Available online at www.sciencedirect.com
SCIENCE DIRECT
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
Journal of Computational and Applied Mathematics 193 (2006) 397–412
www.elsevier.com/locate/cam

Convergence of nonmonotone line search method☆

Zhen-Jun Shi[a, b, *], Jie Shen[c]

[a] College of Operations Research and Management, Qufu Normal University, Rizhao, Shandong 276826, PR China
[b] Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, P.O. Box 2719, Beijing 100080, PR China
[c] Department of Computer and Information Science, University of Michigan, Dearborn, MI 48128, USA

Received 12 April 2005; received in revised form 23 June 2005

Abstract

In this paper, we develop a new nonmonotone line search for general line search method and establish some global convergence theorems. The new nonmonotone line search is a novel form of the nonmonotone Armijo line search and allows one to choose a larger step size at each iteration, which is available in constructing new line search methods and possibly reduces the function evaluations at each iteration. Moreover, we analyze the convergence rate of some special line search methods with the new line search. Preliminary numerical results show that some line search methods with the new nonmonotone line search are available and efficient in practical computation.

MSC: 90C30; 65K05; 49M37

Keywords: Unconstrained optimization; General line search method; Nonmonotone line search; Convergence

1. Introduction

Consider an unconstrained minimization problem

$$\min f(x), \quad x \in R^n, \tag{1}$$

where R^n denotes an n-dimensional Euclidean space and $f : R^n \to R^1$ is a continuously differentiable function. There are many iterative schemes for solving (1). Among them the line search method has the form

$$x_{k+1} = x_k + \alpha_k d_k, \quad k = 0, 1, 2, \ldots, \tag{2}$$

where d_k is a descent direction of $f(x)$ at x_k and α_k is a step size. Denote x_0 the initial point and x_k the current iterate at the kth iteration. Generally, we denote $f(x_k)$ by f_k, $\nabla f(x_k)$ by g_k and $f(x^*)$ by f^*, respectively.

The search direction d_k is generally required to satisfy

$$g_k^{\mathrm{T}} d_k < 0, \tag{3}$$

☆ The work was supported in part by NSF of China under grant 10171054, Postdoctoral Foundation of China and K.-C. Wong Postdoctoral Foundation of CAS under grant 6765700.
* Corresponding author.
E-mail addresses: zjshi@lsec.cc.ac.cn, zjshi@qrnu.edu.cn (Z.-J. Shi), shen@umich.edu (J. Shen).

0377-0427/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2005.06.033

which guarantees that d_k is a descent direction of $f(x)$ at x_k [10,11,18]. In order to guarantee the global convergence of the scheme (2), we sometimes expect that d_k satisfies the sufficient descent condition

$$g_k^{\mathrm{T}} d_k \leqslant -c\|g_k\|^2, \tag{4}$$

where $c > 0$ is a constant.

Moreover, the angle property

$$\cos\langle -g_k, d_k\rangle = -\frac{g_k^{\mathrm{T}} d_k}{\|g_k\| \cdot \|d_k\|} \geqslant \tau, \tag{5}$$

is commonly used in many situations, where $\tau: \ 0 < \tau \leqslant 1$ is a constant and $\langle -g_k, d_k\rangle$ denotes the angle of $-g_k$ and d_k.

In line search methods, if the search direction d_k is given at the kth iteration then the next task is to find a step size α_k along the search direction. The ideal line search rule is the exact one that satisfies

$$f(x_k + \alpha_k d_k) = \min_{\alpha > 0} f(x_k + \alpha d_k). \tag{6}$$

In fact, the exact step size is difficult or even impossible to seek in practical computation, and thus many researchers constructed some inexact line search rules, such as Armijo rule, Goldstein rule and Wolfe rule (see [1,4,17]). Here we introduce the Armijo rule.

Armijo line search rule: Given $s > 0$, $\beta \in (0, 1)$ and $\sigma \in (0, 1)$, α_k is the largest α in $\{s, s\beta, s\beta^2, \ldots\}$ such that

$$f(x_k + \alpha d_k) - f_k \leqslant \sigma\alpha g_k^{\mathrm{T}} d_k. \tag{7}$$

How to choose the parameters (such as s, σ, β) in line search methods is very important in solving practical problems. Several approaches for selecting them have been introduced in many literatures (e.g. [13,15,16]). Many inexact line search rules satisfy the descent property of objective function values, i.e.,

$$f_{k+1} < f_k, \quad \forall k \geqslant 1.$$

As a result, we call these methods monotone line search methods.

As to nonmonotone line search methods, the above descent property is not guaranteed. However, the nonmonotone line search rules are effective or even powerful at some iterations, especially when the iterates are trapped in a narrow curved valley of objective functions. Since Grippo, Lampariello, and Lucidi proposed the nonmonotone line search rule for Newton methods, the new line search approach has been studied by many authors (e.g. [3,5–7]). Some abstract nonmonotone line search rules have been proposed. See [8].

Nonmonotone Armijo rule is stated as follows.

Nonmonontone Armijo rule: Let $s > 0$, $\sigma \in (0, 1)$, $\beta \in (0, 1)$ and let M'' be a nonnegative integer. For each k, let $m(k)$ satisfy

$$m(0) = 0, \quad 0 \leqslant m(k) \leqslant \min[m(k-1), M''], \quad \forall k \geqslant 1. \tag{8}$$

Let α_k be the largest α in $\{s, s\beta, s\beta^2, \ldots\}$ such that

$$f(x_k + \alpha d_k) \leqslant \max_{0 \leqslant j \leqslant m(k)} [f(x_{k-j})] + \sigma\alpha g_k^{\mathrm{T}} d_k. \tag{9}$$

Because the nonmonotone line search rule has many advantages, especially in the case of iterates trapped in a narrow curved valley of objective functions (e.g.[8,19]). This approach seems to avoid the local minima and saddle points of objective functions, so that it seems to play a role in finding global minima of optimization problems. However, the above nonmonotone Armijo rule has two drawbacks. One is that the initial test step size s is a constant, and cannot be adjusted according to the characteristics of objective functions. The other drawback is that the nonmonotone Armijo rule does not use the information of second order derivatives of objective functions, especially the information of the quasi-Newton matrix B_k that approximates the Hessian $\nabla^2 f(x_k)$ of $f(x)$ at x_k.

In this paper, we propose a new nonmonotone line search for general line search method and develop some global convergence properties. The new line search is a novel scheme of the nonmonotone Armijo line search and allows one to find a larger accepted step size and possibly reduces the function evaluations at each iteration. Moreover, we analyze the convergence rate of some special methods with the new line search. Preliminary numerical results show that some line search methods with the new line search are available and efficient in practical computation.

The rest of this paper is organized as follows. In the next section, we describe the new nonmonotone line search and introduce the related line search method. In Sections 3 and 4 we analyze its convergence and convergence rate, respectively. In Sections 5 and 6 we report some numerical results and give a conclusion.

