A Nonmonotone trust region method with adaptive radius for unconstrained optimization problems

Masoud Ahookhosh, Keyvan Amini

Department of Sciences, Razi University, Kermanshah, Iran

ARTICLE INFO
Article history:
Received 3 June 2009
Received in revised form 20 April 2010
Accepted 20 April 2010

Keywords:
Unconstrained optimization
Trust region method
Nonmonotone technique
Global convergence
Superlinear convergence
Quadratic convergence

ABSTRACT
In this paper, we incorporate a nonmonotone technique with the new proposed adaptive trust region radius (Shi and Guo, 2008) in order to propose a new nonmonotone trust region method with an adaptive radius for unconstrained optimization. Both the nonmonotone techniques and adaptive trust region radius strategies can improve the trust region methods in the sense of global convergence. The global convergence to first and second order critical points together with local superlinear and quadratic convergence of the new method under some suitable conditions. Numerical results show that the new method is very efficient and robustness for unconstrained optimization problems.

© 2010 Elsevier Ltd. All rights reserved.

1. Introduction
Consider the following unconstrained optimization problem

$$\min_{x \in \mathbb{R}^n} f(x)$$  \hspace{1cm} (1)

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a twice continuously differentiable function. Many iterative methods are proposed for solving (1), that the most of this methods are divided into two general classes: the line search method and trust region method [1,2]. Trust region methods try to find the area around the current step $x_k$ in which a quadratic model agrees with an objective function. In comparison with quasi-Newton methods, trust region methods converge to a point which not only is a stationary point, but also satisfies in a necessary condition. Throughout this paper, we use the following notation:

- $\| \cdot \|$ is the Euclidean norm.
- $g(x) \in \mathbb{R}^n$ and $H(x) \in \mathbb{R}^{n \times n}$ are the gradient and Hessian of $f$ at $x$ respectively.
- $f_k = f(x_k)$, $g_k = g(x_k)$, $H_k = \nabla^2 f(x_k)$ and $B_k$ be a symmetric approximation of $H_k$.

In the standard trust region methods, a trial step $d_k$ has been chosen by solving the following subproblem:

$$\min_{d \in \mathbb{R}^n} m_k(d) = g_k^T d + \frac{1}{2} d^T B_k d, \quad \|d\| \leq \delta_k.$$  \hspace{1cm} (2)

A crucial issue in solving subproblems is a strategy of choosing a trust region radius $\delta_k$ at each iteration. In the standard trust region method, to determine a radius $\delta_k$ and to make a comparison between the model and the objective function, we define the following ratio

$$r_k = \frac{f(x_k) - f(x_k + d_k)}{\text{pred}_k}$$  \hspace{1cm} (3)

* Corresponding author. Tel.: +98 8314274569.
E-mail addresses: ahook.math@gmail.com (M. Ahookhosh), kamini@razi.ac.ir, keyvanamini1353@yahoo.com (K. Amini).

0898-1221/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
where the numerator is called the actual reduction and denominator is called the predicted reduction which is defined by
\[ \text{pred}_k = m_k(0) - m_k(d_k). \]  
(4)
In the case when \( r_k \) is close to 1, it is concluded that there is a good agreement between the model and the objective function over this step, so it is safe to increase the trust region radius to the next iteration. If \( r_k \) is negative or positive but not close to 1, we must shrink the trust region.

One knows that the standard trust region method is very sensitive on initial radius [3–5]. It is also clear that \( \delta_k \) in (2) is independent from any information about \( g_k \) and \( B_k \). This fact causes an increase in the number of subproblems in some problems that need solving which decreases the efficiency of these methods. In [3], Sartenear provided a new strategy for determining the initial trust region radius which prevented the algorithm from the mentioned phenomena. Recently, Zhang et al. [6], in order to reduce the number of subproblems that need solving, proposed another strategy to determine the trust region radius. They used the following adaptive formula
\[ \delta_k = \rho^p \|g_k\|_2 \|\hat{B}_k^{-1}\| \]
for updating the radius of the neighborhood in problem (2), in which \( \rho \in (0, 1) \), \( p \) is a nonnegative integer, and \( \hat{B}_k = B_k + il \) is a positive definite matrix for some \( i \in \mathbb{N} \). Zhang’s method utilized the information of the gradient and Hessian in the current iterate to construct the trust region radius without any initial trust region radius.

Motivated by Zhang’s strategy, Shi and Guo [4] proposed a new adaptive radius for the trust region method. They choose \( \mu, \rho \in (0, 1) \), and \( q_k \) to satisfy the following inequality
\[ -\frac{g_k^T q_k}{\|g_k\|_2 \|q_k\|} \geq \tau \]  
(5)
with \( \tau \in (0, 1) \), and set
\[ s_k = -\frac{g_k^T q_k}{q_k^T \hat{B}_k q_k} \]  
(6)
in which \( \hat{B}_k \) is generated by the procedure: \( q_k^T \hat{B}_k q_k = q_k^T B_k q_k + i \|q_k\|^2 \), and \( i \) is the smallest nonnegative integer such that
\[ q_k^T B_k q_k = q_k^T \hat{B}_k q_k + i \|q_k\|^2 > 0 \]  
(7)
so, they proposed a new trust region radius as follows
\[ \delta_k = \alpha_k \|q_k\| \]  
(8)
where \( \alpha = \rho^p s_k \), and \( p \) is the least positive integer number so that
\[ r_k \geq \mu \]  
(9)
they proved that the new adaptive trust region method has global, superlinear and quadratic convergence properties and is a numerically efficient method.

