



Payame Noor University



Control and Optimization in Applied Mathematics (COAM)

Vol. 1, No. 1, Spring-Summer 2016(31-40), ©2016 Payame Noor University, Iran

On Efficiency of Non-Monotone Adaptive Trust Region and Scaled Trust Region Methods in Solving Nonlinear Systems of Equations

R. Hekmati*

PhD student of mathematics, University of Houston,
Houston, Texas, USA.

Received: May 30, 2013; Accepted: June 6, 2014.

Abstract. In this paper we run two important methods for solving some well-known problems and make a comparison on their performance and efficiency in solving nonlinear systems of equations. One of these methods is a non-monotone adaptive trust region strategy and another one is a scaled trust region approach. Each of methods showed fast convergence in special problems and slow convergence in other ones; we try to categorize these problems and find out that which method has better numerical behavior. The robustness of methods is demonstrated by numerical experiments.

Keywords. Non-monotone adaptive, Scaled trust region, Nonlinear systems of equations, Numerical comparison.

MSC. 49M37; 49MXX.

* Corresponding author

rhekmati@math.uh.edu; rasoul_hekmati@yahoo.com

<http://mathco.journals.pnu.ac.ir>

1 Introduction

We consider the following algebraic nonlinear system of equations

$$F(x) = 0, \quad x \in X \quad (1)$$

where $F : X \rightarrow R^n$ is a continuously differentiable mapping and X is a subset of R^n .

This problem arises in many fields in real-life cases like cross-sectional properties of structural elements, mechanical linkages, chemical equilibrium, nonlinear fitting, function approximation, etc. Globally convergent methods for problem (1) has been addressed in many papers like method is proposed by Kanzow ([7]) or gauss-newton method, etc. (See e.g. [3, 9, 10, 13, 14]). Here we are interested in trust region approach and choose STRN (scaled trust region newton method) NMAdapt (non-monotone adaptive trust region method) for solving problem (1).

STRN have designed for solving bound-constrained nonlinear equations ([2]):

Where

$$\Omega = \{x \in R^n : l_i \leq x_i \leq u_i, \forall i = 1, 2, \dots, n\}$$

and

$$F(x) = (F_1(x), \dots, F_n(x))^T.$$

By choosing $l = -\infty$, $u = +\infty$, the method will be able to solve problem (1) as robust method. At each step of STRN a quadratic function is minimized over an elliptical trust-region whose shapes depend on the bounds. By changing STRN's codes we eliminate the sensitivity of algorithm to box bounds and make STRN a standard trust region strategy. The changes are showed in next section. Convergence analysis of this method is performed and this globally convergent method has the ultimate rate of convergence to a solution quadratic ([2]). Numerical experiments show its efficiency. The adaptive non-monotone method that we use to compare with manipulated STRN is proposed by Hong-Wei (See [6]).

2 Algorithms

For the traditional trust region methods, at each iteration point x_k , the trial step obtains by solving the following trust-region sub-problem which Δ_k is the trust region radius:

$$\begin{cases} \min \phi_k(d) = \frac{1}{2} \|F_k + F'_k d\|^2 \\ s.t. \quad \|d\| \leq \Delta_k. \end{cases}$$

This method has been studied by many authors ([4, 11, 14]). Non-monotone trust region that we use in this paper is proposed by Hong-Wei (See [6]), it also has an adaptive strategy in order to make the method more efficient:

Let

$$\|F_{l(k)}\| = \max \{\|F_{k-j}\|\}, \quad k = 1, 2, \dots$$

where $0 \leq j \leq m(k)$, $m(k) = \min \{M, k\}$, $M \geq 0$ is an integer constant. Now define $\varphi_{l(k)} = \frac{1}{2} \|F_{l(k)}\|^2$.

The actual reduction

$$Ared_k = \varphi(x_k + d_k) - \varphi_{l(k)},$$

and the predicted reduction

$$Pred_k = \varphi_k(d_k) - \varphi(x_k),$$

the ratio of actual reduction over predict reduction is equal to:

$$\gamma_k = \frac{Ared_k}{Pred_k}.$$

Finally the trust region radius $\Delta_k = c^p \|g_k\|^\beta M_k$, $0 < c < 1$, $M_k = \|B_k^{-1}\|$ and p is a non-negative integer (See [6]).

3 Algorithm NMAadapt

Initial: choose constant $\rho, \tau, c \in (0, 1)$, $p = 0$, $\varepsilon > 0$, $M \geq 0$, $x_0 \in R^n$, $k = 0$

Step1: if $\|g_k\| < \varepsilon$, stop.

