AIMS Mathematics

http://www.aimspress.com/journal/Math

AIMS Mathematics, 8(9): 21161–21175.
DOI:10.3934/math.20231078
Received: 30 March 2023
Revised: 30 May 2023
Accepted: 04 June 2023
Published: 03 July 2023

*Research article*

# A new Newton method for convex optimization problems with singular Hessian matrices

**Tianji Wang and Qingdao Huang***

School of Mathematics, Jilin University, Changchun 130012, China

* **Correspondence:** Email: huangqd@jlu.edu.cn.

**Abstract:** In this paper, we propose a new Newton method for minimizing convex optimization problems with singular Hessian matrices including the special case that the Hessian matrix of the objective function is singular at any iteration point. The new method we proposed has some updates in the regularized parameter and the search direction. The step size of our method can be obtained by using Armijo backtracking line search. We also prove that the new method has global convergence. Some numerical experimental results show that the new method performs well for solving convex optimization problems whose Hessian matrices of the objective functions are singular everywhere.

**Keywords:** convex optimization; regularized Newton method; singular Hessian matrices; global convergence
**Mathematics Subject Classification:** 49M15, 90C25

## 1. Introduction

We consider the convex optimization problem as follows:

$$\min_{x\in R^n} f(x), \tag{1.1}$$

where $f : R^n \to R$ is twice continuously differentiable. We denote the gradient and the Hessian matrix of $f$ at $x$ by $g(x) = \nabla f(x)$ and $G(x) = \nabla^2 f(x)$, respectively. $\|\cdot\|$ denotes the Euclidean norm. The function $f(x)$ is convex and differentiable, so if

$$g(x^*) = 0, \quad x^* \in R^n, \tag{1.2}$$

then, $x^*$ will be a solution of (1.1).

In order to solve (1.1), there are many different kinds of methods [1–8]. Almost all of these methods use line search to get a sequence $\{x_k\}$:

$$x_{k+1} = x_k + \alpha_k d_k,$$

where $\alpha_k > 0$ is the step size, $d_k$ is the search direction and it is descent, that is to say $f(x_{k+1}) < f(x_k)$.

To simplify the notation, in this paper, we set $g_k = g(x_k)$ and $G_k = G(x_k)$ which denote the gradient and the Hessian matrix of $f$ at $x_k$, respectively. Similarly, the ideas of some optimization methods can also be used to solve systems of nonlinear equations, such as the cubic Newton-like method [9], Chebyshev-like method [10,11] and high order hybrid method [12]. These methods are used to solve systems of nonlinear equations by improving the original optimization methods. Therefore, the application and promotion of all kinds of classical optimization methods are very extensive. Among these efficient optimization methods, the classical Newton method is famous for its quadratic convergence property, but when we use classical Newton method to solve convex optimization problems, it has to satisfy the condition that the Hessian matrix of the objective function is positive definite. However, the Hessian matrix of the objective function is not always positive definite. Especially, when $G_k$ is not positive definite but it is nonsingular, the Newton search direction $d_k = -G_k^{-1}g_k$ may not be a descent direction, then we use $-d_k$ as the search direction instead. But when $G_k$ is singular, we could not calculate $G_k^{-1}$, so the search direction is not well defined. That is to say, if the Hessian matrices are singular, the classical Newton method may not work well. It is too difficult to ensure Hessian matrices of $f(x)$ are nonsingular everywhere. So, how to solve the optimization problem with singular Hessian matrix is very important. In particular, when the Hessian matrices of the objective function $f(x)$ are singular at any iteration point, such as $f(x) = \frac{1}{2}\sum_{i=1}^{n-1}(x_i - x_{i+1})^2 + \frac{1}{12}\sum_{i=1}^{n-1}\alpha_i(x_i - x_{i+1})^4$, research on how to quickly and effectively solve the minimum value of this kind of special function is not common.

In the following chapters, this paper will put forward a new Newton method to solve this kind of optimization problem with special objective function. For the cases when $G_k$ might be positive semidefinite, there are many modifications of the classical Newton method which can solve optimization problems effectively and satisfy global convergence as well [13–16]. Goldfeld et al. [17] first proposed a modified Newton method to solve some optimization problems when their Hessian matrices are not positive definite. The framework of the method is as follows:

**Algorithm 1.1**
**step 1:** Set $\overline{G}_k = G_k + \nu_k I$, if $G_k$ is positive definite, $\nu_k = 0$; otherwise, $\nu_k \neq 0$.
**step 2:** Compute the Cholesky factorization of $\overline{G}_k$, $\overline{G}_k = L_k D_k L_k^{\mathrm{T}}$.
**step 3:** Compute $d_k$ by $\overline{G}_k d_k = -g_k$.
**step 4:** Set $x_{k+1} = x_k + d_k$.

In Algorithm 1.1, they add a new parameter $\nu_k$. In step 1, they set $\nu_k > 0$, but it could not be too big because it should be a little bit bigger than the absolute value of the minimum negative eigenvalue of $G_k$. In step 2, $\overline{G}_k$ is a symmetric positive definite matrix, so it has a Cholesky factorization $\overline{G}_k = L_k D_k L_k^{\mathrm{T}}$, where $L_k$ is a lower triangular matrix with diagonal identity and $D_k$ is a diagonal matrix. How to determine the value of $\nu_k$ is the key to this method. Gill and Murray [18] proposed a modified Cholesky factorization algorithm to get $\nu_k$. They set $\nu_k = \min\{b_1, b_2\}$, where $b_1, b_2$ are two upper bounds on $\nu_k$, and $b_1$, $b_2$ can be calculated as follows. In order to get $b_1$, they apply the modified algorithm to $G_k$, and they have $G_k + E = L_k D_k L_k^{\mathrm{T}}$. $E$ is a diagonal matrix and it has positive diagonal elements. If $G_k$ is positive definite, $E = 0$ and they set $\nu_k = 0$, otherwise, they compute an upper bound on $\nu_k$ by the Gerschgorin disk theorem, which is defined as follows:

$$b_1 = |\min_{1\le i\le n}\{(G_k)_{ii} - \sum_{j\neq i}|(G_k)_{ij}|\}| \geq |\min_i \lambda_i|,$$

where $\lambda_i$ is the eigenvalue of $G_k$. Besides that, they let $b_2 = \max_i\{e_{ii}\}$, where $e_{ii}$ is the diagonal element of $E$ and $b_2$ is an upper bound on $\nu_k$. The new Cholesky factorization algorithm can also work well when $G_k$ is not positive definite.

