Scholars Bulletin
(A Multidisciplinary 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)

An Adjustable Algorithm Based on Non-Monotone Strategy for Optimization Problems

Shaojing Lian, Qiang Hua
Key Lab. of Machine Learning and Computational Intelligence, College of Mathematics and Information Science, Hebei University, Baoding, China

***Corresponding Author:**
Shaojing Lian
Email: 48458272@qq.com

Abstract: This paper devotes to incorporating a non-monotone strategy with an adjusted trust region radius to propose a more efficient trust region approach for unconstrained optimization. The primary objective of the paper is to introduce a more relaxed trust region approach based on a novel extension in trust region ratio and radius. The global convergence is proved under some reasonable conditionst.
Keywords: unconstrained optimization; trust region; non-monotone strategy; adjustable radius; global convergence.

INTRODUCTION

In this paper, we consider the following unconstrained optimization problem

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

where $f(x): R^n \to R$ is a real-valued twice continuously differentiable function. Two important methods have been developed for solving this problem, namely, line search and trust region methods [1, 2]. Trust region methods are a prominent class of methods for unconstrained optimization problems defining an area around the current step x_k in which the quadratic model has a good agreement with the objective function. In these methods, in each iterate, a trial step d_k is obtained by solving the following the quadratic sub-problem:

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

where $f_k = f(x_k)$, $g_k = \nabla f(x_k)$, and $B_k \in R^{n\times n}$ is a symmetric matrix which is the Hessian matrix or its approximation of $f(x)$ at the current point x_k, Δ_k is called the trust radius and $\|\cdot\|$ refers to the 2-norm. A crucial point in trust region methods is a strategy of choosing the trust region radius Δ_k, at every iterate. In the standard trust region method, based on agreement between the model and the objective function, the radius of trust region is updated by the following ratio

$$r_k = \frac{f(x_k) - f(x_k + d_k)}{q_k(0) - q_k(d_k)} \tag{3}$$

The numerator and the denominator of (3) have been called the actual reduction and the predicted reduction, respectively. It can be concluded that there will be a good agreement between the model and the objective function over current region of trust whenever r_k be close to 1. In this case, it is safe to increase the trust region radius in the next iterate. Otherwise, the trust region radius must be shrunk.

It is well known that the standard trust region method is very sensitive on the initial trust region radius [3-5]. In other word, we know that the standard trust region radius Δ_k is independent from g_k and B_k, so we do not know the radius Δ_k is suitable to the whole of implementation. This situation possibly increases the number of solving sub-problems in the inner steps of the method and so decreases the efficiency of the method. It is obvious that if we decrease the number of ineffective iterates, we can decline the number of solving sub-problems in each step. In [4], Sartenaer proposed an approach to determine the initial radius monitoring agreement between the model and the objective function along the steepest descent direction computed at the starting point. The first adjustable strategy to determine the trust region radius, for decreasing the number of solving sub-problems, was proposed by Zhang et al. in [6].

This strategy used the information of gradient and Hessian in current iterate to construct the trust region radius Δ_k without requiring any initial trust region radius. Inspired by Zhang's strategy, Shi and Guo in [5] proposed a automatically adjustable radius for trust region methods. They proved that the new method preserves the global, the super-linear and the quadratic convergence properties of the standard method. The numerical experiments have been showed that this method is more efficient than Zhang's method and standard trust region method. We describe these trust region radii in the next section.

