

RESEARCH

Open Access



# A class of new derivative-free gradient type methods for large-scale nonlinear systems of monotone equations

Xiaowei Fang<sup>1\*</sup>

\*Correspondence:  
fangxiaowei@163.com

<sup>1</sup>Department of Mathematics,  
Huzhou University, Huzhou, China

## Abstract

In this paper, we present a class of new derivative-free gradient type methods for large-scale nonlinear systems of monotone equations. The methods combine the RMIL conjugate gradient method, the strategy of hyperplane projection and the derivative-free line search technique. Under some appropriate assumptions, the global convergence of the given methods is established. Numerical experiments indicate that the proposed methods are effective.

**Keywords:** Large-scale nonlinear systems of monotone equations; Hyperplane projection method; Gradient type method; Derivative-free line search

## 1 Introduction

In this paper, we consider the following nonlinear systems of monotone equations:

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

where  $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$  is monotone and continuous, which means  $F$  satisfies  $(F(x) - F(y))^T \times (x - y) \geq 0$  for all  $x, y \in \mathbb{R}^n$ . Nonlinear systems of monotone equations are widely used in economy, finance, engineering, industry and many other fields, so there are numerous iterative algorithms for solving (1).

Recently, La Cruz [3] presented a spectral method that uses the residual vector as search direction for solving large-scale systems of nonlinear monotone equations. Solodov and Svaiter [12] proposed a method which combined projection, proximal point and Newton method. According to the work of Solodov and Svaiter [12], Zhang and Zhou [17] developed a spectral gradient projection method for solving nonlinear monotone equations.

In particular, the conjugate gradient methods are widely used methods for solving large-scale nonlinear equations because of the low memory and simplicity [8, 10]. In the last few years, some authors proposed a series of methods for solving nonlinear monotone equations, which combined conjugate gradient methods and projection method [12]. For instance, Cheng [2] extended the PRP conjugate method to monotone equations. Yan *et al.* [13] proposed two modified HS conjugate method. Li and Li [7] designed a class of

© 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, unless indicated otherwise in a credit line to the material. If material is not 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/>.

derivative-free methods based on line search technique. Ahookhosh *et al.* [1] developed two derivative-free conjugate gradient procedures. Papp and Rapajić [9] described some new FR type methods. Yuan *et al.* [14, 15] proposed new three-terms conjugate gradient methods. Dai *et al.* [4] gave a modified Perrys conjugate gradient method. Zhou *et al.* [20, 21] developed a class of methods. Zhang [16] developed a residual method-based secant condition.

For unconstrained optimization problem, Rivaie *et al.* [11] designed a RMIL conjugate gradient method. Fang and Ni [6] extended the RMIL method to solve large-scale nonlinear systems of equations with the ideas of nonmonotone line search. Numerical experiments show that the RMIL method is practically effective.

For systems of monotone equations, we describe a class of new derivative-free gradient type method, which is inspired by the efficiency of the RMIL method [11], and the strategy of projection method [12].

This paper is organized as follows. In Sect. 2, we propose the algorithm. In Sect. 3, we establish the global convergence. In Sect. 4, numerical results show the efficiency of the proposed methods. In Sect. 5, we give some conclusions. We denote by  $\| \cdot \|$  the Euclidean norm.

### 2 Algorithm

In this section, we first consider the conjugate gradient method for the following unconstrained optimization problem:

$$\min f(x), \tag{2}$$

where  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  is smooth.

Quite recently, Rivaie *et al.* [11] developed RMIL conjugate gradient method, and the search direction  $d_k$  is given by

$$d_k = \begin{cases} -g_k & \text{if } k = 0, \\ -g_k + \frac{g_k^T(g_k - g_{k-1})}{\|d_{k-1}\|^2} d_{k-1} & \text{if } k \geq 1, \end{cases} \tag{3}$$

where  $g_k$  is the gradient of  $f$ .

Numerical results show that RMIL conjugate gradient method is superior and more efficient than other conjugate gradient method.

We focus on the method for solving monotone equations (1). We have the projection procedure in [12], by performing some line search techniques to find a point

$$z_k = x_k + \alpha_k d_k, \tag{4}$$

such that

$$(F(z_k), x_k - z_k) > 0. \tag{5}$$

On the other hand, for any  $x^*$  such that  $F(x^*) = 0$ , by the monotonicity of  $F$ , we obtain

$$(F(z_k), x^* - z_k) = -(F(x^*) - F(z_k), x^* - z_k) \leq 0. \tag{6}$$

Equations (5) and (6) imply that the hyperplane

$$H_k = \{x \in \mathbb{R}^n \mid (F(z_k), x - z_k) = 0\} \tag{7}$$

strictly separates the zeros of systems of monotone equations (1) from  $x_k$ . Therefore, Solodov and Svaiter [12] could compute the next iterate  $x_{k+1}$  by projecting  $x_k$  onto the hyperplane  $H_k$ . Specifically,  $x_{k+1}$  is obtained by

$$x_{k+1} = x_k - \frac{F(z_k)^T(x_k - z_k)}{\|F(z_k)\|^2} F(z_k). \tag{8}$$

The steplength  $\alpha_k$  of (4) is determined by a proper line search technique. Recently, Zhang and Zhou[17], Zhou[18] presented the following derivative-free line search rule:

$$-F(x_k + \alpha_k d_k)^T d_k \geq \sigma \alpha_k \|F(x_k + \alpha_k d_k)\| \|d_k\|^2, \tag{9}$$

where  $\alpha_k = \max\{s, \rho s, \rho^2 s, \dots\}$ ,  $d_k$  is the search direction,  $\sigma, s, \rho$  are constants, and  $\sigma > 0, s > 0, 1 > \rho > 0$ .

