International Journal of Recent Engineering Science (IJRES),
ISSN: 2349-7157, *Volume 2 Issue 5 September to October 2015*

Non-monotone Trust Region Method Combined with Wolfe Line Search Strategy for Unconstrained Optimization

Changyuan Li[a], Qinghua Zhou[b] and Xiao Wu[c]
College of Mathematics and Information Science, Hebei University,
Baoding, 071002, Hebei Province, China

Abstract

In this paper, we propose and analyze a new trust region algorithm for unconstrained optimization problems which is combining a new non-monotone trust region method with non-monotone Wolfe line search technique. The new algorithm solves the trust region sub-problem only once at each iteration. The global convergence of the new algorithm is proved under some mild conditions.

Keywords: *unconstrained optimization, non-monotone trust region method, non-monotone Wolfe line search, global convergence*

I. INTRODUCTION

Consider the following unconstrained optimization problem:

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

where $f : R^n \to R$ is a twice continuously differentiable function.

For solving (1), trust region methods usually compute d_k by solving the quadratic sub-problem:

$$\min \ m_k(d) = f_k + g_k^T d + \frac{1}{2} d^T B_k d, \quad \|d\| \le \Delta_k. \tag{2}$$

where $f_k = f(x_k)$ and $g_k = \nabla f(x_k)$ are the function value and the gradient vector at the current approximation iterate x_k, respectively, B_k is an $n \times n$ symmetric matrix which may be the exact Hessian $H(x_k)$ or the quasi-Newton approximation and $\Delta_k > 0$ is the trust region radius. In this paper, the notation $\|\cdot\|$ denotes the Euclidean norm on R^n. Some criteria are used to determine whether a trial step d_k is accepted. If not, the sub-problem (2) may be computed several times at each iterate until an acceptable step is found. There is no doubt that the repetitive process will increase the cost to solve the problem.

In order to overcome the above drawback, Nocedal and Yuan [1] put forward a algorithm which combining trust region algorithm and line search method for the first time in 1998, then Gertz [2] proposed a new trust region algorithm that use Wolfe line search at each iteration to obtain a new iteration point regardless of whether d_k is accepted. Both of them improved the computational efficiency by fully using the advantages of two kinds of algorithm.

Algorithms mentioned above are monotonic algorithm. One of the interesting methods for improving the algorithm in optimization is non-monotone techniques. The first non-monotone optimization technique is the "watchdog technique", introduced by Chamberlain et al. in [3] in 1982, in order to overcome the Maratos effect. Motivated by this idea, Grippo et al. present a non-monotone line search technique for solving optimization problems in [4]. They also proposed a truncated Newton method with a non-monotone line search for unconstrained optimization [6]. Besides Deng et al. [5] proposed a non-monotone trust region algorithm in which they combined non-monotone term and trust region method for the first time. Due to the high efficiency of non-monotone techniques, many authors are interested in working on the non-monotone techniques for unconstrained optimization problem [7-12]. In this paper, we present a new non-monotone trust region algorithm combining with non-monotone Wolfe-type line search strategy.

II. NON-MONOTONE TERM AND WOLFE-TYPE LINE SEARCH CONDITION

The general non-monotone form is as follows:

$$f_{l(k)} = f(x_{l(k)}) = \max_{0 \le j \le m(k)} \{f_{k-j}\}, \quad k = 0, 1, 2, \dots$$

where $m(0) = 0,\ 0 \le m(k) \le \min\{M, m(k-1)+1\}$ and $M \ge 0$ is an integer constant. Actually, the most common non-monotone ratio is defined as follows:

$$r_k = \frac{f_{l(k)} - f(x_k + d_k)}{m_k(0) - m_k(d_k)}. \tag{3}$$

Some researchers showed that utilizing non-monotone techniques may improve both the possibility of finding the global optimum and the rate of convergence [4,13]. However, although the non-monotone technique has many advantages, it contains some drawbacks [11, 13]. To overcome those disadvantages, Zhang et al. [13] proposed a new non-monotone technique to replace (3). To improve algorithm, Gu et al. [14] introduced another non-monotone form. They define

$$D_k = \begin{cases} f(x_k) & k = 1; \\ h_k D_{k-1} + (1 - h_k) f(x_k) & k \ge 2 \end{cases} \tag{4}$$

for some fixed $h \in (0,1)$, or a variable h_k. At the same time, they have the new non-monotone ratio:

$$r_k = \frac{D_k - f(x_k + d_k)}{m_k(0) - m_k(d_k)}. \tag{5}$$

In this paper, we determine the step-length a_k by sub-sequent Wolfe line search

$$f(x_k + a_k d_k) \le D_k + b a_k g_k^T d_k, \tag{6}$$

$$g(x_k + a_k d_k)^T d_k \ge g g_k^T d_k. \tag{7}$$

The rest of this paper is organized as follows. In Section 3, we introduce the algorithm of non-monotone trust region method with line search strategy. In Section 4, we analyze the new method and prove the global convergence. Some conclusions are given in Section 5.

