A MODIFIED FLETCHER-REEVES-TYPE DERIVATIVE-FREE METHOD FOR SYMMETRIC NONLINEAR EQUATIONS

DONG-HUI LI
School of Mathematical Sciences, South China Normal University
Guangzhou, 510631, China

XIAO-LIN WANG
College of Mathematics and Econometrics, Hunan University
Changsha, 410082, China

Abstract. In this paper, we propose a descent derivative-free method for solving symmetric nonlinear equations. The method is an extension of the modified Fletcher-Reeves (MFR) method proposed by Zhang, Zhou and Li [25] to symmetric nonlinear equations. It can be applied to solve large-scale symmetric nonlinear equations due to lower storage requirement. An attractive property of the method is that the directions generated by the method are descent for the residual function. By the use of some backtracking line search technique, the generated sequence of function values is decreasing. Under appropriate conditions, we show that the proposed method is globally convergent. The preliminary numerical results show that the method is practically effective.

1. Introduction. We consider the nonlinear equation

$$F(x) = 0,$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is continuously differentiable. We suppose that for each $x \in \mathbb{R}^n$, the Jacobian $F'(x)$ of $F$ at $x$ is symmetric.

Among various methods for solving nonlinear equations (1), Newton’s method is quite welcome due to its nice properties such as the rapid convergence rate, the decreasing of the function value sequence and the global convergence. However, at each iteration, Newton’s method needs the computation of the derivative $F'$ as well as the solution of some system of linear equations.

On the other hand, the derivative-free methods including quasi-Newton methods \[10, 18, 19, 24, 26, 27\], the so called spectral method \[4, 14, 15\] and some recently developed derivative-free methods for monotone nonlinear equations \[20, 23\], can be used to solve nonlinear equations where the derivative is not available or very difficult to compute. The existing derivative-free methods are globally convergent if some nonmonotone line search strategy is used. We refer to a survey paper \[17\] for a summary in the derivative-free quasi-Newton methods for solving nonlinear equations. However, since the first-order information is not available, the existing derivative-free methods are generally not descent method in the sense that the

2000 Mathematics Subject Classification. Primary: 65H10; Secondary: 90C30.

Key words and phrases. Symmetric nonlinear equations, derivative-free method, descent direction, global convergence.

The paper is supported by NSF grant 10771057 and 11071087.
directions generated by the method may not be descent directions for the residual function of (1).

Recently, Gu, Li, Qi and Zhou [10] got a way to construct norm descent quasi-Newton directions and proposed a norm descent Gauss-Newton-based BFGS method for solving symmetric nonlinear equations. It was proved that the method with Armijo-type line search is globally convergent. This technique was extended by Zhang and Li [24] to solving some kinds of nonsmooth equations.

In this paper, we further study the derivative-free methods for nonlinear equations. We focus our attention to the symmetric nonlinear equation (1). We will extend the recently developed modified Fletcher-Reeves (MFR) method [25] for solving unconstrained optimization to the symmetric nonlinear equations. The MFR method is a modification to the well-known Fletcher-Reeves (FR) [8] method for solving optimization problems. The FR method is the first nonlinear conjugate gradient method for solving optimization problems. It enjoys some nice properties such as the finite quadratic termination property and the global convergence property. The global convergence of the FR method for solving optimization problems has been well studied. The first global convergence result for the FR method was given by Zoutendijk [28] in 1970. It was proved in [28] that the FR method with exact line search is globally convergent. The first global convergence result of the FR method with an inexact line search was obtained by Al-Baali [1] in 1985. Other good results about the convergence property of the FR method and its improvement can be found in [5, 7, 9, 11, 13, 16, 21, 22] etc.. We refer to a survey paper [12] for a good review in the convergence property of the FR method and related conjugate gradient methods for optimization.

The MFR method in [25] not only reserves good properties of the FR method but also possesses another nice property that it always generates descent directions for the objective function. This property is independent of the line search used. Under suitable conditions, the MFR method with Armijo line search is also globally convergent [25]. The purpose of this paper is to develop a norm descent MFR type derivative-free method for symmetric nonlinear equations. To this end, we first adopt the strategy in [10] to get an estimation to the gradient of the norm function of (1). We then use the idea of the MFR method in [25] to construct a descent direction for the norm function. By the use of some backtracking line search, we develop an MFR type derivative-free method for solving (1) that generates descent directions for the norm function of the equation (1). Under appropriate conditions, we establish the global convergence of the method. It should be pointed out that in contrast to the derivative-free quasi-Newton methods, the proposed method is not only derivative-free, but also completely matrix-free. Consequently, it can be applied to solve large scale symmetric nonlinear equations. We also do some preliminary numerical experiments to test the proposed method. The results show that the proposed method is practically efficient.

In the next section, we propose the method. Section 3 is devoted to the global convergence of the method. At last we present some numerical results in Section 4.

