SCIENCE CHINA
Mathematics

• ARTICLES •

August 2016 Vol. 59 No. 8: 1511–1524
doi: 10.1007/s11425-016-0279-2

A Barzilai-Borwein conjugate gradient method

DAI YuHong[1,*] & KOU CaiXia[2]

[1]*LSEC, Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing* 100190*, China;*
[2]*School of Science, Beijing University of Posts and Telecommunications, Beijing* 100876*, China*

Email: dyh@lsec.cc.ac.cn, koucx@lsec.cc.ac.cn

Received November 7, 2015; accepted April 27, 2016; published online July 1, 2016

Abstract The linear conjugate gradient method is an *optimal* method for convex quadratic minimization due to the Krylov subspace minimization property. The proposition of limited-memory BFGS method and Barzilai-Borwein gradient method, however, heavily restricted the use of conjugate gradient method for large-scale nonlinear optimization. This is, to the great extent, due to the requirement of a relatively exact line search at each iteration and the loss of conjugacy property of the search directions in various occasions. On the contrary, the limited-memory BFGS method and the Barzilai-Bowein gradient method share the so-called *asymptotical one stepsize per line-search property*, namely, the trial stepsize in the method will asymptotically be accepted by the line search when the iteration is close to the solution. This paper will focus on the analysis of the subspace minimization conjugate gradient method by Yuan and Stoer (1995). Specifically, if choosing the parameter in the method by combining the Barzilai-Borwein idea, we will be able to provide some efficient Barzilai-Borwein conjugate gradient (BBCG) methods. The initial numerical experiments show that one of the variants, BBCG3, is specially efficient among many others without line searches. This variant of the BBCG method might enjoy the asymptotical one stepsize per line-search property and become a strong candidate for large-scale nonlinear optimization.

Keywords conjugate gradient method, subspace minimization, Barzilai-Bowein gradient method, line search, descent property, global convergence

MSC(2010) 49M37, 65K05, 90C30

Citation: Dai Y H, Kou C X. A Barzilai-Borwein conjugate gradient method. Sci China Math, 2016, 59: 1511–1524, doi: 10.1007/s11425-016-0279-2

1 Introduction

Conjugate gradient methods are a class of important methods for solving unconstrained optimization problem

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

especially if the dimension n is large. They are of the form

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \alpha_k \boldsymbol{d}_k, \tag{1.2}$$

where α_k is a stepsize obtained by a line search and $\boldsymbol{d}_k$ is the search direction defined by

$$\boldsymbol{d}_k = \begin{cases} -\boldsymbol{g}_k, & \text{for } k = 1, \\ -\boldsymbol{g}_k + \beta_k \boldsymbol{d}_{k-1}, & \text{for } k \geqslant 2. \end{cases} \tag{1.3}$$

*Corresponding author

In the above, β_k is the so-called conjugate gradient parameter and $\boldsymbol{g}_k$ denotes $\nabla f(\boldsymbol{x}_k)$. In general, if the objective function is quadratic, namely,

$$q(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^{\mathrm{T}} A \boldsymbol{x} + \boldsymbol{b}^{\mathrm{T}}\boldsymbol{x}, \quad \boldsymbol{x} \in \mathbb{R}^n, \tag{1.4}$$

where $A \in \mathbb{R}^{n\times n}$ is a symmetric and positive definite matrix and $\boldsymbol{b} \in \mathbb{R}^n$, and if the line search is exact, namely,

$$\alpha_k = \arg\min_{\alpha>0} f(\boldsymbol{x}_k + \alpha\, \boldsymbol{d}_k), \tag{1.5}$$

the choice of β_k should be such that the corresponding method (1.2)–(1.3) reduces to the linear conjugate gradient method and keeps the finite termination property. Some well-known choices for β_k are Fletcher and Reeves [10], Polak et al. [20, 21], Dai and Yuan [9], and Hestenes and Stiefel [15] ones, and are given by

$$\beta_k^{\mathrm{FR}} = \frac{\|\boldsymbol{g}_k\|^2}{\|\boldsymbol{g}_{k-1}\|^2}, \quad \beta_k^{\mathrm{PRP}} = \frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{g}_{k-1}\|^2}, \quad \beta_k^{\mathrm{DY}} = \frac{\|\boldsymbol{g}_k\|^2}{\boldsymbol{d}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, \quad \beta_k^{\mathrm{HS}} = \frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\boldsymbol{d}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, \tag{1.6}$$

respectively, where $\boldsymbol{y}_{k-1} = \boldsymbol{g}_k - \boldsymbol{g}_{k-1}$ and $\|\cdot\|$ means the Euclidean norm. In recent years, more efficient formulae have been proposed for the conjugate gradient parameter, including [6, 7, 13, 26] among many others. See [3, 14] for recent surveys on nonlinear conjugate gradient methods.

It is well known that the linear conjugate gradient method is an *optimal* method for convex quadratic minimization due to the Krylov subspace minimization property (for example, see [24]). The proposition of limited-memory BFGS method [17] and Barzilai-Borwein gradient method [1], however, heavily restricted the use of conjugate gradient method for large-scale nonlinear optimization. This is, to the great extent, due to the requirement of a relatively exact line search at each iteration and the loss of conjugacy property of the search directions in various occasions. The efficient conjugate gradient codes, either CG_Descent by Hager and Zhang [13] or CGOPT by Dai and Kou [6], require the calculations of approximately two stepsizes per line search. On the contrary, the limited-memory BFGS method and the Barzilai-Bowein gradient method share the so-called *asymptotical one stepsize per line search property*; namely, the trial stepsize in the method will asymptotically be accepted by the line search when the iteration is close to the solution (see [8, 16, 23]). Therefore the following question is intriguing: Does there exist some conjugate gradient method which also has the asymptotical one stepsize per line search property?

To provide a partial answer to this question, we are specially interested in the conjugate gradient method by Yuan and Stoer [25]. As it is known, a remarkable property of the linear conjugate gradient method is that, if the objective function is given in (1.4) and if the exact line search (1.5) is used, then for $k = 1, 2, \ldots$, each iteration $\boldsymbol{x}_{k+1}$ is the minimizer of the objective function over the Krylov subspace, namely,

$$\boldsymbol{x}_{k+1} = \arg\min\{q(\boldsymbol{x}) : \boldsymbol{x} \in \boldsymbol{x}_1 + \mathrm{Span}\{\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_k\}\}. \tag{1.7}$$

However, this result is not realistic for nonlinear objective functions in which situation inexact line searches are often utilized. In fact, even for quadratic objective functions, the conjugacy property of the search directions may be harmful if the line search is exact, as argued by Yuan and Stoer [25]. Based on these considerations, for the general unconstrained problem (1.1), Yuan and Stoer [25] proposed a conjugate gradient method that computes search directions by minimizing an approximate quadratic model in the two-dimensional subspace spanned by the current gradient and the previous search direction. As the essence of nonlinear optimization is to minimize some merit function by achieving a certain descent gradually, we think that the method by Yuan and Stoer [25] has its intrinsic advantage, but such a method has not received much attention since its proposition in 1995. To emphasize the importance of this method, we shall call this method simply by subspace minimization conjugate gradient (SMCG) method. In this paper, we shall give some new analysis of the SMCG method and focus on how to choose the parameter in the method. Specifically, if we choose the parameter in the method by combining the Barzilai-Borwein idea, we shall call the corresponding method by the BBCG method. The initial numerical experiments show that one of the BBCG variants, BBCG3, is specially efficient among some others without line searches. This implies that the BBCG method might enjoy the asymptotical one stepsize per line-search property and become a strong candidate for large-scale nonlinear optimization.

