Journal of Computational Mathematics, Vol.21, No.5, 2003, 625–636.

A MODIFIED LEVENBERG-MARQUARDT ALGORITHM FOR SINGULAR SYSTEM OF NONLINEAR EQUATIONS *1)

Jin-yan Fan
(*Department of Mathematics, Shanghai Jiaotong University, Shanghai 200240, China*)

Abstract

Based on the work of paper [1], we propose a modified Levenberg-Marquardt algorithm for solving singular system of nonlinear equations $F(x) = 0$, where $F(x) : R^n \to R^n$ is continuously differentiable and $F'(x)$ is Lipschitz continuous. The algorithm is equivalent to a trust region algorithm in some sense, and the global convergence result is given. The sequence generated by the algorithm converges to the solution quadratically, if $||F(x)||_2$ provides a local error bound for the system of nonlinear equations. Numerical results show that the algorithm performs well.

Key words: Singular nonlinear equations, Levenberg-Marquardt method, Trust region algorithm, Quadratic convergence.

1. Introduction

We consider the problem for solving the system of nonlinear equations

$$F(x) = 0, \tag{1.1}$$

where $F(x) : R^n \to R^n$ is continuously differentiable and $F'(x)$ is Lipschitz continuous. Throughout the paper, we assume that the solution set of (1.1) is nonempty and denoted by X^*. And in all cases $||\cdot||$ refers to the 2-norm.

The Levenberg-Marquardt method (see [2, 3]) for nonlinear equations (1.1) computes the trial step by

$$d_k = -(J(x_k)^T J(x_k) + \mu_k I)^{-1} J(x_k)^T F(x_k), \tag{1.2}$$

where $J(x_k) = F'(x_k)$ is the Jocabi, and $\mu_k \geq 0$ is a parameter being updated from iteration to iteration. The Levenberg-Marquardt step (1.2) is a modification of the Newton's step. The parameter μ_k is introduced to overcome the difficulties caused by singularity or near singularity of $J(x_k)$.

There are various choices of the parameter μ_k in (1.2). Recently, paper [11] shows, if the parameter is chosen as $\mu_k = ||F(x_k)||^2$, and if the initial point is sufficiently close to x^*, then, under a weaker condition than nonsingularity that $||F(x)||$ provides a local error bound near the solution, the Levenberg-Marquardt method has a quadratic rate of convergence. Paper [1] extends the result in [11], and obtains that the quadratic convergence still holds if the parameter is chosen as $\mu_k = ||F(x_k)||$. Although the numerical results show that the choice of $\mu_k = ||F(x_k)||$ performs better than that of $\mu_k = ||F(x_k)||^2$ [1], it does not perform very well when the sequence is far away from the solution. In this paper, based on the work of papers [1] and [11], we consider a modified Levenberg-Marquardt method, in which the parameter is chosen as $\mu_k||F(x_k)||$ with μ_k being updated by trust region techniques.

* Received April 4, 2001; final revised June 20, 2002.

Definition 1.1. *Let N be a subset of R^n such that $N \cap X^* \neq \emptyset$. We say that $||F(x)||$ provides a local error bound on N for system (1.1), if there exists a positive constant $c > 0$ such that*

$$||F(x)|| \geq c\,dist(x, X^*), \qquad \forall x \in N.$$

Note that, if $J(x^*)$ is nonsingular at a solution x^* of (1.1), then x^* is an isolated solution, hence $||F(x)||$ provides a local error bound on some neighbourhood of x^*. However, the converse is not necessarily true, see example in [11]. Thus, a local error bound condition is weaker than nonsingularity.

In the next section, we present the modified algorithm, and show that it is equivalent to a trust region algorithm. In section 3, global convergence result is proved. The algorithm converges to a stationary point if a trial step is accepted only when the actual reduction of the 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 the algorithm converges quadratically if $||F(x)||$ provides a local error bound condition near the solution. Finally in section 5, we present the numerical results for some singular systems of nonlinear equations.

2. Modified Levenberg-Marquardt Algorithm and Trust Region

In this section, we first present the general trust region algorithm, then present our new Levenberg-Marquardt algorithm. The relationship between these two algorithms is given.

