J. Oper. Res. Soc. China (2014) 2:291–305
DOI 10.1007/s40305-014-0051-1

A Nonmonotone Hybrid Method of Conjugate Gradient and Lanczos-type for Solving Nonlinear Systems

Chun-Xia Jia · Jue-Yu Wang · De-Tong Zhu

Received: 8 October 2013 / Revised: 22 July 2014 / Accepted: 26 July 2014 /
Published online: 2 September 2014

Abstract In this paper, we construct a new algorithm which combines the conjugate gradient and Lanczos methods for solving nonlinear systems. The iterative direction can be obtained by solving a quadratic model via conjugate gradient and Lanczos methods. Using the backtracking line search, we will find an acceptable trial step size along this direction which makes the objective function nonmonotonically decreasing and makes the norm of the step size monotonically increasing. Global convergence and local superlinear convergence rate of the proposed algorithm are established under some reasonable conditions. Finally, we present some numerical results to illustrate the effectiveness of the proposed algorithm.

Keywords Nonmonotonic technique · Nonlinear systems · Lanczos method · Conjugate gradient

The authors gratefully acknowledge the partial supports of the National Natural Science Foundation of China (No. 11371253).

C.-X. Jia (✉) · J.-Y. Wang
Department of Mathematics, Shanghai Normal University, Shanghai 200234, China
e-mail: jiachx@shnu.edu.cn

J.-Y. Wang
e-mail: shnu201005@hotmail.com

D.-T. Zhu
Department of Mathematics, Shanghai Normal University, Guilin Road 100, Shanghai 200234, China
e-mail: dtzhu@shnu.edu.cn

1 Introduction

This paper is concerned with the development of conjugate gradient and Lanczos methods for the solution of nonlinear systems:

$$F(x) = 0, \tag{1.1}$$

where $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a given continuously differentiable mapping.

There are quite a few literatures proposing affine-scaling algorithm for solving problems appeared during the last few years. Sun [15] gave a convergence proof for an affine-scaling algorithm for convex quadratic programming without nondegeneracy assumptions, and Ye [17] introduced affine-scaling algorithm for nonconvex quadratic programming. Classical methods also can be used to solve (1.1), for example, nonlinear conjugate gradient method, which can be easily programmed and computed, is one of the most popular and useful method for solving large-scale optimization problems(see [3, 4, 9, 10]). The idea of conjugate gradient path in unconstrained optimization is given in [1]; which is defined as linear combination of a sequence of conjugate directions that are obtained by applying standard conjugate direction method to approximate quadratic function of unconstrained optimization. The Lanczos method for solving the quadratic-model trust region subproblem in a weighted l_2-norm is proposed by Gould et al. in [5]. By combining Lanczos method with conjugate gradient path, we can construct a new path (see [7, 11]), which has both properties of Lanczos vectors and properties of conjugate gradient path.

Stimulated by the progress in these aspects, in this paper, an algorithm via the conjugate gradient and Lanczos methods is proposed to solve (1.1). Define the merit function

$$f(x) = \frac{1}{2}\|F(x)\|^2 = \frac{1}{2}\sum_{i=1}^{n} F_i^2(x). \tag{1.2}$$

The necessary condition of the problem (1.1) is to solve the following optimization problem $\min f(x)$. The basic idea in the proposed algorithm is based on the minimal value of the following quadratic programming subproblem

$$\min \psi_k(p) = \frac{1}{2}\|F_k^{'}p + F_k\|^2 = \frac{1}{2}p^{\mathrm{T}}F_k^{'\mathrm{T}}F_k^{'}p + g_k^{\mathrm{T}}p + \frac{1}{2}\|F_k\|^2, \tag{1.3}$$

where $F_k = F(x_k), F_k^{'} = F'(x_k), g_k = \nabla f(x_k) = F_k^{'\mathrm{T}}F_k$, and $\psi_k(p)$ is an adequate representation of $f(x)$ around x_k.

The paper is organized as follows. In Sect. 2, the concrete algorithm for solving (1.1) is stated. In Sect. 3, we prove the global convergence of the proposed algorithm. Further, we establish that the proposed algorithm has strong global convergence and local convergence rate in Sect. 4. Finally, the results of numerical experiments of the proposed algorithm are reported in Sect. 5.

2 Algorithm

This section describes and designs the conjugate gradient and Lanczos methods in association with nonmonotonic backtracking technique for solving the nonlinear system (1.1).

2.1 Algorithm NCGL

We are now in a position to give a precise statement of the nonmonotone hybrid method of conjugate gradient and lanczos technique.

Initialization Step

Choose parameters $\beta \in \left(0, \frac{1}{2}\right)$, $\omega \in (0,1), \varepsilon > 0$ and positive integer M as nonmonotonic parameter. Let $m(0) = 0$ and $\xi \in (0,1)$, give a starting strict feasibility interior point $x_0 \in \mathbb{R}^n$. Set $k = 0$, go to the main step.

