The new spectral conjugate gradient method for large-scale unconstrained optimisation

Li Wang¹, Mingyuan Cao¹, Funa Xing¹ and Yueting Yang¹*

Abstract
The spectral conjugate gradient methods are very interesting and have been proved to be effective for strictly convex quadratic minimisation. In this paper, a new spectral conjugate gradient method is proposed to solve large-scale unconstrained optimisation problems. Motivated by the advantages of approximate optimal stepsize strategy used in the gradient method, we design a new scheme for the choices of the spectral and conjugate parameters. Furthermore, the new search direction satisfies the spectral property and sufficient descent condition. Under some suitable assumptions, the global convergence of the developed method is established. Numerical comparisons show better behaviour of the proposed method with respect to some existing methods for a set of 130 test problems.

Keywords: Approximate optimal stepsize; Spectral conjugate gradient method; Global convergence

1 Introduction
Consider the following unconstrained optimisation:

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

where $f: \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and bounded from below. Conjugate gradient method is one of the most effective line search methods for solving unconstrained optimisation problem (1) due to its features of low memory requirement and simple computation. Let $x_0$ be an arbitrary initial approximate solution of problem (1). The iterative formula of conjugate gradient is given by

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

The search direction $d_k$ is defined by

$$d_k = \begin{cases} 
-g_0, & \text{if } k = 0, \\
-g_k + \beta_k d_{k-1}, & \text{if } k \geq 1, 
\end{cases} \tag{3}$$

© The Author(s) 2020. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.
where \( g_k = \nabla f(x_k) \) is the gradient of \( f(x) \) at \( x_k \) and \( \beta_k \) is a conjugate parameter. Different choices of \( \beta_k \) correspond to different conjugate gradient methods. Well-known formulas for \( \beta_k \) can be found in \([8, 12–14, 17, 26]\). The stepsize \( \alpha_k > 0 \) is usually obtained by the Wolfe line search

\[
\begin{align*}
    f(x_k + \alpha_k d_k) &\leq f(x_k) + c_1 \alpha_k g_k^T d_k, \\
g_k^T d_k &\geq c_2 g_k^T d_k,
\end{align*}
\]

where \( 0 < c_1 \leq c_2 < 1 \). In order to exclude the points that are far from stationary points of \( f(x) \) along the direction \( d_k \), the strong Wolfe line search is used, which requires \( \alpha_k \) to satisfy (4) and

\[
|g_{k+1}^T d_k| \leq c_2 |g_k^T d_k|.
\]

Combining the conjugate gradient method and spectral gradient method \([3]\), a spectral conjugate gradient method (SCG) was proposed by Bergin et al. \([5]\). Let \( s_{k-1} = x_k - x_{k-1} = \alpha_{k-1} d_{k-1} \) and \( y_{k-1} = g_k - g_{k-1} \). The direction \( d_k \) is termed as

\[
d_k = -\theta_k g_k + \beta_k s_{k-1},
\]

where the spectral parameter \( \theta_k \) and the conjugate parameter \( \beta_k \) are defined by

\[
\theta_k = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}, \quad \beta_k = \frac{(\theta_k y_{k-1} - s_{k-1})^T g_k}{d_{k-1}^T y_{k-1}},
\]

respectively. Obviously, if \( \theta_k = 1 \), the method is one of the classical conjugate gradient methods; if \( \beta_k = 0 \), the method is the spectral gradient method.

The SCG \([5]\) was modified by Yu et al. \([32]\) in order to achieve the descent directions. Moreover, there are other ways to determine \( \theta_k \) and \( \beta_k \). For instance, based on the descent condition, Wan et al. \([29]\) and Zhang et al. \([35]\) presented the modified PRP and FR spectral conjugate gradient method, respectively. Due to the strong convergence of the Newton method, Andrei \([1]\) proposed an accelerated conjugate gradient method, which took advantage of the Newton method to improve the performance of the conjugate gradient method. Following this idea, Parvaneh et al. \([24]\) proposed a new SCG, which is a modified version of the method suggested by Jian et al. \([15]\). Masoud \([21]\) introduced a scaled conjugate gradient method which inherited the good properties of the classical conjugate gradient. More references in this field can be seen in \([6, 10, 20, 28, 34]\).

