Journal of Computational Mathematics, Vol. 24, No. 6, 2006, 761–770.

AN ADAPTIVE NONMONOTONIC TRUST REGION METHOD WITH CURVILINEAR SEARCHES [*1)]

Qun-yan Zhou
(*School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China Department of Mathematics, Jiangsu Teacher University of Technology, Changzhou 213001, China*)

Wen-yu Sun
(*School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China*)

Abstract

In this paper, an algorithm for unconstrained optimization that employs both trust region techniques and curvilinear searches is proposed. At every iteration, we solve the trust region subproblem whose radius is generated adaptively only once. Nonmonotonic backtracking curvilinear searches are performed when the solution of the subproblem is unacceptable. The global convergence and fast local convergence rate of the proposed algorithms are established under some reasonable conditions. The results of numerical experiments are reported to show the effectiveness of the proposed algorithms.

Mathematics subject classification: 65K05, 90C30.
Key words: Unconstrained optimization, Preconditioned gradient path, Trust region method, Curvilinear search.

1. Introduction

Consider the following unconstrained nonlinear programming problem

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

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

Trust region methods are very popular for solving problem (1.1). Many different versions have been suggested by using trust region strategy. The idea of combining the optimal path and the modified gradient path with the trust region strategy to solve (1.1) is originally due to Bulteau and Vial [2]. The two paths can be expressed by eigenvalues and eigenvectors of the Hessian matrix of the quadratic model function. However, the calculation of the full eigensystem of a symmetric matrix is usually time-consuming. To overcome this difficulty, Xu and Zhang in [14] have employed the stable Bunch-Parlett factorization method to factorize the Hessian to form a preconditioned optimal path within the trust region for unconstrained optimization. This idea is extended to the modified gradient path [7] and we call the corresponding path the preconditioned modified gradient path.

In addition, Nocedal and Yuan [9] suggest a combination of the trust region and line search methods, and the new algorithm preserves the strong convergence properties of trust region methods. This algorithm is extended to the nonmonotone case in [16]. Recently, Zhu [18, 19]

[*] Received November 30, 2005.
[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.

proposes an approximate trust region method and an inexact line search approach, respectively, with the preconditioned modified gradient path and the optimal path.

Besides, it is well known that the impact of the selection of trust-region radius on computational behavior is quite notable. In [4] [10] [20], a new trust region subproblem, whose trust region radius is determined automatically by the information of the gradient and the Hessian or its approximation, is employed. With this new subproblem, an adaptive trust region method is constructed. Numerical results show that the adaptive trust region algorithm is quite competitive with the traditional trust region methods.

Motivated by the above ideas, in this paper we combine a new adaptive trust region method with curvilinear searches to form a new algorithm which is different from the algorithm in [18]. Particularly, the curvilinear path used here is a preconditioned modified gradient path.

This paper is organized as follows. Section 2 describes the adaptive nonmonotonic trust region algorithm with curvilinear searches. In section 3 we investigate the global convergence and the superlinear convergence rate. The numerical results of a set of standard test problems are presented in section 4. Finally, some concluding remarks are addressed in section 5.

2. Adaptive Nonmonotonic Trust Region Algorithm with Curvilinear Searches

As known, in the algorithms to solve the trust region subproblem approximately along curvilinear paths, the step is the solution of a subproblem in the form

$$\begin{aligned} \min \quad & q_k(\delta) = f_k + g_k^T\delta + \frac{1}{2}\delta^T B_k \delta, \\ \text{s.t.} \quad & \delta \in \Gamma_k, \quad \|\delta\| \le \Delta_k, \end{aligned} \tag{2.1}$$

where $f_k = f(x_k), g_k = \nabla f(x_k), \delta = x - x_k, B_k$ is either $\nabla^2 f(x_k)$ or its approximation, Δ_k the trust region radius, Γ_k a curvilinear path in either the full-dimensional space or a lower dimensional subspace, and $\|\cdot\|$ is the l_2 norm.

At the beginning of this section, we first recall the forming of the preconditioned modified gradient path and its relative properties.

The Bunch-Parlett factorization method [1] factorizes the matrix B_k into the form

$$P_k B_k P_k^T = L_k D_k L_k^T, \tag{2.2}$$

where P_k is a permutation matrix, L_k a unit lower triangular matrix and D_k a block diagonal matrix with 1×1 and 2×2 diagonal blocks. The matrices D_k and B_k have the same number of positive, zero and negative eigenvalues. Besides, the elements of L_k and L_k^{-1} are bounded, i.e., there exist positive constants c_1, c_2, c_3, c_4, which are independent of the matrix B_k, such that

$$c_1 \le \|L_k\| \le c_2, \quad c_3 \le \|L_k^{-1}\| \le c_4, \forall k. \tag{2.3}$$

