RESEARCH ARTICLE
doi: 10.2306/scienceasia1513-1874.2021.064

ScienceAsia 47 (2021): **514–519**

A spectral conjugate gradient method for convex constrained monotone equations

Shi Lei

General Education and International College, Chongqing College of Electronic Engineering, Chongqing 5401331 China

e-mail: shileimath@163.com

Received 9 Jan 2021
Accepted 5 May 2021

ABSTRACT: Based on the projection technique, in this paper we establish a spectral conjugate gradient method to solve nonlinear monotone equations with convex constraints. A nice property is that its search direction always satisfies the sufficient descent condition in each iteration, which is independent of any line search. Because there is not any derivative information, the proposed method is very suitable to solve large-scale nonsmooth monotone equations. By using a derivative-free line search, the global convergence is proved under the Lipschitz continuity. Preliminary numerical experiments show that the proposed method is effective and promising.

KEYWORDS: nonlinear monotone equations, spectral conjugate gradient method, derivative-free projection method, global convergence

MSC2010: 90C30 65K05

INTRODUCTION

Consider the following nonlinear monotone equations:

$$F(x)=0, \qquad x\in\Omega, \tag{1}$$

where $F:\Omega\to\mathbb{R}^n$ is a continuous mapping, and $\Omega\subseteq\mathbb{R}^n$ is a nonempty closed convex set. The monotonicity of F means

$$(F(x)-F(y))^{\mathrm{T}}(x-y)\geqslant 0, \quad \forall\, x,y\in\mathbb{R}^n. \tag{2}$$

Problem (1) originates from some applications, such as the chemical equilibrium systems, the economic equilibrium problems, the power flow equations and so on.

In recent years, two types of methods are widely studied for solving problem (1). The first type is a derivative method, e.g., Newton method, quasi-Newton method, Gauss-Newton method and their various variants, see [1–8]. These methods are famous for their rapid convergence rate from a good initial guess. However, they need the use of the Jacobian matrix of F or an approximation of the Jacobian matrix F in each iteration, which are causes of failure to solve large-scale nonsmooth monotone equations.

The other type is a derivative-free method based on the structures of some one-order gradient methods, e.g., conjugate gradient method, spectral gradient method, and spectral conjugate gradient method. It is well known that these one-order gradient methods are famous for their simplicity, low storage requirements, efficient computation, and nice convergence, which attract many researchers to extend them for solving large-scale nonsmooth monotone equations based on the hyperplane projection technique [4]. Zhang and Zhou [9] proposed a derivative-free projection approach based on the spectral gradient method for solving large-scale nonlinear unconstrained equations, where $\Omega=\mathbb{R}^n$. Yu et al [10] further studied the the algorithm [9] for solving nonlinear convex constrained equations. Li and Li [11] proposed two derivative-free projection methods based on the famous Polak-Ribire-Polyak conjugate gradient method [12] for solving large-scale unconstrained equations. Liu and Li [13] studied a three-term derivative-free projection method for unconstrained equations. Xiao and Zhu [14] successfully extended the famous CG_DESCENT conjugate gradient method [15] to solve nonlinear convex constrained monotone equations and reconstructed the sparse signal in compressive sensing. Liu and Li [16] further studied the algorithm [14] and proposed a modified derivative-free projection method. Recently, Cao [17] also established a three-term derivative-free projection method for solving convex constrained monotone equations based on the structures of the famous Dai-Yuan conjugate gradient method and the three-term conjugate gradient method. Inspired by the above research, we further study a derivative-free projection method for large-scale monotone equations with convex constraints.

In this paper, we propose a spectral conjugate gradient method and extend it to solve nonlinear problem (1). The method has two beneficial properties: First, its search direction satisfies the sufficient descent condition, which is independent of any line search. Second, it does not need any gradient information or matrix storage, so it is suitable to solve large-scale non-smooth nonlinear monotone equations.

Throughout this paper, $\|\cdot\|$ denotes the Euclidean norm of a vector.

