Comput Optim Appl (2015) 60:89–110
DOI 10.1007/s10589-014-9662-z

A family of three-term conjugate gradient methods with sufficient descent property for unconstrained optimization

Mehiddin Al-Baali · Yasushi Narushima · Hiroshi Yabe

Received: 31 January 2013 / Published online: 20 May 2014

Abstract Recently, conjugate gradient methods, which usually generate descent search directions, are useful for large-scale optimization. Narushima et al. (SIAM J Optim 21:212–230, 2011) have proposed a three-term conjugate gradient method which satisfies a sufficient descent condition. We extend this method to two parameters family of three-term conjugate gradient methods which can be used to control the magnitude of the directional derivative. We show that these methods converge globally and work well for suitable choices of the parameters. Numerical results are also presented

Keywords Unconstrained optimization · Three-term conjugate gradient method · Sufficient descent condition · Global convergence

M. Al-Baali
Department of Mathematics and Statistics, Sultan Qaboos University, P.O. Box 36, Al-Khoud 123, Muscat, Oman
e-mail: albaali@squ.edu.om

Y. Narushima (✉)
Department of Management System Science, Yokohama National University, 79-4, Tokiwadai, Hodogaya-ku, Yokohama 240-8501, Japan
e-mail: narushima@ynu.ac.jp

H. Yabe
Department of Mathematical Information Science, Tokyo University of Science, 1-3, Kagurazaka, Shinjuku-ku, Tokyo 162-8601, Japan
e-mail: yabe@rs.kagu.tus.ac.jp

1 Introduction

Consider the unconstrained optimization problem

$$\text{minimize } f(x),$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function. Iterative methods are usually used for solving this problem by generating a sequence of points x_k, using the formula

$$x_{k+1} = x_k + \alpha_k d_k, \tag{1.1}$$

for $k \geq 0$, where x_0 is a given initial point, α_k is a positive stepsize and d_k is a search direction.

Conjugate gradient methods are usually effective for large-scale optimization and define the search direction by

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

where g_k denotes the gradient $g(x_k)$ of f computed at x_k and β_k is a parameter. Well-known formulae for β_k are the Hestenes-Stiefel (HS) [20,25], Fletcher-Reeves (FR) [1,11], Polak-Ribière (PR) [23], Conjugate Descent (CD) [9], Liu-Storey [21], and Dai-Yuan (DY) [7], given respectively by

$$\begin{aligned} \beta_k^{HS} &= \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \quad \beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \quad \beta_k^{PR} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \\ \beta_k^{CD} &= \frac{\|g_k\|^2}{-g_{k-1}^T d_{k-1}}, \ \beta_k^{LS} = \frac{g_k^T y_{k-1}}{-g_{k-1}^T d_{k-1}}, \ \beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}, \end{aligned} \tag{1.2}$$

where $y_{k-1} = g_k - g_{k-1}$ and $\|\cdot\|$ denotes the ℓ_2 Euclidean norm. Note that the difference in points $s_{k-1} = x_k - x_{k-1}$ will be used in the subsequent sections. If $f(x)$ is a strictly convex quadratic function and α_k is the exact one-dimensional minimizer, then all formulae in (1.2) are identical. Several properties of these methods have been obtained (see for example [16]).

In order to obtain the global convergence property of conjugate gradient methods for general objective functions, the sufficient descent condition

$$g_k^T d_k \leq -c\|g_k\|^2, \tag{1.3}$$

for some positive constant c, is usually enforced for all values of k. Several modified conjugate gradient methods which ensure this condition have been proposed. In particular, Zhang, Zhou and Li have proposed a modified FR method, a three-term PR method and a three-term HS method (see [28,29] and [30], respectively). Cheng [4] has also proposed a modified PR method. These methods satisfy the sufficient descent

condition (1.3) with $c = 1$. On the other hand, Narushima, Yabe and Ford [22] have proposed a family of three-term conjugate gradient methods, defined by

$$d_k = \begin{cases} -g_k & \text{if } k = 0, \\ -g_k + \beta_k (g_k^T p_k)^\dagger [(g_k^T p_k) d_{k-1} - (g_k^T d_{k-1}) p_k] & \text{if } k \geq 1, \end{cases} \tag{1.4}$$

where p_k is a parameter vector and

$$a^\dagger = \begin{cases} \dfrac{1}{a} & \text{if } a \neq 0, \\ 0 & \text{if } a = 0. \end{cases}$$

Note that the search direction (1.4) always satisfies $g_k^T d_k = -\|g_k\|^2$ and the corresponding method (referred to as 3TCG) yields those in [4,28–30] as special cases. In the 3TCG method, the magnitude of the sufficient descent condition (namely c) is fixed by 1. Our interest here is to investigate the affect of the magnitude on the performance of the method. We will construct a general form of the 3TCG method which satisfies the equation $g_k^T d_k = -\gamma_k \|g_k\|^2$, where γ_k is a positive parameter which can be controlled by the user. By choosing this parameter effectively, an efficient method will be developed.

This paper is organized as follows. On the basis of the 3TCG method of [22], we propose in Sect. 2 a family of three-term conjugate gradient methods which depends on the above two parameters (β_k and γ_k) and enforces the sufficient descent condition on all iterations. This condition is used in Sect. 3 to show that the family converges globally for general nonlinear objective functions. In Sect. 4, we consider several choices for the parameters and, in Sect. 5, we describe some numerical results and particularly investigate the effects of the magnitude of the parameter γ_k in several cases. It is concluded that certain choices for the parameters work well in practice.

2 A family of three-term conjugate gradient methods

We now consider a family of three-term conjugate gradient method whose search direction is given by

$$d_k = \begin{cases} -g_k & \text{if } k = 0 \text{ or } |g_k^T p_k| \leq \theta \|g_k\| \|p_k\|, \\ -g_k + \beta_k d_{k-1} + \eta_k p_k & \text{otherwise}, \end{cases} \tag{2.1}$$

where p_k is any nonzero vector, $0 < \theta < 1$ is a constant, β_k is a parameter and

$$\eta_k = -\frac{(\gamma_k - 1)\|g_k\|^2 + \beta_k g_k^T d_{k-1}}{g_k^T p_k}. \tag{2.2}$$

Here, γ_k denotes another parameter which, for convenience, is bounded by $\bar{\gamma}_1 \leq \gamma_k \leq \bar{\gamma}_2$, where $0 < \bar{\gamma}_1 \leq 1 \leq \bar{\gamma}_2$, noting that the choice $\gamma_k = 1$ reduces direction (2.1) to that of [22].

We observe that the steepest descent condition (which appears in the first case of (2.1) and referred to as SDC) ensures that the search direction d_k is well defined for any values of p_k and β_k. In particular, we observe the following cases. The choice of $p_k = y_{k-1}$ reduces SDC to the condition $|g_k^T y_{k-1}| \leq \theta \|g_k\| \|y_{k-1}\|$ which plays as a restarting criterion, since it holds for sufficiently large values of θ. However, the choice of $p_k = g_k$ reduces SDC to the condition $\theta \geq 1$ which yields that the second case in (2.1) is always used to define the search direction. For the choice of $p_k = d_{k-1}$, the second case in (2.1) is reduced to

$$d_k = -g_k - \frac{(\gamma_k - 1)\|g_k\|^2}{g_k^T d_{k-1}} d_{k-1}.$$

This direction with $\gamma_k = 1 + \frac{g_k^T d_{k-1}}{g_{k-1}^T d_{k-1}}$ is reduced further to the usual conjugate gradient direction with $\beta_k = \beta_k^{CD}$.

In general, for any choice of p_k, it follows from (2.1) that

$$g_k^T d_k = -\gamma_k \|g_k\|^2, \tag{2.3}$$

with $\gamma_k = 1$ for the first case of (2.1). This sufficient descent condition clearly shows that the value of the directional derivative $g_k^T d_k$ depends on the choice of the parameter γ_k. In particular, using the above bounds on γ_k, condition (2.3) yields that

$$-\bar{\gamma}_2 \|g_k\|^2 \leq g_k^T d_k \leq -\bar{\gamma}_1 \|g_k\|^2. \tag{2.4}$$

To illustrate the behaviour of the search direction, we first use the second case of (2.1) to state it as follows

$$d_k = \left(I - \frac{p_k g_k^T}{g_k^T p_k} \right) (-g_k + \beta_k d_{k-1}) + \frac{p_k g_k^T}{g_k^T p_k} (-\gamma_k g_k). \tag{2.5}$$

The matrix $\left(I - p_k g_k^T / g_k^T p_k\right)$ is a projection into the orthogonal complement of Span$\{g_k\}$ along Span$\{p_k\}$, and the matrix $p_k g_k^T / g_k^T p_k$ is a projection into Span$\{p_k\}$ along the orthogonal complement of Span$\{g_k\}$ (for illustration, see Fig. 1 and note that $P = \frac{p_k g_k^T}{g_k^T p_k}$).