Now we can use L_k and P_k to scale the variables, that is, we use the new variable

$$\widehat{\delta} = L_k^T P_k \delta,$$

and take the following trust region subproblem

$$\begin{aligned} \min \quad & \widehat{q}_k(\widehat{\delta}) = f_k + \widehat{g}_k^T\widehat{\delta} + \frac{1}{2}\widehat{\delta}^T D_k \widehat{\delta}, \\ \text{s.t.} \quad & \widehat{\delta} \in \Gamma_k, \quad \|\widehat{\delta}\| \le \Delta_k, \end{aligned} \tag{2.4}$$

where $\widehat{g}_k = L_k^{-1} P_k g_k$. In the above subproblem, $\widehat{\delta}$ is required to be within the trust region rather than $P_k^T L_k^{-T}\widehat{\delta}$, which improves the efficiency of calculation of the solution step.

The modified gradient path Γ_k in (2.4) can be formulated easily because the full eigensystem of the block matrix D_k can be calculated easily (see [14]). Let $d_1 \le d_2 \le \cdots \le d_n$ be eigenvalues of D_k and $u^1, u^2, \cdots, u^n$ be corresponding orthonormal eigenvectors. We partition the set $\{1, \cdots, n\}$ into I^+, I^- and N according to $d_i > 0, d_i < 0$ and $d_i = 0$ for $i \in \{1, \cdots, n\}$ respectively. Concretely, the preconditioned modified gradient path can be given in a closed form (see [17]):

$$\Gamma(\tau) = \Gamma_1(t_1(\tau)) + \Gamma_2(t_2(\tau)), \quad \tau \in [0, +\infty). \tag{2.5}$$

Let $\widehat{g}_k^i = \widehat{g}_k^T u^i, i = 1, \cdots, n$. If $\widehat{g}_k^i \ne 0$ for some $i \in I^- \cup N$, then $\Gamma_2(t_2(\tau)) = 0$. For the path $\Gamma(\tau)$, the definitions of $\Gamma_1(t_1(\tau))$ and $\Gamma_2(t_2(\tau))$ are as follows:

$$\Gamma_1(t_1(\tau)) = \sum_{i \in I} \frac{e^{-d_i t_1(\tau)} - 1}{d_i} \widehat{g}_k^i u^i - t_1(\tau) \sum_{i \in N} \widehat{g}_k^i u^i,$$

$$\Gamma_2(t_2(\tau)) = \begin{cases} t_2(\tau) u^1, & \text{if} \quad d_1 < 0, \\ 0, & \text{if} \quad d_1 \ge 0, \end{cases}$$

where

$$t_1(\tau) = \begin{cases} \frac{\tau}{1-\tau}, & \text{if} \quad 0 < \tau < 1, \\ +\infty, & \text{if} \quad \tau \ge 1, \end{cases} \quad \text{and} \quad t_2(\tau) = \max\{\tau - 1, 0\}.$$

The path $\Gamma(\tau)$ is continuous. If D_k is positive definite, the path is finite with endpoint which is called the Newton point or quasi-Newton point. When the point x moves from x_k along the path, the following two properties hold [7]:

(P1) the distance to x_k is monotonically increasing;

(P2) the value of $\widehat{q}_k(\widehat{\delta})$ is monotonically decreasing.

As mentioned in the previous section, the choice of the trust region radius in the subproblem is very important to the efficiency of trust region algorithms. In the subproblem proposed in [20], $\Delta_k = c^p \|g_k\| \overline{M}_k, 0 < c < 1, \overline{M}_k = \|\widehat{B}_k^{-1}\|$, where p is a nonnegative integer whose initial value is 0, and $\widehat{B}_k$ is a safely positive definite matrix based on Schnabel and Eskow's modified Cholesky factorization [11]. In this paper, instead of using the modified Cholesky factorization, we directly use the eigensystem of D_k which is available and a byproduct in finding the gradient path. It can also make sure that the Newton's point is in the trust region. Concretely, the trust region subproblem (2.4) is as follows:

$$\begin{aligned} \min \quad & \widehat{q}_k(\widehat{\delta}(\tau)) = f_k + \widehat{g}_k^T \widehat{\delta}(\tau) + \frac{1}{2} \widehat{\delta}(\tau)^T D_k \widehat{\delta}(\tau), \\ \text{s.t.} \quad & \widehat{\delta}(\tau) = \Gamma_k(\tau), \quad \|\widehat{\delta}(\tau)\| \le \|\widehat{g}_k\| \|\widehat{D}_k^{-1}\|, \end{aligned} \tag{2.6}$$

where $\widehat{D}_k = U \text{diag}\{\widehat{d}_1, \widehat{d}_2, \cdots, \widehat{d}_n\} U^T, \widehat{d}_i = \max\{\varepsilon, d_i\} (i = 1, \cdots, n), \varepsilon$ is a small constant, $d_i (i = 1, \cdots, n)$ are the eigenvalues of D_k and the matrix U is an orthogonal matrix whose columns are corresponding orthonormal eigenvectors of D_k. From the properties of the preconditioned modified gradient path, we can get the solution of (2.6) easily.