2. Algorithm. In this section, we propose the MFR-type derivative-free method. First we simply recall the descent derivative-free quasi-Newton method in [10] for solving the symmetric nonlinear equation (1). Denote

\[ f(x) = \frac{1}{2} \| F(x) \|^2. \]  (2)
We call a direction \( d \in \mathbb{R}^n \) a descent direction of \( f \) at \( x \) if it satisfies
\[
\nabla f(x)^T d < 0,
\]
where \( \nabla f(x) = F'(x)F(x) \) denotes the gradient of \( f \) at \( x \). Taking into account the symmetry of \( F(x) \), we have
\[
\nabla f(x) = F'(x)F(x) = \lim_{\lambda \to 0} \frac{F(x + \lambda F(x)) - F(x)}{\lambda}
\]
The above relation shows that the vector \((F(x + \lambda F(x)) - F(x))\lambda^{-1}\) is a good estimate to \( \nabla f(x) \) if \( \lambda \) is small. Based on this observation, Gu, Li, Qi and Zhou [10] got an estimation to the steepest descent direction
\[
- g(\lambda) \triangleq -(F(x + \lambda F(x)) - F(x))\lambda^{-1}. \tag{3}
\]
Let \( x_k \) be the current iterate and \( g_k(\lambda) \) be defined by (3) with \( x = x_k \). Then we have for all sufficient small \( \lambda \)
\[
- \nabla f(x_k)^T g_k(\lambda) < 0.
\]
Let \( \lambda_k > 0 \) be sufficiently small such that \( -\nabla f(x_k)g_k(\lambda_k) < 0 \). Then the quasi-Newton direction \( d_k \) in [10] is determined by the following system of linear equations:
\[
B_k d + g_k(\lambda_k) = 0,
\]
where \( B_k \) is an approximation to \( F'(x_k)F'(x_k) = F'(x_k)^2 \).

By the use of the approximate gradient function \( g_k(\lambda) \), we are going to develop a descent MFR type method for solving symmetric nonlinear equation (1). The basic idea is to use \( -g_k(\lambda_k) \) instead of the steepest descent direction in the MFR method for solving (2). Let us have a look at the MFR method proposed by Zhang, Zhou and Li [25].

Consider the unconstrained optimization problem:
\[
\min f(x), \quad x \in \mathbb{R}^n.
\]
Let \( x_k \) be the current iterate. The direction \( d_k \) in the MFR method of [10] for solving the unconstrained optimization is determined by
\[
d_k = \begin{cases} 
-\nabla f(x_k), & \text{if } k = 0, \\
- \theta_k \nabla f(x_k) + \beta_k^{FR} d_{k-1}, & \text{if } k \geq 1,
\end{cases}
\]
where
\[
\beta_k^{FR} = \frac{||\nabla f(x_k)||^2}{||\nabla f(x_{k-1})||^2}, \quad \theta_k = \frac{d_{k-1}^T y_{k-1}}{||\nabla f(x_{k-1})||^2}
\]
and \( y_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1}) \). This direction enjoys a nice property that
\[
d_k^T \nabla f(x_k) = -||\nabla f(x_k)||^2.
\]
This means that \( d_k \) provides a descent direction of \( f \) at \( x_k \).

We use this idea to develop an MFR type method for solving (1). The key is to get the direction \( d_k \) that takes the form of the MFR direction without computation of derivative and is a descent direction of \( f \) defined by (2) at \( x_k \).

Suppose at the \( k \)-th iteration, we already have the iterate \( x_k \). Let \( d_{k-1} \) be the direction generated by the previous iteration. Similar to the direction \( d_k \) in the MFR method, we define \( d_k(\lambda) \) with parameter \( \lambda \) as follows:
\[
d_k(\lambda) = \begin{cases} 
-g_k(\lambda), & \text{if } k = 0, \\
- \theta_k(\lambda) g_k(\lambda) + \beta_k^{FR}(\lambda) d_{k-1}, & \text{if } k \geq 1,
\end{cases} \tag{4}
\]
\[
\beta_k^{FR}(\lambda) = \frac{\|g_k(\lambda)\|^2}{\|g_{k-1}\|^2}, \quad \theta_k(\lambda) = 1 + \frac{g_k(\lambda)^T d_{k-1}}{\|g_k(\lambda)\|^2} \beta_k^{FR}(\lambda) = 1 + \frac{g_k(\lambda)^T d_{k-1}}{\|g_{k-1}\|^2}, \quad (5)
\]

where \(g_{k-1}\) is an estimation to \(\nabla f(x_k)\) to be determined, and \(g_k(\lambda)\) is defined by (3). By direct computation, we can easily get

\[
g_k(\lambda)^T d_k(\lambda) = -\|g_k(\lambda)\|^2. \quad (6)
\]

