Appl Math Optim (2007) 56: 379–394
DOI 10.1007/s00245-007-9009-6

A Trust Region Method for Optimization Problem with Singular Solutions

Juliang Zhang · Lingyun Wu · Xiangsun Zhang

Published online: 1 September 2007

Abstract In this paper, we propose a trust region method for minimizing a function whose Hessian matrix at the solutions may be singular. The global convergence of the method is obtained under mild conditions. Moreover, we show that if the objective function is LC^2 function, the method possesses local superlinear convergence under the local error bound condition without the requirement of isolated nonsingular solution. This is the first regularized Newton method with trust region technique which possesses local superlinear (quadratic) convergence without the assumption that the Hessian of the objective function at the solution is nonsingular. Preliminary numerical experiments show the efficiency of the method.

Keywords Nonlinear convex optimization · Trust region method · Local error bound · Superlinear convergence · Nonsingularity

1 Introduction

Consider the following optimization problem

$$\min_{x \in R^n} f(x), \tag{1}$$

This work is partly supported by the National Natural Science Foundation of China (Grant Nos. 70302003, 10571106, 60503004, 70671100) and Science Foundation of Beijing Jiaotong University (2007RC014).

J. Zhang (✉)
Research Institute of Material Flow, School of Economics and Management, Beijing Jiaotong University, Beijing, 100044, China
e-mail: zhangjl@bjtu.edu.cn

L. Wu · X. Zhang
Institute of Applied Mathematics, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, 100080, China

where $f : R^n \to R$ is LC^2 on R^n. That is, it is a twice continuously differentiable function and the Hessian, $\nabla^2 f$, of f is Lipschitz continuous on R^n, namely, there is a constant $L > 0$ such that

$$\|\nabla^2 f(x) - \nabla^2 f(y)\| \le L\|x - y\| \tag{2}$$

for all $x, y \in R^n$. Throughout the paper, denote by $g(x) = \nabla f(x)$ and $G(x) = \nabla^2 f(x)$ the gradient vector and the Hessian matrix of f at x respectively. Let X be the set of local minimizers of $f(x)$.

Trust region methods are quite a way of solving (1). Because of their strong convergence properties and robustness, the trust region methods have been studied extensively and numerous softwares based on trust region methods have been developed since 1970's [2, 4, 6, 7, 11]. Furthermore, they are locally superlinearly (quadratically) convergent at a nonsingular solution $\bar{x}$. We call a solution $\bar{x}$ of problem (1) nonsingular if $G(\bar{x})$ is a nonsingular matrix. Otherwise we call a solution $\bar{x}$ singular. It is clear that a nonsingular solution is locally isolated. We are particularly interested in the case where problem (1) may have singular solutions. When the trust region methods are applied to such a problem, the superlinear (quadratic) rate of convergence may no longer be guaranteed.

Recently, there have been some progresses in convergence analysis of Newton-type methods and trust region methods for nonlinear equation system

$$H(x) = 0 \tag{3}$$

with singular solutions. Yamashita and Fukushima [10] first proved that the Levenberg–Marquadt method is quadratically convergent if $\|H(x)\|$ provides a local error bound for problem (3). Then Fan and Yuan [3], Zhang [14] improved the results in [10]. Zhang and Wang [13] proposed a trust region method for solving (3) and proved that the trust region method retains a superlinear convergence if $\|H(x)\|$ provides a local error bound for problem (3). Li et al. [5] proposed several regularized Newton methods for convex minimization problems and prove the superlinear convergence of their methods under the local error bound condition. Following the definition of the local error bound for nonlinear equation system (3) in [10], the concept of a local error bound for problem (1) is defined as follows:

Definition 1.1 [5] A function $F : R^n \to R$ is said to provide a local error bound for problem (1) near $\bar{x} \in X$ if there exist a neighborhood $\Omega(\bar{x})$ of $\bar{x}$ and a constant $\bar{c} > 0$ such that for all $x \in \Omega(\bar{x})$

$$F(x) \ge \bar{c} \cdot \mathrm{dist}(x, X), \tag{4}$$

where $\mathrm{dist}(x, X)$ denotes the distance from point x to the set X.

In the remaining part of the paper, we will omit the phrase "for problem (1)" when we talk about a local error bound. It is easy to show that if the Hessian matrix $G(\bar{x})$ is nonsingular, $\|g(x)\|$ provides a local error bound near $\bar{x}$. However, the converse is not true in general. An example is given in [10], which shows that the local error bound condition is weaker than nonsingularity of $G(\bar{x})$.

As mentioned above, the superlinear (quadratic) convergence of all the trust region methods at present are obtained under the assumption that the solution $\bar{x}$ of (1) is isolated and $G(\bar{x})$ is nonsingular. This assumption seems too restrictive to properly determine the function category in which the trust region methods have superlinear convergence. The purpose of this paper is to show that trust region method with some improvements will possesses the superlinear (quadratic) convergence without the assumption of nonsingularity. In detail, our trust region method possesses the following merits:

- The subproblem retains the possible sparsity of $G(x)$;
- The trust region radius is adjusted adaptively;
- Global convergence is guaranteed;
- Superlinear (quadratic) convergence is obtained without the assumption that the solution $\bar{x}$ is isolated and $G(\bar{x})$ is nonsingular;
- The performance of the method is notable.

Note that in our method (see Sect. 2), the trust region radius approaches zero as the method proceed. This is the key to prove the superlinear convergence of our method without the assumption that $G(\bar{x})$ is positive definite. For the classic trust region methods, we can show that the trust region radius is bounded away from zero. Then the Newton step (Quasi-Newton step) is a feasible solution of trust region subproblem eventually. The trial step is the Newton step (Quasi-Newton step) and this trial step is accepted after finite iterations. So the trust region methods reduce to the Newton method after finite iterations and superlinear convergence is achieved under the nonsingularity assumption. However, if the Hessian of $f(x)$ at the solution of (1) is singular, the solution of the trust region subproblem in traditional trust region methods (see (10)) is on the boundary and the trial step does not converge to 0. Therefore, the classic trust region methods can not converge superlinearly without the assumption that $G(\bar{x})$ is nonsingular. In our algorithm, the trust region radius δ_k is the same order as $\|x_k - \bar{x}\|$. This ensures the superlinear convergence of the new trust region method.