ALGORITHM

In this section, we first consider the following unconstrained optimization problem

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

where $f:\mathbb{R}^n\to\mathbb{R}$ is a continuously nonlinear differentiable function, and g_k is the gradient of f at point x_k. The conjugate gradient method proposed by Rivaie et al [18] for solving the unconstrained problem is to generate a sequence $\{x_k\}$ which satisfies the following relation:

$$x_{k+1}=x_k+\alpha_k d_k, \qquad k\geqslant 0,$$

where α_k is the step-size, and d_k is the search direction generated by

$$d_k=\begin{cases}-g_k, & \text{if} \quad k=0,\\ -g_k+\beta_k^{\mathrm{RMIL}} d_{k-1}, & \text{if} \quad k\geqslant 1,\end{cases}$$

where β_k^{RMIL} is the conjugate parameter defined as

$$\beta_k^{\mathrm{RMIL}}=\frac{g_k^{\mathrm{T}} y_{k-1}}{\|d_{k-1}\|^2}.$$

Here, $y_{k-1}=g_k-g_{k-1}$. β_k^{RMIL} has similar construction of β_k^{PRP} in [12], where β_k^{PRP} is computed as

$$\beta_k^{\mathrm{PRP}}=\frac{g_k^{\mathrm{T}} y_{k-1}}{\|g_{k-1}\|^2}, \qquad \forall\, k\geqslant 0.$$

Thus, RMIL method essentially performs a restarting search when a sufficient small step-size is generated. This can effectively avoid the Marotos effect phenomenon and improve the computation. But the descent property and the global convergence of the RMIL method are only proved under the exact line search.

It is well known that the spectral gradient method is an efficient method for solving the unconstrained optimization problem, and its search direction is computed by

$$d_k=-\theta_k g_k, \qquad \forall\, k\geqslant 0,$$

where θ_k is the spectral parameter. For different methods correspond to different choices of the parameter θ_k, please see [19–21].

To inherit and improve numerical performance of the RMIL method, we adjust its search direction based on the spectral gradient method, and establish an derivative-free iteration method for solving the problem (1),

$$d_k=\begin{cases}-F_k, & \text{if} \quad k=0,\\ -\theta_k F_k+\beta_k^{\mathrm{dRMIL}} d_{k-1}, & \text{if} \quad k\geqslant 1,\end{cases} \tag{3}$$

where

$$\beta_k^{\mathrm{dRMIL}}=\frac{F_k^{\mathrm{T}}\bar{y}_{k-1}}{\|d_{k-1}\|^2}, \tag{4}$$

$$\theta_k=1+\beta_k^{\mathrm{dRMIL}}\frac{F_k^{\mathrm{T}} d_{k-1}}{\|F_k\|^2}, \tag{5}$$

where $\bar{y}_{k-1}=F_k-F_{k-1}$ and $F_k=F(x_k)$. Throughout this paper, we denote $F(x_k)$ as F_k.

To describe our algorithm, we use the following projection operator $P_\Omega[\cdot]$,

$$P_\Omega[x]=\arg\min\{\|x-y\| \mid y\in\Omega\}, \quad x\in\mathbb{R}^n.$$

It has the famous nonexpansive property, namely, for any $x,y\in\mathbb{R}^n$,

$$\|P_\Omega[x]-P_\Omega[y]\|\leqslant\|x-y\|. \tag{6}$$

In the following, we describe the algorithm for solving the problem (1).

Algorithm 1

Step 1: Choose the initial point $x_0\in\mathbb{R}^n$, $\rho\in(0,1)$, $\sigma\in(0,1)$. Set $k=0$.

Step 2: If $x_k\in\Omega$ and $F_k=0$, stop. Otherwise, generate d_k by (3).

Step 3: Let $z_k=x_k+\alpha_k d_k$, where the step-size $\alpha_k=\max\{\rho^i \mid i=0,1,2,\dots\}$ satisfies

$$-F(x_k+\alpha_k d_k)^{\mathrm{T}} d_k\geqslant\sigma\alpha_k\|d_k\|^2. \tag{7}$$

Step 4: If $z_k\in\Omega$ and $F(z_k)=0$, stop. Otherwise, determine the next iterative point as

$$x_{k+1}=P_\Omega[x_k-\lambda_k F(z_k)],$$

where $\lambda_k=F(z_k)^{\mathrm{T}}(x_k-z_k)/\|F(z_k)\|^2$.

Step 5: Set $k := k+1$, go to step 2.

Remark 1 From (3)–(5), it is not difficult to obtain that

$$F_k^{\mathrm{T}} d_k = -\|F_k\|^2, \qquad \forall\, k \geqslant 0. \tag{8}$$

This property is independent of any line search. If F is a gradient vector of a real valued function $f : \mathbb{R}^n \to \mathbb{R}$, (8) is the sufficient descent condition which plays an important role in proving the global convergence of the conjugate gradient method.

