Non-Monotone Trust Region Method Combined with Line Search Strategy

Yunfeng ZHANG, Qinghua ZHOU
College of Mathematics and Information Science, Hebei University, Baoding, China

Abstract: This paper revises some trust-region methods equipped with non-monotone strategies for solving unconstrained optimization problems. Unlike the traditional non-monotone trust-region method, our proposed algorithm avoids resolving the sub-problem whenever a trial step is rejected. Instead, it performs a non-monotone Armijo-type line search in direction of the rejected trial step to construct a new point. Theoretical analysis indicates that the new approach preserves the global convergence to the first-order critical points under classical assumptions.

Keywords: Unconstrained optimization; Trust-region method; Armijo-type line search; Non-monotone technique

1. Introduction

We consider the unconstrained minimization problem

$$\min\ f(x),\ \text{subject to } x \in R^2, \quad (1)$$

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:

$$\min\quad m_k(d) = f_k + g_k{}^T d + \frac{1}{2} d^T B_k d \quad (2)$$

where $f_k = f(x_k)$, $g_k = \nabla f(x_k)$, B_k is the exact Hessian $G_k = \nabla^2 f(x_k)$, or a symmetric approximation for it, $\delta_k > 0$ is a real number called trust-region radius, and $\|\cdot\|$ denotes the Euclidean norm. Once the step d is computed, the quality of the model in the trust-region is evaluated by a ratio of the actual reduction of objective, $f_k - f(x_k + d)$, to the predicted reduction of model, $m_k(0) - m_k(d)$, i.e.,

$$r_k = \frac{f_k - f(x_k + d)}{m_k(0) - m_k(d)}. \quad (3)$$

In particular, Armijo's line search satisfies

$$f(x_k + \alpha_k d_k) \le f_k + \sigma \alpha_k g_k{}^T d_k \quad (4)$$

where $\sigma \in (0, 1/2)$, and α_k is the largest $\alpha \in \{s, \rho s, \ldots\}$ with $s > 0$ and $\rho \in (0,1)$ such that (2) holds, see [9].

To improve the performance of Armijo's line search, Grippo et al. in 1986 [8] introduced a variant of Armijo's rule using the term $f_{l(k)}$ in place of f_k in (4) defined by

$$f_{l(k)} = \max_{0 \le j \le m(k)} \{ f_{k-j} \},\ k = 0,1,2,\ldots, \quad (5)$$

where $m(0) = 0, m(k) \le \min\{m(k-1)+1, N\}$ for positive integer N. It was shown that the associated scheme is globally convergent, and numerical results reported in Grippo et al. [10] and Toint [11] showed the effectiveness of the proposed idea. Motivated by these results, the non-monotone strategies have received much attention during past few decades. For example, in 2004, Zhang & Hager in [12] proposed the non-monotone term

$$C_k = \begin{cases} f_0 & if \quad k = 0, \\ (\eta_{k-1} Q_{k-1} C_{k-1} + f(x_k)) / Q_k & if \quad k \ge 1, \end{cases}$$

$$Q_k = \begin{cases} 1 & if\ \ k = 0, \\ \eta_{k-1} Q_{k-1} + 1 & if\ \ k \ge 1, \end{cases}$$

where $0 \le \eta_{\min} \le \eta_{k-1} \le \eta_{\max} \le 1$. Mo et al. in [13] and Ahookhosh et al. in [3] studied the non-monotone term

$$D_k = \begin{cases} f_k & if\ \ k = 1, \\ \eta_k D_{k-1} + (1 - \eta_k) f_k & if\ \ k \ge 2, \end{cases}$$

where $\eta_k \in [\eta_{\min}, \eta_{\max}], \eta_{\min} \in [0,1], \eta_{\max} \in [\eta_{\min}, 1]$. Recently, Amini et al. in [7] proposed the non-monotone term

$$R_k = \eta_k f_{l(k)} + (1 - \eta_k) f_k,$$

where $0 \le \eta_{\min} \le \eta_{\max} \le 1$ and $\eta_k \in [\eta_{\min}, \eta_{\max}]$.