This paper is organized as follows. In the next section, we review the SMCG method proposed by Yuan and Stoer [25]. In Section 3, we present some new theoretical properties of the SMCG method. In Section 4, we consider how to choose the parameter ρ_k by combining the Barzilai-Borwein idea and present some preliminary numerical results. In Section 5, we discuss how to choose the parameter ρ_k by some other ideas. Conclusions and discussions are made in the last section.

2 A review of the SMCG method

In this section, we review the SMCG method, which was early due to Yuan and Stoer [25].

Assume that B_k is some approximation of the function Hessian at $\boldsymbol{x}_k$ and denote $\boldsymbol{s}_{k-1} = \alpha_{k-1}\boldsymbol{d}_{k-1} = \boldsymbol{x}_k - \boldsymbol{x}_{k-1}$. Consider the following quadratic approximate function:

$$q_k(\boldsymbol{d}) = \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{d} + \frac{1}{2}\boldsymbol{d}^{\mathrm{T}}B_k\boldsymbol{d} \tag{2.1}$$

in the two-dimensional subspace

$$\Omega_k = \mathrm{Span}\{\boldsymbol{g}_k, \boldsymbol{d}_{k-1}\}. \tag{2.2}$$

We are particularly interested in choosing the next search direction via the following subproblem:

$$\min_{\boldsymbol{d}\in\Omega_k} q_k(\boldsymbol{d}). \tag{2.3}$$

Consider the general case that $\boldsymbol{g}_k$ and $\boldsymbol{d}_{k-1}$ are not collinear, namely, $\dim(\Omega_k) = 2$. In this case, substituting $\boldsymbol{d} = \mu\,\boldsymbol{g}_k + \nu\,\boldsymbol{s}_{k-1}$ into (2.3) yields

$$\min_{(\mu,\nu)\in\mathbb{R}} \begin{pmatrix} \|\boldsymbol{g}_k\|^2 \\ \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \mu \\ \nu \end{pmatrix} + \frac{1}{2}\begin{pmatrix} \mu \\ \nu \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \boldsymbol{g}_k^{\mathrm{T}}B_k\boldsymbol{g}_k & \boldsymbol{g}_k^{\mathrm{T}}B_k\boldsymbol{s}_{k-1} \\ \boldsymbol{s}_{k-1}^{\mathrm{T}}B_k\boldsymbol{g}_k & \boldsymbol{s}_{k-1}^{\mathrm{T}}B_k\boldsymbol{s}_{k-1} \end{pmatrix} \begin{pmatrix} \mu \\ \nu \end{pmatrix}. \tag{2.4}$$

Remembering that B_k is an approximation to the Hessian $\nabla^2 f(\boldsymbol{x}_k)$ and because $\nabla^2 f(\boldsymbol{x}_k)\boldsymbol{s}_{k-1} \approx \boldsymbol{y}_{k-1}$, it is natural to ask B_k to satisfy the quasi-Newton equation

$$B_k\boldsymbol{s}_{k-1} = \boldsymbol{y}_{k-1}. \tag{2.5}$$

Assuming that the value

$$\rho_k \approx \boldsymbol{g}_k^{\mathrm{T}}B_k\boldsymbol{g}_k \tag{2.6}$$

has been estimated in some way, we consider the following prediction quadratic function:

$$q_{\mathrm{pred}}(\mu,\nu) = \begin{pmatrix} \|\boldsymbol{g}_k\|^2 \\ \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \mu \\ \nu \end{pmatrix} + \frac{1}{2}\begin{pmatrix} \mu \\ \nu \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \rho_k & \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1} \\ \boldsymbol{y}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k & \boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} \end{pmatrix} \begin{pmatrix} \mu \\ \nu \end{pmatrix} \tag{2.7}$$

and further, the prediction subproblem

$$\min_{(\mu,\nu)\in\mathbb{R}} q_{\mathrm{pred}}(\mu,\nu). \tag{2.8}$$

If ρ_k satisfies

$$\Delta_k = \rho_k\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2 > 0, \tag{2.9}$$

which is always true if the approximation Hessian B_k is positive definite in the convex quadratic case, it is easy to get the unique solution of (2.8),

$$\begin{pmatrix} \mu_k^* \\ \nu_k^* \end{pmatrix} = \frac{1}{\Delta_k}\begin{pmatrix} \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} - \boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\|\boldsymbol{g}_k\|^2 \\ \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\|\boldsymbol{g}_k\|^2 - \rho_k\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} \end{pmatrix}. \tag{2.10}$$

Consequently, the search direction $\boldsymbol{d}_k$ is given by

$$\begin{aligned}\boldsymbol{d}_k &= \mu_k^* \boldsymbol{g}_k + \nu_k^* \boldsymbol{s}_{k-1} \\ &= \frac{1}{\Delta_k}[(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} - \boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\|\boldsymbol{g}_k\|^2)\,\boldsymbol{g}_k + (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\|\boldsymbol{g}_k\|^2 - \rho_k\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})\,\boldsymbol{s}_{k-1}].\end{aligned} \tag{2.11}$$

One interesting property of the SMCG method is that, if the line search is exact, namely, $\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} = 0$, the above direction is parallel to the Hestenes-Stiefel conjugate gradient direction independently of the choice of ρ_k. As inexact line searches are usually used in practical calculations, two choices for ρ_k have been suggested by Yuan and Stoer [25]. The first one is

$$\rho_k = \frac{2(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}. \tag{2.12}$$

It makes $\cos^2\theta_k$ to take an average value $\frac{1}{2}$, where θ_k is the angle between the vector $B_k^{\frac{1}{2}}\boldsymbol{g}_k$ and $B_k^{\frac{1}{2}}\boldsymbol{s}_{k-1}$. The second one is

$$\rho_k = \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\left(\|\boldsymbol{g}_k\|^2 - \frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{\|\boldsymbol{s}_{k-1}\|^2}\right) + \frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, \tag{2.13}$$

which is derived by taking B_k to the scaled memoryless BFGS update, namely, setting

$$B_k = \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\left(I - \frac{\boldsymbol{s}_{k-1}\boldsymbol{s}_{k-1}^{\mathrm{T}}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{s}_{k-1}}\right) + \frac{\boldsymbol{y}_{k-1}\boldsymbol{y}_{k-1}^{\mathrm{T}}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}. \tag{2.14}$$

3 Analysis of the SMCG method

In this section, we present some basic theoretical properties of the SMCG method. At first, we have the following finite termination result.

Theorem 3.1. *Consider the SMCG method for the convex quadratic function* (1.4) *with* $n = 2$, *where* $A \in \mathbb{R}^{n\times n}$ *is a symmetric and positive definite matrix and* $\boldsymbol{b} \in \mathbb{R}^n$. *Assume that* $\boldsymbol{x}_1$ *is the starting point and that a Cauchy steepest descent step is taken at the first iteration. Then we must have that* $\boldsymbol{g}_j = 0$ *for some* $j \leqslant 4$.