In order to introduce the nonmonotone technique, let $\widehat{\delta}_k(\tau_k) = \Gamma_k(\tau_k)$ be the solution of the subproblem (2.6) and set

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

where $m(0) = 0$ and $0 \le m(k) \le \min\{m(k-1) + 1, M\}, k \ge 1$. We define

$$Pred(\delta_k) = f(x_{l(k)}) - \widehat{q}_k(\widehat{\delta}_k(\tau_k)) \tag{2.7}$$

to be the predicted reduction of $f(x)$,

$$Ared(\delta_k) = f(x_{l(k)}) - f(x_k + \delta_k(\tau_k)) \tag{2.8}$$

to be the actual reduction of $f(x)$, and

$$r_k = \frac{Ared(\delta_k)}{Pred(\delta_k)} \tag{2.9}$$

to be a measure of the improvement.

Now we state our adaptive nonmonotonic trust region algorithm with curvilinear searches.

Algorithm 2.1.

Step 0. *Given* $x_0 \in R^n, \eta > 0, \gamma \in (0, \frac{1}{2}), \alpha \in (0, 1), \epsilon > 0, M > 0$. *Set* $k = 0, m(k) = 0$, *and compute* $f_0 = f(x_0)$.

Step 1. *Compute* g_k. *If* $\|g_k\| \le \epsilon$, *stop with the approximate solution* x_k; *otherwise compute* B_k *and* $f(x_{l(k)})$.

Step 2. *Factorize* B_k *into the form* $P_k B_k P_k^T = L_k D_k L_k^T$ *and calculate the eigenvalues and orthonormal eigenvectors of* D_k. *Form the preconditioned modified gradient path* $\Gamma_k(\tau)$, *and set the trust region radius as* $\Delta_k = \|\widehat{D}_k^{-1}\| \|\widehat{g}_k\|$.

Step 3. *Solve the subproblem (2.6) for* $\widehat{\delta}_k(\tau_k)$, *and then set* $\delta_k(\tau_k) = P_k^T L_k^{-T} \widehat{\delta}_k(\tau_k)$.

Step 4. *Compute* $Pred(\delta_k), Ared(\delta_k)$ *and* r_k.

Step 5. *If* $r_k \ge \eta$, *set* $x_{k+1} = x_k + \delta_k(\tau_k)$; *otherwise select* τ_k^*, *which is the largest number in* $\{\tau_k, \alpha\tau_k, \alpha^2\tau_k, \cdots\}$ *such that*

$$f(x_k + \delta_k(\tau_k^*)) \le f(x_{l(k)}) + \gamma(1 - e^{-\tau_k^*}) g_k^T \delta_k'(0), \tag{2.10}$$

where $\delta_k'(0)$ *is the derivative of* $\delta_k(\tau_k)$ *at* $\tau_k = 0$. *Then set* $x_{k+1} = x_k + \delta_k(\tau_k^*)$.

Step 6. *Set* $m(k+1) = \min[m(k) + 1, M], k = k + 1$ *and go to Step 1.*

Remark 2.2. 1) In Step 3, the solution $\widehat{\delta}_k(\tau_k)$ is obtained as follows. If D_k is positive definite, then the endpoint with $\tau_k = \infty$, which is the Newton point or quasi-Newton point, is the solution of the subproblem. Otherwise, the nonlinear equation $\|\Gamma_k(\tau)\| = \Delta_k$ is solved to get τ_k. By the properties of the preconditioned modified gradient path, the equation $\|\Gamma_k(\tau)\| = \Delta_k$ has a solution and can be solved easily by the Newton-Raphson scheme or bisection method.

2) From Step 5 of the algorithm, we know that only when $r_k < \eta$, we search along the gradient path with the initial search factor τ_k which is just supplied by solving subproblem (2.6). When the solution of the trust region subproblem is good enough, i.e. $r_k \ge \eta$, we will not perform the curvilinear searches. In this sense the new algorithm allows a larger degree of non-monotonicity.

3. Convergence Analysis

In this section, we investigate the convergence properties of Algorithm 2.1. The following assumption is required.

Assumption 3.1. *(i) The function* $f : R^n \to R$ *is twice continuously differentiable.*

(ii) The level set $L(x_0) = \{x \,|\, f(x) \le f(x_0)\}$ *is bounded and* $f(x)$ *is continuously differentiable in* $L(x_0)$ *for any given* $x_0 \in R^n$.