At the beginning of each iteration in a general trust region algorithm for nonlinear equations, a trial step d_k is computed by solving the subproblem:

$$\begin{aligned} \min_{d \in R^n} \ & ||F_k + J_k d||^2 \stackrel{\triangle}{=} \varphi_k(d) \\ s.\ t.\ \ & ||d|| \leq \Delta_k, \end{aligned} \tag{2.1}$$

where $F_k = F(x_k), J_k = J(x_k)$, and $\Delta_k > 0$ is the current trust region bound. The actual reduction and the predicted reduction of the function are defined as follows:

$$Ared_k = ||F_k||^2 - ||F(x_k + d_k)||^2,$$

and

$$Pred_k = \varphi_k(0) - \varphi_k(d_k).$$

The ratio between these two reductions is defined by

$$r_k = \frac{Ared_k}{Pred_k},$$

which is used to decide whether the trial step is acceptable and to adjust the new parameter μ_k. Paper [12] presents a class of trust region algorithms for nonlinear equations in any arbitrary norm, and gives the global and local convergence results. The general trust region algorithm for nonlinear equations in 2-norm can be stated as follows :

Algorithm 2.1. *(Trust region algorithm for nonlinear equations)*

Step 1. Given $x_1 \in R^n, \Delta_1 > 0, \varepsilon \geq 0, 0 \leq p_0 \leq p_1 \leq p_2 < 1, k := 1$.

Step 2. If $||J_k^T F_k|| \leq \varepsilon$, then stop;
Solve (2.1) giving d_k.

Step 3. Compute $r_k = Ared_k / Pred_k$;

set

$$x_{k+1} = \begin{cases} x_k + d_k, & \text{if } r_k > p_0, \\ x_k, & \text{otherwise.} \end{cases} \tag{2.2}$$

Step 4. Choose Δ_{k+1} *as*

$$\Delta_{k+1} = \begin{cases} \min\{\frac{\Delta_k}{4}, \frac{||d_k||}{2}\} & \text{if } r_k < p_1, \\ \Delta_k, & \text{if } r_k \in [p_1, p_2], \\ \max\{4||d_k||, 2\Delta_k\}, & \text{if } r_k > p_2; \end{cases} \tag{2.3}$$

$k := k + 1$; *go to Step 2.*

The algorithm above has a strongly global convergence with $p_0 > 0$, that is, all accumulation points are stationary points. However, for $p_0 = 0$, it only has a weekly global convergence, that is, at least one accumulation point is a stationary point.

We now present our Modified Levenberg-Marquardt algorithm.

Algorithm 2.2. *(Modified Levenberg-Marquardt algorithm for nonlinear equations)*

Step 1. Given $x_1 \in R^n, \varepsilon \geq 0, \mu_1 > m > 0, 0 \leq p_0 \leq p_1 \leq p_2 < 1, k := 1.$

Step 2. If $||J_k^T F_k|| \leq \varepsilon$, *then stop;*

Solve $(J_k^T J_k + \mu_k ||F_k|| I) d_k = -J_k^T F_k$ *giving* d_k.

Step 3. Compute $r_k = Ared_k / Pred_k$;

set

$$x_{k+1} = \begin{cases} x_k + d_k, & \text{if } r_k > p_0, \\ x_k, & \text{otherwise.} \end{cases}$$

Step 4. Choose μ_{k+1} *as*

$$\mu_{k+1} = \begin{cases} 4\mu_k, & \text{if } r_k < p_1, \\ \mu_k, & \text{if } r_k \in [p_1, p_2], \\ \max\{\frac{\mu_k}{4}, m\}, & \text{if } r_k > p_2; \end{cases} \tag{2.4}$$

$k := k + 1$; *go to Step 2.*

It is easily seen from our new algorithm that

$$d_k = -(J_k^T J_k + \mu_k ||F_k|| I)^{-1} J_k^T F_k \tag{2.5}$$

is the solution of

$$\min_{d \in R^n} ||F_k + J_k d||^2 + \mu_k ||F_k|| ||d||^2 \triangleq \psi_k(d_k). \tag{2.6}$$

Define

$$\Delta_k = ||(J_k^T J_k + \mu_k ||F_k|| I)^{-1} J_k^T F_k||, \tag{2.7}$$

then it is not difficult to show that the Levenberg-Marquardt step (2.5) ia also a solution to the problem (2.1). In fact, if we let the trust region radius Δ_k be given by (2.7) in every iteration, then our algorithm is essentially a trust region algorithm, and in every iteration, this trust region method has active constraints. However, the general trust region algorithm updates the trust region by (2.3) directly, while our algorithm modifies the parameter μ_k in every iteration, which in turn modifies the value Δ_k from (2.7) implicitly. Many other papers also consider the Levenberg-Marquardt method and the trust regionthe method, for more details, see [4, 5, 13, 14], etc. .