On the other hand, Grippo et al. in [7, 8] provided a non-monotone strategy to line search methods for unconstrained optimization problems. In their non-monotone line search, step-length α_k is accepted if it satisfies the following Armijo-type condition

$$f(x_k+\alpha_k d_k) \le \max_{0\le j\le m_k} f(x_{k-j}) + \phi\alpha_k \nabla f(x_k)^T d_k \tag{4}$$

where $\phi \in (0,1)$, $m_0 = 0$ and $0 \le m_k \le \min\{m_{k-1}+1, M\}$ with an integer constant $M > 0$. Theoretical analysis and numerical experiments have been indicated the efficiency and robustness of this strategy to improve both the possibility of finding the global optimum and the rate of convergence of algorithm [8]. Motivated by these outstanding results, many researchers have interested to work on combination the non-monotone strategy with the trust region methods [9-12]. Gu and Mo in [13] proposed a new non-monotone strategy. The method substitutes $\max_{0\le j\le m_k} f(x_{k-j})$ with D_k which is defined as follows

$$D_k = \begin{cases} f_k\,, & k = 0 \\ \eta_k D_{k-1} + (1-\eta_k) f_k\,, & k \ge 1 \end{cases} \tag{5}$$

In their proposal, the ratio (3) changed as

$$r_k = \frac{D_k - f(x_k + d_k)}{m_k(0) - m_k(d_k)} \tag{6}$$

The investigation have been proved that the combination of the non-monotone strategy with trust region made a new method which has been inherited the strong theoretical properties of trust region as well as the computational robustness of the non-monotone strategy.

The rest of this paper organized as follows. In Section 2, we describe a novel hybrid of non-monotone trust region methods with an adjustable radius. Some properties and the global convergence of the new method are investigated in Section 3. Finally, some concluding remarks are delivered in Section 4.

NOVEL TRUST REGION ALGORITHM

In this section, we describe the trust region radius of Shi and Guo [5]. Then we introduce a new non-monotone trust region algorithm with automatically adjustable radius based on the adjustable radius of Shi and Guo and the idea of the non-monotone strategy of Gu and Mo. We also establish some properties of the new algorithm.

In 2008, Shi and Guo in [5] proposed a variant adjustable radius for trust region method. They selected parameters $\mu, \rho \in (0,1)$, $\tau \in (0,1]$ and q_k to be satisfy in

$$-\frac{g_k^T q_k}{\|g_k\|\|q_k\|} \ge \tau \tag{7}$$

The method provides a new trust region radius by

$$\Delta_k = \sigma_k \|q_k\| \tag{8}$$

where $\sigma_k = \rho^{p_k} s_k$, and p_k is the smallest positive integer number p such that $r_k \ge \mu$. They also determine the term s_k by

$$s_k = -\frac{g_k^T q_k}{q_k^T \hat{B}_k q_k} \quad (9)$$

in which $\hat{B}_k$ is generated by the procedure: $\hat{B}_k = B_k + iI$, where i is the smallest nonnegative integer such that the condition $q_k^T \hat{B}_k q_k > 0$ holds. It is obvious that if the matrix B_k be a positive definite matrix, then there is no need to substitute $\hat{B}_k$ by B_k. To avoid this substitution, we take advantage of a positive definite version of the BFGS quasi-Newton formula as follows

$$B_{k+1} = B_k + \frac{y_k y_k^T}{s_k^t y_k} - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} \quad (10)$$

where $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$. In order to preserve positive definite property, we do not update B_k whenever the curvature condition, i.e. $s_k^T y_k > 0$, does not hold, i.e. $\hat{B}_k = B_k$. We now can outline our non-monotone trust region algorithm with adjustable radius as follows:

Algorithm 1: non-monotone adjustable trust region algorithm

Input: An initial point $x_0 \in R^n$, a symmetric positive definite matrix $B_0 \in R^{n\times n}$, $k_{\max} \in N$, $\rho, \mu \in (0,1)$, $\eta_0 \in [\eta_{\max}, \eta_{\min}]$ and $\varepsilon > 0$.

Begin $\Delta_0 = \|g_0\|, R_0 = f_0, p = 0, r_0 = 0, k = 0$.

While ($\|g_k\| \geq \varepsilon$ and $k \leq k_{\max}$) {Start of outer loop}

While ($r_k < \mu$) {Start of inner loop}

Specify the trial point d_k by solving the sub-problem (2).

Determine the trust-region ratio r_k using (6).

If $r_k < \mu$, set $p = p+1$ and update the trust region radius Δ_k with (8).