It is not difficult to show by the definition of \(d_k(\lambda)\), the limit \(\lim_{\lambda \to 0} d_k(\lambda) = \nabla f(x_k)\) exists. Moreover, it follows from the fact \(\lim_{\lambda \to 0} g_k(\lambda) = \nabla f(x_k)\) that

\[
d_k(0) \triangleq \lim_{\lambda \to 0} d_k(\lambda) = \begin{cases} -\nabla f(x_k), & \text{if } k = 0, \\ -\bar{\theta}_k \nabla f(x_k) + \bar{\beta}_k^{FR} d_{k-1}, & \text{if } k \geq 1, \end{cases}
\]

where

\[
\bar{\beta}_k^{FR} = \frac{\|\nabla f(x_k)\|^2}{\|g_{k-1}\|^2}, \quad \bar{\theta}_k = \frac{d_{k-1}^T \bar{y}_{k-1}}{\|g_{k-1}\|^2} \text{ and } \bar{y}_{k-1} = \nabla f(x_k) - g_{k-1}.
\]

If \(g_{k-1}\) is close to \(\nabla f(x_k)\), then \(d_k(0)\) is close to the direction generated by the MFR method. Therefore we have

\[
\nabla f(x_k)^T d_k(0) = \lim_{\lambda \to 0} g_k(\lambda)^T d_k(\lambda) = -\|\nabla f(x_k)\|^2. \quad (7)
\]

This shows that when \(\lambda\) is sufficiently small, \(d_k(\lambda)\) will be a descent direction of the norm function of \(f\) defined by (2). In our method, we choose \(\lambda\) such that the following inequality is satisfied:

\[
f(x_k + \lambda d_k(\lambda)) \leq f(x_k) + \sigma_1 (F(x_k + \lambda F(x_k)) - F(x_k))^T d_k(\lambda) - \sigma_2 \|\lambda F(x_k)\|^2 - \sigma_3 \|\lambda d_k(\lambda)\|^2, \quad (8)
\]

where constants \(\sigma_1, \sigma_2\) and \(\sigma_3\) satisfy \(\sigma_1 \in (0, 1)\) and \(\sigma_2 > 0, \sigma_3 > 0\).

The following lemma shows that inequality (8) holds for all \(\lambda > 0\) sufficiently small. It follows from (7) immediately.

**Lemma 2.1.** Let \(F\) be continuously differentiable and \(F'(x)\) is symmetric for every \(x\). Then the inequality (8) holds for all \(\lambda > 0\) sufficiently small.

The following procedure provides a way to determine \(d_k\) and \(g_k\). It is very similar to the Procedure 1 in [10].

**Procedure 1.**

Let \(g_k(\lambda)\) be defined by (3) with \(x = x_k\) and \(d_k(\lambda)\) be computed by (4) and (5). Given constant \(\sigma_1, \rho \in (0, 1)\) and \(\sigma_2 > 0, \sigma_3 > 0\). Let \(i_k\) be the smallest nonnegative integer such that inequality (8) holds with \(\lambda = \rho^i, i = 0, 1, \ldots\). Let \(d_k = d_k(\rho^{i_k})\) and \(g_k = g_k(\rho^{i_k})\).

To get a relatively larger steplength, we use the following procedure to determine the steplength \(\lambda_k\).

**Procedure 2.**

Let \(d_k\) be generated by Procedure 1. Let constants \(\sigma_i, i = 1, 2, 3\) and \(\rho\) be the same as those in Procedure 1. If \(i_k = 0\), we let \(\lambda_k = 1\). Otherwise, we let \(j_k\) be the largest positive integer \(j_k \in \{0, 1, 2, \ldots, i_k - 1\}\) satisfying

\[
f(x_k + \rho^{i_k-j_k} d_k) \leq f(x_k) + \sigma_1 (F(x_k + \rho^{i_k-j_k} F(x_k)) - F(x_k))^T d_k - \sigma_2 \|\rho^{i_k-j_k} F(x_k)\|^2 - \sigma_3 \|\rho^{i_k-j_k} d_k\|^2. \quad (9)
\]

Let \(\lambda_k = \rho^{i_k-j_k}\).
Now, we give the steps of the MFR type derivative-free method for solving the symmetric nonlinear equation (1) as follows.

**Algorithm 1. (MFR type derivative-free method)**

1. **Step 0.** Given initial point $x_0 \in \mathbb{R}^n$. Let $k := 0$.
2. **Step 1.** Use Procedure 1 to determine the direction $d_k$.
3. **Step 2.** Use Procedure 2 to determine the steplength $\lambda_k$.
4. **Step 3.** Let the next iterate be $x_{k+1} = x_k + \lambda_k d_k$. Let $k := k + 1$. Go to Step 1.

It is easy to see that the sequence of the function evaluations $\{f(x_k)\}$ generated by Algorithm 1 is decreasing. The following proposition shows that Algorithm 1 is well defined.