Main Step

1. Evaluate $f_k = f(x_k) \overset{\text{def}}{=} \frac{1}{2}\|F(x_k)\|^2, g_k = \nabla f(x_k) \overset{\text{def}}{=} (F_k')^{\mathrm{T}} F_k$.
2. If $\|g_k\| = \|(F_k')^{\mathrm{T}} F_k\| \leqslant \varepsilon$, stop with the approximate solution x_k.
3. $q_0 = 0, v_1 = 0, r_1 = \nabla$
   $\psi_k(v_1) = g_k, d_1 = -g_k, \theta_1 = 1, \gamma_1 = \|r_1\|, q_1 = \frac{r_1}{\gamma_1}$. Let $i = 1$.
4. Compute $w_i = F_k' d_i$. If

$$\|w_i\| \neq 0 \tag{2.1}$$

$$r_i \neq 0 \tag{2.2}$$

   go to step 5, otherwise go to step 6.
5. Calculate

$$\begin{aligned}
\lambda_i &= \frac{\theta_i^2 \|r_i\|^2}{\|w_i\|^2}, \\
v_{i+1} &= v_i + \lambda_i d_i, \\
\theta_{i+1} &= -\lambda_i \theta_i \gamma_i, \\
\delta_i &= \|F_k' q_i\|^2, \\
r_{i+1} &= F_k'^{\mathrm{T}} F_k' q_i - \delta_i q_i - \gamma_i q_{i-1}, \\
\gamma_{i+1} &= \|r_{i+1}\|, \\
q_{i+1} &= \frac{r_{i+1}}{\gamma_{i+1}}, \\
\beta_i &= \frac{\theta_{i+1} r_{i+1}^{\mathrm{T}} F_k'^{\mathrm{T}} w_i}{\|w_i\|^2}, \\
d_{i+1} &= -\theta_{i+1} r_{i+1} + \beta_i d_i.
\end{aligned}$$

Calculate

$$f(x_k) - f(x_k + v_{i+1}) \geqslant \xi\Big[f(x_k) - \psi_k(v_{i+1})\Big]. \tag{2.3}$$

If (2.3) is not satisfied, set $i \Leftarrow i+1$, go to 4.

6. If $i = 1, p_k = d_1$, otherwise, $p_k = v_i$.
7. Choose $\alpha_k = 1, \omega, \omega^2, \cdots$, until the following inequality is satisfied:

$$f(x_k + \alpha_k p_k) \leqslant f(x_{l(k)}) + \alpha_k \beta g_k^{\mathrm{T}} p_k, \tag{2.4}$$

where $f(x_{l(k)}) = \max\limits_{0 \leqslant j \leqslant m(k)} \{f(x_{k-j})\}$.

8. Set

$$x_{k+1} = x_k + \alpha_k p_k. \tag{2.5}$$

9. Take the nonmonotone control parameter $m(k+1) = \min\{m(k)+1, M\}$. Then set $k \leftarrow k+1$ and go to step 1.

2.2 Properties of the Proposed Algorithm

The following lemmas give some properties of the algorithm.

Lemma 2.1 (see [7, 11]) *Suppose that the directions q_i and d_i are generated by step 5 of Algorithm NCGL, $1 \leqslant i \leqslant l \leqslant n_k$, the following properties hold:*

$$q_i^{\mathrm{T}} q_j = 0, \quad 1 \leqslant j < i \leqslant l \leqslant n_k \tag{2.6}$$

$$Q_i^{\mathrm{T}} F_k'^{\mathrm{T}} F_k' Q_i = T_i, \quad i = 1, 2, \cdots, n_k \tag{2.7}$$

$$r_i^{\mathrm{T}} d_j = 0, \quad 1 \leqslant j < i \leqslant l \leqslant n_k \tag{2.8}$$

$$w_i^{\mathrm{T}} w_j = 0, \quad i \neq j \tag{2.9}$$

$$d_i^{\mathrm{T}} d_j \geqslant 0, \quad 1 \leqslant i, j \leqslant n_k \tag{2.10}$$

where $Q_i = [q_1, q_2, \cdots, q_i]$ and the tridiagonal matrix T_i is

$$T_i = \begin{bmatrix} \delta_1 & \gamma_2 & & & \\ \gamma_2 & \delta_2 & \gamma_3 & & \\ & \ddots & \ddots & \ddots & \\ & & & \delta_{i-1} & \gamma_i \\ & & & \gamma_i & \delta_i \end{bmatrix}.$$

Lemma 2.2 (see [7, 11]) *Suppose that* $\nabla\psi_k(v_{i+1}) = F_k^{'\mathrm{T}} F_k^{\mathrm{T}} v_{i+1} + g_k = \theta_{i+1} r_{i+1}$ (see [5]), where $\theta_{i+1} = \langle e_{i+1}, h_{i+1}\rangle$, e_i *is a unit vector and its the ith* 1, h_{i+1} *satisfy* $T_{i+1}h_{i+1} + \gamma_1 e_1 = 0$. *Then we have*

$$\theta_{i+1} = -\lambda_i\theta_i\gamma_i \ (\theta_1 = 1).$$

3 Global Convergence Analysis

Throughout this section, we assume that $F : \mathbb{R}^n \to \mathbb{R}^n$ is continuously differentiable. Given $x_0 \in \mathbb{R}^n$, the algorithm generates a sequence $\{x_k\} \subset \mathbb{R}^n$. In our analysis, the level set of f is denoted by

$$\mathcal{L}(x_0) = \{x \in \mathbb{R}^n | f(x) \leqslant f(x_0)\}.$$

In order to discuss the properties of Algorithm NCGL in detail, we will summarize as follows.

Lemma 3.1 *Consider the step* v_j *be obtained from NCGL. Then the norm of the step* v_j *is monotonically increasing and the quadratic function* $\psi_k(v_j)$ *is monotonically decreasing, that is,* $\|v_j\| \leqslant \|v_{j+1}\|$ and $\psi_k(v_{j+1}) \leqslant \psi_k(v_j)$.