Proof. Assume that $\boldsymbol{g}_j \neq 0$ for $j = 1, 2, 3$. Since the first step is a Cauchy steepest descent step, we must have

$$\boldsymbol{g}_2^{\mathrm{T}}\boldsymbol{s}_1 = 0. \tag{3.1}$$

By the definition of $\boldsymbol{s}_2$, for any value of ρ_2, we know that $\boldsymbol{s}_2$ is A-conjugate to $\boldsymbol{s}_1$, namely,

$$\boldsymbol{s}_2^{\mathrm{T}}A\boldsymbol{s}_1 = 0, \tag{3.2}$$

which with $\boldsymbol{y}_2 = A\boldsymbol{s}_2$ shows that

$$\boldsymbol{y}_2^{\mathrm{T}}\boldsymbol{s}_1 = 0. \tag{3.3}$$

Since the dimension of the problem is only 2 and $\boldsymbol{y}_2 = \boldsymbol{g}_3 - \boldsymbol{g}_2$, we know by (3.1) and (3.3) that $\boldsymbol{g}_2$, $\boldsymbol{g}_3$ and $\boldsymbol{y}_2$ are collinear and there must exist some real number $a \neq 0$ such that

$$\boldsymbol{y}_2 = a\,\boldsymbol{g}_3. \tag{3.4}$$

Now we look at $\boldsymbol{s}_3 = \mu_3\boldsymbol{g}_3 + \nu_3\boldsymbol{s}_2$. By the definitions in (2.10), we have that

$$\mu_3^* = \frac{\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{y}_2\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{s}_2 - \boldsymbol{s}_2^{\mathrm{T}}\boldsymbol{y}_2\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{g}_3}{\Delta_3} = \frac{a\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{g}_3\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{s}_2 - a\,\boldsymbol{s}_2^{\mathrm{T}}\boldsymbol{g}_3\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{g}_3}{\Delta_3} = 0 \tag{3.5}$$

and

$$\nu_3^* = -\frac{\rho_3\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{s}_2 - \boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{y}_2\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{g}_3}{\rho_3\,\boldsymbol{s}_2^{\mathrm{T}}\boldsymbol{y}_2 - (\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{y}_2)^2} = -\frac{\rho_3\,\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{s}_2 - a\,(\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{g}_3)^2}{a\,\rho_3\,\boldsymbol{s}_2^{\mathrm{T}}\boldsymbol{g}_3 - a^2\,(\boldsymbol{g}_3^{\mathrm{T}}\boldsymbol{g}_3)^2} = -\frac{1}{a}. \tag{3.6}$$

Thus $\boldsymbol{s}_3 = -\frac{1}{a}\,\boldsymbol{s}_2$. Therefore by this, $\boldsymbol{y}_2 = A\boldsymbol{s}_2$ and (3.4), we obtain

$$\boldsymbol{g}_4 = \boldsymbol{g}_3 + \boldsymbol{y}_3 = \boldsymbol{g}_3 + A\boldsymbol{s}_3 = \boldsymbol{g}_3 - \frac{1}{a}\,A\boldsymbol{s}_2 = \boldsymbol{g}_3 - \frac{1}{a}\,\boldsymbol{y}_2 = 0, \tag{3.7}$$

which concludes our proof. □

Remark 3.2. The above theorem can be extended to the case that an exact line search is carried out at any iteration. This is because in this case, the exactness of the line search implies that

$$\boldsymbol{g}_{k+1}^{\mathrm{T}}\boldsymbol{s}_k = 0. \tag{3.8}$$

The definition of $\boldsymbol{s}_{k+1}$ implies that $\boldsymbol{s}_{k+1}^{\mathrm{T}}A\boldsymbol{s}_k = 0$, which means that $\boldsymbol{y}_{k+1}^{\mathrm{T}}s_k = 0$. Since the problem dimension is only $n = 2$, we know that $\boldsymbol{g}_{k+1}$, $\boldsymbol{g}_{k+2}$ and $\boldsymbol{y}_k$ are collinear. Therefore we may again assume that $\boldsymbol{y}_{k+1} = a\,\boldsymbol{g}_{k+2}$ and then obtain the similar statement. This exposes the special property of the SMCG method.

Now, substituting (2.11) into the objective function in the prediction model (2.8), we can obtain the predicted function reduction,

$$\Delta f_{\mathrm{pred},k} = \frac{1}{2}\left[\frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} + \frac{(\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}\,(\rho_k\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2)}\right]. \tag{3.9}$$

In the case of convex quadratic objective function, we know that the prediction model is an approximation to the real model

$$\min_{(\mu,\nu)\in\mathbb{R}} q_{\mathrm{real}}(\mu,\nu), \tag{3.10}$$

where

$$q_{\mathrm{real}}(\mu,\nu) = \begin{pmatrix} \|\boldsymbol{g}_k\|^2 \\ \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \mu \\ \nu \end{pmatrix} + \frac{1}{2}\begin{pmatrix} \mu \\ \nu \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k & \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1} \\ \boldsymbol{y}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k & \boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} \end{pmatrix} \begin{pmatrix} \mu \\ \nu \end{pmatrix}. \tag{3.11}$$

Noticing that

$$q_{\mathrm{real}}(\mu,\nu) = q_{\mathrm{pred}}(\mu,\nu) + \frac{1}{2}\begin{pmatrix} \mu \\ \nu \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} \boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k - \rho_k & \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1} \\ \boldsymbol{y}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k & \boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} \end{pmatrix} \begin{pmatrix} \mu \\ \nu \end{pmatrix}, \tag{3.12}$$

we can obtain the real function reduction achieved by the direction (2.11),

$$\begin{aligned}\Delta f_{\mathrm{real},k} &= \Delta f_{\mathrm{pred},k} + \frac{1}{2}\frac{(\rho_k - \boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k)(\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{(\rho_k\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2)^2} \\ &= \Delta f_{\mathrm{real},k}^{(1)} + \Delta f_{\mathrm{real},k}^{(2)},\end{aligned} \tag{3.13}$$

where

$$\Delta f_{\mathrm{real},k}^{(1)} = \frac{1}{2}\frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} \tag{3.14}$$

and

$$\Delta f_{\mathrm{real},k}^{(2)} = \frac{1}{2}\frac{(\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{(\rho_k\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2)\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}(1+\tau_k), \tag{3.15}$$

where

$$\tau_k = \frac{(\rho_k - \boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k)\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\rho_k\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}. \tag{3.16}$$

From the relation (3.13), we can see that the real function reduction, $\Delta f_{\mathrm{real},k}$, achieved by the direction (2.11) can be divided into two parts. If the directional derivative of the current gradient $\boldsymbol{g}_k$ along the previous search $\boldsymbol{s}_{k-1}$, namely, $\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}$, is not so close to zero, then the first part $\Delta f_{\mathrm{real},k}^{(1)}$ is not so small and the SMCG method will gain enough remainder descent along the previous direction. Otherwise, if the absolute value of $\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}$ is small, we can show that the second part $\Delta f_{\mathrm{real},k}^{(2)}$ will not be so small and the SMCG method will gain enough descent as well. This property comes from the basic construction of the SMCG method. Following this line, we are able to establish the following linear convergence result.

