Journal of Computational Mathematics, Vol.25, No.1, 2007, 104–112.

A NONMONOTONE SECOND-ORDER STEPLENGTH METHOD FOR UNCONSTRAINED MINIMIZATION *1)

Qun-yan Zhou
(*Department of Basic Science, Jiangsu Teachers University of Technology, Changzhou 213001, China; School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China Email: zhouqunyan@sina.com*)
Wen-yu Sun
(*School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China Email: wyusun@public1.ptt.js.cn*)

Abstract

In this paper, a nonmonotone method based on McCormick's second-order Armijo's step-size rule [7] for unconstrained optimization problems is proposed. Every limit point of the sequence generated by using this procedure is proved to be a stationary point with the second-order optimality conditions. Numerical tests on a set of standard test problems are presented and show that the new algorithm is efficient and robust.

Mathematics subject classification: 65K05, 90C30.
Key words: Nonmonotone method, Armijo's line search, Direction of negative curvature, Unconstrained optimization.

1. Introduction

Consider the unconstrained optimization problem

$$\min_{x\in R^n} f(x), \tag{1.1}$$

where $f(x)$ is a real-valued twice continuously differentiable function.

There are two classes of basic global approaches to solve problem (1.1): the line search method and the trust region method. Most of these methods naturally require monotone decrease of the objective values to guarantee the global convergence. However, this usually slows the convergence rate of the minimization process, especially in the presence of steep-sided valleys. Recently, several algorithms with nonmonotone techniques have been proposed both in line search methods [5, 6, 11, 16], and trust region methods [3, 4, 10, 15]. Theoretical properties and numerical tests show that the nonmonotone techniques are efficient and competitive [12].

In [7] McCormick modified Armijo's rule and proposed a second-order Armijo's step-size rule, which includes second-order derivative information in the line-search. Using directions of negative curvature, this method can handle the cases where the Hessian matrices are not positive definite, so that the sequence generated by this method converges to a second-order stationary point.

Nonmonotone techniques now are proved to be popular and efficient to deal with optimization problems, especially for ill-conditioned optimization problems. In this paper, we will combine the nonmonotone technique with the second-order Armijo's step-size rule to form a nonmonotone version of the second-order steplength method for unconstrained minimization.

* Received April 21, 2005; final revised February 14, 2006; accepted March 24, 2006.
1) This work was supported by the National Natural Science Foundation of China (grant No. 10231060), the Specialized Research Fund of Doctoral Program of Higher Education of China at No.20040319003 and the Graduates' Creative Project of Jiangsu Province, China.

We first introduce some standard notations used throughout our paper:

1. The notation $\|\cdot\|$ denotes the Euclidean norm on R^n.

2. $g(x) \in R^n$ is the gradient of $f(x)$ evaluated at x, and $H(x) \in R^{n\times n}$ is the Hessian of $f(x)$ at x.

3. If $\{x_k\}$ is a sequence of points generated by an algorithm, we denote $f_k = f(x_k), g_k = g(x_k)$ and $H_k = H(x_k)$.

4. $\lambda_{min}(\cdot)$ stands for the minimal eigenvalue of a matrix.

This paper is organized as follows. In section 2, we describe a nonmonotone algorithm model with the second-order steplength rule and discuss how to determine the descent pair. In section 3 we prove the global convergence which establishes that each limit point of the sequence generated from our algorithm is the second-order stationary point. The numerical results by solving a set of standard test problems are presented in section 4. Finally, in section 5, we give the conclusions.

2. The Nonmonotone Second-order Steplength Method

First of all, we give the definitions of the indefinite point and the descent pair.

Definition 2.1. *A point x is an indefinite point if $H(x)$ has at least one negative eigenvalue. Further, if x is an indefinite point, then d is a direction of negative curvature if $d^T H(x)d < 0$.*

Definition 2.2. *If $s^T g(x) \le 0, d^T g(x) \le 0, d^T H(x)d < 0$, then (s, d) is called a descent pair at the indefinite point x; if x is not an indefinite point and $s^T g(x) < 0, d^T g(x) \le 0$, $d^T H(x)d = 0$, then (s, d) is called a descent pair with zero curvature direction.*

Obviously, when $H(x)$ is positive definite, d must be a zero vector and we only need to consider the descent direction s.