Remark 2 From (6) and the Cauchy-Schwarz inequality, for any $k \geqslant 0$ we have

$$\begin{aligned} \|x_{k+1} - x_k\| &= \|P_\Omega[x_k - \lambda_k F(z_k)] - x_k\| \\ &\leqslant |\lambda_k| \cdot \|F(z_k)\| \\ &= \frac{|F(z_k)^{\mathrm{T}}(x_k - z_k)|}{\|F(z_k)\|^2} \cdot \|F(z_k)\| \\ &\leqslant \|x_k - z_k\|. \end{aligned} \tag{9}$$

In addition, if F is Lipschitz continuous, i.e., there exists a constant $L > 0$ such that

$$\|F(x) - F(y)\| \leqslant L\|x - y\|, \qquad \forall\, x, y \in \mathbb{R}^n. \tag{10}$$

Then from (3)–(5), the Cauchy-Schwarz inequality and (9), we have

$$\begin{aligned} \|d_k\| &\leqslant |\theta_k| \cdot \|F_k\| + |\beta_k^{\mathrm{dRMIL}}| \cdot \|d_{k-1}\| \\ &\leqslant \|F_k\| + 2|\beta_k^{\mathrm{dRMIL}}| \cdot \|d_{k-1}\| \\ &\leqslant \|F_k\| + 2\frac{\|F_k\| \cdot L\|x_k - x_{k-1}\|}{\|d_{k-1}\|} \\ &\leqslant \|F_k\| + 2\frac{\|F_k\| \cdot L\|x_{k-1} - z_{k-1}\|}{\|d_{k-1}\|} \\ &= (1 + 2\alpha_{k-1}L)\|F_k\|. \end{aligned}$$

Thus, from (8) and the definition of the step-size α_k we have

$$\|F_k\| \leqslant \|d_k\| < (1 + 2\rho L)\|F_k\|, \quad \forall\, k \geqslant 0. \tag{11}$$

GLOBAL CONVERGENCE

Now we prove the global convergence of Algorithm 1. The following assumption is needed.

Assumption A (i) The mapping F is continuous, and the solution set S of the problem (1) is nonempty convex. (ii) The mapping F is monotone and Lipschitz continuous, namely, F satisfies (2) and (10).

Lemma 1 *Let the mapping F satisfy Assumption A, the line search (7) terminates in a finite number of backtracking steps.*

Proof: Suppose that there exists an iteration index $\bar{k}$ such that the line search (7) does not hold for any nonnegative integer i, i.e.

$$-F(x_{\bar{k}} + \rho^i d_{\bar{k}})^{\mathrm{T}} d_{\bar{k}} < \sigma \rho^i \|d_{\bar{k}}\|^2.$$

By the continuity of F and $\rho \in (0,1)$, and taking $i \to \infty$ we have

$$-F(x_{\bar{k}})^{\mathrm{T}} d_{\bar{k}} \leqslant 0,$$

which contradicts with (8). Thus, the line search (7) can terminate in a finite number of backtracking steps. □

Based on Assumption A, we provide the following lemma but omit its proof, which is similar to the proof of Lemma 3.2 in [16]. This lemma shows that the sequences $\{x_k\}$ and $\{z_k\}$ generated by Algorithm 1 are bounded, and satisfy some nice properties.

Lemma 2 *Let the mapping F satisfy Assumption A. The sequences $\{x_k\}$ and $\{z_k\}$ are generated by Algorithm 1, then the sequences $\{x_k\}$ and $\{z_k\}$ are bounded. Moreover,*

$$\lim_{k\to\infty} \|x_k - z_k\| = 0. \tag{12}$$

and

$$\lim_{k\to\infty} \|x_{k+1} - x_k\| = 0. \tag{13}$$

Lemma 3 *Let the mapping F satisfy Assumption A. The sequence $\{x_k\}$ is generated by Algorithm 1, then there exists a positive constant r such that*

$$\|F_k\| \leqslant r, \qquad \forall\, k \geqslant 0. \tag{14}$$

Proof: For any $x^* \in S$, from (2) we have

$$F(z_k)^{\mathrm{T}}(z_k - x^*) = (F(z_k) - F(x^*))^{\mathrm{T}}(z_k - x^*) \geqslant 0.$$