Now, we extend RMIL conjugate gradient method [11] for solving nonlinear systems of monotone equations, which combined the projection method [12] and derivative-free line search technique [17, 18]. The steps of our algorithm are listed as follows.

**Algorithm 1** (MRMIL) *Step 0:* Choose an initial point  $x_0 \in \mathbb{R}^n$ . Let  $\delta > 0, \sigma > 0, s > 0, \epsilon > 0, 1 > \rho > 0, k_{\max} > 0$ . Set  $k = 0$ .

*Step 1:* Choose the search direction  $d_k$  that satisfies the following sufficient descent condition:

$$F_k^T d_k \leq -\delta \|F_k\|^2, \tag{10}$$

and determine the initial steplength  $\alpha = s$ .

*Step 2:* If

$$-F(x_k + \alpha d_k)^T d_k \geq \sigma \alpha \|F(x_k + \alpha d_k)\| \|d_k\|^2, \tag{11}$$

then set  $\alpha_k = \alpha, z_k = x_k + \alpha_k d_k$  and go to step 3.

Else set  $\alpha_k = \rho \alpha_k$ , and go to step 2.

*Step 3:* If  $\|F(z_k)\| > \epsilon$ , then compute

$$x_{k+1} = x_k - \frac{F(z_k)^T(x_k - z_k)}{\|F(z_k)\|^2} F(z_k) \tag{12}$$

and go to step 4, otherwise stop.

*Step 4:* If  $\|F(x_{k+1})\| > \epsilon$  and  $k < k_{\max}$ , then set  $k = k + 1$  and go to step 1, otherwise stop.

Let  $y_{k-1} = F_k - F_{k-1}, \beta_k = \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2}, 0 < \gamma < 1$ . Now, based on the direction  $d_k$  of RMIL conjugate gradient algorithm for unconstrained optimization, we are going to construct three directions  $d_k$  that satisfy the sufficient descent condition(10).

*MRMIL1 direction:*

$$d_k = \begin{cases} -F_k & \text{if } k = 0, \\ -\theta_k F_k + \beta_k d_{k-1} & \text{if } k \geq 1, \end{cases} \tag{13}$$

where  $\theta_k = \frac{(F_k^T y_{k-1})^2}{4\gamma \|F_k\|^2 \|d_{k-1}\|^2} + 1$ ,  $0 < \gamma < 1$ . We set  $u = \sqrt{2\gamma} \|d_{k-1}\|^2 F_k$ ,  $v = \frac{1}{\sqrt{2\gamma}} (F_k^T y_{k-1}) d_{k-1}$ , and use  $u^T v \leq \frac{1}{2}(\|u\|^2 + \|v\|^2)$ , then, for  $k \in \mathbb{N}$ , we have

$$\begin{aligned} F_k^T d_k &= -\theta_k \|F_k\|^2 + \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2} F_k^T d_{k-1} \\ &= -\|F_k\|^2 + \frac{F_k^T y_{k-1} \|d_{k-1}\|^2 F_k^T d_{k-1} - \frac{1}{4\gamma} (F_k^T y_{k-1})^2 \|d_{k-1}\|^2}{\|d_{k-1}\|^4} \\ &\leq (\gamma - 1) \|F_k\|^2. \end{aligned} \tag{14}$$

The MRMIL1 method is the Algorithm 1 with MRMIL1 direction which is defined by (13).

*MRMIL2 direction:*

$$d_k = \begin{cases} -F_k & \text{if } k = 0, \\ -\theta_k F_k + \beta_k d_{k-1} & \text{if } k \geq 1, \end{cases} \tag{15}$$

where  $\theta_k = \frac{(F_k^T d_{k-1})^2 \|y_{k-1}\|^2}{4\gamma \|F_k\|^2 \|d_{k-1}\|^4} + 1$ ,  $0 < \gamma < 1$ . We set  $u = \sqrt{2\gamma} \|d_{k-1}\|^2 F_k$ ,  $v = \frac{1}{\sqrt{2\gamma}} (F_k^T d_{k-1}) y_{k-1}$ , and use  $u^T v \leq \frac{1}{2}(\|u\|^2 + \|v\|^2)$ , then, for  $k \in \mathbb{N}$ , we have

$$\begin{aligned} F_k^T d_k &= -\theta_k \|F_k\|^2 + \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2} F_k^T d_{k-1} \\ &= -\|F_k\|^2 + \frac{F_k^T y_{k-1} \|d_{k-1}\|^2 F_k^T d_{k-1} - \frac{1}{4\gamma} (F_k^T d_{k-1})^2 \|y_{k-1}\|^2}{\|d_{k-1}\|^4} \\ &\leq (\gamma - 1) \|F_k\|^2. \end{aligned} \tag{16}$$

The MRMIL2 method is Algorithm 1 with the MRMIL2 direction which is defined by (15).

