A New Non-monotonic Self-adaptive Trust Region Algorithm with Non-monotonic Line Search

Xiao Wu[1], Qinghua Zhou[1,*], Changyuan Li[1], Xiaodian Sun[2]

[1]College of Mathematics and Information Science, Hebei University, Baoding, Hebei Province, China, 071002.

[2]Laboratory for Statistical Genomics and Systems Biology, Department of Environmental Health, University of Cincinnati College of Medicine, 3223 Eden Ave. ML56, Cincinnati OH 45267-0056, USA

(wuxiao616@163.com, qinghua.zhou@gmail.com(*corresponding author), mbydlcy@126.com, sunxd@uc.edu)

Abstract. ***We consider an efficient trust-region framework which employs a new non-monotone line search technique for unconstrained optimization problems. Unlike the traditional non-monotonic trust-region method, the new point is given by the non-monotonic Wolfe line search at each iteration, and the trust region radius is updated at a variable rate. The new algorithm solves the trust region sub-problem only once at each iteration. Under certain conditions, the global convergence of the algorithm is proved.***

Key words: *non-monotonic, self-adaptive, trust region, line search*

1. INTRODUCTION

We consider the following unconstrained optimization problem

$$\min_{x\in R^n} f(x). \qquad (1)$$

where $f(x)$ is twice continuously differentiable.

Trust region method is a very effective and robust technique for solving problem (1) to find the global optimal solution. At the kth step, it calculates a trial step d_k by solving the following trust region sub-problem

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

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. After obtaining d_k, the trust region method computes the ratio r_k between the actual reduction and the predicted reduction of the objective function to check if d_k is acceptable. Then the trust region radius Δ_k is also updated according to the value of r_k, since r_k reflect the extent to which the quadratic model $\phi_k(d)$ approximates the objective function $f(x_k + d)$. Hei [1] proposed a self-adaptive trust region algorithm, in which Δ_k is updated by R-function at a variable rate.

Non-monotonic line search technique for unconstrained optimization is first proposed by Grippo et al. [2]. Due to its high efficiency, many authors generalized the non-monotone technique to trust region method and proposed non-monotone trust region method, see [3,4].

Yang and Sun [5] proposed a new region algorithm where the trust region radius is updated at a variable rate.

Moreover, the new algorithm performed a backtracking line search from the failed point instead of resolving the trust region sub-problem.

Wang [6] presented a non-monotonic trust region algorithm with line search. Unlike traditional non-monotonic trust region algorithms, the new point is given by the non-monotonic Wolfe line search at each iteration, and trust region radius is adjusted by sub-problem approximate solution and line search step length.

Mosoud Ahookhosh [7] introduced a variant non-monotone strategy and incorporate it into trust-region framework to construct more reliable approach. The new non-monotone strategy is a convex combination of the maximum of function value of some prior successful iterates and the current function value.

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

$$f(x_k+\alpha_k d_k)\le R_k+\beta\alpha_k g_k^T d_k.$$

$$g(x_k+\alpha_k d_k)^T d_k \ge \gamma g_k^T d_k.$$

where $0<\beta<\gamma<1$ and

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

where $\eta_k\in[\eta_{\min},\eta_{\max}]$ for $\eta_{\min}\in[0,1)$ and $\eta_{\max}\in[\eta_{\min},1]$.And we improve the self-adaptive trust region algorithm proposed in [1] by adopting the above ideas: backtracking line search and non-monotonic technique. Now, we can outline our new non-monotone self-adaptive trust region algorithm with non-monotonic line search.

2. NEW ALGORITHM

Before describing the new algorithm, we need the following definition and conclusion ([5]).

