Mathematics
https://www.aimspress.com/journal/Math

AIMS Mathematics, 11(1): 2527–2546.
DOI: 10.3934/math.2026102
Received: 17 October 2025
Revised: 07 January 2026
Accepted: 19 January 2026
Published: 26 January 2026

Research article

The modified Levenberg–Marquardt method with nonmonotone technique

Yuya Zheng, Yueting Yang*and Mingyuan Cao*

School of Mathematics and Statistics, Beihua University, Jilin 132013, China

* **Correspondence:** Email: yyt2858@163.com, cmy0918@beihua.edu.cn.

Abstract: In this paper, we propose a modified Levenberg–Marquardt (LM) method with a nonmonotone technique for solving nonlinear equations. Under the Hölderian continuity and the Hölderian local error bounds conditions, which are weaker than the local error bounds and the Lipschitz continuity, the global convergence and local convergence of the algorithm are proved. Numerical experiments also show that the algorithm is effective.

Keywords: Levenberg–Marquardt method; nonlinear equations; convergence rate; nonmonotone technique; Hölderian local error bounds
Mathematics Subject Classification: 65K05, 90C30

1. Introduction

Consider the system of nonlinear equations

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

where $F(x) : \mathbb{R}^n \to \mathbb{R}^m$ is continuously differentiable. We assume that the solution set of (1.1) is nonempty and defined as X^*. In all cases, $\|\cdot\|$ is defined as 2–norm. We take $\|F(x)\|$ as the merit function of (1.1).

The nonlinear equations have been used in various fields, such as differential equation problems, robotic systems, transportation, neural network control problems, and so on [1–6]. One of the most well-known methods for solving this problem is the Gauss–Newton method [7]. At every iteration, the trial step should be computed as

$$d_k^{GN} = -(J_k^T J_k)^{-1} J_k^T F_k, \tag{1.2}$$

where $F_k = F(x_k)$ and $J_k = F'(x_k)$, which is the Jacobian matrix $J(x)$ of $F(x)$ at x_k. If the Jacobian matrix $J(x)$ of $F(x)$ is Lipschitz continuous and nonsingular at the solution, the Gauss–Newton method has quadratic convergence. However, the Gauss–Newton method may not be well defined if the Jacobian matrix is singular or near singular. Marquardt [8] explored the method proposed by Levenberg [9] and introduced a positive parameter $\lambda_k \geq 0$ to compute a trial step d_k,

$$d_k = -(J_k^T J_k + \lambda_k I)^{-1} J_k^T F_k. \tag{1.3}$$

The method was called the Levenberg–Marquardt (LM) method. Obviously, if J_k was nonsingular and $\lambda_k = 0$, the LM trial step reduced to the Gauss–Newton trial step.

For the LM method, it is important to choose the parameter λ_k. Yamashita and Fukushima [10] used the LM parameter $\lambda_k = \|F_k\|^2$ to show that the LM method had quadratic convergence under the local error bound condition and $J(x)$ was Lipschitz continuous. Fan [11] introduced the parameter $\lambda_k = \mu_k \|F_k\|$ with μ_k being updated in each iteration and proved the method had quadratic convergence under the same conditions. Fan and Yuan [12] considered $\lambda_k = \|F_k\|^\delta$ with $\delta \in [1, 2]$ and showed the method had quadratic convergence. Amini and Rostami [13] found that if the sequence $\{x_k\}$ was far away from the solution set and $\delta \geq 1$, then $\lambda_k = \|F_k\|^\delta$ might be very large. This made the LM step smaller and hindered the iteration from approaching the solution set quickly, so they introduced a new LM parameter $\lambda_k = \mu_k \|F_k\|^{\delta_k}$ with δ_k as follows:

$$\delta_k = \begin{cases} \frac{1}{\|F_k\|}, & \text{if } \|F_k\| \geq 1, \\ 1 + \frac{1}{k}, & \text{otherwise.} \end{cases} \tag{1.4}$$

The above research adopted the trust-region framework. In their research, the trial step d_k can be regarded as the solution to the subproblem

$$\begin{aligned} &\min_{d \in \mathbb{R}^n} \|F_k + J_k d\|^2 \\ &\ s.t.\ \|d\| \leq \Delta_k := \|d_k\|. \end{aligned} \tag{1.5}$$

We set the ratio as

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

where $Ared_k$ denotes the actual reduction and is given by

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

and $Pred_k$ denotes the predicted reduction and is given by

$$Pred_k = \|F_k\|^2 - \|F_k + J_k d_k\|^2. \tag{1.6}$$

The ratio of r_k is used to decide whether to accept the LM trial step d_k and update the LM parameters.