Recently, Liu et al. \([18, 19]\) introduced approximate optimal stepsizes \( (\alpha_k^{AOS}) \) for gradient method. They constructed a quadratic approximation model of \( f(x_k - \alpha g_k) \)

\[
\varphi(\alpha) = f(x_k - \alpha g_k) = f(x_k) - \alpha \|g_k\|^2 + \frac{1}{2} \alpha^2 g_k^T B_k g_k,
\]

where the approximation Hessian matrix \( B_k \) is symmetric and positive definite. By minimising \( \varphi(\alpha) \), they obtained \( \alpha_k^{AOS} = \frac{\|g_k\|^2}{g_k^T B_k g_k} \) and proposed the approximate optimal gradient methods. If \( B_k = \frac{1}{\|g_{k-1}\|^2} I \) is selected, then the \( \alpha_k^{AOS} \) reduces to \( \alpha_k^{BB1} \), and the corresponding
method is BB method [3]. If \( B_k = 1/\bar{\alpha}_{BB}^k I \) is chosen, where \( \bar{\alpha}_{BB}^k \) is some modified BB step-size, then the \( \alpha_{AOS}^k \) reduces to \( \bar{\alpha}_{BB}^k \), and the corresponding method is some modified BB method [4, 7, 30]. And if \( B_k = 1/t I, \ t > 0 \), then the \( \alpha_{AOS}^k \) is the fixed stepsize \( t \), and the corresponding method is the gradient method with fixed stepsize [16, 22, 33]. In this sense, the approximate optimal gradient method is a generalisation of the BB methods.

In this paper, we propose a new spectral conjugate gradient method based on the idea of the approximate optimal stepsize. Compared with the SCG method [5], the proposed method generates the sufficient descent direction per iteration and does not require more computation costs. Under some assumption conditions, the global convergence of the proposed method is established.

The rest of this paper is organised as follows. In Sect. 2, a new spectral conjugate gradient algorithm is presented and its computational costs are analysed. The global convergence of the proposed method is established in Sect. 3. In Sect. 4, some numerical experiments are used to show that the proposed method is superior to the SCG [5] and DY [8] methods. Conclusions are drawn in Sect. 5.

2 The new spectral conjugate gradient algorithm

In this section, we propose a new spectral conjugate gradient method with the form of (7). Let \( \bar{d}_k \) be a classical conjugate gradient direction. We firstly consider the approximate model of \( f(x_k + \alpha \bar{d}_k) \)

\[
\psi(\alpha) \equiv f(x_k + \alpha \bar{d}_k) = f(x_k) + \alpha \bar{g}_k^T \bar{d}_k + \frac{1}{2} \alpha^2 \bar{d}_k^T B_k \bar{d}_k.
\] (8)

By \( \frac{d\psi}{d\alpha} = 0 \), we obtain the approximate optimal stepsize \( \alpha^*_k \) associated with \( \psi(\alpha) \)

\[
\alpha^*_k = -\frac{\bar{g}_k^T \bar{d}_k}{\bar{d}_k^T B_k \bar{d}_k}.
\] (9)

Here, we choose BFGS update formula to generate \( B_k \), that is,

\[
B_k = B_{k-1} - \frac{B_{k-1} s_{k-1} s_{k-1}^T B_{k-1}}{s_{k-1}^T B_{k-1} s_{k-1}} + \frac{y_{k-1} y_{k-1}^T}{s_{k-1}^T y_{k-1}}.
\] (10)

