Scholars Bulletin
(A Multidisciplinary Bi-weekly Journal)
An Official Publication of "Scholars Middle East Publishers",
Dubai, United Arab Emirates
Website: http://scholarsbulletin.com/

ISSN 2412-9771 (Print)
ISSN 2412-897X (Online)

A new non-monotone trust region method based on simple quadratic models

Liran Yang, Qinghua Zhou, Weili Zheng
College of Mathematics and Information Science, Hebei University, Baoding- 071002, Hebei Province, China

***Corresponding Author:**
Qinghua Zhou
Email: qinghua.zhou@gmail.com

Abstract: In this paper, we propose a non-monotone trust region method for solving unconstrained optimization problem. Unlike the traditional trust region methods, our new algorithm is simple by combining non-monotone strategy with a scale approximation of the objective function's Hessian. Theoretical analysis indicates that the new method preserves the global convergence under some mild conditions.
Keywords: non-monotone trust region method; unconstrained optimization; global convergence

INTRODUCTION

Trust region method is a familiar iterative method to solve the unconstrained optimization problem:

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

Where $f(x): R^n \to R$ is a twice continuously differentiable function. For a given iteration point x_k, it computes a trial step d_k by solving the following quadratic sub-problem:

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

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 called the trust radius and $\|\cdot\|$ refers to the Euclidean norm. The ratio ρ_k between the actual reduction $f(x_k) - f(x_{k+1})$ and the predicted reduction of the model $q_k(0) - q_k(d_k)$ plays an important role to decide whether d_k is accepted or not and how to adjust the trust region radius.

It is well-known that traditional trust region methods consist of some drawbacks. Many numerical experiments [1, 2] indicate that enforcing monotonicity of $\{f(x_k)\}$ may slow the rate of convergence when the iteration is trapped near a narrow valley.

One of the most efficient methods for overcoming mentioned above drawbacks is non-monotone techniques. In 1986, Grippo et al. [3] proposed 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 $\beta \in (0, \tfrac{1}{2})$, $f(x_{l(k)}) = \max_{0 \le j \le m_k} f(x_{k-j}), m_0 = 0, 0 \le m_k \le \min\{m_{k-1} + 1, M\} (k \ge 1)$, and $M \ge 0$ is an integer. It has been proved that the sequence $\{f(x_k)\}$ is not increasing. Since then, many researchers [4-6] have exploited the non-monotone technique and a lot of numerical tests have showed that the non-monotone technique proposed by Grippo et al. [3] is efficient at some extent. In 1993, Deng et al. in [1] made some changes and applied it to the trust region method, and proposed a non-monotone trust region method for unconstrained optimization. Theoretical analysis and numerical results show that algorithms with non-monotone strategy are more effective than algorithms without it. From then on a variety of the non-monotone trust region methods have been presented [2, 7].

Although the non-monotone technique has many advantages, however, it has some disadvantages too. The iterations may not satisfy the condition (1.3) for sufficiently large k, for any fixed bound M on the memory. Zhang and Hager [8] also pointed out that the numerical results are dependent on the choice of parameter M in some cases. In order to overcome these disadvantages, Zhang and Hager [8] proposed another non-monotone line search method, they replaced the maximum function value with an average of function values. 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

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

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). Then, this non-monotone was applied to the trust region methods [9, 10]. Recently, Gu and Mo [11] introduced another non-monotone strategy. They replaced C_k in (1.4) with R_k

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

for $\eta_k\in[\eta_{\min},\eta_{\max}]$ .This non-monotone technique is efficient and robust which is showed by numerical experiments in [11].

The key problem is how to solve the trust region sub-problem (1.2) for the trust region method. Many efficient methods for sub-problem (1.2) have been proposed [1, 2, 7]. However, when the scale of problem (1.1) is large, these methods may be too slow because all these methods have to store a symmetric matrix B_k and the algorithms are complicated relatively.