Amini [14] proposed a new LM parameter $\lambda_k = \frac{\mu_k \|F_k\|}{1+\|F_k\|}$ and developed an algorithm with nonmonotone technique. The nonmonotone technique sets

$$\|F_{l(k)}\| = \max_{0 \leq j \leq n(k)} \left\{ \left\| F_{k-j} \right\| \right\}, k = 0, 1, 2, \ldots, \tag{1.7}$$

and let

$$\bar{A}red_k = \|F_{l(k)}\|^2 - \|F(x_k + d_k)\|^2, \tag{1.8}$$

where $n(k) = \min\{N_0, k\}$ and N_0 was a given positive constant. It did not require the merit function to be strictly reduced in each iteration, but $\|F_{k+1}\|$ should have been smaller than the maximum value of $\|F_k\|$ at the previous $n(k)$ steps, which showed the sequence $\{\|F_{l(k)}\|\}$ exhibited a decreasing trend. The ratio between the actual and predicted reductions was defined as

$$\hat{r}_k = \frac{\bar{A}red_k}{Pred_k}. \tag{1.9}$$

Amini proved the algorithm had quadratic convergence under the local error bound condition, and $J(x)$ was Lipschitz continuous.

Some nonlinear equations may fail to satisfy a local error bound condition based on $\|F(x)\|$, and $J(x)$ may not satisfy Lipschitz continuity. But they may satisfy the weaker conditions, that is, the Hölderian local error bound and $J(x)$ being Hölderian continuous. Recently, Wang and Fan [15] considered the algorithm by using parameters $\lambda_k = \eta_k\|F_k\|^\alpha$ and $\lambda_k = \eta_k\|J_k^T F_k\|^\alpha$ with $\alpha > 0$. The convergence analysis of the algorithm was discussed under the Hölderian local error bound of $\|F_k\|$, and $J(x)$ was Hölderian continuous. Chen and Ma [16] proposed the parameter $\lambda_k = \theta\|F_k\|^\delta + (1-\theta)\|J_k^T F_k\|^\delta$, where $\theta \in [0, 1]$ and $\delta \in [1, 2]$ and analyzed the convergence of this modified method under the Hölderian local error bound of $\|F_k\|$ and the Hölderian continuity of its Jacobian matrix. Zeng [17] considered the parameter $\lambda_k = \frac{\mu_k\|F_k\|}{1+\|F_k\|}$ and analyzed the convergence of the algorithm under the two same key conditions. Under such weaker conditions, the convergence of the LM algorithm is worth studying.

The paper is organized as follows: In Section 2, we introduce a new modified algorithm, and its global convergence under the Hölderian continuity of the Jacobian matrix is presented. In Section 3, we verify the convergence rate of the modified algorithm under the Hölderian local error bound condition and the Hölderian continuity of the Jacobian matrix. In Section 4, the numerical experiments are implemented and show that the proposed algorithm is effective. Finally, we conclude the paper in Section 5.

2. The new algorithm and its global convergence

In this section, we propose a modified LM algorithm with a nonmonotone technique, and prove its global convergence under the Hölderian continuity condition.

A new LM parameter is introduced as follows:

$$\lambda_k = \frac{\mu_k\|F_k\|^{\delta_k}}{1 + \|J_k^T F_k\|^{\delta_k}}, \delta_k = \begin{cases} \frac{1}{\|F_k\|}, & \text{if } \|F_k\| \geq 1, \\ 1 + \frac{1}{\ln(k+e)}, & \text{otherwise.} \end{cases} \tag{2.1}$$

The selection of the parameter δ_k is inspired by (1.4). The strategy can adaptively adjust the parameter λ_k. When $k = 0$, the initial point x_0 may be far from the solution, and we may have $\|F_0\| \geq 1$. We then take $\delta_0 = \frac{1}{\|F_0\|}$, which is equivalent to the parameter in [13]. In previous iterations, the sequence $\{x_k\}$ is far from the solution set of (1.1) and $\|F_k\|$ and $\|J_k^T F_k\|$ may be very large, so it leads to $\frac{\|F_k\|^{\delta_k}}{1+\|J_k^T F_k\|^{\delta_k}}$ being approximately equal to $\frac{1}{2}$ and λ_k being close to $\frac{\mu_k}{2}$. If the sequence $\{x_k\}$ is close to the solution set, $||F_k|| < 1$ may be satisfied. We set $\delta_k = 1 + \frac{1}{\ln(k+e)}$, and compared with the parameter $\delta_k = \frac{1}{k}$ in [13], the term $\frac{1}{\ln(k+e)}$ decreases more slowly than $\frac{1}{k}$. So, λ_k does not decrease quickly. This prevents d_k from growing quickly and prevents the iteration from deviating from the solution set. Meanwhile, $||J_k{}^T F_k||$ becomes sufficiently small. As k increases, δ_k tends to 1. Then, λ_k is close to $\mu_k ||F_k||$.