On the other hand, Chamberlain et al. [7] proposed a watchdog technique for constrained optimization problems, in which some standard line search condition is relaxed to overcome the Marotose effect. Based on this idea, in 1986 Grippo et al. [8], presented a nonmonotone line search technique for solving optimization problems. They also proposed a truncated Newton method with a nonmonotone line search for unconstrained optimization [9]. Due to the high efficiency of nonmonotone techniques, many authors are interested in working on the combination of nonmonotone techniques and trust region methods [12–15.5]. In nonmonotone trust region techniques, the ratio (3) has been changed slightly, which compares the actual reduction with predicted one. In these methods, the new point is compared with the worst point in previous steps which means that these methods are more relaxed. Numerical results of these algorithms show that the nonmonotone trust region methods are more efficient than the monotone trust region methods, especially in the presence of the narrow curved valley.

In 2003, Zhang et al. [5] combined the nonmonotone technique with the adaptive trust region method and obtained good numerical results. Fu and Sun [16] combined Zhang’s adaptive trust region method with another nonmonotone technique, and constructed a new nonmonotone adaptive trust region. Due to the fact that Shi’s adaptive trust region method is more efficient than Zhang’s adaptive trust region method [4], in this paper, we incorporate Shi’s adaptive trust region method with a nonmonotone technique in order to propose the new nonmonotone trust region method with an adaptive radius. We show that our new proposed method has global convergence properties together with the local super linear and quadratic convergence rate under suitable conditions. The new method has been tested on some test problems and compared with some other trust region techniques. Numerical experiments confirm the efficiency and effectiveness of the new proposed method.

This paper organized as follows: In Section 2, we describe our nonmonotone trust region algorithm with an adaptive radius. In Section 3, we first prove that the new algorithm is well defined, and then the global convergence is investigated. Section 4 is devoted to verifying the local superlinear and quadratic convergence results. The second order necessary condition is proved. Numerical results are given in Section 5 in order to indicate that the algorithm is very efficient. Finally, some conclusions are delivered in Section 6.
2. New algorithm

In this section, we describe a new nonmonotone trust region method with an adaptive radius and give some properties of the new algorithm. First, we define

\[ f_{0(k)} = \max_{0 \leq j \leq n(k)} \{ f_{j-1} \}, \quad k = 0, 1, 2, \ldots \]

where \( n(k) = \min\{N, k\} \) and \( N \geq 0 \) is an integer constant. The new trust region method, at the current iterate \( x_k \), needs to solve the following subproblem:

\[ \min_{d \in \mathbb{R}^n} m_k(d) = g_k^T d + \frac{1}{2} d^T B_k d, \quad \|d\| \leq \delta_k \]  

(11)

similar to [17], we solve (11) inaccurately such that

\[ \text{pred}_k \geq \beta \|g_k\| \min \left\{ \delta_k, \frac{\|g_k\|}{\|B_k\|} \right\}. \]

(12)

Now, we can outline our new nonmonotone trust region algorithm with an adaptive radius as follows:

**Algorithm 2.1**  
(A new nonmonotone trust region with adaptive radius).

Step 1. An initial point \( x_0 \in \mathbb{R}^n \) and a symmetric matrix \( B_0 \in \mathbb{R}^{n \times n} \) are given. The constants \( 0 < \mu < 1, 0 < \rho < 1, N \geq 0 \) and \( \epsilon > 0 \) are also given. Compute \( f(x_0) \) and set \( k = 0 \) and \( p = 0 \).

Step 2. Compute \( g_k \). If \( \|g_k\| \leq \epsilon \), stop.

Step 3. Choose \( q_k \) to satisfy (5).

Step 4. Solve (11) to determine \( d_k \), and set \( \hat{x}_{k+1} = x_k + d_k \).

Step 5. Compute \( n(k), f_{0(k)} \) and \( \text{pred}_k \). Set

\[ \hat{r}_k = \frac{f_{0(k)} - f(x_k + d_k)}{\text{pred}_k}. \]

If \( \hat{r}_k < \mu \), then set \( p = p + 1 \) and go to Step 3.

Step 6. Set \( x_{k+1} = \hat{x}_{k+1} \), \( p = 0 \), generate \( B_{k+1} \), \( k = k + 1 \) and go to Step 2.

Note that \( B_k \) can be generated by a quasi Newton updating formula. We also note that, in the case of \( N = 0 \), the new algorithm is reduced to the adaptive trust region algorithm proposed by Shi and Guo [4].

Throughout this paper, we consider the following assumptions in order to analyze the new trust region algorithm:

(H2) \( B_k \) is uniformly bounded, i.e., \( \exists 0 \leq \delta_k \) such that \( \|B_k\| \leq M \), for all \( k \in \mathbb{N} \cup \{0\} \).