2. New nonmonotone line search

We first assume that

(H1): The objective function $f(x)$ has a lower bound on R^n.

(H2): Given $x_0 \in R^n$, the gradient $g(x)$ of $f(x)$ is Lipschitz continuous on an open convex set B that contains the level set $L_0 = \{x \in R^n \mid f(x) \leqslant f(x_0)\}$, i.e., there exists a constant $L > 0$ such that

$$\|g(x) - g(y)\| \leqslant L\|x - y\|, \quad \forall x, y \in B. \tag{10}$$

Sometimes we require that $f(x)$ is twice continuously differentiable. In what follows, we first describe the new nonmonotone line search.

New nonmonotone line search (*NNLS*): Let M'' be a nonnegative integer. For each k, let $m(k)$ satisfy (8). Given $\beta \in (0, 1)$, $\sigma \in (0, \frac{1}{2})$ and $\delta \in [0.5, 2)$, B_k is a symmetric positive definite matrix that approximates the Hessian of $f(x)$ at the iterate x_k and

$$s_k = -\frac{\delta g_k^{\mathrm{T}} d_k}{d_k^{\mathrm{T}} B_k d_k}.$$

Choose α_k to be the largest α in $\{s_k, s_k\beta, s_k\beta^2, \ldots\}$ such that

$$f(x_k + \alpha d_k) - \max_{0 \leqslant j \leqslant m(k)} [f_{k-j}] \leqslant \sigma\alpha[g_k^{\mathrm{T}} d_k + \tfrac{1}{2}\alpha d_k^{\mathrm{T}} B_k d_k]. \tag{11}$$

Remark 2.1. The above NNLS has two advantages compared with the original nonmonotone Armijo line search. One is that the initial test step size s may be adjusted automatically at different iterates, i.e., s_k varies with k at each step. In numerical experiments, we will see that taking s_k as an initial trial step size at the kth step is a reasonable and useful choice. If we let the parameters $(\sigma, \beta, s_k = s)$ take the same values in the NNLS and the original nonmonotone Armijo line search, let α_k and α_k' denote the step sizes defined by the former and latter line searches, respectively, we can easily see that

$$\alpha_k \geqslant \alpha_k', \quad \forall k.$$

This shows that one can choose a larger accepted step size in the NNLS than in the original nonmonotone Armijo line search. In other words, α_k is easier to seek than α_k' in practical computation. As a result, the function evaluations at each step may be reduced by using the NNLS.

The other advantage is that B_k can be modified by means of quasi-Newton formulae such as, BFGS, DFP, and other formulae. This enables us to use quasi-Newton formulae to modify the nonmonotone line search, and improve the numerical performance of the resultant line search algorithms. In practical computation, if $d_k^{\mathrm{T}} B_k d_k \leqslant 0$ then we take the first integer $i > -d_k^{\mathrm{T}} B_k d_k / \|d_k\|^2$ and set $B_k := B_k + iI$, where I is the $n \times n$ identity matrix.

In order to solve large scale unconstrained optimization problems, we may take $d_k = -g_k$ and $B_k = L_k I$ (I denotes $n \times n$ unit matrix) in some nonmonotone line search algorithms to avoid the storage and computation of some matrices [14]. In this case,

$$s_k = -\frac{\delta g_k^{\mathrm{T}} d_k}{d_k^{\mathrm{T}} B_k d_k} = \frac{\delta}{L_k}.$$

We can estimate L_k by using some approaches. For example, if $k>1$ then let $\delta_{k-1}=x_k-x_{k-1}$, $\gamma_k=g_k-g_{k-1}$ and let

$$L_k=\frac{\delta_k^{\mathrm{T}}\gamma_k}{\|\delta_k\|^2},$$

which is a solution to the minimization problem

$$\min_{L\in R^1}\ \|(LI)\delta_k-\gamma_k\|.$$

The new related algorithm is introduced as follows.

Algorithm (A). *Step* 0. Given a nonnegative integer $M''\geqslant 0$, $\beta\in(0,1)$, $\sigma\in(0,\frac{1}{2})$, $\delta\in[0.5,2)$, $x_0\in R^n$, and a symmetric positive definite matrix B_0, set $k:=0$;

Step 1. If $\|g_k\|=0$ then stop else go to Step 2;

Step 2. $x_{k+1}=x_k+\alpha_k d_k$, where d_k is a descent direction, for example, (3) holds; set $i>-d_k^{\mathrm{T}}B_kd_k/\|d_k\|^2$, $B_k:=B_k+iI$ and α_k is defined by the NNLS;

Step 3. Let $\delta_k=x_{k+1}-x_k$, $\gamma_k=g_{k+1}-g_k$, and modify B_k as B_{k+1} by using BFGS or DFP formula or other quasi-Newton formulae;

Step 4. Set $k:=k+1$ and go to step 1.

3. Global convergence

In order to establish the global convergence, we further assume that

(H3): There exist $M\geqslant m>0$ such that, for any k,

$$m\|d_k\|^2\leqslant d_k^{\mathrm{T}}B_kd_k\leqslant M\|d_k\|^2. \tag{12}$$

Lemma 3.1. *Assume that* (H1), (H2) *and* (H3) *hold*, d_k *satisfies* (3). *Algorithm* (*A*) *generates an infinite sequence* $\{x_k\}$. *Then*, *there exists a constant* $\eta_0>0$ *such that*

$$\max_{0\leqslant j\leqslant m(k)}[f_{k-j}]-f_{k+1}\geqslant\eta_0\left(\frac{g_k^{\mathrm{T}}d_k}{\|d_k\|}\right)^2,\quad\forall k. \tag{13}$$

Proof. Let $K_1=\{k|\ \alpha_k=s_k\}$ and $K_2=\{k|\ \alpha_k<s_k\}$. The proof is divided into two parts.

At first, if $k\in K_1$ then, by (11) and (12), we have

$$\begin{aligned}\max_{0\leqslant j\leqslant m(k)}[f_{k-j}]-f_{k+1}&\geqslant-\alpha_k\sigma[g_k^{\mathrm{T}}d_k+\tfrac{1}{2}\alpha_kd_k^{\mathrm{T}}B_kd_k]\\&=\sigma\frac{\delta g_k^{\mathrm{T}}d_k}{d_k^{\mathrm{T}}B_kd_k}\left[g_k^{\mathrm{T}}d_k-\frac{1}{2}\cdot\frac{\delta g_k^{\mathrm{T}}d_k}{d_kB_kd_k}\cdot d_k^{\mathrm{T}}B_kd_k\right]\\&=\frac{\sigma\delta(2-\delta)}{2}\cdot\frac{(g_k^{\mathrm{T}}d_k)^2}{d_kB_kd_k}\\&\geqslant\frac{\sigma\delta(2-\delta)}{2M}\left(\frac{g_k^{\mathrm{T}}d_k}{\|d_k\|}\right)^2,\quad\forall k\in K_1.\end{aligned}$$

