 | 4/1 | 8/2 | 5/2 | 15/3 | 7/3 |
| 13 | 221/25 | 6552/649 | 223/38 | 245/22 | 225/28 |
| 14 | 7/2 | 10/4 | 8/5 | 17/3 | 9/4 |
| 15 | 18/4 | 33/5 | 39/15 | 41/18 | 34/6 |
| 16 | 92/55 | 476/166 | 106/104 | 8119/2891 | 123/96 |
| 17 | 599/31 | 5452/368 | 601/51 | 600/41 | 601/45 |
| 18 | 4/1 | 5/1 | 10001/2730 | 9/3 | 5/2 |
| 19 | 593/159 | 1818/473 | 637/401 | 6005/2342 | 722/290 |
| 20 | 15/14 | 5480/2287 | 263/727 | 442/356 | 2177/1444 |
| 21 | 369/227 | 5415/4426 | 386/335 | 376/331 | 385/272 |
| 22 | 2/1 | 3/1 | 3/1 | 4/1 | 3/1 |
| 23 | 10/1 | 11/2 | 11/2 | 12/3 | 12/2 |
| 24 | 111/32 | 128/22 | 121/55 | 633/123 | 235/52 |
| 25 | 7/2 | 8/2 | 8/4 | 11/5 | 8/3 |
| 26 | 26/5 | 41/6 | 29/7 | 35/8 | 29/8 |
| 27 | 3572/987 | 5628/919 | 3877/998 | 6500/837 | 4754/877 |
| 28 | 2/1 | 6/1 | 6/1 | 11/11 | 6/2 |

## 5. Conclusions

In this paper, we have developed a novel three-term CG algorithm (NTTCG) based on modified gradient-differences for solving unconstrained optimization problems. Global convergence has been proved for this algorithm.

By applying our method to solve the 750 benchmark test problems, the numerical results have demonstrated that NTTCG outperforms the compared four algorithms in the literature. Especially, compared with the existing methods, NTTCG can find the optimal solutions of the unconstrained optimization problems, using less number of iterations, or less CPU time consumed.

In future research, it is valuable to study the method of obtaining the iteration direction by minimizing a quadratic approximate model of the objective function or the conic model in a specific subspace spanned by $g_{k+1}$, $s_k$ and $y_k$.

It is also interesting to extend the algorithm proposed in this paper to solve nonlinear system of monotone equations since it has been shown in [28–34] that recovering sparse signals and restoring blurred images can be formulated as a system of equations, and the CG methods can solve nonlinear system of monotone equations efficiently. Furthermore, as done in [35], we can also modify our algorithm to solve symmetric system of nonlinear equations.

## Acknowledgments

The authors are very grateful to the editors and referees for their constructive advice.

This research is supported by the National Social Science Foundation of China (Grant No. 21BGL122).

## Availability of data and materials

The data and material used to support the findings in this research can be provided by the corresponding author upon request.

## Conflict of interest

We declare that all the authors have no any conflicts of interest about this submission and publication of this article.

## References

1. J. Y. Tang, Z. Wan, Orthogonal dual graph-regularized nonnegative matrix factorization for co-clustering, *J. Sci. Comput.*, **87** (2021), 1–37. https://doi.org/10.1007/s10915-021-01489-w

2. T. Li, Z. Wan, New adaptive barzilar-borwein step size and its application in solving large scale optimization problems, *ANZIAM J.*, **61** (2019), 76–98. https://doi.org/10.1017/S1446181118000263

3. C. M. Hu, Z. M. Wan, S. Zhu, Z. Wan, An integrated stochastic model and algorithm for constrained multi-item newsvendor problems by two-stage decision-making approach, *Math. Comput. Simul.*, **193** (2022), 280–300. https://doi.org/10.1016/j.matcom.2021.10.018

4. Z. Wan, C. Q. Yu, Reutilization of the existent sales network for recycling unwanted smartphones by nonlinear optimization, *J. Cleaner Prod.*, **350** (2022), 131349. https://doi.org/10.1016/j.jclepro.2022.131349

5. S. Huang, Z. Wan, A new nonmonotone spectral residual method for nonsmooth nonlinear equations, *J. Comput. Appl. Math.*, **313** (2017), 82–101. http://doi.org/10.1016/j.cam.2016.09.014

6. S. Deng, Z. Wan, X. Chen, An improved spectral conjugate gradient algorithm for nonconvex unconstrained optimization problems, *J. Optim. Theory Appl.*, **157** (2013), 820–842. http://doi.org/10.1007/s10957-012-0239-7

7. W. W. Hager, H. C. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search, *SIAM J. Optim.*, **16** (2005), 170–192. http://doi.org/10.1137/030601880

8. N. Andrei, A simple three-term conjugate gradient algorithm for unconstrained optimization, *J. Comput. Appl. Math.*, **241** (2013), 19–29. http://doi.org/10.1016/j.cam.2012.10.002

9. N. Andrei, On three-term conjugate gradient algorithms for unconstrained optimization, *Appl. Math. Comput.*, **219** (2013), 6316–6327. http://doi.org/10.1016/j.amc.2012.11.097

10. L. Zhang, W. Zhou, D. H. Li, A descent modified Polak-Ribière-Polyak conjugate gradient method and its global convergence, *IMA J. Numer. Anal.*, **26** (2006), 629–640. http://doi.org/10.1093/imanum/drl016

11. W. L. Liu, Z. M. Wan, Z. Wan, B. Gong, Sustainable recycle network of heterogeneous pharmaceuticals with governmental subsidies and service-levels of third-party logistics by bi-level programming approach, *J. Cleaner Prod.*, **249** (2020), 119324. http://doi.org/10.1016/j.jclepro.2019.119324