Step2: compute d_k by solving (5) and calculate $m(k)$, $F_{l(k)}$, $Pred_k$ and γ_k .

If $\gamma_k < \rho$, then $p := p + 1$, go to step2. Otherwise, go to step3.

Step3: $x_{k+1} = x_k + d_k$, generate B_{k+1} , set $p = 0$, $k := k + 1$, go to step 1.

For global convergence proofs, see [6].

The STRN method is proposed by Bellavia, etc. in [2] gives $x_k \in \text{int}(\Omega)$ which Ω is the box that restrict our movement along descend direction p_k . Now we should look along p_k for the next approximation x_{k+1} , let $\lambda(p_k)$ be the stepsize along p_k , That is

$$\lambda(p_k) = \begin{cases} \infty & \Omega = R^n \\ \min \varphi_i(p_k) & \Omega \subset R^n \end{cases}$$

and

$$\varphi_i(p_k) = \begin{cases} \max \left\{ \frac{l_i - (x_k)_i}{(p_k)_i}, \frac{u_i - (x_k)_i}{(p_k)_i} \right\} & (p_k)_i \neq 0 \\ \infty & (p_k)_i = 0 \end{cases}.$$

If $\lambda(p_k) > 1$ it is obvious that $x_{k+1} = x_k + p_k$ is within Ω and if $\lambda(p_k) \leq 1$ then a step-back strategy along p_k should be choosed ([2]).

Let $\theta \in (0, 1)$ and define

$$\xi(p_k) = \begin{cases} 1 & \lambda(p_k) > 1 \\ \max \{\theta, 1 - \|p_k\|\} \lambda(p_k) & \text{otherwise} \end{cases}$$

and

$$\begin{aligned} a(p_k) &= \xi(p_k) p_k, \\ x_{k+1} &= x_k + a(p_k), \end{aligned}$$

then the new iteration will be feasible.

For choosing the search direction p_k Newton method can be used

$$F'_k p_k^N = -F_k$$

and the quadratic model

$$m_k(p) = \frac{1}{2} \|F'_k p + F_k\|^2 = \frac{1}{2} \|F_k\|^2 + F_k^T F'_k p + \frac{1}{2} p^T F'_k{}^T F'_k p,$$

is trusted to be an adequate approximation of the function $f(x) = \frac{1}{2} \|F(x)\|^2$ for solving the sub-problem

$$\min_p \{m_k(p) : \|p\| \leq \Delta_k\},$$

where $\Delta_k > 0$ is trust region radius. If $\|p_k^N\| \leq \Delta_k$ the Newton step is the global minimum of $m_k(p)$ (See [2]).

When the nonlinear system is bound-constrained, the strategy should changes, because if the step direction points to a nearby constraint, a small fraction of p_k should be taken to stay within in Ω and this may preclude the convergence of the sequence. To prevent this occurrence the affine scaling mapping proposed by Coleman and Li should be taken ([1]).

Now consider the gradient $F'^T(x) F(x)$ of the objective function f and definite $v(x)$ by

$$\begin{aligned} v_i(x) &= x_i - u_i \quad \text{if } \left(F'^T(x) F(x)\right)_i < 0 \text{ and } u_i < \infty \\ v_i(x) &= x_i - l_i \quad \text{if } \left(F'^T(x) F(x)\right)_i \geq 0 \text{ and } l_i > -\infty \\ v_i(x) &= -1 \quad \text{if } \left(F'^T(x) F(x)\right)_i < 0 \text{ and } u_i = \infty \\ v_i(x) &= 1 \quad \text{if } \left(F'^T(x) F(x)\right)_i \geq 0 \text{ and } l_i = -\infty. \end{aligned}$$

Let $D(x)$ be the diagonal scaling matrix such that

$$D(x) = \text{diag} \left(|v_1(x)|^{-\frac{1}{2}}, |v_2(x)|^{-\frac{1}{2}}, \dots, |v_n(x)|^{-\frac{1}{2}} \right)$$

(See for more details [2]).