Thus

$$\max_{0\leqslant j\leqslant m(k)}[f_{k-j}]-f_{k+1}\geqslant\frac{\sigma\delta(2-\delta)}{2M}\left(\frac{g_k^{\mathrm{T}}d_k}{\|d_k\|}\right)^2,\quad\forall k\in K_1. \tag{14}$$

Second, if $k \in K_2$, since $\alpha_k < s_k$, we have $\alpha_k\beta^{-1} \in \{s_k, s_k\beta, s_k\beta^2, \ldots\}$. Let $\alpha = \alpha_k\beta^{-1}$, we can deduce that (11) does not hold, i.e.,

$$\max_{0 \leqslant j \leqslant m(k)} [f_{k-j}] - f(x_k + \alpha_k\beta^{-1}d_k) < -\sigma\alpha_k\beta^{-1}[g_k^{\mathrm{T}}d_k + \tfrac{1}{2}\alpha_k\beta^{-1}d_kB_kd_k], \quad k \in K_2,$$

thus

$$f_k - f(x_k + \alpha_k\beta^{-1}d_k) < -\sigma\alpha_k\beta^{-1}[g_k^{\mathrm{T}}d_k + \tfrac{1}{2}\alpha_k\beta^{-1}d_kB_kd_k], \quad k \in K_2.$$

Using the mean value theorem on the left-hand side of the above inequality, there exists $\theta_k \in [0, 1]$ such that

$$-\alpha_k\beta^{-1}g(x_k + \alpha_k\beta^{-1}\theta_kd_k)^{\mathrm{T}}d_k < -\sigma\alpha_k\beta^{-1}[g_k^{\mathrm{T}}d_k + \tfrac{1}{2}\alpha_k\beta^{-1}d_kB_kd_k], \quad k \in K_2.$$

Dividing by $\alpha_k\beta^{-1}$ on the two sides of the above inequality, we obtain

$$g(x_k + \alpha_k\beta^{-1}\theta_kd_k)^{\mathrm{T}}d_k > \sigma\left[g_k^{\mathrm{T}}d_k + \tfrac{1}{2}\alpha_k\beta^{-1}d_kB_kd_k\right], \quad k \in K_2. \tag{15}$$

By (H2), Cauchy–Schwarz inequality, and (15), we have

$$\begin{aligned}
\alpha_k\beta^{-1}L\|d_k\|^2 &\geqslant \|g(x_k + \theta_k\alpha_k\beta^{-1}d_k) - g_k\| \cdot \|d_k\| \\
&\geqslant (g(x_k + \theta_k\alpha_k\beta^{-1}d_k) - g_k)^{\mathrm{T}}d_k \\
&\geqslant \sigma[g_k^{\mathrm{T}}d_k + \tfrac{1}{2}\alpha_k\beta^{-1}d_kB_kd_k] - g_k^{\mathrm{T}}d_k \\
&> \sigma g_k^{\mathrm{T}}d_k - g_k^{\mathrm{T}}d_k \\
&= -(1-\sigma)g_k^{\mathrm{T}}d_k, \quad k \in K_2,
\end{aligned}$$

thus

$$\alpha_k > \frac{\beta(1-\sigma)}{L}\left(-\frac{g_k^{\mathrm{T}}d_k}{\|d_k\|^2}\right).$$

Let

$$\alpha_k' = \frac{\beta(1-\sigma)}{L}\left(-\frac{g_k^{\mathrm{T}}d_k}{\|d_k\|^2}\right),$$

we have

$$\alpha_k' < \alpha_k < s_k, \quad \forall k \in K_2. \tag{16}$$

By (16) we have

$$d_k^{\mathrm{T}}B_kd_k < \frac{\delta L}{\beta(1-\sigma)} \cdot \|d_k\|^2, \quad k \in K_2. \tag{17}$$

By (11), (16) and (17), for $k \in K_2$, it holds that

$$
\begin{aligned}
\max_{0 \leqslant j \leqslant m(k)} [f_{k-j}] - f_{k+1} &\geqslant -\sigma\alpha_k[g_k^{\mathrm{T}} d_k + \tfrac{1}{2}\alpha_k d_k B_k d_k] \\
&\geqslant -\sigma \max_{\alpha_k' \leqslant \alpha \leqslant s_k} \{\alpha[g_k^{\mathrm{T}} d_k + \tfrac{1}{2}\alpha d_k^{\mathrm{T}} B_k d_k]\} \\
&\geqslant -\sigma\alpha_k' g_k^{\mathrm{T}} d_k - \tfrac{1}{2}\sigma s_k^2 d_k B_k d_k \\
&= \frac{\sigma\beta(1-\sigma)(g_k^{\mathrm{T}} d_k)^2}{L\|d_k\|^2} - \frac{\sigma\delta^2(g_k^{\mathrm{T}} d_k)^2}{2d_k^{\mathrm{T}} B_k d_k} \\
&= \frac{\sigma\beta(1-\sigma)(g_k^{\mathrm{T}} d_k)^2}{L\|d_k\|^2} - \frac{\sigma\delta\beta(1-\sigma)(g_k^{\mathrm{T}} d_k)^2}{2L\|d_k\|^2} \\
&= \frac{\beta\sigma(1-\sigma)(2-\delta)}{2L}\left(\frac{g_k^{\mathrm{T}} d_k}{\|d_k\|}\right)^2, \quad \forall k \in K_2.
\end{aligned}
$$

Therefore

$$
\max_{0 \leqslant j \leqslant m(k)} [f_{k-j}] - f_{k+1} \geqslant \frac{\beta\sigma(1-\sigma)(2-\delta)}{2L}\left(\frac{g_k^{\mathrm{T}} d_k}{\|d_k\|}\right)^2, \quad \forall k \in K_2. \tag{18}
$$

By combining (14) and (18), and by letting

$$
\eta_0 = \min\left(\frac{\sigma\delta(2-\delta)}{2M}, \frac{\beta\sigma(1-\sigma)(2-\delta)}{2L}\right),
$$

we can prove the truth of (13). □

Corollary 3.1. *If the conditions of Lemma* 3.1 *hold and d_k satisfies* (4), *then*

$$
\max_{0 \leqslant j \leqslant m(k)} [f_{k-j}] - f_{k+1} \geqslant \eta_1 \frac{\|g_k\|^4}{\|d_k\|^2}, \quad \forall k, \tag{19}
$$

where $\eta_1 = \eta_0 c^2$.

Corollary 3.2. *If the conditions of Lemma* 3.1 *hold and d_k satisfies* (5), *then*

$$
\max_{0 \leqslant j \leqslant m(k)} [f_{k-j}] - f_{k+1} \geqslant \eta \|g_k\|^2, \quad \forall k, \tag{20}
$$

where $\eta = \eta_0\tau^2$ and η_0 is defined as in the proof of Lemma 3.1.