**Proposition 1.** Let $\{x_k\}$ be generated by Algorithm 1. Suppose that $F$ is continuously differentiable and for each $x \in \mathbb{R}^n$, $F'(x)^T = F'(x)$. If $\nabla f(x_k) \neq 0$ for all $k$, then we have $g_k \neq 0$, $\forall k$. Moreover, Procedure 1 terminates finitely.

**Proof.** It follows from Lemma 2.1 that the inequality (8) holds for all $\lambda > 0$ sufficiently small. We only need to show that for any $k$, $d_k$ is well defined. By the definition of $d_k$ and $d_k(\lambda)$, it suffices to show that $g_k \neq 0$, $\forall k \geq 0$. We prove it by induction.

For $k = 0$, if $g_0 = g_0(\rho^{\alpha}) = 0$, where $\rho^{\alpha}$ is determined by Procedure 1, then we have $d_0 = d_0(\rho^{\alpha}) = -g_0 = 0$. As a result, the inequality (8) does not hold with $\lambda = \rho^{\alpha}$. It is a contradiction. Consequently, we must have $g_0 \neq 0$.

Suppose for some $k \geq 1$, $g_{k-1} \neq 0$. We are going to show that $g_k \neq 0$. Suppose on the contrary that $g_k = 0$. Then we have $\theta_k(\rho^{\alpha}) = 0$, $\beta_1(\rho^{\alpha}) = 0$, where $\rho^{\alpha}$ is determined by Procedure 1. By the definition of $d_k$, it holds that $d_k = 0$. Consequently, inequality (8) does not holds with $\lambda = \rho^{\alpha}$. This yields a contradiction too. The proof is then complete. \[\blacksquare\]

3. **Global convergence.** In this section, we prove the global convergence of Algorithm 1. We first make the following assumption.

**Assumption 1.**

1. The level set $\Omega = \{x \in \mathbb{R}^n|f(x) \leq f(x_0)\}$ is bounded.
2. In some neighborhood $N$ of $\Omega$, $F$ is continuously differentiable and $F'(x)$ is Lipschitz continuous, namely, there exists a constant $L > 0$ such that
   \[\|F'(x) - F'(y)\| \leq L\|x - y\|, \forall x, y \in N.\] (10)

Since $\{f(x_k)\}$ is decreasing, it is clear that the sequence $\{x_k\}$ generated by Algorithm 1 is contained in $\Omega$ and hence bounded. In addition, it is easy to see that there are positive constants $M_1$, $M_2$ and $L_1$ such that

\[\|F'(x)\| \leq M_1, \|F(x)\| \leq M_2, \forall x, y \in N,\] (11)

\[\|\nabla f(x) - \nabla f(y)\| \leq L_1\|x - y\|, \forall x, y \in N.\] (12)

In the latter part of this section, without specification, we always suppose that the conditions in Assumption 1 hold and that $F'(x)$ is symmetric for every $x \in N$. The following lemma can be obtained by the steps of Algorithm 1 and the conditions of Assumption 1 directly.
Lemma 3.1. Let the sequence \( \{x_k\} \) be generated by Algorithm 2.1. Then we have
\[
\sum_{k=0}^{\infty} \|\lambda_k F(x_k)\|^2 < \infty
\]
and
\[
\sum_{k=0}^{\infty} \|\lambda_k d_k\|^2 < \infty.
\]

The following lemma estimates a lower bound to the steplength.

Lemma 3.2. Let \( \{x_k\} \) be generated by the MFR type method. Then there is a constant \( C > 0 \) such that the steplength \( \lambda_k \) generated by Procedure 2 satisfies
\[
\lambda_k \geq \min\{1, -\frac{-\nabla f(x_k)^T d_k}{\|d_k\|^2 + \|F(x_k)\|^2 C}\}.
\]

Proof. It follows from Procedure 2 that if \( \lambda_k \neq 1 \), then \( \rho^{-1} \lambda_k \) dose not satisfy inequality (8). This implies
\[
\sigma_2 \rho^{-1} \lambda_k F(x_k) \|d_k\|^2 + \sigma_3 \rho^{-1} \lambda_k d_k \geq -[f(x_k + \rho^{-1} \lambda_k d_k) + f(x_k)] + \sigma_1 (F(x_k + \rho^{-1} \lambda_k F(x_k)) - F(x_k))^T d_k.
\]

By the use of the mean-value theorem and inequality (12), there is a \( t_k \in (0, 1) \) such that
\[
\begin{align*}
f(x_k + \rho^{-1} \lambda_k d_k) - f(x_k) &= \rho^{-1} \lambda_k \nabla f(x_k + t_k \rho^{-1} \lambda_k d_k)^T d_k \\
&= \rho^{-1} \lambda_k \nabla f(x_k)^T d_k + \rho^{-1} \lambda_k (\nabla f(x_k + t_k \rho^{-1} \lambda_k d_k) - \nabla f(x_k))^T d_k \\
&\leq \rho^{-1} \lambda_k \nabla f(x_k)^T d_k + L_1 \rho^{-2} \lambda_k^2 \|d_k\|^2.
\end{align*}
\]