In Algorithm 2.2, m is a given constant and is the lowerbound of the the parameter μ_k. It plays the role to prevent the step from being too large when the sequence is near the solution.

3. Global Convergence

To get the global convergence of the algorithm, we first make the following assumption.

Assumption 3.1. *$F(x)$ is continuously differentiable, and both $F(x)$ and its Jacobi $J(x)$ are Lipschitz continuous, i. e., there exist positive constants L_1 and L_2 such that*

$$||J(y) - J(x)|| \leq L_1||y - x||, \qquad \forall x, y, \tag{3.1}$$

and

$$||F(y) - F(x)|| \leq L_2||y - x||, \qquad \forall x, y. \tag{3.2}$$

Now from the famous result of Powell [8], we obtain the following lemma.

Lemma 3.1. *Let d_k be computed by (2.5), then the inequality*

$$Pred_k = \varphi_k(0) - \varphi_k(d_k) \geq ||J_k^T F_k|| \min\{||d_k||, \frac{||J_k^T F_k||}{||J_k^T J_k||}\} \tag{3.3}$$

holds for all $k \geq 1$.

Theorem 3.1. *Under the conditions of Assumption 3.1, if $p_0 > 0$, then the sequence generated by Algorithm 2.2 satisfies*

$$\lim_{k\to\infty} ||J_k^T F_k|| = 0. \tag{3.4}$$

Proof. If the theorem is not true, then there exist a positive τ and infinite many k such that

$$||J_k^T F_k|| \geq \tau. \tag{3.5}$$

Let K and T be the sets of all indices that satisfy

$$S = \{k \mid ||J_k^T F_k|| \geq \tau/2\},$$

and

$$T = \{k \mid x_{k+1} \neq x_k, k \in S\}.$$

Then we have from Lemma 3.1 that

$$\begin{aligned} ||F_1||^2 &\geq \sum_{k\in S}(||F_k||^2 - ||F_{k+1}||^2) \\ &= \sum_{k\in T}(||F_k||^2 - ||F_{k+1}||^2) \\ &\geq \sum_{k\in T} p_0 Pred_k \\ &\geq \sum_{k\in T} p_0||J_k^T F_k|| \min\{||d_k||, \frac{||J_k^T F_k||}{||J_k^T J_k||}\} \\ &\geq \sum_{k\in T} \frac{p_0\tau}{2} \min\{||d_k||, \frac{\tau}{L_1^2}\}. \end{aligned} \tag{3.6}$$

The above inequality implies that

$$\sum_{k\in T} ||d_k|| < +\infty. \tag{3.7}$$

Therefore it follows from (3.2) and (3.7) that

$$\sum_{k\in T}\left|\,||J_k^TF_k|| - ||J_{k+1}^TF_{k+1}||\,\right| < +\infty. \tag{3.8}$$

Relation (3.8) and the fact that (3.5) holds for infinitely many k indicate that there exists $\hat{k}$ such that $||J_{\hat{k}}^TF_{\hat{k}}|| \geq \tau$ and

$$\sum_{k\in T, k\geq\hat{k}}\left|\,||J_k^TF_k|| - ||J_{k+1}^TF_{k+1}||\,\right| < \frac{\tau}{2}. \tag{3.9}$$

By induction, we see that $||J_k^TF_k|| \geq \tau/2$ for all $k \geq \hat{k}$. This result and (3.7) imply that $\lim_{k\to\infty} x_k$ exists, which shows that $\mu_k \to +\infty$. On the other hand, it follows from (3.5), (3.7) and Lemma 3.1 that

$$\begin{aligned} r_k &= \frac{Ared_k}{Pred_k} \\ &= 1 + \frac{||F_k + J_kd_k||O(||d_k||^2) + O(||d_k||^4)}{Pred_k} \\ &\leq 1 + \frac{||F_k + J_kd_k||O(||d_k||^2) + O(||d_k||^4)}{||J_k^TF_k||\min\{||d_k||, \frac{||J_k^TF_k||}{||J_k^TJ_k||}\}} \\ &\leq 1 + \frac{O(||d_k||^2)}{||d_k||} \\ &\to 1. \end{aligned} \tag{3.10}$$

In view of Algorithm 2.2, we know there exists a positive constant $M > m$ such that

$$\mu_k < M$$

holds for all large k, which gives a contradiction. Therefore we see that assumption (3.5) can not be true. The proof is completed.