Lemma 3.2. *If the conditions of Lemma* 3.1 *hold*, *then*,

$$
\max_{1 \leqslant j \leqslant M''} [f(x_{M''l+j})] \leqslant \max_{1 \leqslant i \leqslant M''} [f(x_{M''(l-1)+i})] - \eta_0 \min_{1 \leqslant j \leqslant M''} \left(\frac{g_{M''l+j-1}^{\mathrm{T}} d_{M''l+j-1}}{\|d_{M''l+j-1}\|}\right)^2 \tag{21}
$$

and

$$
\sum_{l=1}^{\infty} \min_{1 \leqslant j \leqslant M''} \left(\frac{g_{M''l+j-1}^{\mathrm{T}} d_{M''l+j-1}}{\|d_{M''l+j-1}\|}\right)^2 < +\infty. \tag{22}
$$

Proof. By (H1) and Lemma 3.1, it suffices to show that the following inequality holds for $j = 1, 2, \ldots, M''$,

$$
f(x_{M''l+j}) \leqslant \max_{1 \leqslant i \leqslant M''} [f(x_{M''(l-1)+i})] - \eta_0 \left(\frac{g_{M''l+j-1}^{\mathrm{T}} d_{M''l+j-1}}{\|d_{M''l+j-1}\|}\right)^2. \tag{23}
$$

Since the NNLS and Lemma 3.1 imply

$$f(x_{M''l+1}) \leqslant \max_{0 \leqslant i \leqslant m(M''l)} f(x_{M''l-i}) - \eta_0 \left(\frac{g_{M''l}^{\mathrm{T}} d_{M''l}}{\|d_{M''l}\|} \right)^2, \tag{24}$$

it follows from this and

$$0 \leqslant m(M''l) \leqslant M''$$

that (23) holds for $j = 1$. Suppose that (23) holds for any $j : 1 \leqslant j \leqslant M'' - 1$. With the descent property of d_k, this implies

$$\max_{1 \leqslant i \leqslant j} [f(x_{M''l+i})] \leqslant \max_{1 \leqslant i \leqslant M''} [f(x_{M''(l-1)+i})]. \tag{25}$$

By the NNLS, the induction hypothesis,

$$m(M''l + j) \leqslant M'',$$

Lemma 3.1and (25), we obtain

$$\begin{aligned} f(x_{M''l+j+1}) &\leqslant \max_{0 \leqslant i \leqslant m(M''l+j)} [f(x_{M''l+j-i})] - \eta_0 \left(\frac{g_{M''l+j}^{\mathrm{T}} d_{M''l+j}}{\|d_{M''l+j}\|} \right)^2 \\ &\leqslant \max \left\{ \max_{1 \leqslant i \leqslant M''} f(x_{M''(l-1)+i}), \max_{1 \leqslant i \leqslant j} f(x_{M''l+i}) \right\} - \eta_0 \left(\frac{g_{M''l+j}^{\mathrm{T}} d_{M''l+j}}{\|d_{M''l+j}\|} \right)^2 \\ &\leqslant \max_{1 \leqslant i \leqslant M''} [f(x_{M''(l-1)+i})] - \eta_0 \left(\frac{g_{M''l+j}^{\mathrm{T}} d_{M''l+j}}{\|d_{M''l+j}\|} \right)^2. \end{aligned}$$

Thus, (23) is also true for $j + 1$. By induction, (23) holds for $1 \leqslant j \leqslant M''$. This shows that (21) holds.

Since $f(x)$ is bounded from below by (H1), it follows that

$$\max_{1 \leqslant i \leqslant M''} [f(x_{M''l+i})] > -\infty.$$

By summing (23) over l, we can get

$$\sum_{l=1}^{\infty} \min_{1 \leqslant j \leqslant M''} \left(\frac{g_{M''l+j-1}^{\mathrm{T}} d_{M''l+j-1}}{\|d_{M''l+j-1}\|} \right)^2 < +\infty.$$

Therefore (22) holds. The proof is complete. □

Corollary 3.3. *If the conditions of Lemma* 3.2 *hold and d_k satisfies* (4), *then*

$$\max_{1 \leqslant j \leqslant M''} [f(x_{M''l+j})] \leqslant \max_{1 \leqslant i \leqslant M''} [f(x_{M''(l-1)+i})] - \eta_1 \min_{1 \leqslant j \leqslant M''} \frac{\|g_{M''l+j-1}\|^4}{\|d_{M''l+j-1}\|^2}, \tag{26}$$

and

$$\sum_{l=1}^{\infty} \min_{1 \leqslant j \leqslant M''} \frac{\|g_{M''l+j-1}\|^4}{\|d_{M''l+j-1}\|^2} < +\infty, \tag{27}$$

and thus

$$\lim_{l \to \infty} \min_{1 \leqslant j \leqslant M''} \frac{\|g_{M''l+j-1}\|^4}{\|d_{M''l+j-1}\|^2} = 0, \tag{28}$$

where $\eta_1 = \eta_0 c^2$.

Corollary 3.4. *If the conditions of Lemma* 3.2 *hold and d_k satisfies* (5), *then*

$$\max_{1\leqslant j\leqslant M''}[f(x_{M''l+j})]\leqslant \max_{1\leqslant i\leqslant M''}[f(x_{M''(l-1)+i})]-\eta \min_{1\leqslant j\leqslant M''}\|g_{M''l+j-1}\|^2, \tag{29}$$

and

$$\sum_{l=1}^{\infty}\min_{1\leqslant j\leqslant M''}\|g_{M''l+j-1}\|^2<+\infty, \tag{30}$$

and thus

$$\lim_{l\to\infty}\min_{1\leqslant j\leqslant M''}\|g_{M''l+j-1}\|=0, \tag{31}$$

where $\eta=\eta_0\tau^2$.

Theorem 3.1. *If the conditions of Lemma* 3.2 *hold and d_k satisfies* (5), *then*

$$\lim_{k\to\infty}\|g_k\|=0. \tag{32}$$

Proof. By the NNLS, (H3) and Cauchy–Schwarz inequality, we have

$$\begin{aligned}\alpha_k\leqslant s_k&=-\frac{\delta g_k^{\mathrm T}d_k}{d_k^{\mathrm T}B_kd_k}\\&\leqslant-\frac{\delta g_k^{\mathrm T}d_k}{m\|d_k\|^2}\leqslant\frac{\delta\|g_k\|}{m\|d_k\|}.\end{aligned}$$

By the above inequality, (H2) and (H3), it holds that

$$\begin{aligned}\|g_{k+1}\|&=\|g_{k+1}-g_k+g_k\|\\&\leqslant\|g_{k+1}-g_k\|+\|g_k\|\\&\leqslant L\alpha_k\|d_k\|+\|g_k\|\\&\leqslant\left(1+\frac{\delta L}{m}\right)\|g_k\|.\end{aligned}$$

Let

$$c_3=1+\frac{\delta L}{m},$$

it follows that

$$\|g_{k+1}\|\leqslant c_3\|g_k\|. \tag{33}$$

Let

$$\|g_{M''l+\phi(l)}\|=\min_{0\leqslant i\leqslant M''-1}\|g_{M''l+i}\|.$$

By Corollary 3.4 we obtain

$$\lim_{l\to\infty}\|g_{M''l+\phi(l)}\|=0, \tag{34}$$

where

$$0\leqslant\phi(l)\leqslant M''-1.$$

By (33), we have

$$\|g_{M''(l+1)+i}\|\leqslant c_3^{2M''}\|g_{M''l+\phi(l)}\|,\quad i=0,\ldots,M''-1. \tag{35}$$

Therefore, it follows from (34) and (35) that (32) holds. □

4. Convergence rate

In order to analyze the convergence rate, we confine our discussion to the case of uniformly convex objective functions. We further assume that
(H4): $f(x)$ is twice continuously differentiable and uniformly convex on R^n.