Proof Because $v_1 = 0$, $\lambda_i > 0$ and $v_j^{\mathrm{T}} d_j = \sum_{i=1}^{j-1} \lambda_i d_i^{\mathrm{T}} d_j \geqslant 0$,

$$\begin{aligned}\|v_{j+1}\|^2 &= (v_j + \lambda_j d_j)^{\mathrm{T}}(v_j + \lambda_j d_j)\\ &= \|v_j\|^2 + 2\lambda_j v_j^{\mathrm{T}} d_j + \lambda_j^2\|d_j\|^2 \geqslant \|v_j\|^2,\end{aligned}$$

which means that $\|v_j\| \leqslant \|v_{j+1}\|$ holds.

Using the expression of ψ_k, v_j and (2.9), it is clear that

$$\begin{aligned}&\psi_k(v_{j+1}) - \psi_k(v_j)\\ &= g_k^{\mathrm{T}}(v_{j+1} - v_j) + \frac{1}{2} v_{j+1}^{\mathrm{T}} F_k^{'\mathrm{T}} F_k' v_{j+1} - \frac{1}{2} v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k' v_j\\ &= \lambda_j g_k^{\mathrm{T}} d_j + \frac{1}{2}\left(\sum_{i=1}^{j}\lambda_i d_i\right)^{\mathrm{T}} F_k^{'\mathrm{T}} F_k' \left(\sum_{i=1}^{j}\lambda_i d_i\right) - \frac{1}{2}\left(\sum_{i=1}^{j-1}\lambda_i d_i\right)^{\mathrm{T}} F_k^{'\mathrm{T}} F_k' \left(\sum_{i=1}^{j-1}\lambda_i d_i\right)\\ &= \lambda_j g_k^{\mathrm{T}} d_j + \frac{1}{2}\lambda_j^2\|w_j\|^2\\ &= \frac{1}{2}\lambda_j\left[2g_k^{\mathrm{T}} d_j + \theta_j^2 r_j^{\mathrm{T}} r_j\right].\end{aligned}$$

Noting

$$g_k^{\mathrm{T}}d_j + \theta_j^2 r_j^{\mathrm{T}} r_j = d_j^{\mathrm{T}} r_1 - \theta_j r_j^{\mathrm{T}}\left(-\theta_j r_j + \beta_{j-1}d_{j-1}\right) = d_j^{\mathrm{T}} r_1 - \theta_j r_j^{\mathrm{T}} d_j$$
$$= d_j^{\mathrm{T}}\left(r_1 - \theta_j r_j\right) = d_j^{\mathrm{T}}\left(r_1 - g_k - F_k^{'\mathrm{T}}F_k^{'}v_j\right) = -d_j^{\mathrm{T}}\sum_{i=1}^{j-1}\lambda_i F_k^{'\mathrm{T}}F_k^{'}d_i = 0$$

and $\theta_j^2 r_j^{\mathrm{T}} r_j \geqslant 0$, we get $g_k^{\mathrm{T}}d_j \leqslant 0$, so $2g_k^{\mathrm{T}}d_j + \theta_j^2 r_j^{\mathrm{T}} r_j < 0$, that is, $\psi_k(v_{j+1}) - \psi_k(v_j) \leqslant 0$. This completes the proof of this lemma. □

The following lemma shows the relation between the gradient g_k of the objective function and the step p_k generated by the proposed algorithm. We can see from the following lemma that the direction of the trial step is a sufficiently descent direction.

Assumption 3.2 Sequence $\{x_k\}$ generated by the algorithm is contained in the compact set $\mathcal{L}(x_0)$.

Assumption 3.3 $\|p_k\|$ and $F_k^{'\mathrm{T}}F_k^{'}$ are uniformly bounded, that is, there exist constants χ_p and χ satisfy $\|p_k\| \leqslant \chi_p$ and $\|F_k^{'\mathrm{T}}F_k^{'}\| \leqslant \chi$ for all k.

Lemma 3.4 *Consider the step $p_k = v_j$ be obtained from NCGL. Then*

(1) *$\{g_k^{\mathrm{T}}v_j\}$ is monotonically decreasing, that is, $g_k^{\mathrm{T}}v_{j+1} \leqslant g_k^{\mathrm{T}}v_j$, $1 \leqslant j \leqslant n_k$.*
(2) *$g_k^{\mathrm{T}}p_k$ satisfies the following sufficient descent condition*

$$g_k^{\mathrm{T}}p_k \leqslant -\min\left\{1, \frac{1}{\chi}\right\}\|g_k\|^2. \tag{3.1}$$

Proof

(1) From (2.10), the following is true:

$$g_k^{\mathrm{T}}v_{j+1} - g_k^{\mathrm{T}}v_j = g_k^{\mathrm{T}}(v_{j+1} - v_j) = \lambda_j g_k^{\mathrm{T}}d_j = -\lambda_j d_1^{\mathrm{T}}d_j \leqslant 0.$$

(2) If $\|w_1\| = 0$, then $p_k = v_1 = d_1$ and

$$g_k^{\mathrm{T}}p_k = g_k^{\mathrm{T}}d_1 = -g_k^{\mathrm{T}}g_k = -\|g_k\|^2 \leqslant -C_1\|g_k\|^2.$$