4. Local Convergence

In this section, we study the local convergence properties of Algorithm 2.2. We assume that the sequence $\{x_k\}$ converges to a solution $x^* \in X^*$, and we also make the following assumption.
Assumption 4.1. $||F(x)||$ *provides a local error bound on* $N(x^*, b)$ *for the system (1.1), i.e., there exist two positive constants* $c > 0$ *and* $b < 1$ *such that*

$$||F(x)|| \geq c_1\,dist(x, X^*), \qquad \forall x \in N(x^*, b) = \{x \mid ||x - x^*|| \leq b\}. \tag{4.1}$$

Under such a condition, we can show that $\{x_k\}$ converges to x^* quadratically. In the following, we denote $\bar{x}_k$ the vector in X^* that satisfies

$$||x_k - \bar{x}_k|| = \mathrm{dist}(x_k, X^*).$$

Suppose $\mathrm{rank}(J(x^*)) = r$, and the singular value decomposition (SVD) of $J(x^*)$ is

$$\begin{aligned} J(x^*) &= U^*\Sigma^*V^{*T} \\ &= (U_1^*, U_2^*)\begin{pmatrix} \Sigma_1^* & \\ & 0 \end{pmatrix}\begin{pmatrix} V_1^{*T} \\ V_2^{*T} \end{pmatrix} \\ &= U_1^*\Sigma_1^*V_1^{*T}, \end{aligned}$$

where $\Sigma_1^* = \mathrm{diag}(\sigma_1^*, \cdots, \sigma_r^*)$ with $\sigma_1^* \geq \cdots \geq \sigma_r^* > 0$. Suppose the SVD of $J(x_k)$ is

$$\begin{aligned} J(x_k) &= U_k \Sigma_k V_k^T \\ &= (U_{k,1}, U_{k,2}, U_{k,3}) \begin{pmatrix} \Sigma_{k,1} & & \\ & \Sigma_{k,2} & \\ & & 0 \end{pmatrix} \begin{pmatrix} V_{k,1}^T \\ V_{k,2}^T \\ V_{k,3}^T \end{pmatrix} \\ &= U_{k,1}\Sigma_{k,1}V_{k,1}^T + U_{k,2}\Sigma_{k,2}V_{k,2}^T, \end{aligned} \tag{4.2}$$

where $\Sigma_{k,1} = \mathrm{diag}(\sigma_{k,1}, \cdots, \sigma_{k,r})$ and $\Sigma_{k,2} = \mathrm{diag}(\sigma_{k,r+1}, \cdots, \sigma_{k,r+q})$ satisfying $\sigma_{k,1} \geq \cdots \geq \sigma_{k,r} \geq \sigma_{k,r+1} \geq \cdots \geq \sigma_{k,r+q} > 0, q \geq 0$. In the following, if the context is clear, we write (4.2) as

$$J_k = U_1\Sigma_1 V_1^T + U_2 \Sigma_2 V_2^T.$$

To prove the quadratic convergence of the sequence, we first give the following two lemmas.

Lemma 4.1. *Under the conditions of Assumption 3.1, we have*

(a) $||U_1U_1^T F_k|| \leq O(||x_k - \bar{x}_k||)$*;*

(b) $||U_2U_2^T F_k|| \leq O(||x_k - x^*||^2)$*;*

(c) $||U_3U_3^T F_k|| \leq O(||x_k - \bar{x}_k||^2)$*.*

Proof. Result (a) follows immediately from (3.2). By the theory of matrix perturbation [10] and Assumption 3.1, we have

$$||\mathrm{diag}(\Sigma_1 - \Sigma_1^*, \Sigma_2, 0)|| \leq ||J_k - J^*|| \leq L_1||x_k - x^*||.$$

The above relation gives

$$||\Sigma_1 - \Sigma_1^*|| \leq L_1||x_k - x^*|| \qquad \text{and} \qquad ||\Sigma_2|| \leq L_1||x_k - x^*||. \tag{4.3}$$

Let $s_k = -J_k^+ F_k$, where J_k^+ is the pseudo-inverse of J_k. It is easy to see that s_k is the least squares solution of $\min ||F_k + J_k s||$, so we obtain from Assumption 3.1 that

$$||U_3U_3^T F_k|| = ||F_k + J_k s_k|| \leq ||F_k + J_k(\bar{x}_k - x_k)|| \leq O(||x_k - \bar{x}_k||^2).$$