(iii) Matrices $\{B_k\}$ *are uniformly bounded.*

The following inequalities, obtained from the definitions of $\widehat{g}_k$, D_k and inequality (2.3), hold:

$$\|g_k\|/c_2 \le \|g_k\|/\|L_k\| \le \|\widehat{g}_k\| \le \|L_k^{-1}\|\|g_k\| \le c_4\|g_k\|, \tag{3.1}$$

$$\|B_k\|/c_2^2 \le \|B_k\|/\|L_k\|^2 \le \|D_k\| \le \|L_k^{-1}\|^2\|B_k\| \le c_4^2\|B_k\|, \tag{3.2}$$

$$\|\delta_k\|/c_4 \le \|\delta_k\|/\|L_k^{-1}\| \le \|\widehat{\delta}_k\|. \tag{3.3}$$

Assumption 3.1 and (3.2) imply that there exists a positive number $M > 0$ such that

$$\|D_k\| \le M, \forall k.$$

Moreover, from the formulation of $\widehat{D}_k$, there exists a positive scalar $\Lambda \ge \varepsilon$ such that

$$0 < \varepsilon \le \lambda_i(\widehat{D}_k) \le \Lambda, \quad k = 1, 2, \cdots,$$

from which we know that

$$1/\Lambda \le \|\widehat{D}_k^{-1}\| \le 1/\varepsilon, \forall k. \tag{3.4}$$

Lemma 3.2. *If Assumption 3.1 holds and $\{x_k\}$ is generated by Algorithm 2.1, then the sequence $\{x_k\}$ remains in $L(x_0)$ and $\{f(x_{l(k)})\}$ is nonincreasing and convergent.*

Proof. The proof is similar to that of the theorem in [5].

Lemma 3.3. *If there exists a constant $\sigma > 0$ such that $\|g_k\| \ge \sigma$ for all k, then $\Delta_k \ge \epsilon_1, \forall k$, where $\epsilon_1 = \frac{\sigma}{\Lambda c_2}$.*

Proof. From the definition of Δ_k, (3.1) and (3.4), we have

$$\Delta_k = \|\widehat{D}_k^{-1}\|\|\widehat{g}_k\| \ge \frac{1}{c_2}\|\widehat{D}_k^{-1}\|\|g_k\| \ge \frac{1}{\Lambda c_2}\|g_k\| \ge \frac{\sigma}{\Lambda c_2} \triangleq \epsilon_1,$$

which establishes the lemma.

Similar to Lemma 2.1 in [19], we give the following lemma without proof.

Lemma 3.4. *Let the step $\widehat{\delta}_k(\tau)$ in the trust region subproblem be obtained from the preconditioned modified gradient path. Then we have the norm function $\|\Gamma(\tau)\|$ of the path is monotonically increasing for $\tau \in (0, +\infty)$, and the predicted reduction $Pred(\delta_k)$ satisfies the following sufficient descent condition:*

$$Pred(\delta_k) = f(x_{l(k)}) - \widehat{q}_k(\widehat{\delta}_k(\tau_k)) \ge w_1\|\widehat{g}_k\| \min\{\Delta_k, \frac{\|\widehat{g}_k\|}{\|D_k\|}\}, \tag{3.5}$$

where w_1 is a constant independent of k.

Before introducing the following lemma, we denote $NCS = \{k : r_k \ge \eta\}, CS = \{0, 1, 2, \cdots\} \setminus NCS$. It is obvious that when $k \in CS$, curvilinear searches are needed to perform.

Lemma 3.5. *If there exists a constant $\sigma > 0$ such that $\|g_k\| \ge \sigma$ for all k, then there exists a constant $\overline{\tau} > 0$ such that $\tau_k^* \ge \overline{\tau}, k \in CS$.*

Proof. When $k \in CS$, by Taylor's theorem, $g_k^T\delta_k'(0) = -\|\hat{g}_k\|^2 < 0$ and $1 - e^{-\tau} < \tau$, we have

$$\begin{aligned}
& f(x_k + \delta_k(\tau)) - f(x_{l(k)}) - \gamma(1 - e^{-\tau})g_k^T\delta_k'(0) \\
\le\ & f(x_k + \delta_k(\tau)) - f(x_k) - \gamma(1 - e^{-\tau})g_k^T\delta_k'(0) \\
\le\ & f(x_k) + \tau g_k^T\delta_k'(0) + o(\tau) - f(x_k) - \gamma\tau g_k^T\delta_k'(0) \\
=\ & (1 - \gamma)\tau g_k^T\delta_k'(0) + o(\tau) \\
=\ & (\gamma - 1)\tau\|\hat{g}_k\|^2 + o(\tau) \\
\le\ & (\gamma - 1)\tau\|g_k\|^2/c_2^2 + o(\tau) \\
\le\ & (\gamma - 1)\tau\sigma^2/c_2^2 + o(\tau).
\end{aligned} \tag{3.6}$$

It is easy to know that when τ is sufficiently small, $(\gamma-1)\tau\sigma^2/c_2^2+o(\tau)<0$ which means that when $k\in CS$, τ_k^* always can be found.