If $\|w_1\| > 0$, then there exists $j_0 \geqslant 2$ such that $p_k = v_{j_0}$. The results that $\{g_k^{\mathrm{T}}v_j\}$ is monotonically decreasing and $g_k^{\mathrm{T}}v_2 = g_k^{\mathrm{T}}(v_1 + \lambda_1 d_1) = \lambda_1 g_k^{\mathrm{T}}d_1 = -\lambda_1\|g_k\|^2$ yield:

$$g_k^{\mathrm{T}}p_k \leqslant g_k^{\mathrm{T}}v_2 = -\lambda_1\|g_k\|^2.$$

Assumption 3.3 shows $\lambda_1 = \frac{\theta_1^2\|r_1\|^2}{d_1^{\mathrm{T}} F_k'^{\mathrm{T}} F_k' d_1} \geqslant \frac{\|g_k\|^2}{\|g_k\|^2 \cdot \|F_k'^{\mathrm{T}} F_k'\|} \geqslant \frac{1}{\chi}$. Therefore,

$$g_k^{\mathrm{T}} p_k \leqslant -\frac{1}{\chi}\|g_k\|^2 \leqslant -\min\left\{1, \frac{1}{\chi}\right\}\|g_k\|^2.$$

□

Lemma 3.5 *The predicted reduction satisfies the estimate:*

$$f(x_k) - \psi_k(p_k) \geqslant \|g_k\|^2 \min\left\{1, \frac{1}{2\chi}\right\}. \tag{3.2}$$

Proof The proof is analogous to that of Lemma 3.4. If $\|w_1\| = 0$, then $p_k = v_1 = d_1$ and

$$f(x_k) - \psi_k(p_k) = -g_k^{\mathrm{T}} d_1 - \frac{1}{2}\|w_1\|^2 = -g_k^{\mathrm{T}} d_1 = \|g_k\|^2.$$

For the case of $\|w_1\| > 0$, Since $\{\psi_k(v_j)\}$ is monotonically decreasing, it follows that

$$\begin{aligned} f(x_k) - \psi_k(p_k) &\geqslant f(x_k) - \psi_k(\lambda_1 d_1) \\ &= -\lambda_1 g_k^{\mathrm{T}} d_1 - \frac{1}{2}\lambda_1^2\|w_1\|^2 = \lambda_1\|g_k\|^2 - \frac{\lambda_1}{2}\|g_k\|^2 \\ &= \frac{\lambda_1}{2}\|g_k\|^2 \geqslant \|g_k\|^2 \min\left\{1, \frac{1}{2\chi}\right\}. \end{aligned}$$

The conclusion of the lemma holds. □

We are now ready to state one of our main results of the proposed algorithm, which also needs the following assumptions.

Assumption 3.6 $g(x) = \nabla f(x)$ is Lipschitz continuous, that is, there exists a constant γ such that

$$\|g(x) - g(y)\| \leqslant \gamma\|x - y\| \forall x, y \in \mathcal{L}(x_0).$$

Assumption 3.7 $F'_* = F'(x_*)$ is nonsingular, where x_* is the limit point.

Theorem 3.8 *Assume that Assumptions 3.2, 3.3 and 3.6 hold. Let $\{x_k\} \subset \mathbb{R}^n$ be a sequence generated by NCGL. Then*

$$\liminf_{k\to\infty} \|F_k'^T F_k\| = 0. \tag{3.3}$$

Proof Taking into account that $m(k+1) \leqslant m(k) + 1$ and $f(x_{k+1}) \leqslant f(x_l(k))$, we get

$$f(x_{l(k+1)}) = \max_{0\leqslant j\leqslant m(k+1)} f(x_{k+1-j}) \leqslant \max_{0\leqslant j\leqslant m(k)+1} f(x_{k+1-j}) = f(x_{l(k)}).$$

This means $\{f(x_{l(k)})\}$ is nonincreasing for all k and hence $\{f(x_{l(k)})\}$ is convergent.

If the conclusion of the theorem is not true, there exists some $\varepsilon > 0$ such that

$$\|F_k^{\prime\mathrm{T}} F_k\| \geqslant \varepsilon.$$

From (2.4) and (3.1), we obtain

$$\begin{aligned} f(x_{l(k)}) &= f(x_{l(k)-1} + \alpha_{l(k)-1} p_{l(k)-1}) \\ &\leqslant f\left(x_{l(l(k)-1)}\right) + \beta \alpha_{l(k)-1} g_{l(k)-1}^{\mathrm{T}} p_{l(k)-1} \\ &\leqslant f\left(x_{l(l(k)-1)}\right) - \alpha_{l(k)-1} \beta \varepsilon^2 \min\left\{1, \frac{1}{\chi}\right\}. \end{aligned} \tag{3.4}$$

Since $\{f(x_{l(k)})\}$ is convergent, it follows from (3.4) that

$$\lim_{k\to\infty} \alpha_{l(k)-1} = 0, \tag{3.5}$$

Equation (3.5) and Assumption 3.3 imply $\lim_{k\to\infty} \alpha_{l(k)-1} \|p_{l(k)-1}\| = 0$. Analogous to the proof of theorem in [6], we have

$$\lim_{k\to\infty} f(x_k) = \lim_{k\to\infty} f(x_{l(k)}).$$

Similar to the proof of (3.5), we obtain

$$\lim_{k\to\infty} \alpha_k = 0. \tag{3.6}$$

The acceptance rule in step 7 yields

$$f\left(x_k + \frac{\alpha_k}{\omega} p_k\right) > f\left(x_{l(k)}\right) + \frac{\alpha_k}{\omega} \beta g_k^{\mathrm{T}} p_k \geqslant f(x_k) + \frac{\alpha_k}{\omega} \beta g_k^{\mathrm{T}} p_k, \tag{3.7}$$