Then, using (7) we have

$$F(z_k)^{\mathrm{T}}(x_k - x^*) \geqslant F(z_k)^{\mathrm{T}}(x_k - z_k) \geqslant \sigma\alpha_k^2\|d_k\|^2 > 0.$$

This inequality along with (6) and the Cauchy-Schwarz inequality give

$$\begin{aligned} \|x_{k+1} - x^*\|^2 &= \|P_\Omega[x_k - \lambda_k F(z_k)] - x^*\|^2 \\ &\leqslant \|x_k - x^* - \lambda_k F(z_k)\|^2 \\ &= \|x_k - x^*\|^2 - 2\lambda_k F(z_k)^{\mathrm{T}}(x_k - x^*) \\ &\quad + (\lambda_k)^2\|F(z_k)\|^2 \\ &\leqslant \|x_k - x^*\|^2 - 2\lambda_k F(z_k)^{\mathrm{T}}(x_k - z_k) \\ &\quad + (\lambda_k)^2\|F(z_k)\|^2 \\ &= \|x_k - x^*\|^2 - \lambda_k F(z_k)^{\mathrm{T}}(x_k - z_k) \\ &\leqslant \|x_k - x^*\|^2 - \frac{(F(z_k)^{\mathrm{T}}(x_k - z_k))^2}{\|F(z_k)\|^2}, \end{aligned}$$

which implies that the sequence $\{\|x_k - x^*\|\}$ is decreasing. By using (10),

$$\|F(x_k)\| = \|F(x_k) - F(x^*)\| \leqslant L\|x_k - x^*\| \leqslant L\|x_0 - x^*\|.$$

Denoting $r = L\|x_0 - x^*\|$, then (14) holds. □

Theorem 1 *Let the mapping F satisfy Assumption A. The sequences $\{x_k\}$ and $\{z_k\}$ are generated by Algorithm 1, then*

$$\lim_{k\to\infty} \inf \|F_k\| = 0. \qquad (15)$$

Proof: Suppose that the conclusion (15) is not true, there exists a constant $\mu > 0$ such that

$$\|F_k\| \geqslant \mu, \qquad \forall\, k \geqslant 0. \qquad (16)$$

From (12) and the definition of z_k in Step 3,

$$\lim_{k\to\infty} \alpha_k \|d_k\| = 0. \qquad (17)$$

From the definition of α_k, $\rho^{-1}\alpha_k$ does not satisfy the line search (7), i.e.,

$$-F(x_k + \alpha_k \rho^{-1} d_k)^{\mathrm{T}} d_k < \sigma \rho^{-1} \alpha_k \|d_k\|^2. \qquad (18)$$

From (8), (10)–(11) and (18),

$$\begin{aligned}\|F_k\|^2 &= -F_k^{\mathrm{T}} d_k \\ &= \left[F(x_k + \rho^{-1}\alpha_k d_k) - F_k\right]^{\mathrm{T}} d_k \\ &\quad - F(x_k + \rho^{-1}\alpha_k d_k)^{\mathrm{T}} d_k \\ &\leqslant \rho^{-1}(L+\sigma)\alpha_k \|d_k\|^2 \\ &\leqslant \rho^{-1}(L+\sigma)(1+2\rho L)\alpha_k \|d_k\| \cdot \|F_k\|.\end{aligned}$$

From (16), it is easy to get

$$\alpha_k \|d_k\| \geqslant \frac{\rho \|F_k\|}{(L+\sigma)(1+2\rho L)} \geqslant \frac{\rho \mu}{(L+\sigma)(1+2\rho L)} > 0.$$

This generates a contradiction with (17). Thus, the proof is completed. □

NUMERICAL EXPERIMENTS

In this section, we select some nonlinear monotone equations, and test Algorithm 1 (SCG method). At the same time, we compare the performance of SCG method with PCG method presented by Liu and Li [16]. All codes were written in Matlab.

The followings are test problems, where F is defined as $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^{\mathrm{T}}$.

Problem 1 The logarithmic function in [22] with the convex constraint $\Omega = \mathbb{R}_n^+$, i.e.

$$f_i(x) = \log(x_i + 1) - \frac{x_i}{n}, \qquad i = 1, 2, \dots, n,$$

where $x_0 = (1, 1, \dots, 1)^{\mathrm{T}}$.