Suppose that the conclusion of this lemma is not true, then there exists an infinite subset $K\subseteq CS$ such that

$$\lim_{k\to\infty,\, k\in K}\tau_k^*=0.$$

From the algorithm, we have that for large enough $k\in K$,

$$f(x_k+\delta_k(\frac{\tau_k^*}{\omega}))-f(x_k)\geq f(x_k+\delta_k(\frac{\tau_k^*}{\omega}))-f(x_{l(k)})>\gamma(1-e^{-\frac{\tau_k^*}{\omega}})g_k^T\delta_k'(0). \tag{3.7}$$

For the composite function $f(x_k+\delta_k(\frac{\tau_k^*}{\omega}))$, we have

$$f(x_k+\delta_k(\frac{\tau_k^*}{\omega}))=f(x_k)+\frac{\tau_k^*}{\omega}g_k^T\delta_k'(0)+o(\frac{\tau_k^*}{\omega}). \tag{3.8}$$

By (3.7) and (3.8), from $g_k^T\delta_k'(0)\leq 0$ and $1-e^{-\frac{\tau_k^*}{\omega}}<\frac{\tau_k^*}{\omega}$, we have that

$$(1-\gamma)\frac{\tau_k^*}{\omega}g_k^T\delta_k'(0)+o(\frac{\tau_k^*}{\omega})\geq[\frac{\tau_k^*}{\omega}-\gamma(1-e^{-\frac{\tau_k^*}{\omega}})]g_k^T\delta_k'(0)+o(\frac{\tau_k^*}{\omega})>0. \tag{3.9}$$

Dividing (3.9) by $\frac{\tau_k^*}{\omega}$ and noting that $1-\gamma>0$, and $g_k^T\delta_k'(0)\leq 0$, we can obtain

$$\lim_{k\to\infty,\, k\in K}g_k^T\delta_k'(0)=0. \tag{3.10}$$

By (3.10) and $\tau_k^*\to 0$ as $k\to\infty$, $k\in K$, we have

$$-\lim_{k\to\infty,\, k\in K}\|\hat{g}_k\|^2=\lim_{k\to\infty,\, k\in K}g_k^T\delta_k'(0)=0,$$

which implies $\lim\limits_{k\to\infty,\, k\in K}\|g_k\|=0$. It contradicts $\|g_k\|\geq\sigma$ for all k. So, the conclusion is true.

Theorem 3.6. *If Assumption 3.1 holds and $\{x_k\}$ is generated by Algorithm 2.1, then*

$$\liminf_{k\to\infty}\|g_k\|=0. \tag{3.11}$$

Proof. Suppose, by contradiction, that the conclusion is not true, then there exists a positive constant $\sigma>0$ such that for all k, $\|g_k\|\geq\sigma$. From Lemma 3.3, Lemma 3.4 and Lemma 3.5, we know that when $k\in NCS$,

$$\begin{aligned} f(x_{l(k)})-f(x_{k+1}) &\geq \eta w_1\|\widehat{g}_k\|\min\{\Delta_k,\frac{\|\widehat{g}_k\|}{\|D_k\|}\} \\ &\geq \eta w_1\frac{\sigma}{c_2}\min\{\epsilon_1,\frac{\sigma}{c_2M}\}\triangleq z_1; \end{aligned} \tag{3.12}$$

when $k\in CS$,

$$\begin{aligned} f(x_{l(k)})-f(x_{k+1}) &\geq -\gamma(1-e^{-\tau_k^*})g_k^T\delta'(0) \\ &\geq -\gamma(1-e^{-\overline{\tau}})g_k^T\delta'(0) \\ &= \gamma(1-e^{-\overline{\tau}})\|\widehat{g}_k\|^2 \\ &\geq \gamma(1-e^{-\overline{\tau}})\sigma^2/c_2^2\triangleq z_2. \end{aligned} \tag{3.13}$$

Denote $z_0=\min\{z_1,z_2\}$, then from (3.12) and (3.13) we have

$$\begin{aligned} f(x_{l(k)}) &\geq f(x_{k+1})+z_0, \\ f(x_{l(k+1)}) &\geq f(x_{k+2})+z_0, \\ &\vdots \\ f(x_{l(k+M)}) &\geq f(x_{k+M+1})+z_0. \end{aligned}$$

Taking the maximal values of the two sides, we have that for all k,

$$f(x_{l(k)}) \geq \max\{f(x_{k+1}), f(x_{k+2}), \cdots, f(x_{k+M+1})\} + z_0. \tag{3.14}$$

On the other hand, $f(x_{l(k+M+1)}) \leq \max\{f(x_{k+1}), f(x_{k+2}), \cdots, f(x_{k+M+1})\}, \forall k$. Therefore, from (3.14), we have

$$f(x_{l(k)}) \geq f(x_{l(k+M+1)}) + z_0, \forall k.$$

By Lemma 3.2 and the above inequalities, when $k \to \infty$, we have $0 \geq z_0 > 0$, which is a contradiction. So, we complete the proof.