On the other hand, by Taylor's Theorem and Assumption 3.6,

$$\begin{aligned} f\left(x_k + \frac{\alpha_k}{\omega} p_k\right) - f(x_k) &= \frac{\alpha_k}{\omega} g_k^{\mathrm{T}} p_k + \frac{\alpha_k}{\omega} \int_0^1 g\left(x_k + t\frac{\alpha_k}{\omega} p_k - g(x_k)\right)^{\mathrm{T}} p_k \mathrm{d}t \\ &\leqslant \frac{\alpha_k}{\omega} g_k^{\mathrm{T}} p_k + \frac{1}{2}\gamma \left(\frac{\alpha_k}{\omega}\right)^2 \|p_k\|^2, \end{aligned} \tag{3.8}$$

where γ is Lipschitz constant for $g(x)$. From (3.7) and (3.8), we have

$$\frac{\alpha_k}{\omega} g_k^{\mathrm{T}} p_k + \frac{1}{2}\gamma \left(\frac{\alpha_k}{\omega}\right)^2 \|p_k\|^2 > \beta \frac{\alpha_k}{\omega} g_k^{\mathrm{T}} p_k.$$

So

$$\alpha_k \geqslant \frac{2\omega(\beta - 1)}{\gamma \|p_k\|^2} g_k^{\mathrm{T}} p_k \geqslant \frac{2\omega(1-\beta)}{\gamma \chi_p^2} \min\left\{1, \frac{1}{\chi}\right\} \varepsilon^2 > 0. \tag{3.9}$$

The observation that $\lim_{k\to\infty} \alpha_k \geqslant \frac{2\omega(1-\beta)}{\gamma\chi_p^2} \min\left\{1, \frac{1}{\chi}\right\} \varepsilon^2 > 0$ contradicts (3.6). □

4 Properties of the Local Convergence

Theorem 3.8 indicates that at least one limit point of $\{x_k\}$ is a stationary point. In this section, we shall first extend this theorem to a stronger result and the local convergent rate.

Theorem 4.1 *Assume that Assumptions 3.2, 3.3 and 3.6 hold. Let $\{x_k\}$ be a sequence generated by Algorithm NCGL. Then*

$$\lim_{k\to+\infty} \|F_k^{'\mathrm{T}} F_k\| = 0. \tag{4.1}$$

Proof Assuming that the conclusion is not true, there is an $\varepsilon_1 \in (0,1)$ and a subsequence $\left\{(F'_{m_i})^{\mathrm{T}} F_{m_i}\right\}$ such that for all $m_i, i = 1, 2, \cdots$

$$\|(F'_{m_i})^{\mathrm{T}} F_{m_i}\| \geqslant \varepsilon_1.$$

Consider any index m_i such that $\|\nabla f_{m_i}\| \geqslant \varepsilon_1$. Assumption 3.6 implies

$$\|\nabla f(x) - \nabla f(x_{m_i})\| \leqslant \gamma \|x - x_{m_i}\|.$$

Defining the scalar $R = \frac{n-1}{nr}\varepsilon_1$ and the ball $\mathcal{B}(x_{m_i}, R) = \{x | \|x - x_{m_i}\| \leqslant R\}$, where n can be some very large integer. If $x \in \mathcal{B}(x_{m_i}, R)$, then

$$\begin{aligned}
\|\nabla f(x)\| &\geqslant \|\nabla f_{m_i}\| - \|\nabla f(x) - \nabla f_{m_i}\| \\
&\geqslant \varepsilon_1 - \gamma\|x - x_{m_i}\| \geqslant \varepsilon_1 - \frac{n-1}{n}\varepsilon_1 = \frac{1}{n}\varepsilon_1 = \varepsilon_2,
\end{aligned}$$

where $\varepsilon_2 = \frac{1}{n}\varepsilon_1$. If the entire sequence $\{x_k\}_{k\geqslant m_i}$ stays the ball $\mathcal{B}(x_{m_i}, R)$, we would have $\|\nabla f_k\| \geqslant \varepsilon_2 > 0$ for all $k \geqslant m_i$. The reasoning in the proof of Theorem 3.8 can be used to show that this scenario does not occur. Therefore, the sequence $\{x_k\}_{k\geqslant m_i}$ eventually leaves $\mathcal{B}(x_{m_i}, R)$, and there exists another subsequence $\{(F'_{n_i})^{\mathrm{T}} F_{n_i}\}$ such that

$$\|(F'_k)^{\mathrm{T}} F_k\| \geqslant \varepsilon_2, \text{ for } \; m_i \leqslant k < n_i$$

and

$$\|(F'_{n_i})^{\mathrm{T}} F_{n_i}\| \leqslant \varepsilon_2,$$

for an $\varepsilon_2 \in (0, \varepsilon_1)$.

Similar to the proof of Theorem 3.8, we have

$$\lim_{k\to\infty, m_i\leqslant k<n_i} f(x_{l(k)}) = \lim_{k\to\infty, m_i\leqslant k<n_i} f(x_k). \tag{4.2}$$

The acceptance rule in step 7 yields

$$f(x_{l(k)}) - f(x_k + \alpha_k p_k) \geqslant -\alpha_k \beta g_k^{\mathrm{T}} p_k \geqslant \alpha_k \beta \tau \varepsilon_2 C_1 \geqslant 0.$$

It follows from this that $\lim\limits_{k\to\infty, m_i\leqslant k<n_i} \alpha_k = 0$, which contradicts to (3.9). So (4.1) holds. □

The following theorem shows the convergence rate for the proposed algorithm.