Let $\tilde{J}_k = U_1\Sigma_1 V_1^T$ and $\tilde{s}_k = -\tilde{J}_k^+ F_k$. Since $\tilde{s}_k$ is the least squares solution of $\min ||F_k + \tilde{J}_k s||$, it follows from (3.1) and (4.3) that

$$\begin{aligned} ||(U_2U_2^T + U_3U_3^T)F_k|| &= ||F_k + \tilde{J}_k\tilde{s}_k|| \\ &\leq ||F_k + \tilde{J}_k(\bar{x}_k - x_k)|| \\ &\leq ||F_k + J_k(\bar{x}_k - x_k)|| + ||(\tilde{J}_k - J_k)(\bar{x}_k - x_k)|| \\ &\leq L_1||\bar{x}_k - x_k)||^2 + ||U_2\Sigma_2V_2^T(\bar{x}_k - x_k)|| \\ &\leq O(||x_k - x^*||^2). \end{aligned}$$

Due to the orthogonality of U_2 and U_3, we get result (b).

Lemma 4.2. *Under the conditions of Assumption 3.1 and Assumption 4.1, we have*

$$||d_k|| \leq O(||x_k - \bar{x}_k||). \tag{4.4}$$

Proof. Since d_k is the solution of (2.6), it folows from Assumption 3.1 and (4.1) that

$$\begin{aligned} ||d_k||^2 &\le \frac{1}{\mu_k||F_k||}\psi_k(\bar{x}_k - x_k) \\ &= \frac{1}{\mu_k||F_k||}(||F_k + J_k(\bar{x}_k - x_k)||^2 + \mu_k||F_k||||\bar{x}_k - x_k||^2) \\ &\le \frac{L_1^2||\bar{x}_k - x_k||^4}{c_1 m||\bar{x}_k - x_k||} + ||\bar{x}_k - x_k||^2 \\ &= O(||\bar{x}_k - x_k||^2). \end{aligned}$$

Thus we obtain (4.4).

Now we can give our main result in the following.

Theorem 4.1. *Under the conditions of Assumption 3.1 and Assumption 4.1, the sequence $\{x_k\}$ generated by Algorithm 2.2 converges to the solution quadratically.*

Proof. First we show that for all large k, the predicted reduction satisfies

$$Pred_k \ge c_2||F_k||||d_k||, \tag{4.5}$$

where c_2 is a positive constant. We consider two cases. If $||\bar{x}_k - x_k|| \le ||d_k||$, then it follows from Assumption 3.1, (4.1) and the definition of d_k that

$$\begin{aligned} ||F_k|| - ||F_k + J_k d_k|| &\ge ||F_k|| - ||F_k + J_k(\bar{x}_k - x_k)|| \\ &\ge c_1||\bar{x}_k - x_k|| + O(||\bar{x}_k - x_k||^2) \\ &\ge \hat{c}_1||\bar{x}_k - x_k||. \end{aligned} \tag{4.6}$$

On the other case $||\bar{x}_k - x_k|| > ||d_k||$, we have

$$\begin{aligned} ||F_k|| - ||F_k + J_k d_k|| &\ge ||F_k|| - ||F_k + \frac{||d_k||}{||\bar{x}_k - x_k||}J_k(\bar{x}_k - x_k)|| \\ &\ge \frac{||d_k||}{||\bar{x}_k - x_k|}(||F_k|| - ||F_k + J_k(\bar{x}_k - x_k)||) \\ &\ge \frac{||d_k||}{||\bar{x}_k - x_k||}(c_1||\bar{x}_k - x_k|| + O(||\bar{x}_k - x_k||^2)) \\ &\ge \bar{c}_1||d_k||. \end{aligned} \tag{4.7}$$

Inequalities (4.6), (4.7) and (4.4), together with Lemma 4.2 show that

$$\begin{aligned} Pred_k &= (||F_k|| + ||F_k + J_k d_k||)(||F_k|| - ||F_k + J_k d_k||) \\ &\ge ||F_k||(||F_k|| - ||F_k + J_k d_k||) \\ &\ge c_2||F_k||||d_k|| \end{aligned}$$

holds for some constant $c_2 > 0$. Thus, it follows from (3.2), (4.1), (4.5) and Lemma 4.2 that

$$\begin{aligned} r_k &= \frac{Ared_k}{Pred_k} \\ &= 1 + \frac{O(||d_k||^2)||F_k + J_k d_k|| + O(||d_k||^4)}{Pred_k} \\ &\le 1 + \frac{O(||d_k||^2)||F_k|| + O(||d_k||^4)}{||F_k||||d_k||} \\ &= 1 + O(||d_k||) \\ &\to 1. \end{aligned} \tag{4.8}$$

The inequality above implies that there exists a constant $M > m$ such that

$$\mu_k < M \tag{4.9}$$

holds for all large k.

