Trust Region Algorithms for Nonlinear Equations *

Ya-xiang Yuan
State Key Laboratory of Scientific and Engineering Computing
ICMSEC, Chinese Academy of Sciences, Beijing 10080, China

Abstract

In this paper, we consider the problem of solving nonlinear equations $F(x) = 0$, where $F(x)$ from $\Re^n$ to $\Re^m$ is continuously differentiable. We study a class of general trust region algorithms for solving nonlinear equation by minimizing a given norm $||F(x)||$. The trust region algorithm for nonlinear equations can be viewed as an extension of the Levenberg-Marquardt algorithm for nonlinear least squares. Global convergence of trust region algorithms for nonlinear equations are studied and local convergence analyses are also given.

Key words: nonlinear equation, trust region, convergence.

1. Introduction

We consider the problem of solving nonlinear equations:

$$f_i(x) = 0, \quad i = 1, ..., m \tag{1.1}$$

where $f_i(x)$ are nonlinear functions defined in $\Re^n$. The system is called an overdetermined system if $m > n$, an underdetermined system if $m < n$. Even if $m = n$, due to the nonlinearity of $f_i(x)$, system (1.1) may have no solutions. Hence, it is usual to minimize the residual:

$$\min_{x \in \Re^n} ||F(x)||, \tag{1.2}$$

where $F(x) = (f_1(x), \ldots, f_m(x))^T$ is a vector function from $\Re^n$ to $\Re^m$ and $||.||$ is a norm in $\Re^m$.

When $n = m$, the classical Levenberg-Marquardt method (Levenberg (1944), Marquardt (1963)) for nonlinear equations computes trial steps by

$$d_k = (J(x_k)^T J(x_k) + \sigma_k I)^{-1} J(x_k)^T F(x_k) \tag{1.3}$$

*presented at Conference on Scientific Computation, Hong Kong, 17-19 March, 1994, this work was done when the author was visiting Hong Kong Baptist College, supported by Croucher Foundation

where $J(x_k) = \nabla F(x_k)$ is the Jacobi, and $\sigma_k \geq 0$ is a parameter being updated from iteration to iteration. Levenberg-Marquardt step (1.3) is a modification of the Newton's step

$$d_k^N = -J(x_k)^{-1}F(x_k). \tag{1.4}$$

The parameter σ_k can be viewed as a safeguard to prevent d_k to be too large when $J(x_k)$ is nearly singular. Furthermore, when $J(x_k)$ is singular, the Newton's step is undefined. A positive σ_k guarantee that (1.3) is well defined.

The original idea of Levenberg-Marquardt method is to modify the Newton's step in order to overcome the difficulties caused by possible singularity or near singularity of the Jacobi. Let

$$\rho_k = ||(J(x_k)^T J(x_k) + \sigma_k I)^{-1} J(x_k)^T F(x)||_2, \tag{1.5}$$

it is easy to see that the Levenberg-Marquardt step (1.3) is the unique solution of the following subproblem

$$\min \quad ||F(x_k) + J(x_k)d||_2^2 \tag{1.6}$$

$$\text{subject to} \quad ||d||_2^2 \leq \rho_k^2. \tag{1.7}$$

Because of this, we can regard the Levenberg-Marquardt method as a trust region algorithm. The difference is that, the classical Levenberg-Marquadt method choose a suitable σ_k at each step, while trust region algorithms update Δ_k. It is that directly controlling Δ_k is preferable to adjusting σ_k (see Moré (1978)).

A trust region algorithm requires the next iterate point x_{k+1} is chosen from the current trust region. Normally the trust region is a ball centered at the current iterate. Thus, the trust region trial step s_k satisfies the "trust region" condition:

$$||d|| \leq \Delta_k, \tag{1.8}$$

where $\Delta_k > 0$ is the trust region radius which is updated each iteration.

In the next section, we give a general trust region algorithm that is based on minimizing a penalty function, and briefly discuss some special algorithms that belong our framework. In Section 3, global convergence results of the algorithm is proved. The first convergence result is that at least one accumulation point is a stationary point, which is a direct consequence of the convergence results for nonsmooth optimization given by Yuan (1985). The second result is that any accumulation point is a stationary point if a trial step is accepted only when the actual reduction of the penalty function is at least a fraction of the predicted reduction (the reduction in the approximation model). In Section 4, local convergence analyses are made. It is shown that if the algorithm converges to a first order strict minimum then the convergence rate is quadratic. Without the assumption that the limit point is a first order strict minimum, the only known local analyses results are made by Powell and Yuan (1984) which is only for the case that not all equations are satisfied at the limit point.