A diagonal-sparse quasi-Newton method, which replaces the scalar matrix with the diagonal matrix, was proposed in [12]. Based on the diagonal-sparse quasi-Newton method [12], Sun et al. [13] developed a non-monotone trust region algorithm with simple quadratic models, in which the approximation of Hessian matrix in the sub-problem is a diagonal positive definite matrix. Concretely, $B_k=diag(b_k^1,b_k^2,\ldots,b_k^n)$ so that $(b_k^1,b_k^2,\ldots,b_k^n)$ is the solution of

$$\min_{L\le b_k^i\le U}\sum_{i=1}^{n}(y_{k-1}^i-b_k^i d_{k-1}^i)^2 \tag{1.7}$$

where $d_{k-1}=x_k-x_{k-1}$, $y_{k-1}=g_k-g_{k-1}$, and $0<L<U$ are two constants. Then $b_k^i\ (i=1,2,\ldots,n)$ is given by

$$b_k^i=\begin{cases}\frac{y_{k-1}^i}{d_{k-1}^i}, & if\ L\le\frac{y_{k-1}^i}{d_{k-1}^i}\le U;\\ L, & if\ \frac{y_{k-1}^i}{d_{k-1}^i}<L;\\ U, & if\ \frac{y_{k-1}^i}{d_{k-1}^i}>U;\\ \frac{L+U}{2}, & if\ d_{k-1}^i=0.\end{cases}$$

Obviously, the memory requirements and computational complexity for estimating B_k are low.

Inspired by the ideas introduced above, we use the new scale approximation of the minimizing function’s Hessian in the trust region sub-problem, and then combine it with the non-monotone strategy proposed by Gu and Mo [11].

This paper is organized as follows. In Section 2, we describe our new non-monotone trust region method. The properties of this new algorithm and the global convergence property are given in Section 3. Finally, some concluding remarks are given in Section 4.

New non-monotone trust region method

If we give the initial point x_0, then f_0 and g_0 can be computed. Suppose I is the $n\times n$ identity matrix and set $B_0=I$. We can get the next iteration point $x_1=x_0+d_0$. Suppose that $x_k\ (k\ge 1)$ have been obtained. We compute the approximation of the Hessian of the function f at x_k. From $x_k = x_{k-1} + d_{k-1}$ we have $x_{k-1} = x_k - d_{k-1}$. By the Taylor's theorem, we can obtain

$$f(x_{k-1}) = f(x_k - d_{k-1}) \approx f(x_k) - g_k^T d_{k-1} + \tfrac{1}{2} d_{k-1}^T \nabla^2 f(x_k) d_{k-1}, \quad (2.1)$$

We consider $\gamma(x_k)I$ as an approximation of $\nabla^2 f(x_k)$, where $\gamma(x_k) \in R$. And the $\gamma(x_k)$ can be expressed as

$$\gamma(x_k) = \begin{cases} \frac{2}{d_{k-1}^T d_{k-1}}[f(x_{k-1}) - f(x_k) + g_k^T d_{k-1}], & if\ f(x_{k-1}) - f(x_k) + g_k^T d_{k-1} > 0, \\ \frac{2\delta}{d_{k-1}^T d_{k-1}}, & otherwise. \end{cases} \quad (2.2)$$

So, the sub-problem (1.2) can be modified as

$$\begin{aligned} \min \quad & q_k(d) = f_k + g_k^T d + \tfrac{1}{2}\gamma(x_k) d^T d, \\ s.t. \quad & \|d\| \le \Delta_k, \end{aligned} \quad (2.3)$$

The sub-problem (2.3) can be solved easily. In fact, if $\left\| -\frac{1}{\gamma(x_k)} g_k \right\| \le \Delta_k$, set $d_k = -\frac{1}{\gamma(x_k)} g_k$; otherwise d_k of sub-problem (2.3) is the solution of the following problem [15]:

$$\begin{aligned} \min \quad & q_k(d) = f_k + g_k^T d + \tfrac{1}{2}\gamma(x_k) d^T d, \\ s.t. \quad & \|d\| = \Delta_k. \end{aligned} \quad (2.4)$$

By solving (2.4), we can compute the solution $d_k = -\frac{\Delta_k}{\|g_k\|} g_k$.