In order to deal with more cases when $G_k$ is a positive semidefinite matrix, in addition to the Cholesky factorization method described above, more and more people are trying to use the regularization method. Classical Tickonov regularization of the cost function is a common regularization method. The essence of classical Tickonov regularization is to transform the singular covariance matrix inversion into the non-singular matrix inversion by adding a very small perturbation $\lambda > 0$ to each diagonal element of the covariance matrix of the non-full rank matrix $A$, so as to greatly improve the numerical stability of solving the non-full rank matrix $Ax = b$, where $b$ is a vector, the perturbation $\lambda$ is also called the regularization parameter. This regularization idea can also be applied to the Newton method for solving convex optimization problems when $G_k$ is positive semidefinite. In recent years, the regularized Newton method is another more common modified Newton method which adds a regularized parameter $\mu_k > 0$ to avoid the case when $G_k$ is singular [19–22]. The search direction $d_k$ can be obtained by

$$(G_k + \mu_k I)d_k = -g_k. \tag{1.3}$$

Recently, there are many different kinds of definitions of $\mu_k$. In [23] Polyak proposed a regularized Newton method with $\mu_k = \|g_k\|$. Ueda and Yamashita also proposed a regularized Newton method [24], they set

$$\mu_k = \Lambda_k + c\|g_k\|^\delta, \quad \Lambda_k = \max\{0, -\lambda_{\min}(\nabla^2 f(x_k))\},$$

where $c > 0$ is a positive constant, $\delta \geq 0$ and $\lambda_{\min}(\nabla^2 f(x_k))$ is the minimal eigenvalue of the square matrix $\nabla^2 f(x_k)$. Besides that, they [25] proposed an adaptive regularized Newton method in which

$$\mu_k = \max\{0, -\lambda_{\min}(\nabla^2 f(x_k))\} + \sigma_k\|g_k\|^\delta,$$

where $\sigma_k > 0$ is defined as the adaptive regularized parameter and $\delta \geq 0$.

In order to solve the convex optimization with singular Hessian matrices more quickly and effectively, there are also some updates in the search direction $d_k$. For some given definitions of $\mu_k$, normally, the search direction can be obtained by

$$d_k = -(G_k + \mu_k I)^{-1} g_k. \tag{1.4}$$

Fan and Yuan [26] proposed a regularized Newton method which can be applied to solve convex optimization problems with singular Hessian matrices, and they proved the method had quadratic convergence. In their method, the modification of $d_k$ is crucial. The search direction of the classical Newton method is $d_k^N = -G_k^{-1} g_k$, and if $G_k$ is singular we may need to compute the Moore-Penrose step $d_k^{MP} = -G_k^{+} g_k$, which is the solution of $\min_d \|g_k + G_k d\|^2$. The $G_k^{+}$ can be obtained by singular value decomposition, but sometimes it is inhibitory. In order not to compute $d_k^{MP}$, Fan and Yuan first give the regularized parameter $\mu_k = \sigma_k\|g_k\|$ where $\sigma_k > 0$ is an adaptive regularized parameter and $d_k$ is the solution of $(G_k + \mu_k I)d = -g_k$. The parameter $\mu_k$ makes an error between $d_k$ and $d_k^{MP}$, so they replace $-g_k$ by $-g_k + \mu_k d_k^{MP}$. Then, they have $(G_k + \mu_k I)d = -g_k + \mu_k d_k^{MP}$, and $d_k^{MP}$ is a solution of it. Finally, they replace the $d_k^{MP}$ with $d_k$, which is the best approximation they can obtain. So they have $(G_k + \mu_k I)d = -g_k + \mu_k d_k$, the solution of it is a correction of $d_k$, which is defined by $s_k$. They set

$$s_k = d_k + \tilde{d}_k, \quad \tilde{d}_k = (G_k + \mu_k I)^{-1} \mu_k d_k. \tag{1.5}$$

So, $s_k$ is closer to $d_k^{MP}$ than $d_k$.

In [27], Zhou and Chen proposed a regularized Newton method and they proved the cubic convergence of it under the constraint of local error. In their method, they also modified the search direction $d_k$, and they set $s_k = d_k + \hat{d}_k$, where $\hat{d}_k$ is the solution of

$$(G_k + \mu_k I)d = -g(y_k), \quad y_k = x_k + d_k, \quad \mu_k = \sigma_k \|g_k\|, \tag{1.6}$$

where $\sigma_k > 0$ is also an adaptive regularized parameter.

The similarity between the methods in the above two articles is that they all update the iteration by $x_{k+1} = x_k + s_k$ without line search. Li et al. [28] proposed two inexact regularized Newton methods which have local quadratic convergence under a local error bound condition, the search directions of two methods are different, one of them is that

$$(G_k + \mu_k I)d_k + g_k = r_k, \quad \|r_k\| = O(\|g_k\|^2), \quad \mu_k = C\|g_k\|, \tag{1.7}$$

another is that

$$(G_k + \mu_k I)d_k + g_k = r_k, \quad \|r_k\| \leq \alpha\mu_k\|d_k\|, \quad \mu_k = C\|g_k\|, \tag{1.8}$$

where $\alpha \in (0, 1)$, $C > 0$. It is clear that the $d_k$ of (1.7) and (1.8) are not exactly the same as (1.4), but when $r_k = 0$, they are the same.