Definition 1 Any one-dimensional function $R_\eta(t)$ defined in $R=(-\infty,+\infty)$ with the parameter $\eta\in(0,1)$ is an R *-function* if and only if it satisfies:

1) $R_\eta(t)$ is non-decreasing in $(-\infty,+\infty)$;
2) $\lim_{t\to-\infty} R_\eta(t)=\beta_1$, where $\beta_1\in(0,1)$ is a small constant;
3) $R_\eta(t)\le 1-\gamma_1$, for all $t<\eta$ where $\gamma_1\in(0,1-\beta_1)$ is a constant;
4) $R_\eta(t)=1+\gamma_2$, where $\gamma_2\in(0,+\infty)$ is a constant;
5) $\lim_{t\to+\infty} R_\eta(t)=\beta_2$, where $\beta_2\in(1+\gamma_2,+\infty)$ is a constant.

Theorem 2 An R *-function* $R_\eta(t)$ (where $\eta\in(0,1)$) satisfies

$$0<\beta_1\le R_\eta(t)\le 1-\gamma_1<1, \forall t\in(-\infty,\eta). \quad (3)$$

$$1<1+\gamma_2\le R_\eta(t)\le\beta_2<+\infty, \forall t\in[\eta,+\infty). \quad (4)$$

Now, we give a description of our new algorithm.

Algorithm 1

Step 1 Given $x_0, B_0\in R^{n\times n}, \Delta_0>0, \varepsilon\ge 0, 0<\beta_1<1,$ $0<\gamma_1<1-\beta_1, \gamma_2>0, \beta_2>1+\gamma_2, 0<c_1<1, N>0,$ $0<\beta<\gamma<1$. Set $k=0, m(0)=0$.

Step 2 Compute $g(x_k)$. If $\|g(x_k)\|\le\varepsilon$, stop.

Step 3 Solve the sub-problem (2) for d_k satisfying

$$pred_k\ge\tau\|g_k\|\min\{\Delta_k,\|g_k\|/\|B_k\|\}. \quad (5)$$

$$g_k^T d_k\le-\tau\|g_k\|\min\{\Delta_k,\|g_k\|/\|B_k\|\}. \quad (6)$$

Step 4 Compute $f_{l(k)}, R_k$ and $\rho_k=(R_k-f(x_k+d_k))/pred_k$,find the step-length α_k satisfying in

$$f(x_k+\alpha_k d_k)\le R_k+\beta\alpha_k g_k^T d_k. \quad (7)$$

$$g(x_k+\alpha_k d_k)^T d_k\ge\gamma g_k^T d_k. \quad (8)$$

Set $x_{k+1}=x_k+\alpha_k d_k$, go to Step 5.

Step 5 $\Delta_{k+1}=R_{c_1}(\rho_k)\Delta_k$. Update the matrix B_{k+1} by a quasi-Newton formula, set

$$m(k)=\min\{m(k-1)+1, N\}, k=k+1.$$

go to Step 1.

3. CONVERGENCE ANALYSIS

To analyze the new algorithm, we make the following assumptions.

Assumption

(H1) $f(x)$ is continuously differentiable and has a lower bound on the level set $L(x)=\{x \mid f(x)\leq f(x_0)\}$ is compact.

(H2) The matrix B_k is an uniformly bounded matrix, i.e. there exist $b>0$ such that $\| B_k \|\leq b$ for all $k\in N$.

(H3) $\nabla f(x)$ is uniformly continuous, there exist a constant $L>0$ such that

$$\| \nabla f(x)-\nabla f(y)\|\leq L\| x-y\|, \forall x,y\in R^n. \quad (9)$$

(H4) There exists a constant e such that the trial step d_k satisfies

$$\| d_k \|\leq e\| g_k \|. \quad (10)$$

Lemma 3 Suppose that the sequence $\{x_k\}$ be generated by Algorithm 1, then the sequence $\{f_{l(k)}\}$ is a decreasing sequence.