We now show that the sequence has quadratic convergence. By the SVD of J_k, we know the step at the current iterate is

$$d_k = -V_1(\Sigma_1^2 + \mu_k ||F_k|| I)^{-1} \Sigma_1 U_1^T F_k - V_2(\Sigma_2^2 + \mu_k ||F_k|| I)^{-1} \Sigma_2 U_2^T F_k. \tag{4.10}$$

So we have

$$\begin{aligned} F_k + J_k d_k &= F_k - U_1 \Sigma_1 (\Sigma_1^2 + \mu_k ||F_k|| I)^{-1} \Sigma_1 U_1^T F_k \\ &\quad - U_2 \Sigma_2 (\Sigma_2^2 + \mu_k ||F_k|| I)^{-1} \Sigma_2 U_2^T F_k \\ &= \mu_k ||F_k|| U_1 (\Sigma_1^2 + \mu_k ||F_k|| I)^{-1} U_1^T F_k \\ &\quad + \mu_k ||F_k|| U_2 (\Sigma_2^2 + \mu_k ||F_k|| I)^{-1} U_2^T F_k + U_3 U_3^T F_k. \end{aligned} \tag{4.11}$$

Since $\{x_k\}$ converges to x^*, without loss of generality, we assume that $L_1 ||x_k - x^*|| < \sigma_r^*/2$ holds for all large k. Thus we abtain from (4.3) that

$$||(\Sigma_1^2 + \mu_k ||F_k|| I)^{-1}|| \leq ||\Sigma_1^{-2}|| \leq \frac{1}{(\sigma_r^* - L_1 ||x_k - x^*||)^2} < \frac{4}{\sigma_r^{*2}}.$$

It then follows from (3.2),(4.9) and Lemma 4.1 that

$$||F_k + J_k d_k|| \leq \quad O(||x_k - x^*||^2). \tag{4.12}$$

Therefore, we have

$$\begin{aligned} c||x_{k+1} - \bar{x}_{k+1}|| &\leq ||F(x_k + d_k)|| \\ &\leq ||F_k + J_k d_k|| + O(||d_k||^2) \\ &\leq O(||x_k - x^*||^2) \\ &= O(||d_k||^2) \\ &\leq O(||x_k - \bar{x}_k||^2). \end{aligned} \tag{4.13}$$

It now follows from

$$||x_k - \bar{x}_k|| \leq ||d_k|| + ||x_{k+1} - \bar{x}_{k+1}||$$

that

$$||x_k - \bar{x}_k|| \leq 2||d_k||$$

holds for all large k. Thus we see from (4.13) and Lemma 4.2 that $||d_{k+1}|| = O(||d_k||^2)$. Therefore $\{x_k\}$ converges to the solution quadratically, namely,

$$||x_{k+1} - x^*|| \leq O(||x_k - x^*||^2).$$

This completes our proof.

5. Numerical Results

We tested our modified Levenberg-Marquardt Algorithm 2.2 on some singular problems, and compared it with the general trust region Algorithm 2.1.

The test problems were created by modifying those described in Moré, Garbow and Hillstrom [6] for general nonsingular system of nonliner equation except Problem 2(Powell singular function), where $n = 4$ and $\text{rank}(J(x^*)) = 2$. And they have the same form as given in [9]

$$\hat{F}(x) = F(x) - J(x^*)A(A^TA)^{-1}A^T(x - x^*), \tag{5.1}$$

where $F(x)$ is the standard nonsingular test function, x^* is its root, and $A \in R^{n\times k}$ has full column rank with $1 \le k \le n$. Obviously, $\hat{F}(x^*) = 0$ and

$$\hat{J}(x^*) = J(x^*)(I - A(A^TA)^{-1}A^T)$$

has rank $n - k$. A disadvantage of these problems is that $\hat{F}(x)$ may have roots that are not roots of $F(x)$. We created two sets of singular problems, with $\hat{J}(x^*)$ having rank $n - 1$ and $n - 2$, by using

$$A \in R^{n\times 1}, \qquad A^T = (1, 1, \cdots, 1)$$

and

$$A \in R^{n\times 2}, \qquad A^T = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & -1 & 1 & -1 & \cdots & \pm 1 \end{pmatrix},$$

respectively. Meanwhile, we made a slight alteration on the variable dimension problem, which has $n + 2$ equations in n unknowns; we eliminate the $n - 1$th and nth equations. (The first n equations in the standard problem are linear.)