Similarly, by the use of the mean-value theorem to the one dimensional function
\[
\phi(t) \triangleq (F(x_k + t\rho^{-1} \lambda_k F(x_k)) - F(x_k))^T d_k,
\]
there is a \( \bar{t}_k \in (0, 1) \) such that
\[
\begin{align*}
(F(x_k + \rho^{-1} \lambda_k F(x_k)) - F(x_k))^T d_k &= \rho^{-1} \lambda_k F(x_k)^T F'(x_k + \bar{t}_k \rho^{-1} \lambda_k F(x_k)) d_k \\
&= \rho^{-1} \lambda_k F(x_k)^T F'(x_k) d_k + \rho^{-1} \lambda_k F(x_k)^T F'(x_k) d_k \\
&\geq \rho^{-1} \lambda_k F(x_k)^T F'(x_k) d_k - L \rho^{-2} \lambda_k^2 \|F(x_k)\|^2 \|d_k\| \\
&= \rho^{-1} \lambda_k \nabla f(x_k)^T d_k - L \rho^{-2} \lambda_k^2 \|F(x_k)\|^2 \|d_k\|,
\end{align*}
\]
where the inequality follows from (11).

We get from inequalities (16), (17) and (18)
\[
\rho^{-1} \lambda_k (\sigma_2 \|F(x_k)\|^2 + \sigma_3 \|d_k\|^2 + L_1 \|d_k\|^2 + L \|F(x_k)\|^2 \|d_k\|) \leq -(1 - \sigma_1) \nabla f(x_k)^T d_k.
\]

Consequently, it holds that
\[
\lambda_k \geq -(1 - \sigma_1) \rho \frac{-\nabla f(x_k)^T d_k}{\sigma_2 \|F(x_k)\|^2 + (\sigma_3 + L_1) \|d_k\|^2 + L \|F(x_k)\|^2 \|d_k\|}.
\]

Since \( \{\|F(x_k)\|\} \) is bounded, the last inequality shows that there is a constant \( C > 0 \) such that
\[
\lambda_k \geq \frac{-\nabla f(x_k)^T d_k}{\|d_k\|^2 + \|F(x_k)\|^2 C}
\]
if \( \lambda_k \neq 1 \). By the line search rule, we get (15). \( \square \)
The next lemma reveals a good property of $d_k$, which is similar to a property of $d_k$ generated by the MFR method for unconstrained optimization. We let $\lambda_k = \beta^k$, where $\beta^k$ is determined by Procedure 1. Then we have $g_k = g_k(\lambda_k)$ and $d_k = a_k(\lambda_k)$. The equality (5) is written as

$$g_k^T d_k = -\|g_k\|^2.$$  

(19)

For simplicity, we abbreviate $\theta_k(\lambda_k)$ and $\beta_k^{FR}(\lambda_k)$ as $\theta_k$ and $\beta_k^{FR}$ respectively, where $\theta_k$ and $\beta_k^{FR}$ are defined by (5).

**Lemma 3.3.** Let $\{x_k\}$ be generated by Algorithm 1. Then we have

$$\|d_k\|^2 \geq \frac{1}{\|g_k\|^2} \sum_{i=0}^{k-1} \|g_i\|^2.$$  

(20)

**Proof.** By the definition of $\theta_k$, we can rewrite $d_k$ as

$$d_k = -g_k + \beta_k^{FR}(I - g_kg_k^T)g_k.$$  

(21)

where $I \in \mathbb{R}^{n \times n}$ denotes the identity matrix. We note that the two terms in the right hand side of (21) are orthogonal. Taking Euclidian norm in both sides of (21), we get for each $i \geq 1$

$$\|d_i\|^2 = \|g_i\|^2 + (\beta_i^{FR})^2 \|d_i-1\|^2 \leq \|g_i\|^2 + (\beta_i^{FR})^2 \|d_i-1\|^2 = \|g_i\|^2 + \frac{\|g_i\|^4}{\|d_i-1\|^2} \|d_i-1\|^2.$$  

Dividing by $\|g_i\|^4$ in both sides of the last equality and then taking summation from $i = 1$ to $k$, we get the desired inequality (20).

The following theorem establishes the global convergence of Algorithm 1.

**Theorem 3.4.** Let $\{x_k\}$ be generated by the MFR type method. Then we have

$$\lim_{k \to \infty} \|\nabla f(x_k)\| = 0.$$  

(22)

Moreover, if there is an accumulation point $\bar{x}$ of $\{x_k\}$ at which $F'(\bar{x})$ is nonsingular, then every accumulation point of $\{x_k\}$ is a solution of (1).