To reduce the computational and storage costs, the memoryless BFGS schemes are usually used to substitute \( B_k \), see [2, 23, 25]. In this paper, we choose \( B_{k-1} \) as a scalar matrix \( \xi \frac{\|y_{k-1}\|^2}{s_{k-1}^T y_{k-1}} I, \ \xi > 0 \). Then (10) can be rewritten as

\[
B_k = \xi \frac{\|y_{k-1}\|^2}{s_{k-1}^T y_{k-1}} I - \xi \frac{\|y_{k-1}\|^2}{s_{k-1}^T y_{k-1}} s_{k-1} s_{k-1}^T + \frac{y_{k-1} y_{k-1}^T}{s_{k-1}^T y_{k-1}}.
\] (11)

It is easy to prove that if \( s_{k-1}^T y_{k-1} > 0 \), then \( B_k \) is symmetric and positive definite. If the direction \( \tilde{d}_k \) is chosen as DY formula [8], i.e.,

\[
\tilde{d}_k = d_k^{DY} = -g_k + \beta_k^{DY} s_{k-1}, \quad \beta_k^{DY} = \frac{\|g_k\|^2}{s_{k-1}^T y_{k-1}}.
\] (12)
Substituting (11) and (12) into (9), we have

\[ \alpha_k^* = \frac{-s_k^T g_k}{\xi \| y_k-1 \|^2 p_k} \]  

(13)

where

\[ p_k = 1 - \frac{(g_k^T s_k-1)^2}{\| g_k \|^2 \| s_k-1 \|^2} \left( \frac{g_k^T y_k-1}{\| g_k \| \| y_k-1 \|} + \frac{\| g_k \|}{\| y_k-1 \|} \right)^2. \]  

(14)

To ensure the sufficient descent property of the direction and the bounded property of spectral parameter \( \theta_k \), the truncating technique in [19] is adopted to choose \( \theta_k \) and \( \beta_k \) as follows:

\[
\begin{align*}
\theta_k &= \max\{\min\{\alpha_k^*, \bar{\rho}_k\}, \rho_k\}, \\
\beta_k &= \theta_k \beta_{DY}^k,
\end{align*}
\]  

(15)

where \( \bar{\rho}_k = \frac{y_k-1^T y_k-1}{y_k-1^T y_k-1} \) and \( \rho_k = \frac{y_k-1^T y_k-1}{y_k-1^T y_k-1} \).

Based on the above analyses, we describe the following algorithm.

**Algorithm 2.1 (NSCG)**

Step 0. Let \( x_0 \in \mathbb{R}^n, \varepsilon > 0, 0 < c_1 \leq c_2 < 1 \) and \( 1 \leq \xi \leq 2 \). Compute \( f_0 = f(x_0) \) and \( g_0 = \nabla f(x_0) \). Set \( d_0 := -g_0 \) and \( k := 0 \).

Step 1. If \( \| g_k \| \leq \varepsilon \), stop.

Step 2. Compute \( \alpha_k \) by (4) and (6).

Step 3. Set \( x_{k+1} = x_k + \alpha_k d_k \), and compute \( g_{k+1} \).

Step 4. Compute \( \theta_{k+1} \) and \( \beta_{k+1} \) by (15).

Step 5. Compute \( d_{k+1} \) by (7), set \( k := k + 1 \). Return to Step 1.

**Remark 1** By contrast with the SCG algorithm formula, the extra computational work of NSCG algorithm seems to require the inner products \( g_k^T s_k-1 \) per iteration. But \( g_k^T s_k-1 \) should be computed while implementing the Wolfe conditions. It implies that the extra computational work can be negligible.

**Remark 2** It is well known that \( s_k-1^T y_k-1 > 0 \) can be guaranteed by the Wolfe line search. Since (11) implies a memoryless quasi-Newton update, from the references [27] and [31], it can be seen

\[ m \leq \rho_k \leq \bar{\rho}_k \leq M, \]

where \( m \) and \( M \) are positive constants. Together with (15), the parameter \( \theta_k \) satisfies that

\[ m \leq \theta_k \leq M. \]  

(16)

The following theorem indicates that the search direction generated by NSCG algorithm satisfies the sufficient descent condition.
Theorem 2.1 *The search direction $d_k$ generated by NSCG algorithm is a sufficient descent direction, i.e.,*

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

where $c = m/(1 + c_2) > 0$. \hspace{1cm} (17)

