Scholars Journal of Engineering and Technology (SJET) **ISSN 2321-435X (Online)**
Sch. J. Eng. Tech., 2015; 3(6):640-644 **ISSN 2347-9523 (Print)**

(An International Publisher for Academic and Scientific Resources)
www.saspublisher.com

Research Article

A new non-monotone self-adaptive trust region method for unconstrained optimization

Baowei Liu
Department of Mathematics, Cangzhou Normal University, Cangzhou, Hebei Province, China

***Corresponding author**
czlbw@sina.cn
Email: qinghua.zhou@gmail.com

Abstract: In this paper, we consider a novel non-monotone self-adaptive trust region method for solving unconstrained optimization problem. Unlike the usual trust region methods, our proposed the new algorithm does not only uses current iterative but also the previous information to update the trust region radius at each iteration. The global convergence of the algorithm is established under some reasonable assumptions.
Keywords: non-monotone strategy; self-adaptive trust region method; unconstrained optimization; global convergence.

INTRODUCTION

In this paper, we consider the following unconstrained optimization problem:

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

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

Already, there are many methods to solve problem (1.1) such as line search methods and trust region methods. In the trust region method, at each iterative point x_k, it needs to compute a trial step d_k by solving the following quadratic sub-problem:

$$\begin{aligned} \min \quad & q_k(d) = f_k + g_k^T d + \tfrac{1}{2} d^T B_k d, \\ s.t. \quad & \|d\| \le \Delta_k, \end{aligned} \qquad (1.2)$$

where $f_k = f(x_k)$, $g_k = \nabla f(x_k)$, $B_k \in R^{n\times n}$ is a symmetric matrix which is the Hessian matrix or its approximation of $f(x)$ at the current point x_k, $\Delta_k > 0$ is the trust radius and $\|\cdot\|$ denotes to the Euclidean norm. By computing the ratio

$$r_k = \frac{f(x_k) - f(x_{k+1})}{q_k(0) - q_k(d_k)},$$

we can decide whether d_k is acceptable or not and how to adjust the trust region radius. The iteration is said to be successful if $r_k \ge \eta_1$. Then we obtain the new point x_{k+1}, and the trust region radius is updated. If not, the iteration is unsuccessful, and the trail point is rejected.

However, there exists a difficulty that how to adjust the trust region radius. In order to choose an self-adaptive trust region radius, many adaptive trust region methods have been studied in [1, 2].

In 1997, Sartenaer [3] introduced a strategy that can automatically determine the initial trust region radius. The strategy requires additional evaluations of the objective function. Zhang et al. [4] presented another efficient strategy of updating the trust region radius. That is, $\Delta_k = c^p(\|g_k\|/\gamma)$, $0 < c < 1$, $\gamma = \min(\|B_k\|, 1)$ and p is a positive integer. But, there still exist some drawbacks in the adaptive trust region method. Recently, a new updated rule is introduced by Cui et al. in [5]. They presented a new self-adaptive trust region method. The main difference between other methods and the new method is that in the new method of the trust region radius is defined by not only the current iterative information of

g_k and B_k but also the previous iterative information of d_{k-1}. In the new method, it does not need to compute $\|B_k\|$ or $\|B_k^{-1}\|$, which can decrease the cost of computation.

In the 1980s, Grippo et al. [6] firstly gave a non-monotone line search for Newton's method. This algorithm accepts the step-size α_k whether

$$f(x_k+\alpha_k d_k) \le f(x_{l(k)}) + \beta\alpha_k \nabla f(x_k)^T d, \tag{1.3}$$

where $\beta \in (0, \frac{1}{2})$, $f(x_{l(k)}) = \max_{0\le j\le m_k} f(x_{k-j}), m_0 = 0, 0 \le m_k \le \min\{m_{k-1}+1, M\} (k \ge 1),$ and $M \ge 0$ is an integer. Since then, many researchers [4, 7] have exploited the non-monotone technique and a lot of numerical tests have showed that the non-monotone technique proposed by Grippo et al. [6] is efficient at some extent. In 1993, Deng et al. [7] made some changes and applied it to the trust region method, and developed a non-monotone trust region method for unconstrained optimization. Zhang et al. [8] proposed another non-monotone line search method. In detail, their method finds a step-size α_k satisfying the following condition:

$$f(x_k+\alpha_k d_k) \le C_k + \beta\alpha_k \nabla f(x_k)^T d, \tag{1.4}$$

where

$$C_k = \begin{cases} f(x_k), & k=0, \\ \frac{\eta_{k-1}Q_{k-1}C_{k-1}+f(x_k)}{Q_k}, & k \ge 1, \end{cases} \quad Q_k = \begin{cases} 1, & k=0, \\ \eta_{k-1}Q_{k-1}+1, & k \ge 1, \end{cases} \tag{1.5}$$

and $\eta_{k-1} \in [\eta_{\min}, \eta_{\max}], \eta_{\min} \in [0,1)$ and $\eta_{\max} \in [\eta_{\min}, 1)$ are two chosen parameters. Numerical results showed that this non-monotone technique was superior to (1.3). In 2008, Gu and Mo [9] introduced a new simple non-monotone strategy as follow:

$$D_k = \begin{cases} f_k, & k=0, \\ \eta_k D_{k-1} + (1-\eta_k) f_k, & k \ge 1 \end{cases} \tag{1.6}$$

for $\eta_k \in [\eta_{\min}, \eta_{\max}]$ .This non-monotone technique is efficient and robust which is showed by numerical experiments in [9].

Inspired by the ideas introduced above, we use the new technique to update the radius, then applied it to the trust region method with non-monotone strategy proposed by Gu and Mo [9]. The purpose of this paper is to present a new non-monotone adaptive trust region method.

The rest of the paper is organized as follows. In Section 2, we describe our new non-monotone self-adaptive trust region algorithm. In Section 3, we prove the global convergence properties of this novel algorithm. Finally, some conclusions are summarized in Section 4.

Algorithm

In this section, we will describe our new non-monotone self-adaptive trust region method. After we obtain d_k, the ratio r_k is defined by

$$r_k = \frac{Ared_k}{\Pr ed_k} = \frac{D_k - f(x_k+d_k)}{q_k(0)-q_k(d_k)}, \tag{2.1}$$

Algorithm 2.1

Step 1. Given $x_0 \in R^n$, $\Delta_0 > \|g_0\|$, $B_0 \in R^{n\times n}$, $0 \le c_0 < c_1 < 1, 0 < c_2 < 1$ $\varepsilon \ge 0$, set $k := 0$.

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

Step 3. Solve the sub-problem (1.2) for d_k. Compute D_k, $Ared_k$, $Pred_k$ and r_k.

Step 4. If $r_k < c_0$, set $\Delta_{k+1} = c_2\Delta_k$, go to the Step 3; otherwise, go to Step 5.

Step 5. Set $x_{k+1} = x_k + d_k$. Compute g_{k+1} and B_{k+1}, and let $c_3 = {\|d_k\|^2}/{d_k^T B_{k+1} d_k}$. (2.2)

Step 6. Update the trust region radius Δ_{k+1} as

$$\Delta_{k+1} = \begin{cases} \max\{c_3 \|g_{k+1}\|, 4\|d_k\|\} & if\ r_k \ge c_1, \\ c_3 \|g_{k+1}\| & otherwise. \end{cases} \quad (2.3)$$

And set $k := k+1$, go to Step 2.

Remark: Step 3- Step 4 is called the internal circulation and the cycling index is denoted by p_k at the current iterative point x_k.

Global convergence

In this section, we will prove the global convergence properties of Algorithm 2.1. The following assumptions are necessary to analyze the convergence properties.

(H1) The level set $L(x_0) = \{x \in R^n \,|\, f(x) \le f(x_0)\}$ is bounded for any given $x_0 \in R^n$.

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

Lemma 3.1. (See Lemma 13.3.1 in [10]) If d_k is the solution to sub-problem (1.2), then

$$Pred_k = q_k(0) - q_k(d_k) \ge \tfrac{1}{2}\|g_k\| \min\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\}. \quad (3.1)$$

Lemma 3.2. $f_k - f(x_k + d_k) - Pred_k = O(\|d_k\|^2)$

Proof. From Taylor expansion and the definition of $Pred_k$, this lemma is obviously true..