In this paper, we propose a new Newton method to solve some special convex optimization problems whose Hessian matrices of the objective functions are singular everywhere. To begin with, we define a new regularized parameter $\mu_k = \sigma_k\|g_k\|^\delta$, where $\sigma_k > 0$ is an adaptive regularized parameter and $0 < \delta \leq 1$. As for the modification of search direction, we will make sure it is a descent direction and when the Hessian matrices of $f(x)$ are singular it can also work well. In this case, we define a new search direction as follows,

$$\bar{d}_k = (G_k + \mu_k I)^{-1}\nu\mu_k^2 d_k, \tag{1.9}$$

where $d_k$ is the same as (1.4). And in order to obtain satisfactory results, we add the parameter $\nu > 0$ is a constant. Different from that in [26] and [27], we use Armijo backtracking line search along $\bar{d}_k$ to get the step size $\alpha_k$. In a large number of numerical experiments, we can see that the calculation result is better when $10 \leq \nu \leq 50$. The new search direction $\bar{d}_k$ allows our method to maintain the fast convergence of classical Newton method and by the numerical experiments we can see that it will increase computational efficiency.

The paper is organized as follows: In Section 2, we introduce the new Newton method in detail. In Section 3, we present the global convergence analysis of the algorithm. We show some numerical experimental results in Section 4. In the end, we will give the conclusion of this paper in Section 5.

## 2. The algorithm

In this section, we give the new Newton method. As we know, $f(x)$ is convex and $G_k$ is the Hessian matrix of $f(x)$, $G_k$ is symmetric positive semidefinite. Obviously, $G_k + \mu_k I$ is positive definite. Firstly, we can get $d_k$ by solving

$$(G_k + \mu_k I)d_k = -g_k, \tag{2.1}$$

and by (1.9) we know that $\bar{d}_k$ is the solution of

$$(G_k + \mu_k I)d = \nu\mu_k^2 d_k. \tag{2.2}$$

Then, we have

$$\bar{d}_k^{\mathrm{T}} g_k = -\nu g_k^{\mathrm{T}} (G_k + \mu_k I)^{-2} \mu_k^2 g_k < 0, \tag{2.3}$$

where $\bar{d}_k$ is the same as (1.9) and we can see that $\bar{d}_k$ is always a descent direction for $f(x_k)$ at $x_k$. We use the Armijo backtracking line search to obtain $\alpha_k$. The adaptive parameter $\sigma_k$ will be updated by using the trust region method like that in [21] which thinking of $\sigma_k$ as the trust region radius, and the parameters $\gamma_1, \gamma_2$ are used to adjust the reduction and expansion of the trust region radius.

Next, we will give the algorithm of our new method.

**Algorithm 2.1**

**step 0:** Given $x_0 \in R^n, \sigma_0 \geq \sigma_{\min},\ 0 < \delta \leq 1,\ 0 < \beta < 1,\ 0 < \gamma_1 < 1 < \gamma_2,\ 0 < \eta_1 < \eta_2 \leq 1,\ 0 < \eta < 1,\ \sigma_{\min} > 0, 0 < \tau \leq 1, \nu > 0$. Set $\mu_0 = \sigma_0 \|g_0\|^\delta$ and $k = 0$.

**step 1:** If $\|g_k\| = 0$, stop. Otherwise, compute $d_k$ and $\bar{d}_k$ by solving $(G_k + \mu_k I) d_k = -g_k$, $(G_k + \mu_k I)\bar{d}_k = \nu \mu_k^2 d_k$ successively.

**step 2:** Execute Armijo backtracking line search along $\bar{d}_k$. Compute $q_k$ by

$$q_k = \begin{cases} f(x_k), & if\ k = 0; \\ \tau f(x_k) + (1-\tau) q_{k-1}, & if\ k \geq 1. \end{cases} \tag{2.4}$$

Set $i = 0, \alpha_k^{(i)} = 1$.

**while** $f(x_k + \alpha_k^{(i)} \bar{d}_k) > q_k + \eta \alpha_k^{(i)} g_k^{\mathrm{T}} \bar{d}_k$

$\alpha_k^{(i+1)} = \beta \alpha_k^{(i)}$

$i = i + 1$

**end while**.

Set $\alpha_k = \alpha_k^{(i)}, x_{k+1} = x_k + \alpha_k \bar{d}_k$.

**step 3:** Compute $\mu_{k+1}$ as

$$\sigma_{k+1} = \begin{cases} \max\{\sigma_{\min}, \gamma_1 \sigma_k\}, & if\ \alpha_k \geq \eta_2; \\ \sigma_k, & if\ \eta_1 \leq \alpha_k < \eta_2; \\ \gamma_2 \sigma_k, & if\ \alpha_k < \eta_1. \end{cases} \tag{2.5}$$

And $\mu_{k+1} = \sigma_{k+1} \|g_{k+1}\|^\delta$.

Set $k = k + 1$, and go to step 1.

## 3. Convergence analysis

In this section, we suppose that the gradient of $f(x)$ at the initial point $x_0$ exists, that is to say $g_0 \neq 0$. We first give two assumptions.

**Assumption 3.1.** By Algorithm 2.1, we can get the set $\{x_k\}$ which is contained in a compact set $\Omega \subset R^n$.

**Assumption 3.2.** The Hessian matrix $G_k$ satisfies $\|G_k\| \leq m_G$, $m_G > 0$.

**Lemma 3.1.** *Let the set $\{x_k\}$ is infinite and satisfy Assumption 3.1. Suppose that $\|g_k\| > 0$ for all $k \geq 0$. Then,*

$$q_k \geq f(x_k)\ and\ q_k > q_{k+1},\ k \geq 0. \tag{3.1}$$

*Proof.* We use induction to prove this lemma.

By (2.4) we know $q_0 = f(x_0)$. From the Armijo backtracking line search rule, $f(x_1) = f(x_0+\alpha_0\bar{d}_0) \leq f(x_0) + \eta\alpha_0 g_0^{\mathrm{T}}\bar{d}_0 < f(x_0) = q_0$. In addition, $q_0 - q_1 = \tau[q_0 - f(x_1)] > 0$. So, when $k = 0$, the conclusion of this lemma holds.

Next, we assume that $q_{k-1} \geq f(x_{k-1})$, and $q_{k-1} > q_k$ hold for some $k > 1$.