In the following, we establish the superlinear convergence.

Theorem 3.7. *Suppose that Assumption 3.1 holds and that Algorithm 2.1 produces an infinite sequence $\{x_k\}$ which converges to x_* where $\nabla^2 f(x_*)$ is positive definite. If B_k is positive definite for k sufficiently large and the following condition holds*

$$\lim_{k\to\infty} \frac{\|g(x_k + \delta_k(\tau_k))\|}{\|\delta_k(\tau_k)\|} = 0, \tag{3.15}$$

then x_k converges to x_* superlinearly.

Proof. By the assumption, $\widehat{D}_k = D_k$ when k is sufficiently large. Moreover, $\widehat{\delta}_k = -D_k^{-1}\widehat{g}_k$ is the solution of the following subproblem

$$\begin{aligned} \min \quad & f_k + \widehat{g}_k^T \widehat{\delta}(\tau) + \frac{1}{2}\widehat{\delta}(\tau)^T D_k \widehat{\delta}(\tau), \\ \text{s.t.} \quad & \widehat{\delta}(\tau) = \Gamma_k(\tau), \quad \|\widehat{\delta}(\tau)\| \leq \|\widehat{g}_k\| \|\widehat{D}_k^{-1}\|. \end{aligned} \tag{3.16}$$

In the following, we first prove that for k sufficiently large, we have

$$\frac{f(x_{l(k)}) - f(x_k + \delta_k(\tau_k))}{f(x_{l(k)}) - q_k(\delta_k(\tau_k))} > \eta. \tag{3.17}$$

By use of (3.15), we have

$$g_k + \nabla^2 f(x_k)\delta_k(\tau_k) = o(\|\delta_k(\tau_k)\|).$$

Since $x_k \to x_*$, then when $k \to \infty$,

$$g_k + \nabla^2 f(x_*)\delta_k(\tau_k) = o(\|\delta_k(\tau_k)\|),$$

which implies

$$\delta_k(\tau_k) = -(\nabla^2 f(x_*))^{-1} g_k + o(\|\delta_k(\tau_k)\|).$$

Then

$$\|\delta_k(\tau_k)\| \leq \|(\nabla^2 f(x_*))^{-1}\| \|g_k\| + o(\|\delta_k(\tau_k)\|).$$

Because $\nabla^2 f(x_*)$ is positive definite, we have $\|\delta_k(\tau_k)\| = O(\|g_k\|)$, further,

$$\|\delta_k(\tau_k)\| = O(\|\widehat{g}_k\|). \tag{3.18}$$

From Theorem 3.6 and the convergence of $\{x_k\}$, we have that $g_k \to 0$, which implies that $\delta_k(\tau_k) \to 0$. From (3.15) and $\delta_k(\tau_k) = P_k^T L_k^{-T} \widehat{\delta}_k(\tau_k) = -B_k^{-1} g_k$ as $k \to \infty$, we obtain

$$\delta_k(\tau_k)^T (g(x_k + \delta_k(\tau_k)) - g_k - B_k \delta_k(\tau_k)) = o(\|\delta_k(\tau_k)\|^2).$$

This equation together with $x_k \to x_*$ as $k \to \infty$ implies that

$$\delta_k(\tau_k)^T \nabla^2 f(x_*)\delta_k(\tau_k) - \delta_k(\tau_k)^T B_k \delta_k(\tau_k) = o(\|\delta_k(\tau_k)\|^2).$$

The above equation induces

$$f(x_k + \delta_k(\tau_k)) - q_k(\delta_k(\tau_k)) = o(\|\delta_k(\tau_k)\|^2). \tag{3.19}$$

From (3.19), Lemma 3.4 and (3.14) we obtain that when $k \to \infty$,

$$\left|\frac{f(x_{l(k)}) - f(x_k + \delta_k(\tau_k))}{f(x_{l(k)}) - q_k(\delta_k(\tau_k))} - 1\right| = \left|\frac{f(x_k + \delta_k(\tau_k)) - q_k(\delta_k(\tau_k))}{f(x_{l(k)}) - q_k(\delta_k(\tau_k))}\right| \le \frac{o(\|\delta_k(\tau_k)\|^2)}{w_1\|\widehat{g}_k\| \min\{\Delta_k, \frac{\|\widehat{g}_k\|}{M}\}} \to 0.$$

Therefore, when k is sufficiently large, we obtain

$$\frac{f(x_{l(k)}) - f(x_k + \delta_k(\tau_k))}{f(x_{l(k)}) - q_k(\delta_k(\tau_k))} > \eta.$$

So, from the definition of the algorithm, we have that, when k is sufficiently large, $x_{k+1} = x_k + \delta_k(\tau_k)$, where $\delta_k(\tau_k) = -B_k^{-1} g_k$. It says that, when k is sufficiently large, our nonmonotonic adaptive trust region method with curvilinear searches is equivalent to the standard Newton or quasi-Newton method. Therefore, the sequence $\{x_k\}$ converges to x_* superlinearly.