MoCormick's second-order Armijo's step rule is to find the smallest nonnegative integer $i(k)$ from $0, 1, \cdots$, when H_k is indefinite, such that

$$f(y_k(i)) - f(x_k) \le \rho 2^{-i}(s_k^T g_k + \frac{1}{2} d_k^T H_k d_k), \tag{2.1}$$

where

$$y_k(i) = x_k + s_k 2^{-i} + d_k 2^{-i/2}, \tag{2.2}$$

$0 < \rho < 1$ is a preassigned constant and (s_k, d_k) is a descent pair. Then set

$$x_{k+1} = y_k(i(k)).$$

In fact, no matter whether H_k is indefinite or not, we can use the rule (2.1) in every iteration because we can let d_k be a zero vector whenever H_k is positive definite. Clearly, when H_k is positive definite, the second-order step-size rule (2.1) is reduced to the classical Armijo's step rule. In the following, we will assume that the rule (2.1) is used in every iteration.

In order to satisfy (2.1) for a finite integer $i(k)$, it is sufficient that

$$s_k^T g_k < 0$$

whenever $g_k \ne 0$, and

$$d_k^T H_k d_k < 0$$

whenever $g_k = 0$. Such a descent pair (s_k, d_k) does not exist only when x_k is a second-order stationary point. In this case the algorithm will be terminated.

In [7], it is supposed that the second-order step-size rule is used in conjunction with a non-ascent algorithm. In fact, this is not necessary, and it may cause severe loss of efficiency. We can relax the accepting condition on $y_k(i)$. Let

$$f(x_{l(k)}) = \max_{0\le j\le m(k)} f(x_{k-j}), \tag{2.3}$$

where $m(0) = 0$ and $0 \le m(k) \le \min\{m(k-1)+1, M\}, k \ge 1$, and M is a nonnegative integer. We modify (2.1) as follows:

$$f(y_k(i)) - f(x_{l(k)}) \le \rho 2^{-i}(s_k^T g_k + \frac{1}{2} d_k^T H_k d_k), \tag{2.4}$$

that is to say, we only need to find the smallest nonnegative integer $i(k)$ from $0, 1, \cdots$, such that (2.4) is satisfied, then set $x_{k+1} = y_k(i(k))$. Since $f(x_k) \le f(x_{l(k)})$, we can easily deduce that (2.4) is more relaxed, and when $M = 0$, (2.4) is reduced to (2.1).

Now we describe the nonmonotone second-order steplength algorithm as follows.

Algorithm 2.3. *(NSOSM)*

Step 0. Given $x_0 \in R^n, M > 0, 0 < \rho < 1$, set $k = 0, m(0) = 0$, and compute $f_0 = f(x_0)$.

Step 1. Compute g_k, H_k. If the stopping criterion holds, stop.

Step 2. Compute the descent pair (s_k, d_k) and $f(x_{l(k)})$, set $i = 0$.

Step 3. Compute $p_k = 2^{-i}s_k + 2^{-i/2}d_k$ and $f(x_k + p_k)$.

Step 4. If $f(x_k + p_k) > f(x_{l(k)}) + \rho 2^{-i}(s_k^T g_k + \frac{1}{2} d_k^T H_k d_k)$, set $i = i+1$, go to Step 3.

Step 5. Set $x_{k+1} = x_k + p_k, m(k+1) = min[m(k)+1, M], k = k+1$, go to Step 1.

In order to make sure that the limit point of $\{x_k\}$ is a second-order point, stronger conditions on the descent pair (s_k, d_k) must be imposed. Now we discuss how to compute the descent pair (s_k, d_k) in Step 2.