*MRMIL3 direction:*

$$d_k = \begin{cases} -F_k & \text{if } k = 0, \\ -F_k + \beta_k d_{k-1} - \theta_k y_{k-1} & \text{if } k \geq 1, \end{cases} \tag{17}$$

where  $\theta_k = \frac{F_k^T y_{k-1}}{4\gamma \|d_{k-1}\|^2}$ ,  $0 < \gamma < 1$ . We set  $u = \sqrt{2\gamma} \|d_{k-1}\|^2 F_k$ ,  $v = \frac{1}{\sqrt{2\gamma}} (F_k^T y_{k-1}) d_{k-1}$ , and use  $u^T v \leq \frac{1}{2}(\|u\|^2 + \|v\|^2)$ , then, for  $k \in \mathbb{N}$ , we have

$$\begin{aligned} F_k^T d_k &= -\|F_k\|^2 + \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2} F_k^T d_{k-1} - \theta_k F_k^T y_{k-1} \\ &= -\|F_k\|^2 + \frac{F_k^T y_{k-1} \|d_{k-1}\|^2 F_k^T d_{k-1} - \frac{1}{4\gamma} (F_k^T y_{k-1})^2 \|d_{k-1}\|^2}{\|d_{k-1}\|^4} \\ &\leq (\gamma - 1) \|F_k\|^2. \end{aligned} \tag{18}$$

The MRMIL3 method is Algorithm 1 with the MRMIL3 direction which is defined by (17).

Using (13), (15) and (17), we get

$$\|F_0^T d_0\| = -\|F_0\|^2. \tag{19}$$

From (14), (16), (18) and (19), it is not difficult to show that the directions  $d_k$  defined by the MRMIL1, MRMIL2 and MRMIL3 directions satisfy the sufficient descent condition

$$F_k^T d_k \leq -\delta \|F_k\|^2, \quad \forall k \in \mathbb{N} \cup 0, \tag{20}$$

if we let  $\delta = 1 - \gamma$  and  $0 < \gamma < 1$ .

### 3 Convergence analysis

In this section, so as to obtain the global convergence of MRMIL1, MRMIL2 and MRMIL3 method, we give the following assumptions.

#### Assumption 3.1

- (1) The solution set of the systems of monotone equations  $F(x) = 0$  is nonempty.
- (2)  $F(x)$  is Lipschitz continuous on  $\mathbb{R}^n$ , namely

$$\|F(x) - F(y)\| \leq L \|x - y\|, \quad \forall x, y \in \mathbb{R}^n, \tag{21}$$

where  $L$  is a positive constant.

Assumption 3.1 implies that

$$\|F(x)\| \leq \kappa, \quad \forall x \in \mathbb{R}^n, \tag{22}$$

where  $\kappa$  is a positive constant.

Now, we get Lemma 3.1 whose proof is similar to those in [12] and is omitted.

**Lemma 3.1** *Suppose Assumption 3.1 is satisfied and the sequence  $\{x_k\}$  is generated by the Algorithm 1. For any  $x^*$  such that  $F(x^*) = 0$ , we have*

$$\|x_{k+1} - x^*\|^2 + \|x_{k+1} - x_k\|^2 \leq \|x_k - x^*\|^2.$$

*In addition, the sequence  $\{x_k\}$  satisfies*

$$\lim_{k \rightarrow \infty} \|x_{k+1} - x_k\| = 0. \tag{23}$$

**Lemma 3.2** *Suppose Assumption 3.1 is satisfied and the sequences  $\{x_k, d_k\}$  are generated by the Algorithm 1 with the MRMIL1, MRMIL2 or MRMIL3 direction. Then we have*

$$\|d_k\| \leq \kappa \left( 1 + Ls + \frac{(Ls)^2}{4\gamma} \right), \tag{24}$$

where  $\kappa, \gamma, s, L$  are constants, and  $\kappa > 0, 1 > \gamma > 0, s > 0, L > 0$ .

*Proof* From (4) and the step 3 of Algorithm 1, we have

$$\|x_{k+1} - x_k\| = \frac{\|F(z_k)^T(x_k - z_k)F(z_k)\|}{\|F(z_k)\|^2} \leq \|x_k - z_k\| = \alpha_k \|d_k\|. \tag{25}$$

By step 1 and step 2 of Algorithm 1, we get

$$\alpha_k \leq s, \quad \forall k \in \mathbb{N} \cup 0. \tag{26}$$

□

For  $k \in \mathbb{N}$ , the boundedness of  $d_k$  can be divided into three cases.