2. Trust Region Algorithms

Trust region algorithms for nonlinear equations are based on minimizing a certain penalty function, such a penalty function attains its minimum at the solutions of the nonlinear equations.

Let $h(F)$ be a function defined in $\Re^m$. If $h(F)$ satisfies that $h(0) = 0$ and that

$$h(F) > 0, \qquad \forall F \neq 0, \quad F \in \Re^m, \tag{2.1}$$

then $h(.)$ is a penalty for the nonlinear equations (1.1). The simplest penalty functions are norm penalty functions, namely $h(.) = ||.||$. In the following, we give a trust region algorithm that is based on the minimization of $h(F(x))$.

At the beginning of each iteration, a trial step s_k is computed by solving the subproblem:

$$\min_{d \in \Re^n} \quad h(F_k + J_k d) + \frac{1}{2} d^T B_k d \equiv \phi_k(d) \tag{2.2}$$

$$\text{s. t.} \qquad ||d|| \leq \Delta_k, \tag{2.3}$$

where $B_k \in \Re^{n \times n}$, $||.||$ is a given norm in $\Re^n$, and $\Delta_k > 0$ is the current trust region bound. The first part of $\phi_k(d)$, $h(F_k + J_k d)$ is the first order approximation of the penalty function $h(F(x_k + d))$. The second part of $\phi_k(d)$ is a second order approximation term.

The difference between $\phi_k(0)$ and $\phi_k(s_k)$ is the reduction of the approximation function $\phi_k(d)$ along the trial step s_k, which can be served as a prediction of the reduction of the penalty function. This predicted reduction is denoted by $Pred_k$, namely,

$$Pred_k = \phi_k(0) - \phi_k(s_k). \tag{2.4}$$

The actual reduction of the penalty function is

$$Ared_k = h(F(x_k)) - h(F(x_k + s_k)) \quad . \tag{2.5}$$

The ratio between these two reductions

$$r_k = \frac{Ared_k}{Pred_k} \tag{2.6}$$

plays a key role in trust region algorithms. The next iterate x_{k+1} is chosen by the following formula:

$$x_{k+1} = \begin{cases} x_k + s_k & \text{if } r_k \geq c_0 \\ x_k & \text{otherwise} \end{cases}, \tag{2.7}$$

where $c_0 \in [0, 1)$ is a constant. The trust region bound for the next iteration, Δ_{k+1} also depends on the value of r_k:

$$\Delta_{k+1} \in \begin{cases} [c_3||s_k||, \quad c_4\Delta_k] & \text{if } r_k < c_2 \\ [\Delta_k, \quad c_1\Delta_k] & \text{otherwise} \end{cases}, \tag{2.8}$$

where $c_i (i = 1, 2, 3, 4)$ are positive constants that satisfy that $c_1 > 1 > c_4 > c_3$ and $c_2 \in [c_0, 1)$. The choices of the constants $c_i (i = 0, 1, ..., 4)$ are not crucial to the algorithm, for example we can let $c_0 = 0$, $c_1 = 2$, $c_2 = 0.25$, $c_3 = 0.25$, $c_4 = 0.5$. See also Moré (1983), Fletcher (1987). However, the author thinks that there is a fundamental difference between algorithms that use $c_0 = 0$ and those with $c_0 > 0$, even though almost all algorithms that use a positive c_0 chooses a very small c_0, say 0.0001. Usually, we can show weakly global convergence for algorithms with $c_0 = 0$ and strongly global convergence for those with $c_0 > 0$. Here weekly global convergence means that at least one accumulation is a stationary point, while strongly global convergence means that all accumulation points are stationary point. We prefer the choice $c_0 = 0$, because weakly global convergence is also sufficient for finding an approximation solution, and because $c_0 > 0$ implies that a point at which the penalty function has a lower value may be rejected.

A general trust region algorithm for nonlinear equations based on penalty function $h(.)$ can be stated as follows:

Algorithm 2.1

Step 1 *Given* $x_1 \in \Re^n$, $\Delta_1 > 0$, $\epsilon \geq 0$, $B_1 \in \Re^{n\times n}$ *symmetric;*
$0 < c_3 < c_4 < 1 < c_1$, $0 \leq c_0 \leq c_2 < 1$, $c_2 > 0$, $k := 1$.

Step 2 *If* $h(F(x_k)) \leq \epsilon$ *then stop;*
Solve (2.2)-(2.3) giving s_k.