As we know, $\bar{d}_{k-1}$ is a descent direction and $q_{k-1} \geq f(x_{k-1})$, so in Algorithm 2.1, the linear search rule is well defined. Hence, $f(x_k) = f(x_{k-1} + \alpha_{k-1}\bar{d}_{k-1}) \leq q_{k-1} + \eta\alpha_{k-1}g_{k-1}^{\mathrm{T}}\bar{d}_{k-1} < q_{k-1}$. By the definition of $q_k$, $q_k - f(x_k) = (1-\tau)[q_{k-1} - f(x_k)] \geq 0$. Since $q_k \geq f(x_k)$, we have $f(x_{k+1}) < q_k$ by the line search rule. Therefore, $q_k - q_{k+1} = \tau[q_k - f(x_{k+1})] > 0$. □

The regularized parameter $\mu_k = \sigma_k\|g_k\|^{\delta}$, then if $\|g_k\| > 0$, $G_k + \mu_k I$ is positive definite. Then, we will give both an upper bound and a lower bound of $-g_k^{\mathrm{T}}\bar{d}_k$ in the following lemma.

**Lemma 3.2.** *Suppose Assumption 3.2 holds. If* $\|g_k\| > 0$*, then we have*

$$-g_k^{\mathrm{T}}\bar{d}_k \geq \frac{\nu\mu_k^2\|g_k\|^2}{(m_G + \mu_k)^2}, \tag{3.2}$$

and

$$-g_k^{\mathrm{T}}\bar{d}_k \leq \nu\|g_k\|^2. \tag{3.3}$$

*Proof.* By the definition of $d_k$ and $\bar{d}_k$, we have

$$d_k = -(G_k + \mu_k I)^{-1}g_k \tag{3.4}$$

and

$$\bar{d}_k = -(G_k + \mu_k I)^{-2}\nu\mu_k^2 g_k. \tag{3.5}$$

In order to complete the proof, we just do the calculation.

$$\begin{aligned} -g_k^{\mathrm{T}}\bar{d}_k &= \nu g_k^{\mathrm{T}}(G_k + \mu_k I)^{-2}\mu_k^2 g_k \\ &\geq \frac{\nu\mu_k^2\|g_k\|^2}{\lambda_{\max}^2(G_k + \mu_k I)} \\ &= \frac{\nu\mu_k^2\|g_k\|^2}{[\lambda_{\max}(G_k) + \mu_k]^2}. \end{aligned}$$

Since $\|G_k\| \leq m_G$, the lower bound has been obtained.

Next, we prove the upper bound.

$$\begin{aligned} -g_k^{\mathrm{T}}\bar{d}_k &= g_k^{\mathrm{T}}(G_k + \mu_k I)^{-2}\mu_k^2 g_k \\ &\leq \frac{\nu\mu_k^2\|g_k\|^2}{\lambda_{\min}^2(G_k + \mu_k I)} \\ &= \frac{\nu\mu_k^2\|g_k\|^2}{[\lambda_{\min}(G_k) + \mu_k]^2} \\ &\leq \nu\|g_k\|^2. \end{aligned}$$

Since $G_k$ is positive semidefinite, we have $\lambda_{\min}(G_k) \geq 0$. This finishes the proof. □

As for $\|\bar{d}_k\|$, we will give an upper bound of it in the next lemma.

**Lemma 3.3.** *If* $\|g_k\| > 0$ *for all* $k \geq 0$*, then,*

$$\|\bar{d}_k\| \leq \nu \|g_k\|. \tag{3.6}$$

*Proof.* By the definition of $\bar{d}_k$, we have

$$\begin{aligned}\|\bar{d}_k\| &= \|-(G_k + \mu_k I)^{-2} \nu \mu_k^2 g_k\| \\ &\leq \frac{\nu \mu_k^2 \|g_k\|}{\lambda_{\min}^2 (G_k + \mu_k I)} \\ &= \frac{\nu \mu_k^2 \|g_k\|}{[\lambda_{\min}(G_k) + \mu_k]^2}.\end{aligned}$$

Since $G_k$ is positive semidefinite,

$$\|\bar{d}_k\| \leq \nu \|g_k\|,$$

so we can prove (3.6). □

In addition to the upper bound of $\|\bar{d}_k\|$, we give the lower bound of $\alpha_k$ in next lemma.

**Lemma 3.4.** *Suppose Assumption 3.2 holds, and we further suppose that* $g(x)$ *is Lipschitz continuous which satisfies:*

$$\|g(y) - g(x)\| \leq L_g \|y - x\|, \qquad \forall x, y \in R^n,$$

where $L_g$ is the Lipschitz constant. Then, there exists $c = \frac{2\beta(1-\eta)}{\nu L_g}$ such that

$$\alpha_k \geq \min\left\{1, c\left(\frac{\mu_k}{m_G + \mu_k}\right)^2\right\}. \tag{3.7}$$

*Proof.* If $\alpha_k = 1$, then (3.7) holds.

Now, we consider the case that $\alpha_k < 1$. According to the Armijo backtracking line search rule in Algorithm 2.1, we know that

$$f(x_k + \frac{\alpha_k}{\beta}\bar{d}_k) > q_k + \eta \frac{\alpha_k}{\beta} g_k^{\mathrm{T}} \bar{d}_k \geq f(x_k) + \eta \frac{\alpha_k}{\beta} g_k^{\mathrm{T}} \bar{d}_k. \tag{3.8}$$

Since $g_k$ is Lipschitz continuous, then,

$$f(x_k + \frac{\alpha_k}{\beta}\bar{d}_k) \leq f(x_k) + \frac{\alpha_k}{\beta} g_k^{\mathrm{T}} \bar{d}_k + \frac{1}{2}(\frac{\alpha_k}{\beta})^2 L_g \|\bar{d}_k\|^2. \tag{3.9}$$

By (3.8) and (3.9),

$$\frac{1}{2}\frac{\alpha_k}{\beta} > -\frac{(1-\eta) g_k^{\mathrm{T}} \bar{d}_k}{L_g \|\bar{d}_k\|^2}.$$