**Remark 2.1.** If \( f(x) \) is a twice continuously differentiable function and the level set \( L(x_0) \) is bounded, then (H2) implies that \( \|\nabla^2 f(x)\| \) is uniformly continuous and bounded on the open bounded convex set \( \Omega \) that contains \( L(x_0) \). Hence, there exists a constant \( M_1 > 0 \) such that \( \|\nabla^2 f(x)\| \leq M_1 \) and by using the Mean Value Theorem we have

\[ \|g(x) - g(y)\| \leq M_1 \|x - y\|, \quad \forall x, y \in \Omega. \]

Moreover, if \( g(x) \) is Lipschitz continuous on \( \Omega \), then (H2) holds. Therefore, assumption (H2) is weaker than the assumptions that are used in literature [6,5].

**Remark 2.2.** (H2) and the generating procedure of \( \hat{B}_k \) implies that \( \{\hat{B}_k\} \) is also uniformly bounded. In fact, if \( \|B_k\| \leq M \), for all \( k \), then \( \|\hat{B}_k\| = \|B_k + iI\| \leq 2M + 1 \) because \( M < i \leq M + 1 \) implies that \( q_k^T \hat{B}_k q_k = q_k^T B_k q_k + i\|q_k\|^2 > 0 \).

**Lemma 2.1.** For all \( k \in \mathbb{N} \), we have

\[ \text{pred}_k \geq -m_k(\alpha_k q_k) \geq -\frac{1}{2} \alpha_k g_k^T q_k \]

where \( q_k \) is an optimal solution of the subproblem (11) with respect to \( \alpha_k \leq s_k \).

**Proof.** See Shi and Guo [4]. \( \square \)

**Lemma 2.2.** If \( \text{pred}_k \) indicate the predicted reduction, Then

\[ |f(x_k) - f(x_k + d_k) - \text{pred}_k| \leq O(\|d_k\|^2). \]

**Proof.** See Conn et al. [18]. \( \square \)

**Lemma 2.3.** Suppose that (H1) and (H2) hold. Then steps 4 and 5 of the new algorithm are well-defined, i.e., in each iteration, these steps are terminated after finite iterates.
Proof. First, we prove that when $p$ is sufficiently large, (9) holds. Let $d_k^i$ be the solution of subproblem (2) corresponding to $p = i$ at $x_k$, and $\text{pred}_{k(i)}$ be the predicted reduction corresponding to $p = i$ at $x_k$. It follows from Lemma 2.1 that

$$\text{pred}_k \geq -m_k(\alpha_k q_k) \geq -\frac{1}{2}\alpha_k g_k^T q_k.$$  

Using this inequality and Lemma 2.2, we have

$$\frac{|f(x_k) - f(x_k + d_k)|}{\text{pred}_{k(i)}} - 1 = \frac{|f(x_k) - f(x_k + d_k) - \text{pred}_{k(i)}|}{\text{pred}_{k(i)}} \leq \frac{O(||d_k^i||^2)}{2\alpha_k g_k^T q_k} \leq \frac{O(\delta_{k(i)}^2)}{2\delta_{k(i)} g_k^T q_k / ||q_k||} = \frac{O(\delta_{k(i)})}{2g_k^T q_k / ||q_k||}\]$$

where the last inequality is obtained using (8) and (11). Now, as $i \to \infty$, then $\alpha_{k(i)} = \rho^i s_k ||q_k|| \to 0$ and consequently, using (8), the right hand side of the preceding inequality tends to zero. Which implies that for $p$ sufficiently large (9) holds. Now, using (10), we have

$$\hat{r}_k = \frac{f_{l(k)} - f(x_k + d_k)}{\text{pred}_k} \geq \frac{f(x_k) - f(x_k + d_k)}{\text{pred}_k} \geq \mu.$$  

Therefore, when $p$ is sufficiently large, $\hat{r}_k \geq \mu$. This implies that steps 4 and 5 of the new trust region algorithm are well-defined. \[\Box\]

Lemma 2.4. Suppose that (H1) and (H2) hold, and the sequence $\{x_k\}$ generated by the new algorithm, then we have

$$\{x_k\} \subset L(x_0).$$

Proof. We proceed by induction. The result evidently holds for $k = 0$. Assume that $x_k \in L(x_0)$, then we show that $x_{k+1} \in L(x_0)$. From definition of algorithm, we have that $\hat{r}_k \geq \mu > 0$, so by $\text{pred}_k \geq 0$, we have

$$f_{l(k)} \geq f_{k+1} + \mu \text{pred}_k \geq f_{k+1}$$

because of $l(k) \leq k$, from induction hypothesis, we know that $f_{l(k)} \leq f_0$, so by (13) we have

$$f_{k+1} \leq f_0.$$  

Therefore $x_{k+1} \in L(x_0).$ Which completes the proof. \[\Box\]

Lemma 2.5. Suppose that (H1) and (H2) hold, then the sequence $\{f_{l(k)}\}$ is not monotonically increasing. Therefore the sequence $\{f_{l(k)}\}$ is convergent.

Proof. Since $x_{k+1}$ is accepted by the algorithm, we have

$$\hat{r}_k = \frac{f_{l(k)} - f(x_k + d_k)}{\text{pred}_k} \geq \mu$$

so,

$$f_{l(k)} - f(x_k + d_k) \geq \mu \text{pred}_k \geq 0, \quad \forall k$$

therefore, we will have

$$f_{l(k)} \geq f_{k+1}, \quad \forall k.$$  

Now, we consider the following two cases:

(i) $k \geq N$,

Using (10) and (14), we can write

$$f_{l(k+1)} = \max_{0 \leq j \leq n(k+1)} \{f_{k-j+1}\} \leq \max\{f_{k+1}, f_k, \ldots, f_{k-N+1}\} \leq \max\{f_{l(k)}, f_{k+1}\} \leq f_{l(k)}.$$
Suppose that $k < N$.