Case 1 (MRMIL1 direction): The MRMIL1 direction is defined by (13). Using (21), (22), (25) and (26), we have

$$\begin{aligned} \|d_k\| &= \left\| -\left(\frac{(F_k^T y_{k-1})^2}{4\gamma \|F_k\|^2 \|d_{k-1}\|^2} + 1\right)F_k + \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2} d_{k-1} \right\| \\ &\leq \|F_k\| \left(1 + \frac{\|y_{k-1}\|^2}{4\gamma \|d_{k-1}\|^2} + \frac{\|y_{k-1}\|}{\|d_{k-1}\|}\right) \\ &\leq \|F_k\| \left(1 + \frac{(L\alpha_{k-1})^2}{4\gamma} + L\alpha_{k-1}\right) \\ &\leq \kappa \left(1 + Ls + \frac{(Ls)^2}{4\gamma}\right). \end{aligned} \tag{27}$$

Case 2 (MRMIL2 direction): Analogously, the MRMIL2 direction is defined by (15). By (21), (22), (25) and (26), we get

$$\begin{aligned} \|d_k\| &= \left\| -\left(\frac{(F_k^T d_{k-1})^2 \|y_{k-1}\|^2}{4\gamma \|F_k\|^2 \|d_{k-1}\|^4} + 1\right)F_k + \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2} d_{k-1} \right\| \\ &\leq \|F_k\| \left(1 + \frac{\|y_{k-1}\|^2}{4\gamma \|d_{k-1}\|^2} + \frac{\|y_{k-1}\|}{\|d_{k-1}\|}\right) \\ &\leq \kappa \left(1 + Ls + \frac{(Ls)^2}{4\gamma}\right). \end{aligned} \tag{28}$$

Case 3 (MRMIL3 direction): The definition of MRMIL3 direction given by (17). (21), (22), (25) and (26) implies

$$\begin{aligned} \|d_k\| &= \left\| -F_k + \frac{F_k^T y_{k-1}}{\|d_{k-1}\|^2} d_{k-1} - \frac{F_k^T y_{k-1}}{4\gamma \|d_{k-1}\|^2} y_{k-1} \right\| \\ &\leq \|F_k\| \left(1 + \frac{\|y_{k-1}\|}{\|d_{k-1}\|} + \frac{\|y_{k-1}\|^2}{4\gamma \|d_{k-1}\|^2}\right) \\ &\leq \kappa \left(1 + Ls + \frac{(Ls)^2}{4\gamma}\right). \end{aligned} \tag{29}$$

From (13), (15), (17) and (22), we get

$$\|d_0\| = \|-F_0\| \leq \kappa. \tag{30}$$

Combining with (27), (28) and (29), we find (24).

**Lemma 3.3** *Suppose Assumption 3.1 is satisfied and the sequence  $\{x_k, \alpha_k, d_k, F_k\}$  are generated by the Algorithm 1 with the MRMIL1, MRMIL2 or MRMIL3 direction. If there exists a constant  $\epsilon > 0$ , such that  $\|F_k\| \geq \epsilon$  for all  $k \in \mathbb{N} \cup 0$ , then we have*

$$\alpha_k \geq \min \left\{ s, \frac{(1 - \gamma)\epsilon^2}{\rho^{-1}\kappa^2(L + \sigma\kappa\rho^{-1}(\rho + Ls + (Ls)^2 + \frac{(Ls)^3}{4\gamma}))(1 + Ls + \frac{(Ls)^2}{4\gamma})^2} \right\}. \tag{31}$$

*Proof* If  $\alpha_k \neq s$ , by the step 2 of Algorithm 1, we know that  $\rho^{-1}\alpha_k$  does not satisfy (11). Then we have

$$-F(x_k + \rho^{-1}\alpha_k d_k)^T d_k < \sigma\rho^{-1}\alpha_k \|F(x_k + \rho^{-1}\alpha_k d_k)\| \|d_k\|^2. \tag{32}$$

Combining with Assumption 3.1, (22) and (24), we have

$$\begin{aligned} \|F(x_k + \rho^{-1}\alpha_k d_k)\| &= \|F(x_k + \rho^{-1}\alpha_k d_k) - F(x_k)\| + \|F(x_k)\| \\ &\leq L\rho^{-1}\alpha_k \|d_k\| + \kappa \\ &\leq \kappa\rho^{-1} \left( \rho + Ls + (Ls)^2 + \frac{(Ls)^3}{4\gamma} \right). \end{aligned} \tag{33}$$

It follows from (14), (16), (18), (21), (24), (32) and (33) that

$$\begin{aligned} (1 - \gamma)\|F_k\| &\leq -F_k^T d_k \\ &= [F(x_k + \rho^{-1}\alpha_k d_k) - F(x_k)]^T d_k - [F(x_k + \rho^{-1}\alpha_k d_k)]^T d_k \\ &< [L + \sigma \|F(x_k + \rho^{-1}\alpha_k d_k)\|] \rho^{-1}\alpha_k \|d_k\|^2 \\ &\leq \rho^{-1}\alpha_k \kappa^2 \left( L + \sigma\kappa\rho^{-1} \left( \rho + Ls + (Ls)^2 + \frac{(Ls)^3}{4\gamma} \right) \right) \left( 1 + Ls + \frac{(Ls)^2}{4\gamma} \right)^2. \end{aligned} \tag{34}$$