*Proof* From (6), we have

$$l_k = \frac{g_k^T s_{k-1}}{g_k^T s_{k-1}} \in [-c_2, c_2].$$

Pre-multiplying (7) by $g_k^T$, from (15), (16) and (18), we have

$$g_k^T d_k = -\theta_k\|g_k\|^2 + \beta_k g_k^T s_{k-1}$$

$$= \theta_k\|g_k\|^2 \frac{1}{l_k - 1}$$

$$\leq - \frac{m}{1 + c_2}\|g_k\|^2$$

$$= -c\|g_k\|^2,$$

where $c = m/(1 + c_2) > 0$. \hspace{1cm} □

3 Convergence analysis

In this section, the convergence of NSCG algorithm is analysed. We consider that $\|g_k\| \neq 0$ for all $k \geq 0$, otherwise a stationary point is obtained. We make the following assumptions.

Assumption 3.1

(i) The level set $\Omega = \{x|f(x) \leq f(x_0)\}$ is bounded.

(ii) In some open neighbourhood $N$ of $\Omega$, the function $f$ is continuously differentiable and its gradient is Lipschitz continuous, i.e., there exists a constant $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\| \text{ for any } x, y \in N.$$ \hspace{1cm} (19)

Assumption 3.1 implies that there exists a constant $\Gamma \geq 0$ such that

$$\|g(x)\| \leq \Gamma \text{ for any } x \in \Omega.$$ \hspace{1cm} (20)

The following lemma called Zoutendijk condition [36] was originally given by Zoutendijk et al.

**Lemma 3.1** Suppose that Assumption 3.1 holds. Let the sequences $\{d_k\}$ and $\{\alpha_k\}$ be generated by NSCG algorithm. Then

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty.$$ \hspace{1cm} (21)

From Assumption 3.1, Theorem 2.1 and Lemma 3.1, the following result can be proved.
Lemma 3.2 Suppose that Assumption 3.1 holds. Let the sequences \( \{d_k\} \) and \( \{\alpha_k\} \) be generated by NSCG algorithm. Then either
\[
\liminf_{k \to \infty} \|g_k\| = 0 \quad (22)
\]
or
\[
\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \quad (23)
\]

Proof. It is sufficient to prove that if (22) is not true, then (23) holds. We use proofs by contradiction. Suppose that there exists \( \gamma > 0 \) such that
\[
\|g_k\| \geq \gamma \quad \text{for any } k \geq 0. \quad (24)
\]

From (7) and Theorem 2.1, we have
\[
\frac{\|d_k\|^2}{\|d_{k-1}\|^2} = \frac{(\alpha_{k-1}\beta_k)^2\|d_{k-1}\|^2 - \theta_k^2\|g_k\|^2 - 2\theta_k d_k^T g_k}{\|d_{k-1}\|^2} \geq (\alpha_{k-1}\beta_k)^2 - \theta_k^2 \frac{\|g_k\|^2}{\|d_{k-1}\|^2}. \quad (25)
\]

Besides, pre-multiplying (7) by \( g_k^T \), we have
\[
g_k^T d_k - \alpha_{k-1}\beta_k g_k^T d_{k-1} = -\theta_k \|g_k\|^2.
\]

By using the triangle inequality and (6), we get
\[
|g_k^T d_k| + c_2 |\alpha_{k-1}| |\beta_k| |g_{k-1}^T d_{k-1}| \geq \theta_k \|g_k\|^2. \quad (26)
\]

Together with Cauchy’s inequality, (26) yields
\[
(g_k^T d_k)^2 + (\alpha_{k-1}\beta_k)^2 (g_{k-1}^T d_{k-1})^2 \geq \frac{\theta_k^2}{1 + c_2^2} \|g_k\|^4. \quad (27)
\]