Problem 2 The function in [10], i.e.,

$$f_i(x) = x_i - \sin|x_i - 1|, \qquad i = 1, 2, \dots, n,$$

where $x_0 = (-0.5, -0.5, \dots, -0.5)^{\mathrm{T}}$ and $\Omega = \left\{x \in \mathbb{R}^n \mid \sum_{i=1}^n x_i \leqslant n,\ x_i \geqslant -1,\ i = 1, 2, 3, \dots, n\right\}$.

Problem 3 The gradient of ARWHEAD function in the CUTEr library [23] with the convex constraint $\Omega = \mathbb{R}_+^n$, i.e.,

$$\begin{aligned} f_i(x) &= -4 + 4x_i(x_i^2 + x_n^2), \quad i = 1, 2, \dots, n-1, \\ f_n(x) &= 4x_n \sum_{i=1}^{n-1} (x_i^2 + x_n^2), \end{aligned}$$

where $x_0 = (0, 0, \dots, 0)^{\mathrm{T}}$.

Problem 4 The Trigexp function in [24] with the convex constraint $\Omega = \mathbb{R}_+^n$, i.e.,

$$\begin{aligned} f_1(x) &= 3x_1^3 + 2x_2 - 5 + \sin(x_1 - x_2)\sin(x_1 + x_2), \\ f_i(x) &= -x_{i-1}\, \mathrm{e}^{x_{i-1} - x_i} + x_i(4 + 3x_i^2) + 2x_{i+1} \\ &\quad + \sin(x_i - x_{i+1})\sin(x_i + x_{i+1}) - 8, \\ f_n(x) &= -x_{n-1}\, \mathrm{e}^{x_{n-1} - x_n} + 4x_n - 3, \end{aligned}$$

where $i = 2, 3, \dots, n-1$ and $x_0 = (2, 2, \dots, 2)^{\mathrm{T}}$.

Problem 5 The gradient of ENGVAL1 function in the CUTEr library [23] with the convex constraint $\Omega = \mathbb{R}_+^n$, i.e.,

$$\begin{aligned} f_1(x) &= 4x_1(x_1^2 + x_2^2) - 4, \\ f_i(x) &= 4x_i(x_{i-1}^2 + x_i^2) + 4x_i(x_i^2 + x_{i+1}^2) - 4, \\ f_n(x) &= 4x_n(x_{n-1}^2 + x_n^2), \end{aligned}$$

where $i = 2, 3, \dots, n-1$ and $x_0 = (2, 2, \dots, 2)^{\mathrm{T}}$.

Problem 6 The discrete boundary value problem in [25] with the convex constraint $\Omega = \{x \in \mathbb{R}^n \mid x_i \geqslant -5,\ i = 1, 2, \dots, n\}$, i.e.,

$$\begin{aligned} f_i(x) &= 2x_1 + 0.5h^2(x_1 + h)^3 - x_2, \\ f_i(x) &= 2x_i + 0.5h^2(x_i + ih)^3 - x_{i-1} + x_{i+1}, \\ f_n(x) &= 2x_n + 0.5h^2(x_n + nh)^3 - x_{n-1}, \end{aligned}$$

where $i = 2, 3, \ldots, n-1$, $h = 1/(n+1)$ and $x_0 = (-1, -1, \ldots, -1)^{\mathrm{T}}$.

Problem 7 The five-diagonal system in [22] with the convex constraint $\Omega = \mathbb{R}^n_+$.

$$\begin{aligned}
f_1(x) &= 4(x_1 - x_2^2) + x_2 - x_3^2,\\
f_2(x) &= 8x_2(x_2^2 - x_1) - 2(1 - x_2) + 4(x_2 - x_3^2)\\
&\quad + x_3 - x_4^2,\\
f_i(x) &= 8x_i(x_i^2 - x_{i-1}) - 2(1 - x_i) + 4(x_i - x_{i+1}^2)\\
&\quad + x_{i-1}^2 - x_{i-2} + x_{i+1} - x_{i+2}^2,\\
f_{n-1}(x) &= 8x_{n-1}(x_{n-1}^2 - x_{n-2}) - 2(1 - x_{n-1})\\
&\quad + 4(x_{n-1} - x_n^2) + x_{n-2}^2 - x_{n-3},\\
f_n(x) &= 8x_n(x_n^2 - x_{n-1}) - 2(1 - x_n) + x_{n-1}^2 - x_{n-2},
\end{aligned}$$

where $i = 3, 4, \ldots, n-2$ and $x_0 = (0, 0, \ldots, 0)^{\mathrm{T}}$.