After obtaining d_k, then the ratio ρ_k is computed by

$$\rho_k = \frac{Ared_k}{Pred_k} = \frac{R_k - f(x_k + d_k)}{q_k(0) - q_k(d_k)}, \quad (2.5)$$

Algorithm 2.1

Step 1. Given $x_0 \in R^n, \Delta_0 > 0, 0 < \mu < \nu_1 < \nu_2 < 1, 0 < c_1 < 1, c_2 > c_3 > 1, \delta > 0$ $0 < \varepsilon < 1, \theta > 0$. Set $k = 0, \gamma_0 = 1$. Choose parameters $\eta_{\min} \in [0,1)$ and $\eta_{\max} \in [\eta_{\min}, 1)$.

Step 2. Compute g_k. If $\|g_k\| = 0$, stop. Otherwise, go to Step 3.

Step 3. Solve the sub-problem (2.3) for d_k.

Step 4. Compute R_k, $Ared_k$, $Pred_k$ and ρ_k.

Step 5. If $\rho_k < \mu$, set $\Delta_k = c_1 \Delta_k$, go to the Step 3.

Step 6. Set $x_{k+1} = x_k + d_k$. Compute Δ_{k+1} as follows:

$$\Delta_{k+1} = \begin{cases} c_2 \Delta_k, & if\ \rho_k \ge \nu_2, and\ \|d_k\| = \Delta_k, \\ c_3 \Delta_k, & if\ \rho_k \ge \nu_1, \\ \Delta_k, & otherwise. \end{cases}$$

Step 7. Compute $\gamma(x_{k+1})$. If $\gamma(x_{k+1}) \le \varepsilon$ or $\gamma(x_{k+1}) \ge \frac{1}{\varepsilon}$, set $\gamma(x_{k+1}) = \theta$.

Step 8. Choose $\eta_k \in [\eta_{\min}, \eta_{\max}]$. Set $k = k + 1$, go to Step 2.

It is obvious that for all k, $0 < \min(\varepsilon, \theta) \le \gamma(x_k) \le \max(\frac{1}{\varepsilon}, \theta)$ (2.6)

CONVERGENCE ANALYSIS

In this section, we will prove the global convergence property of Algorithm 2.1. The following assumption is necessary to analyze the convergence property.

Assumption 1. The level set $L(x_0) = \{x \in R^n \mid f(x) \le f(x_0)\}$ is bounded for any given $x_0 \in R^n$.

Lemma 2. (See Lemma 3.2 in [14]) If d_k is the solution to sub-problem (2.3), then

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

Lemma 3. Let $\{x_k\}$ be the sequence generated by Algorithm 2.1. For any fixed $k \geq 0$, we have

$$f_{k+1} \leq R_{k+1}. \tag{3.2}$$

Proof. Let $k \geq 0$ be an arbitrary fixed integer. We obtain $R_{k+1} - f_{k+1} = \eta_{k+1}(R_k - f_{k+1})$ from the definition of R_k. By $\rho_k \geq \mu$, (2.5) and Lemma 2, we have

$$R_k - f_{k+1} \geq \mu \operatorname{Pr}ed_k \geq \frac{\mu}{2}\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\gamma(x_k)}\right\} \geq 0, \tag{3.3}$$

$$R_{k+1} - f_{k+1} = \eta_{k+1}(R_k - f_{k+1}) \geq \mu\eta_{k+1} \operatorname{Pr}ed_k \geq \frac{\mu}{2}\eta_{k+1}\|g_k\| \min\left\{\Delta_k, \frac{\|g_k\|}{\gamma(x_k)}\right\} \geq 0 \tag{3.4}$$

Therefore, we have $R_{k+1} \geq f_{k+1}$.