Theorem 3.3. *Consider the SMCG method for the convex quadratic function given by* (1.4). *Suppose that*

$$\max\left\{\boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k, \frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}\right\} \leqslant \rho_k \leqslant M\,\|\boldsymbol{g}_k\|^2 \tag{3.17}$$

for some number $M < +\infty$ and all $k \geqslant 1$. Then we have that

$$\Delta f_k^{\mathrm{real}} \geqslant \frac{1}{2(\sqrt{\|A\|_2}+\sqrt{M})^2}\|\boldsymbol{g}_k\|^2, \quad \textit{for all } k \geqslant 1. \tag{3.18}$$

Therefore the method is globally convergent and the convergence is Q-linear in the objective function value.

Proof. Denote the constant

$$c = \frac{1}{1+\sqrt{\frac{M}{\|A\|_2}}} \in (0,1). \tag{3.19}$$

We divide into two cases:

$$|\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}| \geqslant c\,\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} \tag{3.20}$$

and

$$|\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}| < c\,\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}. \tag{3.21}$$

In the case of (3.20), noticing that $|\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}| \leqslant \|\boldsymbol{g}_k\|\,\|\boldsymbol{y}_{k-1}\|$ and $\frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{y}_{k-1}\|_2} \geqslant \frac{1}{\|A\|_2}$, we have that

$$\Delta f_{\mathrm{real},k} \geqslant \Delta f_{\mathrm{real},k}^{(1)} \geqslant \frac{c^2\,\|\boldsymbol{g}_k\|^4\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{2(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2} \geqslant \frac{c^2\,\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{2\|\boldsymbol{y}_{k-1}\|^2} \geqslant \frac{c^2}{2\|A\|_2}\|\boldsymbol{g}_k\|^2. \tag{3.22}$$

In the case of (3.21), we have that

$$\begin{aligned}\Delta f_{\mathrm{real},k} \geqslant \Delta f_{\mathrm{real},k}^{(2)} &\geqslant \frac{(\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{2(\rho_k\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2)\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}\\ &\geqslant \frac{(1-c)^2(\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}{2\,M\,\|\boldsymbol{g}_k\|^2\,(\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1})^2} \geqslant \frac{(1-c)^2}{2\,M}\,\|\boldsymbol{g}_k\|^2.\end{aligned} \tag{3.23}$$

Substituting the value of c in (3.19) into (3.22) and (3.23), we know the truth of (3.18).

Denote the constant $c_1 = \frac{1}{2(\sqrt{\|A\|_2}+\sqrt{M})^2}$. Summing (3.18) for $k = 1, \ldots, K$, we get that

$$f(\boldsymbol{x}_1) - f(\boldsymbol{x}_{K+1}) = \sum_{k=1}^{K}\Delta f_{\mathrm{real},k} \geqslant c_1\sum_{k=1}^{K}\|\boldsymbol{g}_k\|^2. \tag{3.24}$$

Letting K tend to infinity and noticing that $\lim_{k\to\infty} f(\boldsymbol{x}_{K+1}) > -\infty$, we can get from the above relation that

$$\sum_{k\geqslant 1}\|\boldsymbol{g}_k\|^2 < +\infty, \tag{3.25}$$

which implies that $\lim_{k\to\infty}\|\boldsymbol{g}_k\| = 0$ and hence the method is globally convergent.

Furthermore, the relation (3.18) can be rewritten as

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}_{k+1}) \geqslant c_1\|\boldsymbol{g}_k\|^2. \tag{3.26}$$

In addition, for the convex quadratic function in (1.4), it is easy to verify that $f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) = \boldsymbol{g}_k^{\mathrm{T}}A^{-1}\boldsymbol{g}_k$, where $\boldsymbol{x}^*$ is the solution of (1.4). Hence we have for all $k \geqslant 1$,

$$f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*) \leqslant c_2\|\boldsymbol{g}_k\|^2, \tag{3.27}$$

where $c_2 = \frac{1}{\lambda_{\min}(A)}$ and $\lambda_{\min}(A)$ is the minimal eigenvalue of the matrix A. Combining (3.26) and (3.27), we can obtain

$$[f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*)] - [f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}^*)] \geqslant c_1\|\boldsymbol{g}_k\|^2 \geqslant \frac{c_1}{c_2}[f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*)]. \tag{3.28}$$

It follows that

$$f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}^*) \leqslant \left(1 - \frac{c_1}{c_2}\right)[f(\boldsymbol{x}_k) - f(\boldsymbol{x}^*)], \tag{3.29}$$

which implies that the method is Q-linear in the objective function value. □

If ρ_k satisfies (3.17), we can prove that the direction $\boldsymbol{d}_k$ satisfies the sufficient descent condition

$$\boldsymbol{d}_k^{\mathrm{T}}\boldsymbol{g}_k \leqslant -\frac{1}{M}\|\boldsymbol{g}_k\|^2. \tag{3.30}$$

As a matter of fact, we have by direct calculations that

$$\boldsymbol{d}_k^{\mathrm{T}}\boldsymbol{g}_k = -\frac{\|\boldsymbol{g}_k\|^4}{\Delta_k}\left[\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - 2\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\|\boldsymbol{g}_k\|^2} + \rho_k\left(\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\|\boldsymbol{g}_k\|^2}\right)^2\right]. \tag{3.31}$$

Denote the term in the square brackets of (3.31) by η_k and treat it as a one variable function of $\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\|\boldsymbol{g}_k\|^2}$. By taking minimization, we can get that

$$\eta_k \geqslant \frac{\Delta_k}{\rho_k}. \tag{3.32}$$

Thus by this and (3.31), we obtain

$$\boldsymbol{d}_k^{\mathrm{T}}\boldsymbol{g}_k \leqslant -\frac{\|\boldsymbol{g}_k\|^4}{\rho_k}. \tag{3.33}$$

The above relation and (3.17) implies the truth of the sufficient descent condition (3.30). The sufficient descent condition is quite useful for the extension of the SMCG method to general nonlinear optimization. Nevertheless, we shall not go further in this issue in this paper.

Now let us focus on the condition (3.17). For the convex quadratic case, we have that $\boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k \leqslant \|A\|_2\|\boldsymbol{g}_k\|^2$ and $\frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} \leqslant \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}\|\boldsymbol{g}_k\|^2 \leqslant \|A\|_2\|\boldsymbol{g}_k\|^2$. Thus the condition (3.17) is valid provided that $M \geqslant \|A\|_2$. For general nonlinear optimization, we need pay special attention on this condition so that there exists some suitable ρ_k.

Another remark related to Theorem 3.3 is as follows. Although the convergence theorem requires that the quantity $\frac{\rho_k}{\|\boldsymbol{g}_k\|^2}$ is bounded (see the relation (3.17)), we do not need to impose this condition on our algorithms. This is a good property comparing with the gradient method $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k\boldsymbol{g}_k$ with constant stepsize $\alpha_k \equiv \frac{1}{2L}$, where L is the Lipschitz constant. It is well known that if the estimate to L is too large, the method will be very slow; otherwise, if the estimate to L is too small, the method may not converge at all. One intuitive explanation of such an advantage of the SMCG method is that, if it does not provide a good estimate to the curvature along the gradient $-\boldsymbol{g}_k$, the method will provide a compensation along the direction in the next iteration and receive a certain descent in the function value due to the two-dimensional subspace minimization mechanism.

4 The Barzilai-Borwein conjugate gradient method

In this section, we discuss how to choose the parameter ρ_k in the SMCG method by combining the Barzilai-Borwein idea.