Therefore,

$$\begin{aligned}\alpha_k &> \frac{2\beta(1-\eta)}{L_g}\cdot\frac{-g_k^{\mathrm{T}}\bar{d}_k}{\|\bar{d}_k\|^2}\\ &\geq \frac{2\beta(1-\eta)}{L_g}\cdot\frac{\nu\mu_k^2\|g_k\|^2}{(m_G+\mu_k)^2}\cdot\frac{1}{\nu^2\|g_k\|^2}\\ &= \frac{2\beta(1-\eta)}{\nu L_g}\cdot\left(\frac{\mu_k}{m_G+\mu_k}\right)^2.\end{aligned}$$

We use (3.2) and (3.6) in the second inequality. Let $c=\frac{2\beta(1-\eta)}{\nu L_g}$, then we can complete the proof. □

**Theorem 3.1.** Suppose Assumption 3.1 and Assumption 3.2 hold, if $\{f(x_k)\}$ has a lower bound and there exists $c>0$ such that (3.7) holds, then,

$$\liminf_{k\to\infty}\|g_k\|=0. \tag{3.10}$$

*Proof.* According to Algorithm 2.1, we have

$$q_k - f(x_{k+1}) \geq \eta\alpha_k(-g_k^{\mathrm{T}}\bar{d}_k). \tag{3.11}$$

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

$$q_k - f(x_{k+1}) \geq \frac{\eta\alpha_k\nu\mu_k^2\|g_k\|^2}{(m_G+\mu_k)^2} \geq \frac{c\eta\nu\mu_k^4\|g_k\|^2}{(m_G+\mu_k)^4}. \tag{3.12}$$

$$\sum_{k=0}^{+\infty}(q_k-q_{k+1}) = \sum_{k=0}^{+\infty}\tau[q_k-f(x_{k+1})] \geq \sum_{k=0}^{+\infty}\frac{c\tau\eta\nu\mu_k^4\|g_k\|^2}{(m_G+\mu_k)^4}. \tag{3.13}$$

Since $\{f(x_k)\}$ has a lower bound and we can know from (2.3) that $q_k \geq f(x_k)$, $q_k > q_{k+1}$, therefore, $\{q_k\}$ also has a lower bound and the first term in (3.13) is convergent. Thus, we have

$$\sum_{k=0}^{+\infty}\frac{\mu_k^4\|g_k\|^2}{(m_G+\mu_k)^4} < \infty. \tag{3.14}$$

**Case (i)** If

$$\liminf_{k\to\infty}\|g_k\|=0, \tag{3.15}$$

then, we can complete the proof.
**Case (ii)** If there exists $\epsilon>0$ such that $\|g_k\|\geq\epsilon$, then, we have

$$\lim_{k\to\infty}\mu_k = 0. \tag{3.16}$$

From Algorithm 2.1, we know $\sigma_k \geq \sigma_{\min} > 0$, so, it contradicts (3.16).

Above all, we can prove this theorem. □

## 4. Numerical experimental results

In this section, we give some numerical experimental results on comparisons of the new Newton method, the adaptive regularized Newton method (ARNM), a common form of regularized Newton method (RNM) and the method proposed by Fan and Yuan in [26]. In Fan and Yuan's method, they use (1.5) as the search direction in their algorithm, which is similar to our algorithm, so we compare it with Algorithm 2.1 and call it FY in subsequent paper.

Now, we give the ARNM and the RNM.

**Algorithm 4.1: ARNM**

**step 0:** Let $\delta = 0.25$, $\beta = 0.5$, $\gamma_1 = 0.5$, $\gamma_2 = 2$, $\eta_1 = 0.25$, $\eta_2 = 0.75$, $\eta = 0.5$, $\sigma_{min} = 0.5$, $\tau = 0.5$. Given $x_0 \in R^n$, $\sigma_0 \geq \sigma_{min}$, set $\mu_0 = \sigma_0 \|g_0\|^\delta$ and $k = 0$.

**step 1:** If $\|g_k\| = 0$, stop. Otherwise, compute $d_k$ by solving $(G_k + \mu_k I)d_k = -g_k$.

**step 2:** Execute Armijo backtracking line search along $d_k$. Compute $q_k$ by

$$q_k = \begin{cases} f(x_k), & if\ k = 0; \\ \tau f(x_k) + (1-\tau) q_{k-1}, & if\ k \geq 1. \end{cases} \tag{4.1}$$

Set $i = 0, \alpha_k^{(i)} = 1$.

**while** $f(x_k + \alpha_k^{(i)} d_k) > q_k + \eta \alpha_k^{(i)} g_k^{\mathrm{T}} d_k$

$\alpha_k^{(i+1)} = \beta \alpha_k^{(i)}$

$i = i + 1$

**end while**.

Set $\alpha_k = \alpha_k^{(i)}, x_{k+1} = x_k + \alpha_k d_k$.

**step 3:** Update $\sigma_k$. Let

$$\sigma_{k+1} = \begin{cases} \max\{\sigma_{\min}, \gamma_1 \sigma_k\}, & if\ \alpha_k \geq \eta_2; \\ \sigma_k, & if\ \eta_1 \leq \alpha_k < \eta_2; \\ \gamma_2 \sigma_k, & if\ \alpha_k < \eta_1. \end{cases} \tag{4.2}$$

And $\mu_{k+1} = \sigma_{k+1} \|g_{k+1}\|^\delta$.

Set $k = k + 1$, and go to step 1.

**Algorithm 4.2: RNM**

**step 0:** Let parameters $\delta = 0.55$, $\sigma = 0.4$, $\tau = 0$. Given $x_0 \in R^n$, set $\mu_k = \|g_k\|^{1+\tau}$ and $k = 0$.

**step 1:** If $\|g_k\| = 0$, stop. Otherwise, compute $d_k$ by solving $(G_k + \mu_k I)d_k = -g_k$.

**step 2:** Find the smallest nonnegative integer $m$ such that

$$f(x_k + \delta^m d_k) \leq f(x_k) + \sigma \delta^m g_k^{\mathrm{T}} d_k.$$

Let $m_k = m$, $\alpha_k = \delta^{m_k}$ and $x_{k+1} = x_k + \alpha_k d_k$.