Therefore, from (25) and (27), we obtain
\[
\frac{(g_k^T d_k)^2}{\|d_k\|^2} + \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} \geq \frac{\theta_k^2}{1 + c_2^2} \frac{\|g_k\|^4}{\|d_k\|^2}.
\]

\[
= \frac{\|g_k\|^4}{\|d_k\|^2} \left[ \frac{1}{\|d_k\|^2} \left( \left( g_k^T d_k \right)^2 + \left( g_{k-1}^T d_{k-1} \right)^2 \right) \right] \geq \frac{\|g_k\|^4}{\|d_k\|^2} \left[ \frac{1}{1 + c_2^2} \|g_k\|^4 + \left( g_{k-1}^T d_{k-1} \right)^2 \left( \frac{\|d_k\|^2}{\|d_{k-1}\|^2} - (\alpha_{k-1}\beta_k)^2 \right) \right] \geq \frac{\|g_k\|^4}{\|d_k\|^2} \left[ \frac{\theta_k^2}{1 + c_2^2} \|g_k\|^4 + \left( g_{k-1}^T d_{k-1} \right)^2 \left( \frac{\|d_k\|^2}{\|d_{k-1}\|^2} - (\alpha_{k-1}\beta_k)^2 \right) \right] \geq \frac{\|g_k\|^4}{\|d_k\|^2} \frac{\theta_k^2}{1 + c_2^2} \left[ \frac{1}{\|d_{k-1}\|^2} \left( \frac{g_{k-1}^T d_{k-1}}{\|g_k\|^2} \right)^2 \frac{1}{\|d_k\|^2} \|g_k\|^2 \right] \geq \frac{\|g_k\|^4}{\|d_k\|^2} \frac{\theta_k^2}{1 + c_2^2} \left[ \frac{1}{\|d_{k-1}\|^2} \left( \frac{g_{k-1}^T d_{k-1}}{\|g_k\|^2} \right)^2 \frac{1}{\|d_k\|^2} \|g_k\|^2 \right].
\]
It follows from Lemma 3.1 that
\[
\lim_{k \to \infty} \frac{(\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\| \mathbf{d}_{k-1} \|^2} = 0.
\]

By use of (24) and \( \theta_k \geq m \), for all sufficiently large \( k \), there exists a positive constant \( \lambda \) such that
\[
\theta_k^2 \left[ \frac{1}{1 + c_2^2} - \frac{(\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\| \mathbf{d}_{k-1} \|^2} \frac{1}{\| \mathbf{g}_k \|^2} \right] \geq \lambda. \tag{29}
\]

Therefore, from (28) and (29) we have
\[
\frac{(\mathbf{g}_k^T \mathbf{d}_k)^2}{\| \mathbf{d}_k \|^2} + \frac{(\mathbf{g}_k^T \mathbf{d}_{k-1})^2}{\| \mathbf{d}_{k-1} \|^2} \geq \lambda \| \mathbf{g}_k \|^4 \| \mathbf{d}_k \|^2
\]
holds for all sufficiently large \( k \). Combining with the Zoutendijk condition, we deduce that inequality (23) holds. \( \square \)

**Corollary 3.1** Suppose that all the conditions of Lemma 3.2 hold. If
\[
\sum_{k=0}^\infty \frac{1}{\| \mathbf{d}_k \|^2} = +\infty, \tag{30}
\]
then
\[
\liminf_{k \to \infty} \| \mathbf{g}_k \| = 0.
\]

**Proof** Suppose that there is a positive constant \( \gamma \) such that \( \| \mathbf{g}_k \| \geq \gamma \) for all \( k \geq 0 \). From Lemma 3.2, we have
\[
\sum_{k=0}^\infty \frac{1}{\| \mathbf{d}_k \|^2} \leq \frac{1}{\gamma^4} \sum_{k=0}^\infty \| \mathbf{g}_k \|^4 \| \mathbf{d}_k \|^2 \leq \infty,
\]
which contradicts (30), i.e., Corollary 3.1 is true. \( \square \)

In the following, we establish the global convergence theorem of NSCG algorithm.