We used $p_0 = 0.0001, p_1 = 0.25$ and $p_2 = 0.75$, which are popular for tests in trust region method. And the choices of m is $m = 10^{-8}$. We applied Algorithm 2.6 in [7] to solve the trust region subproblem (2.1) in Algorithm 2.1. And the initial trust region radius is chosen as

$$\Delta_1 = ||(J_1^TJ_1 + \mu_1||F_1||I)^{-1}J_1^TF_1||. \tag{5.2}$$

That is, the first trial steps of the two algorithms are the same.

Table 1. Results on Powell singular problem

			LM			TR		
Problem	n	x_0	NF	NJ	n.s.x^*?	NF	NJ	n.s.x^*?
2	4	1	10	10	N	10	10	N
		10	13	13	N	13	13	N
		100	16	16	N	16	16	N

We test several choices of initial μ_1 for Algorithm 2.1 and Algorithm 2.2. Algorithm 2.1 performs little better than Algorithm 2.2 for the choice of $\mu_1 = 10^{-2}$, while it performs little worse than Algorithm 2.2 for $\mu_1 = 10^{-6}$. We only listed the results for $\mu_1 = 10^{-4}$. The algorithm is terminated when the norm of $J_k^TF_k$, e.g.,the derivative of $\frac{1}{2}||F(x)||^2$ at the k-th iteration, is less than $\varepsilon = 10^{-5}$, or when the number of the iterations exceeds $100(n + 1)$. The results for the Powell singular problem are listed in Table 1; those for the first set problems of rank $n-1$ are listed in Table 2, and the second set of rank $n-2$ in Table 3. The third column of the table indicates that the starting point is $x_1, 10x_1$, and $100x_1$, where x_1 is suggested by Moré, Garbow and Hillstrom in [6]; "NF" and "NJ" represent the numbers of function calculations and Jacobi calculations, respectively; and "n.s.x^*?" gives a Y(yes) if the method converges to the same solution as the corresponding nonsingular problem, a N(no) otherwise. If the method failed to find the solution in $100(n + 1)$ iterations, we denoted it by the sign "–". And if the iterations have underflows or overflows, we denoted it by OF.

From the results, we observe that for Powell singular problem, our Algorithm 2.2 performs the same as the general trust region Algorithm 2.1.

For the first singular set problems with $\text{rank}(F'(x^*)) = n-1$, results are excluded for Problem 3 on $100x_0$ case and for Problem 6 on $10x_0$ and $100x_0$ case, because both the algorithms failed to find the solution in $100(n+1)$ iterations. For five problems, our algorithm outperforms the general trust region algorithm, while for the other three, Algorithm 2.1 outperforms ours.

Table 2. Results on first singular test set with $\text{rank}(F'(x^*)) = n-1$

			LM			TR		
Problem	n	x_0	NF	NJ	n.s.x^*?	NF	NJ	n.s.x^*?
1	2	1	15	15	Y	15	15	Y
		10	17	17	Y	17	17	Y
		100	21	21	Y	21	21	Y
3	2	1			–	28	27	Y
		10	294	181	Y	–	–	–
4	4	1	16	16	Y	16	16	Y
		10	19	19	Y	19	19	Y
		100	22	22	Y	22	22	Y
5	3	1	8	8	N	8	8	N
		10	8	8	N	8	8	N
		100	8	8	N	8	8	N
6	31	1	43	23	N	–	–	–
8	10	1	8	8	Y	9	9	Y
		10	23	23	Y	23	23	Y
		100			OF			OF
9	10	1	4	4	N	4	4	N
		10	7	7	N	8	8	N
		100	9	9	N	10	10	N
10	30	1	5	5	Y	6	6	Y
		10	7	7	Y	9	9	Y
		100	10	10	N	10	10	N
11	30	1	15	8	Y	9	7	Y
		10	30	16	Y	20	14	Y
		100	95	80	N*	156	121	N*
12	10	1	14	14	Y	14	14	Y
		10	16	16	Y	16	16	Y
		100	19	19	Y	19	19	Y
13	30	1	23	10	Y	11	10	Y
		10	28	15	Y	17	15	Y
		100	31	18	Y	19	19	Y
14	30	1	11	11	Y	11	11	Y
		10	17	17	Y	17	17	Y
		100	22	22	Y	23	23	Y