The remainder of the paper is arranged as follows. In Sect. 2, we state the algorithm model and study the global convergence. In Sect. 3, we show the local convergence of the algorithm. In Sect. 4, numerical results are reported to show the efficiency of the method. In Sect. 5, some conclusions are given.

2 Algorithm Model and Global Convergence

In our algorithm, at each iterative point x_k, the trial step is obtained by solving the following subproblem

$$\begin{aligned} \min \quad & \Phi_k(d) = g_k^T d + \frac{1}{2} d^T (B_k + \mu_k I) d \\ \text{s.t.} \quad & \|d\| \le c^p \|g_k\|^\gamma \triangleq \delta_k^p, \end{aligned} \tag{5}$$

where $g_k = g(x_k)$, B_k is an $n \times n$ matrix, which equals to $G_k = G(x_k)$ or is obtained by some iterative formula (such as quasi-Newton formula). I is an identity matrix with proper dimension. p is a nonnegative integer. c and γ satisfy $0 < c < 1$ and $0 < \gamma < 1$ respectively. μ_k is a adjustable parameter satisfying $\mu_k = \Theta((\text{dist}(x_k, X)^\rho))$ $(0 < \rho \leq 1)$, which means that there exist two constants $0 < c_1 < C_1$ such that

$$c_1 \cdot \text{dist}(x_k, X)^\rho \leq \mu_k \leq C_1 \cdot \text{dist}(x_k, X)^\rho. \tag{6}$$

If $\|g(x)\|$ provides a local error bound, there are many choices of μ_k satisfying (6), for example, $\mu_k = \|g(x_k)\|^\rho$ or $\mu_k = \|G_k g_k\|^\rho$, etc., see [14]. Let d_k^p be the solution of (5) corresponding to p. Then we define the actual reduction as

$$\text{Ared}_k(d_k^p) = f(x_k + d_k^p) - f(x_k), \tag{7}$$

the predict reduction as

$$\text{Pred}_k(d_k^p) = \Phi_k(d_k^p), \tag{8}$$

and the ratio of actual reduction over predict reduction as

$$r_k^p = \frac{\text{Ared}_k(d_k^p)}{\text{Pred}_k(d_k^p)}. \tag{9}$$

Now we state our trust region method in detail.

Algorithm 2.1 Step 0. Given an initial point $x_0 \in R^n$ and an initial symmetric matrix B_0. Let $0 < \eta < 1, 0 < c < 1, \epsilon > 0, 0 < \gamma < 1, 0 < \rho < 1$ and $p = 0$. Set $k := 0$;

Step 1. If $\|g(x_k)\| \leq \epsilon$, stop. Otherwise, solve (5) to obtain the trial step d_k^p;

Step 2. Calculate $\text{Pred}_k(d_k^p)$, $\text{Ared}_k(d_k^p)$ and r_k^p. If $r_k^p \geq \eta$, let $x_{k+1} = x_k + d_k^p$ and go to Step 3. Otherwise, set $p := p + 1$ and go to Step 1.

Step 3. Update B_{k+1}. Set $k := k + 1$ and $p = 0$. Choose $\gamma \in (0, 1)$ and go to Step 1.

Remark (i) B_k may be set as G_k. In order to reduce computation, B_k can also be updated by some quasi-Newton formula (for example BFGS, DFP, etc.).

(ii) Traditionally, the trust region subproblem is

$$\begin{aligned} \min \quad & \Phi_k(d) = g_k^T d + \frac{1}{2} d^T B_k d \\ \text{s.t.} \quad & \|d\| \leq \Delta_k. \end{aligned} \tag{10}$$

The differences between (5) and (10) are that there is an additional term in the objective in (5), and the adjustment of trust region radius in classic trust region method, Δ_k, is based on the information of iteration x_{k-1} while the determination of trust region radius in our method is based on the information of the current iterative point. The motivation to adopt the new update rule is that we think that the approximation of $\Phi_k(d)$ to $f(x_k + d) - f(x_k)$ is different at different point and it is reasonable to adopt the information at the current point to adjust the trust region radius (please see [15, 16] for explanation in detail). By using (5) as the trust region subproblem and the new update rule of the trust region radius, we can prove the superlinear (quadratic) convergence without the assumption that $G(\bar{x})$ is nonsingular. If x_k is far from the solution set of $f(x)$, μ_k may be very large. This makes the trial step very small and the algorithm inefficient. In this case, we can choose μ_k satisfying $\mu_k \leq 0.01$ simultaneously (for example μ_k can be chosen as $\min\{\|g_k\|^\rho, 0.01\}$), the results in this paper still hold.

In order to analyze the global convergence of the algorithm, we need the following assumption.

Assumption 2.1 (i) $f(x)$ is continuously differentiable;
(ii) $\{x_k\}$ is a bounded sequence;
(iii) $\{B_k\}$ is a bounded sequence.

By Assumption 2.1, we have that $\{B_k + \mu_k I\}$ and $\{G_k\}$ are bounded sequences. Hence there exists $M > 0$ such that

$$\|B_k + \mu_k I\| \leq M, \quad \forall k, \tag{11}$$

and

$$\|G_k\| \leq M, \quad \forall k. \tag{12}$$

First, we give several lemmas, which will be used in the analysis of the global convergence.

Lemma 2.1 $|\mathrm{Ared}_k(d_k^p) - \mathrm{Pred}_k(d_k^p)| = O(\|d_k^p\|^2)$.

Proof By Taylor expansion theory, (11) and (12), we have

$$\begin{aligned}\mathrm{Ared}_k(d_k^p) - \mathrm{Pred}_k(d_k^p) &= f(x_k + d_k^p) - f(x_k) - g_k^T d_k^p - \frac{1}{2} d_k^{pT} B_k d_k^p - \frac{1}{2}\mu_k \|d_k^p\|^2 \\ &= \frac{1}{2} d_k^{pT}(G_k - B_k - \mu_k I) d_k^p + o(\|d_k^p\|^2) = O(\|d_k^p\|^2). \quad \square\end{aligned}$$

Lemma 2.2 $\mathrm{Pred}_k(d_k^p) \leq -\frac{1}{2}\min\{\|g_k\|/M, \delta_k^p\}\|g_k\|$.