**step 3:** Set $k = k + 1$, and go to step 1.

We test all algorithms on the unconstrained nonlinear optimization problem which can be found in [28]. The function is:

$$f(x) = \frac{1}{2} \sum_{i=1}^{n-1} (x_i - x_{i+1})^2 + \frac{1}{12} \sum_{i=1}^{n-1} \alpha_i (x_i - x_{i+1})^4, \tag{4.3}$$

where $\alpha_i \geq 0$, $(i = 1, \ldots, n-1)$ are constants. We choose different values of $x_0$, $\alpha_i$ and $n$ to test all the algorithms. Clearly, $f(x)$ is convex and the minimum set of it is

$$X = \{x \in R^n | x_1 = x_2 = \cdots = x_n\}.$$

The Hessian matrix of $f(x)$ is as follows:

$$\nabla^2 f(x) = \begin{bmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{bmatrix} + \begin{bmatrix} a_1 & -a_1 & & & \\ -a_1 & a_1 + a_2 & -a_2 & & \\ & \ddots & \ddots & \ddots & \\ & & -a_{n-2} & a_{n-2} + a_{n-1} & -a_{n-1} \\ & & & -a_{n-1} & a_{n-1} \end{bmatrix}$$

where $a_i = \alpha_i(x_i - x_{i+1})^2$, $(i = 1, \ldots, n-1)$ are symmetric semidefinite for all $x$. Besides, the matrix $\nabla^2 f(x)$ is singular for any $x$ because the sum of each of its columns is zero.

All programs are written in Matlab. When comparing with the ARNM and the RNM, for Algorithm 2.1, we give the values of each parameter as follows: $\delta = 0.25, \beta = 0.5$, $\gamma_1 = 0.5$, $\gamma_2 = 2$, $\eta_1 = 0.25$, $\eta_2 = 0.75$, $\eta = 0.5$, $\sigma_{\min} = 0.5$, $\tau = 0.5$, $\nu = 37$. When compared with Fan and Yuan's method (FY), we adjust the parameter values appropriately, we set $\delta = 0.5, \beta = 0.5$, $\gamma_1 = 0.5$, $\gamma_2 = 2$, $\eta_1 = 0.25$, $\eta_2 = 0.75$, $\eta = 0.5$, $\sigma_{\min} = 0.25$, $\tau = 0.5$, $\nu = 25$. For the FY, the values of all parameters are the same as that in [23]. For the test function, we choose $n = 10, 20, 30, \ldots, 100, 500, 1000$, and for every $n$ we set $\alpha_i = 0, 1, i$, $(i = 1, \ldots, n-1)$. While the parameter $n$ and $\alpha_i$ are selected, we set $(x_0)_i = i, n-i, 1/i$, $(i = 1, \ldots, n-1)$.

In order to compare the performance of these algorithms, we use the same distribution function as that in [29]. We use $S$ to represent a set of solutions, and $P_s$ is a set of problems which can be solved by the solutions in $S$. Besides, a measure for evaluation required to solve a problem $p$ by a solution $s$ is denoted by $t_{p,s}$, and for each $p$ we have $t_p^* = \min\{t_{p,s} | s \in S\}$. The distribution function is defined as follows:

$$F_s^S(\rho) = \frac{|\{p \in P_s | t_{p,s} \leq \rho t_p^*\}|}{|P_s|}, \rho \geq 1. \tag{4.4}$$

In the analysis of our numerical experimental results, $S$ represents the set of all the algorithms and $s \in S$ represents one of these algorithms. $P_s$ is a set of all convex optimization problems that need to be solved and $p \in P$ is one of these optimization problems. The measure for evaluation $t_{p,s}$ can be the number of times the objective function is solved when solving the optimization problem or the number of iterations, CPU time and so on.

Figure 1 shows the comparison of Algorithm 2.1, the ARNM and the RNM for $n_f$ and $n_l$, where $n_f$ denotes the number of times the objective function is solved and $n_l$ denotes the number of iterations. Figure 2 shows the comparison of Algorithm 2.1, the ARNM and the RNM for $n_t$, where $n_t$ denotes the CPU time. Figure 3 shows the comparison of Algorithm 2.1 and the FY for $n_l$ and $n_t$.