We introduce a stable factorization method for general symmetric matrices, which was presented by Bunch and Parlett [2]. The method factorizes the Hessian matrix H_k into the following form

$$PH_kP^T = LD_kL^T, \tag{2.5}$$

where P is a permutation matrix, L a unit lower triangular matrix and D_k a block diagonal matrix with 1×1 and 2×2 diagonal blocks. If H_k is positive definite, D_k is just diagonal. The factorization has the following properties (see [2] and [14]):

Property 1. D_k and H_k have the same inertia.

Property 2. There exist positive constants a_1, a_2, a_3 and a_4 which are independent of H_k, such that

$$a_1 \le \|L\| \le a_2, \ a_3 \le \|L^{-1}\| \le a_4; \tag{2.6}$$

Property 3. Suppose that H_k is not positive definite and let μ_k and λ_k be the most negative eigenvalues of H_k and D_k, respectively. Then the following relation holds:

$$\lambda_k \|L\|^2 \le \mu_k \le \lambda_k / \|L^{-1}\|^2. \tag{2.7}$$

Based on the Bunch-Parlett factorization, we can determine the descent pair. We get the spectral decomposition of D_k in (2.5) as follows:

$$D_k = U\Lambda_k U^T,$$

where $\Lambda_k = diag(\lambda_1^{(k)}, \lambda_2^{(k)}, \cdots, \lambda_n^{(k)})$, and U is an orthogonal matrix. Then set

$$\bar{\lambda}_j^{(k)} = \max\{|\lambda_j^{(k)}|, \varepsilon n \max_{1 \le i \le n} |\lambda_i^{(k)}|, \varepsilon\},$$

$$\bar{\Lambda}_k = diag(\bar{\lambda}_1^{(k)}, \cdots, \bar{\lambda}_n^{(k)}),$$

where ε is a relative machine precision, and set

$$\bar{D}_k = U\bar{\Lambda}_k U^T.$$

The direction s_k can be obtained by solving

$$(P^T L\bar{D}_k L^T P)s = -g_k. \tag{2.8}$$

Let z_k be the unit eigenvector of D_k corresponding to the minimal eigenvalue λ_k. Clearly, the direction

$$t_k = |\min\{\lambda_k, 0\}|^{\frac{1}{2}} P^T L^{-T} z_k \tag{2.9}$$

satisfies $t_k^T H_k t_k = \lambda_k |\min\{\lambda_k, 0\}| \leq 0$. This shows that t_k is a direction of nonpositive curvature. To make it a non-ascent direction, we can choose

$$d_k = \begin{cases} -t_k, & g_k^T t_k > 0, \\ t_k, & g_k^T t_k \leq 0. \end{cases} \tag{2.10}$$

The matrix H_k is positive semidefinite if and only if λ_k is nonnegative because of Property 1 of the Bunch-Parlett factorization. So, when H_k is positive semidefinite, d_k determined by (2.10) is a zero vector.

3. Convergence Analysis

In this section, we will discuss the convergence properties of Algorithm 2.3. The following assumption is required.

Assumption 3.1. *The level set $L(x_0) = \{x \,|\, f(x) \leq f(x_0)\}$ is bounded and $f(x)$ is twice continuously differentiable in $L(x_0)$.*

Lemma 3.2. *Suppose that Assumption* 3.1 *holds, and the descent pair (s_k, d_k) is determined by* (2.8) *and* (2.10)*, then*

$$d_k^T H_k d_k = \lambda_k |\min\{0, \lambda_k\}|, \tag{3.1}$$

and there exist constants $0 < c_1 \leq 1, c_2 > 0$ and $c_3 > 0$, such that

$$-\frac{s_k^T g_k}{\|s_k\|\|g_k\|} \geq c_1, \quad when \quad g_k \neq 0, \tag{3.2}$$

$$c_3\|g_k\| \geq \|s_k\| \geq c_2\|g_k\|. \tag{3.3}$$

Proof. Because (3.1) and (3.3) are obvious, we only need to prove (3.2). Since $f : R^n \to R$ and x_0 satisfy Assumption 3.1, the compactness of $L(x_0)$ and the continuity of $H(x)$ imply that $\{H_k\}$ are uniformly bounded. Thus, there exist two positive scalars $\varepsilon \leq M$ such that $0 < \varepsilon \leq \lambda_i(\bar{D}_k) \leq M (i = 1, 2, \cdots, n)$, where $\lambda_i(\bar{D}_k)$ is any eigenvalue of $\bar{D}_k$. Set

