RESEARCH ARTICLE
doi: 10.2306/scienceasia1513-1874.2021.018

A three-term derivative-free projection method for convex constrained monotone equations

Haisong Cao

School of Mathematics and Statistics, North China University of Water Resources and Electric Power, Zhengzhou 450045 China

e-mail: hscao678@163.com

Received 9 Sep 2020
Accepted 14 Dec 2020

ABSTRACT: In this work, we propose a three-term derivative-free projection method to solve nonlinear monotone equations with convex constraints based on the structures of the famous Dai-Yuan (DY) conjugate gradient method and the three-term conjugate gradient method. The proposed derivative-free method is suitable for solving large-scale problems due to its simple structure and lower storage requirement. The search direction satisfies the sufficient descent property independent of any line search. The global convergence is established under some conditions. The preliminary numerical results indicate that the proposed method is robust and effective.

KEYWORDS: nonlinear monotone equations, three-term conjugate gradient method, projection method, global convergence

MSC2010: 90C30 65K05

INTRODUCTION

Iterative projection method is one of the most popular and effective methods for solving large-scaled nonlinear equations

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

where $F:\Omega\to\mathbb{R}^n$ is a continuous and monotone function, and $\Omega\subseteq\mathbb{R}^n$ is a nonempty closed convex set. A function F is said to be monotone if it satisfies

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

Many practical problems can be transformed into solving nonlinear equations, for example, the chemical equilibrium systems [1], the economic equilibrium problems [2], the power flow systems [3]. This is why many researchers are keen on iterative projection methods for solving nonlinear equations for many years, see recent references [4–14].

Three-term conjugate gradient method is one of the most popular methods for solving large-scale unconstrained optimization problems because of its good descent property, computing performance and stable convergence, see references [15–21]. This stimulates many researchers to use the structures of three-term conjugate gradient methods for solving nonlinear equations. For example, based on the hyperplane projection method [22], Li and Wang [23] proposed a three-term Fletcher-Reeves derivative-free method for large-scale symmetric nonlinear equations, which is an extension of the modified Fletcher-Reeves conjugate gradient method [24]. Gao and He [25] chose a part of the Liu-Storey (LS) conjugate parameter as a new conjugate parameter and further proposed a three-term conjugate gradient (TTCG) method for solving nonlinear monotone equations with convex constraints. Motivated by the modified Dai-Yuan (DY) method [26], Koorapetse and Kaelo [27] proposed a new three-term conjugate gradient-based projection method to solve nonlinear monotone equations with convex constraints. The common feature of these methods is that they are stable in descent property and convergence, and the computing performance is satisfactory.

The DY conjugate gradient method [28] is one of the most famous conjugate gradient methods for solving unconstrained optimization problems, which is known for the stability. In order to establish the stable and effective method for solving monotone nonlinear equations with convex constraints, in this paper we propose a three-term derivative-free projection method based on the structures of the methods [25, 27] and the DY conjugate gradient method [28]. This method inherits the stability of the DY method, and greatly improves its computing performance. The search direction of the proposed method satisfies the sufficient descent property independent of any line search. The global convergence can be obtained under some conditions. The numerical results show that the proposed method is stable and effective by comparing with the PDY method [5] and the TTCG method [25].

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

ALGORITHM

In this section, we firstly give the projection operator $P_\Omega[\cdot]$ which is defined as a mapping from $\mathbb{R}^n$ to a nonempty closed convex set Ω, i.e.,

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

This operator has a famous non-expansive property, i.e., for any $x, y \in \mathbb{R}^n$,

$$\|P_\Omega[x] - P_\Omega[y]\| \leqslant \|x-y\|. \tag{3}$$

The DY method [28] is one of the traditional conjugate gradient methods for solving unconstrained optimization problems, in which its search direction is defined as

$$d_0 = -g_0, \qquad d_k = -g_k + \beta_k^{\mathrm{DY}} d_{k-1},$$

where g_k is the gradient value of the objective function f at point x_k, $\beta_k^{\mathrm{DY}} = \|g_k\|^2 / \big(d_{k-1}^{\mathrm{T}} y_{k-1}\big)$, $y_{k-1} = g_k - g_{k-1}$. This method is favored by many researchers because of its stable convergence.