Lemma 4. (See Lemma 3.4 in [14]) Suppose that the sequence $\{x_k\}$ is generated by Algorithm 2.1. The algorithm is well defined, i.e., it could not cycle infinitely in the inner cycle.

Lemma 5. (See Lemma 3.6 in [14]) Suppose that Assumption 1 holds, and there is a positive number $\tau > 0$ such that $\|g_k\| \geq \tau$ for all k, then there exists a $\overline{\Delta} > 0$, such that for all k, we have $\Delta_k \geq \overline{\Delta}$.

Lemma 6. Suppose that Assumption 1 holds, the sequence $\{x_k\}$ generated by Algorithm 2.1 is contained in the level set $L(x_0)$.

Proof. The constatation follows from Lemma 3, Assumption 1 and $R_0 = f_0$.

Lemma 7. Suppose that the sequence $\{x_k\}$ generated by Algorithm 2.1. Then the sequence $\{R_k\}$ is decreasing.

Proof. Combining (2.5) and (3.1), and $\rho_k \geq \mu$, we have

$$f_{k+1} \leq R_k - \mu \operatorname{Pr}ed_k \leq R_k - \frac{1}{2}\mu\|g_k\| \min\{\Delta_k, \frac{g_k}{\gamma(x_k)}\}. \tag{3.5}$$

$$\begin{aligned} R_{k+1} &= \eta_{k+1}R_k + (1-\eta_{k+1})f_{k+1} \\ &\leq \eta_{k+1}R_k + (1-\eta_{k+1})(R_k - \tfrac{1}{2}\mu\|g_k\|\min\{\Delta_k, \tfrac{\|g_k\|}{\gamma(x_k)}\}) \\ &= R_k - (1-\eta_{k+1})\tfrac{1}{2}\mu\|g_k\|\min\{\Delta_k, \tfrac{\|g_k\|}{\gamma(x_k)}\}. \end{aligned} \tag{3.6}$$

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

Theorem 8. Suppose that Assumption 1 holds. Let the sequence $\{x_k\}$ generated by Algorithm 2.1, then we have

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{3.7}$$

Proof. We assume that Formula (3.7) is not true, that is, there exists a positive constant $\tau > 0$, such that

$$\|g_k\| \geq \tau \quad \text{for all } k. \tag{3.8}$$

From (3.6), (2.6) and Lemma 5 we obtain that

$$\begin{aligned} R_{k+1} &= \eta_{k+1}R_k + (1-\eta_{k+1})f_{k+1} \\ &\leq \eta_{k+1}R_k + (1-\eta_{k+1})(R_k - \tfrac{1}{2}\mu\|g_k\|\min\{\Delta_k, \tfrac{\|g_k\|}{\gamma(x_k)}\}) \\ &= R_k - (1-\eta_{k+1})[\tfrac{1}{2}\mu\|g_k\|\min\{\Delta_k, \tfrac{\|g_k\|}{\gamma(x_k)}\}], \\ &\leq R_k - \tfrac{1}{2}(1-\eta_{k+1})\mu\tau\min\{\overline{\Delta}, \tfrac{\tau}{\max(\frac{1}{\varepsilon},\theta)}\}, \end{aligned} \tag{3.9}$$

Where $\overline{\Delta} > 0$. So, by (3.9) we have

$$R_k - R_{k+1} \geq \frac{1}{2}(1-\eta_{k+1})\mu\tau\min\{\overline{\Delta}, \frac{\tau}{\max(\frac{1}{\varepsilon},\theta)}\}. \tag{3.10}$$