Else $x_{k+1} = \hat{x}_{k+1}$.

Break;

End if

End while {End of inner loop}

$p = 0$; Determine Δ_k using (8); update B_{k+1} if $s_k^T y_k > 0$; generate η_{k+1} by an adaptive formula; calculate D_{k+1} by (5); $k = k+1$

End while {End of outer loop}

End

CONVERGENCE ANALYSIS

In this section, we prove the some properties of the new algorithm that are prominent to prove its convergence analysis. Throughout the paper, we consider the following assumptions in order to analyze the convergence of the new algorithm: (H1) the objective function $f(x)$ has a lower bound on R^n and $g(x) = \nabla f(x)$ is uniformly continuous on open convex set Ω that contains the level set $L(x_0) = \{x \in R^n \mid f(x) \leq f(x_0)\}$. (H2) B_k is uniformly bounded, i.e., there exists a constant $M > 0$ such that $\|B_k\| \leq M$, for all k.

Remark 1: To establish strong theoretical results, it is supposed that the model $m_k(d)$ decreases at least as much as a fraction of that obtained in Cauchy point, i.e. there exists $0 < \beta < 1$ such that, for all k,

$$m_k(0) - m_k(d_k) \geq \beta \|g_k\| \min\{\Delta_k, \frac{\|g_k\|}{\|B_k\|}\} \quad (11)$$

Remark 2: If $f(x)$ is a twice continuously differentiable function and the level set $L(x_0)$ is bounded, then (H1) implies that $\left\|\nabla^2 f(x)\right\|$ is uniformly continuous and bounded on the open bounded convex set Ω that contains $L(x_0)$. Hence, there exists a constant M_1 such that $\left\|\nabla^2 f(x)\right\| \le M_1$ and by using mean value theorem we have
$\|g(x)-g(y)\| \le M_1 \|x-y\| \quad \forall x, y \in \Omega$

Lemma 1: Suppose that the sequence $\{x_k\}$ be generated by algorithm 1. Then, for all $k \in N$, we have
$m_k(0)-m_k(d_k) \ge m_k(0)-m_k(\sigma_k d_k) \ge -\frac{1}{2}\sigma_k g_k^T q_k$
where d_k is the optimal solution of the sub-problem (2) with respect to $\sigma_k \le s_k$.
Proof. A proof of this lemma can be observed in [5].

Lemma 2: Suppose that the sequence $\{x_k\}$ be generated by Algorithm 1, then we have
$|m_k(d_k)-f(x_k+d_k)| \le O(\|d_k\|^2)$
Proof. See Coon, Gould and Toint [14].