Proof. Using definition of R_k and $f_{l(k)}$, we observe that

$$\begin{aligned} R_k &= \eta_k f_{l(k)}+(1-\eta_k)f_k \\ &\leq \eta_k f_{l(k)}+(1-\eta_k)f_{l(k)} \\ &= f_{l(k)} \end{aligned} \quad (11)$$

$$\begin{aligned} f(x_k+\alpha_k d_k) &\leq R_k+\beta\alpha_k g_k^T d_k \\ &\leq R_k. \end{aligned} \quad (12)$$

These two inequalities show that

$$f_{k+1}\leq R_k\leq f_{l(k)}. \quad (13)$$

We have $m(k+1)\leq m(k)+1$, thus from the definition of $f_{l(k+1)}$ and (13), we can write

$$\begin{aligned} f_{l(k+1)} &= \max_{0\leq j\leq m(k+1)}\{f_{k-j+1}\} \\ &\leq \max_{0\leq j\leq m(k)+1}\{f_{k-j+1}\} \\ &= \max\{f_{l(k)}, f_{k+1}\} \\ &\leq f_{l(k)}. \end{aligned} \quad (14)$$

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

Lemma 4 Suppose that (H1)-(H4) hold and the sequence $\{x_k\}$ be generate by Algorithm 1, then we have

$$\lim_{k\to\infty} f(x_{l(k)})=\lim_{k\to\infty} f(x_k). \quad (15)$$

Proof. Using (7) and (11), we obtain

$$\begin{aligned} f(x_{l(k)}) &= f(x_{l(k)-1}+\alpha_{l(k)-1}d_{l(k)-1}) \\ &\leq R_{l(k)-1}+\beta\alpha_{l(k)-1}g_{l(k)-1}^T d_{l(k)-1} \\ &\leq f_{l(l(k)-1)}+\beta\alpha_{l(k)-1}g_{l(k)-1}^T d_{l(k)-1}. \end{aligned}$$

Preceding inequality, along with $\beta>0$ and Corollary 2.1 in [8] implies that

$$\lim_{k\to\infty}\alpha_{l(k)-1}g_{l(k)-1}^T d_{l(k)-1}=0.$$

Thus using (10), we can conclude that

$$\lim_{k\to\infty}\alpha_{l(k)-1}\| d_{l(k)-1}\|=0.$$

The rest of the proof is similar to a theorem in [2].

Corollary 5 Suppose that the sequence $\{x_k\}$ be generated by Algorithm 1, then we have

$$\lim_{k\to\infty} R_k=\lim_{k\to\infty} f(x_k). \quad (16)$$

Proof. The proof is similar to that of Corollary 8 in [7].

Lemma 6 Suppose that (H2) and (H3) hold. Let α_k be sequence generated by Algorithm 1 and $\{\Delta_k\}$ are bounded upper, if $g(x)$ is uniformly continuous satisfies

$$\| g_k \|\geq\varepsilon>0 \quad \forall k.$$

where $\varepsilon>0$, then there exists a constant $\tilde{\alpha}>0$ such that

$$\alpha_k\geq\tilde{\alpha} \quad \forall k. \quad (17)$$

Proof. By the same way as in the proof of Lemma 5.1 in [9], we have the conclusion.

Lemma 7 If $g(x)$ is uniformly continuous and the sequence $\{x_k\}$ be generated by Algorithm 1 satisfies

$$\| g_k \|\geq\varepsilon>0. \quad (18)$$

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 02 Issue: 04 | July-2015 www.irjet.net p-ISSN: 2395-0072

where $\varepsilon > 0$ is a constant, then there exists a constant $c > 0$ such that

$$\Delta_k \geq \frac{c}{M_k}, k = 0, 1, \cdots \tag{19}$$

where M_k is defined by $M_k = 1 + \max_{1 \leq i \leq k} \| B_k \|$.

Proof. Because $\nabla f(x)$ is uniformly continuous, there exist a positive number σ such that

$$\begin{aligned} & d_k^T [\nabla f(x+d) - \nabla f(x)] \\ & \leq \frac{1}{2} \tau (1-c_1) \varepsilon \| d_k \|. \end{aligned} \tag{20}$$

is satisfied for all $\| d_k \| < \sigma$. We show by induction that (19) holds with

$$c = \min\{\Delta_0 M_0, \beta_1 \sigma M_0, \beta_1 \varepsilon, \tau(1-c_1)\beta_1 \varepsilon\}. \tag{21}$$

when $k = 0$, it is obvious that $\Delta_0 \geq c/M_0$, so (19) holds for $k = 0$. Assume that (19) is true for k, we will prove that (19) holds for $k+1$.