We now estimate the norm of the search direction of the proposed method which is used in the subsequent section. For the first case of (2.1), it follows that

$$\|d_k\| = \|g_k\|.$$

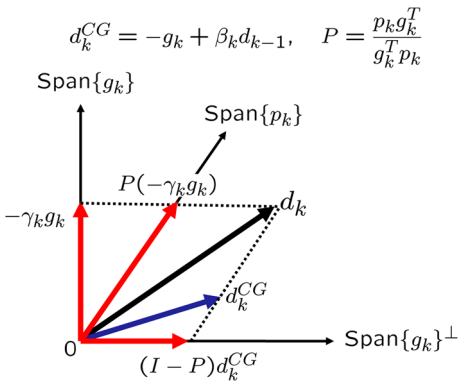

Fig. 1 Image of the search direction

Otherwise, since $\left\| I - \frac{p_k g_k^T}{g_k^T p_k} \right\| = \frac{\|g_k\|\|p_k\|}{|g_k^T p_k|}, \left\| \frac{p_k g_k^T}{g_k^T p_k} \right\| \leq \frac{\|g_k\|\|p_k\|}{|g_k^T p_k|}, 0 < \bar{\gamma}_2$ and $0 < \theta < 1$, expression (2.5) yields

$$\begin{aligned} \|d_k\| &\leq \left\| I - \frac{p_k g_k^T}{g_k^T p_k} \right\| \| - g_k + \beta_k d_{k-1}\| + \left\| \frac{p_k g_k^T}{g_k^T p_k} \right\| \| - \gamma_k g_k\| \\ &\leq \frac{1}{\theta}\{(1+\bar{\gamma}_2)\|g_k\| + |\beta_k|\|d_{k-1}\|\} \\ &\leq \frac{1+\bar{\gamma}_2}{\theta}\{\|g_k\| + |\beta_k|\|d_{k-1}\|\} \\ &\leq \mu(\|g_k\| + |\beta_k|\|d_{k-1}\|), \end{aligned} \tag{2.6}$$

where $\mu = (1+\bar{\gamma}_2)/\theta$ which is strictly greater than one.

We now outline the algorithm of a general version of the 3TCG method (referred to as G3TCG) as follows.

Algorithm G3TCG

Step 0 Give an initial point x_0 and a value for $\theta \in (0, 1)$. Set $k = 0$ and compute g_0.
Step 1 Stop if a stopping criterion holds.
Step 2 Compute d_k by (2.1) for given β_k and γ_k.
Step 3 Determine a stepsize $\alpha_k > 0$.
Step 4 Compute x_{k+1} by (1.1).
Step 5 Set $k := k + 1$ and go to Step 1.

In the line search Step 3, the stepsize α_k is chosen to satisfy either the Wolfe conditions

$$f(x_k) - f(x_k + \alpha_k d_k) \geq -\delta \alpha_k g_k^T d_k, \tag{2.7}$$

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma_1 g_k^T d_k, \tag{2.8}$$

where $0 < \delta < \frac{1}{2}$ and $\delta < \sigma_1 < 1$, or the generalized strong Wolfe conditions (2.7) and

$$-\sigma_2 g_k^T d_k \geq g(x_k + \alpha_k d_k)^T d_k \geq \sigma_1 g_k^T d_k, \tag{2.9}$$

where $\sigma_2 > 0$. Note that condition (2.9) implies condition (2.8) and both of them ensure the curvature condition $d_{k-1}^T y_{k-1} > 0$, because they imply (by (2.4)) that

$$d_{k-1}^T y_{k-1} \geq -(1-\sigma_1) g_{k-1}^T d_{k-1} \geq \bar{\gamma}_1 (1-\sigma_1) \|g_{k-1}\|^2. \tag{2.10}$$

Condition (2.9) also implies that

$$|g_k^T d_{k-1}| \leq \max\{\sigma_1, \sigma_2\} |g_{k-1}^T d_{k-1}| \leq \bar{\gamma}_2 \max\{\sigma_1, \sigma_2\} \|g_{k-1}\|^2. \tag{2.11}$$

3 Convergence analysis

In this section, we obtain the global convergence property of Algorithm G3TCG for a certain condition on β_k and the following standard assumptions on the objective function.

Assumption 1 1. The level set $\mathcal{L} = \{x | f(x) \leq f(x_0)\}$ is bounded, namely, there exists a constant $\widehat{a} > 0$ such that

$$\|x\| \leq \widehat{a}, \tag{3.1}$$

for all $x \in \mathcal{L}$.

2. In some neighborhood $\mathcal{N}$ of $\mathcal{L}$, f is continuously differentiable and its gradient g is Lipschitz continuous with a Lipschitz constant $L > 0$, i.e.

$$\|g(u) - g(v)\| \leq L\|u - v\|,$$

for all $u, v \in \mathcal{N}$.

This assumption implies that there exists a positive constant $\widehat{g} > 0$ such that

$$\|g(x)\| \leq \widehat{g}, \tag{3.2}$$

for all $x \in \mathcal{L}$. In the rest of this section, we assume $g_k \neq 0$ for all k, otherwise a stationary point has been found. Under Assumption 1, we have the following lemma, which is easily obtained from the Zoutendijk condition [31]. The proof of the lemma is given for example in [24].

Lemma 1 *Suppose that Assumption 1 is satisfied. Consider any iterative method of the form (1.1) such that the sufficient descent condition (1.3) and the Wolfe conditions (2.7) and (2.8) are satisfied. If*

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

then $\liminf_{k\to\infty} \|g_k\| = 0$ *holds.*

We sometimes use the counterproposition of Lemma 1, which is that $\sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} < \infty$ holds if $\liminf_{k\to\infty} \|g_k\| \neq 0$. The following property corresponds to *Property* $(*)$ derived by Gilbert and Nocedal [12] for standard conjugate gradient methods (see also Dai and Liao [6]). This property implies that β_k will be small when the step s_{k-1} is small.

Property A *Assume Algorithm G3TCG generates a sequence of points such that the inequality* $\|g_k\| \geq \varepsilon$ *holds for some* $\varepsilon > 0$ *and all values of k. Then we say that the algorithm has Property A if there exist constants* $b > 1$ *and* $\xi > 0$ *such that for all k,*

$$|\beta_k| \leq b \tag{3.3}$$

and

$$\|s_{k-1}\| \leq \xi \implies |\beta_k| \leq \frac{1}{4\mu^4 b}, \tag{3.4}$$

where $\mu > 1$ *is defined as in* (2.6).

We now consider Algorithm G3TCG which satisfies this property and

$$\beta_k \geq \min\{\nu_k^{(1)}, \nu_k^{(2)}\} \equiv \nu_k, \tag{3.5}$$

where

$$\nu_k^{(1)} = \frac{-1}{\|d_{k-1}\| \min\{\bar{\nu}_1, \|g_{k-1}\|\}}, \quad \nu_k^{(2)} = \bar{\nu}_2 \frac{g_{k-1}^T d_{k-1}}{\|d_{k-1}\|^2}, \tag{3.6}$$

and where $\bar{\nu}_1$ and $\bar{\nu}_2$ are positive constants. Since this condition may not hold in certain cases, we modify it to

$$\beta_k^+ = \max\{\zeta_k, \beta_k\}, \tag{3.7}$$

where $\zeta_k \in [\nu_k, 0]$, so that $\beta_k^+ \geq \nu_k$. Note that the choices of $\zeta_k \equiv 0$, $\zeta_k \equiv \nu_k^{(1)}$ and $\zeta_k \equiv \nu_k^{(2)}$ reduce formula (3.7) to those proposed in [12], [15] and [5], respectively.

In the rest of this section, we assume without loss of generality that

$$|g_k^T p_k| > \theta \|g_k\| \|p_k\| \tag{3.8}$$

for some positive constant $\theta < 1$, as in (2.1), so that the search direction d_k is defined by the second case of (2.1) for all $k \geq 1$. We note that if condition (3.8) does not hold

infinitely many times, then the search direction becomes the steepest descent direction infinitely many times, which implies that $\liminf_{k\to\infty} \|g_k\| = 0$.

The next lemma corresponds to Lemma 3.4 of Dai and Liao [6] and its proof is a simple extension to that in [6].