In the following, based on the DY method [28] and the structure of the three-term conjugate gradient method, the specific steps of our proposed method are presented in Algorithm 1. For simplicity, we abbreviate $F(x_k)$ as F_k.

Algorithm 1 (TTMDY method)
Step 0: Select $x_0 \in \mathbb{R}^n$, $\rho \in (0,1)$, $\beta, \sigma > 0$, $r > 1$. Set $k = 0$.
Step 1: Set $d_0 = -F_0$.
Step 2: Set $z_k = x_k + \alpha_k d_k$, where the step-size $\alpha_k = \max\{\beta\rho^i \mid i = 0,1,2,\ldots\}$ satisfies

$$-F(x_k + \alpha_k d_k)^{\mathrm{T}} d_k \geqslant \sigma \alpha_k \|d_k\|^2. \tag{4}$$

Step 3: If $z_k \in \Omega$ and $F(z_k) = 0$, stop. Otherwise, obtain the next iterative point as

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

where

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

Step 4: If $F(x_{k+1}) = 0$, stop. Otherwise, obtain the next search direction d_{k+1} as follows:

$$d_{k+1} = -F_{k+1} + \beta_{k+1}^{\mathrm{mDY}} d_k + \theta_{k+1} y_k. \tag{5}$$

Here

$$\beta_{k+1}^{\mathrm{mDY}} = \frac{\|F_{k+1}\|^2}{d_k^{\mathrm{T}} w_k}, \quad \theta_{k+1} = -\frac{F_{k+1}^{\mathrm{T}} d_k}{d_k^{\mathrm{T}} w_k},$$

where $w_k = y_k + t_k d_k$, $y_k = F_{k+1} - F_k$, and $t_k = r\|F_k\|/\|d_k\| + \max\big\{0, -d_k^{\mathrm{T}} y_k / d_k^{\mathrm{T}} d_k\big\}$.
Step 5: Set $k := k+1$, go to step 2.

The following remark shows that the parameters β_k^{mDY} and θ_k defined in the TTMDY method are meaningful before the solution of the problem (1) is reached.

Remark 1 From the definitions of w_k and t_k, for $\forall k \geqslant 1$ we have

$$\begin{aligned} d_{k-1}^{\mathrm{T}} w_{k-1} &= d_{k-1}^{\mathrm{T}} y_{k-1} + t_{k-1}\|d_{k-1}\|^2 \\ &\geqslant d_{k-1}^{\mathrm{T}} y_{k-1} + r\|F_{k-1}\| \cdot \|d_{k-1}\| - d_{k-1}^{\mathrm{T}} y_{k-1} \\ &= r\|F_{k-1}\| \cdot \|d_{k-1}\|. \end{aligned} \tag{6}$$

The following theorem indicates that the search direction d_k obtained by the TTMDY method satisfies the sufficient descent property which plays an important role in proving the global convergence. Moreover, this property can guarantee the iterative points to approach the solution of the problem (1) step by step.

Theorem 1 *Let the sequences $\{d_k\}$ and $\{F_k\}$ be generated by the TTMDY method, then*

$$F_k^{\mathrm{T}} d_k \leqslant -\big(1 - \tfrac{1}{r}\big)\|F_k\|^2, \quad \forall k \geqslant 0. \tag{7}$$

Proof: By (5), for $k \geqslant 1$ we have

$$\begin{aligned} F_k^{\mathrm{T}} d_k &= -\|F_k\|^2 + \beta_k^{\mathrm{mDY}} F_k^{\mathrm{T}} d_{k-1} + \theta_k F_k^{\mathrm{T}} y_{k-1} \\ &= -\|F_k\|^2 + \frac{F_k^{\mathrm{T}} F_{k-1} \cdot F_k^{\mathrm{T}} d_{k-1}}{d_{k-1}^{\mathrm{T}} w_{k-1}} \\ &\leqslant -\|F_k\|^2 + \frac{\|F_k\|^2 \|F_{k-1}\| \cdot \|d_{k-1}\|}{r\|F_{k-1}\| \cdot \|d_{k-1}\|} \\ &= -\big(1 - \tfrac{1}{r}\big)\|F_k\|^2, \end{aligned}$$

where the first inequality follows from the Cauchy-Schwarz inequality and (6). In addition, from Step 1 we have $F_0^{\mathrm{T}} d_0 = -\|F_0\|^2$. Thus, (7) holds. □

GLOBAL CONVERGENCE

In this section, we always assume $F(x_k) \neq 0$ for any $k \geqslant 0$, otherwise we obtain the solution of the problem (1). We also need the following assumptions to prove the global convergence of the TTMDY method.