Step 3 *Compute* r_k;
Set x_{k+1} *by (2.7);*
Choose Δ_{k+1} *satisfying (2.8).*

Step 4 *Update* B_{k+1};
$k := k + 1$; *go to Step 2.*

The algorithm stated above is very general as there are many choices of $h(.)$. For example, Fletcher (1982) and Yuan (1983) assume that $h(.)$ is a convex function, El Hallabi and Tapia (1993) let $h(.)$ be any arbitrary norm $||.||$. The most commonly used $h(.)$ are the three norms $||.||_2$, $||.||_1$ and $||.||_\infty$. We briefly discuss them as follows:

2.1. 2-norm Trust Region Algorithms

Trust region algorithms that based on the 2-norm L_2 for solving nonlinear equations are studied by Moré (1978), Powell (1970). The trial step s_k is computed by solving the subproblem

$$\min ||F_k + J_k d||_2 \tag{2.9}$$

$$\text{s. t.} \qquad ||d||_2 \leq \Delta_k, \tag{2.10}$$

where $F_k = F(x_k)$ and $J_k = J(x_k)$. If s_k is an solution of (2.9)-(2.10), there exist $\sigma_k \geq 0$ such that

$$(J_k^T J_k + \sigma_k I)s_k = -J_k^T F_k \tag{2.11}$$

$$\sigma_k(\Delta_k - ||s_k||_2) = 0. \tag{2.12}$$

First we have the following result:

Lemma 2.2 *Let s_k be a solution of (2.9)-(2.10), then there is a unique $\sigma_k \geq 0$ that satisfy (2.11)-(2.12).*

If $||J_k^+ F_k||_2 \leq \Delta_k$, we can let

$$s_k = -J(x_k)^+ F(x_k). \tag{2.13}$$

Otherwise, $\sigma_k > 0$ and satisfies

$$||(J_k^T J_k + \sigma_k I)^{-1} J_k^T F_k||_2 = \Delta_k. \tag{2.14}$$

Thus, similar to techniques for subproblems of trust region algorithms for unconstrained optimization (see, Gay (1981) and Moré and Sorenson (1983)), we can apply Newton's method to the nonlinear equation:

$$\psi(\sigma) = \frac{1}{||(J_k^T J_k + \sigma I)^{-1} J_k^T F_k||_2} - \frac{1}{\Delta_k} = 0. \tag{2.15}$$

The reason for considering (2.15) instead of the simpler equation

$$||(J_k^T J_k + \sigma I)^{-1} J_k^T F_k||_2 - \Delta_k = 0, \tag{2.16}$$

is that the function $\psi(\sigma)$ is very close to a linear function. Thus Newton's method would give a faster convergence. Applying Newton's method to (2.15), we can computing σ_k by the following iterative scheme:

$$\sigma_k^0 = 0, \tag{2.17}$$

$$\sigma_k^{i+1} = \sigma_k^i - \frac{(s_k^i)^T (J_k^T J_k + \sigma_k^i I)^{-1} s_k^i}{||s_k^i||_2^3} \left(\frac{1}{||s_k^i||_2} - \frac{1}{\Delta_k} \right), \tag{2.18}$$

where $s_k^i = -(J_k^T J_k + \sigma_k^i I)^{-1} J_k^T F_k$. Due to the concavity of $\psi(\sigma)$, it can be seen that the sequence σ_k^i converges to σ_k quadratically. From the concavity of $\psi(\sigma)$, we have that

$$\begin{aligned} \psi(\sigma_k^{i+1}) &\geq \psi(\sigma_k^i) + \psi'(\sigma_k^{i+1})(\sigma_k^{i+1} - \sigma_k^i) \\ &= \psi(\sigma_k^i) \left(1 - \frac{\psi'(\sigma_k^{i+1})}{\psi'(\sigma_k^i)} \right), \end{aligned} \tag{2.19}$$

which implies that either

$$\psi(\sigma_k^{i+1}) \geq \frac{1}{2} \psi(\sigma_k^i), \tag{2.20}$$

or

$$\psi'(\sigma_k^{i+1}) \leq \frac{1}{2} \psi'(\sigma_k^i) \tag{2.21}$$

holds. The above inequalities show that for initial point $\sigma_k^0 = 0$, Newton's iteration (2.18) is a polynomial time algorithm.

An important property of the trial step s_k is the sufficient descent property:

Lemma 2.3 *Let s_k be a solution of subproblem (2.9)-(2.10), then*

$$||F_k||_2 - ||F_k + J_k s_k||_2 \geq \min\left[1, \frac{\Delta_k}{||J_k^+ F_k||_2}\right]\left(||F_k||_2 - ||(I - J_k J_k^+)F_k||_2\right). \tag{2.22}$$

Proof Define $s_k^* = -J_k^+ F_k$. If $||s_k^*||_2 \leq \Delta_k$, then $||F_k + J_k s_k||_2 = ||F_k + J_k s_k^*||_2$, which gives (2.22).