Lemma 4.1. *Assume that* (H4) *holds, then* (H1) *and* (H2) *hold. Moreover, $f(x)$ has a unique minimizer x^*, and there exists $0 < m' \leqslant M'$ such that*

$$m'\|y\|^2 \leqslant y^{\mathrm{T}}\nabla^2 f(x)y \leqslant M'\|y\|^2, \quad \forall x, y \in R^n; \tag{36}$$

$$\tfrac{1}{2}m'\|x-x^*\|^2 \leqslant f(x) - f(x^*) \leqslant \tfrac{1}{2}M'\|x-x^*\|^2, \quad \forall x \in R^n; \tag{37}$$

$$M'\|x-y\|^2 \geqslant (g(x)-g(y))^{\mathrm{T}}(x-y) \geqslant m'\|x-y\|^2, \quad \forall x, y \in R^n; \tag{38}$$

and thus

$$M'\|x-x^*\|^2 \geqslant g(x)^{\mathrm{T}}(x-x^*) \geqslant m'\|x-x^*\|^2, \quad \forall x \in R^n. \tag{39}$$

By (39) *and* (38) *we can also obtain, from the Cauchy–Schwarz inequality, that*

$$M'\|x-x^*\| \geqslant \|g(x)\| \geqslant m'\|x-x^*\|, \quad \forall x \in R^n, \tag{40}$$

and

$$\|g(x)-g(y)\| \leqslant M'\|x-y\|, \quad \forall x, y \in R^n. \tag{41}$$

By (40) *and* (37) *we can also obtain the following relation*

$$\frac{m'}{2M'^2}\|g(x)\|^2 \leqslant f(x) - f(x^*) \leqslant \frac{M'}{2m'^2}\|g(x)\|^2. \tag{42}$$

Its proof can be seen from the literature (e.g. [10,11]).

4.1. Linear convergence rate

Lemma 4.2. *If the assumptions* (H3) *and* (H4) *hold, d_k satisfies* (5) *and Algorithm* (*A*) *generates an infinite sequence $\{x_k\}$, then there exist constants c_4 and $c_5 \in (0, 1)$ such that*

$$f_k - f^* \leqslant c_4 c_5^k [f_0 - f^*]. \tag{43}$$

Proof. Similar proof can be seen from the literature [3,19]. □

Theorem 4.1. *If the assumptions* (H3) *and* (H5) *hold, d_k satisfies* (5) *and Algorithm* (*A*) *generates an infinite sequence $\{x_k\}$, then $\{x_k\}$ converges to the unique minimizer x^* of $f(x)$ at least R-linearly.*

Proof. By Lemma 4.1 and Theorem 3.1, we can prove

$$\lim_{k\to\infty} x_k = x^*.$$

By (37) and (43) we have

$$\begin{aligned}\|x_k - x^*\|^2 &\leqslant \frac{2}{m'}[f_k - f^*] \\ &\leqslant 2c_4c_5^k[f_0 - f^*]/m' \\ &= \frac{2c_4[f_0 - f^*]}{m'} \cdot (\sqrt{c_5})^{2k}.\end{aligned}$$

Therefore,

$$\begin{aligned}\lim_{k\to\infty} \|x_k - x^*\|^{1/k} &\leqslant \lim_{k\to\infty} \sqrt{c_5}\cdot\left(\sqrt{\frac{2c_4[f_0 - f^*]}{m'}}\right)^{1/k}\\ &= \sqrt{c_5}\\ &< 1,\end{aligned}$$

which shows that $\{x_k\}$ converges to x^* at least R-linearly. $\square$

4.2. Superlinear convergence rate

We further assume that

(H5): B_k and d_k generated by Algorithm (A) satisfy the following condition

$$\lim_{k\to\infty} \frac{\|[B_k - \nabla^2 f(x^*)]d_k\|}{\|d_k\|} = 0, \tag{44}$$

and $\delta = 1$ in Algorithm (A).

Lemma 4.3. *If* (H3), (H4) *and* (H5) *hold*, $d_k = -B_k^{-1}g_k$ *and Algorithm* (*A*) *generates an infinite sequence* $\{x_k\}$, *then there exists* k' *such that*

$$\alpha_k = s_k = 1, \quad \forall k \geqslant k'. \tag{45}$$

Proof. If $d_k = -B_k^{-1}g_k$ and $\delta = 1$ then

$$s_k = -\frac{g_k^{\mathrm{T}} d_k}{d_k^{\mathrm{T}} B_k d_k} = 1.$$

By Theorem 3.1 and (H3) we have

$$\lim_{k\to\infty} \|d_k\| = 0. \tag{46}$$

Assumption (H5) implies that

$$d_k^{\mathrm{T}}[B_k - \nabla^2 f(x^*)]d_k = \mathrm{o}(\|d_k\|^2). \tag{47}$$

By the mean value theorem, (H3), (46) and (47), for sufficiently large k, we have

$$\begin{aligned}f(x_k + d_k) - f_k &= g_k^{\mathrm{T}} d_k + \int_0^1 (1-t) d_k^{\mathrm{T}} \nabla^2 f(x_k + t d_k) d_k \,\mathrm{d}t\\ &= [g_k^{\mathrm{T}} d_k + \tfrac{1}{2} d_k B_k d_k] + \int_0^1 (1-t) d_k^{\mathrm{T}}[\nabla^2 f(x_k + t d_k) - \nabla^2 f(x^*)] d_k \,\mathrm{d}t\\ &\quad + \frac{1}{2} d_k^{\mathrm{T}}[\nabla^2 f(x^*) - B_k] d_k\\ &= [g_k^{\mathrm{T}} d_k + \tfrac{1}{2} d_k B_k d_k] + \mathrm{o}(\|d_k\|^2)\\ &\leqslant \sigma[g_k^{\mathrm{T}} d_k + \tfrac{1}{2} d_k B_k d_k].\end{aligned}$$

Thus

$$f(x_k + d_k) \leqslant \max_{0\leqslant j\leqslant M'} [f_{k-j}] + \sigma[g_k^{\mathrm{T}} d_k + \tfrac{1}{2}\, d_k B_k d_k].$$