**Theorem 3.1** Suppose that Assumption 3.1 holds and the sequence \( \{x_k\} \) is generated by NSCG algorithm. If there exists a constant \( \gamma > 0 \) such that \( \| \mathbf{g}_k \| \geq \gamma \), then the algorithm satisfies
\[
\liminf_{k \to \infty} \| \mathbf{g}_k \| = 0. \tag{31}
\]

**Proof** From Theorem 2.1, we have
\[
\mathbf{g}_k^T \mathbf{s}_{k-1} \leq -c \| \mathbf{g}_{k-1} \| \| \mathbf{s}_{k-1} \|.
\]
Observe that \( y_k^T s_k^{-1} = g_k^T s_k^{-1} - g_k^T s_{k-1} \geq (c_2 - 1) g_k^T s_{k-1} \), we have

\[
\begin{align*}
y_k^T s_k^{-1} &\geq c(1 - c_2) \|g_k\| \|s_k^{-1}\|
\end{align*}
\]

Moreover, from (15), (17) and (20), we get

\[
\beta_k \leq M \frac{\|g_k\|^2}{y_k^T s_k^{-1}} \leq \frac{M}{c(1 - c_2)} \frac{\|g_k\|^2}{\|g_k^{-1}\| \|s_k^{-1}\|}
\]

\[
\leq \frac{M \Gamma^2}{cY(1 - c_2)} \frac{1}{\|s_k^{-1}\|} = \frac{\mu}{\|s_k^{-1}\|},
\]

where \( \mu = M \Gamma^2/cY(1 - c_2) \). Thus

\[
\|d_k\| \leq |\theta_k| \|g_k\| + |\beta_k| \|s_k^{-1}\| \leq M \Gamma + \mu.
\]

This implies that \( \sum_{k=0}^{\infty} 1/\|d_k\|^2 = \infty \). By Corollary 3.1, (31) holds. \( \square \)

4 Numerical results

In this section, we show the computational performance of NSCG algorithm. All codes are written in Matlab R2015b and run on PC with 2.50 GHz CPU processor and 4.00 GB RAM memory. Our test problems consist of 130 examples [9] from 100 to 5,000,000 variables.

We implement the same stopping criterion

\[
\|g_k\| \leq \varepsilon \quad \text{or} \quad |f(x_{k+1}) - f(x_k)| \leq \varepsilon \max \{1.0, |f(x_k)|\}.
\]

Set the parameters \( \varepsilon = 10^{-6} \), \( \xi = 1.0001 \), \( c_1 = 0.0001 \) and \( c_2 = 0.9 \).

Liu et al. [19] proposed GM_AOS 1, GM_AOS 2 and GM_AOS 3 algorithms, and GM_AOS 2 algorithm was slightly better than the other algorithms. When the quadratic model is considered, the algorithm developed by [18] is identical with GM_AOS 1 algorithm. In a certain sense, our algorithm can be viewed as an extension of SCG algorithm [5] and a modification of DY algorithm [8]. Therefore, we adopt the performance profiles introduced by Dolan et al. [11] to display the numerical performances of NSCG, SCG, DY and GM_AOS 2 algorithms.

It is noticed that the number of iterations (Itr), the number of function evaluations (NF), the number of gradient evaluations (NG) and the CPU time (Tcpu) are important factors showing the numerical performance of an optimal method. In profiles, the top curve is the method that solved the most problems in a time that was within a factor of the best time. The horizontal axis gives the percentage (\( \tau \)) of the test problems for which a method is the fastest (efficiency), while the vertical side gives the percentage (\( \psi \)) of the test problems that are successfully solved by each of the methods. Moreover, we present the number of problems solved by the tested algorithms with a minimum number of Itr, NF and NG and the minimum Tcpu. If programme runs failure, we denote the number of Itr, NF, NG by a large positive integer, respectively, and denote the Tcpu by 1000 seconds. In this way, only NSCG algorithm can solve all test problems. However, SCG, DY and GM_AOS 2 algorithms do 98.5%, 93.8% and 92.3% of problems, respectively.