Then we have

$$\begin{aligned} \alpha_k &> \frac{(1 - \gamma)\|F_k\|}{\rho^{-1}\kappa^2(L + \sigma\kappa\rho^{-1}(\rho + Ls + (Ls)^2 + \frac{(Ls)^3}{4\gamma}))(1 + Ls + \frac{(Ls)^2}{4\gamma})^2} \\ &> \frac{(1 - \gamma)\epsilon^2}{\rho^{-1}\kappa^2(L + \sigma\kappa\rho^{-1}(\rho + Ls + (Ls)^2 + \frac{(Ls)^3}{4\gamma}))(1 + Ls + \frac{(Ls)^2}{4\gamma})^2}. \end{aligned} \tag{35}$$

This implies (31). □

**Theorem 3.4** *Suppose Assumption 3.1 is satisfied and the sequences  $\{x_k, \alpha_k, d_k, F_k\}$  are generated by the Algorithm 1 with the MRMIL1, MRMIL2 or MRMIL3 direction. Then we have*

$$\liminf_{k \rightarrow \infty} \|F_k\| = 0. \tag{36}$$

*In particular, the sequence  $\{x_k\}$  converges to  $x^*$  and  $F(x^*) = 0$ .*

*Proof* If (36) does not hold, then there exists a constant  $\epsilon > 0$  such that

$$\|F_k\| \geq \epsilon, \quad \forall k \geq 0. \tag{37}$$

From (14), (16) and (18), we get

$$\|F_k\| \|d_k\| \geq \|F_k d_k\| \geq (1 - \gamma) \|F_k\|^2. \tag{38}$$

Using (37) and (38), we have

$$\|d_k\| \geq (1 - \gamma) \|F_k\| \geq (1 - \gamma)\epsilon. \tag{39}$$

Suppose  $\tilde{x}$  is an arbitrary accumulation point of  $\{x_k\}$  and  $K_1$  is an infinite index set such that

$$\lim_{k \in K_1, k \rightarrow \infty} x_k = \tilde{x}. \tag{40}$$

From (23), (25) and (40), we get

$$\lim_{k \in K_1, k \rightarrow \infty} \alpha_k \|d_k\| = 0. \tag{41}$$

On the other hand, together with the conclusion of Lemma 3.3 and (39), we obtain

$$\alpha_k \|d_k\| \geq \min \left\{ (1 - \gamma)\epsilon s, \frac{(1 - \gamma)^2 \epsilon^3}{\rho^{-1} \kappa^2 (L + \sigma \kappa \rho^{-1} (\rho + Ls + (Ls)^2 + \frac{(Ls)^3}{4\gamma})) (1 + Ls + \frac{(Ls)^2}{4\gamma})^2} \right\} > 0. \tag{42}$$

Equations (41) and (42) are a contradiction, then the conclusion (36) is hold. From Assumption 3.1, Lemma 3.1 and (36), we see that the sequence  $\{x_k\}$  converges to some accumulation point  $x^*$  such that  $F(x^*) = 0$ . □

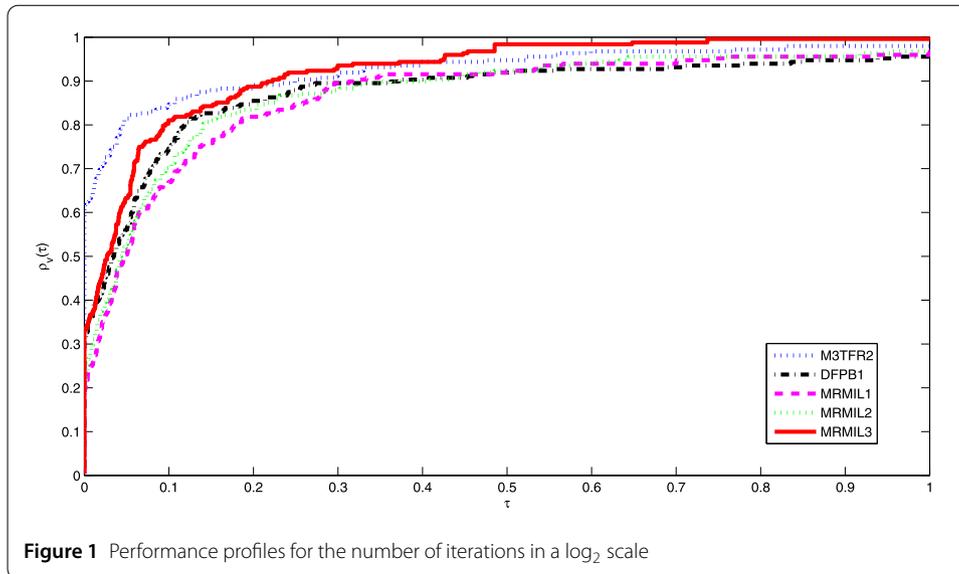
#### 4 Numerical experiments

In this section, we report some numerical test results for the MRMIL1 method, the MRMIL2 method, and the MRMIL3 method and compare with the DFPB1 method in [1] and the M3TFR2 method in [9]. Our tests are implemented in Matlab R2011a, run on a personal computer with 8 GB RAM and Intel CPU I5-3470.