Assumption A

(i) The function $F(\cdot)$ is monotone on $\mathbb{R}^n$, and the solution set Ω^* of the problem (1) is nonempty.

(ii) The function $F(\cdot)$ is Lipschitz continuous on $\mathbb{R}^n$, i.e., there exists a positive constant L such that

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

The following lemma shows that if the sequence $\{x_k\}$ is generated by the TTMDY method and x^* is a solution of the problem (1), then the sequence $\{\|x_k - x^*\|\}$ is decreasing and convergent, thus the sequence $\{x_k\}$ is bounded.

Lemma 1 *Suppose that Assumption A holds, and the sequences $\{x_k\}$ and $\{z_k\}$ are generated by the TTMDY method. For any $x^* \in \Omega^*$ we have*

$$\|x_{k+1}-x^*\|^2 \leqslant \|x_k-x^*\|^2 - c\|x_k - z_k\|^4, \quad c \in (0,1),$$

and the sequence $\{x_k\}$ is bounded. Furthermore,

$$\lim_{k\to\infty} \|x_k - z_k\| = 0. \tag{8}$$

Proof: The conclusions can be proved follows from Lemma 3.1 in [29]. Here we omit it. □

Remark 2 From the continuity of the function $F(\cdot)$ and the boundedness of the sequence $\{x_k\}$, it holds that the sequence $\{\|F_k\|\}$ is bounded, i.e., there exists a constant $M > 0$ such that

$$\|F_k\| \leqslant M, \quad \forall k \geqslant 0. \tag{9}$$

Lemma 2 *Suppose Assumption A holds, and the sequences $\{d_k\}$ and $\{F_k\}$ are generated by the TTMDY method. Then we have*

$$\|d_k\| \leqslant \Big(1 + \tfrac{M}{r\|F_{k-1}\|} + \tfrac{2M}{r\|F_{k-1}\|}\Big)\|F_k\|, \quad \forall k \geqslant 0. \tag{10}$$

Proof: By (9) we have

$$\|y_{k-1}\| \leqslant \|F_k\| + \|F_{k-1}\| \leqslant 2M, \quad \forall k \geqslant 1.$$

Thus, for any $k \geqslant 1$ it follows from (6) and the Cauchy-Schwarz inequality that

$$\begin{aligned}\|d_k\| &\leqslant \|F_k\| + \beta_k^{\mathrm{mDY}}\|d_{k-1}\| + |\theta_k|\cdot\|y_{k-1}\| \\ &\leqslant \|F_k\| + \frac{\|F_k\|^2}{r\|F_{k-1}\|\cdot\|d_{k-1}\|}\|d_{k-1}\| \\ &\quad + 2M\frac{\|F_k\|\cdot\|d_{k-1}\|}{r\|F_{k-1}\|\cdot\|d_{k-1}\|} \\ &= \Big(1 + \tfrac{M}{r\|F_{k-1}\|} + \tfrac{2M}{r\|F_{k-1}\|}\Big)\|F_k\|.\end{aligned}$$

For $k = 0$, we have $\|d_0\| = \|F_0\|$. Thus, the conclusion (10) holds. □

Theorem 2 *Suppose Assumption A holds, and the sequences $\{F_k\}$ and $\{d_k\}$ are generated by the TTMDY method. Then we have*

$$\liminf_{k\to\infty} \|F_k\| = 0. \tag{11}$$

Proof: Assume that (11) does not hold, i.e., there exists a constant $m > 0$ such that

$$\|F_k\| \geqslant m, \quad \forall k \geqslant 0. \tag{12}$$

It follows from (10) that

$$\|d_k\| \leqslant \big(1 + \tfrac{3M}{rm}\big)\|F_k\|, \quad \forall k \geqslant 0. \tag{13}$$