4. Numerical Results

In this section, the nonmonotonic adaptive trust region method with curvilinear searches is tested on a set of standard test problems from [3] [6] and [8]. Table 1 lists the function names we used in the numerical experiment.

A MATLAB program is coded to perform the experiments. The stopping rule in the experiment is $\|g_k\| \le 10^{-5}$. In addition, we assume $B_k = \nabla^2 f(x_k)$. The other parameters are as follows : $M = 10, \eta = 0.1, \alpha = 0.5, \gamma = 0.0001, \varepsilon = 10^{-5}$.

Table 1 Test Functions

No.	Function Name	No.	Function Name
1	Helical Valley	2	Gaussian
3	Powell Badly Scaled	4	Box 3-Dimensional
5	Variably Dimensioned	6	Watson
7	Penalty I	8	Penalty II
9	Brown Badly Scaled	10	Brown and Dennis
11	Gulf Res. and Dev.	12	Extended Rosenbrock
13	Extended Powell Singular	14	Beale
15	Wood	16	Chebyquad
17	Cube	18	Sc. Rosenbrock($c = 10^4$)
19	Sc. Rosenbrock($c = 10^6$)	20	Cliff

We compare the new algorithm (ANTRCS) with the nonmonotone curvilinear search method (NCS) and the usual nonmonotone trust region method with the preconditioned modified gradient path (NTR). It is not difficult to find that when B_k is positive definite, the path is the same when τ_k is set as any value in the interval $[1, +\infty]$. Thus, for nonmonotone curvilinear search method in our experiment, if D_k is positive definite, we set the initial value τ_k^0 of τ_k as 1. In addition, from Assumption 3.1, we know that τ_k can't be $+\infty$ whenever B_k is not positive definite. Therefore, in order to make the algorithm more effective, we select τ_k^0 according to the following simple adaptive rule. If $\|\Gamma_k(0.8)\| \le 100$, we set $\mu = 0.8$, otherwise $\mu = 0.2$. Then we distinguish two cases. If $\Gamma_2(t_2(\tau))$ is relevant, $\tau_k^0 = \mu^{-n}$, otherwise $\tau_k^0 = \mu$. Besides, when we solve the trust region subproblem, the bisection method is used to solve the equation $\|\Gamma_k(\tau)\| = \Delta_k$.

The numerical results are listed in Table 2. We denote the size of problems by N, the number of function evaluations by F, the number of gradient evaluations by G, and the final function value by FVAL. In the list, 'fail' means that when the iteration number exceeds 500, we stop the algorithm. From the table we can see that for most problems, the numbers F and G of the new algorithm are in general smaller than those of the other two algorithms, especially for the ill-conditioned problems. For thirty tests on the twenty functions, the new algorithm performs better than others in 12 tests and almost the same as the other algorithms in 14 tests. This means the new adaptive nonmonotonic trust region algorithm with curvilinear searches is more effective.