Lemma 2 *Suppose that Assumption 1 is satisfied. Let $\{x_k\}$ be a sequence of points generated by Algorithm G3TCG with the generalized strong Wolfe conditions (2.7) and (2.9). Assume that there exists a positive constant ε such that $\|g_k\| \geq \varepsilon$ holds for all k. If the method has Property A and $\beta_k \geq v_k$ holds, then $d_k \neq 0$ and the following relation holds*:

$$\sum_{k=1}^{\infty} \|u_k - u_{k-1}\|^2 < \infty,$$

where $u_k = d_k/\|d_k\|$.

Proof Since $d_k \neq 0$ follows from (2.4) and $\|g_k\| \geq \varepsilon$, the vector u_k is well-defined. Using this inequality and Lemma 1, we obtain

$$\sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} < \infty. \tag{3.9}$$

Letting $\beta_k^{(1)} = \max\{0, \beta_k\}$, $\beta_k^{(2)} = \min\{0, \beta_k\}$, $\omega_k = \beta_k^{(1)} \dfrac{\|d_{k-1}\|}{\|d_k\|}$ and

$$v_k = \frac{-g_k + \beta_k^{(2)} d_{k-1} + \eta_k p_k}{\|d_k\|}, \tag{3.10}$$

we obtain

$$u_k = v_k + \omega_k u_{k-1}.$$

Then it follows from $\|u_k\| = \|u_{k-1}\| = 1$ that

$$\|v_k\| = \|u_k - \omega_k u_{k-1}\| = \|\omega_k u_k - u_{k-1}\|,$$

and hence, $\omega_k \geq 0$ implies

$$\begin{aligned} \|u_k - u_{k-1}\| &\leq (1 + \omega_k)\|u_k - u_{k-1}\| \\ &= \|u_k - \omega_k u_{k-1} + \omega_k u_k - u_{k-1}\| \\ &\leq \|u_k - \omega_k u_{k-1}\| + \|\omega_k u_k - u_{k-1}\| \\ &= 2\|v_k\|. \end{aligned} \tag{3.11}$$

On the other hand, expressions (2.2), (2.11), (3.2), (3.3), (3.5), (3.6) and (3.8) yield

$$\begin{aligned}
&\| - g_k + \beta_k^{(2)} d_{k-1} + \eta_k p_k \| \\
&\quad = \left\| -g_k + \beta_k^{(2)} d_{k-1} + (1-\gamma_k)\frac{\|g_k\|^2}{g_k^T p_k} p_k - \beta_k \frac{g_k^T d_{k-1}}{g_k^T p_k} p_k \right\| \\
&\quad \le \widehat{g} - \min\{0, \beta_k\}\|d_{k-1}\| + \frac{(1+\bar{\gamma}_2)\widehat{g}}{\theta} + |\beta_k| \frac{|g_k^T d_{k-1}|}{\theta\varepsilon} \\
&\quad \le \widehat{g} + \max\{|\nu_k^{(1)}|, |\nu_k^{(2)}|\}\|d_{k-1}\| + \frac{(1+\bar{\gamma}_2)\widehat{g}}{\theta} + b\frac{\bar{\gamma}_2 \max\{\sigma_1, \sigma_2\}\|g_{k-1}\|^2}{\theta\varepsilon} \\
&\quad \le \widehat{g} + \max\left\{\frac{1}{\min\{\bar{\nu}_1, \varepsilon\}}, \bar{\nu}_2 \widehat{g}\right\} + \frac{(1+\bar{\gamma}_2)\widehat{g}}{\theta} + \frac{b\bar{\gamma}_2 \max\{\sigma_1, \sigma_2\}\widehat{g}^2}{\theta\varepsilon} \equiv c_1.
\end{aligned}$$

Therefore, we obtain from (3.9), (3.10) and (3.11) that

$$\sum_{k=1}^{\infty} \|u_k - u_{k-1}\|^2 \le 4 \sum_{k=1}^{\infty} \|v_k\|^2 \le 4c_1^2 \sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} < \infty,$$

which completes the proof. □

The next lemma shows that if the gradients are bounded away from zero and Property A holds, then a certain fraction of steps cannot be too small. Although the lemma corresponds to [6, Lemma 3.5] and [12, Lemma 4.2] and its proof is similar to that of [12, Lemma 4.2], we state it here for complete readability. For given positive constant λ and positive integer Δ, we let

$$\mathcal{K}_{k,\Delta}^{\lambda} = \{i \in \mathbb{N} \mid k \le i \le k + \Delta - 1, \ \|s_{i-1}\| > \lambda\}$$

be a set of indices, which consists of a number of elements denoted by $|\mathcal{K}_{k,\Delta}^{\lambda}|$.

Lemma 3 *Suppose that all assumptions of Lemma 2 hold. Then there exist a constant $\lambda > 0$ and an index $\widehat{k} \ge k_0$, for any $\Delta \in \mathbb{N}$ and any index k_0, such that*

$$|\mathcal{K}_{\widehat{k},\Delta}^{\lambda}| > \frac{\Delta}{2}.$$

Proof We prove this lemma by contradiction. Assume that for any $\lambda > 0$, there exist Δ and k_0 such that

$$|\mathcal{K}_{k,\Delta}^{\lambda}| \le \frac{\Delta}{2} \tag{3.12}$$

for all $k \ge k_0$. Let $b > 1$ and $\xi > 0$ be given as in Property A and set $\lambda = \xi$. We choose Δ and k_0 such that (3.12) holds. Then it follows from (3.3), (3.4), (3.12), $b > 1$ and $\mu > 1$ that

$$\prod_{k=k_0+i\Delta+1}^{k_0+(i+1)\Delta} |\beta_k| = \prod_{k\in\mathcal{K}^{\lambda}_{k',\Delta}} |\beta_k| \prod_{k\notin\mathcal{K}^{\lambda}_{k',\Delta}} |\beta_k| \le b^{\Delta/2} \left(\frac{1}{4\mu^4 b}\right)^{\Delta/2} \le 1 \quad (3.13)$$

for any $i \ge 0$, where $k' = k_0 + i\Delta + 1$. Therefore, we have

$$\prod_{j=k_0+1}^{k_0+i\Delta} 2\mu^2\beta_j^2 = \prod_{j=k_0+1}^{k_0+\Delta} 2\mu^2\beta_j^2 \cdots \prod_{j=k_0+(i-1)\Delta+1}^{k_0+i\Delta} 2\mu^2\beta_j^2 \le (2\mu^2)^{i\Delta} \left(\frac{1}{2\mu^2}\right)^{2i\Delta} \le 1. \quad (3.14)$$

For all indices $i \ge 1$ and $k_0 \le \ell \le k_0 + i\Delta$, there exists an index i' such that $k_0 + i'\Delta \le \ell \le k_0 + (i'+1)\Delta \le k_0 + i\Delta$ and

$$\prod_{j=\ell}^{k_0+i\Delta} 2\mu^2\beta_j^2 = \prod_{j=\ell}^{k_0+(i'+1)\Delta} 2\mu^2\beta_j^2 \prod_{j=k_0+(i'+1)\Delta+1}^{k_0+(i'+2)\Delta} 2\mu^2\beta_j^2 \cdots \prod_{j=k_0+(i-1)\Delta+1}^{k_0+i\Delta} 2\mu^2\beta_j^2,$$

which implies by (3.3), (3.13), $\mu \ge 1$ and $b > 1$ that

$$\prod_{j=\ell}^{k_0+i\Delta} 2\mu^2\beta_j^2 \le (2\mu^2 b^2)^{\Delta} \equiv c_2. \quad (3.15)$$

Relations (2.6), (3.2), (3.14) and (3.15) yield

$$\begin{aligned}
\|d_{k_0+i\Delta}\|^2 &\le 2\mu^2(\|g_{k_0+i\Delta}\|^2 + |\beta_{k_0+i\Delta}|^2\|d_{k_0+i\Delta-1}\|^2) \\
&\le 2\mu^2\widehat{g}^2 + 2\mu^2\widehat{g}^2 \sum_{\ell=k_0+2}^{k_0+i\Delta} \left(\prod_{j=\ell}^{k_0+i\Delta} 2\mu^2\beta_j^2\right) + \|d_{k_0}\| \prod_{j=k_0+1}^{k_0+i\Delta} 2\mu^2\beta_j^2 \\
&\le 2\mu^2\widehat{g}^2 + 2\mu^2 c_2\widehat{g}^2(i\Delta - 1) + c_3,
\end{aligned}$$

where $c_3 = \|d_{k_0}\|$. Thus, we have

$$\sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} \ge \sum_{i=1}^{\infty} \frac{1}{\|d_{k_0+i\Delta}\|^2} \ge \sum_{i=1}^{\infty} \frac{1}{2\mu^2\widehat{g}^2 + 2\mu^2 c_2\widehat{g}^2(i\Delta - 1) + c_3} = \infty,$$

which yields, from Lemma 1, $\liminf_{k\to\infty} \|g_k\| = 0$. Since this contradicts the assumption $\|g_k\| \ge \varepsilon$, we obtain the desired result. □

We now give a sufficient condition for the global convergence of the family of methods (1.1) and (2.1), using Lemmas 2 and 3 and *Property* A, as in the following theorem. Since the theorem corresponds to [6, Theorem 3.6] and its proof is the

same as that in [6, Theorem 3.6], we will not prove it here. However, we state it for readability.