Theorem 4.2 *Assume that* $F(x)$ *is twice continuously differentiable, Assumptions 3.2, 3.3, 3.6 and 3.7 hold and* $\{x_k\}$ *is a sequence produced by Algorithm NCGL which convergence to* x_* *. Then the convergence is superlinear. i.e.,*

$$\lim_{k\to\infty}\frac{\|x_{k+1}-x_*\|}{\|x_k-x_*\|}=0. \tag{4.3}$$

Proof From Lemma 2.1,

$$0=\theta_j r_j^{\mathrm{T}}\left(\sum_{i=1}^{j-1}\lambda_i d_i\right)=\theta_j r_j^{\mathrm{T}} v_j=\left(g_k+F_k^{'\mathrm{T}}F_k^{'}v_j\right)^{\mathrm{T}}v_j=g_k^{\mathrm{T}}v_j+v_j^{\mathrm{T}}F_k^{'\mathrm{T}}F_k^{'}v_j. \tag{4.4}$$

Assumption 3.7 implies $F_k^{'\mathrm{T}}F_k^{'}$ is positive definite uniformly for sufficiently large k, so

$$v_j^{\mathrm{T}}F_k^{'\mathrm{T}}F_k^{'}v_j\geqslant\zeta\|v_j\|^2, \tag{4.5}$$

where $\zeta>0$ is a constant. Equations (4.4) and (4.5) show

$$\zeta\|v_j\|^2\leqslant v_j^{\mathrm{T}}F_k^{'\mathrm{T}}F_k^{'}v_j=-g_k^{\mathrm{T}}v_j\leqslant\|g_k\|\cdot\|v_j\|.$$

It follows from Theorem 4.1 that

$$\|v_j\|\leqslant\frac{1}{\zeta}\|g_k\|\to 0.$$

Noting $F(x)$ is twice continuously differentiable and $F(x_*)=0$, we have

$$\begin{aligned}
&|\psi_k(v_j)-f(x_k+v_j)|\\
&=\left|g_k^{\mathrm{T}}v_j+\frac{1}{2}v_j^{\mathrm{T}}F_k^{'\mathrm{T}}F_k^{'}v_j-\left(g_k^{\mathrm{T}}v_j+\frac{1}{2}v_j^{\mathrm{T}}\nabla^2 f(x_k)v_j+o\left(\|v_j\|^2\right)\right)\right|\\
&=\left|\frac{1}{2}v_j^{\mathrm{T}}\left(F_k^{'\mathrm{T}}F_k^{'}-\nabla^2 f(x_k)\right)v_j-o\left(\|v_j\|^2\right)\right|\\
&=o\left(\|v_j\|^2\right).
\end{aligned}$$

Using (4.5), we can get

$$
\begin{aligned}
&f(x_k) - \psi_k(v_j) = -g_k^{\mathrm{T}} v_j - \frac{1}{2} v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} v_j \\
&= \left(-\theta_j r_j + F_k^{'\mathrm{T}} F_k^{'} v_j \right)^{\mathrm{T}} v_j - \frac{1}{2} v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} v_j \left(\text{ because } \theta_j r_j = \nabla \varphi_k(v_j) = g_k + F_k^{'\mathrm{T}} F_k^{'} v_j \right) \\
&= -\theta_j r_j^{\mathrm{T}} v_j + v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} v_j - \frac{1}{2} v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} v_j = -\theta_j r_j^{\mathrm{T}} \left(\sum_{i=0}^{j-1} \lambda_i d_i \right) + \frac{1}{2} v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} v_j \\
&= \frac{1}{2} v_j^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} v_j \geqslant \frac{\zeta}{2} \|v_j\|^2.
\end{aligned}
\tag{4.6}
$$

Therefore,

$$
\frac{f(x_k) - f(x_k + v_j)}{f(x_k) - \psi_k(v_j)} \geqslant 1 - \frac{o\left(\|v_j\|^2\right)}{f(x_k) - \psi_k(v_j)} \geqslant 1 - \frac{o\left(\|v_j\|^2\right)}{\frac{\zeta}{2}\|v_j\|^2} \to 1. \tag{4.7}
$$

The above inequality means that there exists $\xi \in (0, 1)$ such that

$$
f(x_k) - f(x_k + v_j) \geqslant \xi [f(x_k) - \psi_k(v_j)].
$$

So, each v_j generated by step 5 of the algorithm must satisfy (2.3) for sufficiently large k, from the algorithm, we can deduce that $p_k = -\left(F_k^{'\mathrm{T}} F_k^{'}\right)^{-1} g_k$.