$$\bar{H}_k = P^T L\bar{D}_k L^T P,$$

and let y be corresponding normalized eigenvector of $\lambda_{min}(\bar{H}_k)$. Then set $z = L^T P y$. So, we have $y = P^{-1}L^{-T}z$ and $\|y\| \leq \|L^{-T}\|\|z\| \leq a_4\|z\|$ following from Property 2 of Bunch-Parlett factorization. Thus $\|z\| \geq \frac{1}{a_4}$, and

$$\lambda_{min}(\bar{H}_k) = y^T\bar{H}_k y = y^T P^T L\bar{D}_k L^T P y = z^T\bar{D}_k z \geq \frac{\varepsilon}{a_4^2}.$$

As a result,

$$-s_k^T g_k = s_k^T P^T L\bar{D}_k L^T P s_k = s_k^T \bar{H}_k s_k \geq \frac{\varepsilon\|s_k\|^2}{a_4^2}. \tag{3.4}$$

On the other hand, from(2.8), we obtain

$$\|g_k\| \le \|P^T L\bar{D}_k L^T P\|\|s_k\| \le a_2^2 M\|s_k\|. \tag{3.5}$$

Thus

$$-s_k^T g_k \ge \frac{\varepsilon}{a_2^2 a_4^2 M}\|s_k\|\|g_k\| \triangleq c_1\|s_k\|\|g_k\|, \tag{3.6}$$

It is obvious that $0 < c_1 \le 1$. So (3.2) has been proved.

The first of the following lemmas follows easily from the proof of Theorem in [5], so we omit the proof.

Lemma 3.3. *Suppose that Assumption* 3.1 *holds and* $\{x_k\}$ *is generated by Algorithm* 2.3*. Then the sequence* $\{x_k\}$ *remains in* $L(x_0)$*, and* $\{f(x_{l(k)})\}$ *is nonincreasing and convergent.*

Lemma 3.4. *Algorithm* 2.3 *cannot cycle infinitely between Step* 3 *and Step* 4*.*

Proof. Since $g_k^T d_k \le 0$, $g_k^T s_k + \frac{1}{2}d_k^T H_k d_k < 0$ and $0 < \rho < 1$, then for sufficiently large i we have

$$\begin{aligned}
&f(x_k + 2^{-i}s_k + 2^{-i/2}d_k)\\
&= f(x_k) + 2^{-i}g_k^T s_k + 2^{-i/2}g_k^T d_k + \frac{1}{2}2^{-i}d_k^T H_k d_k + o(2^{-i})\\
&\le f(x_k) + 2^{-i}g_k^T s_k + \frac{1}{2}2^{-i}d_k^T H_k d_k + o(2^{-i})\\
&\le f(x_{l(k)}) + 2^{-i}(g_k^T s_k + \frac{1}{2}d_k^T H_k d_k) + o(2^{-i})\\
&< f(x_{l(k)}) + \rho 2^{-i}(g_k^T s_k + \frac{1}{2}d_k^T H_k d_k).
\end{aligned} \tag{3.7}$$

It follows from (3.7) that Algorithm 2.3 cannot cycle infinitely between Step 3 and Step 4.

Now we are in a position to state our main theorem.

Theorem 3.5. *Suppose that Assumption* 3.1 *holds and that* $\{x_k\}$ *is an infinite sequence generated from Algorithm* 2.3*. Then every limit point* $\bar{x}$ *of the sequence is a second order stationary point, i.e.* $g(\bar{x}) = 0$ *and the Hessian matrix* $H(\bar{x})$ *is at least positive semidefinite.*