Table 3. Results on second singular test set with $\text{rank}(F'(x^*)) = n - 2$

			LM			TR		
Problem	n	x_0	NF	NJ	n.s.x^*?	NF	NJ	n.s.x^*?
1	2	1	11	11	N	24	24	N
		10	13	13	N	31	31	N
		100	17	17	N	38	38	N
3	2	1	35	25	N	59	51	N
		10	59	54	N	16	16	N
		100	25	18	N	44	43	N
4	4	1	14	14	N	–	–	–
		10	17	17	N	–	–	–
		100	20	20	N	–	–	–
5	3	1	13	13	Y	13	13	Y
		10	14	14	Y	14	14	Y
		100	24	18	Y	66	44	Y
6	31	1	93	67	N	–	–	–
8	10	1	8	8	Y	329	329	Y
		10	23	23	Y			OF
		100			OF			OF
9	10	1	9	4	N	6	4	N
		10	16	9	N	10	8	N
		100	10	10	N	10	10	N
10	30	1	12	8	Y	12	8	N
		10	15	10	N	23	16	N
		100	10	10	N	10	10	N
11	30	1	14	9	N	25	17	N
		10	28	15	Y	24	17	Y
		100	58	46	N*	89	65	N*
12	10	1	14	14	Y	–	–	–
		10	16	16	N	–	–	–
		100	19	19	N	–	–	–
13	30	1	22	9	Y	11	10	Y
		10	27	14	Y	15	14	Y
		100	31	19	Y	21	18	Y
14	30	1	11	11	Y	11	11	Y
		10	17	17	Y	17	17	Y
		100	22	22	Y	23	23	Y

For the second singular set problems with $\text{rank}(F'(x^*)) = n - 2$, we can see that Algorithm 2.2 usually performs better than Algorithm 2.1. Hence, our algorithm seems to be more efficient for problems with higher rank deficiency of $J(x^*)$, and with smaller initial μ_1, that is, the larger initial trust region radius.

Finally, it is worth pointing out that on the $100x_0$ case for Problem 11, both methods converge to a stationary point of $\min_{x\in R^n} ||F(x)||$, instead of that of $F(x) = 0$.

Acknowledgments. The author would like to thank her supervisor, Prof. Yaxiang Yuan for his careful reading of the paper, and for his insightful comments, which significantly improves the original version of the paper.

References

[1] J. Y. Fan and Y. X. Yuan, On the convergence of a new Levenberg-Marquardt method, Report No. 005, AMSS, Chinese Academy of Sciences, 2001.

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

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

[4] J. J. Moré, The Levenberg-Marquardt algorithm: implementation and theory, in: G. A. Watson, ed., Lecture Notes in Mathematics 630: Numerical Analysis, Springer-Verlag, Berlin, 1978, 105-116.

[5] 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 Art, Springer, Berlin, 1983, 258-287.

[6] J. J. Moré, B. S. Garbow and K. H. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Software,* **7** (1981), 17-41.

[7] J. Nocedal and Y. X. Yuan, Combining trust region and line search techniques, in: Y. Yuan, ed., Advances in Nonlinear Programming Kluwer, 1998, 153–175.

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

[9] R. B. Schnabel and P. D. Frank, Tensor methods for nonlinear equations, *SIAM J. Numer. Anal.,* **21** (1984), 815-843.

[10] G. W. Stewart and J. G. Sun, Matrix Perturbation Theory, Academic Press, San Diego, CA, 1990.

[11] N. Yamashita and M. Fukushima, On the rate of convergence of the Levenberg-Marquardt method, Report, presented in the first Sino-Japan optimization meeting, October, 2000.

[12] Y. X. Yuan, Trust region algorithms for nonlinear programming, in: Z.C.SHi, ed., Contemporary Mathematics Vol 163, American Mathematics Society, 1994, 205–225.

[13] Y. X. Yuan, Trust region algorithms for nonlinear equations, *Information,* **1** (1998), 7-20.

[14] Y. X. Yuan, A review of trust region algorithms for optimization, In: J.M. Ball and J.C.R. Hunt, eds., ICM99: Proceedings of the Fourth International Congress on Industrial and Applied Mathematics, Oxford University Press, 2000, 271–282.