Theorem 1 *Suppose that Assumption 1 is satisfied. Let* $\{x_k\}$ *be a sequence of points generated by Algorithm G3TCG with the generalized strong Wolfe conditions* (2.7) *and* (2.9)*. If this algorithm has Property A and* β_k *satisfies condition* (3.5)*, then the sequence* $\{x_k\}$ *converges globally in the sense that* $\liminf_{k\to\infty} \|g_k\| = 0$.

4 Some choices for the parameters β_k and γ_k

In Sect. 3, we obtained the global convergence property of Algorithm G3TCG which has Property A and satisfies the inequality $\beta_k \geq \nu_k$ as in (3.5). Since we do not discuss concrete choices of the parameters γ_k and β_k, we consider some choices below.

We will first prove the global convergence of the proposed family of methods with β_k defined by either β_k^{HS}, β_k^{PR} or β_k^{LS} which are given in (1.2). We will also extend this result to the formulae of Dai-Liao (DL) [6], Hager-Zhang (HZ) [15], Yu-Guan-Li (DPR) [26] and Zhang (DLS) [27], which are respectively given by

$$\beta_k^{DL} = \frac{g_k^T (y_{k-1} - t s_{k-1})}{d_{k-1}^T y_{k-1}}, \tag{4.1}$$

$$\beta_k^{HZ} = \beta_k^{HS} - \frac{\phi \|y_k\|^2}{(d_{k-1}^T y_{k-1})^2} g_k^T d_{k-1}, \tag{4.2}$$

$$\beta_k^{DPR} = \beta_k^{PR} - \frac{\phi \|y_k\|^2}{\|g_{k-1}\|^4} g_k^T d_{k-1}, \tag{4.3}$$

$$\beta_k^{DLS} = \beta_k^{LS} - \frac{\phi \|y_k\|^2}{(-g_{k-1}^T d_{k-1})^2} g_k^T d_{k-1}, \tag{4.4}$$

where $t \geq 0$ and $\phi > 1/4$ are parameters.

Theorem 1 establishes the global convergence of the proposed methods (1.1) and (2.1) with Property A and $\beta_k \geq \nu_k$. If this inequality does not hold, we modify the value of β_k by using (3.7), as we have mentioned in Sect. 3. Thus, we need to prove that each method choice has only Property A as shown below. Hence if $\|g_k\| > \varepsilon$ holds, for any value of k and some positive constant ε, then there exists some positive constant c_4 such that

$$|\beta_k| \leq c_4 \|s_{k-1}\|. \tag{4.5}$$

Then, letting $b = 1 + 2\widehat{a} c_4$ and $\xi = 1/(4c_4\mu^4 b)$, we have both $|\beta_k| \leq b$ and $|\beta_k| \leq 1/(4\mu^4 b)$ if $\|s_{k-1}\| \leq \xi$. Thus, if (4.5) holds, then the method under consideration has Property A and we easily prove the following lemma.

Lemma 4 *Suppose that Assumption 1 holds and* α_k *satisfies the generalized strong Wolfe conditions* (2.7) *and* (2.9)*. If there exists a positive constant* ε *such that* $\|g_k\| > \varepsilon$*, then condition* (4.5) *holds with* β_k *replaced by either*

β_k^{HS}, β_k^{PR}, β_k^{LS}, β_k^{DL}, β_k^{HZ}, β_k^{DPR} or β_k^{DLS} (*the latter four choices are defined by* (4.1), (4.2), (4.3) *and* (4.4), *respectively, while the other three are given in* (1.2)).

Proof From Assumption 1, it follows that

$$|g_k^T y_{k-1}| \le \widehat{g} L \|s_{k-1}\|. \tag{4.6}$$

Substituting this inequality into the considered choices of β_k in (1.2) and using either (2.10) or (2.4) with $\|g_k\| > \varepsilon$, we obtain the following inequalities:

$$|\beta_k^{HS}| = \left|\frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}\right| \le \frac{\widehat{g} L}{\bar{\gamma}_1(1-\sigma_1)\varepsilon^2}\|s_{k-1}\|, \tag{4.7}$$

$$|\beta_k^{PR}| = \left|\frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}\right| \le \frac{\widehat{g} L}{\varepsilon^2}\|s_{k-1}\|, \tag{4.8}$$

$$|\beta_k^{LS}| = \left|\frac{g_k^T y_{k-1}}{-g_{k-1}^T d_{k-1}}\right| \le \frac{\widehat{g} L}{\bar{\gamma}_1\varepsilon^2}\|s_{k-1}\|. \tag{4.9}$$

Substituting the above four inequalities into (4.1), (4.2), (4.3) and (4.4), respectively, and using (2.10), (2.11) and/or (3.2), whenever necessary, we obtain the following bounds:

$$|\beta_k^{DL}| = \left|\frac{g_k^T (y_{k-1} - t s_{k-1})}{d_{k-1}^T y_{k-1}}\right| \le \frac{(L+t)\widehat{g}}{\bar{\gamma}_1(1-\sigma_1)\varepsilon^2}\|s_{k-1}\|,$$

$$|\beta_k^{HZ}| = \left|\beta_k^{HS} - \frac{\phi\|y_k\|^2}{(d_{k-1}^T y_{k-1})^2} g_k^T d_{k-1}\right|$$
$$\le \left(\frac{\widehat{g} L}{\bar{\gamma}_1(1-\sigma_1)\varepsilon^2} + \frac{2\widehat{a}\phi L^2}{\bar{\gamma}_1^2(1-\sigma_1)^2\varepsilon^4}\bar{\gamma}_2\widehat{g}^2\max\{\sigma_1,\sigma_2\}\right)\|s_{k-1}\|,$$

$$|\beta_k^{DPR}| = \left|\beta_k^{PR} - \frac{\phi\|y_k\|^2}{\|g_{k-1}\|^4} g_k^T d_{k-1}\right|$$
$$\le \left(\frac{\widehat{g} L}{\varepsilon^2} + \frac{2\widehat{a}\phi L^2}{\varepsilon^4}\bar{\gamma}_2\widehat{g}^2\max\{\sigma_1,\sigma_2\}\right)\|s_{k-1}\|,$$

$$|\beta_k^{DLS}| = \left|\beta_k^{LS} - \frac{\phi\|y_k\|^2}{(-g_{k-1}^T d_{k-1})^2} g_k^T d_{k-1}\right|$$
$$\le \left(\frac{\widehat{g} L}{\bar{\gamma}_1\varepsilon^2} + \frac{2\widehat{a}\phi L^2}{\bar{\gamma}_1^2\varepsilon^4}\bar{\gamma}_2\widehat{g}^2\max\{\sigma_1,\sigma_2\}\right)\|s_{k-1}\|.$$

The above eight inequalities clearly show that condition (4.5) holds with β_k replaced by the left hand side of these inequalities and appropriate values of the constant c_4. Thus, the proof is complete. □

The next theorem follows directly from Theorem 1 and Lemma 4.

Theorem 2 *Suppose that Assumption 1 is satisfied. Let $\{x_k\}$ be a sequence of points generated by Algorithm G3TCG with the generalized strong Wolfe conditions (2.7) and (2.9). If we choose β_k^+ by (3.7) with β_k replaced by either β_k^{HS}, β_k^{PR}, β_k^{LS}, β_k^{DL}, β_k^{HZ}, β_k^{DPR} or β_k^{DLS}, then the algorithm converges globally in the sense that* $\liminf_{k\to\infty} \|g_k\| = 0$.