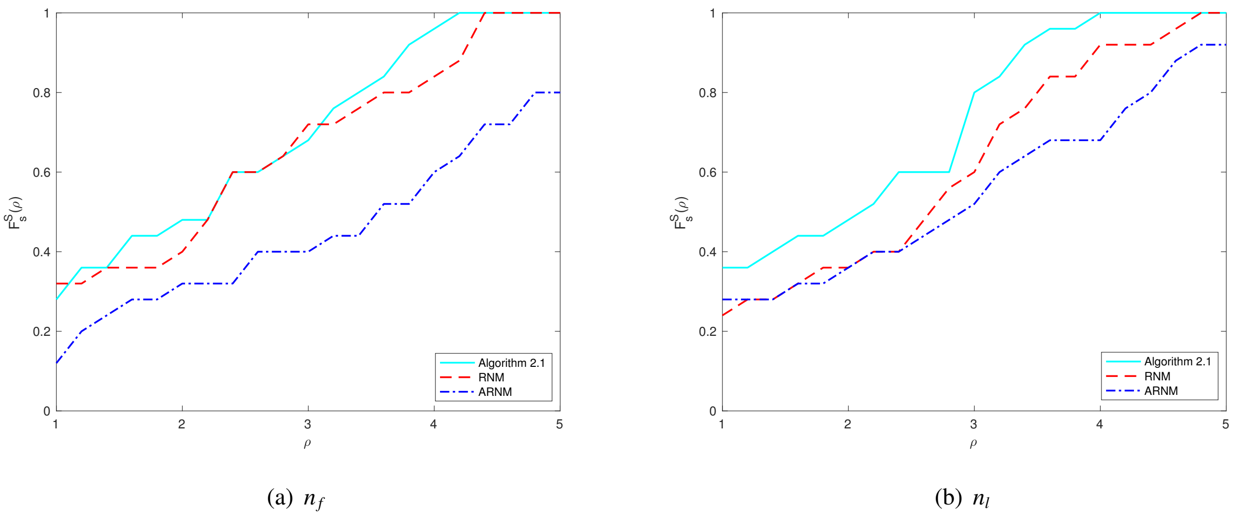



**Figure 1.** Comparison of Algorithm 2.1, the ARNM and the RNM for $n_f$ and $n_l$.

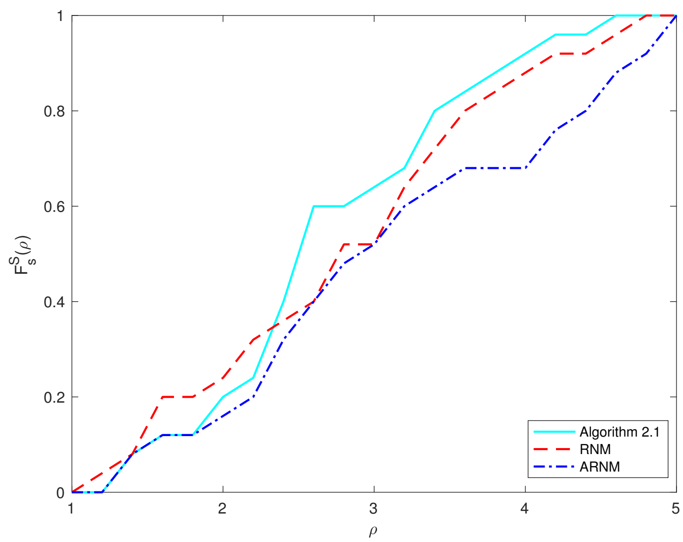
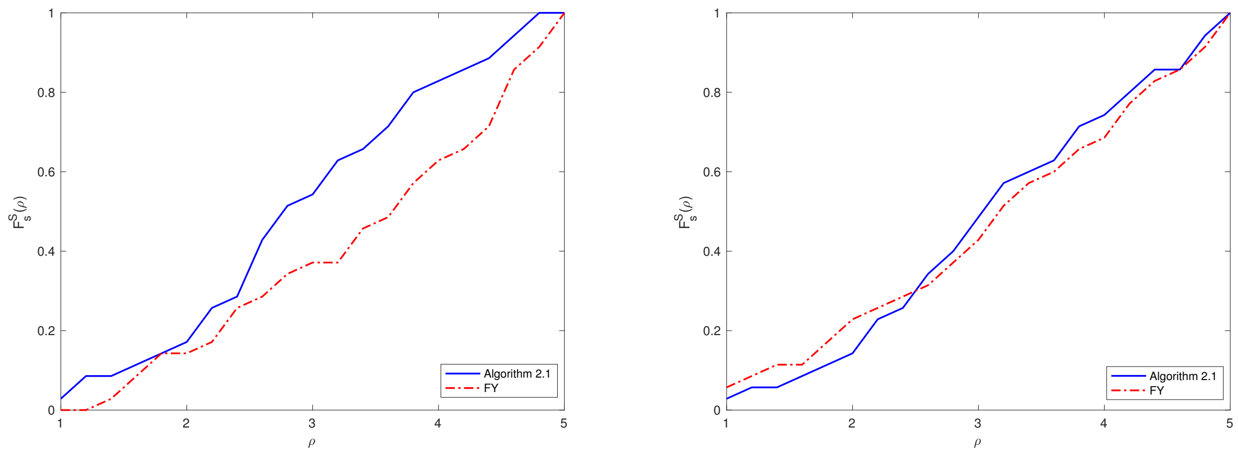



**Figure 2.** Comparison of Algorithm 2.1, the ARNM and the RNM for $n_t$.




(a) $n_l$ (b) $n_t$

**Figure 3.** Comparison of Algorithm 2.1, and the FY for $n_l$ and $n_t$.

From Figure 1 we can see that for comparison with the RNM, $n_f$ of them do not have much difference but $n_l$ of Algorithm 2.1 is less than that of the RNM. When compared with the ARNM, both $n_f$ and $n_l$ of Algorithm 2.1 are much less than those of the RNM. From Figure 2, we can easily see that $n_t$ of Algorithm 2.1 has advantages over the RNM and the ARNM. For comparison with the FY, we can see from Figure 3 that $n_l$ of Algorithm 2.1 is much less than that of the FY and $n_t$ of Algorithm 2.1 doesn't have significant advantages compared with the FY.

## 5. Concluding remarks

In this paper, we have proposed a new Newton method which can solve convex optimization problems with singular Hessian matrices, especially for the special case when the Hessian matrices of the objective function are singular at any iteration point. We have shown the global convergence of the new method. Furthermore, we have provided some numerical experimental results and from the figures we can clearly see the comparison between Algorithm 2.1 and other methods in different aspects. In terms of the number of times the objective function is solved, Algorithm 2.1 does better than the ARNM and the RNM. In terms of the number of iterations, Algorithm 2.1 has a slight advantage over the ARNM, the RNM and the FY. As for the CPU time, Algorithm 2.1 is less than the ARNM and the RNM, but there is no significant difference compared with the FY. So overall, Algorithm 2.1 performs better than the ARNM, the RNM and the FY.

## Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.


## Acknowledgments

This work is supported by National Natural Science Foundation of China grant 12171196.

## Conflict of interest

The authors declare that they have no competing interests.