Lemma 3: Suppose that the sequence $\{x_k\}$ be generated by Algorithm 1, then we have
$$f_{k+1} \le D_{k+1} \le D_k \tag{12}$$
Proof. Let iterate k be a successive iterate. From $r_k \ge \mu$ and (11), we have
$$D_k - f_{k+1} \ge \mu \operatorname{Pr} ed_k \ge \mu\beta \|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\} \ge 0 \tag{13}$$
$$D_{k+1} - f_{k+1} = \eta_{k+1}(D_k - f_{k+1}) \ge \mu\beta\eta_{k+1} \|g_k\| \min\left\{\Delta_k, \tfrac{\|g_k\|}{\|B_k\|}\right\} \ge 0 \tag{14}$$
From inequalities (13) and (14), we have $f_{k+1} \le D_{k+1}$.
$$D_{k+1} = \eta_{k+1} D_k + (1-\eta_{k+1}) f_{k+1} \le \eta_{k+1} D_k + (1-\eta_{k+1}) D_{k+1} \tag{15}$$
Thus, $\eta_{k+1}(D_k - D_{k+1}) \ge 0$.
Hence, (12) holds and the proof is completed.
Lemma 4: Suppose that (H1) and (H2) hold and the sequence $\{x_k\}$ be generated by Algorithm 1. Then inner loop is well-defined.
Proof. We prove $r_k \ge \mu$, for sufficiently large p_k. Let d_k^p is a solution of the sub-problem (2) corresponding to $p-th$ inner loop execution in x_k. From Lemma 3.1, it obtains that
$m_k(0)-m_k(d_k) \ge -\frac{1}{2}\sigma_k g_k^T q_k$
This fact together with Lemma 3.2 imply
$$\left|\frac{f(x_k)-f(x_k+d_k^p)}{m_k(0)-m_k(d_k^p)} - 1\right| = \left|\frac{f(x_k)-f(x_k+d_k^p)-(m_k(0)-m_k(d_k^p))}{m_k(0)-m_k(d_k^p)}\right|$$
$$\le \frac{O(\|d_k^p\|^2)}{-\frac{1}{2}\sigma_k g_k^T q_k} \le \frac{O(\Delta_{k(p)}^2)}{-\frac{1}{2}\Delta_{k(p)} g_k^T q_k / \|q_k\|} \le \frac{O(\Delta_{k(p)})}{-\frac{1}{2} g_k^T q_k / \|q_k\|}$$
where the last inequality is obtained using (2) and (8). If $i \to \infty$, then $\sigma_{k(p)} = \rho^p s_k \|q_k\| \to 0$ and using (8), right hand side of the preceding inequality tends to zero. Thus, using (12), we obtain
$$\frac{C_k - f(x_k+d_k^p)}{m_k(0)-m_k(d_k^{pk})} \ge \frac{f(x_k)-f(x_k+d_k^{pk})}{m_k(0)-m_k(d_k^{pk})} \ge \mu$$
Therefore, for sufficiently large p_k, we get $r_k \ge \mu$. This straight forwardly implies that inner loop of the algorithm is well-defined.

Lemma 5: Suppose that (H1) and (H2) hold and the sequence $\{x_k\}$ be generated by Algorithm 1. Then we have

$$\lim_{k\to\infty} D_k = \lim_{k\to\infty} f(x_k) \tag{16}$$

Proof. Due to the definition of D_k, we have

$$D_{k+1} = \eta_{k+1} D_k + (1-\eta_{k+1}) f_{k+1}$$

By (12), we obtain

$$f_{k+1} - D_{k+1} = \eta_{k+1}(f_{k+1} - D_k) \le \eta_{k+1}(D_{k+1} - D_k) \tag{17}$$

From lemma 3.3, we know that $\{D_k\}$ is convergent. We notice that $\eta_{\min} \in [0,1)$, $\eta_{\max} \in [\eta_{\min}, 1)$, $\eta_{k+1} \in [\eta_{\max}, \eta_{\min}]$. And as $k \to \infty$, we have

$$\lim_{k\to\infty} f_{k+1} - D_{k+1} \le \lim_{k\to\infty} \eta_{k+1}(D_{k+1} - D_k) = 0 \tag{18}$$

Therefore, the lemma is true.

Theorem 6: Suppose that (H1) and (H2) hold, then Algorithm 1 either stops at a stationary point of (1) or generates an infinite sequence $\{x_k\}$ such that

$$\lim_{k\to\infty} -\frac{g_k^T q_k}{\|q_k\|} = 0 \tag{19}$$

Proof. If Algorithm 1 does not stop at a stationary point, we prove that (19) holds. Suppose that Algorithm 1 generates the sequence $\{x_k\}$ and

$$\lim_{k\to\infty} -\frac{g_k^T q_k}{\|q_k\|} \ne 0$$

Thus, there exists ε_0 and an infinite subset $K \subseteq \{0,1,2,\cdots\}$ such that

$$-\frac{g_k^T q_k}{\|q_k\|} \ge \varepsilon_0 \quad \forall k \in K \tag{20}$$

From (H2), we know that there exists a constant $M > 0$ such that $\|B_k\| \le M$, for all $k \in N$. Hence we have that

$$q_k^T B_k q_k \le M \|q_k\|^2 \quad \forall k \in N \tag{21}$$

By defining $K_1 = \{k \in K | \sigma_k = s_k\}$ and $K_2 = \{k \in K | \sigma_k < s_k\}$, it is obvious that $K = K_1 \cup K_2$ is an infinite subset of $\{0,1,2,\cdots\}$. We now prove that neither K_1 nor K_2 can be an infinite set contradicting with (20).