Proof. For convenience, we denote $2^{-i(k)}$ by α_k. For $k > M$, it follows from (2.4), (3.2), (3.3) and (3.1) that

$$\begin{aligned}
f(x_{l(k)}) &\le f(x_{l(l(k)-1)}) + \rho\alpha_{l(k)-1}[s_{l(k)-1}^T g_{l(k)-1} + \frac{1}{2}d_{l(k)-1}^T H_{l(k)-1}d_{l(k)-1}]\\
&\le f(x_{l(l(k)-1)}) + \rho\alpha_{l(k)-1}[-c_1c_2\|g_{l(k)-1}\|^2 + \frac{1}{2}\lambda_{l(k)-1}|\min\{\lambda_{l(k)-1}, 0\}|].
\end{aligned}$$

By Lemma 3.3, $\{f(x_{l(k)})\}$ admits a limit for $k \to \infty$. Taking the limit we obtain

$$\lim_{k\to\infty}\rho\alpha_{l(k)-1}[-c_1c_2\|g_{l(k)-1}\|^2 + \frac{1}{2}\lambda_{l(k)-1}|\min\{\lambda_{l(k)-1}, 0\}|] = 0. \tag{3.8}$$

Since $\alpha_{l(k)-1} > 0, -c_1c_2\|g_{l(k)-1}\|^2 \le 0$ and $\lambda_{l(k)-1}|\min\{\lambda_{l(k)-1}, 0\}| \le 0$, it follows from (3.8) that

$$\lim_{k\to\infty}\alpha_{l(k)-1}\|g_{l(k)-1}\|^2 = 0,\ \lim_{k\to\infty}\alpha_{l(k)-1}\lambda_{l(k)-1}|\min\{\lambda_{l(k)-1}, 0\}| = 0, \tag{3.9}$$

which implies that

$$\lim_{k\to\infty}\alpha_{l(k)-1}\|g_{l(k)-1}\| = 0,\ \lim_{k\to\infty}\alpha_{l(k)-1}^{\frac{1}{2}}|\min\{\lambda_{l(k)-1}, 0\}|^{\frac{1}{2}} = 0. \tag{3.10}$$

Combining with (3.3), (2.9) and (2.10), we obtain that

$$\lim_{k\to\infty} \alpha_{l(k)-1}\|s_{l(k)-1}\| = 0,\ \lim_{k\to\infty} \alpha_{l(k)-1}^{\frac{1}{2}}\|d_{l(k)-1}\| = 0, \tag{3.11}$$

so that

$$\lim_{k\to\infty} \|x_{l(k)} - x_{l(k)-1}\| = 0.$$

By the uniform continuity of $f(x)$ on $L(x_0)$, we find that

$$\lim_{k\to\infty} f(x_{l(k)-1}) = \lim_{k\to\infty} f(x_{l(k)}). \tag{3.12}$$

Let $\hat{l}(k) = l(k+M+2)$. By induction as [5], we can show that for any given j

$$\lim_{k\to\infty} \alpha_{\hat{l}(k)-j}\|s_{\hat{l}(k)-j}\| = 0,\ \lim_{k\to\infty} \alpha_{\hat{l}(k)-j}^{\frac{1}{2}}\|d_{\hat{l}(k)-j}\| = 0, \tag{3.13}$$

and then

$$\lim_{k\to\infty} f(x_{\hat{l}(k)-j}) = \lim_{k\to\infty} f(x_{l(k)}). \tag{3.14}$$

Now for any k,

$$x_{k+1} = x_{\hat{l}(k)} - \sum_{j=1}^{\hat{l}(k)-k-1} \alpha_{\hat{l}(k)-j} s_{\hat{l}(k)-j} - \sum_{j=1}^{\hat{l}(k)-k-1} \alpha_{\hat{l}(k)-j}^{\frac{1}{2}} d_{\hat{l}(k)-j},$$

it follows that

$$\lim_{k\to\infty} \|x_{k+1} - x_{\hat{l}(k)}\| = 0.$$

Since $\{f(x_{l(k)})\}$ admits a limit, it follows from the uniform continuity of $f(x)$ on $L(x_0)$ that

$$\lim_{k\to\infty} f(x_k) = \lim_{k\to\infty} f(x_{\hat{l}(k)}) = \lim_{k\to\infty} f(x_{l(k)}). \tag{3.15}$$