M. Ahookhosh and S. Ghaderi in [2] proposed a novel non-monotone strategy based on a weighted average of former successive iterates. T_k is defined as follows

$$T_k = \begin{cases} f_{l(k)} & if \quad k < N, \\ \max\{\bar{T}_k, f_k\} & if \quad k \ge N, \end{cases}$$

$$\bar{T}_k = (1-\eta_{k-1}) f_k + \eta_{k-1}\bar{T}_{k-1} + \xi_k \left(f_{k-N} - f_{k-N-1} \right) \quad (6)$$

where

$$\xi_k = \eta_{k-1}\eta_{k-2}...\eta_{k-N-1} = \frac{\eta_{k-1}}{\eta_{k-N-2}}\eta_{k-2}...\eta_{k-N-1}\eta_{k-N-2} = \frac{\eta_{k-1}}{\eta_{k-N-2}}\xi_{k-1}$$

. It is clear that the new term uses a stronger term $f_{l(k)}$ defined by (5) for first $k < N$ iterations and then employs the relaxed convex term proposed above.

The non-monotone techniques have changed the ratio (3). It defined as follows

$$\hat{r}_k = \frac{T_k - f(x_k + d)}{m_k(0) - m_k(d)}, \quad (7)$$

After introducing the novel non-monotone term, we now state 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 Zhang and Hager.

Zhang et al. [6] proposed an adjustable strategy to determine the trust-region radius based on information of g_k and B_k in current iterate. The method exploits the following adaptive formula $\delta_k = \rho^{p_k} \|g_k\| \cdot \|\hat{B}_k^{-1}\|$,for updating the radius of the neighborhood in problem (2), in which $\rho \in (0,1)$, p_k is a nonnegative integer, and $\hat{B}_k = B_k + i\mathrm{I}$ is a positive definite matrix for some $i \in \mathrm{N}$.

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 $-\frac{g_k^T q_k}{\|g_k\| \cdot \|q_k\|} \ge \tau$ with $\tau \in (0,1)$, and set $s_k = -\frac{g_k^T q_k}{q_k^T \hat{B}_k q_k}$,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 $q_k^T \hat{B}_k q_k = q_k^T B_k q_k + i\|q_k\|^2 > 0$ .So, they proposed a new trust region radius as follows

$$\delta_k = \alpha_k \|q_k\|, \quad (8)$$

where $\alpha_k = \rho^{p_k} s_k$, and p_k is the least positive integer number so that

$$\hat{r}_k \ge \mu. \quad (9)$$

This paper organized as follows. In Section 2, we describe the novel trust region line search algorithm and give its properties. In Section 3, we first prove that the new algorithm is well defined, and then the global convergence is investigated. Finally, some conclusions are given in Section 4.

2. Novel Trust-region line Search Algorithm

Now, we have everything to describe our algorithm. We first solve the sub-problem (2) in order to compute the trial step d_k and then compute the ratio (7). If $\hat{r}_k \ge \mu$, then we accept the trial step and set $x_{k+1} = x_k + d_k$. Otherwise, we determine the step-length $\alpha_k \in \{s, \rho s, \rho^2 s...\}$ by subsequent Armijo-type line search

$$f(x_k + \alpha_k d_k) \le T_k + \sigma\alpha_k g_k^T d_k, \quad (10)$$

where s is a positive constant, $\rho \in (0,1)$ and $\sigma \in (0,1/2)$. In this case, we set $x_{k+1} = x_k + \alpha_k d_k$. Now, we can outline our new non-monotone trust-region line search algorithm as follows:

2.1. New Non-monotone Trust-Region Line Search Algorithm

step1. An initial point $x_0 \in R^n$, a symmetric matrix $B_0 \in R^{n\times n}$ and initial trust-region radius $\delta_0 > 0$ are given. The constants $0 < \mu < 1$, $0 < \rho < 1$, $0 < \sigma < 1/2$, $0 \le \eta_{\min} < 1$ and $\eta_{\min} \le \eta_{\max} < 1$, $N \ge 0$, $\theta > 0$ and $\varepsilon > 0$ are also given.