First, we assume that K_1 is an infinite subset of K. Lemma 3.1 and (21) lead us to

$$\begin{aligned} & D_k - f(x_k + d_k) \ge \mu(m_k(0) - m_k(d_k)) \\ & \ge -\tfrac{1}{2}\mu\sigma_k g_k^T q_k \ge -\tfrac{1}{2}\mu s_k g_k^T q_k \qquad\qquad k \in K_1 \\ & = \tfrac{1}{2}\mu \frac{(g_k^T q_k)^2}{q_k^T B_k q_k} \ge \frac{\mu}{2M} \frac{(g_k^T q_k)^2}{\|q_k\|} \ge \frac{\mu}{2M}\varepsilon_0^2 \end{aligned}$$

The previous inequality together with Lemma 3.5, as $k \to \infty$, suggest

$$\frac{\mu}{2M}\varepsilon_0^2 \le 0$$

This is a contradiction. Thus K_1 can not be an infinite subset of K.

Now, we let that K_2 be an infinite subset of K. From Lemma 3.1, we get

$$\begin{aligned} & D_k - f(x_k + d_k) \ge \mu(m_k(0) - m_k(d_k)) \\ & \ge -\tfrac{1}{2}\mu\sigma_k \|q_k\| \frac{g_k^T q_k}{\|q_k\|} \ge \tfrac{1}{2}\mu\sigma_k \|q_k\| \varepsilon_0 \end{aligned}$$

This inequality along with Lemma 3.5, as $k \to \infty$, imply that

$$\lim_{k\to\infty} \Delta_k = \lim_{k\to\infty} \sigma_k \|q_k\| = 0, \quad k \in K_2 \tag{22}$$

Now, suppose that $\tilde{d}_k$ is an optimal solution of the following sub-problem

$$\min \ g_k^T d + \tfrac{1}{2} d^T B_k d \qquad s.t. \ \|d\| \le \tilde{\Delta}_k, \ \tilde{\Delta}_k = \tfrac{\Delta_k}{\rho}$$

From the definition of Δ_k, it is clear that

$$\frac{D_k - f(x_k + \tilde{d}_k)}{m_k(0) - m_k(d_k)} < \mu, \quad \|\tilde{d}_k\| \le \tilde{\Delta}_k, \quad \forall k \in K_2 \tag{23}$$

On the other hand, (22) suggests that

$$\lim_{k\to\infty} \tilde{\Delta}_k = 0, \quad k \in K_2 \tag{24}$$

Using Lemma 3.1, (20) and (24), for $k \in K_2$, we can write

$$\left| \frac{f(x_k) - f(x_k + \tilde{d}_k)}{m_k(0) - m_k(d_k)} - 1 \right| = \left| \frac{f(x_k) - f(x_k + \tilde{d}_k) - (m_k(0) - m_k(d_k))}{m_k(0) - m_k(d_k)} \right|$$

$$\le \frac{O(\tilde{\sigma}_k^2 \|q_k\|^2)}{-\frac{1}{2}\tilde{\sigma}_k g_k^T q_k} \le \frac{O(\tilde{\sigma}_k^2)}{-\frac{1}{2}\tilde{\sigma}_k g_k^T q_k / \|q_k\|} \le \frac{O(\tilde{\sigma}_k^2)}{-\frac{1}{2}\varepsilon_0} \to 0$$

Thus, for sufficiently large $k \in K_2$, we get

$$\frac{C_k - f(x_k + \tilde{d}_k)}{m_k(0) - m_k(d_k)} \ge \frac{f(x_k) - f(x_k + \tilde{d}_k)}{m_k(0) - m_k(d_k)} \ge \mu \tag{25}$$

This is a contradiction with (23). Hence, there exists no infinite subset of K such that (20) holds. Therefore, the proof is completed.