**Proof.** Since the sequence of function values $\{f(x_k)\}$ is decreasing, we only need to verify (22).

Denote $\lambda = \limsup_{k \to \infty} \lambda_k$. We can easily get from Algorithm 1 $\lambda \in [0, 1]$. If $\lambda > 0$, we get from (13) that $\liminf_{k \to \infty} \|F(x_k)\| = 0$, which implies (22).

Suppose $\lambda = 0$ or equivalently $\lim_{k \to \infty} \lambda_k = 0$. By the definition of $g_k$ and $\lambda_k$, it is not difficult to get

$$\liminf_{k \to \infty} \|g_k\| = \liminf_{k \to \infty} \|\nabla f(x_k)\|.$$  

(23)

It then suffices to show

$$\liminf_{k \to \infty} \|g_k\| = 0.$$  

(24)

For the sake of contradiction, we suppose that the (24) dose not hold. Then there exists a constant $\varepsilon > 0$ such that $\|g_k\| \geq \varepsilon$ for all $k \geq 0$. By the boundedness
of \(|\{\|F(x_k)\|\}|\), we get from (15) and (23) that there is a constant \(M > 0\) such that the following inequality holds for all \(k\) sufficiently large

\[
\lambda_k \geq \frac{-g_k^T d_k}{\|d_k\|^2 + M} C = \frac{\|g_k\|^2}{\|d_k\|^2 + M} C, 
\]

where the equality follows from (19). This together with (14) implies

\[
\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{(\|d_k\|^2 + M)^2} \|d_k\|^2 < \infty.
\]

However, we get from (19) that \(\|d_k\| \geq \|g_k\| \geq \varepsilon\). The last inequality implies

\[
\sum_{k=0}^{\infty} \|d_k\|^2 + 2M + M^2 \varepsilon^{-2} < \infty. 
\]

On the other hand, we get from (20)

\[
\|d_k\|^2 \leq k \varepsilon^{-2} \|g_k\|^4 \leq M \varepsilon^{-2} k,
\]

where \(M\) is an upper bound of \(\{\|g_k\|^4\}\). So, we get from (25)

\[
\sum_{k=0}^{\infty} M \varepsilon^{-2} k + 2M + M^2 \varepsilon^{-2} \leq \sum_{k=0}^{\infty} \|d_k\|^2 + 2M + M^2 \varepsilon^{-2} < \infty.
\]

It is a contradiction. Consequently, (24) holds and the proof is complete. \(\blacksquare\)

4. Numerical experiments. In this section, we report some preliminary numerical results with the proposed method. We test the performance of Algorithm 1 on the following two problems coming from [2], [18] and [27] with various dimensions and different initial points.

**Problem 1.** The unconstrained optimization problem

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

with Engval function \(f : \mathbb{R}^n \to \mathbb{R}\) defined by

\[
f(x) = \sum_{i=2}^{n} \{(x_{i-1}^2 + x_i^2)^2 - 4x_{i-1} + 3\}
\]

The related symmetric nonlinear equation is

\[
F(x) = \frac{1}{4} \nabla f(x) = 0
\]

where \(F(x) = (F_1(x), F_2(x), \ldots, F_n(x))^T\) with

\[
F_1(x) = x_1(x_1^2 + x_2^2) - 1,
\]

\[
F_i(x) = x_i(x_{i-1}^2 + 2x_i^2 + x_{i+1}^2) - 1, \quad i = 2, 3, \ldots, n - 1,
\]

\[
F_n(x) = x_n(x_{n-1}^2 + x_n^2).
\]

**Problem 2.** The function \(F\) is given by

\[
F(x) = Ax + g(x)
\]

where \(g(x) = (e^{x_1} - 1, e^{x_2} - 1, \ldots, e^{x_n} - 1)^T\) and \(A\) is given by [2].
We compared the performance of Algorithm 1, which we call MFR-t, with that of the descent BFGS method in [10], which we call DBFGS, and the DF_SANE method in [15] on the two problems. The algorithms were coded in MATLAB 6.5 and run on a personal computer with a 1.66GHZ CPU processor.

The parameters are specified as follows. For Algorithm 1, we set \( \sigma_1 = 10^{-4}, \sigma_2 = 10^{-4}, \sigma_3 = 10^{-4}, \rho = 0.4 \). For the DBFGS method in [10], we set \( c = 10^{-5}, tol = 10^{-5}, maxstep = 2000 \) and \( B_0 = A \). The parameters of the DF_SANE method were set to be the same as those used in [15]. We stopped the iteration if the inequality \( f \leq 10^{-5} \) is satisfied or the number of iterations exceeds \( 10^4 \).

We first tested the algorithms on small and medium size problems and compared them in the total number of iterations and the CPU time used. The results are listed in Tables 1 and 2. The meaning of each column in Tables 1 and 2 is stated as follows.