4.1 The Barzilai-Borwein gradient method

To find a new steplength for α_k in the gradient method, which is of the form $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{g}_k$, Barzilai and Borwein [1] regards $D_k = (1/\alpha_k)\, I$ as some approximation to the Hessian and asks D_k to satisfy certain quasi-Newton property

$$\min_{\alpha_k \in \mathbb{R}} \|D_k \boldsymbol{s}_{k-1} - \boldsymbol{y}_{k-1}\|_2 \tag{4.1}$$

or

$$\min_{\alpha_k \in \mathbb{R}} \|\boldsymbol{s}_{k-1} - D_k^{-1} \boldsymbol{y}_{k-1}\|_2. \tag{4.2}$$

The solutions to (4.1) and (4.2) are

$$\alpha_k^{\mathrm{BB1}} = \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{s}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{y}_{k-1}} \tag{4.3}$$

and

$$\alpha_k^{\mathrm{BB2}} = \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{y}_{k-1}}{\|\boldsymbol{y}_{k-1}\|^2}, \tag{4.4}$$

respectively. To sum up, we see that the essence of the idea of Barzilai and Borwein [1] is to approximate the Hessian by the matrix $(1/\alpha_k^{\mathrm{BB1}})\, I$ or $(1/\alpha_k^{\mathrm{BB2}})\, I$.

Comparing with the steepest descent (SD) method, which was due to Cauchy in 1847, the Barzilai-Borwein (BB) method often requires less computational work and speeds up the convergence greatly. Due to its simplicity and efficiency, the BB method has been extended or generalized in many occasions or applications. For example, Raydan [23] designed an efficient global Barzilai and Borwein algorithm for unconstrained optimization by incorporating the nonmonotone line search by Grippo et al. [11]. It is observed in [23] that the BB stepsize can always be accepted by the line search near the solution (see the third paragraph in Section 4 of the paper).

Several attentions have been paid to theoretical properties of the BB method in spite of the potential difficulties due to its heavy nonmonotone behaviour. For two-dimensional convex quadratic functions, Barzilai and Borwein [1] presented an interesting R-superlinear convergence result for their method. For general n-dimensional strong convex quadratic functions, the BB method is also convergent (see [22]) and the convergence rate is R-linear (see [8]). Furthermore, Dai and Liao [8] presented a theoretical analysis for why the Barzilai-Borwein gradient method enjoys the asymptotical one stepsize per line search property. More theoretical analysis and applications of the Barzilai-Borwein gradient method can be found in Dai and Fletcher [5], Dai [4] and Hager et al. [12].

4.2 Choices of ρ_k based on the Barzilai-Borwein idea

To estimate the value of ρ_k in (2.6), we shall incorporate the idea of Barzilai-Borwein [1] in the SMCG method and call the resulted method by BBCG method. As seen from the previous subsection, the essence of the idea of Barzilai and Borwein [1] is to approximate the Hessian by the matrix $(1/\alpha_k^{\mathrm{BB1}})\, I$ or $(1/\alpha_k^{\mathrm{BB2}})\, I$. To incorporate the idea of Barzilai and Borwein [1] in the SMCG method, we then consider to approximate the Hessian B_k by $(1/\alpha_k^{\mathrm{BB1}})\, I$ or $(1/\alpha_k^{\mathrm{BB2}})\, I$ in estimating the value ρ_k in (2.6). This leads the following choices of ρ_k,

$$\rho_k^{\mathrm{BBCG1}} = \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{y}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{s}_{k-1}} \|\boldsymbol{g}_k\|^2 \tag{4.5}$$

and

$$\rho_k^{\mathrm{BBCG2}} = \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{y}_{k-1}} \|\boldsymbol{g}_k\|^2. \tag{4.6}$$

One good property of the formula (4.6) for ρ_k is that, if $\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{y}_{k-1} > 0$, the relation (2.9) is always satisfied with this choice except the vectors $\boldsymbol{g}_k$ and $\boldsymbol{y}_{k-1}$ are collinear. Nevertheless, our numerical studies in Subsection 3.3 show that the formula (4.5) is clearly superior to (4.6).

To go a step further, we may introduce a parameter $\lambda_k \geqslant 1$ into (4.6) and obtain the following choice of ρ_k,

$$\rho_k^{\mathrm{BBCG3}} = \lambda_k \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} \|\boldsymbol{g}_k\|^2. \tag{4.7}$$

As the parameter λ_k increases, we can see that relation (2.9) still holds and the Hessian of the objective function (2.7) will become more positive definite. We think that the prediction model (2.8) requires to act at least two roles: One is to approximate the original function in the two-dimensional subspace spanned by $-\boldsymbol{g}_k$ and $\boldsymbol{d}_{k-1}$ and the other is to be such that the search direction $\boldsymbol{d}_k$ obtained by the prediction model is downhill so that the original function can receive a descent along such a direction. Since a value of λ_k is a little larger than one can balance the two roles in a better way, we feel that this will make the method perform better. This turned out to be true by our numerical experiments in Subsection 3.3 and we shall choose

$$\lambda_k \equiv \frac{3}{2}. \tag{4.8}$$

We will find that the choice of ρ_k in (4.7) with λ_k given in (4.8) outperforms the choice of ρ_k in (4.5).

4.3 Numerical experiments

We tested the BBCG method with Matlab (version 7.9.0) and compared it with some other methods. For a given dimension n, we generated the matrix A by $A = A_1^{\mathrm{T}} A_1$ with $A_1 = 100(\mathrm{rand}(n,n) - 0.5)$ and the vector $\boldsymbol{b} = 100(\mathrm{rand}(n,1) - 0.5)$. The initial point is generated by $\boldsymbol{x}_1 = \mathrm{rand}(n,1)$. The Cauchy steepest descent stepsize is used at the first iteration for all the methods. The stopping condition is

$$\|\boldsymbol{g}_k\|_2 \leqslant 10^{-6}\|\boldsymbol{g}_1\|_2. \tag{4.9}$$

Ten simulations were made for a given dimension. For each simulation, we marked with the symbol "DGT" if the method is divergent and the gradient norm grows to infinity, or "Failed" if the number of iterations exceeds 50000. Otherwise, we listed down the required number of iterations in the numerical tables.

For the first numerical comparison, we fix the dimension $n = 20$. In Table 1, YS1, YS2, BBCG1, BBCG2 and BBCG3 are corresponding to the SMCG method with ρ_k being chosen by (2.12), (2.13) and (4.5)–(4.7), respectively. The choices (5.5) and (5.8) will be given in the next section. SCG stands for the spectral conjugate gradient method by Birgin and Martínez [2], who first tried the combination of the Barzilai-Borwein gradient method and the conjugate gradient method. More exactly, they considered the direction $\boldsymbol{d}_k = -\alpha_k^{\mathrm{BB1}} \boldsymbol{g}_k + \beta_k \boldsymbol{s}_{k-1}$, where $\beta_k = \frac{(\alpha_k^{\mathrm{BB1}} \boldsymbol{y}_{k-1} - \boldsymbol{s}_{k-1})^{\mathrm{T}} \boldsymbol{g}_k}{\boldsymbol{s}_{k-1}^{\mathrm{T}} \boldsymbol{y}_{k-1}}$, which satisfies that $\boldsymbol{d}_k^{\mathrm{T}} \boldsymbol{y}_{k-1} = -\boldsymbol{g}_k^{\mathrm{T}} \boldsymbol{s}_{k-1}$. BB denotes the Barzilai-Borwein gradient method with the stepsize given by (4.5), since this choice is generally preferred over (4.6) in numerical experiments. No line searches are carried for each method. We generated ten test problems. The number of the required iterations by each method is listed in Table 1. From Table 1, we can see that YS2, SCG and BBCG2 cannot provide an inexact solution in case of no line searches. Although the performance of YS2 is not so good at all, its improvement, BBCG3, performs much better than YS1 and BB.