From Figs. 1–4, we can see that NSCG algorithm is the top performer, being more successful and more robust than SCG, DY and GM_AOS 2 algorithms. For example, in Fig. 1,
subject to Itr, NSCG algorithm outperforms in 62 problems (i.e., it achieves the minimum number of iterations in 130 problems), SCG algorithm outperforms in 28 problems, DY algorithm outperforms in 23 problems, while GM_AOS 2 outperforms in 17 problems. Observe that NSCG algorithm is also the fastest of the three algorithms in Figs. 2, 3 and 4. To conclude, NSCG algorithm is more effective than other algorithms with respect to all the measures (Itr, NF, NG, Tcpu).

5 Conclusions
In this paper, a new spectral conjugate gradient method is proposed based on the idea of approximate optimal stepsize. Besides, the memoryless BFGS formula is embedded in our algorithm to reduce the computational and storage costs. Under some assumptions, global convergence of the proposed method is established. Numerical results show that this method is efficient and competitive.
Figure 3  Performance profiles for the number of gradient evaluations

Figure 4  Performance profiles for the CPU time

Acknowledgements
The authors are grateful to the editor and the anonymous reviewers for their valuable comments and suggestions, which have substantially improved this paper.

Funding
This work is supported by the Innovation Talent Training Program of Science and Technology of Jilin Province of China (20180519011JH), the Science and Technology Development Project Program of Jilin Province (20190303132SF), the Doctor Research Startup Project of Beihua University (170220014), the Project of Education Department of Jilin province (JKH20200028KJ) and the Graduate Innovation Project of Beihua University (2018014, 2019006).

Availability of data and materials
All data generated or analysed during this study are included in this manuscript.

Competing interests
The authors declare that there are no competing interests regarding the publication of this paper.

Authors' contributions
The authors conceived of the study, drafted the manuscript. All authors read and approved the final version of this paper.