We now deal with the parameter γ_k. In order to establish the conditions $\bar{\gamma}_1 \le \gamma_k \le \bar{\gamma}_2$, as required in (2.2) for given $\bar{\gamma}_1$ and $\bar{\gamma}_2$, we let

$$\gamma_k = \max\{\bar{\gamma}_1, \min\{\bar{\gamma}_2, \widehat{\gamma}_k\}\},$$

for a certain choice $\widehat{\gamma}_k$ of the parameter. In a general conjugate gradient method, the sufficient descent property $g_k^T d_k = -\|g_k\|^2$ is obtained if the exact line search condition $g_k^T d_{k-1} = 0$ is satisfied. Thus, considering (2.3), it is natural to control γ_k so that $\gamma_k = 1$ if the latter condition holds. In particular, we consider the following choices of the form $\widehat{\gamma}_k = \widehat{\gamma}_k^{(i)}$, for $i = 1, 2, \ldots, 16$:

$$\begin{aligned}
\widehat{\gamma}_k^{(1)} &= 1 - \bar{\gamma}\tfrac{|\beta_k g_k^T d_{k-1}|}{\|g_k\|\|d_{k-1}\|}, & \widehat{\gamma}_k^{(2)} &= 1 + \bar{\gamma}\tfrac{|\beta_k g_k^T d_{k-1}|}{\|g_k\|\|d_{k-1}\|},\\
\widehat{\gamma}_k^{(3)} &= 1 - \bar{\gamma}\tfrac{\beta_k g_k^T d_{k-1}}{\|g_k\|\|d_{k-1}\|}, & \widehat{\gamma}_k^{(4)} &= 1 + \bar{\gamma}\tfrac{\beta_k g_k^T d_{k-1}}{\|g_k\|\|d_{k-1}\|},\\
\widehat{\gamma}_k^{(5)} &= 1 - \bar{\gamma}|\beta_k g_k^T d_{k-1}|, & \widehat{\gamma}_k^{(6)} &= 1 + \bar{\gamma}|\beta_k g_k^T d_{k-1}|,\\
\widehat{\gamma}_k^{(7)} &= 1 - \bar{\gamma}\beta_k g_k^T d_{k-1}, & \widehat{\gamma}_k^{(8)} &= 1 + \bar{\gamma}\beta_k g_k^T d_{k-1},\\
\widehat{\gamma}_k^{(9)} &= 1 - \bar{\gamma}\tfrac{|g_k^T d_{k-1}|}{\|g_k\|\|d_{k-1}\|}, & \widehat{\gamma}_k^{(10)} &= 1 + \bar{\gamma}\tfrac{|g_k^T d_{k-1}|}{\|g_k\|\|d_{k-1}\|},\\
\widehat{\gamma}_k^{(11)} &= 1 - \bar{\gamma}\tfrac{g_k^T d_{k-1}}{\|g_k\|\|d_{k-1}\|}, & \widehat{\gamma}_k^{(12)} &= 1 + \bar{\gamma}\tfrac{g_k^T d_{k-1}}{\|g_k\|\|d_{k-1}\|},\\
\widehat{\gamma}_k^{(13)} &= 1 - \bar{\gamma}|g_k^T d_{k-1}|, & \widehat{\gamma}_k^{(14)} &= 1 + \bar{\gamma}|g_k^T d_{k-1}|,\\
\widehat{\gamma}_k^{(15)} &= 1 - \bar{\gamma} g_k^T d_{k-1}, & \widehat{\gamma}_k^{(16)} &= 1 + \bar{\gamma} g_k^T d_{k-1},
\end{aligned}$$

where $\bar{\gamma}$ is a nonnegative constant. Note that these 16 choices belong to four different types of formulae, because they can be defined by $\widehat{\gamma}_k^{(j+4\ell)}$, for $j = 1, 2, 3, 4$ and $\ell = 0, 1, 2, 3$.

Moreover, by taking (2.5) into account, γ_k can be regarded as a sizing parameter of the direction $-g_k$. Accordingly, we consider the following two choices of $\widehat{\gamma}_k$:

$$\widehat{\gamma}_k^{(17)} = \frac{\|s_{k-1}\|^2}{s_{k-1}^T y_{k-1}} \quad \text{or} \quad \widehat{\gamma}_k^{(18)} = \frac{s_{k-1}^T y_{k-1}}{\|y_{k-1}\|^2}.$$

Note that the search directions $d_k = -\widehat{\gamma}_k^{(17)} g_k$ and $d_k = -\widehat{\gamma}_k^{(18)} g_k$ define the Barzilai-Borwein method [2,10].

5 Numerical results

In this section, we report some numerical results obtained by applying the proposed methods to a set of standard test problems. The program was coded in C by modifying the software package CG-DESCENT of Hager and Zhang [15,17,18], which can be obtained from the website [14]. All computations were carried out on Lenovo G570 PC with Intel Core i5-2430M CPU (2.40GHz×2) and 8.0GB RAM. We run virtual Linux OS Ubuntu 11 on Windows 7 by using VMware Player 4.04, and assigned one processor and 5.9GB RAM to Ubuntu 11.

Our set of test problems consists of 132 tests which are used by Hager [14] and belong to the CUTEr library [3,13] for unconstrained optimization. The name codes of these tests and the values of their dimension n, which we used and ranged in [2,10000], are given in Table 1. Although Hager [14] considers 145 tests, we do not consider the remaining tests here due to the fact that the memory of our PC was insufficient for some of them and different local solutions were obtained when different solvers were applied to those omitted problems.

The methods which we consider for comparisons are mentioned in the second column of Table 2, where the first column consists of abbreviation names of these methods. We set $t = 1$, for GDL, and $\phi = 2$, for GHZ, GDPR and GDLS. We also used $\bar{\gamma}_1 = 0.01$, $\bar{\gamma}_2 = 100$, $\theta = 10^{-12}$ and $\bar{\gamma} = 0.8$ for the proposed methods. In the discussion below, we have added either number "1" or "2" immediately after the abbreviation name to indicate that either choice $p_k = g_k$ or $p_k = y_{k-1}$ is used, respectively (for example, GHS1 and GHS2 denote GHS with the latter two choices, respectively). Moreover, we used modification (3.7) with $\zeta_k = \nu_k^{(2)}$, defined in (3.6), for $\bar{\nu}_2 = 0.4$ so that condition (3.5) is ensured on all iterations of the methods.

As mentioned above, we have implemented all the methods under considerations on the basis of the software package CG-DESCENT (version 5.3). Although this version is not the most recent one, we used it for a fair comparison of the conjugate gradient methods. We will briefly describe the recent CG-DESCENT version 6.6 [19] at the end of this section.

For each iteration, a new point is computed such that the Wolfe conditions (2.7)–(2.8) are satisfied. If the condition

$$|f(x_k + \alpha_k d_k) - f(x_k)| \le \omega C_k,$$

where $\omega > 0$ is a small number and

$$\begin{aligned} C_k &= C_{k-1} + (|f(x_k)| - C_{k-1})/Q_k, & C_{-1} &= 0,\\ Q_k &= 1 + \Delta Q_{k-1}, & Q_{-1} &= 0, \end{aligned}$$

is satisfied, then we switch permanently the Wolfe conditions to the condition $f(x_k + \alpha_k d_k) \le f(x_k) + 10^{-6}|f(x_k)|$ and the approximate Wolfe conditions

$$-(1 - 2\delta) g_k^T d_k \ge g(x_k + \alpha_k d_k)^T d_k \ge \sigma_1 g_k^T d_k.$$