From the line search (4), if $\alpha_k \neq \beta$, then $\alpha_k' = \frac{\alpha_k}{\rho}$ satisfies

$$-F(x_k + \alpha_k' d_k)^{\mathrm{T}} d_k < \sigma\alpha_k'\|d_k\|^2.$$

This inequality together with (7) yield

$$\begin{aligned}\big(1-\tfrac{1}{r}\big)\|F_k\|^2 &\leqslant -F_k^{\mathrm{T}} d_k \\ &= (F(x_k+\alpha_k' d_k) - F_k)^{\mathrm{T}} d_k - F(x_k + \alpha_k' d_k)^{\mathrm{T}} d_k \\ &\leqslant L\alpha_k'\|d_k\|^2 + \sigma\alpha_k'\|d_k\|^2 \\ &= (L+\sigma)\alpha_k'\|d_k\|^2.\end{aligned}$$

where the second inequality follows from the Cauchy-Schwarz inequality and Assumption A(ii). Thus, we have

$$\alpha_k\|d_k\| \geqslant \frac{\rho\big(1-\frac{1}{r}\big)\|F_k\|^2}{(L+\sigma)\|d_k\|},$$

From (12) and (13) we have

$$\alpha_k\|d_k\| \geqslant \frac{\rho\big(1-\frac{1}{r}\big)m}{(L+\sigma)\big(1+\frac{3M}{rm}\big)},$$

which contradicts (8). □

NUMERICAL RESULTS

In this section, we test some nonlinear equations, and compare the performance of the TTMDY method with those of the PDY method [5] and the TTCG method [25]. The parameters used in the TTCG method and the PDY method come from the corresponding references. The parameters in the TTMDY method are selected as $\rho = 0.75$, $\sigma = 0.001$, and $r = 1.1$. All the methods are coded in MATLAB 7.0. In our experiments, the methods are stopped whenever the inequality $\|F(x_k)\| \leqslant 10^{-6}$ or $\|F(z_k)\| \leqslant 10^{-6}$, or the total number of iterations exceeds 1000.

Problem 4.1. The problem is tridiagonal exponential problem [30] with additional constrained condition $\Omega = \mathbb{R}^n_+$, i.e.,

$$\begin{aligned} F_1(x) &= x_1 - \mathrm{e}^{\cos(\frac{x_1+x_2}{n+1})}, \\ F_i(x) &= x_i - \mathrm{e}^{\cos(\frac{x_{i-1}+x_i+x_{i+1}}{n+1})}, \quad i = 2,3,\dots,n-1, \\ F_n(x) &= x_n - \mathrm{e}^{\cos(\frac{x_{n-1}+x_n}{n+1})}. \end{aligned}$$

Problem 4.2. The problem comes from [31], i.e.,

$$F_i(x) = \mathrm{e}^{x_i} - 1, \quad i = 1,2,3,\dots,n$$

and $\Omega = \mathbb{R}^n_+$.

Problem 4.3. The problem comes from [25], i.e.,

$$F_i(x) = x_i - \sin(|x_i| - 1), \quad i = 1,2,3,\dots,n,$$

where $\Omega = \left\{x \in \mathbb{R}^n \,\middle|\, \sum_{i=1}^n x_i \leqslant n,\ x_i \geqslant -1, \quad i = 1,2,3,\dots,n\right\}$.

Problem 4.4. The problem can be viewed as a modification of Problem 4 in [25], i.e.,

$$\begin{aligned} F_1(x) &= x_1 + \sin x_1 - 1, \\ F_i(x) &= -x_{i-1} + 2x_i + \sin x_i - 1, \quad i = 2,3,\dots,n-1, \\ F_n(x) &= x_n + \sin x_n - 1, \end{aligned}$$

and $\Omega = \mathbb{R}^n_+$.