Publisher’s Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.
References

1. Andrei, N.: New accelerated conjugate gradient algorithms as a modification of Dai–Yuan’s computational scheme for unconstrained optimization. J. Comput. Appl. Math. 234(12), 3397–3410 (2010)
2. Babaie-Kafaki, S.: On optimality of the parameters of self-scaling memoryless quasi-Newton updating formulae. J. Optim. Theory Appl. 167(1), 91–101 (2015)
3. Barzilai, J., Borwein, J.: Two-point step size gradient methods. IMA J. Numer. Anal. 8, 141–148 (1988)
4. Biglari, F., Solimanpur, M.: Scaling on the spectral gradient method. J. Optim. Theory Appl. 158(2), 626–635 (2013)
5. Birgin, E., Martínez, J.: A spectral conjugate gradient method for unconstrained optimization. Appl. Math. Optim. 43, 117–128 (2001)
6. Dai, Y., Kou, C., A Barzilai–Borwein conjugate gradient method. Sci. China Math. 59(8), 1511–1524 (2016)
7. Dai, Y., Yuan, J., Yuan, Y.: Modified two-point stepsize gradient methods for unconstrained optimization problems. Comput. Optim. Appl. 22, 103–109 (2002)
8. Dai, Y., Yuan, Y.: A nonlinear conjugate gradient method with a strong global convergence property. SIAM J. Optim. 10, 177–182 (2000)
9. Dai, Y., Yuan, Y.: An efficient hybrid conjugate gradient method for unconstrained optimization. Ann. Oper. Res. 70, 1155–1167 (2001)
10. Deng, S., Wan, Z.: An improved spectral conjugate gradient algorithm for nonconvex unconstrained optimization problems. J. Optim. Theory Appl. 157, 820–842 (2013)
11. Dolan, E., Moré, J.: Benchmarking optimization software with performance profiles. Math. Program. 91, 201–213 (2002)
12. Fletcher, R., Reeves, C.: Function minimization by conjugate gradients. Comput. J. 7(2), 149–154 (1964)
13. Fager, W., Zhang, H.: A new conjugate gradient method with guaranteed descent and an efficient line search. SIAM J. Optim. 16, 170–192 (2005)
14. Hestenes, M., Stiefel, E.: Methods of conjugate gradient for solving linear systems. J. Res. Natl. Bur. Stand. 49, 409–436 (1952)
15. Jian, J., Chen, Q., Jiang, X., Zeng, Y., Yin, J.: A new spectral conjugate gradient method for large-scale unconstrained optimization. Optim. Methods Softw. 32(5), 503–515 (2017)
16. Liu, J., Liu, H., Zheng, Y.: A new supermemory gradient method without line search for unconstrained optimization. In: The Sixth International Symposium on Neural Networks, vol. 56, pp. 641–647 (2009)
17. Liu, Y., Storey, C.: Efficient generalized conjugate gradient, part I: theory. J. Optim. Theory Appl. 7, 149–154 (1964)
18. Liu, Z., Liu, H.: An efficient gradient method with approximate optimal steplength for large scale unconstrained optimization. Numer. Algorithms 78(1), 21–39 (2017)
19. Liu, Z., Liu, H.: Several efficient gradient methods with approximate optimal steplength for large scale constrained optimization. J. Comput. Appl. Math. 328, 400–441 (2018)
20. Livieris, I., Pintelas, P.: A new class of spectral conjugate gradient methods based on a modified secant equation for unconstrained optimization. J. Comput. Appl. Math. 239, 396–405 (2013)
21. Masoud, F.: A scaled conjugate gradient method for nonlinear unconstrained optimization. Optim. Methods Softw. 32(5), 1095–1112 (2017)
22. Naramgi, Y.: A memory gradient method without line search for constrained optimization. SUT J. Math. 42, 191–206 (2008)
23. Nocedal, J.: Updating quasi-Newton matrices with limited storage. Math. Comput. 35(151), 773–782 (1980)
24. Parvaneh, F., Keyvan, A.: A modified spectral conjugate gradient method with global convergence. J. Optim. Theory Appl. 182, 667–690 (2019). https://doi.org/10.1007/s10957-019-01527-6
25. Perry, J.: A class of conjugate gradient algorithms with a two step variable metric memory. Discussion paper 269, Center for Mathematical Studies in Economics and Management Science, Northwestern University, Chicago (1977)
26. Polak, B.: The conjugate gradient method in extremal problems. USSR Comput. Math. Math. Phys. 9(4), 94–112 (1969)
27. Raydan, M., Svaiter, B.: Relaxed steepest descent and Cauchy–Barzilai–Borwein method. Comput. Optim. Appl. 21, 155–167 (2002)
28. Sun, M., Liu, J.: A new spectral conjugate gradient method and its global convergence. Int. J. Inf. Comput. Sci. 8(1), 75–80 (2013)
29. Wan, Z., Yang, Z., Wang, Y.: New spectral PRP conjugate gradient method for unconstrained optimization. Appl. Math. Lett. 24(1), 16–22 (2011)
30. Xiao, Y., Wang, Q., Wang, D.: Notes on the Dai–Yuan–Yuan modified spectral gradient method. J. Comput. Appl. Math. 234(10), 2986–2992 (2010)
31. Yang, Y., Xu, C.: A compact limited memory method for large scale unconstrained optimization. Eur. J. Oper. Res. 180, 48–56 (2007)
32. Yu, G., Guan, L., Chen, W.: Spectral conjugate gradient methods with sufficient descent property for large-scale unconstrained optimization. Optim. Methods Softw. 23(2), 275–293 (2008)
33. Yu, Z.: Global convergence of a memory gradient method without line search. J. Appl. Math. Comput. 26, 545–553 (2008)
34. Zhang, L., Zhou, W.: Spectral gradient projection method for solving nonlinear monotone equations. J. Comput. Appl. Math. 196(2), 478–484 (2006)
35. Zhang, L., Zhou, W., Li, D.: Global convergence of a modified Fletcher–Reeves conjugate gradient method with Armijo-type line search. Numer. Math. 104(4), 561–572 (2006)
36. Zoutendijk, G.: Nonlinear programming, computational method. In: Abadie, J. (ed) Integer and Nonlinear Programming. North-Holland, Amsterdam, pp. 37–86 (1970)