It is clear that $n(k) = k$ and, for any $k, f_k \leq f_0$. Hence

$$f_{l(k)} = f_0.$$ 

Therefore, in both cases, the sequence $\{f_{l(k)}\}$ is not monotonically increasing. Moreover, (H1) and Lemma 2.4 imply that $f_{l(k)}$ is bounded. Thus, $f_{l(k)}$ is convergent. \hfill \Box

### 3. Convergence analysis

As we stated in introduction, the trust region methods have good convergence properties [18,1,10,2,19–21]. These methods have the global convergence property together with superlinear and quadratic convergence rate under some mild conditions. In this section, we discuss some convergence properties of the new trust region algorithm, and prove the global convergence.

**Lemma 3.1.** Suppose that $\{x_k\}$ is generated by the Algorithm 2.1 and $\|d_k\| \leq \delta_k$, then there exists a positive scalar $\bar{c}$ such that

$$\|d_k\| \leq \bar{c}\|g_k\|. \quad (15)$$

**Proof.** From the definition of $\alpha_k$ and (11), we have that $\|d_k\| \leq \rho^p s_k \|q_k\|$. By (7), we also have that $q_k^T B_k q_k > 0$, so there exists a sufficiently small $\lambda \in (0, 1)$ such that

$$0 < \lambda \|q_k\|^2 \leq q_k^T B_k q_k$$

thus,

$$\|d_k\| \leq \rho^p s_k \|q_k\| = -\rho^p \frac{q_k^T g_k}{q_k^T B_k q_k} \|q_k\| \leq -\rho^p \frac{\bar{c} q_k^T g_k}{\lambda \|q_k\|^2} \leq \frac{\rho^p}{\lambda} \|g_k\|$$

where the last inequality is obtained by applying the Cauchy inequality. Setting $\bar{c} = \frac{\rho^p}{\lambda}$ completes the proof of lemma. \hfill \Box

**Lemma 3.2.** Suppose that $\{x_k\}$ is generated by the Algorithm 2.1, then we have

$$\lim_{k \to \infty} f(x_{l(k)}) = \lim_{k \to \infty} f(x_k). \quad (16)$$

**Proof.** If $x_{k+1}$ be a successful iteration, we can write

$$\frac{f_{l(k)} - f(x_k + d_k)}{pred_k} \geq \mu$$

thus,

$$f_{l(k)} - f(x_k + d_k) \geq \mu pred_k. \quad (17)$$

By replacing $k$ with $l(k) - 1$, we have

$$f_{l(k) - 1} - f_{l(k)} \geq \mu pred_{l(k) - 1}.$$ 

This inequality together with Lemma 2.5 imply that

$$\lim_{k \to \infty} pred_{l(k) - 1} = 0. \quad (18)$$

Now according to Lemma 3.1, (12), and (H2) we obtain

$$\text{pred}_{l(k) - 1} \geq \beta \|g_{l(k) - 1}\| \min \left\{ \delta_{l(k) - 1}, \frac{\|g_{l(k) - 1}\|}{\|B_{l(k) - 1}\|} \right\} \geq \beta \|g_{l(k) - 1}\| \min \left\{ \|d_{l(k) - 1}\|, \frac{\|d_{l(k) - 1}\|}{\bar{c} M} \right\} \geq \frac{\beta}{\bar{c} M} \|d_{l(k) - 1}\|^2 \min \left\{ 1, \frac{1}{\bar{c} M} \right\} = \kappa \|d_{l(k) - 1}\|^2$$

where $\kappa = \frac{\beta}{\bar{c} M} \min \left\{ 1, \frac{1}{\bar{c} M} \right\}$. Therefore, from (18), we have

$$\lim_{k \to \infty} \|d_{l(k) - 1}\| = 0. \quad (19)$$
Using uniform continuity of \( f(x) \), (19) imply that
\[
\lim_{k \to \infty} f(x_{\hat{l}(k)}) = \lim_{k \to \infty} f(x_{\hat{l}(k) - 1}). \tag{20}
\]

Now we define \( \hat{l}(k) = l(k + N + 2) \). By induction, for all \( j \geq 1 \), we can prove
\[
\lim_{k \to \infty} \|d_{\hat{l}(k) - j}\| = 0. \tag{21}
\]

For \( j = 1 \), since \( \{\hat{l}(k)\} \subseteq \{l(k)\} \), then (21) follows from (19). Assume that (21) holds for given \( j \). We prove (21) for \( j + 1 \). Let \( k \) be large enough so that \( \hat{l}(k) - (j + 1) > 0 \). Using (17) and substituting \( k \) with \( \hat{l}(k) - j - 1 \), we have
\[
f(x_{\hat{l}(k) - j - 1}) - f(x_{\hat{l}(k) - j}) \geq \mu \text{ pred}_l(\hat{l}(k) - j - 1).
\]

Following the same arguments for deriving (19), we deduce that
\[
\lim_{k \to \infty} \|d_{\hat{l}(k) - j - 1}\| = 0.
\]

so,
\[
\lim_{k \to \infty} f(x_{\hat{l}(k) - j - 1}) = \lim_{k \to \infty} f(x_{\hat{l}(k)}).
\]

Therefore (21) holds. Like (20), for any given \( j \geq 1 \), we have that \( \lim_{k \to \infty} f(x_{\hat{l}(k) - j}) = \lim_{k \to \infty} f(x_{\hat{l}(k)}) \).