In order to compare all methods, we employ the performance profiles [5], which are defined by the following fraction:

$$\rho_v(\tau) = \frac{1}{|P|} \left| \left\{ p \in P : \log_2 \left( \frac{t_{p,v}}{\min\{t_{p,v} : v \in V\}} \right) \leq \tau \right\} \right|,$$

where  $P$  is the test set,  $|P|$  is the number of problems in the test set  $P$ ,  $V$  is the set of optimization solvers, and  $t_{p,v}$  is the CPU time (or the number of the function evaluations, or the number of iterations) for  $p \in P$  and  $v \in V$ .



We test the following problems with different starting points and various sizes: (1) the problems 1–7 from [2, 7, 13, 19] with sizes 1000, 5000, 10,000, 50,000; (2) the problem 8 from [7] with sizes 10,000, 20,164, 40,000;

All problems are initialized with the following eight starting points:  $x_0^1 = 10 \cdot (1, 1, \dots, 1)^T$ ,  $x_0^2 = -10 \cdot (1, 1, \dots, 1)^T$ ,  $x_0^3 = (1, 1, \dots, 1)^T$ ,  $x_0^4 = -(1, 1, \dots, 1)^T$ ,  $x_0^5 = 0.1 \cdot (1, 1, \dots, 1)^T$ ,  $x_0^6 = (1, \frac{1}{2}, \dots, \frac{1}{n})^T$ ,  $x_0^7 = (\frac{1}{n}, \frac{2}{n}, \dots, 1)^T$ ,  $x_0^8 = (\frac{n-1}{n}, \frac{n-2}{n}, \dots, 0)^T$ .

For all methods, the stopping criteria are (1)  $\|F(x_k)\| \leq \epsilon$  or (2)  $\|F(z_k)\| \leq \epsilon$  or (3) the number of iterations exceeds  $k_{\max}$ , where  $\epsilon = 10^{-4}$ ,  $k_{\max} = 10^5$ . Similar to [1, 9], we used the same parameters for five methods: initial steplength  $s = \left\| \frac{F_k^T d_k}{(F(x_k + 10^{-8} d_k) - F_k)^T d_k / 10^{-8}} \right\|$ ,  $\rho = 0.7$ ,  $\sigma = 0.3$ ,  $\gamma = \frac{1}{4}$ .

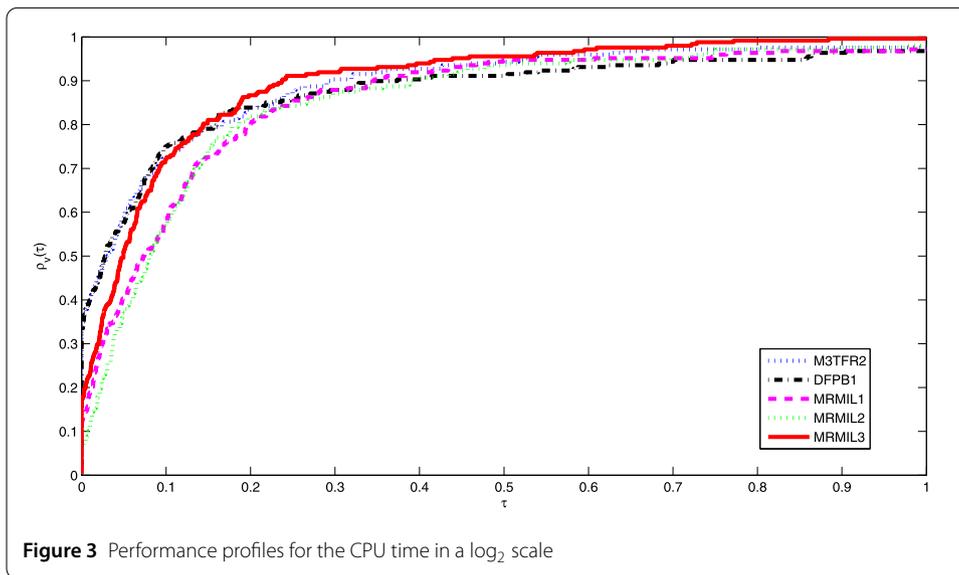
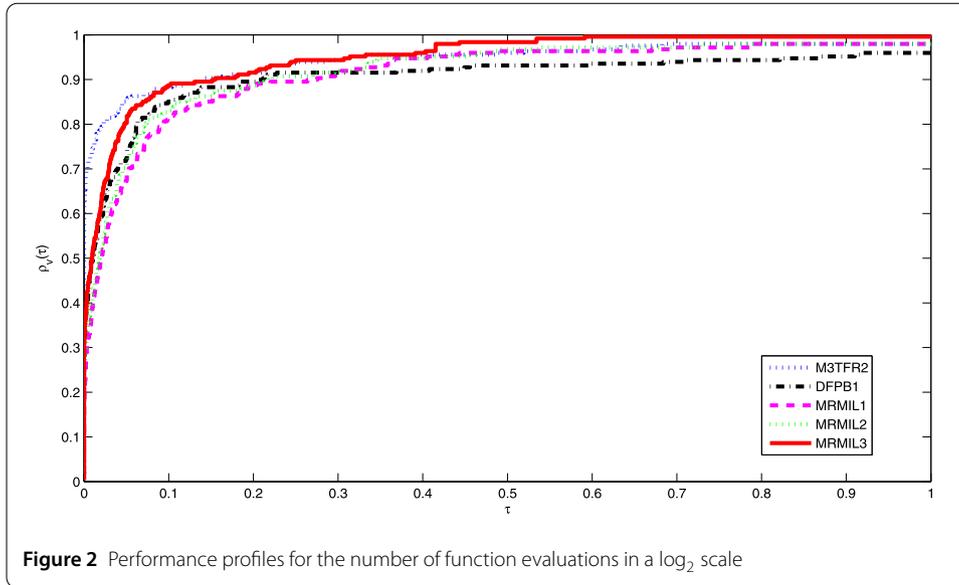
Figure 1 is for the iterations performance profiles related to five methods. As we can see, MRMIL3 guarantees better results than M3TFR2, DFPB1, MRMIL1 and MRMIL2 as it solves a higher percentage of problems when  $\tau \geq 0.2$ . It also can be seen that DFPB1, MRMIL1 and MRMIL2 have similar performances, especially when  $\tau \geq 0.3$ . Furthermore, M3TFR2 gives better results than DFPB1, MRMIL1 and MRMIL2, and the difference is significantly small as the performance ratio  $\tau$  increases.