Since M_k is non-decreasing, to prove (19) for $k+1$, it suffices to establish the relation

$$\Delta_{k+1} \geq \frac{c}{M_k}. \tag{22}$$

Since $\Delta_{k+1} = R_{c_1}(\rho_k)\Delta_k$, it follow form Theorem 2 that when $\rho_k \geq c_1, \Delta_{k+1} \geq \Delta_k \geq c/M_k$. In this case, relation (22) is trivial. Therefore, in the remainder of proof, $\rho_k < c_1$, thus Δ_{k+1} is in the range $\beta_1 \Delta_k \leq \Delta_{k+1} \leq (1-\gamma_1)\Delta_k$.

If $\| d_k \| \geq \sigma$, then

$$\begin{aligned} \Delta_{k+1} & \geq \beta_1 \Delta_k \\ & \geq \beta_1 \| d_k \| \\ & \geq \beta_1 \sigma \\ & \geq \beta_1 \sigma M_0 / M_k \\ & \geq c/M_k. \end{aligned}$$

If $\| d_k \| < \sigma$, it follows from Algorithm 1 that

$$\begin{aligned} f(x_k + d_k) - R_k & > -c_1(\phi_k(0) - \phi_k(d_k)) \\ & = c_1(g_k^T d_k + d_k^T B_k d_k / 2). \end{aligned} \tag{23}$$

From (20) and Lemma 2.2. in [8], we know that

$$\begin{aligned} & f(x_k + d_k) - R_k \\ & \leq f(x_k + d_k) - f(x_k) \\ & = \int_0^1 d_k^T [\nabla f(x_k + \theta d_k) - \nabla f(x_k)] d\theta + \nabla f(x_k)^T d_k \\ & < g_k^T d_k + \frac{\tau \varepsilon (1-c_1)}{2} \| d_k \|. \end{aligned} \tag{24}$$

From (23) and (24), we have

$$(1-c_1)(g_k^T d_k + \tau\varepsilon \| d_k \| / 2) > c_1 d_k^T B_k d_k / 2. \tag{25}$$

Moreover, we have by (5) and (18) that

$$-g_k^T d_k - d_k^T B_k d_k / 2 \geq \tau \varepsilon \min\{\Delta_k, \varepsilon / \| B_k \|\}. \tag{26}$$

Multiplying (26) by $(1-c_1)$ and combining it with (25), we get

$$\begin{aligned} & \Delta_k^2 \| B_k \| \\ & \geq -d_k^T B_k d_k \\ & > 2(1-c_1)\tau\varepsilon \min\{\Delta_k, \varepsilon/\| B_k \|\} - (1-c_1)\tau\varepsilon \| d_k \| \\ & \geq (1-c_1)\tau\varepsilon \{2\min\{\Delta_k, \varepsilon/\| B_k \|\} - \Delta_k\} \\ & = (1-c_1)\tau\varepsilon \min\{\Delta_k, 2\varepsilon/\| B_k \| - \Delta_k\}. \end{aligned}$$

Using the above condition, we can give a constant lower bound on the product

$$\Delta_k \| B_k \| > \min\{(1-c_1)\tau\varepsilon, \varepsilon\} \geq c/\beta_1.$$

always holds. So

$$\Delta_{k+1} \geq \beta_1 \Delta_k > c/\| B_k \| > c/M_k.$$

Thus, (19) holds for $k+1$. Therefore, by induction, (19) holds for all k.

Theorem 8 Assume that Assumption holds. Let $\{x_k\}$ be the sequence generated by Algorithm 1, then

$$\liminf_{k \to \infty} \| g_k \| = 0. \tag{27}$$

Proof. We prove it by contradiction. Assume that (27) is not true, that is, there exists a constant $\varepsilon > 0$ such that $\| g_k \| \geq \varepsilon, \forall k$.