Proof By the definition of d_k^p, we know that for any $\alpha \in (0, 1)$

$$\begin{aligned}\mathrm{Pred}_k(d_k^p) &= \Phi_k(d_k^p) \leq \Phi_k\left(-\alpha \frac{\delta_k^p}{\|g_k\|} g_k\right) \\ &= -\alpha\delta_k^p \|g_k\| + \frac{\alpha^2 \delta_k^{p2} g_k^T (B_k + \mu_k I) g_k}{2\|g_k\|^2} \leq -\alpha\delta_k^p\|g_k\| + \frac{1}{2}\alpha^2\delta_k^{p2} M.\end{aligned}$$

Thus,

$$\mathrm{Pred}_k(d_k^p) \leq \min_{0\leq\alpha\leq 1}\left\{-\alpha\delta_k^p\|g_k\| + \frac{1}{2}\alpha^2\delta_k^{p2} M\right\} \leq -\frac{1}{2}\min\{\|g_k\|/M, \delta_k^p\}\|g_k\|. \quad \square$$

Lemma 2.3 *Algorithm 2.1 does not cycle between Steps 1 and 2 infinitely.*

Proof If the algorithm cycles between Step 1 and Step 2 at x_k infinitely, then for all $i = 1, 2, \ldots,$ we have $x_{k+i} = x_k$, $p = i$ and $\|g_k\| > \epsilon$. Hence

$$\delta_k^i \to 0 \quad \text{and} \quad r_k^i < \eta. \tag{13}$$

Therefore by Lemmas 2.1 and 2.2, as $i \to \infty$

$$|r_k^i - 1| = \frac{|\mathrm{Ared}_k(d_k^i) - \mathrm{Pred}_k(d_k^i)|}{|\mathrm{Pred}_k(d_k^i)|} \le \frac{O({\delta_k^i}^2)}{0.5\delta_k^i \epsilon} \to 0.$$

Thus, for i sufficiently large

$$r_k^i \ge \eta, \tag{14}$$

which contradicts (13). □

Theorem 2.1 *Suppose that Assumption 2.1 holds. If $\epsilon = 0$, either the algorithm terminates finitely at a solution of (1) or generates an infinite sequence $\{x_k\}$ such that*

$$\lim_{k\to\infty} \|g_k\| = 0. \tag{15}$$

Therefore any accumulation point of $\{x_k\}$ is a stationary point of (1).

Proof Due to Assumption 2.1(i) and (ii), the sequence $\{f(x_k)\}$ is bounded from below. Suppose that the algorithm does not stop finitely and (15) is not true, then there exist a positive constant $\bar{\epsilon}$ and an infinite subsequence $\{k_i\}$ such that $\|g_{k_i}\| \ge \bar{\epsilon}$. Define set $T = \{k \mid \|g_k\| \ge \bar{\epsilon}\}$, then T is an infinite set.

By Lemma 2.2 and Step 2 of Algorithm 2.1, we have

$$\sum_{k\in T}[f(x_k) - f(x_{k+1})] \ge -\sum_{k\in T}\eta \cdot \mathrm{Pred}_k \ge \sum_{k\in T}\eta \cdot \frac{1}{2}\bar{\epsilon}\min\left\{\delta_k^{p(k)}, \frac{\bar{\epsilon}}{M}\right\},$$

where $p(k)$ is the largest p value obtained in Step 2 at iterative point x_k.

Since $\{f(x_k)\}$ is bounded from below, we have

$$\sum_{k\in T}\eta \cdot \frac{1}{2}\bar{\epsilon}\min\left\{\delta_k^{p(k)}, \frac{\bar{\epsilon}}{M}\right\} < +\infty.$$

Then $\delta_k^{p(k)} \to 0$, as $k \to +\infty$ and $k \in T$. Since $\|g_k\| \ge \bar{\epsilon}$ for all $k \in T$, $p(k) \to \infty$ as $k \to \infty$ and $k \in T$. Therefore, we can assume $p(k) \ge 1$ for all $k \in T$.

From the determination rule of $p(k)$ $(k \in T)$ in Step 2, we know that the solution $\bar{d}_k$ corresponding to the following subproblem

$$\begin{aligned} \min_{d\in R^n} \quad & \Phi_k(d) = g_k^T d + \frac{1}{2}d^T B_k d + \frac{1}{2}\mu_k\|d\|^2 \\ \text{s.t.} \quad & \|d\| \le c^{p_k - 1}\|g_k\|^\gamma = \frac{\delta_k^{p(k)}}{c} \end{aligned} \tag{16}$$

is unacceptable. That is, let $\bar{x}_{k+1} = x_k + \bar{d}_k$, we have

$$\frac{f(x_k) - f(\bar{x}_{k+1})}{-\Phi_k(\bar{d}_k)} < \eta, \quad k \in T. \tag{17}$$

On the other hand, we have

$$\delta_k^{p(k)-1} \to 0, \quad k \in T.$$

It follows from Lemma 2.1 that

$$f(\bar{x}_{k+1}) - f(x_k) - \Phi_k(\bar{d}_k) = \mathrm{O}(\|\bar{d}_k\|^2).$$

By Lemma 2.2, $\delta_k^{p(k)-1} \to 0, k \in T$ and the definition of T, for sufficiently large $k \in T$, we have

$$\left| \frac{f(x_k) - f(\bar{x}_{k+1})}{-\Phi_k(\bar{d}_k)} - 1 \right| \le \frac{\mathrm{O}(\|\bar{d}_k\|^2)}{\frac{1}{2}\|g_k\| \min\{\delta_k^{p(k)-1}, \frac{\|g_k\|}{M}\}} \le \frac{\mathrm{O}((\delta_k^{p(k)-1})^2)}{\frac{1}{2}\bar{\epsilon}\delta_k^{p(k)-1}}. \tag{18}$$

Let $k \to \infty, k \in T$, (18) implies

$$\frac{f(x_k) - f(\bar{x}_{k+1})}{-\Phi_k(\bar{d}_k)} \to 1,$$

which contradicts (17). Therefore the theorem is true. □

3 Superlinear Convergence

In order to simplify the presentation, we assume that $f(x)$ is a convex function. For general nonconvex minimization problem, we need only to assume that $x_k \to x^*$ and x^* is a local minimizer of $f(x)$ and $f(x)$ is locally convex at x^*. The results in this section still hold.