Consider Lemma 3 and Lemma 7, we know that $\{R_k\}$ is decreasing and $f_k \leq R_k$ for all $k \geq 0$. Meanwhile from Assumption 1 and Lemma 6, we know that $\{f_k\}$ is bounded below. Therefore $\{R_k\}$ is convergent. It follows from (3.10) that

$$\sum_{k=0}^{\infty}(1-\eta_{k+1})\min\{\overline{\Delta}, \frac{\tau}{\max(\frac{1}{\varepsilon},\theta)}\} < \infty \tag{3.11}$$

Set $\min\{\overline{\Delta}, \frac{\tau}{\max(\frac{1}{\varepsilon},\theta)}\} = \lambda$. Formula (3.11) can be written as

$$\sum_{k=0}^{\infty}(1-\eta_{k+1})\lambda < \infty \tag{3.12}$$

In the fact that $\eta_k \in [0,1)$, we have $\sum_{k=0}^{\infty}(1-\eta_{k+1})\lambda$ is not convergent. This is a contradiction with Formula (3.12). Theorem 8 has been proved.

CONCLUSIONS

In this paper, we present a new non-monotone trust region method based on simple quadratic models. The form of the new method is very simple. Under some mild conditions, we proved the global convergence result of the proposed method.

Acknowledgments

This work is supported by the Natural Science Foundation of Hebei Province (Grant No. A2014201003, A2014201100).

REFERENCES

1. Deng, N, Xiao, Y. & Zhou, F (1993). Nonmontonic trust region algorithm, J. Optim. Theory Appl. 26, 259-285.
2. Fu, J., & Sun, W. (2005). Nonmonotone adaptive trust-region method for unconstrained optimization problems. Applied Mathematics and Computation, 163(1), 489-504.
3. Grippo, L., Lampariello, F., & Lucidi, S. (1986). A nonmonotone line search technique for Newton's method. SIAM Journal on Numerical Analysis, 23(4), 707-716.
4. Raydan, M.(1997). The Barzilai and Barwein gradient method for large scale unconstrained minimization problem, SIAM J. Optim 7,26-33.
5. Sun, W. Y., & Zhou, Q. Y. (2007). An unconstrained optimization method using nonmonotone second order Goldstein’s line search. Science in China Series A: Mathematics, 50(10), 1389-1400.
6. Zhou, Q., & Sun, W. (2007). A nonmonotone second order line search method for unconstrained optimization, J. Comput. Math. 25, 104-112.
7. Zhou, Q., & Sun, W. (2006). An adaptive nonmonotone trust region method with curvilinear optimization, J. Comput. Math. 24, 761-770.
8. Zhang, H., & Hager, W. W. (2004). A nonmonotone line search technique and its application to unconstrained optimization. SIAM Journal on Optimization, 14(4), 1043-1056.
9. Mo, J., Liu, C., & Yan, S. (2007). A nonmonotone trust region method based on nonincreasing technique of weighted average of the successive function values. Journal of Computational and Applied Mathematics, 209(1), 97-108.
10. Qing-Jun, W. (2010). Nonmonotone trust region algorithm for unconstrained optimization problems. Applied Mathematics and Computation, 217(8), 4274-4281.
11. Gu, N. Z., & Mo, J. T. (2008). Incorporating nonmonotone strategies into the trust region method for unconstrained optimization. Computers & Mathematics with Applications, 55(9), 2158-2172.
12. Shi, Z. J., & Sun, G. (2006). A diagonal-sparse quasi-Newton method for unconstrained optimization problem. J. Sys. Sci. Math. Sci, 26(1), 101-112.
13. Q. Sun, L. Duan, B. Cui (2009). A nonmonotone trust region algorithm with simple quadratic models, J. Systems Sci. Math. Sci. 29 , 407-483.
14. Zhou, Q., Chen, J., & Xie, Z. (2014). A nonmonotone trust region method based on simple quadratic models. Journal of Computational and Applied Mathematics, 272, 107-115.
15. Sun, W., & Yuan, Y. X. (2006). Optimization theory and methods: nonlinear programming (Vol. 1). Springer Science & Business Media.