Now let $D(x) = D(x_k)$ and the elliptical trust-region constrain will be obtained

$$\|D_k p\| \leq \Delta_k.$$

So, we have this subproblem:

$$\min_p \{m_k(p) : \|D_k p\| \leq \Delta_k\}.$$

The Cauchy point is

$$d_k = -D_k^{-2} \nabla f(x_k) = -D_k^{-2} F_k'^T F_k ,$$

and by considering the trust region bound

$$p_k^c = \tau_k d_k = -\tau_k D_k^{-2} F_k'^T F_k ,$$

where

$$\tau_k = \underset{\tau > 0}{\operatorname{argmin}} \{m_k(\tau d_k) : \|\tau D_k d_k\| \leq \Delta_k\} .$$

For global convergence, $a(p_k)$ should gives a sufficient reduction in the quadratic model m_k , from [5] we have

$$\rho_k^c(p_k) = \frac{m_k(0) - m_k(a(p_k))}{m_k(0) - m_k(a(p_k^c))} \geq \beta_1,$$

for $\beta_1 \in (0, 1]$. Above condition does not guarantee a good agreement between m_k and the objective function f . Then we have this condition

$$\rho_k^f(p_k) = \frac{f(x_k) - f(x_k + a(p_k))}{m_k(0) - m_k(a(p_k))} \geq \beta_2,$$

for $\beta_2 \in (0, 1]$ (See [2]).

4 Scaled Trust-Region Newton Method's Algorithm

This algorithm is proposed by Bellavia et al. (See for more details [2]):

Let

$$x_0 \in \operatorname{int}(\Omega), \Delta_0 > 0, \theta \in (0, 1), 0 < a_1 \leq a_2 < 1, \beta_1 \in (0, 1], \\ 0 < \beta_2 < \beta_3 < 1, 0 < \delta_1 < 1 < \delta_2.$$

For $k=0, 1, 2, \dots$

1. Compute the matrix D_k .
2. Repeat
 - 2.1 find $p_k = \operatorname{argmin}_{\|D_k p\| \leq \Delta_k} m_k(p)$.
 - 2.2 compute the Cauchy point p_k^c .
 - 2.3 compute $a(p_k)$ and $a(p_k^c)$
 - 2.4 if $\rho_k^c(p_k) < \beta_1$ then set $p_k = p_k^c$.
 - 2.5 set $\Delta_k^* = \Delta_k$.
 - 2.6 set $\Delta_k = \delta_1 \Delta_k$.
3. Set $x_{k+1} = x_k + a(p_k)$, $\Delta_k = \Delta_k^*$.
4. If $\rho_k^f(p_k) \geq \beta_3$

$$\Delta_{k+1} = \delta_2 \Delta_k.$$

Else

$$\Delta_{k+1} = \Delta_k.$$

For more explains about algorithm see [2]. Convergence results of method had showed completely in [2] and we pass this section. In order to solve unconstrained system of equations, we change STRN's MATLAB codes by defining $\Omega = R^n$ and $l = -\infty$, $u = +\infty$ and the scaling matrix $D = I$.

5 Numerical Experiments

When an algorithm is presented in the optimization literature, it should be tested on a set of problems. The aim of this testing is to show that the algorithm works, and even it works better than other algorithms in the same realm. Testing an algorithm is a tedious and error-prone job because it requires the coding of functions. But it is necessary for finding that it is not tuned for special problems.

We choose some problems from [12] that had been used for test non-monotone adaptive trust region method in [6]. The stopping criterion used is $\|g_k\| < \varepsilon$, where

$$\varepsilon = 10^{-8}.$$

Note that B_k are obtained by a BFGS approach and $\alpha = 0.1, c = 0.5, \beta = 0.6$. for STRN method we set $\Delta_0 = 1, \beta_0 = 0.1, \beta_2 = 0.25, \beta_3 = 0.75$ the trust region size is updating using this rules:

$$\begin{aligned}\Delta_k &= \min \{0.25\Delta_k, 0.5\|D_k p_k\|\} \text{ in step 6} \\ \Delta_{k+1} &= \max \{\Delta_k, 2\|D_k p_k\|\} \text{ in step 4.}\end{aligned}$$

The detailed results are summarized in Table 1. The columns of table have this meaning:

Prob: the name of the problem.

Dim: the dimension of the problem.

NMAadapt: the non-monotone adaptive trust region method.

STRN: scaled trust region newton method.

It, Fe: number of iterations, function evaluations respectively.

$\frac{1}{2}\|F(x)\|^2$: final amount of least-square function, suitable for knowing the accuracy of solution.

0.ab-c: means $0.ab \times 10^{-c}$ or $a.b \times 10^{-c-1}$.

In Table 1 test functions have complicated features, for example the Rosenbrock function is useful for test the ability of the algorithm to follow curved valleys. For some functions, STRN reports an error because no improvement for the nonlinear residual could be obtained, that means:

$$\left| \|F(x_k)\| - \|F(x_{k-1})\| \right| \leq 100\varepsilon \|F(x_k)\|.$$

Or the matrix was singular to working precision.