This implies that there exists k' such that (45) holds. $\square$

Theorem 4.2. *If* (H3), (H4) *and* (H5) *hold*, $d_k = -B_k^{-1} g_k$ *and Algorithm* (A) *generates an infinite sequence* $\{x_k\}$, *then* $\{x_k\}$ *converges to* x^* *superlinearly.*

Proof. By Theorem 3.1 we have $\{x_k\} \to x^*$. By Lemma 4.2, there exists k' such that (45) holds. By Lemma 4.2, for $k \geqslant k'$, we have

$$x_{k+1} = x_k + d_k,$$

where $d_k = -B_k^{-1} g_k$. By Lemma 4.1 and the mean value theorem, it follows that

$$\begin{aligned} g_{k+1} - g_k &= \int_0^1 \nabla^2 f(x_k + t(x_{k+1} - x_k))(x_{k+1} - x_k)\,\mathrm{d}t \\ &= \int_0^1 \nabla^2 f(x_k + t d_k) d_k \,\mathrm{d}t \\ &= \nabla^2 f(x^*) d_k + \int_0^1 [\nabla^2 f(x_k + t d_k) - \nabla^2 f(x^*)] d_k \,\mathrm{d}t \\ &= \nabla^2 f(x^*) d_k + \mathrm{o}(\|d_k\|), \end{aligned}$$

thus

$$\begin{aligned} g_{k+1} &= g_k + \nabla^2 f(x^*) d_k + \mathrm{o}(\|d_k\|) \\ &= - B_k d_k + \nabla^2 f(x^*) d_k + \mathrm{o}(\|d_k\|) \\ &= - [B_k - \nabla^2 f(x^*)] d_k + \mathrm{o}(\|d_k\|). \end{aligned}$$

By (44) and the above equality we have

$$\lim_{k \to \infty} \frac{\|g_{k+1}\|}{\|d_k\|} = 0. \tag{48}$$

By Lemma 4.1 it holds that

$$\begin{aligned} \frac{\|g_{k+1}\|}{\|d_k\|} &\geqslant \frac{m'\|x_{k+1} - x^*\|}{\|d_k\|} \\ &= \frac{m'\|x_{k+1} - x^*\|}{\|x_{k+1} - x_k\|} \\ &\geqslant \frac{m'\|x_{k+1} - x^*\|}{\|x_{k+1} - x^*\| + \|x_k - x^*\|} \\ &= m' \frac{((\|x_{k+1} - x^*\|)/(\|x_k - x^*\|))}{1 + ((\|x_{k+1} - x^*\|)/(\|x_k - x^*\|))}, \end{aligned}$$

and by (48) it follows that

$$\lim_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0,$$

which implies that $\{x_k\}$ converges to x^* superlinearly. □

Remark 4.1. We can see from the above theorem that, if we take $d_k = -B_k^{-1} g_k$ at each step, the related algorithm with the NNLS will reduce to quasi-Newton method and has superlinear convergence rate under some mild conditions.

4.3. Quadratic convergence rate

If we take $B_k = \nabla^2 f(x_k)$ in Algorithm (A) and let $\delta = 1$, then (H5) holds. We have the following result.

Theorem 4.3. *Assume that* (H3) *and* (H4) *hold*, $B_k = \nabla^2 f(x_k)$ *and* $\delta = 1$ *for sufficiently large* k. *Moreover, there exists a neighborhood* $N(x^*, \varepsilon) = \{x \in R^n \mid \|x - x^*\| < \varepsilon\}$ *of* x^* *such that* $\nabla^2 f(x)$ *is Lipschitz continuous on* $N(x^*, \varepsilon)$, *i.e., there exists* $L(\varepsilon)$ *such that*

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \leqslant L(\varepsilon)\|x - y\|, \ \forall x, y \in N(x^*, \varepsilon). \tag{49}$$

Algorithm (A) *generates an infinite sequence* $\{x_k\}$. *Then* $\{x_k\}$ *converges to* x^* *quadratically.*

Proof. By Theorem 3.1, Lemmas 4.1 and 4.3, it follows that $\{x_k\}$ converges to x^* and there exists k' such that for all $k \geqslant k'$, $x_k \in N(x^*, \varepsilon)$, $B_k = \nabla^2 f(x_k)$, and $\alpha_k = 1$. Let $\varepsilon_k = x_k - x^*$. By the mean value theorem we have

$$\begin{aligned}
\varepsilon_{k+1} &= x_{k+1} - x^* \\
&= x_k - x^* + d_k \\
&= \varepsilon_k - \nabla^2 f(x_k)^{-1} g_k \\
&= \varepsilon_k - \nabla^2 f(x_k)^{-1} (g_k - g^*) \\
&= \varepsilon_k - \nabla^2 f(x_k)^{-1} \int_0^1 \nabla^2 f(x^* + t\varepsilon_k)\varepsilon_k \, \mathrm{d}t \\
&= \nabla^2 f(x_k)^{-1} \left[\nabla^2 f(x_k)\varepsilon_k - \int_0^1 \nabla^2 f(x^* + t\varepsilon_k)\varepsilon_k \, \mathrm{d}t \right] \\
&= \nabla^2 f(x_k)^{-1} \int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x^* + t\varepsilon_k)]\varepsilon_k \, \mathrm{d}t.
\end{aligned}$$

This and (49) imply that

$$\begin{aligned}
\|\varepsilon_{k+1}\| &= \|\nabla^2 f(x_k)^{-1} \int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x^* + t\varepsilon_k)] \, \mathrm{d}t \, \varepsilon_k\| \\
&\leqslant \|\nabla^2 f(x_k)^{-1}\| \int_0^1 \|\nabla^2 f(x_k) - \nabla^2 f(x^* + t\varepsilon_k)\| \, \mathrm{d}t \, \|\varepsilon_k\| \\
&\leqslant \|\nabla^2 f(x_k)^{-1}\| \cdot L(\varepsilon)\|\varepsilon_k\|^2 \int_0^1 (1 - t) \, \mathrm{d}t \\
&= \frac{1}{2} L(\varepsilon)\|\nabla^2 f(x_k)^{-1}\| \cdot \|\varepsilon_k\|^2.
\end{aligned}$$

Therefore,

$$\lim_{k\to\infty} \frac{\|\varepsilon_{k+1}\|}{\|\varepsilon_k\|^2} \leqslant \frac{1}{2} L(\varepsilon) \lim_{k\to\infty} \|\nabla^2 f(x_k)^{-1}\| = \frac{1}{2} L(\varepsilon)\|\nabla^2 f(x^*)^{-1}\|,$$

which implies that $\{x_k\}$ converges to x^* quadratically. $\square$

Remark 4.2. The above theorem shows that if we take $B_k = \nabla^2 f(x_k)$ for sufficiently large k in Algorithm (A), then the algorithm reduces to Newton method finally. The results on convergence rate also show that the NNLS is available in practical computation. Firstly, it guarantees that the related algorithm has global convergence under mild conditions. Secondly, it is possible to reduce the function evaluations at each iteration and make the algorithm converge more quickly. And finally, it does not add any more amount of computation and storage and thus is a promising and available algorithm.