- "Init": the initial point \( x_1 = (1,1/2, \ldots ,1/n)^T \);
- "a": means \( x_0 = (a,a, \ldots ,a)^T \);
- "Dim": the dimension of the problem;
- "iter": the total number of iterations;
- "time": the CPU time in seconds;
- "f": the final value of the norm function \( f \) of the equation.

In each table, we also counted the average number of iterations or the average CPU time, which corresponds to the "average" row.

Table 3 lists the average CPU time and the average iteration numbers for each algorithm on the two problems with different dimensions.

We see from Tables 4.1-4.3 that the proposed method can solve the symmetric nonlinear equations efficiently. In all cases, the method terminated at a solution of the problem. In most cases, the MFR-t method performed much better than the DBFGS method did. This feature becomes more evident as the dimension is increased to 200. The results in Table 3 also show that the MFR-t method performed better than the DF_SANE method did, especially for Problem 1 (see Table 1). However, The MFR-t method performed not as well as the DF_SANE method for Problem 2 (see Table 2).

| init  | Dim | Iter  | time   | f    | Iter  | time   | f    |
|-------|-----|-------|--------|------|-------|--------|------|
| 0     | 50  | 51    | 0.1870 | 8.4E-06 | 43    | 0.4840 | 6.5E-06 | 730  | 0.4600 | 9.8E-06 | 0.5600 | 9.6E-06 |
| 1/n^2 | 50  | 42    | 0.1250 | 7.3E-06 | 43    | 0.4690 | 9.7E-06 | 732  | 0.4370 | 9.9E-06 | 0.5600 | 9.6E-06 |
| -1/n^2 | 50 | 47    | 0.1720 | 9.0E-06 | 44    | 0.5000 | 6.2E-06 | 727  | 0.4210 | 9.9E-06 | 0.5600 | 9.6E-06 |
| 0.01  | 50  | 46    | 0.1720 | 8.1E-06 | 39    | 0.3280 | 4.9E-06 | 757  | 0.4530 | 9.8E-06 | 0.5600 | 9.6E-06 |
| -0.01 | 50  | 15    | 0.0470 | 4.5E-06 | 37    | 0.3280 | 5.4E-06 | 668  | 0.3910 | 9.9E-06 | 0.5600 | 9.6E-06 |
| x^1   | 50  | 37    | 0.1250 | 8.6E-06 | 43    | 0.4840 | 3.8E-06 | 63   | 0.0470 | 8.4E-06 | 0.5600 | 9.6E-06 |
|       |     |       | average|       |       |        |        |      |        |        |        |        |
| 0     | 100 | 36    | 0.1400 | 9.6E-06 | 65    | 1.4840 | 9.8E-06 | 744  | 0.5000 | 9.9E-06 | 0.5600 | 9.6E-06 |
| 1/n^2 | 100 | 53    | 0.1880 | 9.0E-06 | 68    | 1.3280 | 8.0E-06 | 775  | 0.5150 | 9.8E-06 | 0.5600 | 9.6E-06 |
| -1/n^2 | 100 | 31    | 0.1100 | 7.2E-06 | 67    | 1.3600 | 9.2E-06 | 773  | 0.5000 | 9.9E-06 | 0.5600 | 9.6E-06 |
| 0.01  | 100 | 46    | 0.1870 | 9.1E-06 | 65    | 1.3440 | 8.3E-06 | 792  | 0.5470 | 9.8E-06 | 0.5600 | 9.6E-06 |
| -0.01 | 100 | 27    | 0.1090 | 9.0E-06 | 70    | 1.6250 | 9.8E-06 | 709  | 0.4680 | 9.8E-06 | 0.5600 | 9.6E-06 |
| x^1   | 100 | 46    | 0.1880 | 9.1E-06 | 66    | 1.9060 | 8.6E-06 | 797  | 0.5470 | 9.9E-06 | 0.5600 | 9.6E-06 |
|       |     | average| 39.8   | 0.1537 | 9.2E-06 | 66.8  | 1.5078 | 9.0E-06 | 770  | 0.5128 | 9.9E-06 | 0.5600 | 9.6E-06 |

TABLE 4.1: Test results for Problem 1.
We then tested the MFR\textsuperscript{t} method and the DF\textsubscript{SANE} method on the two problems with larger dimensions. The results are listed in Tables 4 and 5. We see from Table 4 that the MFR\textsuperscript{t} method also performed better than the DF\textsubscript{SANE} method did on Problem 1. The results in Table 5 show that both methods can solve Problem 2 efficiently. However, the performance of the MFR\textsuperscript{t} method is not as good as that of the DF\textsubscript{SANE} method.
REFERENCES

[1] M. Al-Baali, *Descent property and global convergence of the Fletcher-Reeves method with inexact line search*, IMA Journal of Numerical Analysis, 5 (1985), 121–124.

[2] S. Bellavia and B. Morini, *A globally convergent Newton-GMRES subspace method for systems of nonlinear equations*, SIAM Journal on Scientific Computing, 23 (2001), 940–960.