Table 1: Numerical results for some test problems

Experiments		NMAadapt			STRN		
Problem	Dim	It	Fe	$\frac{1}{2}\ F(x)\ ^2$	It	Fe	$\frac{1}{2}\ F(x)\ ^2$
<i>Rosenbrock</i>	2	15	28	0.22 – 17	12	15	0.32 – 10
<i>Powell singular</i>	4	10	17	0.65 – 7	14	15	0.45 – 7
<i>Powell badly scaled</i>	2	163	213	0.71 – 8	11	12	0.76 – 8
<i>wood</i>	4	31	59	0.47 – 12	42	49	0.24 – 7
<i>Helical valley</i>	3	12	21	0.44 – 14	8	10	0.75 – 7
<i>Watson</i>	3	62	117	0.26 – 7	60	110	0.23 – 7
<i>Brown almost linear</i>	30	24	36	0.15 – 13	<i>Error</i>		
<i>Discrete boundary</i>	10	16	29	0.63 – 10	3	4	0.47 – 14
<i>Discrete integral</i>	10	5	8	0.98 – 16	4	5	0.16 – 13
<i>Trigonometric</i>	30	85	142	0.93 – 5	<i>Error</i>		
<i>Variably dimensioned</i>	10	98	166	0.26 – 9	<i>Error</i>		
<i>Broyden tridiagonal</i>	10	73	125	0.68 – 16	4	5	0.10 – 8
<i>Broyden banded</i>	30	14	26	0.11 – 13	<i>Error</i>		

Table 1 gives us some useful information, but before analysis this results let us bring Figure 1 in order to have better presentation. In this figure we just bring results for problems that both methods were able to solve.

The advantage of STRN in low dimension problems is clear from Figure 1. We have run STRN algorithm before for a vast collection of problems come from the library NLE accessible through the web site:

WWW.polymath-software.com/library.

It contains over 70 real-life problems whose dimensions vary from a single equation to 14 equations, we choose some important problems and STRN was completely successful for solving that problems, but according to Table 1 it fails to solve high dimensioned problems (Brown almost linear, Trigonometric, Variably dimensioned, Broyden banded) while NMAadapt has not any fault. It seems that STRN are strong and robust when we use it for solve low dimensioned nonlinear system of equations and unable for solving high dimensioned problems (at least in our collection of problems chosen from [12]).

Now consider the number of problems as x and the number of iterations as $f(x)$, so we have a function and we can assume its approximation in order to predict behavior of method in other similar problems. By using MATLAB interpolation from 6th degree we have:

Linear model Poly6 for STRN

$$f(x) = p1x^6 + p2x^5 + p3x^4 + p4x^3 + p5x^2 + p6x + p7$$

Coefficients (with 95% confidence bounds):

$$p1 = 0.0007407, p2 = 0.04444, p3 = -1.187$$

$$p4 = 9.323, p5 = -29.35, p6 = 40.3, p7 = -7.111$$

and Linear model Poly6 for NMAadapt:

$$f(x) = p1x^6 + p2x^5 + p3x^4 + p4x^3 + p5x^2 + p6x + p7$$

Coefficients (with 95% confidence bounds):

$$p1 = 0.36, p2 = -10.81, p3 = 126.5$$

$$p4 = -726.1, p5 = 2098, p6 = -2782, p7 = 1306.$$

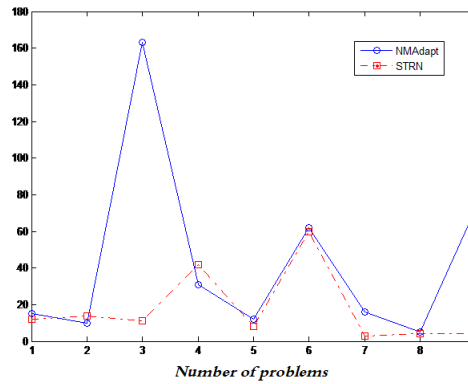


Figure 1: Performance of STRN and NMAadapt

In Figure 2 we can see these polynomials. Obviously STRN has more smooth, steady and predictable manner, while NMAadapt has variation in a wide range and huge amplitude.

6 Conclusions

We run two methods for solving some famous unconstrained problems and find that STRN is more efficient in low dimensioned problems and NMAadapt is better for high dimensioned problems, we also tried to solve high dimension problems (BROYDEN3D and ARGTRIG with $n = 1000$) and STRN was unsuccessful as expected. But it needs more scrutiny and more experiments to claim this certainly.