The parameter values of both methods are listed as follows. PCG method: $\xi = 1$, $\rho = 0.55$, $\sigma = 10^{-4}$ and $r = 0.1$, which come from [16]. SCG method: $\rho = 0.65$ and $\sigma = 10^{-4}$. They implement the same stopping criterion,

$$\|F(x_k)\| \leqslant 10^{-5} \quad \text{or} \quad \|F(z_k)\| \leqslant 10^{-5} \quad \text{if} \quad z_k \in \Omega.$$

In this part, we tested the problems with the different number of variables but with the given initial points. The computational results are reported in Table 1, which contains the number of iterates (Niter), CPU time in second (time) and the final norm of function values ($\|F\|$). The Dim stands for the dimensions of the problems. From Table 1, there are 26 experiments for which the SCG method performed better than the PCG method in terms of number of iterates, and there are 17 experiments which the SCG method performed better than the PCG method in terms of CPU time, except for 8 experiments which amount to the same CPU time. This implies that SCG method performs better than PCG method for the given problems.

CONCLUSION

In this paper we proposed an effective algorithm for solving large-scale nonlinear monotone equations with convex constraints, which does not require the Jacobian matrix of F or an approximation of the Jacobian matrix of F in the analysis and computation. The motivation is to inherit the excellence of the RMIL method and modify its search direction. Numerical results show that the proposed method can solve the selected problems successfully, and performs better than the competitor.

Table 1 The numerical results obtained by SCG and PCG methods.

Prob	Dim	SCG method Niter/time/$\|F\|$	PCG method Niter/time/$\|F\|$
1	1000	5/0.02/3.599×10^{-8}	7/0.02/6.431×10^{-6}
	5000	5/0.02/6.263×10^{-9}	8/0.02/1.013×10^{-6}
	10000	5/0.02/3.618×10^{-9}	8/0.05/1.422×10^{-6}
	15000	6/0.04/5.017×10^{-6}	8/0.05/1.736×10^{-6}
2	1000	8/0.03/2.346×10^{-6}	14/0.03/8.536×10^{-6}
	5000	8/0.03/5.246×10^{-6}	15/0.03/8.576×10^{-6}
	10000	8/0.05/7.419×10^{-6}	16/0.06/5.449×10^{-6}
	15000	8/0.03/9.086×10^{-6}	16/0.09/6.674×10^{-6}
3	1000	9/0.03/5.489×10^{-6}	19/0.03/6.190×10^{-6}
	5000	10/0.03/1.166×10^{-6}	20/0.05/5.525×10^{-6}
	10000	10/0.05/1.649×10^{-6}	20/0.05/7.814×10^{-6}
	15000	10/0.05/2.020×10^{-6}	20/0.08/9.571×10^{-6}
4	1000	14/0.05/4.154×10^{-6}	17/0.06/4.017×10^{-6}
	5000	15/0.09/3.247×10^{-6}	17/0.11/6.242×10^{-6}
	10000	15/0.19/6.447×10^{-6}	17/0.14/7.053×10^{-6}
	15000	15/0.19/9.863×10^{-6}	17/0.17/8.811×10^{-6}
5	1000	25/0.05/7.209×10^{-6}	82/0.12/9.859×10^{-6}
	5000	25/0.05/8.550×10^{-6}	32/0.08/5.945×10^{-6}
	10000	27/0.09/4.448×10^{-6}	31/0.09/2.844×10^{-6}
	15000	24/0.12/3.448×10^{-6}	33/0.14/4.122×10^{-6}
6	1000	26/0.05/4.190×10^{-6}	27/0.06/8.123×10^{-6}
	5000	26/0.07/8.563×10^{-6}	27/0.08/8.328×10^{-6}
	10000	27/0.18/6.666×10^{-6}	29/0.18/5.766×10^{-6}
	15000	27/0.23/6.744×10^{-6}	26/0.21/7.363×10^{-6}
7	1000	1273/1.53/9.668×10^{-6}	1567/1.84/9.890×10^{-6}
	5000	1316/2.42/9.695×10^{-6}	1644/2.67/9.933×10^{-6}
	10000	1290/4.60/9.912×10^{-6}	1615/5.09/9.986×10^{-6}
	15000	1442/7.68/9.708×10^{-6}	1720/8.10/9.967×10^{-6}