On the other hand, for any \( k \), we know that
\[
x_{k+1} = x_{\hat{l}(k)} - \sum_{j=1}^{\hat{l}(k) - j - 1} d_{\hat{l}(k) - j}.
\]

Note that \( \hat{l}(k) - j - 1 \leq N + 1 \), and using (21) we have
\[
\lim_{k \to \infty} \|x_{k+1} - x_{\hat{l}(k)}\| = 0.
\]

Therefore, from the uniform continuity of \( f(x) \), we get
\[
\lim_{k \to \infty} f(x_{\hat{l}(k)}) = \lim_{k \to \infty} f(x_{\hat{l}(k)}) = \lim_{k \to \infty} f(x_{\hat{l}(k)}).
\]

So the proof is completed. \( \square \)

**Theorem 3.3.** Suppose that \( (H1) \) and \( (H2) \) hold, then the Algorithm 2.1 either stops at stationary point of (1) or generates an infinite sequence \( \{x_k\} \) such that
\[
\lim_{k \to \infty} \frac{g_k^T q_k}{\|q_k\|} = 0. \tag{22}
\]

**Proof.** If the Algorithm 2.1 doesn’t stop at a stationary point, we prove that (22) holds. Suppose that the Algorithm 2.1 generates the sequence \( \{x_k\} \) and
\[
\lim_{k \to \infty} \frac{g_k^T q_k}{\|q_k\|} \neq 0
\]

which implies that there exist a \( \epsilon_0 > 0 \) and an infinite subset \( K \subseteq \{0, 1, 2, \ldots\} \), such that
\[
\frac{g_k^T q_k}{\|q_k\|} \geq \epsilon_0, \quad \forall k \in K. \tag{23}
\]

From Remark 2.2, we know that
\[
\exists M_0 > 0 \quad \text{s.t.} \|\hat{B}_k\| \leq M_0, \quad \forall k
\]

thus,
\[
g_k^T \hat{B}_k q_k \leq M_0 \|q_k\|^2, \quad \forall k. \tag{24}
\]

Now, let \( K_1 = \{k \in K | \alpha_k = s_k\} \) and \( K_2 = \{k \in K | \alpha_k < s_k\} \). Obviously, we have that \( K = K_1 \cup K_2 \) is an infinite subset of the set \( \{0, 1, 2, \ldots\} \). We prove that neither \( K_1 \) nor \( K_2 \) can be an infinite set which contradicts (23).
We let that $K_1$ be an infinite subset of $K$. by using of Lemma 2.1 and (24), we have
\[
    f(x_k) - f(x_k + d_k) \geq \mu \text{pred}_k \geq -\frac{1}{2} \mu \alpha_k g_k^T q_k \geq -\frac{1}{2} \mu \alpha_k \frac{g_k^T q_k^2}{q_k^T B q_k} \geq \frac{\mu}{2M_0} \left(\frac{g_k^T q_k}{\|q_k\|}\right)^2 \geq \frac{\mu}{2M_0} \epsilon_0^2, \quad k \in K_1.
\]
As $k \to \infty$, this inequality together with Lemma 3.2 implies
\[
    0 \geq \frac{\mu}{2M_0} \epsilon_0^2
\]
which is a contradiction and shows that $K_1$ cannot be an infinite subset of $K$.

Now, let $K_2$ be an infinite subset of $K$. Lemma 2.1 implies
\[
    f(x_k) - f(x_k + d_k) \geq \mu \text{pred}_k \geq -\frac{1}{2} \mu \delta_k \frac{g_k^T q_k}{\|q_k\|} \geq \frac{1}{2} \mu \delta_k \epsilon_0.
\]
As $k \to \infty$, this inequality together with Lemma 3.1 give us
\[
    \lim_{k \to \infty} \delta_k = 0, \quad k \in K_2.
\]

Now, suppose that $\tilde{d}_k$ is an optimal solution of the following subproblem
\[
    \min_{d \in \mathbb{R}^n} g_k^T d_k + \frac{1}{2} d_k^T B_k d_k, \quad \|d_k\| \leq \tilde{\delta}_k, \quad \tilde{\delta}_k = \delta_k / \rho.
\]
Then, following the steps of Algorithm 2.1, we have
\[
    \frac{f(x_k) - f(x_k + \tilde{d}_k)}{\text{pred}_k} < \mu, \quad k \in K_2.
\]
On the other hand, (25) implies that
\[
    \lim_{k \to \infty} \tilde{\delta}_k = 0, \quad k \in K_2.
\]
Now, using Lemma 2.1, (23) and (27), we have
\[
    \left| \frac{f(x_k) - f(x_k + d_k)}{\text{pred}_k} - 1 \right| = \left| \frac{f(x_k) - f(x_k + d_k) - \text{pred}_k}{\text{pred}_k} \right| \leq O(\|d_k\|^2) \leq -\frac{1}{2} \tilde{\delta}_k g_k^T q_k \leq \frac{O(\tilde{\delta}_k^2)}{\tilde{\delta}_k^2} \leq \frac{O(\tilde{\delta}_k^2)}{\tilde{\delta}_k^2} \epsilon_0 \rightarrow 0 \quad \text{as} \ k \to \infty
\]
where $k \in K_2$. Therefore, in this case the monotone ratio is well-defined, so due to the following inequality we can indicate that the nonmonotone ratio is also well-defined.
\[
    \frac{f(x_k) - f(x_k + d_k)}{\text{pred}_k} \geq \frac{f(x_k) - f(x_k + d_k)}{\text{pred}_k} \geq \mu.
\]
However, for a sufficiently large $k \in K_2$, (28) contradicts (26) and shows that $K_2$ can not to be an infinite subset of $K$. Therefore there exists no infinite subset of $K$ such that (23) holds, so the proof is completed. □
Theorem 3.4. Suppose that conditions of Theorem 3.3 hold and \( q_k \) satisfies (5), then the Algorithm 2.1 either stops finitely or generates an infinite sequence \( \{x_k\} \) such that