The number of function evaluations performance profiles are reported in Figure 2. we note that MRMIL3 performs better than the other four methods when  $\tau \geq 0.2$ . In addition, we also observe that MRMIL1 and MRMIL2 is more efficient than DFPB1 when  $\tau \geq 0.3$ , but more inefficient than M3TFR2.

Figure 3 shows the CPU time performance profiles. When  $\tau \leq 0.15$ , M3TFR2 and DFPB1 uses the shortest CPU time, but MRMIL3 gives the best result when  $\tau \geq 0.15$ .

### 5 Conclusion

In this paper, we give a class of new derivative-free gradient type methods for large-scale nonlinear systems of monotone equations. Under mild assumptions, we prove that the methods possess global convergence properties. Numerical experiments show that the proposed methods are promising, especially the MRMIL3 method, which is the most efficient one.



**Appendix: Test problems**

We now list the following test problems.

Problem 1. ([7], p. 1630) The function  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ , where

$$\begin{aligned}
 f_1(x) &= 2x_1 + \sin(x_1) - 1, \\
 f_i(x) &= -2x_{i-1} + 2x_i + \sin(x_i) - 1, \quad 2 \leq i \leq n - 1, \\
 f_n(x) &= 2x_n + \sin(x_n) - 1.
 \end{aligned}$$

Problem 2. ([7], p. 1630) The function  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ , where

$$f_i(x) = 2x_i - \sin(x_i), \quad 1 \leq i \leq n.$$

Problem 3. ([7], p. 1631) The function  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ , where

$$f_i(x) = 2x_i - \sin |x_i|, \quad 1 \leq i \leq n.$$

Problem 4. ([13], p. 654) The function  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ , where

$$\begin{aligned} f_1(x) &= \frac{1}{3}x_1^3 + \frac{1}{2}x_2^2, \\ f_i(x) &= -\frac{1}{2}x_i^2 + \frac{i}{3}x_i^3 + \frac{1}{2}x_{i+1}^2, \quad 2 \leq i \leq n-1, \\ f_n(x) &= -\frac{1}{2}x_n^2 + \frac{n}{3}x_n^3. \end{aligned}$$

Problem 5. ([13], p. 654) The function  $F(x) = (f_1(x), f_2(x), \dots, f_n(x))^T$ , where

$$\begin{aligned} f_1(x) &= x_1 - \exp\left(\cos\left(\frac{x_1 + x_2}{n+1}\right)\right), \\ f_i(x) &= x_i - \exp\left(\cos\left(\frac{x_{i-1} + x_i + x_{i+1}}{n+1}\right)\right), \quad 2 \leq i \leq n-1, \\ f_n(x) &= x_n - \exp\left(\cos\left(\frac{x_{n-1} + x_n}{n+1}\right)\right). \end{aligned}$$

Problem 6. ([19], p. 2236) The function  $F(x) = Ax + b(x) - e$ , where  $b(x) = (e^{x_1}, e^{x_2}, \dots, e^{x_n})^T$  and  $e = (1, 1, \dots, 1)^T$  are the  $n \times 1$  vectors,

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 2 \end{pmatrix}$$

is the  $n \times n$  matrix.

Problem 7. ([2], p. 19) The function  $F(x) = Ax - e$ , where  $e = (1, 1, \dots, 1)^T$  is the  $n \times 1$  vector, and

$$A = \begin{pmatrix} \frac{5}{2} & 1 & & & \\ 1 & \frac{5}{2} & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 1 \\ & & & 1 & \frac{5}{2} \end{pmatrix}$$

is the  $n \times n$  matrix.