III. NEW ALGORITHM

In this paper, we consider the following assumptions that will be used to analyze the convergence properties of the below new algorithm:

(H1) The level set $L_0 = \{x \in R^n \mid f(x) \le f(x_0)\} \subset \Omega$, where $\Omega \in R^n$ is a closed, bounded set.

(H2) The matrix B_k is a uniformly bounded matrix, i.e. there exists a constant $b > 0$ such that $\|B_k\| \le b$ for all $k \in K$.

(H3) $\nabla f(x)$ is a Lipschitz continuous function, i.e. there exists a constant $L > 0$ such that $\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\|,\ \forall x, y \in R^n$.

(H4) $\{\Delta_k\}$ has the upper bound Δ.

(H5) The constant δ in the following algorithm should satisfy $\delta \in (0, \min\{1, \nu/L\})$.

(H6) There exists a constant $\nu > 0$ such that $d^T B_k d \ge \nu\|d\|^2$ for all $d \in R^n$.

The new algorithm can be described as follows:

Algorithm 0

Step 1 An initial point $x_0 \in R^n$ and a symmetric matrix $B_0 \in R^{n\times n}$ are given. The constants $0 < \mu < 1$, $0 < \beta < \gamma < 1$, $0 < \delta < 1$, $0 < c_1 < 1 < c_2$, $0 < \eta < 1$, $M \ge 0$, $\tau > 0$ and $\varepsilon > 0$ are also given. Compute $f(x_0)$ and set $k = 0$.

Step 2 Compute g_k. If $\|g_k\| \le \varepsilon$ then stop, else go to Step 3.

Step 3 Solve (2) inaccurately to determine d_k, satisfying

$$m_k(0) - m_k(d_k) \ge \tau\|g_k\|\min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}, \tag{8}$$

$$g_k^T d_k \le -\tau\|g_k\|\min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}. \tag{9}$$

Step 4 Compute D_k and ρ_k. If $\rho_k \ge \mu$, go to Step 5. Otherwise, find the step-length α_k satisfying (6) and (7), then set $x_{k+1} = x_k + \alpha_k d_k$ and update $\Delta_{k+1} \in [\|x_{k+1} - x_k\|, c_1\Delta_k]$, go to step 6.

Step 5 Set $x_{k+1} = x_k + d_k$, and

$$\Delta_{k+1}\begin{cases} = \Delta_k, & \text{if } \|d_k\| < \Delta_k \\ \in [\Delta_k, c_2\Delta_k], & \text{if } \|d_k\| = \Delta_k \end{cases}.$$

Step 6 Update the symmetric matrix B_k, set $k = k+1$, go to step 2.

IV. CONVERGENCE ANALYSIS

For the convenience of expression, we Let $I = \{k \in K | \rho_k \geq \mu\}$ and $J = \{k \in K | \rho_k < \mu\}$. Obviously, $K = I \cup J$ is an infinite subset of the set $\{0,1,2,\ldots\}$.

We need the following lemmas in order to prove the convergence of the new algorithm.

Lemma 1 Suppose that H(1)-(H4) hold and the sequence $\{x_k\}$ be generated by Algorithm 0, and there is a $\varepsilon > 0$ such that $\|g_k\| \geq \varepsilon$. Then for all $k \in J$, there exists a constant $\bar{\alpha} > 0$ such that $\alpha_k > \bar{\alpha}$.

Proof. From (7) and (H3), we have

$$L\alpha_k \|d_k\|^2 \geq (g(x_k + \alpha_k d_k) - g_k)^T d_k \geq (\gamma - 1) g_k^T d_k > 0. \tag{10}$$

Thus, we can conclude that

$$\alpha_k \geq \frac{(\gamma - 1) g_k^T d_k}{L\|d_k\|^2} = \frac{(1-\gamma)}{L} \frac{|g_k^T d_k|}{\|d_k\|^2}. \tag{11}$$

This inequality, together with (H2) , (H4) and (9), lead us to have

$$\alpha_k \geq \frac{(1-\gamma)\tau \|g_k\|}{L} \frac{\min\{\Delta_k, \|g_k\|/\|B_k\|\}}{\Delta_k^{\,2}} \geq \frac{(1-\gamma)\tau\varepsilon}{L} \min\left\{\frac{1}{\Delta}, \frac{\varepsilon}{\Delta^2\beta}\right\}. \tag{12}$$

Let $\bar{\alpha} = \frac{(1-\gamma)\tau\varepsilon}{L} \min\left\{\frac{1}{\Delta}, \frac{\varepsilon}{\Delta^2\beta}\right\}$, we complete the proof.