Also from (2.4), (3.2), (3.3) and (3.1), we have

$$\begin{aligned} f(x_{k+1}) &\le f(x_{l(k)}) + \rho\alpha_k[s_k^T g_k + \frac{1}{2} d_k^T H_k d_k] \\ &\le f(x_{l(k)}) + \rho\alpha_k[-c_1 c_2\|g_k\|^2 + \frac{1}{2}\lambda_k|\min\{0,\lambda_k\}|]. \end{aligned} \tag{3.16}$$

Using the same arguments employed for proving (3.9), we obtain

$$\lim_{k\to\infty} \alpha_k\|g_k\|^2 = 0,\ \lim_{k\to\infty} \alpha_k\lambda_k|\min\{0,\lambda_k\}| = 0. \tag{3.17}$$

Now let $\bar{x}$ be any limit point of $\{x_k\}$, i.e., there exists a subsequence $\{x_k\}_{K_1} \subset \{x_k\}$, such that

$$\lim_{k\in K_1, k\to\infty} x_k = \bar{x}.$$

Then by (3.17), either

$$\lim_{k\in K_1, k\to\infty} \|g_k\| = 0,\ \lim_{k\in K_1, k\to\infty} \lambda_k|\min\{0,\lambda_k\}| = 0, \tag{3.18}$$

or there exists $K_2 \subset K_1$, such that

$$\lim_{k\in K_2, k\to\infty} \alpha_k = 0, \tag{3.19}$$

which is just

$$\lim_{k\in K_2, k\to\infty} 2^{-i(k)} = 0.$$

In the first case, (3.18) implies $g(\bar{x}) = 0$ and $\bar{\lambda} \geq 0$ by continuity, where $\bar{\lambda}$ is the limit of the minimal eigenvalue sequence of $\{D_k\}$, i.e.,

$$\lim_{k \in K_1, k \to \infty} \lambda_{min}(D_k) \geq 0.$$

By the properties of Bunch-Parlett factorization, it follows that $\lambda_{min}(H(\bar{x})) \geq 0$, so $H(\bar{x})$ is at least positive semidefinite.

In the second case, from the definition of $i(k)$, there exists an index $\bar{k}$ such that for $k \geq \bar{k}$ and $k \in K_2$,

$$\begin{aligned} f[y_k(i(k)-1)] &> f(x_{l(k)}) + \rho 2^{-[i(k)-1]}[s_k^T g_k + \frac{1}{2} d_k^T H_k d_k] \\ &\geq f(x_k) + \rho 2^{-[i(k)-1]}[s_k^T g_k + \frac{1}{2} d_k^T H_k d_k]. \end{aligned} \tag{3.20}$$

Since $y_k(i(k)-1) = x_k + s_k 2^{-[i(k)-1]} + d_k 2^{-[i(k)-1]/2}$, expanding the left hand side of (3.20) by using Taylor's theorem, and making incorporation such that the appropriate terms are incorporated in $o(2^{-[i(k)-1]})$, we have that

$$\begin{aligned} & o(2^{-[i(k)-1]}) > (\rho - 1) 2^{-[i(k)-1]}[s_k^T g_k + \frac{1}{2} d_k^T H_k d_k] \\ & \geq (\rho - 1) 2^{-[i(k)-1]}[-c_1 c_2 \|g_k\|^2 + \frac{1}{2}\lambda_k |\min\{0, \lambda_k\}|]. \end{aligned} \tag{3.21}$$

Dividing both sides of (3.21) by $(1-\rho)2^{-[i(k)-1]}$ and taking limit as $k \to \infty, k \in K_2$, we have

$$0 \geq c_1 c_2 \|g(\bar{x})\| - \frac{1}{2}\bar{\lambda} |\min\{0, \bar{\lambda}\}|. \tag{3.22}$$

Since each term on the right-hand side of (3.22) is nonnegative, we obtain that $g(\bar{x}) = 0$ and $\bar{\lambda} \geq 0$, which also means that $\bar{x}$ is a second-order stationary point. Combining the above two cases establishes our theorem.