Lemma 3.3. Let $\{x_k\}$ be the sequence generated by Algorithm 2.1. For any fixed $k \ge 0$, we have

$$f_{k+1} \le D_{k+1}. \quad (3.2)$$

Proof. We obtain $D_{k+1} - f_{k+1} = \eta_{k+1}(D_k - f_{k+1})$ for all $k \ge 0$ from the definition of D_k. By $r_k \ge c_0$, (2.1) and Lemma 3.1, we have

$$D_k - f_{k+1} \ge c_0 \operatorname{Pr}ed_k \ge \tfrac{c_0}{2}\|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\} \ge 0 \quad (3.3)$$

$$\begin{aligned} D_{k+1} - f_{k+1} &= \eta_{k+1}(D_k - f_{k+1}) \ge c_0 \eta_{k+1} \operatorname{Pr}ed_k \\ &\ge \tfrac{c_0}{2}\eta_{k+1}\|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\} \ge 0 \end{aligned} \quad (3.4)$$

Therefore, we have $D_{k+1} \ge f_{k+1}$.

Lemma 3.4. (See Lemma 3.3 in [5]) Suppose that Assumption (H1) holds, then

$$Pred_k \ge \tfrac{1}{2}\tfrac{c_2^{p_k}}{M_k}\|g_k\|^2 \quad (3.5)$$

hold for all k, where $M_k = \|B_k\|$ and p_k is the cycling index at the current iterate and the last iterate.

Lemma 3.5. (See Lemma 4.10 in [11]) Suppose that the sequence $\{x_k\}$ generated by Algorithm 2.1. The algorithm is well defined, i.e., it could not cycle infinitely in the inner cycle.

Lemma 3.6. Suppose that the sequence $\{x_k\}$ generated by Algorithm 2.1. Then the sequence $\{D_k\}$ is decreasing.

Proof. From formula (3.3), we know that

$$f_{k+1} \le D_k - c_0 \operatorname{Pr}ed_k \le D_k - \tfrac{1}{2}c_0\|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\}. \quad (3.6)$$

$$\begin{aligned} D_{k+1} &= \eta_{k+1} D_k + (1-\eta_{k+1}) f_{k+1} \\ &\le \eta_{k+1} D_k + (1-\eta_{k+1})(D_k - \tfrac{1}{2}c_0\|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\}.) \\ &= D_k - (1-\eta_{k+1})\tfrac{1}{2}c_0\|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\}. \end{aligned} \quad (3.7)$$

The formula above indicates that the sequence $\{D_k\}$ is monotonically decreasing.

Theorem 3.7. Suppose that Assumption (H1) holds. Let the sequence $\{x_k\}$ generated by Algorithm 2.1, then we have

$$\liminf_{k\to\infty} \|g_k\| = 0. \quad (3.8)$$

Proof. We assume that (3.8) is not true, that is, there exists a positive constant $\tau > 0$, such that

$$\|g_k\| \geq \tau \quad \text{for all } k. \qquad (3.9)$$

From (3.9), Assumption (H2) and Lemma 3.4, we obtain that

$$Pred_k \geq \frac{1}{2}\frac{c_2^{p_k}}{M_k}\|g_k\|^2 \geq \frac{1}{2}\frac{c_2^{p_k}}{M}\tau^2 \qquad (3.10)$$

where p_k is the largest cycling index at the iterate x_k.