Lemma 2 Suppose that (H3), (H5) and (H6) hold, and the sequence $\{x_k\}$ be generated by Algorithm 0. Then for all $k \in J$, we have

$$f_{k+1} - f_k \leq \frac{\delta}{2}\left(1 - \frac{L\delta}{v}\right) g_k^T d_k \leq 0. \tag{13}$$

Proof. The proof is similar to Lemma 3.1 in [15]

Lemma 3 (See Lemma 2 in [16]) Suppose that the sequence $\{x_k\}$ be generated by Algorithm 0. Then we have

$$f_{k+1} \leq D_{k+1} \leq D_k, \quad \forall k \in \infty.$$

Lemma 4 Suppose that (H1) holds and the sequence $\{x_k\}$ is generated by Algorithm 0. Then, the sequence

$\{D_k\}$ is convergent.

Proof. Lemma 3 together with (H1) imply that

$$\exists \lambda \quad s.t.\ \forall n \in \mathbb{N} : \lambda \le f_{k+1} \le D_{k+1} \le D_k \le \cdots \le D_1 \le f_1.$$

This shows that the sequence $\{D_k\}$ is convergent.

Lemma 5 Suppose that (H3), (H5), (H6) hold and the Algorithm 0 generates an infinite sequence $\{x_k\}$. Then for all $k \in \mathbb{N}$, there exists a constant $\varphi > 0$ such that

$$D_{k+1} \le D_k - (1-\eta)\varphi \, \| g_k \| \min\left\{ \Delta_k, \frac{\|g_k\|}{\|B_k\|} \right\},$$

where $\varphi = \min\left\{ \mu\tau, \frac{\delta\tau}{2}\left(1 - \frac{L\delta}{v}\right) \right\}.$

Proof. We still consider two cases:
Case1. $k \in I$. From (7) and (8), we have

$$D_k - f_{k+1} \ge \mu[m_k(0) - m_k(d_k)] \ge \mu\tau \, \| g_k \| \min\left\{ \Delta_k, \frac{\|g_k\|}{\|B_k\|} \right\} \ge 0.$$

Then, we can obtain that

$$f_{k+1} \le D_k - \mu\tau \, \| g_k \| \min\left\{ \Delta_k, \frac{\|g_k\|}{\|B_k\|} \right\}.$$

Case2. $k \in J$. From Lemma2, Lemma 3 and (9), we have

$$\begin{aligned} f_{k+1} &\le f_k + \frac{\delta}{2}\left(1 - \frac{L\delta}{v}\right) g_k^T d_k \\ &\le D_k - \frac{\delta\tau}{2}\left(1 - \frac{L\delta}{v}\right) \| g_k \| \min\left\{ \Delta_k, \frac{\|g_k\|}{\|B_k\|} \right\}. \end{aligned}$$

Let $\varphi = \min\left\{ \mu\tau, \frac{\delta\tau}{2}\left(1 - \frac{L\delta}{v}\right) \right\}$, we can conclude

$$f_{k+1} \le D_k - \varphi \, \| g_k \| \min\left\{ \Delta_k, \frac{\|g_k\|}{\|B_k\|} \right\}. \tag{14}$$

Considering (4) and (14), we obtain for all k

$$
\begin{aligned}
D_{k+1} &= \eta D_k + (1-\eta) f_{k+1} \\
&\le \eta D_k + (1-\eta)\left(D_k - \varphi \,\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}\right) \\
&= D_k - (1-\eta)\varphi \,\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}.
\end{aligned}
$$

Lemma 6 Suppose that (H1)-(H6) hold, if there exists a constant $\varepsilon > 0$ such that $\|g_k\| \ge \varepsilon$, then for all $k \in \infty$, we have

$$\lim_{k\to\infty} \min\{\Delta_k, \varepsilon/M_k\} = 0, \tag{15}$$

where $M_k = 1 + \max_{1\le i\le k} \|B_k\|$.

Proof. From Lemma 5 and the definition of M_k, we have

$$D_{k+1} - D_k \le -(1-\eta)\varphi\, \varepsilon \min\{\Delta_k, \varepsilon/M_k\}. \tag{16}$$

Using the above inequality and Lemma 4, we have (15) holds immediately.

Lemma 7 Suppose Lemma 1 and Lemma 3 hold, then for all sufficiently large k, there exists a constant $c_1 \in (0,1)$ such that

$$\Delta_k \ge c_1 \min\{1, \tau\varepsilon(1-\mu)\}/M_k .$$

Proof. The proof is similar to Lemma 3.8 in [15], we omit it for convenience.