Theorem 7: Suppose that all conditions of Theorem 3.6 hold and q_k satisfies (7). Then Algorithm 1 either stops finitely or generates an infinite sequence $\{x_k\}$ such that

$$\lim_{k\to\infty} \|g_k\| = 0$$

Proof. If Algorithm 1 stops finitely, the proof is obvious. Otherwise, Theorem 3.6 indicates that Algorithm 1 generates an infinite sequence $\{x_k\}$ satisfying in (19).Since q_k satisfies (7), we have

$$0 \le \tau \|g_k\| \le -\frac{g_k^T q_k}{\|g_k\|\|q_k\|} \|g_k\| = -\frac{g_k^T q_k}{\|q_k\|} \to 0$$

Therefore, we have $\lim_{k\to\infty} \|g_k\| = 0$.

CONCLUDING REMARKS

Trust region methods are generally considered to be reliable and effective methods for nonlinear unconstrained optimization, and thus it is worth to improve their structures. In this paper, we combine an adjustable trust region radius with an effective non-monotone technique to propose a novel hybrid non-monotone adjustable trust region method. The radius can be adjusted automatically according to the current iterative information to reduce the total number of iterates and function evaluations.

ACKNOWLEDGEMENT

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

REFERENCES

1. Fletcher, R. (2000). Practical Method of Optimization, Unconstrained Optimization, *Wiley, New York*.
2. Nocedal, J., & Wright, S. J. (2006). Numerical Optimization, *Springer, New York*, 2006.
3. Amini, K., & Ahookhosh, M. (2011). Combination adaptive trust region method by non-monotone strategy for unconstrained nonlinear programming. *Asia-Pacific Journal of Operational Research,* 28 (5):585-600.
4. Sartenaer, A. (1997). Automatic determination of an initial trust region in nonlinear programming. *SIAM Journal on Scientific Computing,* 18(6):1788-1803.
5. Shi, Z. J., & Guo, J. H. (2008). A new trust region method for unconstrained optimization. *Journal of Computational and Applied Mathematics*, 213:509-520.
6. Zhang, X. S., Zhang, J. L., & Liao, L. Z. (2002). An adaptive trust region method and its convergence. *Science in*

China, 45:620-631.
7. Grippo, L., Lampariello, F., & Lucidi, S. (1986). A nonmonotone line search technique for Newton's method. *SIAM Journal on Numerical Analysis*, 23:707-716.
8. Grippo, L., Lampariello, F., & Lucidi, S. (1989). A truncated Newton method with nonmonotone line search for unconstrained optimization. *Journal of Optimization Theory and Applications*, 60:401-419.
9. Toint, Ph. L. (1996). An assessment of nonmonotone line search technique for unconstrained optimization. *SIAM Journal on Scientific Computing,* 17:725-739.
10. Deng, N. Y., Xiao, Y., & Zhou， F. J. (1993). Nonmonotone trust region algorithm. *Journal of Optimization Theory and Applications,* 76:259-285.
11. Toint, Ph. L. (1997). Non-monotone trust-region algorithm for nonlinear optimization subject to convex constraints. *Mathematical Programming* 77:69-94.
12. Zhang, X. S., Zhang, J. L., & Liao, L. Z. (2003). A nonmonotone adaptive trust region method and its convergence. *Computers and Mathematics with Applications,* 45:1469-1477.
13. Gu, N. Z., & Mo, J. T. (2008). Incorporating nonmonotone strategies into the trust region method for unconstrained optimization. *Journal of Computers & Mathematics with Applications*, 55(9):2158-2172.
14. Conn, A. R., Gould, N. I. M., & Toint, Ph. L. (2000). Trust-region Methods, Society for Industrial and Applied Mathematics, *SIAM, Philadelphia, USA*.
15. Powell, M. J. D. (1984). On the global convergence of trust region algorithms for unconstrained optimization. *Mathematical Programming,* 29:297-303.