Problem 8. ([7], p. 1631) The function  $F(x) = Ax + h^2y - 10h^2e$ , where  $h = \frac{1}{r+1}$ ,  $n = r^2$ ,  $x = (x_1, x_2, \dots, x_n)^T$ ,  $y = (x_1^3, x_2^3, \dots, x_n^3)^T$  and  $e = (1, 1, \dots, 1)^T$  are the  $n \times 1$

vectors.  $I$  is the  $m \times m$  identity matrix,

$$B = \begin{pmatrix} 4 & -1 & & & \\ -1 & 4 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 4 \end{pmatrix}$$

is the  $r \times r$  matrix, and

$$A = \begin{pmatrix} B & -I & & & \\ -I & B & -I & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -I \\ & & & -I & B \end{pmatrix}$$

is the  $n \times n$  matrix.

**Acknowledgements**

The author would like to express his sincere thanks to the editors and the referees for their valuable suggestions.

**Funding**

This research was funded by the National Natural Science Foundation of China (11871211) and the Natural Science Foundation of Huzhou University (2018XJKJ46).

**Availability of data and materials**

Not applicable.

**Competing interests**

The author declares to have no competing interests.

**Authors' contributions**

The author carried out the results, and read and approved the current version of the manuscript.

**Publisher's Note**

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 5 February 2020 Accepted: 27 March 2020 Published online: 06 April 2020

**References**

1. Ahookhosh, M., Amini, K., Bahrami, S.: Two derivative-free projection approaches for systems of large-scale nonlinear monotone equations. *Numer. Algorithms* **64**(1), 21–42 (2013)
2. Cheng, W.: A PRP type method for systems of monotone equations. *Math. Comput. Model.* **50**(1–2), 15–20 (2009)
3. Cruz, W.: A spectral algorithm for large-scale systems of nonlinear monotone equations. *Numer. Algorithms* **76**(4), 1109–1130 (2017)
4. Dai, Z., Chen, X., Wen, F.: A modified Perrys conjugate gradient method-based derivative-free method for solving large-scale nonlinear monotone equations. *Appl. Math. Comput.* **270**, 378–386 (2015)
5. Dolan, E., Moré, J.: Benchmarking optimization software with performance profiles. *Math. Program.* **91**(2), 201–213 (2001)
6. Fang, X., Ni, Q.: A new derivative-free conjugate gradient method for large-scale nonlinear systems of equations. *Bull. Aust. Math. Soc.* **95**(3), 500–511 (2017)
7. Li, Q., Li, D.: A class of derivative-free methods for large-scale nonlinear monotone equations. *IMA J. Numer. Anal.* **31**(4), 1625–1635 (2011)
8. Nocedal, J.: Conjugate gradient methods and nonlinear optimization. In: *Linear and Nonlinear Conjugate Gradient-Related Methods*, pp. 9–23. SIAM, Philadelphia (1996)
9. Papp, Z., Rapajić, S.: FR type methods for systems of large-scale nonlinear monotone equations. *Appl. Math. Comput.* **269**, 816–823 (2015)
10. Polak, E.: *Optimization: Algorithms and Consistent Approximations*. Applied Mathematical Sciences, vol. 124. Springer, Berlin (2012)

11. Rivaie, M., Mamat, M., June, L., Mohd, I.: A new class of nonlinear conjugate gradient coefficients with global convergence properties. *Appl. Math. Comput.* **218**(22), 11323–11332 (2012)
12. Solodov, M., Svaiter, B.F.: A globally convergent inexact Newton method for systems of monotone equations. In: Fukushima, M., Qim, L. (eds.) *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*, pp. 355–369. Kluwer Academic, Dordrecht (1998)
13. Yan, Q., Peng, X., Li, D.: A globally convergent derivative-free method for solving large-scale nonlinear monotone equations. *J. Comput. Appl. Math.* **234**(3), 649–657 (2010)
14. Yuan, G., Hu, W.: A conjugate gradient algorithm for large-scale unconstrained optimization problems and nonlinear equations. *J. Inequal. Appl.* **2018**, Article ID 113 (2018)
15. Yuan, G., Zhang, M.: A three-terms Polak–Ribière–Polyak conjugate gradient algorithm for large-scale nonlinear equations. *J. Comput. Appl. Math.* **286**, 186–195 (2015)
16. Zhang, L.: A derivative-free conjugate residual method using secant condition for general large-scale nonlinear equations. *Numer. Algorithms* (2019). <https://doi.org/10.1007/s11075-019-00725-7>
17. Zhang, L., Zhou, W.: Spectral gradient projection method for solving nonlinear monotone equations. *J. Comput. Appl. Math.* **196**(2), 478–484 (2006)
18. Zhou, W.: Convergence properties of a quasi-Newton method and its applications. Ph.D. thesis, College of Mathematics and Econometrics, Hunan University, Changsha, China (2006)
19. Zhou, W., Li, D.: A globally convergent BFGS method for nonlinear monotone equations without any merit functions. *Math. Comput.* **77**(624), 2231–2240 (2008)
20. Zhou, W., Li, D.: On the Q-linear convergence rate of a class of methods for monotone nonlinear equations. *Pac. J. Optim.* **14**(4), 723–737 (2018)
21. Zhou, W., Wang, F.: A PRP-based residual method for large-scale monotone nonlinear equations. *Appl. Math. Comput.* **261**, 1–7 (2015)

Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)

---