Compute $f(x_0)$ and set $k = 0$.

step2. Compute $g(x_k)$. If $\|g(x_k)\| \le \varepsilon$, stop.

step3. Solve the sub-problem (2) to determine a trial step d_k that satisfies $\|d_k\| \le \delta_k$.

step4. Compute T_k and $\hat{r}_k$. If $\hat{r}_k \ge \mu$, set $x_{k+1} = x_k + d_k$ and go to Step 5. Otherwise, find the step-length α_k satisfying in (10), and set $x_{k+1} = x_k + \alpha_k d_k$. Update the trust-region radius by $\delta_{k+1} = \min\{\theta\|x_{k+1} - x_k\|, \delta_k\}$ and go to Step 6.

step5. Set $\delta_k = \alpha_k \|q_k\|$.

step6. Update the matrix B_{k+1} by a quasi-Newton formula, set $k = k+1$ and go to Step 2.

For convenience, we define two index sets as below, $I = \{k : \hat{r}_k \ge \mu\}$ and $J = \{k : \hat{r}_k < \mu\}$.

The following assumptions are used to analyze the convergence properties of Algorithm:

(H1) The objective function f is continuously differentiable and has a lower bound on R^n and is uniformly continuous on open convex set Ω that contains the level set $L(x_0) = \{x \in R^n \mid f(x) \le f(x_0), x_0 \in R^n\}$.

(H2) The sequence $\{B_k\}$ is uniformly bounded, i.e., there exists a constant $M_1 > 0$ such that $\|B_K\| \le M_1$.

(H3) There exists a constant $c > 0$ such that the trial step d_k satisfies $\|d_k\| \le c\|g_k\|$.

(H4) There exists a positive constant m such that, for all $d \in R^n$ and $k \in N$, we have $m\|d\|^2 \le d^T B_k d$.

Remark 2.1 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, $m_k(0) - m_k(d) \ge \beta\|g_k\| \min\left\{\delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}$ and $g_k{}^T d_k \le -\beta\|g_k\| \min\left\{\delta_k, \frac{\|g_k\|}{\|B_k\|}\right\}$.

Remark 2.2 If $f(x)$ is a twice continuously differentiable function and the level set $L(x_0)$ is bounded, then (H1) implies that $\|\nabla^2 f(x)\|$ is uniformly continuous and bounded on the open bounded convex set Ω that contains $L(x_0)$. Hence, there exists a constant $M_2 > 0$ such that $\|G_k = \nabla^2 f(x)\| \le M_2$ and by using mean value theorem we have $\|g(x) - g(y)\| \le M_2\|x - y\|, \forall x, y \in \Omega$.

The following results are required for establishing the global convergence of the algorithm in the next section.

Lemma 2.1 ([4]) Suppose that the sequence $\{x_k\}$ be generated by Algorithm. Then, for all $k \in \mathrm{N}$, we have $m_k(0) - m_k(d) \ge m_k(0) - m_k(\alpha_k q_k) \ge -\frac{1}{2}\alpha_k g_k{}^T q_k$, where d_k is the optimal solution of the sub-problem (2) with respect to $\alpha_k \le s_k$.

Lemma 2.2 ([14]) Suppose that sequence $\{x_k\}$ is generated by Algorithm, then $\left| f_k - f(x_k + d_k) - (m_k(0) - m_k(d_k)) \right| \le O\left(\|d_k\|^2\right)$.

Lemma 2.3 ([2]) Suppose that the sequence $\{x_k\}$ is generated by Algorithm, then we get

$$f_k \le T_k \le f_{l(k)}, \qquad (11)$$

for all $k \in \mathrm{N} \cup \{0\}$.

Lemma 2.4 Suppose that the sequence $\{x_k\}$ is generated by Algorithm. Then, for all $k \in \mathrm{N} \cup \{0\}$, we have $x_k \in L(x_0)$ and $\{f_{l(k)}\}$ is a decreasing sequence.

Proof. It is clear that $T_0 = f_0$. By induction, we show that $x_k \in L(x_0)$, for all $k \in \mathrm{N} \cup \{0\}$. We assume that $x_i \in L(x_0)$, for $i = 1, 2, \ldots, k$. We then prove that $x_{k+1} \in L(x_0)$. To do so, we consider two cases:

Case 1. $k \in I$: We have $T_k - f(x_k + \alpha_k d_k) \ge \mu(m_k(0) - m_k(d_k)) \ge 0$.