According to Lemma 3.5 and the definition of r_k, we know that

$$D_k - f_{k+1} \geq \frac{1}{2}\frac{c_0 c_2^{p_k}\tau^2}{M} \qquad (3.11)$$

By (3.11), Assumption (H1) and the convergence of the sequence $\{D_k\}$, we can obtain

$$\lim_{x\to\infty} p_k = \infty$$

By the definition of Algorithm 2.1, we know that the solution $\widehat{d}_k$ of the following sub-problem

$$\begin{aligned} \min \quad & q_k(d) = g_k^T d + \tfrac{1}{2} d^T B_k d, \\ s.t. \quad & \|d\| \leq \tfrac{\Delta_k}{c_2}, \end{aligned} \qquad (3.12)$$

is not accepted, i.e.,

$$\frac{D_k - f(x_k + \widehat{d}_k)}{q_k(0) - q_k(\widehat{d}_k)} < c_0 \qquad (3.13)$$

On the other hand, from Lemma 3.1, we have

$$D_k - f_{k+1} \geq c_0 \,\mathrm{Pr}\, ed_k \geq \tfrac{1}{2} c_0 \|g_k\| \min\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\} \geq \tfrac{1}{2} c_0 \tau \min\{\Delta_k, \tfrac{\tau}{M}\} \qquad (3.14)$$

By (3.14) and Assumption (H1), we can obtain

$$\lim_{x\to\infty} \Delta_k = 0. \qquad (3.15)$$

Thus, we have

$$\left| \frac{f_k - f(x_k + \widehat{d}_k)}{q_k(0) - q_k(\widehat{d}_k)} - 1 \right| \leq \frac{O(\|\widehat{d}_k\|^2)}{\frac{1}{2}\|g_k\| \min\{\frac{\Delta_k}{c_2}, \frac{\|g_k\|}{\|B_k\|}\}} \leq \frac{c_2 O(\Delta_k^2)}{\frac{1}{2}\tau\Delta_k} \to 0 \qquad (3.16)$$

From (3.15) and (3.16),

$$r_k = \frac{D_k - f(x_k + \widehat{d}_k)}{q_k(0) - q_k(\widehat{d}_k)} \geq \frac{f_k - f(x_k + \widehat{d}_k)}{q_k(0) - q_k(\widehat{d}_k)} \geq c_0, \, k \to \infty \qquad (3.17)$$

This is a contradiction with Formula (3.13). Theorem 3.7 has been proved.

CONCLUSIONS

In this paper, we give a new non-monotone self-adaptive trust region method for unconstrained optimization. In the algorithm, the trust region radius relies on the previous and current iterative information. Under some mild conditions, we establish the global convergence result of the proposed method.

Acknowledgments

The author would be very grateful to the referees for their valuable comments and suggestions.

References

1. Sang Z, Sun Q; A self-adaptive trust region method with line search based on a simple sub-problem model, Journal of Applied Mathematic and Computing, 2009; 232(2): 514-522.
2. Fan JY, Ai WB, Zhang QY, A line search and trust region algorithm with trust region radius converging to zero, Journal of Computational Mathematics, 2004; 22(6): 865–872.
3. Sartenaer; Automatic determination of an initial trust region in nonlinear programming, SIAM Journal on Scientific Computing, 1997; 18(6): 1788–1803.
4. Sun W, Zhou Q, An unconstrained optimization method using nonmonotone second order Goldstein's line search, Sci. China Ser. A 50, 2007: 1389-1400.

5. Cui Z, Wu B, A new self-adaptive trust region method for unconstrained optimization, Journal of Vibration and Control, 2011; 18(9): 1303–1309.
6. Grippo L, Lamparillo F, Lucidi S; A nonmonotone line search technique for Newton's method, SIAM Journal on Numerical Analysis, 1986; 23(4): 707-716.
7. Deng N, Xiao Y, Zhou F; Nonmontonic trust region algorithm, Journal of Optimization Theory and Application, 1993; 76(2): 259-285.
8. Zhang H, Hager W; A nonmonotone line search technique and its application to unconstrained optimization, SIAM Journal on Optimization, 2004; 14(4): 1043-1056.
9. Gu N, Mo J; Incorporating nonmonotone strategies into the trust region method for unconstrained optimization problem, Computer & Mathematics with Applications, 2008; 55(9): 2158-2172.
10. Yuan Y, Sun W, Optimization Theory and Methods, Science Press of China, 1997.
11. Zhou Q, Hang D, Nonmonotone adaptive trust region method with line search based on new diagonal updating, Applied Numerical Mathematics, 2015; 91: 75-88.