In order to show the performance of the TTMDY method, we test the given problems with various dimensions n = 1000, 3000, 5000, 7000, 10 000 and some different initial points: $x_0^1 = (0.1, 0.1, \cdots, 0.1)^{\mathrm{T}}$, $x_0^2 = (0.5, 0.5, \cdots, 0.5)^{\mathrm{T}}$, $x_0^3 = (1, 1, \cdots, 1)^{\mathrm{T}}$, $x_0^4 = (2, 2, \cdots, 2)^{\mathrm{T}}$, $x_0^5 = (3, 3, \cdots, 3)^{\mathrm{T}}$, respectively. The results are presented in Tables 1–4, where we report the dimension of the problem (Dim), the number of iterations (Niter), the number of the function evaluations (NF), and the CPU time in seconds (time). Tables 1–4 indicate that these methods are able to solve all given test problems successfully.

Table 1 The numerical results of Problem 4.1.

	Dim	TTMDY	TTCG	PDY
		Niter/NF/time	Niter/NF/time	Niter/NF/time
x_0^1	1000	17/71/0.01	36/145/0.01	16/34/0.01
	3000	18/76/0.01	37/149/0.03	17/36/0.02
	5000	15/62/0.02	38/153/0.04	17/36/0.02
	7000	13/53/0.02	38/153/0.05	17/38/0.03
	10000	18/76/0.03	38/153/0.07	15/35/0.04
x_0^2	1000	17/71/0.01	36/145/0.01	16/34/0.01
	3000	13/53/0.01	37/149/0.03	16/34/0.02
	5000	20/86/0.02	37/149/0.04	17/36/0.02
	7000	15/62/0.03	38/153/0.05	17/36/0.03
	10000	17/71/0.04	38/153/0.07	17/38/0.04
x_0^3	1000	20/86/0.01	35/141/0.01	16/34/0.01
	3000	15/62/0.01	36/145/0.03	16/34/0.02
	5000	19/81/0.02	37/149/0.04	16/34/0.02
	7000	15/62/0.04	37/149/0.05	17/36/0.03
	10000	13/53/0.03	38/153/0.07	17/36/0.04
x_0^4	1000	22/96/0.01	34/137/0.01	15/32/0.01
	3000	17/71/0.02	35/141/0.03	15/32/0.02
	5000	17/72/0.02	35/141/0.04	16/34/0.02
	7000	18/77/0.02	36/145/0.05	16/34/0.03
	10000	13/53/0.03	36/145/0.06	16/34/0.04
x_0^5	1000	20/87/0.01	32/129/0.01	14/30/0.01
	3000	18/77/0.02	33/133/0.02	15/32/0.02
	5000	12/49/0.02	33/133/0.03	15/32/0.02
	7000	17/72/0.02	34/137/0.04	15/32/0.03
	10000	14/58/0.03	34/137/0.06	15/32/0.03

Table 2 The numerical results of Problem 4.2.

	Dim	TTMDY	TTCG	PDY
		Niter/NF/time	Niter/NF/time	Niter/NF/time
x_0^1	1000	8/32/0.01	30/121/0.01	7/15/0.01
	3000	9/36/0.01	31/125/0.01	8/17/0.01
	5000	9/36/0.01	31/125/0.01	8/17/0.01
	7000	9/36/0.01	32/129/0.02	8/17/0.01
	10000	9/36/0.02	32/129/0.02	8/17/0.01
x_0^2	1000	8/32/0.01	32/129/0.01	12/25/0.01
	3000	9/36/0.01	33/133/0.01	12/25/0.01
	5000	9/36/0.01	34/137/0.02	12/25/0.01
	7000	9/36/0.01	34/137/0.02	13/27/0.01
	10000	9/36/0.01	34/137/0.02	13/27/0.02
x_0^3	1000	9/37/0.01	33/133/0.01	13/27/0.01
	3000	9/37/0.01	34/137/0.01	13/27/0.01
	5000	9/37/0.01	34/137/0.02	14/29/0.01
	7000	9/37/0.01	34/137/0.02	14/29/0.01
	10000	9/37/0.02	35/140/0.02	14/29/0.02
x_0^4	1000	10/45/0.01	34/138/0.01	15/32/0.01
	3000	10/45/0.01	35/142/0.01	15/32/0.01
	5000	11/49/0.01	35/142/0.02	14/32/0.01
	7000	11/49/0.01	35/142/0.02	14/32/0.01
	10000	11/49/0.02	36/146/0.02	15/36/0.02
x_0^5	1000	10/47/0.01	35/144/0.01	15/33/0.01
	3000	10/47/0.01	36/148/0.01	12/31/0.01
	5000	11/51/0.01	36/148/0.02	12/31/0.01
	7000	11/51/0.01	36/148/0.02	12/31/0.01
	10000	11/51/0.02	37/152/0.03	15/42/0.02