Case 2. $k \in J$: using (10) and Remark 2.1, we have $f(x_k + \alpha_k d_k) \le T_k + \sigma\alpha_k g_k{}^T d_k \le T_k$.

These two inequalities along with (11) show that

$$f_{k+1} \le T_k \le f_{l(k)} \le f_0. \qquad (12)$$

Thus, the sequence x_k is contained in $L(x_0)$.

Now, we prove that the sequence $f_{l(k)}$ is a decreasing sequence. To this end, we consider two cases based on $k \in I$ or $k+1 \in J$.

For $k < N$, it is obvious that $m(k) = k$. Since, for any k, $f_k \le f_0$ then we have $f_{l(k)} = f_0$.

For $k \ge N$, we have $m(k+1) \le m(k) + 2$. Thus, from the definition of $f_{l(k)}$ and (11), we can write

$$f_{l(k+1)} = \max_{0 \le j \le m(k+1)} \{f_{k-j}\} \le \max_{0 \le j \le m(k)+1} \{f_{k-j+1}\} = \max\{f_{l(k)}, f_{k+1}\} \le f_{l(k)}.$$

Both cases show that the sequence $\{f_{l(k)}\}$ is a decreasing sequence.

Corollary 2.1 Suppose that (H1) holds and the sequence $\{x_k\}$ is generated by Algorithm. Then the sequence $\{f_{l(k)}\}$ is convergent.

3. Convergence Analysis

In this section, we discuss some convergence properties of the new algorithm, and prove the global convergence.

Lemma 3.1 Suppose that (H1)-(H3) hold and the sequence $\{x_k\}$ is generated by Algorithm, then $\lim_{k\to\infty} f_{l(k)} = \lim_{k\to\infty} f_k$.

Proof. A proof of this lemma can be observed in [2].

Corollary 3.1 Suppose (H1)-(H3) hold and the sequence $\{x_k\}$ is generated by Algorithm, then $\lim_{k\to\infty} T_k = \lim_{k\to\infty} f_k$.

Proof. A proof of this lemma can be observed in [2].

Lemma 3.2 Suppose that the sequence $\{x_k\}$ is generated by Algorithm 2.1. Then, Step 4 of the algorithm is well-defined.

Proof. We consider two cases:

Case 1. $k \in I$.

First we prove that when p is sufficiently large, (9) holds. Let d'^{i}_{k} be the solution of sub-problem (2) corresponding to $p=i$ at x_k, and $m_k(0)-m_k(d^i{}_k)$ be the predicted reduction corresponding to $p=i$ at x_k. It follows from Lemma 2.1 that

$$m_k(0)-m_k(d)\ge m_k(0)-m_k(\alpha_k q_k)\ge -\frac{1}{2}\alpha_k g_k{}^T q_k.$$

Using this inequality and Lemma 2.2, we have

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

$$\le\frac{O(\|d^i{}_k\|^2)}{-\frac{1}{2}\alpha_{k(i)}g_k{}^T q_k}\le\frac{O(\delta^2{}_{k(i)})}{-\frac{1}{2}\delta_{k(i)}g_k{}^T q_k/\|q_k\|}$$

$$=\frac{O(\delta_{k(i)})}{-\frac{1}{2}g_k{}^T q_k/\|q_k\|},$$

where the last inequality is obtained using (2) and (8). Now, as $i\to\infty$, then $\delta_{k(i)}=\rho^i s_k\|q_k\|\to 0$ 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 (11), we have

$$\hat{r}_k=\frac{T_k-f(x_k+d)}{m_k(0)-m_k(d)}\ge\frac{f_k-f(x_k+d)}{m_k(0)-m_k(d)}\ge\mu.$$

Therefore, when p is sufficiently large, $\hat{r}_k\ge\mu$.

Case 2. $k\in J$.

We prove that the line search terminates in the finite number of steps. For establishing a contradiction, assume that there exists $k\in J$ such that

$$f(x_k+\rho^i s_k d_k)>T_k+\sigma\rho^i s_k g_k{}^T d_k,\ \forall i\in\mathrm{N}\cup\{0\} \qquad (12)$$

From Lemma 2.3, we have $f_k\le T_k$. This fact, along with (12), implies that $\frac{f(x_k+\rho^i s_k d_k)-f_k}{\rho^i s_k}>\sigma g_k{}^T d_k$, $\forall i\in\mathrm{N}\cup\{0\}$.