Now Assume $||s_k^*||_2 > \Delta_k$, we have that

$$\begin{aligned} ||F_k||_2 - ||F_k + J_k s_k||_2 &\geq ||F_k||_2 - ||F_k + \frac{\Delta_k}{||s_k^*||_2} J_k s_k^*||_2 \\ &\geq \frac{\Delta_k}{||s_k^*||_2}\left(||F_k||_2 - ||F_k + J_k s_k^*||_2\right) \end{aligned} \tag{2.23}$$

which gives (2.22). □

2.2. Duff-Nocedal-Reid Algorithm

For the case that $n = m$, Duff, Nocedal and Reid (1987) gave a trust region algorithm based on the minimization of the L_1 norm. Their subproblem for computing the trial step is as follows:

$$\min ||F_k + J_k d||_1 \tag{2.24}$$

$$\text{s. t.} \quad ||d||_\infty \leq \Delta_k. \tag{2.25}$$

This subproblem can be expressed as

$$\min \quad \sum_{i=1}^{n} [p_i + q_i] \tag{2.26}$$

$$\text{s. t.} \quad J_k d + p - q = -F_k \tag{2.27}$$

$$-\Delta_k e \leq d \leq \Delta_k e \tag{2.28}$$

$$p \geq 0, \quad q \geq 0, \tag{2.29}$$

where $e = (1, 1, ..., 1)^T$. The Jacobi of the linear constraints (except the bound constraints) of the above linear programming is as follows

$$\begin{pmatrix} J_k^T \\ I \\ -I \end{pmatrix}. \tag{2.30}$$

Therefore, when the simplex method is used for solving (2.26)-(2.29), the coefficient matrix for the basis corresponding to a simplex consists of columns of J_k, I and $-I$. This matrix is normally sparser than J_k. Duff, Nocedal and Reid (1987) use the subroutine LA05 of Harwell library to update efficiently the LU decomposition of the linear programming bases.

Fletcher (1987) use the L_1 penalty

$$P(x) = \mu f_0(x) + ||F(x)||_1 \tag{2.31}$$

to solve the constrained optimization problem:

$$\min \quad f_0(x) \tag{2.32}$$

$$\text{s. t.} \quad F(x) = 0. \tag{2.33}$$

Fletcher's approach, called Sl_1QP method, is to use the approximation function:

$$\bar{\phi}_k(d) = \mu d^T \nabla f_0(x_k) + ||F_k + J_k d||_1 + \frac{1}{2} d^T B_k d \tag{2.34}$$

It can be seen that $\bar{\phi}_k(d) = \phi_k(d)$ if $f_0(x) \equiv 0$. In this sense, Fletcher's Sl_1QP method is a L_1 trust region algorithm when it is applied to solve nonlinear equations.

2.3. Minimax Algorithms

Madsen (1975) give a minimax algorithm for overdetermined systems of nonlinear equations. His algorithm is a two stage algorithm. In the first stage, linear subproblems are used to compute trial step. In the second stage, quadratical models are used in order to obtain a fast convergence rate. We consider the case that only linear models are used. The subproblem is as follows:

$$\min \quad ||F_k + J_k d||_\infty \tag{2.35}$$

$$\text{s. t.} \quad ||d||_\infty \le \Delta_k \tag{2.36}$$

The above subproblem can be rewritten as the following linear programming problem:

$$\min \quad \mu \tag{2.37}$$

$$\begin{aligned} \text{s. t.} \quad & J_k d + p = -F_k, \\ & (e \quad I) \begin{pmatrix} \mu \\ p \end{pmatrix} \ge 0, \\ & (e \quad -I) \begin{pmatrix} \mu \\ p \end{pmatrix} \ge 0, \\ & -\Delta_k e \le d \le \Delta_k e \quad . \end{aligned} \tag{2.38}$$

Recently, Yuan (1992) gives an trust region algorithm for general constrained optimization problems. For equality constrained problem (2.32)-(2.33), Yuan's algorithm computes trial steps s_k by solving

$$\min d^T \nabla f_0(x_k) + \sigma_k ||F_k + J_k d||_\infty + \frac{1}{2} d^T B_k d \tag{2.39}$$

$$\text{s. t.} \qquad ||d||_\infty \le \Delta_k, \tag{2.40}$$

where $\sigma_k > 0$ is a parameter updated each iteration. When we apply Yuan's algorithm to solve nonlinear equations (namely, $f_0(x) \equiv 0$), subproblem (2.39)-(2.40) is exactly the same as (2.2)-(2.3) if we let $h(F) = ||F||_\infty$.