Table 1 Test problems (names & dimensions); Collected by CUTEr

Code	n	Code	n	Code	n	Code	n
AKIVA	2	DIXMAANE	3000	HEART8LS	8	PALMER7C	8
ALLINITU	4	DIXMAANF	3000	HELIX	3	PALMER8C	8
ARGLINA	200	DIXMAANG	3000	HIELOW	3	PENALTY1	1000
ARGLINB	200	DIXMAANH	3000	HILBERTA	2	PENALTY2	200
ARWHEAD	5000	DIXMAANI	3000	HILBERTB	10	POWELLSG	5000
BARD	3	DIXMAANJ	3000	HIMMELBB	2	POWER	10000
BDQRTIC	5000	DIXMAANK	15	HIMMELBF	4	QUARTC	5000
BEALE	2	DIXMAANL	3000	HIMMELBG	2	ROSENBR	2
BIGGS6	6	DIXON3DQ	10000	HIMMELBH	2	S308	2
BOX3	3	DJTL	2	HUMPS	2	SCHMVETT	5000
BRKMCC	2	DQDRTIC	5000	JENSMP	2	SENSORS	100
BROWNAL	200	DQRTIC	5000	KOWOSB	4	SINEVAL	2
BROWNBS	2	EDENSCH	2000	LIARWHD	5000	SINQUAD	5000
BROWNDEN	4	EG2	1000	LOGHAIRY	2	SISSER	2
BROYDN7D	5000	ENGVAL1	5000	MANCINO	100	SNAIL	2
BRYBND	5000	ENGVAL2	3	MARATOSB	2	SPARSINE	5000
CHNROSNB	50	ERRINROS	50	MEXHAT	2	SPARSQUR	10000
CLIFF	2	EXPFIT	2	MOREBV	5000	SPMSRTLS	4999
COSINE	10000	EXTROSNB	1000	MSQRTALS	1024	SROSENBR	5000
CRAGGLVY	5000	FLETCBV2	5000	MSQRTBLS	1024	STRATEC	10
CUBE	2	FLETCHCR	1000	NONCVXU2	5000	TESTQUAD	5000
CURLY10	10000	FMINSRF2	5625	NONDIA	5000	TOINTGOR	50
CURLY20	10000	FMINSURF	5625	NONDQUAR	5000	TOINTGSS	5000
DECONVU	63	FREUROTH	5000	OSBORNEA	5	TOINTPSP	50
DENSCHNA	2	GENHUMPS	5000	OSBORNEB	11	TOINTQOR	50
DENSCHNB	2	GENROSE	500	OSCIPATH	10	TQUARTIC	5000
DENSCHND	3	GROWTHLS	3	PALMER1C	8	TRIDIA	5000
DENSCHNE	3	GULF	3	PALMER1D	7	VARDIM	200
DENSCHNF	2	HAIRY	2	PALMER2C	8	VAREIGVL	50
DIXMAANA	3000	HATFLDD	3	PALMER3C	8	WATSON	12
DIXMAANB	3000	HATFLDE	3	PALMER4C	8	WOODS	4000
DIXMAANC	3000	HATFLDFL	3	PALMER5C	6	YFITU	3
DIXMAAND	3000	HEART6LS	6	PALMER6C	8	ZANGWIL2	2

We used the parameters values of $\delta = 0.1$, $\sigma_1 = 0.9$, $\omega = 10^{-3}$ and $\Delta = 0.7$. Note that the above line search procedure is the default procedure of CG-DESCENT (see [15,17,18], for detail). We stopped the algorithm if either

$$\|g_k\|_\infty \leq 10^{-6}$$

or the CPU time exceeded 500 seconds.

Table 2 Tested methods

CG-DESCENT	Software by Hager and Zhang [15,17,18]
GHS	Algorithm G3TCG with $\beta_k = \beta_k^{HS}$ defined in (1.2)
GPR	Algorithm G3TCG with $\beta_k = \beta_k^{PR}$ defined in (1.2)
GLS	Algorithm G3TCG with $\beta_k = \beta_k^{LS}$ defined in (1.2)
GDL	Algorithm G3TCG with $\beta_k = \beta_k^{DL}$ defined by (4.1)
GHZ	Algorithm G3TCG with $\beta_k = \beta_k^{HZ}$ defined by (4.2)
GDPR	Algorithm G3TCG with $\beta_k = \beta_k^{DPR}$ defined by (4.3)
GDLS	Algorithm G3TCG with $\beta_k = \beta_k^{DLS}$ defined by (4.4)

Table 3 Comparison of the proposed γ_k with various $\widehat{\gamma}_k^{(j)}$, where $j = 0$ implies $\gamma_k = 1$

j	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
GHS1	1	1	3	3	3	4	4	4	3	3	2	2	2	3	4	3	4
GHS2	3	2	3	3	3	4	4	4	4	3	3	3	3	4	4	4	3
HPR1	1	1	3	3	3	4	4	4	3	3	2	3	2	4	3	3	4
GPR2	2	1	3	3	3	4	4	4	4	3	3	2	3	4	4	4	4
GLS1	3	1	3	3	2	4	4	4	3	3	3	2	3	4	4	3	4
GLS2	2	1	3	3	3	4	4	4	4	3	3	1	3	4	4	4	4
GDL1	1	1	3	3	3	4	4	4	4	3	2	1	2	4	3	3	3
GDL2	3	2	3	3	3	4	4	4	4	3	3	2	3	4	4	4	4
GHZ1	2	2	3	3	3	3	4	3	4	3	3	3	3	3	4	4	4
GHZ2	3	3	3	3	3	4	4	4	4	3	3	2	4	4	4	4	4
GDPR1	3	4	3	3	3	4	3	3	4	3	2	1	3	3	3	4	4
GDPR2	3	3	3	3	3	4	4	4	4	3	3	3	3	4	4	4	4
GDLS1	3	2	3	3	1	4	4	4	4	3	3	3	2	4	4	4	3
GDLS2	3	3	3	3	3	4	4	4	4	3	3	3	3	4	4	4	4

We now investigate how the magnitude γ_k affects numerical performance of the proposed methods.We used the value of $\gamma_k = 1$, which was clearly better than the values 0.5 and 2, and observed the following. The methods with $\widehat{\gamma}_k^{(17)}$ and $\widehat{\gamma}_k^{(18)}$ perform very poorly. Thus we will not consider them in the comparison below. To summarize the comparison of the methods with the other choices for γ_k, we present Table 3. The 14 type of methods and 17 choices of γ_k are referred in the first column and first row of the table, respectively, so that the total number of methods under comparison is 238. The rest of the table consists of the four numbers 1, 2, 3 and 4 which we use to measure the level of the performance for each method as follows.

- Number "1" denotes that the method is superior to CG-DESCENT,
- Number "2" denotes that the method is almost comparable with CG-DESCENT,
- Number "3" denotes that the method is slightly outperform CG-DESCENT,
- Number "4" denotes that the method is inferior to CG-DESCENT.

We see from Table 3 that all the methods with the choices of $\widehat{\gamma}_k^{(4+i)}$ and $\widehat{\gamma}_k^{(12+i)}$, for $i = 1, 2, 3, 4$, do not perform well. This result suggests normalizing $\widehat{\gamma}_k$ (namely, dividing it by $\|g_k\|\|d_{k-1}\|$) so that good influence might be obtained. We also observe that the methods with the choice $\widehat{\gamma}_k^{(11)}$ perform reasonably well, but with the choice $\widehat{\gamma}_k^{(1)}$ yield the best performance. Therefore, we will give further comparison detail about the the latter choice for all the methods in Table 2.

For useful comparisons, we adopt the performance profiles of Dolan and Moré [8], which is based on the following. For n_s solvers and n_p problems, the performance profile $P : \mathbb{R} \to [0, 1]$ is defined as follows: Let $\mathcal{P}$ and $\mathcal{S}$ be the set of problems and the set of solvers, respectively. For each problem $p \in \mathcal{P}$ and for each solver $s \in \mathcal{S}$, we define $t_{p,s}$ = computing time (similarly for the number of iterations) required to solve problem p by solver s. The performance ratio is given by $r_{p,s} = t_{p,s}/\min_{s\in\mathcal{S}} t_{p,s}$. Then, the performance profile is defined by $P(\tau) = \frac{1}{n_p}\text{size}\{p \in \mathcal{P}|r_{p,s} \leq \tau\}$, for all $\tau > 0$, where size$A$, for any set A, stands for the number of the elements in that set. Note that $P(\tau)$ is the probability for solver $s \in \mathcal{S}$ such that a performance ratio $r_{p,s}$ is within a factor $\tau > 0$ of the best possible ratio. Note that $n_p = 132$ was used for obtaining the following comparison of figures.

In Figs. 2–4, we adopt the performance profiles based on the CPU time. In order to prevent a measurement error, we set the minimum of the measurement 0.2 seconds. Figures 2 and 3 compare CG-DESCENT with the proposed methods for $p_k = g_k$ and $p_k = y_{k-1}$, respectively. In Fig. 4, we give performance profiles of CG-DESCENT, GHS1, GHZ1, GDLS1, GHS2, GPR2 and GLS2 which perform better than those in Figs. 2 and 3.