Since f is a differentiable function, by taking a limit, as $i\to\infty$, we obtain $g_k{}^T d_k\ge\sigma g_k{}^T d_k$.

Using the fact that $\sigma\in(0,1/2)$, this inequality leads us to $g_k{}^T d_k\ge 0$ which contradicts Remark 2.1. Therefore, Step 4 in Algorithm 2.1 is well-defined.

Lemma 3.3 Suppose that (H3) hold and assume the sequence $\{x_k\}$ does not converge to a stationary point, i.e., there exists a constant $0<\varepsilon<1$ such that for all $k\in N$, we have $\|g(x_k)\|\le\varepsilon$. Then, we have $\lim_{k\to\infty}\min\left\{\delta_k,\frac{\varepsilon}{L_k}\right\}=0$ where $L_k=1+\max_{1\le i\le k}\|B_i\|$.

Proof. Similar to [5], the proof of this lemma can be observed.

Theorem 3.1 Suppose that (H1)-(H4) hold and the sequence $\{x_k\}$ is generated by Algorithm, then $\liminf_{k\to\infty}\|g_k\|=0$. Proof. The proof is similar to the proof of theorem 3.5 in [1].

4. Conclusion

In this paper, a variant non-monotone trust region algorithm for solving unconstrained optimization problem is proposed. Unlike traditional trust region method, the proposed algorithm does not reject a failed trial step, but performs a non-monotone 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.

5. Acknowledgements

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

References

[1] M. Ahookhosh, K. Amini, M.R. Peyghami, A non-monotone trust-region line search method for large-scale unconstrained optimization, Applied Mathematical Modeling, 36(1), (2012) 478-487.

[2] M. Ahookhosh, S. Ghaderi, Two globally convergent non-monotone trust-region methods for unconstrained optimization, Journal of Applied Mathematics and ComputingFebruary, 50(1), (2016) 529-555.

[3] M. Ahookhosh, K. Amini, S. Bahrami, A class of non-monotone Armijo-type line search method for unconstrained optimization, Optimization, 61(4), (2012) 387-404.

[4] Z.J. Shi, J.H. Guo, A new trust region methods for unconstrained optimization, Comput. Appl. Math. 213 (2008) 509-520.

[5] N. Gu, J. Mo, Incorporating non-monotone strategies into the trust region for unconstrained optimization, Comput. Math. Appl. 209 (2007) 97-108.

[6] X.S. Zhang, J.L. Zhang, L.Z. Liao, An adaptive trust region method and its convergence, Science in China 45 (2002) 620-631.

[7] K. Amini, M. Ahookhosh, H. Nosratipour, An inexact line search approach using modified nonmonotone strategy for unconstrained optimization, Numerical Algorithms, 66,(2014) 49-78.

[8] L. Grippo, F. Lampariello, S. Lucidi, A non-monotone line search technique for Newton's method, SIAM Journal on Numerical Analysis, 23, (1986) 707-716.

[9] L. Armijo, Minimization of functions having Lipschitz continuous first partial derivatives, Pacific Journal of Mathematics, 16, (1966)1-3.

[10] L. Grippo, F. Lampariello, S. Lucidi, A non-monotone line search technique for Newton's method, SIAM Journal on Numerical Analysis 23, (1986) 707-716.

[11] Ph.L. Toint, An assessment of non-monotone line-search technique for unconstrained optimization, SIAM Journal on Scientific Computing, 17, (1996) 725-739.

[12] H.C. Zhang, W.W. Hager, A non-monotone line search technique for unconstrained optimization, SIAM journal on Optimization 14(4), (2004) 1043-1056.

[13] J. Mo, C. Liu, S. Yan, A non-monotone trust region method based on non-increasing technique of weighted average of the successive function value, Journal of Computational and Applied Mathematics 209, (2007) 97-108 [4] (2014) 49-78.

[14] A.R. Conn, N.I.M. Gould, Ph.L. Toint, Trust-Region Methods, Society for Industrial and Applied Mathematics, SIAM, Philadelphia, USA, 2000.