To analyze the superlinear convergence, we need the following assumption.

Assumption 3.1 (i) $x_k \to x^*$;

(ii) $f(x)$ is LC^2 and convex;

(iii) For sufficiently large k, B_k is a positive semidefinite matrix and $\|B_k - G_k\| = O(\|d_k^0\|^{\frac{1}{\gamma}})$;

(iv) $\|g(x)\|$ provides a local error bound for problem (1) near x^*, i.e., there exist a neighborhood Ω of x^* and a constant $\bar{c} > 0$ such that for all $x \in \Omega$

$$\|g(x)\| \ge \bar{c} \cdot \mathrm{dist}(x, X). \tag{19}$$

(v) $\mu_k = \Theta((\mathrm{dist}(x_k, X)^\rho))$ $(0 < \rho \le 1)$, i.e., there exist two constants $0 < c_1 < C_1$ such that

$$c_1 \cdot \mathrm{dist}(x_k, X)^\rho \le \mu_k \le C_1 \cdot \mathrm{dist}(x_k, X)^\rho. \tag{20}$$

$\|B_k - G_k\| = O(\|d_k^0\|^{\frac{1}{\gamma}})$ is a little more restrictive than the usually used condition $\|B_k - G_k\| \to 0$. However, if we set $B_k = G_k + \alpha_k I$ and $\alpha_k = 0$ or $\|g_k\|^{\frac{1}{\gamma}}$ or $\|G_k g_k\|^{\frac{1}{\gamma}}$, we can show that $\|B_k - G_k\| \le O(\|d_k^0\|^{\frac{1}{\gamma}})$ holds.

By Assumption (ii), we know that X is the solution set of problem (1) and convex.

From Theorem 2.1 and Assumption 3.1(i), we know that $\lim_{k\to\infty} g_k = 0$ and x^* is a solution of (1).

We first prove that for k sufficiently large, the iterative formula generated by Algorithm 2.1 is

$$x_{k+1} = x_k + \tilde{d}_k, \tag{21}$$

where $\tilde{d}_k = -(B_k + \mu_k I)^{-1} g_k$. To this end, we need to do two things: one is to show that $\tilde{d}_k$ is a solution of (5) corresponding to $p = 0$ for k sufficiently large; another is to prove that $\tilde{d}_k$ is acceptable. First, we give some properties of $\tilde{d}_k$.

Since $G(x^*)$ is symmetric positive semidefinite, there is an orthogonal matrix $Q = (Q_1, Q_2)$ such that

$$G(x^*) = (Q_1, Q_2) \begin{pmatrix} \Lambda & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \end{pmatrix}, \tag{22}$$

where Λ is a diagonal matrix with positive diagonal elements.

Let $Q_k = (Q_{k,1}, Q_{k,2})$ be an orthogonal matrix conformable to $Q = (Q_1, Q_2)$ such that

$$B_k = (Q_{1,k}, Q_{2k}) \begin{pmatrix} \Lambda_{1,k} & 0 \\ 0 & \Lambda_{2,k} \end{pmatrix} \begin{pmatrix} Q_{1,k}^T \\ Q_{2,k}^T \end{pmatrix}, \tag{23}$$

where $\Lambda_{1,k}$ and $\Lambda_{2,k}$ are diagonal matrices. It follows from Assumption 3.1 and Theorem 2.1 that $B_k \to G(x^*)$. Then from the theory of matrix perturbation [9], we know that $\Lambda_{1,k} \to \Lambda$ and $\Lambda_{2,k} \to 0$ as $k \to \infty$. Let $\hat{x} \in X$ be the projection of x onto X, i.e., $\hat{x} \in X$ and $\|x - \hat{x}\| = \text{dist}(x, X)$.

Lemma 3.1 *For k sufficiently large, we have*

(i) $\|Q_{2,k}^T g_k\| = o(\|x_k - \hat{x}_k\|) = o(\text{dist}(x_k, X))$;
(ii) *If $\|g(x)\|$ provides a local error bound near x^*, then there are constants $c_2 > 0$ and $c_3 > 0$ such that*

$$\|Q_{1,k}^T g_k\| \ge c_2 \|x_k - \hat{x}_k\| = c_2 \,\text{dist}(x_k, X)$$

and

$$\|Q_{1,k}^T (x_k - \hat{x}_k)\| \ge c_3 \|x_k - \hat{x}_k\| = c_3 \,\text{dist}(x_k, X)$$

respectively.

Proof (i) Since f is LC^2, it is easy to deduce that

$$\|Q_{2,k}^T g_k\| = \|Q_{2,k}^T (g(x_k) - g(\hat{x}_k))\|$$

$$= \left\| Q_{2,k}^T \int_0^1 G(\hat{x}_k + \tau(x_k - \hat{x}_k)) d\tau (x_k - \hat{x}_k) \right\|$$
$$= \|Q_{2,k}^T G_k(x_k - \hat{x}_k)\| + o(\|x_k - \hat{x}_k\|)$$
$$= \|Q_{2,k}^T B_k(x_k - \hat{x}_k)\| + o(\|x_k - \hat{x}_k\|) + \|Q_{2,k}^T (B_k - G_k)(x_k - \hat{x}_k)\|$$
$$= \|\Lambda_{2,k} Q_{2,k}^T (x_k - \hat{x}_k)\| + o(\|x_k - \hat{x}_k\|)$$
$$= o(\|x_k - \hat{x}_k\|).$$