Next, it will be proved that $p_k = -\left(F_k^{'\mathrm{T}} F_k^{'}\right)^{-1} g_k$ satisfies (2.4). Von Neumann Lemma yields $(F_k^{'\mathrm{T}} F_k^{'})^{-1}$ is bounded. Combining this result with Theorem 4.1, we can deduce

$$
\lim_{k \to \infty} \|p_k\| = 0.
$$

Because $f(x_k)$ is twice continuously differentiable, $g_k^{\mathrm{T}} p_k = -p_k^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} p_k$, by (4.5), we have that

$$
\begin{aligned}
f(x_k + p_k) &= f(x_k) + g_k^{\mathrm{T}} p_k + \frac{1}{2} p_k^{\mathrm{T}} \nabla^2 f(x_k) p_k + o\left(\|p_k\|^2\right) \\
&= f(x_k) + \beta g_k^{\mathrm{T}} p_k + \left(\frac{1}{2} - \beta\right) g_k^{\mathrm{T}} p_k + \frac{1}{2}\left(g_k^{\mathrm{T}} p_k + p_k^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} p_k\right) \\
&\quad + \frac{1}{2} p_k^{\mathrm{T}} \left[\left(F_k^{'\mathrm{T}} F_k^{'} + \sum_{i=1}^{n} \nabla^2 F_i(x_k) F_i(x_k) \right) - F_k^{'\mathrm{T}} F_k^{'} \right] p_k + o\left(\|p_k\|^2\right) \\
&\leqslant f(x_k) + \beta g_k^{\mathrm{T}} p_k - \left(\frac{1}{2} - \beta\right) p_k^{\mathrm{T}} F_k^{'\mathrm{T}} F_k^{'} p_k + o\left(\|p_k\|^2\right) \\
&\leqslant f(x_{l(k)}) + \beta g_k^{\mathrm{T}} p_k - \left(\frac{1}{2} - \beta\right) \zeta \|p_k\|^2 + o\left(\|p_k\|^2\right).
\end{aligned}
$$

So, the step size $\alpha_k = 1$ will be taken for sufficiently large k.

It follows from the above discussions that

$$x_{k+1} = x_k - (F_k^{'\mathrm{T}} F_k^{'})^{-1} g_k,$$

which implies that for sufficiently large k, the step becomes Newton or quasi-Newton step, so (4.3) holds. □

5 Numerical Experiments

In this section, we report some numerical experiments, all codes were written in MATLAB with double precision. In order to check effectiveness of the method, we select the parameters as following: $\varepsilon = 10^{-6}$, $\xi = 0.02$, $\beta = 0.4$, $\omega = 0.5$. Our numerical results are listed in Tables 1, 2, 3 and 4. In these tables, n means the number of variables, NF, NG, and NL stand for the number of function evaluations, gradient evaluations, and line search evaluations, respectively, M denotes the nonmonotonic parameter.

In Table 1, we test our proposed algorithm through 6 problems which are quoted from [14] and [13]. The results show that Algorithm NCGL is highly accurate.

In the next experiments, we compare Algorithm NCGL with three-term CG method (MDL) [18], inexact Newton method [6] and Trust region method (NSCTR) [19], respectively. The numerical results are listed in Tables 2, 3 and 4. NOI in Table 2 means the number of iterations, which is equivalent to NG. The computational experiments presented illustrate that in most cases, our algorithm needs fewer iterations. Therefore, we have better results than those reported.

Table 1 Numerical results

Problem	n	Initial point	NCGL					
			$M = 0$			$M = 3$		
			NG	NF	$f(x_*)$	NG	NF	$f(x_*)$
SC207	2	$(-1.2,1)$	7	9	0	5	5	1.2326×10^{-32}
SC208	2	$(-1.2,1)$	139	154	1.1359×10^{-28}	145	149	8.3816×10^{-29}
SC211	2	$(-1.2,1)$	339	358	1.6024×10^{-27}	87	91	5.3841×10^{-28}
Powell badly scaled	2	$(0,1)$	9	15	1.6713×10^{-4}	13	14	3.9459×10^{-17}
Reklaitis Ragsdell	2	$(8,9)$	10	10	2.1457×10^{-28}	10	10	2.1457×10^{-28}
Extended White & Holst	100	$(-1.2,1,\cdots,-1.2,1)$	553	564	-1.5179×10^{-19}	454	456	1.8288×10^{-23}

Table 2 A comparison of NCGL and MDL

Problem	n	MDL			NCGL					
					$M = 0$			$M = 3$		
		NOI	NF	*Gmin*	NG	NF	$f(x_*)$	NG	NF	$f(x_*)$
Extended Himmelblau	100	25	44	0.755 774e−06	7	13	1.577 7e−30	7	10	5.019 3e−24
Extended BD1	100	63	98	0.720 250e−06	13	16	5.088 2e−020	9	10	1.845 5e−023
Broyden tridiagonal	100	45	89	0.994 144e−06	5	5	5.643 923e−019	5	7	5.643 923e−019
DENSCHNF (CUTE)	100	21	35	0.719 620e−07	7	7	3.257 8e−020	7	7	3.257 8e−020
LIARWHD (CUTE)	100	51	89	0.360 859e−07	12	12	2.706 1e−026	12	12	2.706 1e−026
HIMMELBG (CUTE)	100	9	11	0.360 859e−07	20	22	2.886 6e−008	20	22	2.886 6e−008