We see from Figs. 2 and 3 that GHS1, GHZ1, GDLS1, GHS2, GPR2 and GDL2 perform better than CG-DESCENT. In particular, GHZ1 and GLS2 outperform the other methods. Note that in Fig. 3, the performance profiles for GHS2 and GPR2

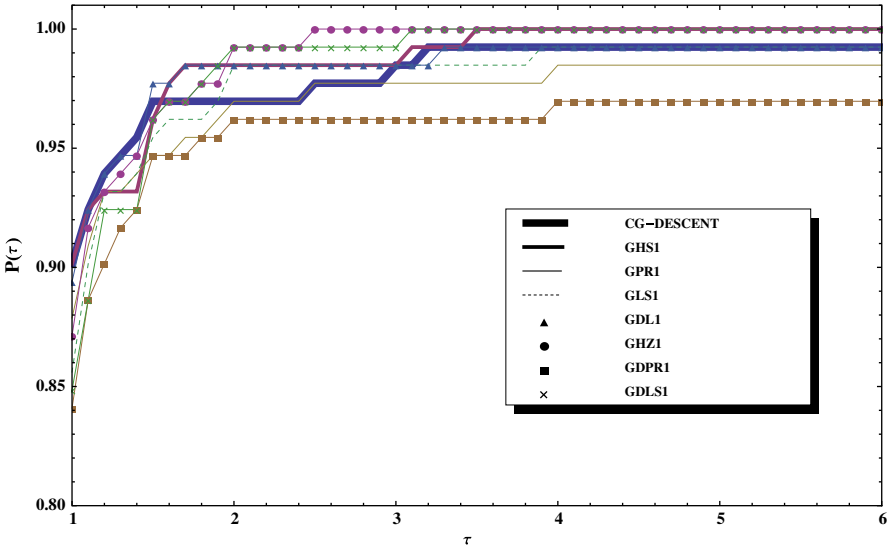

Fig. 2 CPU Performance profile of the methods with $p_k = g_k$

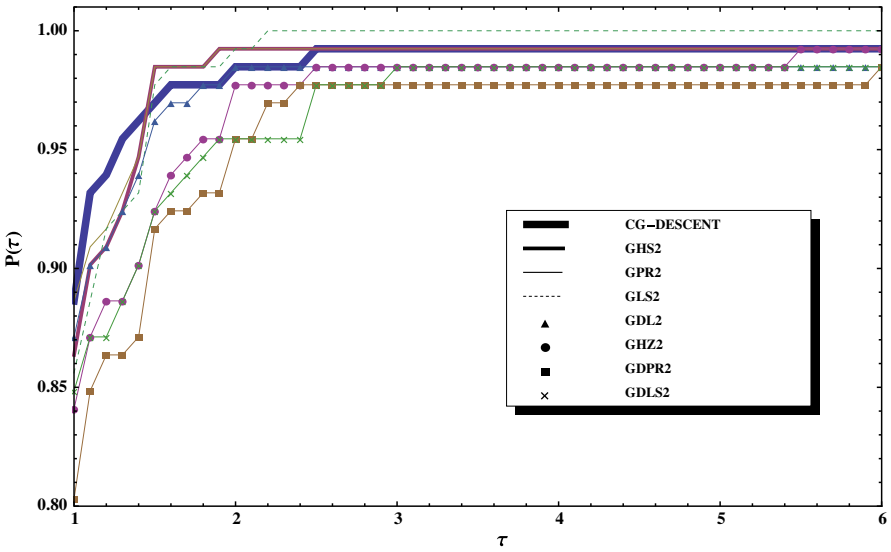

Fig. 3 CPU Performance profile of the methods with $p_k = y_{k-1}$

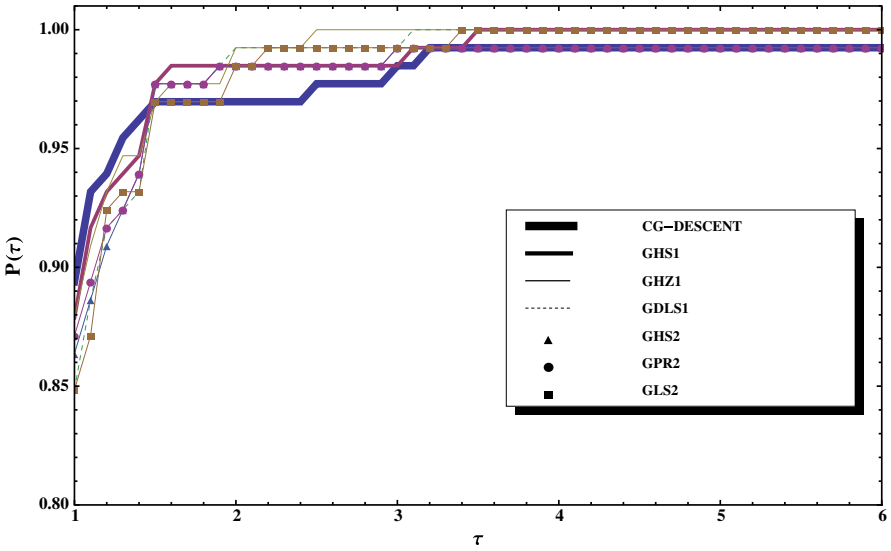

Fig. 4 CPU Performance profile of efficient methods

overlap each other. Figure 4 shows that GHZ1 is superior to the other methods, and the other methods are at least comparable with CG-DESCENT.

As mentioned above, the CG-DESCENT 6.6 version [19] is the latest one, which was superior to GHZ1. In this version, a subspace iteration and a preconditioning step are used. When $g_k \in \mathcal{S}_k$ where $\mathcal{S}_k = \text{Span}\{d_{k-1}, \ldots, d_{k-m}\}$ for some integer m, iterates may converge very slowly. In order to avoid this phenomenon, Hager and Zhang [19] consider the subspace minimization:

$$\min_{z \in \mathcal{S}_k} f(x_k + z). \tag{5.1}$$

If z_k is a solution of this problem and $x_{k+1} = x_k + z_k$, then by the first order optimality condition of (5.1), we have $g(x_{k+1})^T v = 0$ for all $v \in \mathcal{S}_k$. Then $g(x_{k+1}) \notin \mathcal{S}_k$ or $g(x_{k+1}) = 0$ holds. Moreover, they use the following preconditioned HZ method:

$$\begin{aligned} d_k &= -P_k g_k + \beta_k^+ d_k, \\ \beta_k^+ &= \max\left\{ \beta_k, \bar{\nu}_3 \frac{g_{k-1}^T d_{k-1}}{d_{k-1}^T P_k^{-1} d_{k-1}} \right\}, \\ \beta_k &= \frac{g_k^T P_k y_{k-1}}{d_{k-1}^T y_{k-1}} - \varphi \frac{y_{k-1}^T P_k y_{k-1}}{(d_{k-1}^T y_{k-1})^2} g_k^T d_{k-1}, \end{aligned} \tag{5.2}$$

where φ and $\bar{\nu}_3$ are constants such that $\varphi > 1/4$ and $\bar{\nu}_3 > 0$, P_k is a preconditioner matrix which is made by using information obtained in the subspace minimization, and P_k^{-1} is the pseudoinverse of P_k. The outline of the algorithm used in CG-DESCENT 6.6 is based on the following procedures, assuming ϑ_1 and ϑ_2 are positive constants such that $0 < \vartheta_1 < \vartheta_2 < 1$ and using $\mathrm{dist}\{x, \mathcal{S}_k\} = \inf\{\|y - x\| \mid y \in \mathcal{S}_k\}$.

Standard CG iteration. Perform Hager-Zhan's (HZ) conjugate gradient method which is the same as CG-DESCENT 5.3, as long as $\mathrm{dist}\{g_k, \mathcal{S}_k\} > \vartheta_1 \|g_k\|$. When $\mathrm{dist}\{g_k, \mathcal{S}_k\} \le \vartheta_1 \|g_k\|$ is satisfied, branch to the subspace iteration.

Subspace iteration. Solve the subspace problem (5.1) by using a quasi-Newton method. Stop at the iteration where $\mathrm{dist}\{g_{k+1}, \mathcal{S}_k\} \ge \vartheta_2 \|g_{k+1}\|$, and then branch to the preconditioning step.

Preconditioning step. The preconditioned HZ method (5.2) is performed, and return to the standard CG iteration.

Since we expect that the subspace iteration and the preconditioning step work efficiently for other conjugate gradient methods, we introduced these procedures to GHS1 and GHZ1. The resulting methods (referred to as GHS1_6.6 and GHZ1_6.6 respectively) differ from CG-DESCENT 6.6 in the following three points. First, in the standard CG iteration, we used the search direction (2.1) instead of Hager-Zhang's direction. Second, in the line search technique, we impose the generalized strong Wolfe conditions (2.7) and (2.9) with $\delta = 0.001$, $\sigma_1 = 0.2$ and $\sigma_2 = 0.6$, instead of the Wolfe condition (2.7)–(2.8). Third, in the preconditioning step, we use the preconditioned steepest descent direction (namely, a kind of quasi-Newton direction $d_k = -P_k g_k$), instead of the direction (5.2). In Fig. 5, we compare our good performed methods with CG-DESCENT 6.6. Note that the performance profiles of GHS1_6.6 and GHZ1_6.6 overlap each other in $\tau \in [3, 6]$. We see from the figure that these and CG-DESCENT 6.6 methods clearly outperform that of GHS1 and GHZ1. We also find that GHS1_6.6 is superior to the other method and GHZ1_6.6 is almost comparable with CG-DESCENT 6.6.