To compare the three methods with respect to the number of iterations, the number of the function evaluations and the CPU time comprehensively and intuitively, in this paper we apply the performance profiles of Dolan and Moré [32], which is an efficient tool for evaluating and comparing the performances of iterative methods. From the reference [32], we know that the performance profile for each method is measured by the ratio of its computational outcome compared to the computational outcome of the best method, which means that the method with high performance profile is preferable or represents the better method. By the technique in the reference [32], I obtained the performance profiles of the TTMDY method, TTCG method and PDY method, see Figs. 1–3. These figures show the TTMDY

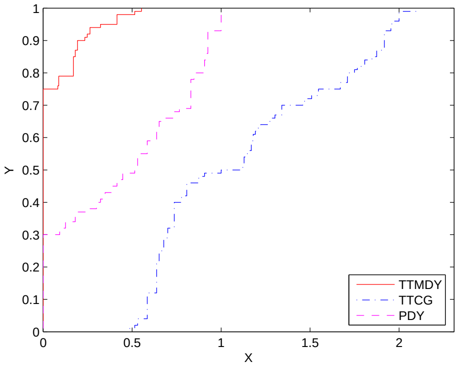

Fig. 1 Performance profiles with respect to the number of iterations.

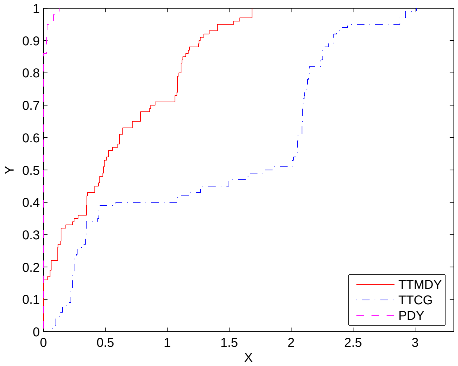

Fig. 2 Performance profiles with respect to the number of the function evaluations.

Table 3 The numerical results of Problem 4.3.

	Dim	TTMDY	TTCG	PDY
		Niter/NF/time	Niter/NF/time	Niter/NF/time
x_0^1	1000	9/53/0.01	14/57/0.01	15/45/0.01
	3000	9/53/0.01	14/57/0.01	16/48/0.01
	5000	9/53/0.01	14/57/0.01	16/48/0.01
	7000	9/53/0.02	14/57/0.01	16/48/0.01
	10000	9/53/0.02	14/57/0.01	16/48/0.02
x_0^2	1000	8/45/0.01	11/44/0.01	6/14/0.01
	3000	8/45/0.01	11/44/0.01	6/14/0.01
	5000	8/45/0.01	12/48/0.01	6/14/0.01
	7000	8/45/0.01	12/48/0.01	7/17/0.01
	10000	8/45/0.03	12/48/0.01	7/17/0.01
x_0^3	1000	9/51/0.01	14/56/0.01	17/51/0.01
	3000	9/51/0.01	15/60/0.01	17/51/0.01
	5000	9/51/0.01	15/60/0.01	18/54/0.01
	7000	9/51/0.02	15/60/0.01	18/54/0.02
	10000	9/51/0.02	15/60/0.01	18/54/0.02
x_0^4	1000	10/55/0.01	15/59/0.01	18/53/0.01
	3000	10/55/0.01	15/59/0.01	18/53/0.01
	5000	10/55/0.01	15/59/0.01	19/56/0.01
	7000	10/55/0.02	15/59/0.01	19/56/0.02
	10000	10/55/0.02	16/63/0.01	19/56/0.02
x_0^5	1000	10/54/0.01	15/59/0.01	17/49/0.01
	3000	10/54/0.01	15/59/0.01	19/55/0.01
	5000	10/54/0.01	16/63/0.01	19/55/0.01
	7000	10/54/0.01	16/63/0.01	19/55/0.02
	10000	10/54/0.02	16/63/0.01	20/59/0.02

Table 4 The numerical results of Problem 4.4.