(ii) It follows from (19) and (i) that

$$\|Q_{1,k}^T g_k\|^2 = \left\| \begin{pmatrix} Q_{1,k}^T \\ Q_{2,k}^T \end{pmatrix} g_k \right\|^2 - \|Q_{2,k}^T g_k\|^2 = \|g_k\|^2 - o(\|x_k - \hat{x}_k\|^2)$$
$$\geq \bar{c}^2 \|x_k - \hat{x}_k\|^2 - o(\|x_k - \hat{x}_k\|^2) \geq \frac{1}{2} \bar{c}^2 \|x_k - \hat{x}_k\|^2,$$

and

$$\bar{c}\|x_k - \hat{x}_k\| \leq \|g_k\| = \|Q_k^T (g(x_k) - g(\hat{x}_k))\| = \|Q_k^T G_k(x_k - \hat{x}_k)\| + o(\|x_k - \hat{x}_k\|)$$
$$= \|Q_k^T B_k(x_k - \hat{x}_k)\| + o(\|x_k - \hat{x}_k\|)$$
$$= \|\Lambda_{1,k} Q_{1,k}^T (x_k - \hat{x}_k)\| + o(\|x_k - \hat{x}_k\|) \leq 2\|\Lambda_{1,k}\| \|Q_{1,k}^T (x_k - \hat{x}_k)\|,$$

which imply that (ii) holds. □

Lemma 3.2 *For k sufficiently large, we have*

(i) *There exists a constant $c_4 > 0$ such that*

$$\|Q_{1,k}^T \tilde{d}_k\| \geq c_4 \operatorname{dist}(x_k, X);$$

(ii) $\|\tilde{d}_k\| = O(\operatorname{dist}(x_k, X))$;

(iii) *There are two positive constants $c_5 < c_6$ such that*

$$c_5 \operatorname{dist}(x_k, X) \leq \|\tilde{d}_k\| \leq c_6 \operatorname{dist}(x_k, X).$$

Proof Since Q_k is orthogonal, we have $\|\tilde{d}_k\|^2 = \|Q_{1,k}^T \tilde{d}_k\|^2 + \|Q_{2,k}^T \tilde{d}_k\|^2$. Therefore, the statement (iii) follows from (i) and (ii). So we only need to prove (i) and (ii).

(i) Notice that $\tilde{d}_k$ is the solution of the following equation

$$(B_k + \mu_k I)d + g_k = 0. \tag{24}$$

Multiplying (24) by $Q_{1,k}^T$ yields

$$(\Lambda_{1,k} + \mu_k I) Q_{1,k}^T \tilde{d}_k + Q_{1,k}^T g_k = 0, \tag{25}$$

where we use $Q_{1,k}^T B_k = \Lambda_{1,k} Q_{1,k}^T$. This implies

$$\begin{aligned}\|Q_{1,k}^T \tilde{d}_k\| &= \|(\Lambda_{1,k} + \mu_k I)^{-1} Q_{1,k}^T g_k\| \geq \|(\Lambda_{1,k} + \mu_k I)\|^{-1} \|Q_{1,k}^T g_k\| \\ &\geq c_2 \|\Lambda_{1,k} + \mu_k I\|^{-1} \|x_k - \hat{x}_k\|. \end{aligned} \tag{26}$$

Since $\Lambda_{1,k} \to \Lambda$ and B_k is positive semidefinite, $2\Lambda > \Lambda_{1,k} > \frac{1}{2}\Lambda > 0$ and $\{\Lambda_{1,k} + \mu_k I\}$ is uniformly positive definite. Moreover, $\mu_k \leq C_1 \|x_k - \hat{x}_k\|^\rho$ by (20). Then (26) implies that (i) holds with some suitable constant $c_4 > 0$.

(ii) Multiplying (24) by $Q_{2,k}^T$, we get

$$(\Lambda_{2,k} + \mu_k I) Q_{2,k}^T \tilde{d}_k + Q_{2,k}^T g_k = 0,$$

where we use $Q_{2,k}^T B_k = \Lambda_{2,k} Q_{2,k}^T$. This implies

$$\begin{aligned}\|Q_{2,k}^T \tilde{d}_k\| &\leq \|(\Lambda_{2,k} + \mu_k I)^{-1} Q_{2,k}^T g_k\| \\ &\leq \|(\Lambda_{2,k} + \mu_k I)^{-1} Q_{2,k}^T (g(x_k) - g(\hat{x}_k) - G_k(x_k - \hat{x}_k))\| \\ &\quad + \|(\Lambda_{2,k} + \mu_k I)^{-1} Q_{2,k}^T B_k (x_k - \hat{x}_k)\| \\ &\quad + \|(\Lambda_{2,k} + \mu_k I)^{-1} Q_{2,k}^T (B_k - G_k)(x_k - \hat{x}_k)\| \\ &\leq \|(\Lambda_{2,k} + \mu_k I)^{-1}\| O(\|x_k - \hat{x}_k\|^2) \\ &\quad + \|(\Lambda_{2,k} + \mu_k I)^{-1} \Lambda_{2,k} Q_{2,k}^T (x_k - \hat{x}_k)\| \\ &\quad + \|(\Lambda_{2,k} + \mu_k I)^{-1}\| O(\|x_k - \hat{x}_k\|^2) \end{aligned} \tag{27}$$

where the second inequality follows from $g(\hat{x}_k) = 0$ and the third inequality follows from $\|Q_{2,k}^T\| \leq 1$, $Q_{2,k}^T B_k = \Lambda_{2,k} Q_{2,k}^T$ and $\|B_k - G_k\| = O(\|d_k^0\|^{\frac{1}{\gamma}}) \leq O(\|g_k\|) = O(\|x_k - \hat{x}_k\|)$. Note that (20) implies

$$\|(\Lambda_{2,k} + \mu_k I)^{-1}\| \leq \mu_k^{-1} \leq c_1^{-1} \|x_k - \hat{x}_k\|^{-\rho}. \tag{28}$$

Moreover, since $\Lambda_{2,k}$ is a diagonal matrix with nonnegative diagonal, we have

$$\|(\Lambda_{2,k} + \mu_k I)^{-1} \Lambda_{2,k}\| \leq 1.$$

Then

$$\begin{aligned}\|(\Lambda_{2,k} + \mu_k I)^{-1} \Lambda_{2,k} Q_{2,k}^T (x_k - \hat{x}_k)\| &\leq \|(\Lambda_{2,k} + \mu_k I)^{-1} \Lambda_{2,k}\| \|Q_{2,k}^T (x_k - \hat{x}_k)\| \\ &\leq \|Q_{2,k}^T (x_k - \hat{x}_k)\| = O(\|x_k - \hat{x}_k\|). \end{aligned}$$

This along with (27), (28) and $0 < \delta \leq 1$ yields

$$\|Q_{2,k}^T \tilde{d}_k\| \leq O(\|x_k - \hat{x}_k\|). \tag{29}$$

By (25), we have

$$\|Q_{1,k}^T \tilde{d}_k\| = \|(\Lambda_{1,k} + \mu_k I)^{-1} Q_{1,k}^T g_k\| \leq \|\Lambda_{1,k}\|^{-1} \|Q_{1,k}^T g_k\| \leq O(\|x_k - \hat{x}_k\|). \tag{30}$$