In the second numerical experiment, we compare the BBCG3 method and the BB method for higher-dimensional problems. Specifically, we choose $n = 50$, $n = 200$ and $n = 500$. Again, ten simulations are made for a given dimension. The numerical results are listed in Table 2. From the table, we can see that the BBCG3 method requires fewer iterations for almost all the testing problems.

For the BBCG3 method, since the ρ_k in (4.7) with λ_k given in (4.8) does not satisfy the condition (3.17), the function value may not be decreasing at every iteration. In other words, the BBCG3 method is a non-monotone method. In the convex quadratic case, it is easy for us to enforce a descent in the objective function by an exact line search. More exactly, assuming that the current iteration is $\boldsymbol{x}_k$ with function value f_k and gradient $\boldsymbol{g}_k$ and the step is $\boldsymbol{s}_k$, if $f(\boldsymbol{x}_k + \boldsymbol{s}_k)$ is not less than f_k, then we take the point $\boldsymbol{x}_k + \alpha_k^* \boldsymbol{s}_k$ with $\alpha_k^* = \frac{-\boldsymbol{g}_k^{\mathrm{T}} \boldsymbol{s}_k}{\boldsymbol{s}_k^{\mathrm{T}} A \boldsymbol{s}_k}$ being the unique minimizer of the line search function $f(\boldsymbol{x}_k + \alpha \boldsymbol{s}_k)$. We denote this modification of BBCG3 by BBCG3$_{ls}$. In this case, we choose $n = 1000$, $n = 2000$ and

$n = 3000$. Again, ten simulations are made for a given dimension. The numerical results are listed in Table 3. For BBCG3$_{ls}$, the number of the required line searches is taken down in the column "nls". The version 8.3.0.532 of MATLAB is used for this test. From Table 3, we can see that for quite many problems (though not all), BBCG3 generates a descent in the objective function at each iteration and hence has the same performance as BBCG3$_{ls}$. For the other testing problems, BBCG3 is slightly better than BBCG3$_{ls}$.

Table 1 A numerical experience with Naive methods (without line searches)

P	YS1	YS2	SCG	BBCG1	BBCG2	BBCG3	(5.5)	(5.8)	BB
1	318	DGT	DGT	724	DGT	192	671	195	227
2	135	DGT	DGT	300	DGT	122	258	148	159
3	1269	DGT	DGT	3508	DGT	481	2887	910	524
4	1244	DGT	DGT	3115	DGT	290	2578	978	857
5	269	DGT	DGT	839	DGT	226	695	289	221
6	1856	DGT	DGT	16573	DGT	911	12338	4166	4705
7	661	DGT	DGT	1165	DGT	420	1309	634	388
8	1872	DGT	DGT	5812	DGT	506	5015	1310	3519
9	2037	DGT	DGT	3946	DGT	365	1871	868	582
10	151	273	DGT	319	DGT	108	260	134	117

Table 2 Comparing BBCG3 with BB

	$n = 50$		$n = 200$		$n = 500$	
P	BBCG3	BB	BBCG3	BB	BBCG3	BB
1	372	1045	530	541	3731	13044
2	241	529	677	1451	1600	3638
3	265	406	1437	8191	2131	1329
4	577	634	889	1621	1233	684
5	721	Failed	740	1450	1344	1971
6	1020	19166	1608	16756	1326	2981
7	311	619	1106	1746	1278	1622
8	1811	10781	1609	13081	1554	654
9	704	1261	2604	36132	2900	3594
10	590	15174	1443	7859	1668	2754

Table 3 Comparing BBCG3 without and with line searches

	$n = 1000$			$n = 2000$			$n = 3000$		
P	BBCG3	BBCG3$_{\rm ls}$	nls	BBCG3	BBCG3$_{\rm ls}$	nls	BBCG3	BBCG$_{\rm ls}$	nls
1	1669	1669	0	6472	6472	0	10994	10991	81
2	3618	3618	0	3950	3950	0	10162	10162	0
3	3051	3051	0	8409	8833	438	11825	12028	1059
4	10390	13474	3847	4071	4071	0	4805	4805	0
5	4926	4926	0	10219	10219	0	13682	12934	271
6	2650	2650	0	5439	5439	0	11824	11824	0
7	2483	2483	0	5011	5011	0	7341	7341	0
8	2607	2607	0	5451	5451	0	11304	11304	0
9	5186	5186	0	4054	4054	0	6315	6315	0
10	2876	2876	0	4453	4453	0	11314	11171	222

5 More choices of parameter ρ_k in the SMCG method

In this section, we investigate two other choices of the parameter ρ_k in the SMCG method. The first one is such that the two-dimensional quadratic termination holds without any exact line searches. The second one is derived so that it has some kind of optimal property. However, as seen from Table 1, neither of them outperforms the BBCG3 method.

5.1 A choice of ρ_k based on two-dimensional quadratic termination

The two-dimensional quadratic termination of the SMCG method, exposed in Theorem 3.1, relied on the fact that one exact line search is used. In this subsection, we shall propose a choice of the parameter ρ_k so that the exact line search is not a necessity. To do so, we give the following lemma.

Lemma 5.1. *Suppose that A is a matrix in $\mathbb{R}^{n\times n}$ and $\boldsymbol{g}_k$ and $\boldsymbol{s}_{k-1}$ are two non-collinear vectors in $\mathbb{R}^n$. Consider the 2×2 matrix A_k, which is the projection of A in the two-dimensional subspace* $\mathrm{Span}(\boldsymbol{g}_k, \boldsymbol{s}_{k-1})$*. Then we have that*

$$\mathrm{Trace}(A_k) = \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - 2\frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{g}_k\|_2^2} + \|\boldsymbol{s}_{k-1}\|^2\frac{\boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k}{\|\boldsymbol{g}_k\|^2}}{\|\boldsymbol{s}_{k-1}\|^2 - \frac{(\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k)^2}{\|\boldsymbol{g}_k\|^2}}, \tag{5.1}$$

where, again, $\boldsymbol{y}_{k-1} = A\boldsymbol{s}_{k-1}$.