[3] A. Griewank, *The ‘global’ convergence of Broyden-like methods with suitable line search*, Journal of Australia Mathematical Society, Ser. B, 28 (1986), 75–92.

[4] W. Cheng and D. H. Li, *A derivative-free nonmonotone line search and its application to the spectral residual method*, IMA Journal of Numerical Analysis, 29 (2009), 814–825.

[5] Y. H. Dai and Y. Yuan, *Convergence of the Fletcher-Reeves method under a generalized Wolfe search*, Journal of Computational Mathematics, 2 (1996), 142–148.

[6] Y. H. Dai and Y. Yuan, *Convergence properties of the Fletcher-Reeves method*, IMA Journal of Numerical Analysis, 16 (1996), 155–164.

[7] Y. H. Dai and Y. Yuan, “Nonlinear Conjugate Gradient Methods,” Shanghai Science and Technology Publisher, Shanghai, 2000.

[8] R. Fletcher and C. Reeves, *Function minimization by conjugate gradients*, Computer Journal, 7 (1964), 149–154.

[9] J. C. Gilbert and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM Journal on Optimization, 2 (1992), 21–42.

[10] G. Z. Gu, D. H. Li, L. Qi and S. Z. Zhou, *Descent directions of Quasi-Newton methods for symmetric nonlinear equations*, SIAM Journal on Numerical Analysis, 40 (2003), 1763–1774.

[11] J. Y. Han, G. H. Liu and H. X. Yin, *Convergence properties of conjugate gradient methods with strong Wolfe linesearch*, Systems Science and Mathematical Science, 11 (1998), 112–116.

[12] W. W. Hager and H. Zhang, *A survey of nonlinear conjugate gradient methods*, Pacific Journal of Optimization, 2 (2006), 35–58.

[13] Y. F. Hu and C. Storey, *Global convergence result for conjugate gradient methods*, Journal of Optimization Theory and Applications, 71 (1991), 399–405.

[14] W. La Cruz and M. Raydan, *Nonmonotone spectral methods for large-scale nonlinear systems*, Optimization Methods and Software, 18 (2003), 583–599.

[15] W. La Cruz, J. M. Martínez and M. Raydan, *Spectral residual method without gradient information for solving large-scale nonlinear systems of equations*, Mathematics of Computation, 75 (2006), 1429–1448.

[16] G. H. Liu, J. Y. Han and H. X. Yin, *Global convergence of the Fletcher-Reeves algorithm with an inexact line search*, Applied Mathematics, Journal of Chinese Universities, Ser. B, 10 (1995), 75–82.

[17] D. H. Li and W. Cheng, *Recent progress in the global convergence of quasi-Newton methods for nonlinear equations*, Hokkaido Journal of Mathematics, 36 (2007), 729–743.

[18] D. H. Li and M. Fukushima, *A globally and superlinearly convergent Gauss-Newton based BFGS method for symmetric nonlinear equations*, SIAM Journal on Numerical Analysis, 37 (1999), 152–172.

[19] D. H. Li and M. Fukushima, *A derivative-free line search and global convergence of Broyden-like methods for nonlinear equations*, Optimization Methods and Software, 13 (2000), 181–201.

[20] Q. Li and D. H. Li, *A class of derivative-free methods for large-scale nonlinear monotone equations*, IMA Journal of Numerical Analysis, (to appear).

[21] M. J. D. Powell, *Some convergence properties of the conjugate gradient method*, Mathematical Programming, 11 (1976), 42–49.

[22] M. J. D. Powell, *Restart procedures of the conjugate gradient method*, Mathematical Programming, 2 (1977), 241–254.
Q. Yan, X. Z. Peng and D. H. Li, *A globally convergent derivative-free method for solving large-scale nonlinear monotone equations*, Journal of Computational and Applied Mathematics, 234 (2010), 649–657.

J. Zhang and D. H. Li, *A norm descent BFGS method for solving KKT systems of symmetric variational inequality problems*, Optimization Methods and Software, 22 (2007), 237–252.

L. Zhang, W. Zhou and D. H. Li, *Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search*, Numerische Mathematik, 104 (2006), 561–572.

W. Zhou and D. H. Li, *Limited memory BFGS method for nonlinear monotone equations*, Journal of Computational Mathematics, 25 (2007), 89–96.

W. Zhou and D. H. Li, *A globally convergent BFGS method for nonlinear monotone equations*, Mathematics of Computation, 77 (2008), 2231–2240.

G. Zoutendijk, *Nonlinear Programming, Computational Methods*, in “Integer and Nonlinear Programming” (eds. J. Abadie), North-Holland, Amsterdam, (1970), 37–86.

Received October 2010; revised October 2010.

E-mail address: dhli@scnu.edu.cn

E-mail address: wofan-1986@163.com