Table 2 Numerical Results of New Algorithm

		NCS		NTR		ANTRCS	
No.	N	F-G	FVAL	F-G	FVAL	F-G	FVAL
1	3	28-16	2.4211e-023	70-41	4.0945e-020	22-20	2.0269e-021
2	2	2-2	1.1293e-008	2-2	1.1293e-008	2-2	1.1293e-008
3	2	176-24	2.0534e-006	37-22	1.4554e-006	40-22	1.3276e-006
4	3	16-12	8.3244e-012	15-14	1.1557e-012	16-14	5.4090e-017
5	10	30-15	2.3734e-016	15-15	1.7471e-026	15-15	1.7471e-026
6	6	12-12	2.2877e-003	12-12	2.2877e-003	12-12	2.2877e-003
	9	13-13	1.3998e-006	13-13	1.3998e-006	13-13	1.3998e-006
	12	13-13	4.7224e-010	fail	-	13-13	4.7224e-010
7	4	17-17	2.2513e-005	17-17	2.2513e-005	17-17	2.2513e-005
	10	24-24	7.0877e-005	24-24	7.0877e-005	24-24	7.0877e-005
8	4	7-7	2.6917e-006	7-7	2.6917e-006	7-7	2.6917e-006
	10	19-19	8.8147e-006	19-19	8.8147e-006	19-19	8.8147e-006
9	2	29-6	0	fail	-	31-6	0
10	4	17-9	8.5822e+004	9-9	8.5822e+004	9-9	8.5822e+004
11	3	73-64	2.5811e-013	59-54	4.6277e-012	73-64	2.5811e-013
12	2	18-12	7.6258e-016	22-12	6.9300e-017	18-12	7.6258e-016
	10	18-12	3.8129e-015	19-12	1.8044e-015	18-12	3.8129e-015
	20	18-12	7.6258e-015	20-14	4.0244e-024	18-12	7.6258e-015
13	4	16-16	4.3788e-009	16-16	4.3788e-009	16-16	4.3788e-009
	16	17-17	3.4598e-009	17-17	3.4598e-009	17-17	3.4598e-009
14	2	13-12	8.4347e-020	9-7	4.8175e-014	12-12	8.4347e-020
15	4	28-28	1.2554e-013	29-29	2.0129e-016	28-28	1.2554e-013
16	7	12-8	1.9371e-014	28-10	1.1938e-019	14-8	1.9371e-014
	8	23-13	3.5169e-003	121-23	3.5169e-003	29-18	3.5169e-003
	9	24-16	3.4525e-016	147-31	6.8752e-015	20-13	5.3879e-015
	10	22-12	4.7727e-003	64-15	4.7727e-003	30-19	4.7727e-003
17	2	18-13	9.6635e-028	20-13	8.0643e-028	16-13	9.6635e-028
18	2	29-12	0	22-12	0	17-12	0
19	2	61-10	1.7820e-012	20-10	1.0445e-012	15-10	1.7820e-012
20	2	145-28	1.9979e-001	28-28	1.9979e-001	28-28	1.9979e-001

5. Conclusions

We have combined the preconditioned modified gradient path trust region method with nonmonotone curvilinear search methods to form a new effective algorithm. From the computational results we can know that the new algorithm has some merits. On one hand, it uses the adaptive trust region method supplying an effective initial τ_k for the curvilinear search. On the other hand, curvilinear searches avoid solving the trust region subproblems many times. The idea and approach in this paper can be extended to other curvilinear paths, such as the preconditioned optimal path and the preconditioned conjugate path.

References

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

[2] J.P. Bulteau and J.Ph. Vial, Curvilinear path and trust region in unconstrained optimization: a convergence analysis, *Mathematical Programming Study*, **39** (1987), 82-101.

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

[4] J. Fu and W. Sun, An adaptive approach of conic trust-region method for unconstrained optimization problems, *Journal of Applied Mathematics and Computing,* **19** (2005), 165-177.

[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] L. Grippo, F. Lampariello and S. Lucidi, A class of nonmonotone stabilization methods in unconstrained optimization, *Numer. Math.*, **59** (1991), 779-805.

[7] G. Liu, T.H. Qin and Y.C. Peng, A simple method to construct gradient path for unconstrained optimizaion, *Journal of Xinyang Teachers College*, **11** (1998), 123-126.

[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 Y.X. Yuan, Combining trust region and line search Techniques, *Advances in Nonlinear Programming*, (1998), 153-175.

[10] A. Sartenaer, Automatic determination of an initial trust region in nonlinear programming, *SIAM J. Scientific Computing*, **18** (1997), 1788-1803.

[11] R.B. Schnabel and E. Eskow, A new modified Cholesky factorization, *SIAM Journal on Scientific Computing*, **11** (1990), 1136-1158.

[12] W. Sun, Nonmonotone trust-region method for solving optimization problems, *Applied Mathematics and Computation*, **156** (2004), 159-174.

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

[14] C.X. Xu and J.Z. Zhang, Scaled optimal path trust-region algorithm, *J. Optim. Theory Appl.*, **102** (1999), 127-146.

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

[16] S.B. Yao, B.C. Shi and Y.H. Peng, A nonmonotone trust region algorithm with linesearch, *Journal of Mathematics*, **23** (2003), 290-294.

[17] D.T. Zhu, Nonmonotonic preconditioned curvilinear path algorithms for unconstrained optimization, *A Journal of Chinese Universities, Numerical Mathmatics*, **12** (2003), 99-210.

[18] D.T. Zhu, Curvilinear paths and trust region methods with nonmonotonic back tracking technique for unconstrained optimization, *J. Comput. Math.*, **19** (2001), 241-258.

[19] D.T. Zhu, Curvilinear paths with nonmonotonic inexact line search technique for unconstrained optimization, *Journal of Mathematical Research and Exposition*, **24** (2004), 627-648.

[20] X.S. Zhang, J.L. Zhang and L.Z. Liao, An adaptive trust region method and its convergence, *Science in China (series A)*, **45** (2002), 620-631.