The above experiment results show that our method perform better than CG-DESCENT 5.3 by choosing the parameters β_k and γ_k suitably. In addition, the hybrid

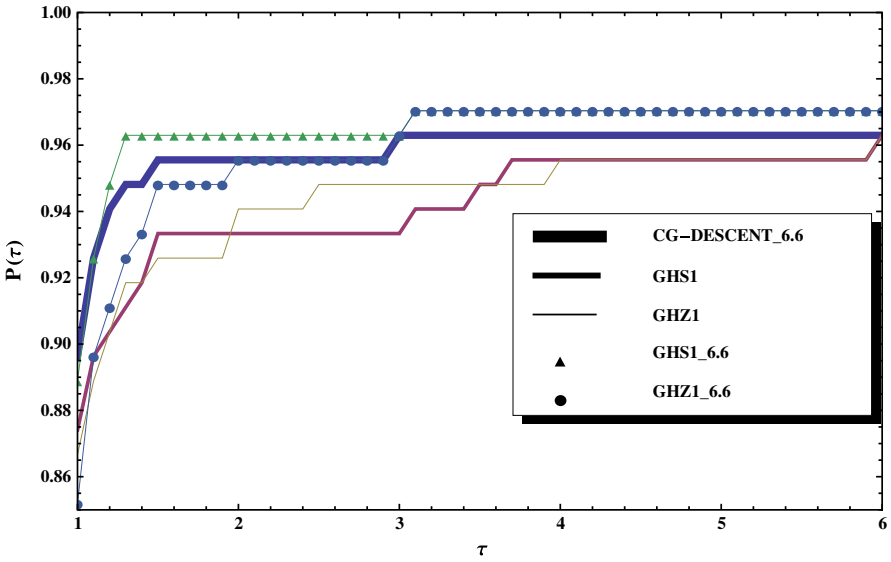

Fig. 5 CPU Performance profile of efficient methods with the subspace iteration

methods (namely, GHS1_6.6 and GHZ1_6.6) are at least comparable with the latest version of CG-DESCENT. It is worth noting that we have not discussed the global convergence of the hybrid method, because it is out of scope of this paper although it might be established with assumptions similar to that considered in [19]. Although we have not constructed away for choosing the best value of the parameter γ_k, the above experiments suggest considering further studies for obtaining useful choices for this parameter.

6 Conclusion

We have proposed a family of three-term conjugate gradient methods which control the parameter γ_k in the sufficient descent condition (2.3). It is shown that these methods have the global convergence property that certain useful conjugate gradient methods have for standard conditions on the objective function. The reported numerical results show the important roll of γ_k for the performance of the methods. In addition, we have compared some results to those required by the CG-DESCENT 5.3 method and have observed that a certain choice for γ_k yields a reasonably high performance algorithm. Moreover, we have incorporated an acceleration procedure of CG-DESCENT 6.6 into the proposed methods and observed well improved performance in practice.

Acknowledgments The authors would like to thank Prof. William W. Hager, the Editor-in-Chief of the journal, and the anonymous reviewers for valuable comments on a draft of this paper. We would also like to thank Prof. Yu-Hong Dai for providing his program code of conjugate gradient methods. The second and third authors are supported in part by the Grant-in-Aid for Scientific Research (C) 25330030 of Japan Society for the Promotion of Science

References

1. Al-Baali, M.: Descent property and global convergence of the Fletcher-Reeves method with inexact line search. IMA J. Numer. Anal. **5**, 121–124 (1985)
2. Barzilai, J., Borwein, J.M.: Two-point stepsize gradient methods. IMA J. Numer. Anal. **8**, 141–148 (1988)
3. Bongartz, I., Conn, A.R., Gould, N.I.M., Toint, P.L.: CUTE: constrained and unconstrained testing environments. ACM Trans. Math. Softw. **21**, 123–160 (1995)
4. Cheng, W.: A two-term PRP-based descent method. Numer. Funct. Anal. Optim. **28**, 1217–1230 (2007)
5. Dai, Y.-H., Kou, C.-X.: A nonlinear conjugate gradient algorithm with an optimal property and an improved Wolfe line search. SIAM J. Optim. **23**, 296–320 (2013)
6. Dai, Y.-H., Liao, L.Z.: New conjugacy conditions and related nonlinear conjugate gradient methods. Appl. Math. Optim. **43**, 87–101 (2001)
7. Dai, Y.-H., Yuan, Y.: A nonlinear conjugate gradient method with a strong global convergence property. SIAM J. Optim. **10**, 177–182 (1999)
8. Dolan, E.D., Moré, J.J.: Benchmarking optimization software with performance profiles. Math. Program. **91**, 201–213 (2002)
9. Fletcher, R.: Practical Methods of Optimization (Second Edition). Wiley, New York (1987)
10. Fletcher, R.: On the Barzilai-Borwein Method, Optimization and Control with Applications, Springer series in Applied Optimization, 96, 235–256, Springer, New York (2005)
11. Fletcher, R., Reeves, C.M.: Function minimization by conjugate gradients. Comput. J. **7**, 149–154 (1964)
12. Gilbert, J.C., Nocedal, J.: Global convergence properties of conjugate gradient methods for optimization. SIAM J. Optim. **2**, 21–42 (1992)
13. Gould, N.I.M., Orban, D., Toint, P.L.: CUTEr and SifDec: a constrained and unconstrained testing environment, revisited. ACM Trans. Math. Softw. **29**, 373–394 (2003)
14. Hager, W.W. http://people.clas.ufl.edu/hager/. Accessed 10 Mar 2014
15. Hager, W.W., Zhang, H.: A new conjugate gradient method with guaranteed descent and an efficient line search. SIAM J. Optim. **16**, 170–192 (2005)
16. Hager, W.W., Zhang, H.: A survey of nonlinear conjugate gradient methods. Pac. J. Optim. **2**, 35–58 (2006)
17. Hager W.W., Zhang, H.: CG_DESCENT Version 1.4 User's Guide, University of Florida (2005). http://people.clas.ufl.edu/hager/. Accessed 10 Mar 2014
18. Hager, W.W., Zhang, H.: Algorithm 851: CG_DESCENT, a conjugate gradient method with guaranteed descent. ACM Trans. Math. Softw. **32**, 113–137 (2006)
19. Hager, W.W., Zhang, H.: The limited memory conjugate gradient method. SIAM J. Optim. **23**, 2150–2168 (2013)
20. Hestenes, M.R., Stiefel, E.: Methods of conjugate gradients for solving linear systems. J. Res. Natl. Bur. Stand. **49**, 409–436 (1952)
21. Liu, Y., Storey, C.: Efficient generalized conjugate gradient algorithms, Part1: Theory. J. Optim. Theory. Appl. **69**, 129–137 (1991)
22. Narushima, Y., Yabe, H., Ford, J.A.: A three-term conjugate gradient method with sufficient descent property for unconstrained optimization. SIAM J. Optim. **21**, 212–230 (2011)
23. Nocedal, J., Wright, S.J.: Numerical Optimization, 2nd edn. Springer Series in Operations Research, Springer, New York (2006)
24. 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. **153**, 733–757 (2012)
25. Sorenson, H.W.: Comparison of some conjugate direction procedures for function minimization. J. Frankl. Inst. **288**, 421–441 (1969)
26. Yu, G., Guan, L., Li, G.: Global convergence of modified Polak-Ribiére-Polyak conjugate gradient methods with sufficient descent property. J. Ind. Manag. Optim. **4**, 565–579 (2008)
27. Zhang, L.: A new Liu-Storey type nonlinear conjugate gradient method for unconstrained optimization problems. J. Comput. Appl. Math. **225**, 146–157 (2009)
28. Zhang, L., Zhou, W., Li, D.H.: Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search. Numer. Math. **104**, 561–572 (2006)

29. Zhang, L., Zhou, W., Li, D.H.: A descent modified Polak-Ribière-Polyak conjugate gradient method and its global convergence. IMA J. Numer. Anal. **26**, 629–640 (2006)
30. Zhang, L., Zhou, W., Li, D.H.: Some descent three-term conjugate gradient methods and their global convergence. Optim. Methods Softw. **22**, 697–711 (2007)
31. Zoutendijk, G.: Nonlinear Programming, Computational Methods, in Integer and Nonlinear Programming. Abadie, J. (ed.) North-Holland, Amsterdam, 37–86, 1970