	Dim	TTMDY	TTCG	PDY
		Niter/NF/time	Niter/NF/time	Niter/NF/time
x_0^1	1000	8/48/0.01	13/53/0.01	8/22/0.01
	3000	9/54/0.01	13/53/0.01	8/22/0.01
	5000	9/54/0.01	13/53/0.01	9/25/0.01
	7000	9/54/0.01	13/53/0.01	9/25/0.02
	10000	9/54/0.02	14/57/0.02	9/25/0.02
x_0^2	1000	7/42/0.01	10/41/0.01	10/30/0.01
	3000	7/42/0.01	11/45/0.01	10/30/0.01
	5000	7/42/0.01	11/45/0.01	11/33/0.01
	7000	7/42/0.01	11/45/0.01	11/33/0.02
	10000	7/42/0.01	11/45/0.02	11/33/0.02
x_0^3	1000	8/47/0.01	13/53/0.01	15/45/0.01
	3000	8/47/0.01	14/57/0.01	15/45/0.01
	5000	8/47/0.01	14/57/0.01	15/45/0.02
	7000	8/47/0.02	14/57/0.02	16/48/0.02
	10000	8/47/0.02	14/57/0.02	16/48/0.02
x_0^4	1000	8/46/0.01	15/61/0.01	15/45/0.01
	3000	9/52/0.01	15/61/0.01	16/48/0.01
	5000	9/52/0.01	15/61/0.01	16/48/0.02
	7000	9/52/0.02	15/61/0.02	16/48/0.02
	10000	9/52/0.01	15/61/0.02	16/48/0.02
x_0^5	1000	9/51/0.01	14/56/0.01	16/47/0.01
	3000	9/51/0.01	14/56/0.01	15/45/0.01
	5000	10/57/0.01	14/56/0.01	16/48/0.02
	7000	10/57/0.02	15/60/0.02	19/58/0.02
	10000	10/57/0.02	15/60/0.02	20/62/0.03

method is the most effective method for the most cases.

Acknowledgements: The author wishes to thank the editor and anonymous referees for providing their valuable suggestions which have significantly improved the quality of the paper.

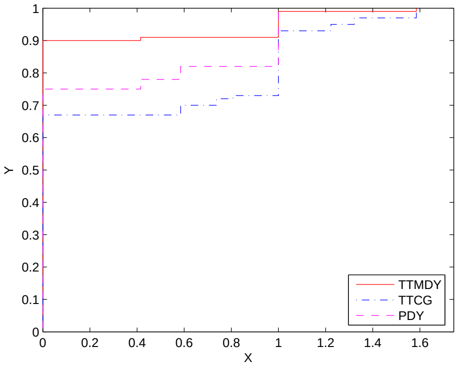

Fig. 3 Performance profiles with respect to CPU time.