\[
\lim_{k \to \infty} \|g_k\| = 0
\]

Proof. If the Algorithm 2.1 stops after finite iterations, then the proof is completed. Otherwise, Theorem 3.3 indicates that the algorithm generates an infinite sequence \( \{x_k\} \) such that satisfies (22) and since \( q_k \) satisfies (5) we have

\[
0 \leq \tau \|g_k\| \leq -\frac{g_k^T q_k}{\|g_k\| \|q_k\|} \|g_k\| = -\frac{g_k^T q_k}{\|q_k\|} \to 0, \quad k \to \infty.
\]

Therefore, we have \( \lim_{k \to \infty} \|g_k\| = 0 \). This completes the proof of theorem. \( \square \)

4. Convergence rate analysis

In this section, we first prove that the new algorithm has both a superlinear and quadratic convergence rate under some suitable conditions. We then investigate the second order necessary condition in the sequel.

In this section, we need \( q_k = -B_k^{-1}g_k \) satisfy in (5). For this purpose, we need to make an additional assumption as follows: \( (H3) \) Matrix \( B_k \) is a uniformly bounded number.

Theorem 4.1. Suppose that \((H1)-(H3)\) hold, \( q_k = -B_k^{-1}g_k \), and the sequence \( \{x_k\} \) is generated by the Algorithm 2.1 converges to \( x^* \). Also suppose that \( H(x) = \nabla^2 f(x) \) is continuous in a neighborhood \( N(x^*, \epsilon) \) of \( x^* \), and \( H(x) \) and \( B_k \) are uniformly positive definite matrices such that

\[
\lim_{k \to \infty} \frac{\|B_k - H(x^*)\|q_k\|}{\|q_k\|} = 0
\]

then the sequence \( \{x_k\} \) converges to \( x^* \) superlinearly.

Proof. For sufficiently large \( k \) using definition of \( \hat{d}_k \), we have that \( \hat{B}_k = B_k \), and it is obvious that \( s_k = 1 \). Therefore, \( \hat{d}_k = \rho^p \|q_k\| \), so \( \hat{d}_k = q_k \) is a feasible solution of subproblem, for \( p = 0 \). By using (29), we get

\[
\lim_{k \to \infty} \frac{\|g_k + H(x)^T \hat{d}_k\|}{\|\hat{d}_k\|} = \lim_{k \to \infty} \frac{\|B_k - H(x^*)\|\hat{d}_k\|}{\|\hat{d}_k\|} = 0
\]

which implies that \( [B_k - H(x^*)]\hat{d}_k = o(\|\hat{d}_k\|) \) and hence \( [H_k - H(x^*)]\hat{d}_k = o(\|\hat{d}_k\|) \). Thus, we can write

\[
[B_k - H_k]\hat{d}_k = o(\|\hat{d}_k\|)
\]

and,

\[
\hat{d}_k = -H(x^*)^{-1} g_k + o(\|\hat{d}_k\|)
\]

thus,

\[
\|\hat{d}_k\| \leq \|H(x^*)^{-1}\| \|g_k\| + o(\|\hat{d}_k\|).
\]

Therefore, we have

\[
\frac{\|g_k\|}{\|\hat{d}_k\|} \geq \frac{1}{\|H(x^*)^{-1}\|} + o(\|\hat{d}_k\|) \|\hat{d}_k\|.
\]

Theorem 3.4 implies that \( g_k \to 0 \) as \( k \to \infty \). On the other hand right hand side of (31) is strictly positive, so \( \hat{d}_k \to 0 \) as \( k \to \infty \). By Lemma 2.1 and this fact that \( -g_k^T q_k = g_k^T B_k q_k \), we have

\[
\text{pred}_k \geq -\frac{1}{2} g_k^T B_k q_k \geq -\frac{1}{2} g_k^T q_k.
\]