12. J. Guo, Z. Wan, A new three-term spectral conjugate gradient algorithm with higher numerical performance for solving large scale optimization problems based on Quasi-Newton equation, *Int. J. Model. Simul. Sci. Comput.*, **12** (2021), 2150053. http://doi.org/10.1142/S1793962321500537

13. M. K. Riahi, I. A. Qattan, On the convergence rate of Fletcher-Reeves nonlinear conjugate gradient methods satisfying strong Wolfe conditions: application to parameter identification in problems governed by general dynamics, *Math. Methods Appl. Sci.*, **45** (2022), 3644–3664. http://doi.org/10.1002/mma.8009

14. M. Powell, Restart procedures for the conjugate gradient method, *Math. Program.*, **12** (1977), 241–254. http://doi.org/10.1007/BF01593790

15. L. Nazareth, A conjugate direction algorithm without line searches, *J. Optim. Theory Appl.*, **23** (1977), 373–387. http://doi.org/10.1007/BF00933447

16. N. Andrei, A modified Polak-Ribière-Polyak conjugate gradient algorithm for unconstrained optimization, *Optimization*, **60** (2011), 1457–1471. http://doi.org/10.1080/02331931003653187

17. L. Zhang, W. Zhou, D. H. Li, Some descent three-term conjugate gradient methods and their global convergence, *Optim. Methods Software*, **22** (2007), 697–711. http://doi.org/10.1080/10556780701223293

18. J. K. Liu, Y. M. Feng, L. M. Zou, Some three-term conjugate gradient methods with the inexact line search condition, *Calcolo*, **55** (2018), 16. http://doi.org/10.1007/s10092-018-0258-3

19. D. Touati-Ahmed, C. Storey, Efficient hybrid conjugate gradient techniques, *J. Optim. Theory Appl.*, **64** (1990), 379–397. http://doi.org/10.1007/bf00939455

20. M. Al-Baali, Descent property and global convergence of the Fletcher-Reeves method with inexact line search, *J. Numer. Anal.*, **5** (1985), 121–124. http://doi.org/10.1093/imanum/5.1.121

21. J. C. Gilbert, J. Nocedal, Global convergence properties of conjugate gradient methods for optimization, *SIAM J. Optim.*, **2** (1992), 21–42. http://doi.org/10.1137/0802003

22. Y. F. Hu, C. Storey, Global convergence result for conjugate gradient methods, *J. Optim. Theory Appl.*, **71** (1991), 399–405. http://doi.org/10.1007/bf00939927

23. S. Huang, Z. Wan, J. Zhang, An extended nonmonotone line search technique for large-scale unconstrained optimization, *J. Comput. Appl. Math.*, **330** (2018), 586–604. http://doi.org/10.1016/j.cam.2017.09.026

24. H. C. Zhang, W. W. Hager, A nonmonotone line search technique and its application to unconstrained optimization, *SIAM J. Optim.*, **14** (2004), 1043–1056. http://doi.org/10.1137/S1052623403428208

25. N. Andrei, An unconstrained optimization test functions collection, *Adv. Model. Optim.*, **10** (2008), 147–161.

26. I. Bongartz, A. R. Conn, N. Gould, P. L. Toint, Cute: Constrained and unconstrained testing environment, *ACM Trans. Math. Software*, **21** (1995), 123–160. http://doi.org/10.1145/200979.201043

27. E. D. Dolan, J. J. Moré, Benchmarking optimization software with performance profiles, *Math. Progra.*, **91** (2002), 201–213. http://doi.org/10.1007/s101070100263

28. J. Guo, Z. Wan, A modified spectral PRP conjugate gradient projection method for solving large-scale monotone equations and its application in compressed sensing, *Math. Probl. Eng.*, **2019** (2019), 5261830. http://doi.org/10.1155/2019/5261830

29. A. Althobaiti, J. Sabi'u, H. Emadifar, P. Junsawang, S. K. Sahoo, A scaled Dai-Yuan projection-based conjugate gradient method for solving monotone equations with applications, *Symmetry*, **14** (2022), 1401. http://doi.org/10.3390/sym14071401

30. J. Sabi'u, K. O. Aremu, A. Althobaiti, A. Shah, Scaled three-term conjugate gradient methods for solving monotone equations with application, *Symmetry*, **14** (2022), 936. http://doi.org/10.3390/sym14050936

31. J. Sabi'u, A. Shah, M. Y. Waziri, A modified Hager-Zhang conjugate gradient method with optimal choices for solving monotone nonlinear equations, *Int. J. Comput. Math.*, **99** (2022), 332–354. http://doi.org/10.1080/00207160.2021.1910814

32. N. Ullah, J. Sabi'u, A. Shah, A derivative-free scaling memoryless Broyden-Fletcher-Goldfarb-Shanno method for solving a system of monotone nonlinear equations, *Numer. Linear Algebra Appl.*, **28** (2021), 2374. http://doi.org/10.1002/nla.2374

33. A. B. Abubakar, J. Sabi'u, P. Kumam, A. Shah, Solving nonlinear monotone operator equations via modified SR1 update, *J. Appl. Math. Comput.*, **67** (2021), 343–373. http://doi.org/10.1007/s12190-020-01461-1

34. J. Sabi'u, A. Shah, M. Y. Waziri, K. Ahmed, Modified Hager-Zhang conjugate gradient methods via singular value analysis for solving monotone nonlinear equations with convex constraint, *Int. J. Comput. Methods*, **18** (2021), 2050043. http://doi.org/10.1142/S0219876220500437

35. J. Sabi'u, K. Muangchoo, A. Shah, A. B. Abubakar, K. O. Aremu, An inexact optimal hybrid conjugate gradient method for solving symmetric nonlinear equations, *Symmetry*, **13** (2021), 1829. http://doi.org/10.3390/sym13101829