3. Global Convergence

We assume that $h(.)$ is a convex function. We call a point as a stationary point if the linear approximation of the penalty function can not be reduced:

Definition 3.1 *x^* is called a stationary point if*

$$\min_{d\in\Re^n} \quad h(F(x^*) + J(x^*)d) = h(F(x^*)). \tag{3.1}$$

Denote

$$\xi(x;d) = h(F(x)) - h(F(x) + J(x)d) \tag{3.2}$$

$$\eta_\rho(x) = \max_{||d||\le\rho} \xi(x;d). \tag{3.3}$$

It is easy to see that x^* is a stationary point if and only if

$$\eta_1(x^*) = 0. \tag{3.4}$$

Due to the convexity of $h(.)$, we have the following result

Lemma 3.2 *Let s_k be a solution of (2.2)-(2.3), then inequality*

$$\begin{aligned} Pred_k &= \phi_k(0) - \phi_k(s_k) \\ &\ge \frac{1}{2}\eta_{\Delta_k}(x_k) \min\left[1, \frac{\eta_{\Delta_k}(x_k)}{||B_k||_2\Delta_k^2}\right] \end{aligned} \tag{3.5}$$

holds.

Using the above lemma, it is easy to establish the following global convergence result:

Theorem 3.3 *If $\epsilon = 0$ in Algorithm 2.1, and if there exists a positive constant c_5 such that*

$$||B_k||_2 \le c_5 k \quad , \tag{3.6}$$

and if Δ_k is bounded above, then the sequence $\{x_k\}$ generated by the Algorithm 2.1 is not bounded away from stationary points.

The techniques for proving Lemma 3.2 and Theorem 3.3 are basically the same as those of Powell (1970, 1975). For detailed proofs, please see Yuan (1985). The above global convergence result allows $c_0 = 0$, which implies that any trial step s_k that reduces the penalty function will be accepted. If $c_0 > 0$, then s_k is acceptable only when the actual reduction is at least certain fraction of the predicted reduction. This slightly stronger condition for accepting trial steps imply a stronger global convergence result:

Theorem 3.4 *Under the conditions of Theorem 3.3, if $c_0 > 0$, and if $\{x_k\}$ and $\{||B_k||\}$ are uniformly bounded, then any accumulation points of $\{x_k\}$ are stationary points.*

Proof Let S be the set of all stationary points. If the theorem is not true, there exist a positive number τ and infinite many k such that

$$\text{dist}(x_k, S) \geq \tau, \tag{3.7}$$

where $\text{dist}(x, Y) = \min_{y\in Y} ||x - y||$. Let K be the set of all indices that

$$\text{dist}(x_k, S) \geq \tau/2. \tag{3.8}$$

By the definition of S, there exists a positive constant δ such that

$$\eta_1(x_k) \geq \delta \tag{3.9}$$

holds for all $k \in K$. From the boundedness of Δ_k, there exists a positive constant $\bar{\delta}$ such that

$$\eta_{\Delta_k} \geq \bar{\delta}\Delta_k \tag{3.10}$$

holds for all $k \in K$. Therefore, we have that

$$Pred_k \geq \hat{\delta}\Delta_k \tag{3.11}$$

for all large $k \in K$. The above inequality implies that

$$\lim_{k\in K, k\to\infty} r_k = 1. \tag{3.12}$$

Thus, for all sufficiently large $k \in K$,

$$h(F(x_{k+1})) \leq h(F(x_k)) + c_0\hat{\delta}\Delta_k. \tag{3.13}$$

Therefore, we have that

$$\sum_{k\in K} \Delta_k < \infty. \tag{3.14}$$

The above relation shows that there exists a $\bar{k} \in K$ such that

$$\sum_{k\in K, k\geq \bar{k}} \Delta_k < \tau/4. \tag{3.15}$$

Now there are infinitely many k in K such that (3.7) holds, which implies that there exists a $\hat{k} > \bar{k}$ and $\text{dist}(x_{\hat{k}}, S) \geq \tau$. Therefore, by induction it follows from (3.15) that $\text{dist}(x_k, S) \geq \tau/2$ holds for all $k \geq \hat{k}$. Thus, $k \in K$ for all large k, which shows that the sequence x_k converges to a non-stationary point. This contradicts to the previous theorem. Therefore the theorem is true. □

The above convergence results can be applied to most know trust region algorithms for nonlinear equations. An exception is the L_1 trust region algorithm of Duff, Nocedal and