Now, Using (30) and Taylor theorem, we can deduce that

\[
|f(x_k) - f(x_k + \hat{d}_k) - \text{pred}_k| = |f_k - \left[ f_k + g_k^T \hat{d}_k - \frac{1}{2} \hat{d}_k^T H_k \hat{d}_k + o(\|\hat{d}_k\|^2) \right] + \left( g_k^T \hat{d}_k + \frac{1}{2} \hat{d}_k^T B_k \hat{d}_k \right)|
\]

\[
\leq \frac{1}{2} \|\hat{d}_k\| \|B_k - H_k\| \|\hat{d}_k\| + o(\|\hat{d}_k\|^2) \leq \frac{1}{2} \|\hat{d}_k\| \|B_k - H_k\| \|\hat{d}_k\| + o(\|\hat{d}_k\|^2)
\]

\[
\leq \frac{1}{2} \|\hat{d}_k\| o(\|\hat{d}_k\|) + o(\|\hat{d}_k\|^2) = o(\|\hat{d}_k\|^2)
\]
thus, we get
\[
\frac{f(x_k) - f(x_k + \hat{d}_k)}{\text{pred}_k} - 1 = \frac{f(x_k) - f(x_k + \hat{d}_k) - \text{pred}}{\text{pred}_k} \\
\leq \frac{o(||\hat{d}_k||^2)}{\text{pred}_k} \leq \frac{o(||\hat{d}_k||^2)}{\frac{q_k^T B_k q_k}{2}} \\
\leq \frac{o(||\hat{d}_k||^2)}{\frac{q_k^T q_k}{||\hat{d}_k||^2}} = \frac{o(||\hat{d}_k||^2)}{||\hat{d}_k||^2} \to 0 \quad \text{as } k \to \infty.
\]

So, for a sufficiently large \( k \), we have
\[
\frac{f(x_k) - f(x_k + \hat{d}_k)}{\text{pred}_k} \geq \frac{f(x_k) - f(x_k + \hat{d}_k)}{\text{pred}_k} \geq \mu.
\]

Therefore \( x_{k+1} = x_k + \hat{d}_k \), for a sufficiently large \( k \), and the new trust region method reduce to standard quasi-Newton method. One knows that the quasi-Newton methods, in the presence of (29), converge superlinearly \([1,2]\). So the sequence \( \{x_k\} \) converges to \( x^* \) superlinearly.

**Theorem 4.2.** Suppose that (H1)–(H3) hold, \( q_k \) satisfies (5), the Algorithm 2.1 generates an infinite sequence \( \{x_k\} \) such that \( x_k \to x^* \) as \( k \to \infty \), \( H(x) \) is Lipschitz continuous and uniformly positive definite matrices in a neighborhood \( N(x^*, \epsilon) \) of \( x^* \), \( B_k = H(x_k) \) and \( q_k = -H_k^{-1} g_k \). Then the sequence \( \{x_k\} \) converges to \( x^* \) quadratically.

**Proof.** Since the matrix \( H(x) \) is Lipschitz continuous, we have
\[
||H(x) - H(x^*)|| \leq L(\epsilon)||x - x^*|| \quad \forall x \in N(x^*, \epsilon)
\]
hence, (29) holds. So all conditions of Theorem 4.1 hold, thus similar to Theorem 4.1 we can prove that \( q_k = \hat{d}_k \to 0 \), as \( k \to \infty \). Now, from Lemma 2.1, we have
\[
\text{pred}_k \geq \frac{1}{2} \alpha_k g_k^T q_k \geq \frac{q_k^T H_k q_k}{2}.
\]

Similar to proof strategy of Theorem 4.1, for a sufficiently large \( k \), we get
\[
\frac{f(x_k) - f(x_k + \hat{d}_k)}{\text{pred}_k} \geq \frac{f(x_k) - f(x_k + \hat{d}_k)}{\text{pred}_k} \geq \mu.
\]

Therefore, the new algorithm reduce to standard Newton method for a sufficiently large \( k \). Thus, the sequence \( \{x_k\} \) converges to \( x^* \) quadratically.

**Theorem 4.3.** Suppose that (H1) and (H2) hold, \( B_k = H_k \), and infinite sequence \( \{x_k\} \) is generated by the Algorithm 2.1. If sequence \( \{x_k\} \) converges to \( x^* \), then \( H(x^*) \) is positive semidefinite matrix, i.e., \( x^* \) satisfies the second order necessary condition.

**Proof.** Assume that \( \lambda^k_1 \) and \( \lambda^k \) are the smallest eigenvalues of \( H_k \) and \( H(x^*) \), respectively. Let \( z_k \) be a normalized eigenvector of \( H_k \) corresponding to the eigenvalue \( \lambda^k_1 \) and \( z_k^T g_k \leq 0 \), \( H_k z_k = \lambda^k_1 z_k \). Suppose that \( H(x^*) \) is not a positive semidefinite matrix, then \( \lambda^* < 0 \) and thus \( \lambda^k_1 < 0 \) for a sufficiently large \( k \). Since \( \delta_k, ||z_k|| = \alpha_k ||q_k|| \), it follows that \( \delta_k z_k \) is a feasible solution to (11). Therefore
\[
- m_k(\delta_k z_k) = - \left( \delta_k g_k^T z_k + \frac{1}{2} \delta^2_k z_k^T B_k z_k \right) \geq - \frac{1}{2} \delta^2_k z_k^T B_k z_k \\
= - \frac{1}{2} \delta^2_k z_k^T H_k z_k = - \frac{1}{2} \delta^2_k z_k^T \lambda^k_1 z_k = - \frac{1}{2} \delta^2_k \lambda^k_1.
\]

Now, using (33) and \( \hat{\gamma} \geq \mu \), we have
\[
f_k - f_{k+1} \geq - \mu m_k(\delta_k z_k) \geq - \frac{1}{2} \mu \delta^2_k \lambda^k_1.
\]

This inequality together with Lemma 3.2 and the fact that \( \lambda^k_1 \to \lambda^* \), as \( k \to \infty \), imply that
\[
\lim_{k \to \infty} \delta_k = 0.
\]

Considering (33) and similar to a procedure for deriving (26) and (28), we get a contradiction. So \( \lambda^* \geq 0 \) and thus \( H(x^*) \) is a positive semidefinite matrix.
that then the new method is more efficient than the adaptive trust region methods. Trust region methods failed for some problems, but then the new method is successful for all test problems. So we can conclude that the new algorithms are compared with NMATR-Z and NMATR-S. The related numerical results are given in Table 1.