## References

1. Z. S. Yu, P. X. Li, An active set quasi-Newton method with projection step for monotone nonlinear equations, *AIMS Mathematics*, **6** (2021), 3606–3623. https://doi.org/10.3934/math.2021215
2. H. C. Zhang, W. H. William, A nonmonotone line search technique and its application to unconstrained optimization, *SIAM J. Optim.*, **14** (2004), 1043–1056. https://doi.org/10.1137/S1052623403428208
3. M. Sun, Y. J. Wang, The conjugate gradient methods for solving the generalized periodic Sylvester matrix equations, *J. Appl. Math. Comput.*, **60** (2019), 413–434. https://doi.org/10.1007/s12190-018-01220-3
4. J. Y. Fan, J. Y. Pan, On the convergence rate of the inexact Levenberg-Marquardt method, *J. Ind. Manag. Optim.*, **7** (2011), 199–210. https://doi.org/10.3934/jimo.2011.7.199
5. W. J. Zhou, D. H. Li, A globally convergence BFGS method for nonlinear monotone equations without any merit functions, *Math. Comput.*, **77** (2008), 2231–2240. https://doi.org/10.1090/S0025-5718-08-02121-2
6. J. Y. Fan, Y. X. Yuan, On the quadratic convergence of the Levenberg-Marquardt method without nonsingular assumption, *Computing*, **74** (2005), 23–39. https://doi.org/10.1007/s00607-004-0083-1
7. M. K. Riaha, A new approach to improve ill-conditioned parabolic optimal control problem via time domain decomposition, *Numer. Algor.*, **72** (2016), 635–666. http://doi.org/10.1007/s11075-015-0060-0
8. T. D. Niri, S. A. S. Fazeli, M. Heydari, A two-step improved Newton method to solve convex unconstrained optimization problems, *J. Appl. Math. Comput.*, **62** (2020), 37–53. http://doi.org/10.1007/s12190-019-01272-z
9. C. Cartis, N. I. M. Gould, P. L. Tpint, Adaptive cubic regularisation methods for unconstrained optimization. Part I: motivation, convergence and numerical results, *Math. Program.*, **127** (2011), 245–295. http://doi.org/10.1007/s10107-009-0286-5
10. D. K. R. Babajee, M. Z. Dauhoo, M. T. Darvishi, A. Karami, A. Barati, Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations, *J. Comput. Appl. Math.*, **233** (2010), 2002–2012. http://doi.org/10.1016/j.cam.2009.09.035
11. X. H. Wang, J. S. Kou, R-order of convergence for the improved multi-point Chebyshev-like methods under generalized continuity condition, *Int. J. Comput. Math.*, **97** (2020), 906–919. http://doi.org/10.1080/00207160.2019.1599867
12. H. Darvishi, M. T. Darvishi, An analytical study on two high-order hybrid methods to solve systems of nonlinear equations, *J. Math.-UK*, **2023** (2023), 9917774. https://doi.org/10.1155/2023/9917774

13. C. Cartis, N. I. M. Gould, P. L. Toint, On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems, *SIAM J. Optim.*, **20** (2010), 2833–2852. https://doi.org/10.1137/090774100

14. H. Zhang, Q. Ni, A new regularized quasi-Newton algorithm for unconstrained optimization, *Appl. Math. Comput.*, **259** (2005), 460–469. https://doi.org/10.1016/j.amc.2015.02.032

15. Y. Nesterov, B. T. Polyak, Cubic regularization of Newton method and its global performance, *Math. Program.*, **108** (2006), 177–205. https://doi.org/10.1007/s10107-006-0706-8

16. C. G. Shen, X. D. Chen, Y. M. Liang, A regularized Newton method for degenerate unconstrained optimization problems, *Optim. Lett.*, **6** (2012), 1913–1933. https://doi.org/10.1007/s11590-011-0386-z

17. S. M. Goldfeld, R. E. Quandt, H. F. Trotter, Maximization by quadratic hill-climbing, *Econometrica*, **34** (1966), 541–551.

18. P. E. Gill, W. Murray, Newton-type methods for unconstrained and linearly constrained optimization, *Math. Program.*, **7** (1974), 311–350. https://doi.org/10.1007/BF01585529

19. Y. J. Li, D. H. Li, Truncated regularized Newton method for convex minimization, *Comput. Optim. Appl.*, **43** (2009), 119–131. https://doi.org/10.1007/s10589-007-9128-7

20. G. L. Zhou, L. Q. Qi, On the convergence of an inexact Newton-type method, *Oper. Res. Lett.*, **34** (2006), 647–652. https://doi.org/10.1016/j.orl.2005.11.001

21. H. Zhang, Q. Ni, A new regularized quasi-Newton method for unconstrained optimization, *Optim. Lett.*, **12** (2018), 1639–1658. https://doi.org/10.1007/s11590-018-1236-z

22. M. C. Yue, Z. R. Zhou, A. M. C. So, On the quadratic convergence of the cubic regularization method under a local error bound condition, *SIAM J. Optim.*, **29** (2019), 904–932. http://doi.org/10.1137/18M1167498

23. R. A. Polyak, Regularized Newton method for unconstrained convex optimization, *Math. Program.*, **120** (2009), 125–145. https://doi.org/10.1007/s10107-007-0143-3

24. K. Ueda, N. Yamashita, Convergence properties of the regularized Newton method for the unconstrained nonconvex optimization, *Appl. Math. Optim.*, **62** (2010), 27–46. https://doi.org/10.1007/s00245-009-9094-9

25. K. Ueda, N. Yamashita, A regularized Newton method without line search for unconstrained optimization, *Comput. Optim. Appl.*, **59** (2014), 321–351. https://doi.org/10.1007/s10589-014-9656-x

26. J. Y. Fan, Y. X. Yuan, A regularized Newton method for monotone nonlinear equations and its application, *Optim. Method. Softw.*, **29** (2014), 102–119. https://doi.org/10.1080/10556788.2012.746344

27. W. J. Zhou, X. J. Chen, On the convergence of a modified regularized Newton method for convex optimization with singular solutions, *J. Comput. Appl. Math.*, **239** (2013), 179–188. https://doi.org/10.1016/j.cam.2012.09.030

28. D. H. Li, M. Fukushima, L. Q. Qi, N. Yamashita, Regularized newton methods for convex minimization problems with singular solutions, *Comput. Optim. Appl.*, **28** (2004), 131–147. https://doi.org/10.1023/B:COAP.0000026881.96694.32

29. E. D. Dolan, J. J. Moré, Benchmarking optimization software with performance profiles, *Math. Program.*, **91** (2002), 201–213. https://doi.org/10.1007/s101070100263

AIMS Press