Remark 3.6. (1) By the continuity of $H(x)$, if there are infinitely many matrices which are not positive semidefinite during the iteration, then the Hessian matrix $H(\bar{x})$ is positive semidefinite with at least one eigenvalue equal to 0. (2)Assume the sequence $\{x_k\}$ generated from Algorithm 2.3 converges to x^* which is a strong local minimizer. Then the convergence rate is quadratic.

4. Numerical Experiment

In this section, Algorithm 2.3 (NSOSM) is tested on a set of standard test problems which are from [1, 8]. A MATLAB program is coded to perform the experiments. The iteration terminates when

$$\|g_k\| \leq 10^{-5}, \ \lambda_{min}(H_k) \geq 0, \tag{4.1}$$

or

$$f(x_{l(k)}) - f(x_{k+1}) \leq 10^{-20} \max\{10^{-10}, |\ f(x_{l(k)})\ |\}. \tag{4.2}$$

That is, when one of (4.1) and (4.2) is satisfied, the iteration terminates. In the program the parameter ρ is set as 0.001.

We run the program with $M = 0$ (monotone second-order steplength method) and $M = 10$ (nonmonotone second-order steplength method). The numerical results are listed in the following table. We denote the size of problems by n, the number of function evaluations by NF, the number of gradient evaluations by NG, the number of indefinite H_k appearing in the iteration of our algorithm by NI. We observe that for most problems, NF and NG of the nonmonotone second-order steplength method are less than those of the monotone case.

As another reference, in the last two columns we list the performance of nonmonotone line search algorithm ($NMLS$) proposed in [5]. The parameters are as follows: $c_1 = 10^{-5}, c_2 = 10^5, \gamma = 10^{-3}, \sigma = 0.5$, and the iteration terminates when $\|g_k\| \leq 10^{-5}$. The parameters c_1 and c_2 are used to make the descent direction gradient-related, which ensures the global convergence. We can see from the table that the performance of the two methods are comparable. Our algorithm sometimes needs more function and gradient evaluations, but it can go past a saddle point in principle, even if the starting point is just the saddle point. This is the main merit of our algorithm. In addition, our algorithm is robust and efficient for ill-conditioned problems though it is only an initial generalization of McCormick's work [7]. So, we believe that the further research on this topic will be necessary and worthwhile.

Table of numerical results

		$NSOSM$						$NMLS$	
		$M = 0$			$M = 10$			$M = 10$	
$Test\,functions$	n	NF	NG	NI	NF	NG	NI	NF	NG
Gaussian	3	2	2	0	2	2	0	2	2
Powell Badly Sc.	2	898	887	3	877	872	6	–	–
Box 3-Dimen.	3	20	16	1	28	25	10	9	9
Var. Dimen.	10	15	15	0	15	15	0	31	10
Watson	6	12	12	0	12	12	0	12	12
	9	13	13	0	13	13	0	13	13
	12	13	13	0	13	13	0	13	13
Penalty I	4	39	31	0	17	17	0	17	17
	10	41	34	0	24	24	0	24	24
Penalty II	4	7	7	0	7	7	0	7	7
	10	135	103	0	19	19	0	19	19
Brown-Dennis	4	9	9	0	9	9	0	84	12
Gulf R.and D.	3	43	32	6	42	34	6	50	39
Trigonometric	20	45	16	4	34	19	3	12	9
	40	32	11	2	42	17	5	32	22
	60	62	14	5	96	22	10	28	18

Table of numerical results (continued)

		$NSOSM$						$NMLS$	
		$M = 0$			$M = 10$			$M = 10$	
$Test\,functions$	n	NF	NG	NI	NF	NG	NI	NF	NG
Ex.Ros.	2	29	22	0	16	12	0	16	12
	10	29	22	0	16	12	0	16	12
	20	29	22	0	16	12	0	16	12
Sc.Ros. $c = 10^4$	2	114	81	0	17	12	0	287	38
$c = 10^6$	2	517	349	0	15	10	0	–	–
Ex.Powell	4	16	16	0	16	16	0	16	16
	16	17	17	0	17	17	0	17	17
Beale	2	16	9	3	47	35	19	25	18
Wood	4	63	39	2	29	29	1	33	30
Cube	2	37	27	0	22	11	1	17	12
Sc.Cube $c = 10^4$	2	167	109	1	26	9	1	739	102
$c = 10^6$	2	705	483	1	33	9	1	–	–

Note: "–" denotes that when NF reaches 1000, the algorithm fails to reach the minimum. If this is the case, then we terminate the iteration.