Reid (1987). Their algorithm requires a very strong condition for accepting trial steps. The condition is

$$||F(x_k + s_k)||_1 \leq ||F(x_k)||_1 - \beta||J(x_k)s_k||_1, \tag{3.16}$$

for some $\beta > 0$. It is shown by Yuan (1994) that the above condition may make the algorithm stucked at a non-stationary point. Yuan's example is a system of 2 linear equations of 2 variables:

$$f_1(u,v) = \alpha - \frac{1}{\delta}u = 0 \tag{3.17}$$

$$f_2(u,v) = \alpha + (1+\frac{1}{\delta})u - v = 0. \tag{3.18}$$

For any give $\beta > 0$, one can choose $\alpha > 0$ and $\delta > 0$ properly such that condition (3.16) will not be satisfied at $(0,0)^T$ for any small Δ. More details can be found in Yuan (1994). Nocedal (1994) informed us that a similar example was also given by Powell in 1987. Of course, if we replace condition (3.16) by

$$||F(x_k + s_k)||_1 \leq ||F(x_k)||_1 - \beta[||F(x_k)||_1 - ||F(x_k) + J(x_k)s_k||_1]. \tag{3.19}$$

then convergent results of this section can be applied to the algorithm of Duff, Nocedal and Reid (1987), because (3.19) is the same as $r_k \geq \beta$.

4. Local Convergence

In this section, we study the local properties of Algorithm 2.1. Thus, throughout this section we assume that

Assumption 4.1 *$f_i(x)(i = 1, ..., m)$ are all twice continuously differentiable; x_k generated by Algorithm 2.1 converges to x^*; $\{||B_k||\}$ is bounded.*

We call x^* is a first order strict minimum of $h(F)$ if there exists a positive number δ such that

$$h(F(x^* + d)) \geq h(F(x^*)) + \delta||d||_2, \tag{4.1}$$

for all small d.

Theorem 4.2 *Under the conditions of Assumption 4.1, if $\Delta_{k+1} \leq \Delta_k$ when $||s_k|| < \Delta_k$, then x_k converges to x^* quadratically.*

Proof It is easy to see that

$$\begin{aligned} h(F(x)) - h(F(x) + J(x)(x^* - x)) &= h(F(x)) - h(F(x^*)) + O(||x - x^*||_2^2) \\ &\geq \delta||x - x^*||_2 + O(||x - x^*||_2^2) \end{aligned} \tag{4.2}$$

So we have

$$\eta_{||x-x^*||}(x) \geq \delta||x - x^*|| + O(||x - x^*||_2^2) \tag{4.3}$$

which shows that there exists $\bar{\delta}$ such that

$$\eta_{\Delta_k}(x_k) \geq \bar{\delta} \min[\Delta_k, ||x_k - x^*||_2] \tag{4.4}$$

for all large k. Inequality (4.4) implies the existence of a positive constant $\hat{\delta}$ that

$$Pred_k \geq \hat{\delta} \min[\Delta_k, ||x_k - x^*||_2] \tag{4.5}$$

for all large k. From the above inequality, we can show that

$$||x_k - x^*|| < \Delta_k/2 \tag{4.6}$$

holds for all large k. Assume that (4.6) is not true, there exist $k_i(i = 1, 2, ...)$ such that $\Delta_{k_i} < \Delta_{k_i-1}$ and that

$$\Delta_{k_i} \leq 2||x_{k_i} - x^*||. \tag{4.7}$$

The above relation and the fact that $\Delta_{k+1} \geq c_3||s_k||$ imply that

$$\begin{aligned} ||s_{k_i-1}|| &= O(||x_{k_i} - x^*||_2) \\ &= O(||x_{k_i-1} - x^*||_2). \end{aligned} \tag{4.8}$$

(4.8) and (4.5) show that

$$||s_{k_i-1}|| = O(Pred_{k_i-1}), \tag{4.9}$$

which yields that $r_{k_i-1} \to 1$. This contradicts the assumption that $\Delta_{k_i} < \Delta_{k_i-1}$. Therefore (4.6) is true for all large k. Now we shows that

$$||x_k + s_k - x^*|| = O(||x_k - x^*||^2) \tag{4.10}$$

Because x^* is a first order strict minimum and B_k are bounded, we have that

$$\begin{aligned} O(||x_k - x^*||^2) &= \phi_k(x^* - x_k) \\ &\geq \phi_k(s_k) \\ &\geq h(F(x_k + s_k)) + O(||s_k||^2) \\ &\geq \delta||x_k + s_k - x^*|| + O(||s_k||^2) \end{aligned} \tag{4.11}$$