### 5. Numerical results

In this section, we present computational results to illustrate the performance of the new trust region method in comparison with other versions of trust region methods. As we have seen, there are different choices of $q_k$ that determine different adaptive radius. Two popular choices of $q_k$ are: $q_k = -g_k$, which is a natural choice, and $q_k = -B_k^{-1}g_k$, which lead us to some interesting convergence properties, as we have mentioned in Section 4.

In the following tables, $n_i$ and $n_f$ represent the number of iterations and function evaluations, respectively. All test problems are selected from Moré et al. [11]. We have implemented the algorithms MATLAB 7.4 on a 3.0 GHz Intel Pentium IV WinXP PC with 1 GB RAM with double precision format. In entire algorithms, we update $B_k$ by the BFGS formula, and the stopping criterion is $\|g_k\| \leq \epsilon$, where $\epsilon = 10^{-8}$. We choose $\mu = 0.1$, $N = 2n$, where $n$ is the dimension of test problems. We declared failure when the algorithm was not convergent in the first 1000 iterations or $B_k$ was a singular matrix. The quadratic subproblems are solved by the nearly exact solution method [2], and in order to compare the algorithms, the related subproblems are solved by the same subroutine.

For convenience, we represent algorithms with the following notations:

- **STR**: Standard trust region method;
- **ATR-Z**: Zhang’s adaptive trust region method;
- **ATR-G**: Shi’s adaptive trust region method with $q_k = -g_k$;
- **ATR-N**: Shi’s adaptive trust region method with $q_k = -B_k^{-1}g_k$;
- **NMATR-Z**: Zhang’s nonmonotone adaptive trust region method;
- **NMATR-S**: Sun’s nonmonotone adaptive trust region method;
- **NMATR-G**: New nonmonotone adaptive trust region method with $q_k = -g_k$;
- **NMATR-N**: New nonmonotone adaptive trust region method with $q_k = -B_k^{-1}g_k$.

One knows that the standard trust region is very sensitive on initial radius [3,4,6]. Table 1 provides a comparison between the new nonmonotone adaptive trust region algorithms, NMATR algorithms, the standard trust region algorithm with a different initial radius. In this table, for these experiments the radius of the standard trust region method is determined by:

$$
\delta_{k+1} = \begin{cases} 
\min\{\delta_k/4, \|d_k\|\}, & \text{if } r_k < \mu \\
\max\{2\delta_k, 4\|d_k\|\}, & \text{if } r_k \geq \mu.
\end{cases}
$$

On the other hand, we know that the adaptive trust region methods have better numerical result in comparison with the standard trust region methods. In Table 2, we have compared the new algorithms with the ATR-Z, ATR-N and ATR-G. Finally, the new algorithms are compared with NMATR-Z and NMATR-S. The related numerical results are given in Table 3.

At a glance in Table 1, one can see a vacillation in the number of iterations in the standard trust region method when the initial radius is changed. Moreover, it has been stated that the standard trust region method not only is very sensitive on the initial radius but also fails for some problems. In contrast, the NMATR-N and NMATR-G methods are convergent for all test problems and have better numerical results than the standard trust region method, in most cases.

In Table 2, we can see that the number of iterations in the new method is less than the number of iterations in adaptive trust region methods for some problems, especially for problems 2, 5, 7, 8, 10, 15, 17, 18. The main point is: the adaptive trust region methods failed for some problems, but the new method is successful for all test problems. So we can conclude that the new method is more efficient than the adaptive trust region methods.
the other trust region methods for solving unconstrained optimization problems. We provided the preliminary numerical experiments to indicate that the new method is more efficient and robustness than order critical points. Moreover, we can prove the superlinear and quadratic convergence rate of the new method. Finally, methods. Theoretical analysis exhibited that the new proposed method has a global convergence to first order and second order critical points. By a different choice of $q_k$, one can give the different trust region methods. By comparison of cpu time, we can conclude that the cpu time of the new method is considerably less than the other nonmonotone trust region methods. These facts lead us to conclude that the new method is more efficient than the other nonmonotone trust region methods.

### 6. Conclusions

In this paper, we combine the nonmonotone technique with adaptive trust region radius strategy to propose a nonmonotone trust region method with an adaptive radius. In the proposed method, the trust region radius can be adjusted automatically according to the current information. By a different choice of $q_k$, one can give the different trust region methods. Theoretical analysis exhibited that the new proposed method has a global convergence to first order and second order critical points. Moreover, we can prove the superlinear and quadratic convergence rate of the new method. Finally, we provided the preliminary numerical experiments to indicate that the new method is more efficient and robustness than the other trust region methods for solving unconstrained optimization problems.
References