5. Conclusions

In this paper, we propose a nonmonotone second-order steplength method for unconstrained optimization problems. The new method is an improvement of the second-order steplength algorithms. We have proved that every limit point of the new algorithm is a second-order stationary point. From the theoretical discussion and numerical experiments, we can see that the new algorithm shows the robustness and efficiency for ill-conditioned optimization problems, especially around saddle points. Our work is only an initial and direct generalization of McCormick's paper [7]. Furthermore, we may consider the nonmonotone second-order Goldstein rule, the nonmonotone second-order Wolfe rule, and the general nonmonotone second-order rule for optimization, along with some advanced approaches to compute the descent pair (s_k, d_k). We believe that further research on this topic will be necessary and worthwhile.

Acknowledgements. We thank the two anonymous referees and editors for their helpful comments and suggestions on an earlier version of this paper.

References

[1] I. Bongartz, A.R. Conn, N.I.M. Gould, and Ph.L. Toint, CUTE: Constrained and unconstrained testing environment, *ACM Trans. Math. Software*, **21** (1995), 123-160.

[2] J.R. Bunch and B.N. Parlett, Direct methods for solving symmetrics indefinite systems of linear equations, *SIAM J. Numer. Anal.*, **8** (1971), 639-655.

[3] N.Y. Deng, Y. Xiao,and F.J. Zhou, Nonmontonic trust region algorithm, *J. Optim. Theory Appl.*, **26** (1993), 259-185.

[4] J. Fu and W. Sun, Nonmonotone adaptive trust-region method for unconstrained optimization problems, *Appl. Math. Comput.*, **163** (2005), 489-504.

[5] L. Grippo, F. Lamparillo, and S. Lucidi, A nonmonotone line search technique for Newton's method, *SIAM J. Numer. Anal.*, **23** (1986), 707-716.

[6] F. Lampariello and M. Sciandrone, Use of the minimum-norm search direction in a nonmonotone version of the Gauss-Newton method, *J. Optim. Theory Appl.*, **1** (2003), 65-82.

[7] G.P. McCormick, A modification of Armijo's step-size rule for negative curvature, *Math. Prog.*, **13** (1977), 111-115.

[8] J.J. Moré, B.S. Garbow and K.E. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Software*, **7** (1981), 17-41.

[9] J. Nocedal and S.J. Wright, Numerical Optimization, Springer-Verlag, New York, 1999.

[10] W. Sun, Nonmonotone trust-region method for solving optimization problems, *Appl. Math. Comput.*, **156** (2004), 159-174.

[11] W. Sun, J. Han, J. Sun, On global convergence of nonmonotone decent method, *J. Comput. Appl. Math.*, **146** (2002), 89-98.

[12] W. Sun, Nonmonotone optimization methods: motivation and development, invited talk presented in International Conference on Numerical Linear Algebra and Optimization, Guilin, China, October 7-10, 2003.

[13] Y. Yuan and W. Sun, Optimization Theory and Methods, Science Press, Beijing, 1997.

[14] J. Zhang and C. Xu, A class of indefinite dogleg path methods for unconstrained minimization, *SIAM J. Optimization*, **3** (1999), 646-667.

[15] J.L. Zhang and X.S. Zhang, A nonmonotone adaptive trust region method and its convergence, *Comput. Math. Appl.*, **45** (2003), 1469-1477.

[16] H. Zhou and W. Sun, Nonmonotone descent algorithm for nonsmooth unconstrained optimization problems, *Int. J. Pure Appl. Math.*, **9** (2003), 153-163.