REFERENCES

1. Dennis JE, Moré JJ (1974) A characterization of superlinear convergence and its application to quasi-Newton methods. *Math Comput* **28**, 549-560.
2. Li DH, Fukushima M (1999) A global and superlinear convergence Gauss-Newton-based BFGS method for symmetric nonlinear equations. *SIAM J Numer Anal* **37**, 152–172.
3. Zhou G, Toh KC (2005) Superlinear convergence of a Newton-type algorithm for monotone equations. *J Optimiz Theory Appl* **125**, 205–221.
4. Solodov MV, Svaiter BF (1999) A globally convergent inexact Newton method for systems of monotone equations. In: Fukushima M, Qi L (eds) *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*, Applied Optimization Vol 22, Springer, Boston, MA, pp 355–369.
5. Zhou WJ, Li DH (2008) A globally convergent BFGS method for nonlinear monotone equations without any merit functions. *Math Comput* **77**, 2231–2240.
6. Yamashita N, Fukushima M (2001) On the rate of convergence of the Levenberg-Marquardt method. In: Alefeld G, Chen X (eds) *Topics in Numerical Analy-*

sis. Computing Supplementa Vol 15, Springer, Vienna, pp 239–249.
7. Kanzow C, Yamashita N, Fukushima M (2005) Levenberg-Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints. *J Comput Appl Math* **172**, 375–397.
8. Wang CW, Wang YJ (2009) A superlinearly convergent projection method for constrained systems of nonlinear equations. *J Global Optim* **44**, 283–296.
9. Zhang L, Zhou W (2006) Spectral gradient projection method for solving nonlinear monotone equations. *J Comput Appl Math* **196**, 478–484.
10. Yu ZS, Lin J, Sun J, Xiao YH, Liu LY, Li ZH (2009) Spectral gradient projection method for monotone nonlinear equations with convex constraints. *Appl Numer Math* **59**, 2416–2423.
11. Li QN, Li DH (2011) A class of derivative-free methods for large-scale nonlinear monotone equations. *IMA J Numer Anal* **31**, 1625–1635.
12. Polak E, Ribire G (1969) Note sur la xonvergence de directions conjugees. *Rev Fr Info Rech Oper* **3**, 35–43.
13. Liu JK, Li SJ (2016) A three-term derivative-free projection method for systems of nonlinear monotone equations. *Calcolo* **53**, 427–450.
14. Xiao YH, Zhu H (2013) A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing. *J Math Anal Appl* **405**, 310–319.
15. Hager WW, Zhang H (2005) A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM J Optimiz* **16**, 170–192.
16. Liu JK, Li SJ (2015) A projection method for convex constrained monotone nonlinear equations with applications. *Comput Math Appl* **70**, 2442–2453.
17. Cao HS (2021) A three-term derivative-free projection method for convex constrained monotone equations. *ScienceAsia* **47**, 235–240.
18. Rivaie M, Mamat M, June LW, Mohd I (2012) A new class of nonlinear conjugate gradient coefficients with global convergence properties. *Appl Math Comput* **218**, 11323–11332.
19. Barizilai JM, Borwein M (1988) Two point step size gradient methods. *IMA J Numer Anal* **8**, 141–148.
20. Han L, Yu GH, Guan LT (2008) Multivate spectral gradient method for unconstrained optimization. *Appl Math Comput* **201**, 621–630.
21. Liu JK, Li SJ (2015) Spectral gradient method for impulse noise removal. *Optim Lett* **9**, 1341–1351.
22. Cruz WL, Raydan M (2003) Nonmonotone spectral methods for large- scale nonlinear systems. *Optim Methods Softw* **18**, 583–599.
23. Conn AR, Gould NIM, Toint PL (1995) CUTE: constrained and unconstrained testing environment. *ACM Trans Math Softw* **21**, 123–160.
24. Gomez-Ruggiero MA, Martínez JM, Moretti AC (1992) Comparing algorithms for solving sparse nonlinear systems of equations. *SIAM J Sci Stat Comput* **13**, 459–483.
25. Moré JJ, Garbow BS, Hillström KE (1981) Testing uncosntrained optimization software. *ACM Trans Math Softw* **7**, 17–41.