Table 3 A comparison of NCGL and Inexact Newton method

Problem	n	Inexact Newton						NCGL					
		$M = 0$			$M = 10$			$M = 0$			$M = 10$		
		NL	NG	$f(x_*)$	NL	NG	$f(x_*)$	NL	NG	$f(x_*)$	NL	NG	$f(x_*)$
Rosenbrock	2	22	30	$<10^{-38}$	12	17	$<10^{-38}$	10	53	$-1.421\,1 \times 10^{-14}$	7	54	$-1.421\,1 \times 10^{-14}$
	10	39	47	$<10^{-38}$	30	31	$<10^{-38}$	7	30	$1.186\,5 \times 10^{-25}$	1	25	$2.983\,3 \times 10^{-19}$
	20	52	61	$<10^{-38}$	44	45	$<10^{-38}$	16	45	$6.142\,1 \times 10^{-27}$	1	40	$6.605\,3 \times 10^{-25}$
Wood	4	40	70	$<10^{-38}$	31	35	$<10^{-38}$	8	291	$1.774\,6 \times 10^{-25}$	1	268	$1.876\,2 \times 10^{-19}$
Powell	4	34	35	0.2×10^{-21}	34	35	0.2×10^{-21}	1	17	$8.649\,4 \times 10^{-10}$	1	17	$8.649\,4 \times 10^{-10}$
Cube	2	28	40	0.5×10^{-26}	11	17	0.2×10^{-33}	4	194	$9.719\,7 \times 10^{-21}$	4	101	$1.266\,5 \times 10^{-21}$
Trigonometric	20	6	8	$<10^{-38}$	6	8	$<10^{-38}$	1	5	$2.623\,8 \times 10^{-20}$	1	5	$2.623\,8 \times 10^{-20}$
	60	6	8	$<10^{-38}$	6	8	$<10^{-38}$	1	5	$1.209\,9 \times 10^{-20}$	1	5	$1.209\,9 \times 10^{-20}$

Table 4 A comparison of NSCTR and NCGL

Problem	n	NSCTR			NCGL ($M = 3$)		
		NG	NF	$f(x_*)$	NG	NF	$f(x_*)$
P1	100	13	16	5.328 164e−008	9	10	5.651 849e−025
P3	100	78	85	2.949 721e−007	19	19	8.437 139e−010
P7	100	112	121	4.134 380e−007	42	43	4.128 950e−021
P8	100	13	17	1.482 496e−009	2	2	8.146 554e−027
P9	100	10	11	1.544 086e−005	16	16	1.359 861e−009

Acknowledgments The authors would like to thank the referees for their valuable comments which greatly improved the presentation of this manuscript.

References

[1] Bulteau, J.P., Vial, JPh: Curvilinear path and trust region in unconstrained optimization, a convergence analysis. Math. Program. Study **30**, 82–101 (1987)
[2] Conn, A., Could, N., Toint, P.: Trust-Region Methods. SIAM, Philadelphia (2000)
[3] Dai, Y., Yuan, Y.: A nonlinear conjugate gradient method with a strong global convergence property. SIAM J. Optim. **10**(1), 177–182 (1999)
[4] Dai, Y., Kou, C.: A nonlinear conjugate gradient algorithm with an optimal property and an improved wolfe life search. SIAM J. Optim. **23**(1), 296–320 (2013)
[5] Gould, N.I.M., Lucidi, S., Roma, M., Toint, P.L.: Solving the trust-region subproblem using the Lanczos method. SIAM J. Optim. **9**(2), 504–525 (1999)
[6] Gripp, L., Lampariello, F., Lucidi, S.: A nonmonotone line search technique for Newton's methods. SIAM J. Numer. Anal. **23**, 707–716 (1986)
[7] Guan, Y., Zhu, D.: Solving the unconstrained nonlinear optimization using the Lanczos path method. Numer. Math. A J. Chin. Univ. **S1**, 46–51 (2005)
[8] Guo, P., Zhu, D.: A nonmonotonic interior point algorithm via optimal path for nonlinear optimization with bounds to variables. J. Shanghai Normal Univ. **33**(3), 23–29 (2004)
[9] Hager, W., Zhang, H.: A new conjugate gradient method with guaranteed descent and efficient line search. SIAM J. Optim. **16**(1), 170–192 (2005)
[10] Hager, W., Zhang, H.: A survey of nonlinear conjugate gradient methods. Pac. J. Optim. **2**, 35–58 (2006)
[11] Jia, C., Zhu, D.: An affine scaling interior algorithm via Lanczos path for solving bound-constrained nonlinear systems. Appl. Math. Comput. **195**, 558–575 (2008)
[12] Narushima, Y., Yabe, H., Ford, J.: A three-term conjugate gradient method with sufficient descent property and unconstrained optimization. SIAM J. Optim. **21**(1), 212–230 (2011)
[13] Neculai, A.: An unconstrained optimization test functions collection. Adv. Model. Optim. **20**(1), 147–161 (2008)
[14] Schittkowski, K.: More Test Examples for Nonlinear Programming Codes. Lecture Notes in Economics and Mathematical Systems. Springer, Berlin (1987)
[15] Sun, J.: A convergence proof for an affine-scaling algorithm for convex quadratic programming without nondegeneracy assumptions. Math. Program. **60**, 69–79 (1993)
[16] Sun, W., Yuan, Y.: Optimization Theory and Methods. Springer, Berlin (2006)
[17] Ye, Y.: On affine scaling algorithms for nonconvex quadratic programming. Math. Program. **56**, 285–300 (1992)
[18] Zhang, J., Xiao, Y., Wei Z.: Nonlinear conjugate gradient methods with sufficient descent condition for large scale unconstrained optimization, Math. Prob. Eng. 16 p (2009). doi:10.1155/2009/243290
[19] Zhou, Q., Zhou, F., Cao, F.: A nonmonotone trust region method based on simple conic models for unconstrained optimization. Appl. Math. Comput. **225**, 295–305 (2013)
[20] Zhu, D.: Curvilinear paths and trust region methods with nonmonotonic back tracking technique for unconstrained optimization. J. Comput. Math. **19**, 241–258 (2001)