Theorem 8 Suppose that (H1)-(H6) hold and $\{B_k\}$ satisfies

$$\sum_{k=0}^{+\infty} \frac{1}{M_k} = +\infty . \tag{17}$$

Then sequence $\{x_k\}$ generated by Algorithm 0 satisfies

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

Proof. Assume that (17) does not hold, then for all $k \in \infty$, there exists a constant $\varepsilon > 0$ such that $\|g_k\| \ge \varepsilon$. From Lemma 7, we have

$$\min\{\Delta_k, \varepsilon/M_k\} \ge \gamma/M_k , \tag{18}$$

where $\gamma = \min\{c_1, c_1\tau\varepsilon(1-\mu), \varepsilon\} = \min\{c_1, c_1\tau\varepsilon(1-\mu)\}$.

From (16) and (18), we have

$$\sum_{k=1}^{\infty} (D_k - D_{k+1}) \ge \sum_{k=1}^{\infty} (1-\eta)\varphi\, \varepsilon \min\{\Delta_k, \varepsilon/M_k\} \ge \sum_{k=1}^{\infty} (1-\eta)\varphi\, \varepsilon\gamma/M_k .$$

Using the above inequality and Lemma 4, we have

$\sum_{k=0}^{+\infty}\frac{1}{M_k}\le\frac{1}{(1-\eta)\varphi\varepsilon\gamma}\sum_{k=1}^{\infty}(\Delta_k-\Delta_{k+1})<\infty$. This contradicts (17). The proof is completed.

V. CONCLUSIONS

In this paper, a variant non-monotone trust region algorithm for solving unconstrained optimization problems is proposed. Unlike traditional trust region method, the proposed algorithm does not reject a trial step, but performs a new non-monotone Wolfe line search in direction of the rejected trial step in order to avoid resolving the trust region sub-problem instead. We analyzed the properties of the algorithm and proved the global convergence theory under some mild conditions.

ACKNOWLEDGMENTS

This work is supported by the national natural science foundation of China (61473111) and the Natural Science Foundation of Hebei Province (Grant No. A2014201003, A2014201100).

REFERENCES

[1] J. Nocedal, Y.X. Yuan, Combining trust region and line search techniques, in: Y. Yuan (Ed.), Advanced in nonlinear programming, Kluwer Academic Publishers, Dordrecht, 1998, pp. 153-175.

[2] E. Michael Gertz, A quasi-Newton trust region method, Math programming, 2004, 100(3): 447-470.

[3] R.M. Chamberlain, M.J.D. Powell, The watchdog technique for forcing convergence in algorithm for constrained optimization, Mathematical Programming Study 16 (1982) 1-17.

[4] L. Grippo, F. Lampariello, S. Lucidi, A nonmonotone line search technique for Newton's method, Society for Industrial and Applied Mathematics 23 (1986) 707-716.

[5] N.Y. Deng, Y. Xiao and F.J. Zhou, Nonmonotone trust region algorithm, Journal of Optimization Theory and Application 76 (1993) 259-285.

[6] L. Grippo, F. Lampariello, S. Lucidi, A truncated Newton method with nonmonotone line search technique for unconstrained optimization, J. Optim, Theory Appl.60(1989)401-419.

[7] Ph.L. Toint, An assessment of nonmonotone linesearch technique for unconstrained optimization, Society for Industrial and Applied Mathematics, 17 (1996) 725-739.

[8] Ph. L. Toint, Non-monotone trust-region algorithm for nonlinear optimization subject to convex constraints, Mathmatical Programming 77 (1997) 69-94.

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

[10] J.T. Mo, K.C. Zhang, Z.X. Wei, A nonmonotone trust region method for unconstrained optimization, Applied Mathematics and Computation 171(2005) 371-384.

[11] M. Ahookhosh, K. Amini, An efficient nonmonotone trust-region method for unconstrained optimization, Numer Algor(2012) 59: 523-540.

[12] S. Huang, Z. Wan, X.H. Chen, A new nonmonotone line search technique for unconstrained optimization, Numer Algor (2015) 68: 671–689.

[13] H.C. Zhang, W.W. Hager, A nonmonotone line search technique and its application to unconstrained optimization, SIAM J. Optim. 14 (4) (2004) 1043-1056.

[14] N.Z. Gu, J.T. Mo, Incorporating nonmonotone strategies into the trust region for unconstrained optimization, Computers and Mathematics with Applications 55 (2008) 2158–2172.

[15] J.T. Mo, C. L, S.C. Yan, A nonmonotone trust region method with fixed stepsize, Journal of Qufu Normal University Vol. 32, No. 3 (2006) 30-34.

[16] C.Y. Li, Q.H. Zhou, A new non-monotone trust region method with fixed step-size for unconstrained optimization, Scholars Journal of Engineering and Technology(2015) 3(5A): 529-534.