Proof. Consider the following orthonormal base of the subspace $\mathrm{Span}(\boldsymbol{g}_k,\ \boldsymbol{s}_{k-1})$,

$$\bar{\boldsymbol{s}}_{k-1} = \frac{\boldsymbol{s}_{k-1}}{\|\boldsymbol{s}_{k-1}\|}, \quad \bar{\boldsymbol{g}}_k = \frac{\tilde{\boldsymbol{g}}_k}{\|\tilde{\boldsymbol{g}}_k\|}, \tag{5.2}$$

where $\tilde{\boldsymbol{g}}_k = \boldsymbol{g}_k - \frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\boldsymbol{s}_{k-1}$. Then $\mathrm{Trace}(A_k) = \bar{\boldsymbol{s}}_{k-1}^{\mathrm{T}}A\bar{\boldsymbol{s}}_{k-1} + \bar{\boldsymbol{g}}_k^{\mathrm{T}}A\bar{\boldsymbol{g}}_k$. The statement follows from direct calculations and the relation $\boldsymbol{y}_{k-1} = A\boldsymbol{s}_{k-1}$. □

As $\boldsymbol{x}_k \in \boldsymbol{x}_{k-1} + \mathrm{Span}(\boldsymbol{g}_{k-1},\ \boldsymbol{s}_{k-2})$, we suppose that

$$\boldsymbol{x}_k = \boldsymbol{x}_{k-1} + \alpha_{k-1}\boldsymbol{g}_{k-1} + \beta_{k-1}\boldsymbol{s}_{k-2}.$$

It follows that

$$\boldsymbol{y}_{k-1} = \alpha_{k-1}A\boldsymbol{g}_{k-1} + \beta_{k-1}\boldsymbol{y}_{k-2},$$

and hence

$$\boldsymbol{g}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} = \alpha_{k-1}\boldsymbol{g}_{k-1}^{\mathrm{T}}A\boldsymbol{g}_{k-1} + \beta_{k-1}\boldsymbol{g}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-2}.$$

Denote $\bar{\rho}_{k-1} = \boldsymbol{g}_{k-1}^{\mathrm{T}}A\boldsymbol{g}_{k-1}$. Then we have that

$$\bar{\rho}_{k-1} = \frac{\boldsymbol{g}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - \beta_{k-1}\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\alpha_{k-1}}. \tag{5.3}$$

In this case, we can obtain the value of $\mathrm{Trace}(A_{k-1})$ exactly:

$$\mathrm{Trace}(A_{k-1}) = \frac{\boldsymbol{s}_{k-2}^{\mathrm{T}}\boldsymbol{y}_{k-2} - 2\frac{\boldsymbol{s}_{k-2}^{\mathrm{T}}\boldsymbol{g}_{k-1}\,\boldsymbol{g}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-2}}{\|\boldsymbol{g}_{k-1}\|_2^2} + \|\boldsymbol{s}_{k-2}\|^2\frac{\bar{\rho}_{k-1}}{\|\boldsymbol{g}_{k-1}\|^2}}{\|\boldsymbol{s}_{k-2}\|^2 - \frac{(\boldsymbol{s}_{k-2}^{\mathrm{T}}\boldsymbol{g}_{k-1})^2}{\|\boldsymbol{g}_{k-1}\|^2}}. \tag{5.4}$$

We may choose ρ_k such that $\mathrm{Trace}(A_{k-1}) = \mathrm{Trace}(A_k)$. More exactly, we choose ρ_k such that

$$\frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - 2\frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{g}_k\|_2^2} + \|\boldsymbol{s}_{k-1}\|^2\frac{\rho_k}{\|\boldsymbol{g}_k\|^2}}{\|\boldsymbol{s}_{k-1}\|^2 - \frac{(\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{g}_k)^2}{\|\boldsymbol{g}_k\|^2}} = \mathrm{Trace}(A_{k-1}). \tag{5.5}$$

One advantage of the above choice of ρ_k is that, if the objective function is exactly a two-dimensional quadratic function, then ρ_k is exactly equal to the value of $\boldsymbol{g}_k^{\mathrm{T}}A\boldsymbol{g}_k$ since the value of $\mathrm{Trace}(A_k)$ is invariant in this case. Hence an exact line search is implied at the second iteration and the algorithm will terminate in four iterations by Theorem 3.1.

5.2 A choice of ρ_k with a certain optimal property

The philosophy in deriving the formula (2.13) seems plausible. On one hand, it is assumed that the scaled memoryless BFGS update is reliable and can be used to estimate the parameter ρ_k. On the other hand, the direction $-B_k^{-1}\boldsymbol{g}_k$ is not used as its search direction. To have a better philosophy, we shall consider the following family of scaled memoryless BFGS update:

$$B_k = \tau_k\left(I - \frac{\boldsymbol{s}_{k-1}\boldsymbol{s}_{k-1}^{\mathrm{T}}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{s}_{k-1}}\right) + \frac{\boldsymbol{y}_{k-1}\boldsymbol{y}_{k-1}^{\mathrm{T}}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, \tag{5.6}$$

where τ_k is some scaled parameter and is suggested [18, 19] to the interval

$$\tau_k \in \left[\frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}},\ 2\frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\right]. \tag{5.7}$$

We shall consider two methods based on (5.6) and ask them to become as close as possible.

The first method is the SMCG method based on (5.6). In this case, we estimate ρ_k by the value of $\boldsymbol{g}_k^{\mathrm{T}}B_k\boldsymbol{g}_k$, namely,

$$\rho_k = \tau_k\left(\|\boldsymbol{g}_k\|^2 - \frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2}{\|\boldsymbol{s}_{k-1}\|^2}\right) + \frac{(\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1})^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}. \tag{5.8}$$

Then we know from (2.11) that the corresponding SMCG direction is parallel to the direction $\boldsymbol{d}_k^{(1)}(\tau_k) = -\boldsymbol{g}_k + \beta_k^{(1)}(\tau_k)\boldsymbol{s}_{k-1}$, where

$$\beta_k^{(1)}(\tau_k) = \frac{\rho_k\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} - \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\|\boldsymbol{g}_k\|^2}{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1} - \boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\|\boldsymbol{g}_k\|^2} = \frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \xi_k\tau_k\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, \tag{5.9}$$

and where

$$\xi_k = \frac{[\|\boldsymbol{g}_k\|^2\|\boldsymbol{s}_{k-1}\|^2 - (\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})^2]\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{(\|\boldsymbol{g}_k\|^2\,\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1} - \boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}\,\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1})\,\|\boldsymbol{s}_{k-1}\|^2}. \tag{5.10}$$

The second method is the conjugate gradient method with an optimal property based on (5.6) (see [6] for details). It starts from the search direction $-B_k^{-1}\boldsymbol{g}_k$, where B_k is given in (5.6).

After some calculations, we know that such a direction is parallel to

$$\boldsymbol{d}_k^{PS} = -\boldsymbol{g}_k + \left[\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \left(\tau_k + \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}\right)\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}\right]\boldsymbol{s}_{k-1} + \frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}\,\boldsymbol{y}_{k-1}. \tag{5.11}$$

Now, denoting the one-dimensional manifold $\mathcal{S}_k = \{-\boldsymbol{g}_k + \beta\boldsymbol{s}_{k-1} : \beta \in \mathcal{R}\}$, we seek the vector in $\mathcal{S}_k$ closest to $\boldsymbol{d}_k^{PS}$ in (5.11) as the next search direction, namely,

$$\boldsymbol{d}_k^{(2)}(\tau_k) = \arg\min\{\|\boldsymbol{d} - \boldsymbol{d}_k^{PS}\| :\ \boldsymbol{d} \in \mathcal{S}_k\}. \tag{5.12}$$

Consequently, we obtain the search direction $\boldsymbol{d}_k^{(2)}(\tau_k) = -\boldsymbol{g}_k + \beta_k^{(2)}(\tau_k)\boldsymbol{s}_{k-1}$, where

$$\beta_k^{(2)}(\tau_k) = \frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \left(\tau_k + \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\right)\frac{\boldsymbol{g}_k^{\mathrm{T}}\boldsymbol{s}_{k-1}}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}. \tag{5.13}$$