It follows from (6), (7), (11) and Lemma 6 that

$$\begin{aligned} f_{l(k)} &= f(x_{l(k)-1}+\alpha_{l(k)-1}d_{l(k)-1}) \\ &\le R_{l(k)-1}+\beta\alpha_{l(k)-1}g_{l(k)-1}^{T}d_{l(k)-1} \\ &\le R_{l(k)-1}-\tau\beta\tilde{\alpha}\varepsilon\min\{\Delta_{l(k)-1},\varepsilon/b\} \\ &\le f_{l(l(k)-1)}-\tau\beta\tilde{\alpha}\varepsilon\min\{\Delta_{l(k)-1},\varepsilon/b\}. \end{aligned} \quad (28)$$

Using Corollary 2.1 in [8] , from (28), we have

$$\lim_{k\to\infty}\Delta_{l(k)-1}=0. \quad (29)$$

For $k > M$, we have $k-N \le k-m(k) \le l(k) \le k$ and hence

$$0 \le k-l(k) \le N. \quad (30)$$

By the updating formula of Δ_k and Theorem 2, $\beta_1^j\Delta_k \le \Delta_{k+j} \le \beta_2^j\Delta_k$ holds for all j. Thus, it follows from (30) that

$$\beta_1^{M+1}\Delta_{l(k)-1} \le \Delta_k \le \beta_2^{M+1}\Delta_{l(k)-1}.$$

Then, from (29), we have $\lim_{k\to\infty}\Delta_k=0$. It contradicts (19) in Lemma 7.

4. CONCLUSIONS

In this paper, we proposed a new non-monotone self-adaptive trust region algorithm with non-monotone line search. After we analyzed the properties of the new algorithm, the global convergence theory is proved. In the near future, we would like to investigate some new type of combinations composed of different functions values in order to sufficiently use the information which the algorithm has already derived.

ACKNOWLEDGEMENTS

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

REFERENCES

[1]Hei L., (2003). *A self-adaptive trust region algorithm*. Journal of Computational Mathematics. 21, 229-236.

[2]Grippo L., Lampariello F., Lucidi S., (1986). *A nonmonotonic line search technique for Newton's method*. SIAM J.Numer.Anal. 23, 707-716.

[3]Fu J H, Sun W., (2005). *Non-monotone adaptive trust-region for unconstrained optimization problems*. Applied Mathematics and Computation. 163, 707-716.

[4]Sun W., (2004). *Non-monotonic trust region method for solving optimization problems*. Applied Mathematics and Computation. 156, 159-174.

[5]Yang Yang, Wenyu Sun., (2007). *A New Non-monotonic Self-adaptive Trust Region Algorithm with Line Search*. Chinese Journal of Engineering Mathrmatics. 24, 788-794.

[6] X. Wang, C.L. Wang., (2014). *A new nonmonotone trust-region method with nonmonotone line search(in Chinese)*. System Science and Mathematics. 34, 179-186.

[7]M. Ahookhosh, K. Amini., (2002). *An efficient nonmonotone trust-region method for unconstrained optimization*. Numer Algor. 59, 523-540.

[8]M. Ahookhosh, K. Amini and M.R. Peyghami., (2012). *A nonmonotone trust-region line search method for large-scale unconstrained optimization*. Applied Mathematical Modelling. 36, 478-487.

[9] S.M. Pang, L.P Chen., (2011). *A trust-region method with nonmonotone line search(in Chinese)*. Mathematica numerica sinca. 33, 48-56.

BIOGRAPHIES

Xiao Wu: Postgraduate student at College of Mathematics and Information Science, Hebei University, China. Her research interest mainly concerns non-monotone line search and trust region methods .

Qinghua Zhou: Professor in HeBei University. Major interest includes Optimization Theory and its Applications.

Changyuan Li: Postgraduate student at the College of Mathematics and Information Science, Hebei University, China. His research interest includes non-monotone trust region methods and derivative-free methods.

Xiaodian Sun: Research Scientist at University of Cincinnati College of Medicine. Her research interest includes dynamic modeling and inference, time series, statistical learning.