(29), (30) and $\|\tilde{d}_k\|^2 = \|Q_{1,k}^T \tilde{d}_k\|^2 + \|Q_{2,k}^T \tilde{d}_k\|^2$ imply that (ii) holds. □

Theorem 3.1 *Suppose that Assumption 3.1 holds. For sufficiently large k the iterative formula is as follows*

$$x_{k+1} = x_k + d_k^0 = x_k + \tilde{d}_k \tag{31}$$

and

$$\begin{aligned} &\mathrm{dist}(x_{k+1}, X) = O((\mathrm{dist}(x_k, X))^{1+\rho}), \\ &\lim_{k\to\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^{1+\rho}} < +\infty, \end{aligned}$$

i.e., Algorithm 2.1 converges superlinearly.

Proof We have from (19) that

$$\delta_k^0 = \|g_k\|^\gamma \geq \bar{c}^\gamma (\mathrm{dist}(x_k, X))^\gamma. \tag{32}$$

By Lemma 3.2(iii), we have

$$\|\tilde{d}_k\| \leq c_6 \,\mathrm{dist}(x_k, X). \tag{33}$$

It follows from Assumption 3.1(i) that $\mathrm{dist}(x_k, X) < 1$ for all k sufficiently large. Then by (32), (33) and $\gamma < 1$ we know that $\tilde{d}_k$ is a feasible solution of (5) corresponding to $p = 0$. By Assumption 3.1(iii), we know that (5) is a strictly convex quadratic programming for sufficiently large k. Hence $d_k^0 = \tilde{d}_k$.

Now we prove that $d_k^0 = \tilde{d}_k$ is acceptable. Note that $\gamma < 1$, then $\frac{1}{\gamma} > 1$. Hence

$$\begin{aligned} \mathrm{Pred}_k(d_k^0) - \mathrm{Ared}_k(d_k^0) &= f(x_k) - f(x_k + d_k^0) + g_k^T d_k^0 + \frac{1}{2} {d_k^0}^T B_k d_k^0 + \frac{1}{2}\mu_k \|d_k^0\|^2 \\ &= \frac{1}{2} {d_k^0}^T (B_k - G_k) d_k^0 + O(\|d_k^0\|^{2+\rho}) \\ &= O(\|d_k^0\|^{2+\frac{1}{\gamma}}) + O(\|d_k^0\|^{2+\rho}) = O((\mathrm{dist}(x_k, X)^{2+\rho}), \end{aligned} \tag{34}$$

where the second equality holds because $f(x_k + d_k^0) = f(x_k) + g_k^T d_k^0 + \frac{1}{2} {d_k^0}^T G_k d_k^0 + O(\|d_k^0\|^3)$. $\gamma < 1$ and Lemma 2.2 imply that for k sufficiently large

$$|\mathrm{Pred}_k(d_k^0)| \geq \frac{1}{2}\|g_k\| \min\left\{\|g_k\|^\gamma, \frac{\|g_k\|}{M}\right\} = \frac{1}{2M}\|g_k\|^2 \geq \frac{\bar{c}^2}{2M}(\mathrm{dist}(x_k, X)^2). \tag{35}$$

It follows from (34) and (35) that

$$|r_k^0 - 1| = \frac{|\mathrm{Ared}_k(d_k^0) - \mathrm{Pred}_k(d_k^0)|}{|\mathrm{Pred}_k(d_k^0)|} = O(\mathrm{dist}(x_k, X)^\rho) \to 0.$$

So $r_k^0 > \eta$ for k sufficiently large, i.e., $d_k^0 = \tilde{d}_k$ is acceptable for k sufficiently large.

From (19) we have

$$\begin{aligned}\bar{c} \cdot \operatorname{dist}(x_{k+1}, X) &= \bar{c} \cdot \operatorname{dist}(x_k + \tilde{d}_k, X) \le \|g(x_k + \tilde{d}_k)\| \\ &\le \|g(x_k + \tilde{d}_k) - g_k - G_k \tilde{d}_k\| + \|(G_k - B_k)\tilde{d}_k\| + \mu_k \|\tilde{d}_k\| \\ &\le O(\|\tilde{d}_k\|^2) + O(\|\tilde{d}_k\|^{1+\frac{1}{\gamma}}) + O(\|\tilde{d}_k\|^{1+\rho}) \\ &= O(\|\tilde{d}_k\|^{1+\rho}) = O((\operatorname{dist}(x_k, X))^{1+\rho}).\end{aligned}$$

It then follows from Lemma 3.2(iii) that

$$\|\tilde{d}_{k+1}\| = O(\|\tilde{d}_k\|^{1+\rho}). \tag{36}$$

Since for k sufficiently large, the iteration formula is

$$x_{k+1} = x_k + \tilde{d}_k$$

and x^* is the limit point of $\{x_k\}$, there exists a positive integer $K > 0$ such that

$$x^* = x_k + \sum_{i=k}^{\infty} \tilde{d}_i, \quad \forall k > K.$$

By (36), there exist $a \in (0, 1)$ and a positive integer $\bar{K} > K$ such that for all $k \ge \bar{K}$

$$a\|\tilde{d}_k\| \ge \sum_{i=k+1}^{\infty} \|\tilde{d}_i\|.$$

It then follows from the triangle inequality that for $k \ge \bar{K}$

$$\left\| \sum_{i=k}^{\infty} \tilde{d}_i \right\| \ge \|\tilde{d}_k\| - \left\| \sum_{i=k+1}^{\infty} \tilde{d}_i \right\| \ge (1-a)\|\tilde{d}_k\| \tag{37}$$

and

$$\left\| \sum_{i=k+1}^{\infty} \tilde{d}_i \right\| \le \|\tilde{d}_{k+1}\| + \left\| \sum_{i=k+2}^{\infty} \tilde{d}_i \right\| \le (1+a)\|\tilde{d}_{k+1}\|. \tag{38}$$

So

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^{1+\rho}} \le \frac{(1+a)\|\tilde{d}_{k+1}\|}{(1-a)^{1+\rho}\|\tilde{d}_k\|^{1+\rho}}$$

for all $k \ge \bar{K}$. Since $\|\tilde{d}_{k+1}\| = O(\|\tilde{d}_k\|^{1+\rho})$, we finally have

$$\limsup_{k \to \infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^{1+\rho}} < +\infty,$$

that is, $\{x_k\}$ converges to x^* superlinearly. □

Remark (i) If $\rho = 1$, the algorithm is quadratically convergent.