We claim that

$$\lim_{k\to\infty} ||s_k|| = 0. \tag{4.12}$$

If the above relation is not true, trial steps will be rejected for infinitely many times. This implies that trust region bound will be reduced by a fraction for infinitely many times. Due to our assumption that the trust region bound does not increase if the current trust region constraint is inactive, it follows that $\Delta_k \to 0$, which implies (4.12). Thus, we have shown that (4.12) is true. Now (4.12), (4.11) and the fact that $x_k \to x^*$ implies that

$$||x_k + s_k - x^*|| = O(||x_k - x^*||^2) \tag{4.13}$$

The above relation and the fact that x^* is a first order strict minimum imply that $r_k \to 1$. Therefore $x_{k+1} = x_k + s_k$ for all large k. This shows that x_k converges quadratically. □

When $n = m$, x^* is a first order strict minimum is equivalent to the assumption that $J(x^*)$ is non-singular at x^*. The above superlinear convergence result shows that $\{B_k\}$ can be any matrices as long as they are uniformly bounded. The simplest choice is to let $B_k = 0$ for all k. This is indeed the case for many algorithms, for example, those given by Duff, Nocedal and Tapia (1987), El Hallabi and Tapia (1993).

Without assuming x^* is a first order strict minimum, Powell and Yuan (1984) studies conditions for the local superlinear convergence of Algorithm 2.1 when $h(.) = ||.||_\infty$ and $h(.) = ||.||_1$. They consider the case that x^* is a stationary point but $F(x^*) \neq 0$, and assume that the following second order sufficient conditions hold, that is, the Hessian of the Lagrange function is positive semi-definite in the null space of Jacobi. Their main result is as follows:

Theorem 4.3 *If $h(.) = ||.||_1$ or $h(.) = ||.||_1$, if x_k generated by Algorithm 2.1 converges to x^*, if $h(F(x^*)) > 0$ and the second order sufficient condition is satisfied at x^*, and if $||s_k|| < \Delta_k$ for all sufficiently large k, then s_k is a superlinear convergent step, namely*

$$\lim_{k\to\infty} \frac{||x_k + s_k - x^*||}{||x_k - x^*||} = 0 \tag{4.14}$$

if and only if

$$\lim_{k\to\infty} \frac{||P^*(W^* - B_k)s_k||}{||s_k||} = 0 \tag{4.15}$$

where $W^* = \sum_{i=1}^m \lambda_i^* \nabla^2 f_i(x^*)$, P^* is the projection from $\Re^n$ to the null space of the Jacobi at x^*, and λ_i^* are the Lagrange multipliers.

More details can be found in Powell and Yuan (1984). The above result indicate that when the null space of $J(x^*)$ is not empty, second order informations are need in order to get superlinear convergence.

Unfortunately, the assumption that $h(F(x^*)) > 0$ in Theorem 4.3 implies that the theorem can not be applied to the case that x_k converges to a solution of the nonlinear system $F(x) = 0$. Consider the following simple problem

$$f_1(u, v) \equiv u + v^2 = 0 \tag{4.16}$$

$$f_2(u, v) \equiv u - v^2 = 0 \tag{4.17}$$

where $(u, v)^T = x \in \Re^2$. It can be see that $x^* = (0, 0)^T$ is the unique solution. For $x_k = (0, v_k)^T$, consider the subproblem

$$\min ||F_k + J_k d|| \tag{4.18}$$

which gives a trial step $s_k = (0, -v_k/2)^T$. Thus $x_{k+1} = x_k/2$, the sequence converges only linearly. For this problem, $||F(x)||$ is second order strict minimum in the sense that

$$||F(x)||_2 \geq ||F(x^*)||_2 + \frac{1}{2}||x - x^*||_2^2 \tag{4.19}$$

for all small x. It satisfies the second order sufficient condition that the Hessian of the Lagrange function is positive definite in the null space of the Jacobi. Now we consider that the trial step s_k is computed by minimizing (2.2)-(2.3). Due to the second order sufficient condition, it is reasonable to let B_k be a positive semi-definite matrix. Because B_k is positive definite, this would produce a trial step s_k whose length is at most $||x_k||/2$. Therefore we can only get linear convergence.

Another assumption, $||s_k|| < \Delta_k$ for all large k, in Theorem 4.3 may fail in computations. Yuan (1984) gave a minimax problem that trust region bounds are active at all iterations. Consequently linear convergence can happen.

5. Discussion

We have discussed in this paper a general trust region algorithm for solving nonlinear equations. Our approach is to rewrite the system of equations into a minimization problem. Our local models are a linearized part and a second order term. It is showed that the second order term does not affect the algorithm much when the iterate points converges to a first order strict minimum.