5. Numerical experiments

In this section we report some numerical results for the NNLS method. At first, in Algorithm(A) we take the parameters $\sigma = 0.38$, $\beta = 0.618$, $\delta = 1$, $B_0 = I$ and B_k is modified by BFGS formula. Algorithm(B) denotes the nonmonotone line

Table 1
Iterations and function evaluations, $d_k = -g_k$

A\P	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5
Algorithm (A)	25, 34	31, 45	27, 39	29, 38	32, 45
Algorithm (B)	38, 62	48, 83	47, 92	53, 73	43, 63

Table 2
Iterations and function evaluations, $d_k = -B_k^{-1} g_k$

A\P	Problem 1	Problem 2	Problem 3	Problem 4	Problem 5
Algorithm (A)	18, 18	22, 22	23, 28	25, 25	23, 23
Algorithm (B)	26, 39	34, 34	42, 42	62, 62	42, 42

Table 3
Iterations and function evaluations, $d_k = -g_k$

A\P	Problem 6	Problem 7	Problem 8	Problem 9	Problem 10
Algorithm (A)	37, 48	38, 42	37, 47	39, 65	41, 88
Algorithm (B)	38, 67	58, 133	64, 192	79, 173	85, 163

Table 4
Iterations and function evaluations, $d_k = -B_k^{-1} g_k$

A\P	Problem 6	Problem 7	Problem 8	Problem 9	Problem 10
Algorithm (A)	26, 26	34, 42	33, 48	32, 32	33, 67
Algorithm (B)	32, 63	44, 78	48, 126	52, 165	82, 135

search algorithm with the original nonmonotone Armijo line search. In Algorithm(B) we take $\sigma = 0.38$, $\beta = 0.618$ and $s = 1$. Test unconstrained optimization problems and their initial points are cited from the literature [12, pp. 384–386]. The dimension of Problem 1, Problem 3 and Problem 5 is taken as $n = 20$, 10 and 10, respectively. The stop criterion is

$$\|g_k\| \leqslant 10^{-9} \|\nabla f(x_0)\|.$$

In the numerical experiments, a portable computer with Pentium IV/1.2 MH CPU and C++ Language with double precision are used to implement the new nonmonotone line search method. The parameter is taken as $M'' = 3$ which indicates the degree of nonmonotonicity of the nonmonotone line search method.

In Tables 1 and 2, a pair of numbers means that the first number denotes the iterations and the second number denotes function evaluations, respectively. Numerical experiments show that the NNLS method is superior to the related method with the original nonmonotone Armijo line search. It can be seen that the NNLS algorithm needs less iterations and less function evaluations in solving the five test problems. This also shows that the new nonmonotone search procedure can easily seek the accepted step size at each iteration.

Test Problems 6–10 and the initial points are from the literature [9], which corresponds to Problems 21–25 with the dimension $n = 100, 300, 500, 1000, 2000$. Tables 3 and 4 also show that the numerical results support the NNLS.

If the initial points for Problems 6–10 are changed as

$$x_0 = (\xi_j), \quad \xi_{2j-1} = -12, \quad \xi_{2j} = 10;$$

$$x_0 = (\xi_j), \quad \xi_{4j-3} = 30, \quad \xi_{4j-2} = -10, \quad \xi_{4j-1} = 5, \quad \xi_{4j} = 10;$$

$$x_0 = (\xi_j), \quad \xi_j = j^2;$$

Table 5
Iterations and function evaluations, $d_k = -g_k$

A\P	Problem 6	Problem 7	Problem 8	Problem 9	Problem 10
Algorithm (A)	38, 43	36, 45	32, 41	42, 53	38, 57
Algorithm (B)	43, 68	62, 145	68, 167	72, 158	78, 144

Table 6
Iterations and function evaluations, $d_k = -B_k^{-1} g_k$

A\P	Problem 6	Problem 7	Problem 8	Problem 9	Problem 10
Algorithm (A)	28, 36	32, 46	35, 45	30, 38	32, 68
Algorithm (B)	37, 42	48, 74	43, 89	58, 135	62, 135

$$x_0 = (1, 2, 3 \ldots, 1000)^{\mathrm{T}};$$

$$x_0 = (\xi_j), \quad \xi_j = n - (j/n),$$

then the numerical results are listed in Tables 5 and 6.

It is shown from Tables 5 and 6 that different initial points affect the convergence very little. This also shows that the new nonmonotone line search makes the algorithm converge stably.

Moreover, the initial step size s_k in the NNLS may be adjusted automatically at different steps. This enables us to adjust the initial step size at each iteration so as to improve the performance of the relevant methods, especially to reduce the function evaluations at each iteration. It is reasonable to take the initial step size $s_k = -\delta g_k^{\mathrm{T}} d_k / d_k^{\mathrm{T}} B_k d_k$ at the kth iteration, where $\delta \in [0.5, 2)$. In fact, if we use the exact line search rule at the kth step, we obtain

$$g_{k+1}^{\mathrm{T}} d_k = 0. \tag{50}$$

By the mean value theorem, we have

$$g(x_k + \alpha_k^* d_k) - g_k = \alpha_k^* \int_0^1 \nabla^2 f(x_k + t\alpha_k^* d_k) d_k \, \mathrm{d}t.$$

Therefore,

$$-g_k^{\mathrm{T}} d_k = \alpha_k^* \int_0^1 d_k^{\mathrm{T}} \nabla^2 f(x_k + t\alpha_k^* d_k) d_k \, \mathrm{d}t,$$

and thus

$$\alpha_k^* = -\frac{g_k^{\mathrm{T}} d_k}{\int_0^1 d_k^{\mathrm{T}} \nabla^2 f(x_k + t\alpha_k d_k) d_k \, \mathrm{d}t}.$$

Since B_k is expected to approximate $\nabla^2 f(x_k)$, it is reasonable to take

$$\alpha_k \in \left(-\frac{g_k^{\mathrm{T}} d_k}{2 d_k^{\mathrm{T}} B_k d_k}, -\frac{2 g_k^{\mathrm{T}} d_k}{d_k^{\mathrm{T}} B_k d_k} \right).$$

Certainly, we hope $s_k = \alpha_k^*$ or s_k is close to α_k^*. Only in this way, can we reduce the function evaluations at each iteration and improve the performance of related algorithms.

Test problems 11–15 and their initial iterates are chosen from the literature [2] and corresponding to SPMSRTLS ($n = 10000$), TQUARTIC($n = 10000$), TRIDIA($n = 10000$), VAREIGVL($n = 5000$), and WOODS($n = 10000$), respectively. The numerical results are reported in Tables 7 and 8.