(ii) If we assume that $\|B_k - G_k\| \to 0$ instead of $\|B_k - G_k\| = O(\|d_k^0\|^{\frac{1}{\gamma}})$, we can prove that $\limsup_{k\to\infty} \frac{\|x_{k+1}-x^*\|}{\|x_k-x^*\|} = 0$. Algorithm 2.1 still converges superlinearly.

4 Numerical Results

In order to see the efficiency of our method, we conducted the numerical experiments on some degenerate problems and some classic test problems from CUTEr, and compared the results by our method with that obtained by the traditional trust region method. Here the traditional trust region method means the trust region method developed by Powell, Moré and Yuan et al. and was described in detail in [2, 12]. At each iterative point, we obtain the trial step by solving subproblem (10) and B_k is chosen as G_k.

The radius of the trust region is updated as follows

$$\Delta_{k+1} = \begin{cases} \dfrac{c_3\|s_k\| + c_4\Delta_k}{2}, & \text{if } r < c_2, \\ \dfrac{(1+c_1)\Delta_k}{2}, & \text{if } r \geq c_2, \end{cases}$$

where $c_1 = 2$, $c_2 = 0.25$, $c_3 = 0.25$, and $c_4 = 0.5$.

In our new trust region method, the parameters are set as $\eta = 10^{-4}$, $\gamma = 0.6$, $\mu_k = \min\{0.01, \|g_k\|^\rho\}$ and $\rho = 0.8$, $c = 0.2$.

All of the algorithms are implemented in Fortran 77, and runs are made on 2.4 GHz PC with 512 M memory. The stopping criterion used is $\|g_k\| < \epsilon$, where $\epsilon = 10^{-6}$. For the convenience of comparison, the subroutine solving the quadratic subproblems is GQTPAR in Minpack and all of the algorithms use the same subroutine to solve the quadratic subproblems.

In Sect. 3, the superlinear convergence of the proposed algorithm has been proved theoretically under the local error bound condition $\|g(x)\| \geq \bar{c} \cdot \mathrm{dist}(x, X)$. In such a case, we concentrate on the local error bound condition. Such a test problem is designed as follows:

$$f(x_1, x_2) = (x_1 - 4x_2)^2.$$

And the solution set is $\{(x_1, x_2) | x_1 - 4x_2 = 0\}$. We set the initial point to $(-5000, 5000)$ and try to seek the minimum. It is easy to verify that the Hessian is singular at any solution, and the local error bound condition is satisfied when $\bar{c} \in [0, \sqrt{34}\,]$. Numerical results indicate that the sequence generated by the algorithm converges to $x^* = (-3.4463797\text{E}{+}03, -8.6159494\text{E}{+}02)$, which is an optimal solution. The iterative step x_k and the norm of gradient at every iteration are recorded in Table 1. These results indicate that our trust region method converges quickly when $\{x_k\}$ approaches the optimal solution.

Next we consider the Powell's singular problem in [8], which is a typical singular test problem,

$$f(x_1, x_2, x_3, x_4) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4.$$

Table 1 Numerical results on designed problem

Iteration k	x_k	$\|g(x_k)\|$
1	$(-0.5000000E+04, 0.5000000E+04)$	0.2061553E+06
2	$(-0.4608029E+04, 0.3432116E+04)$	0.1512066E+06
3	$(-0.4280549E+04, 0.2122195E+04)$	0.1052986E+06
4	$(-0.4014761E+04, 0.1059044E+04)$	0.6803898E+05
5	$(-0.3807774E+04, 0.2310955E+03)$	0.3902236E+05
6	$(-0.3656664E+04, -0.3733458E+03)$	0.1783887E+05
7	$(-0.3570446E+04, -0.7182162E+03)$	0.5752403E+04
8	$(-3.4463797E+03, -8.6159494E+02)$	0.0000000E+00

Table 2 Numerical results on Powell's singular problem

	Iters
Our Method	18
Traditional ($\Delta_0 = 0.01$)	52
Traditional ($\Delta_0 = 1$)	40
Traditional ($\Delta_0 = 100$)	43

And both our method and the traditional method try to seek the minimum $(0, 0, 0, 0)$ (at which the Hessian matrix is not positive definite) started from $(3, -1, 0, 1)$. The results are summarized in Table 2, where iters denotes the number of subproblems solving. The numerical results in Table 2 show that our method has advantage over the traditional method for this problem.

Furthermore, we test the efficiency of our algorithm on some large-scale problems. Numerical experiments are conducted on some problems from CUTEr, which is a famous test environment for constrained and unconstrained optimization [1], and compare the results by our method with that obtained by the traditional trust region method.

The detailed results are summarized in Table 3. Table 3 can be read as follows:

- Column 1 represents the problem name and the problem size or dimension n.
- Columns 2–5 report the numerical results of various algorithms.
- Δ_0 denotes the initial trust region radius for the traditional trust region method.
- In columns 2–5, "#f" denotes the number of calculation of the objective function. "#g" denotes the number of calculation of the gradient. "time" is the runtime in seconds.