Our linear model are based on the Taylor expansion. In other words, we use the Jacobi to build approximate functions. Thus our algorithm can be viewed as a trust region globalization of Newton's method.

In optimization, numerical methods that do not compute any derivatives are called direct methods. For nonlinear equations, there are also methods that do not compute the Jacobi, such as the Brent method. Direct methods are of great interests, especially when the number of variables are very large. Therefore It is interesting to study trust region algorithms that use only function values.

Acknowledgements

This work was carried out during my visit at the Department of Mathematics of Hong Kong Baptist College, under the support of Croucher Foundation. I would like to thank Prof. H.C. Huang and Dr. F.J. Hickernell for the excellent facilities and nice research environment in the Department, and for the many valuable discussions that we had during my enjoyable visit there.

References

[1] I. Duff, J. Nocedal and J. Reid, "The use of linear programming for the solution of sparse sets of nonlinear equations", *SIAM J. Sci. Stat. Comput.* 8(1987) 99-108.

[2] M. El Hallabi and R. Tapia, "A global convergence theory for arbitrary norm trust-region methods for nonlinear equations" Report MASC TR 93-41, Rice University, Houston, USA.

[3] R. Fletcher, "A model algorithm for composite NDO problem", *Math. Prog. Study* 17(1982) 67-76.

[4] R. Fletcher, "Second order correction for nondifferentiable optimization", in: G.A. Watson, ed., *Numerical Analysis* (Springer-Verlag, Berlin, 1982) pp. 85-115.

[5] R. Fletcher, *Practical Methods of Optimization* (second edition) (John Wiley and Sons, Chichester, 1987)

[6] D.M. Gay, "Computing optimal local constrained step", *SIAM J. Sci. Stat. Comp.* 2(1981) 186-197.

[7] K. Levenberg, "A method for the solution of certain nonlinear problems in least squares", *Quart. Appl. Math.* 2(1944) 164-166.

[8] D.W. Marquardt, "An algorithm for least-squares estimation of nonlinear inequalities", *SIAM J. Appl. Math.* 11(1963) 431-441.

[9] J.J. Moré, "The Levenberg-Marquardt algorithm: implementation and theory", in: G.A. Watson, ed., *Proc. Dundee Conference on Numerical Analysis* (Springer-Verlag, Berlin, 1978)

[10] J.J. Moré, "Recent developments in algorithms and software for trust region methods", in: A. Bachem, M. Grötschel and B. Korte, eds., *Mathematical Programming: The State of the Art* (Springer-Verlag, Berlin, 1983) pp. 258-287.

[11] J.J. Moré and D.C. Sorenson, "Computing a trust region step", *SIAM J. Sci. Stat. Comp.* 4(1983) 553-572.

[12] J. Nocedal, private communication, 1994.

[13] M.J.D. Powell, "A new algorithm for unconstrained optimization", in: J.B. Rosen, O.L. Mangassarian and K. Ritter, eds., *Nonlinear Programming* (Academic Press, New York, 1970), pp. 31-66.

[14] M.J.D. Powell, "Convergence properties of a class of minimization algorithms", in: O.L. Mangassarian, R.R. Meyer and S.M. Robinson, eds., *Nonlinear Programming 2* (Academic Press, New York, 1975), pp. 1-27.

[15] M.J.D. Powell, "General algorithm for discrete nonlinear approximation calculations", in: C.K. Chui, L.L. Schumaker and L.D. Ward, eds., *Approximation Theory IV* (Academic Press, N.Y., 1983) pp. 187-218.

[16] M.J.D. Powell, "On the global convergence of trust region algorithm for unconstrained optimization", *Math. Prog.* 29(1984) 297-303.

[17] D.C. Sorensen, "Newton's method with a model trust region modifications", *SIAM J. Numer. Anal.* 19(1982) 409-426.

[18] Y. Yuan, "An example of only linearly convergence of trust region algorithms for nonsmooth optimization", *IMA J. Numer. Anal.* 4(1984) 327-335.

[19] Y. Yuan, "Conditions for convergence of trust region algorithms for nonsmooth optimization", *Math. Prog.* 31(1985) 220-228.

[20] Y. Yuan, "A new trust region algorithm for nonlinear optimization", Report, Computing Center, Academia Sinica, China 1992. (presented at the First International Symposium on Numerical Analysis, Provdiv, Bulgaria, August 13-17 1992)

[21] Y. Yuan, "A short note on the Duff-Nocedal-Reid Algorithm", Technical Report, MATH-041, Dept of Mathematics, Hong Kong Baptist College, Hong Kong, 1994.