Now we look for the optimal τ_k that minimizes $\|\boldsymbol{d}_k^{(1)}(\tau_k) - \boldsymbol{d}_k^{PS}\|$ subject to the constraint (5.7). It is easy to see that this minimization problem is equivalent to minimizing $\|\boldsymbol{d}_k^{(1)}(\tau_k) - \boldsymbol{d}_k^{(2)}(\tau_k)\|$ subject to the same constraint (5.7). By (5.9) and (5.13), we know that the optimal τ_k is the solution of

$$\begin{aligned} \min\quad & \left\|\xi_k\tau_k - \left(\tau_k + \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\right)\right\| \\ \text{s.t.}\quad & \tau_k \in \left[\frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, 2\frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\right]. \end{aligned} \tag{5.14}$$

If $\xi_k = 1$, any value of τ_k is allowed, in which case we can just take $\tau_k = \frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}$ so that the scaled memoryless BFGS method is used. Otherwise, the solution of (5.14) is to truncate $\bar{\tau}_k = (\frac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \frac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}) / (\xi_k - 1)$ into the interval in (5.7). To sum up, we take

$$\tau_k = \begin{cases} \dfrac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, & \text{if } \xi_k = 1, \\ \max\left\{\dfrac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}, \min\left\{\bar{\tau}_k, 2\dfrac{\|\boldsymbol{y}_{k-1}\|^2}{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}} - \dfrac{\boldsymbol{s}_{k-1}^{\mathrm{T}}\boldsymbol{y}_{k-1}}{\|\boldsymbol{s}_{k-1}\|^2}\right\}\right\}, & \text{otherwise.} \end{cases} \tag{5.15}$$

The value of parameter ρ_k is then defined by the relation (5.8).

6 Conclusions and discussions

In this paper, we have provided some new analysis of the SMCG method by Yuan and Stoer [25]. Special attentions have been given on how to choose the parameter ρ_k in the method. Specifically, by combining the Barzilai-Borwein idea, we were able to provide a very efficient way, i.e., (4.7), to choose ρ_k. Our preliminary numerical results show that the corresponding BBCG3 method is specially efficient among some others without any line searches. This implies that the BBCG3 method might enjoy the asymptotical one stepsize per line-search property and become a strong candidate for large-scale nonlinear optimization.

However, there are still many questions under investigation:

(1) Does the BBCG3 method without any line searches converge for convex quadratic minimization? If it converges, what is its convergence rate?

(2) How to extend the BBCG3 method for unconstrained optimization? In such a situation, it is important how to design a suitable line search.

(3) Are there some other more efficient choices of ρ_k in the SMCG method?

(4) How to extend the idea of this paper to other subspace minimization conjugate gradient methods? For example, the three-dimensional subspace minimization conjugate gradient method, which considers the subspace spanned by $-\boldsymbol{g}_k$, $\boldsymbol{s}_{k-1}$ and $\boldsymbol{s}_{k-2}$?

(5) How to extend the BBCG3 method for constrained optimization?

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 81173633, 11401038 and 11331012), the Chinese Academy of Sciences Grant (Grant No. kjcx-yw-s7-03), National Natural Science Foundation of China for Distinguished Young Scientists (Grant No. 11125107), the Key Project of Chinese National Programs for Fundamental Research and Development (Grant No. 2015CB856000) and the Fundamental Research Funds for the Central Universities (Grant No. 2014RC0904). The authors are grateful to Professor Ya-xiang Yuan for his useful suggestions and comments, especially suggesting us the choice of ρ_k in Subsection 5.1. They also thank the two anonymous referees, whose suggestions improve the quality of this paper.

References

1 Barzilai J, Borwein J M. Two-point step size gradient methods. IMA J Numer Anal, 1988, 8: 141–148
2 Birgin E G, Martínez J M. A spectral conjugate gradient method for unconstrained optimization. Appl Math Optim, 2001, 43: 117–128
3 Dai Y H. Nonlinear conjugate gradient methods. Wiley Encyclopedia Oper Res Manag Sci, 2011, doi: 10.1002/9780470400531.eorms0183
4 Dai Y H. A new analysis on the Barzilai-Borwein gradient method. J Oper Res Soc China, 2013, 1: 187–198
5 Dai Y H, Fletcher R. On the asymptotic behaviour of some new gradient methods. Math Program Ser A, 2005, 13: 541–559
6 Dai Y H, Kou C X. A nonlinear conjugate gradient algorithm with an optimal property and an improved Wolfe line search. SIAM J Optim, 2013, 23: 296–320
7 Dai Y H, Liao L Z. New conjugacy conditions and related nonlinear conjugate gradient methods. Appl Math Optim, 2001, 43: 87–101

8 Dai Y H, Liao L Z. *R*-linear convergence of the Barzilai and Borwein gradient method. IMA J Numer Anal, 2002, 22: 1–10

9 Dai Y H, Yuan Y. A nonlinear conjugate gradient method with a strong global convergence property. SIAM J Optim, 1999, 10: 177–182

10 Fletcher R, Reeves C M. Function minimization by conjugate gradients. Comput J, 1964, 7: 149–154

11 Grippo L, Lamparillo F, Lucidi S. A nonmonotone line search technique for Newton's method. SIAM J Numer Anal, 1986, 23: 707–716

12 Hager W W, Ngo C, Yashtini M, et al. An alternating direction approximate newton algorithm for ill-conditioned inverse problems with application to parallel MRI. J Oper Res Soc China, 2015, 3: 139–162

13 Hager W W, Zhang H. A new conjugate gradient method with guaranteed descent and an efficient line search. SIAM J Optim, 2005, 16: 170–192

14 Hager W W, Zhang H. A survey of nonlinear conjugate gradient methods. Pac J Optim, 2006, 2: 35–58

15 Hestenes M R, Stiefel E. Method of conjugate gradient for solving linear system. J Res Nat Bur Stand, 1952, 49: 409–436

16 Liu D C, Nocedal J. On the limited memory BFGS method for large scale optimization. Math Program, 1989, 45: 503–528

17 Nocedal J. Updating quasi-Newton matrices with limited storage. Math Comp, 1980, 35: 773–782

18 Oren S S. Self scaling variable metric (SSVM) algorithms, part II: Implementation and experiments. Manag Sci, 1974, 20: 863–874

19 Oren S S, Luenberger D G. Self scaling variable metric (SSVM) algorithms, part I: Criteria and sufficient conditions for scaling a class of algorithms. Manag Sci, 1974, 20: 845–862

20 Polak E, Ribière G. Note sur la convergence de méthods de directions conjugées. Rev Franaise Informat Recherche Opérationnelle, 1969, 16: 35–43

21 Polyak B T. The conjugate gradient method in extremem problems. USSR Comp Math Math Phys, 1969, 9: 94–112

22 Raydan M. On the Barzilai and Borwein choice of steplength for the gradient method. IMA J Numer Anal, 1993, 13: 321–326

23 Raydan M. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J Optim, 1997, 7: 26–33

24 Sun W, Yuan Y. Optimization Theory and Methods: Nonlinear Programming. New York: Springer, 2006

25 Yuan Y, Stoer J. A subspace study on conjugate gradient algorithms. Z Angew Math Mech, 1995, 75: 69–77

26 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