From Table 3, one can see that the efficiency of the traditional method depends on the choice of the initial trust region radius. For some problems, the traditional method with large initial trust region radius is efficient. But for other problems, the traditional method with small initial trust region radius becomes more efficient. However, there is no general rule for the choice of the initial trust region radius. Our new trust region method takes a strategy of adjusting the trust region radius self–adaptively, a strategy whose preliminary version is introduced in [15, 16]. In general, the performance of the new method is competitive to that of the traditional trust region method. For

Table 3 Numerical results for the large-scale problems

Test problem	Size	New TRM			Traditional TRM ($\Delta_0 = 0.01$)			Traditional TRM ($\Delta_0 = 1$)			Traditional TRM ($\Delta_0 = 100$)		
		#f	#g	Time	#f	#g	Time	#f	#g	Time	#f	#g	Time
BRYBND	1000	18	12	49.05	24	24	31.63	123	115	152.56	14	10	27.78
CHAINWOO	1000	84	51	141.9	238	179	434.21	356	256	729	283	203	561.98
COSINE	1000	17	14	25.03	23	23	26.38	12	12	16.36	#f > 10000		
CRAGGLVY	1000	15	15	14.19	30	30	29.85	19	19	18.83	16	16	14.12
DIXMAANA	3000	10	10	305.45	30	30	726.94	18	18	470.36	14	13	339.02
DIXMAANB	3000	11	11	1091.57	30	30	1324.09	19	19	1231.34	16	14	1018.42
DIXMAANC	3000	13	11	1222.53	32	31	1694.42	19	19	1342.73	16	14	1032.04
DIXMAAND	3000	27	16	1385.46	31	31	1581.74	20	20	1481.89	15	15	1318.41
EIGENALS	930	255	159	883.81	150	138	414.56	138	126	380.71	138	123	383.35
FREUROTH	1000	12	8	15.19	30	30	34.48	19	19	25.34	13	12	12.84
MANCINO	100	36	11	2.12	31	31	4.78	22	21	3.25	10	10	1.54
NCB20B	1000	18	12	34.89	27	20	56.93	15	8	31.9	22	10	64.59
SENSORS	100	26	14	0.7	41	34	1.5	30	22	1.01	28	16	0.79
SINQUAD	1000	19	11	31.7	29	28	39.24	19	18	30.76	17	13	20.04
SPARSINE	1000	101	39	411.25	31	30	36.83	27	26	33.77	55	44	124.31

problem CHAINWOO, CRAGGLVY, DIXMAANA, DIXMAANB, DIXMAANC, FREUROTH, SENSORS, SINQUAD, the new method performs better than the traditional method with all the three radius. For problem BRYBND, COSINE, DIXMAAND, NCB20B, the new method performs worse than the traditional method with one choice of initial trust region radius but better than the other two choices. These evidences show that the new trust region method is notable in practical computation.

The performance of the new method depends on the choice of parameters μ_k, c and γ. During the experiments, we note that our method is insensitive to c. μ_k should be $\min\{0.01, \|g(x_k)\|^\rho\}$. If we set $\mu_k = \|g(x_k)\|^\rho$, μ_k would be very large and the trial step d_k will be very small in the initial stage of the method. This may result in inefficiency. We note that the theoretical results in Sect. 3 still hold if we set $\mu_k = \min\{0.01, \|g(x_k)\|^\rho\}$. The choice of γ has significant affect on the efficiency of the method. If we set γ too small or too large, the method is not efficient. Generally speak, we set $\gamma \in [0.6, 0.8]$.

5 Conclusion and Discussions

A new trust region method for minimization problems with singular solutions is proposed in this paper. The global convergence is obtained under standard conditions. Moreover, we proved the superlinear convergence without the assumption that the solution of the problem is isolated and the Hessian of the objective function at the solution is nonsingular. This is the first trust region method which possesses this property. Although the superlinear convergence results of the method is for convex minimization problems, it can be extended to nonconvex minimization problems. For nonconvex minimization problem, we need only to assume that $x_k \to x^*$, x^* is a local minimizer of $f(x)$ and $f(x)$ is locally convex at x^*. It is not difficult to establish local superlinear convergence for the trust region method in a way similar to that in Sect. 3. Numerical tests show that the proposed method is not only theoretically improved, but also computationally efficient. Meanwhile, the encouraging theoretic and numerical results suggest some directions for future research, such as incorporating an approximate strategy for trust region subproblem, generalizing the results in this paper to constrained optimization problem, etc.

Acknowledgements The authors would like to give our indebted thanks to the editor and the referees for their valuable comments and suggestions, which help us to improve the quality of the paper significantly.

References

1. Bongartz, I., Conn, A.R., Gould, N.I.M., Toint, Ph.L.: CUTE: constrained and unconstrained testing environment. ACM Trans. Math. Softw. **21**, 123–160 (1995)
2. Conn, A.R., Gould, N.I.M., Toint, P.L.: Trust Region Method. Society for Industrial and Applied Mathematics, Philadelphia (2000)
3. Fan, J.Y., Yuan, Y.: On the quadratic convergence of the Levenberg–Marquadt method without nonsingularity assumption. Computing **74**, 23–39 (2005)
4. Fletcher, R.: Practical Methods of Optimizations. Wiley, New York (1987)
5. Li, D.H., Fukushima, M., Qi, L., Yamashita, N.: Regularized Newton methods for convex minimization problems with singular solutions. Working paper, Departerment of Applied Mathematics and Physics, Kyoto University, February 2002
6. Moré, J.J.: Recent developments in algorithm and software for trust region methods. In: Bachem, A., Grötschel, M., Kort, B. (eds.) Mathematical Programming: The State of Art, pp. 258–287. Springer, Berlin (1983)
7. Nocedal, J., Wright, S.J.: Numerical Optimization. Springer, New York (1999)
8. Powell, M.J.D.: An iterative method for finding stationary values of a function of several variables. Comput. J. **5**(2), 147–151 (1962)
9. Stewart, G.W., Sun, J.G.: Matrix Perturbation Theory. Academic, New York (1990)
10. Yamashita, N., Fukushima, M.: On the rate of convergence of the Levenberg–Marquadt method. Computing **15**, 239–249 (2001)
11. Yuan, Y., Sun, W.: Optimization Theory and Algorithm. Scientific Publisher House, China (1997)
12. Yuan, Y.Y.: On the convergence of trust region algorithms. Math. Numer. Sin. **16**(3), 333–346 (1996)
13. Zhang, J.L., Wang, Y.: A new trust region method for nonlinear equation. Math. Methods Oper. Res. **58**, 283–298 (2003)
14. Zhang, J.L.: On the convergence properties of Levenberg–Marquadt method. Optimization **52**, 739–756 (2003)
15. Zhang, X.S., Zhang, J.L., Liao, L.Z.: An adaptive trust region method and its convergence. Science in China (Ser. A) **45**(5), 620–631 (2002)
16. Zhang, X.S., Chen, Z.W., Zhang, J.L.: A self-adaptive trust region method for unconstrained optimization. OR Trans. **5**, 53–62 (2001)