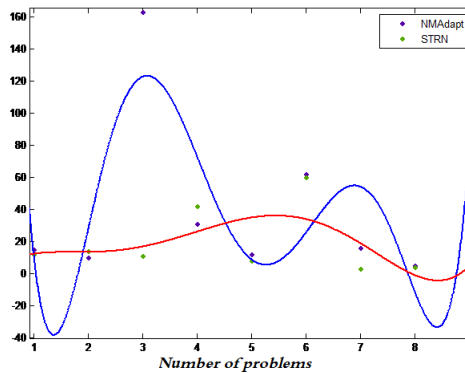


Figure 2: The 6th degree approximation of methods

References

- [1] Bellavia S., Macconi M., Morini, B. (2004). “ STRSCNE: A scaled trust region solver for constrained nonlinear equations ”, Computational Optimization and Applications, 28, 31-50.
- [2] Bellavia S., Macconi M., Morini B. (2013). “ An affine scaling trust-region method approach to bound-constrained nonlinear systems ”, Applied Numerical Mathematics, 44, 257-280.
- [3] Fletcher R. (1987). “ Practical methods of optimizations ”, New York: John Wiley & Sons.
- [4] Fletcher R. (1980). “ Practical method of optimization, unconstrained Optimization ”, Vol. 1, New York: John Wiley & Sons.
- [5] Floudas C. A., Pardalos P. M., Adjiman C., Esposito W. R., Gümüs Z. H., Harding S. T., Klepeis J. L., Meyer C. A., Schweiger C. A. (1991). “ Handbook of test problems in local and global optimization, nonconvex optimization and its applications ”, Vol. 33, Kluwer Academic, Dordrecht.
- [6] Hong-Wei L. (2011). “ A non-monotone adaptive trust region algorithm for nonlinear equations ”, The tenth International Symposium on Operation Research and its Applications (ISORA), Dunhuang, China, 28-31.
- [7] Kanzow C. (2001). “ An active set-type newton method for constrained nonlinear systems”, Complementarity: Applications, Algorithms and Extensions Applied Optimization, 50, 179-200.
- [8] Ferris O. L., Mangasarian J. S., Pang J. S., “ Complementarity: Applications, algorithms and extensions ”, Kluwer Academic, Dordrecht, Springer.
- [9] Levenberge K. (1944). “A method for the solution of certain nonlinear problem in least squares ”, Quarterly Journal of Applied Mathematics, 2, 2, 164-166.

- [10] Marquardt D. W. (1963). “ An algorithm for least-squares estimation of nonlinear inequalities ”, SIAM Journal on Applied Mathematics, 11, 2, 431-441.
- [11] More J. J. (1983). “ Recent developments in algorithms and software for trust region methods, mathematical programming ”, The State of the Art, (Edited by A. Bachem, M. Gortchel and B. Korte) Springer-Verlag, Berlin, 258-287.
- [12] More J. J., Garbow B. S., Hillstom K. E. (1981). “ Testing unconstrained optimization software ”, ACM Transaction on Mathematical Software, 7, 17-41.
- [13] Nocedal J., Wright S. J. (1999). “ Numerical optimization ”, Springer Series in Operation Research, Springer, New York,
- [14] Yuan Y. (1996). “ On the convergence of the trust region algorithms ”, Mathematica Numerica Sinica, 16333-346.
- [15] Yuan Y., Sun W. (1997). “ Optimization theory and algorithm ”, Scientific Publishing House, China.

Archive of SID

درباره عملکرد روش‌های ناحیه اعتماد غیریکنوای سازگار و ناحیه اعتماد مقیاس‌بندی شده برای حل دستگاه معادلات غیرخطی

حکمتی ر.

دانشجوی دکتری ریاضی کاربردی

آمریکا، تگزاس، هاستون، دانشگاه هاستون

rehkmati@math.uh.edu; rasoul_hekmati@yahoo.com

چکیده

در این مقاله دو روش مهم برای حل تعدادی مسئله مشهور مورد استفاده قرار گرفته است و عملکرد این دو روش در حل دستگاه معادلات غیرخطی مورد مقایسه قرار گرفته است. از جمله این روش‌ها، روش ناحیه اعتماد غیریکنوای سازگار و دیگری رویکردی مقیاس‌بندی شده است. هر یک از این روش‌ها همگرایی سریع در مسائل خاص و همگرایی کند در مسائل کلی را نشان داده است. در پایان سعی شده است تا این مسائل رده‌بندی شده و روش برتر از نظر رفتار عددی معین شوند. قدرت روش‌ها به صورت آزمون‌های عددی نشان داده شده است.

کلمات کلیدی

ناحیه اعتماد غیریکنوای سازگار، ناحیه اعتماد مقیاس‌بندی شده، دستگاه معادلات غیرخطی، مقایسه عددی.