REFERENCES

1. Meintjes K, Morgan AP (1987) A methodology for solving chemical equilibrium systems. *Appl Math Comput* **22**, 333–361.
2. Dirkse SP, Ferris MC (1995) MCPLIB: A collection of nonlinear mixed complementarity problems. *Optim Methods Softw* **5**, 319–345.
3. Wood AJ, Wollenberg BF (1996) *Power Generations, Operations, and Control*, Wiley, New York.
4. Hohammad H, Abubakar AB (2020) A descent derivative-free algorithm for nonlinear monotone equations with convex constraints. *RAIRO Oper Res* **54**, 489–505.
5. Liu JK, Feng YM (2019) A derivative-free iterative method for nonlinear monotone equations with convex constraints. *Numer Algorithms* **82**, 245–262.
6. Abubakar AB, Kumam P, Mohammad H, Awwal AM, Sitthithakerngkiet K (2019) A modified Fletcher-Reeves conjugate gradient method for monotone nonlinear equations with some applications. *Mathematics* **7**, ID 745.
7. Abubakar AB, Kumam P, Mohammad H, Awwal AM (2019) An efficient conjugate gradient method for convex constrained monotone nonlinear equations with applications. *Mathematics* **7**, ID 767.
8. Abubakar AB, Kumam P, Mohammad H (2020) A note on the spectral gradient projection method for nonlinear monotone equations with applications. *Comput Appl Math* **39**, ID 129.
9. Awwal AM, Kumam P, Mohammad H, Watthayu W, Abubakar AB (2020) A Perry-type derivative-free algorithm for solving nonlinear system of equations and minimizing ℓ_1 regularized problem. *Optimization* **4**, 1–29.
10. Liu JK, Xu JL, Zhang LQ (2019) Partially symmetrical derivative-free Liu-Storey projection method for convex constrained equations. *Int J Comput Math* **96**, 1787–1798.
11. Abubakar AB, Kumam P, Mohammad H, Awwal AM (2020) A Barzilai-Borwein gradient projection method for sparse signal and blurred image restoration. *J Franklin Inst* **54**, 489–505.
12. Dai ZF, Zhu H (2020) A modified Hestenes-Stiefel-type derivative-free method for large-scale nonlinear monotone equations. *Mathematics* **8**, ID 168.
13. Liu JK, Du XL (2018) A gradient projection method for the sparse signal reconstruction in compressive sensing. *Appl Anal* **97**, 2122–2131.
14. Dai ZF, Zhu H, Kang J (2020) New technical indicators and stock returns predictability. *Int Rev Econ Finance* **71**, 127–142.
15. Zhang L, Zhou WJ, Li DH (2006) A descent modified Polak-Ribére-Polyak conjugate gradient method and its global convergence. *IMA J Numer Anal* **26**, 629–640.
16. Zhang L, Zhou WJ, Li DH (2007) Some descent three-term conjugate gradient methods and their global convergece. *Optim Methods Softw* **22**, 697–711.
17. Narushima Y, Yabe H, Ford JA (2011) A three-term conjugate gradient method with sufficient descent property for unconstrained optimization. *SIAM Journal on Optimization* **21**, 212–230.
18. Andrei N (2013) On three-term conjugate gradient algorithms for unconstrained optimization. *Appl Math Comput* **219**, 6316–6327.
19. Andrei N (2013) A simple three-term conjugate gradient algorithm for unconstrained optimization. *J Comput Appl Math* **241**, 19–29.
20. Andrei N (2015) A new three-term conjugate gradient algorithm for unconstrained optimization. *Numer Algorithms* **68**, 305–321.
21. Liu JK, Feng YM, Zou LM (2018) Some three-term conjugate gradient methods with the inexact line search condition. *Calcolo* **55**, ID 16.
22. Solodov MV, Svaiter BF (1998) 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*, Springer, MA, pp 355–369.
23. Li DH, Wang XL (2011) A modified Fletcher-Reeves-type derivative-free method for symmetric nonlinear equations. *Numer Algebra Control Optim* **1**, 71–82.
24. Zhang L, Zhou WJ, Li DH (2006) Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search. *Numer Math* **104**, 561–572.
25. Gao PT, He CJ (2018) An efficient three-term conjugate gradient method for nonlinear monotone equations with convex constraints. *Calcolo* **55**, ID 53.
26. Zheng Y, Zheng B (2017) Two new Dai-Liao-type conjugate gradient methods for unconstrained optimization problems. *J Optim Theory Appl* **175**, 502–509.
27. Koorapetse M, Kaelo P (2019) A new three-term conjugate gradient-based projection method for solving large-scale nonlinear monotone equations. *Math Model Anal* **24**, 550–563.
28. Dai YH, Yuan YX (1999) A nonlinear conjugate gradient with a strong global convergence property. *SIAM J Optim* **10**, 177–182.
29. Liu JK, Li SJ (2017) Multivariate spectral DY-type projection method for convex constrained nonlinear monotone equation. *J Ind Manag Optim* **13**, 283–295.
30. Bing Y, Lin G (1991) An efficient implementation of Merrill's method for sparse or partially separable systems of nonlinear equations. *SIAM J Optim* **2**, 206–221.
31. Wang CW, Wang YJ, Xu CL (2007) A projection method for a system of nonlinear equations with convex constraints. *Math Methods Oper Res* **66**, 33–46.
32. Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. *Math Program* **91**, 201–213.