From Tables 7 and 8, we can see that the new nonmonotone line search can essentially reduce the function evaluations at each iteration. It is also shown that the initial trial step size $s_k = -g_k^{\mathrm{T}} d_k / d_k^{\mathrm{T}} B_k d_k$ is available and reasonable in the

Table 7
Iterations and function evaluations, $d_k = -g_k$

A\P	Problem11	Problem12	Problem13	Problem14	Problem15
Algorithm (A)	212, 226	15, 18	2344, 2352	18, 20	21, 21
Algorithm (B)	238, 247	28, 63	2945, 2968	55, 57	35, 46

Table 8
Iterations and function evaluations, $d_k = -B_k^{-1} g_k$

A\P	Problem11	Problem12	Problem13	Problem14	Problem15
Algorithm (A)	156, 158	15, 15	1621, 1628	15, 15	18, 18
Algorithm (B)	242, 263	24, 28	2548, 2552	28, 28	26, 26

Table 9
Function evaluations comparison

A\ d_k	$d_k = -g_k$	$d_k = -B_k^{-1} g_k$
Algorithm (A)	3367	2398
Algorithm (B)	5164	4158

new nonmonotone line search. Other numerical experiments that are omitted here also support the NNLS. It is also seen from Tables 7 and 8 that the new method with NNLS is superior to the corresponding nonmonotone method proposed in [19] for some problems.

From statistical viewpoint, the total function evaluations for solving all the mentioned problems are reported in Table 9.

We can see from Table 9 that the NNLS is available and efficient in practical computation. Moreover, the NNLS is essentially superior to the original nonmonotone Armijo line search.

In summary, the NNLS has two advantages. One is that it can adjust the initial trial step size s_k in accordance with the objective function, so as to reduce the function evaluations in each iteration. The other one is that B_k can be modified by means of quasi-Newton formulae such as, BFGS, DFP, and other formulae. This enables us to use quasi-Newton formulae to modify the nonmonotone line search, and improve the numerical performance of the resultant line search algorithms.

6. Conclusions and future research

In this paper, we proposed a new nonmonotone line search (abbreviated as NNLS) for general line search methods and establish some global convergence theorems. This NNLS rule is useful in designing new nonmonotone line search methods and possibly reduces the function evaluations at each iteration. In particular, the related method with the NNLS will reduce to quasi-Newton method if we take $d_k = -B_k^{-1} g_k$ at each step. If we take $d_k = -[\nabla^2 f(x_k)]^{-1} g_k$ at kth iteration, the method will reduce to Newton method for sufficiently large k. Moreover, we analyzed the convergence rate of some special methods with the NNLS, such as linear convergence rate, superlinear convergence rate, and quadratic convergence rate, etc. Preliminary numerical results also showed that the new nonmonotone line search method is superior to the original nonmonotone Armijo line search method.

It is obvious that, if we take $d_k = -g_k$ at each step, then the related method with the NNLS becomes a nonmonotone steepest descent method. In this case, the step size will satisfy

$$\alpha_k \leqslant s_k \leqslant \frac{\delta \|g_k\|^2}{g_k^{\mathrm{T}} B_k g_k} \leqslant \frac{\delta}{m}.$$

Taking initial test step size

$$s_k = -\frac{g_k^{\mathrm{T}} d_k}{d_k^{\mathrm{T}} B_k d_k}$$

at each step is very reasonable. In fact $s_k = 1$ is automatically satisfied in quasi-Newton and Newton-type methods for sufficiently large k. Thus, the iterations and function evaluations for reaching the same precision will be decreased essentially.

In the NNLS, how to choose the matrix B_k is very important. In fact, we can use quasi-Newton formulae such as BFGS, DFP, PSB, etc., to estimate the matrices B_k. It is possible to improve the convergence efficiency of corresponding line search methods. For the future research we should investigate the choosing approaches for parameters in the NNLS, for example, $\sigma \in (0, \frac{1}{2})$, $\beta \in (0, 1)$, $\delta \in [0.5, 2)$ and B_k.

Acknowledgements

The authors want to express their sincere gratitude to two anonymous referees for their valuable suggestions and comments that greatly improved the presentation of the paper.

References

[1] L. Armijo, Minimization of functions having Lipschitz continuous first partial derivatives, Pacific J. Math. 16 (1966) 1–3.
[2] I. Bongartz, A.R. Conn, N.I.M. Gould, P.L. Toint, CUTE: constrained and unconstrained testing environments, ACM Trans. Math. Software 21 (1995) 123–160.
[3] Y.H. Dai, On the nonmonotone line search, J. Optim. Theory Appl. 112 (2) (2002) 315–330.
[4] A.A. Goldstein, On steepest descent, SIAM J. Control 3 (1965) 147–151.
[5] L. Grippo, F. Lampariello, S. Lucidi, A nonmonotone line search technique for Newton's method, SIAM J. Numer. Anal. 23 (1986) 707–716.
[6] L. Grippo, F. Lampariello, S. Lucidi, A truncated Newton method with nonmonotone line search for unconstrained optimization, J. Optim. Theory Appl. 60 (1989) 401–419.
[7] L. Grippo, M. Sciandrone, Nonmonotone globalization techniques for the Barzilai-Borwein gradient method, Comp. Optim. Appl. 23 (2002) 143–169.
[8] B. Hüther, Global convergence of algorithms with nonmonotone line search strategy in unconstrained optimization, Results Math. 41 (2002) 320–333.
[9] J.J. Moré, B.S. Garbow, K.E. Hillstrom, Testing unconstrained optimization software, ACM Trans. Math. Software 7 (1) (1981) 17–41.
[10] J. Nocedal, J.S. Wright, Numerical Optimization, Springer, New York, Inc., 1999.
[11] E. Polak, Optimization: Algorithms and Consistent Approximations, Springer, New York, Inc., 1997.
[12] B.T. Polyak, Introduction to optimization, Optimization Software, Inc., Publications Division, New York, 1987.
[13] Z.J. Shi, Convergence of line search methods for unconstrained optimization, Appl. Math. Comput. 157 (2004) 393–405.
[14] Z.J. Shi, J. Shen, New inexact line search method for unconstrained optimization, J. Optim. Theory Appl. 127 (2) (2005) (to appear).
[15] M.J. Todd, On convergence properties of algorithms for unconstrained minimization, IMA J. Numer. Anal. 9 (1989) 435–441.
[16] M.N. Vrahatis, G.S. Androulakis, J.N. Lambrinos, G.D. Magoulas, A class of gradient unconstrained minimization algorithms with adaptive stepsize, J. Comput. Appl. Math. 114 (2000) 367–386.
[17] P. Wolfe, Convergence conditions for ascent methods, SIAM Review 11 (1969) 226–235.
[18] P. Wolfe, Convergence condition for ascent methods II: some corrections, SIAM Review 13 (1971) 185–188.
[19] H.C. Zhang, W.W. Hager, A nonmonotone line search technique and its application to unconstrained optimization, SIAM J. Optim. 14 (2004) 1043–1056.