The algorithm is described as follows.

Algorithm 1 The nonmonotone modified Levenberg–Marquardt method (NMLM)

Step 0: Input an initial point $x_0 \in \mathbb{R}^n$, the parameters $N_0 > 0$, $\mu_0 > m > 0$, $\varepsilon > 0$, $0 < p_0 \leq p_1 \leq p_2 < 1$. Set $k := 0$.

Step 1: If $||J_k^T F_k|| \leq \varepsilon$, stop. Otherwise, compute $||F_{l(k)}||$ by (1.7);

Step 2: Compute λ_k by (2.1) and d_k by (1.3);

Step 3: Compute $Pred_k$, $\bar{A}red_k$, and $\hat{r}_k$ by (1.6), (1.8), and (1.9), respectively;

Step 4: If $\hat{r}_k < p_0$, set $\mu_k := 4\mu_k$, go to Step 2;
Otherwise, set $x_{k+1} = x_k + d_k$, go to Step 5;

Step 5: Choose μ_{k+1} as

$$\mu_{k+1} = \begin{cases} 4\mu_k, & \text{if } \hat{r}_k < p_1, \\ \mu_k, & \text{if } \hat{r}_k \in [p_1, p_2], \\ \max\left\{\frac{\mu_k}{4}, m\right\}, & \text{otherwise.} \end{cases}$$

Step 6: Calculate $||F_{k+1}||$ and $||J_{k+1}^T F_{k+1}||$. Set $k := k + 1$ and go to Step 1.

Remark 2.1. *In Algorithm 1, there is an inner loop from Step 2 to Step 4. If $\hat{r}_k < p_0$, the trial step d_k is rejected and recalculated by increasing the parameter μ_k. The inner loop avoids repeated evaluations of $||F_k||$, $||J_k||$, and $||J_k^T F_k||$ at the same iterate. By updating only the parameter λ_k and the trial step d_k, the strategy significantly reduces the computational cost. The efficiency and numerical performance of the algorithm can be improved. The inner loop should be terminated in a finite number of steps. In fact, if the inner loop runs indefinitely, for sufficiently large k, μ_k and λ_k may become very large, which makes $||d_k||$ tend to 0. From (1.6) and (1.8), the inequality $\hat{r}_k \geq p_0$ will definitely occur. Then, the inner loop is terminated. This strategy means that when the trust-region subproblem does not fit well, we directly adjust the parameter μ_k to generate a new trial step, which reduces the computational cost.*

Remark 2.2. *In Step 5, based on the idea of trust region, the ratio $\hat{r}_k$ reflects the degree of approximation of the trust-region subproblem to the original problem. If $\hat{r}_k < p_1$, the approximation is still insufficient, so we adjust λ_k by increasing μ_k. If $\hat{r}_k \in [p_1, p_2]$, it indicates a relatively good approximation, and thus μ_k is kept unchanged. If $\hat{r}_k > p_2$, the trust-region subproblem provides a good approximation of the original problem. At this time, we set $\mu_{k+1} = \frac{\mu_k}{4}$. If μ_{k+1} keeps being set to $\frac{\mu_k}{4}$ repeatedly, μ_{k+1} may become very small, which makes $||d_k||$ too large. So, we choose a protective measure $\mu_{k+1} = \max\left\{\frac{\mu_k}{4}, m\right\}$, where m is a positive constant.*

Lemma 2.1. *Suppose that the sequence $\{x_k\}$ is generated by Algorithm 1, then the predicted reduction $Pred_k$ satisfies*

$$Pred_k \geq ||J_k^T F_k|| \min\{||d_k||, \frac{||J_k^T F_k||}{||J_k^T J_k||}\} \tag{2.2}$$

for all $k \in \mathbb{N}$.

The proof is similar to the proof of Theorem 4 in [18].

Lemma 2.2. *If the sequence* $\{x_k\}$ *is generated by Algorithm 1, then the sequence* $\{\|F_{l(k)}\|\}$ *is convergent.*

Proof. From Step 4, we know that $\hat{r}_k \geq p_0 > 0$ for all k. From Lemma 2.1,

$$\bar{A}red_k \geq p_0 Pred_k \geq p_0 \|J_k^T F_k\| \min\{\|d_k\|, \frac{\|J_k^T F_k\|}{\|J_k^T J_k\|}\} > 0. \tag{2.3}$$

By (1.8), we obtain

$$\|F_{l(k)}\| > \|F_{k+1}\| \quad \forall k. \tag{2.4}$$

For $k \geq N_0$, we have

$$\|F_{l(k+1)}\| = \max\{\|F_{k+1}\|, \|F_k\|, \cdots, \left\|F_{k-N_0+1}\right\|\} \leq \max\{\|F_{k+1}\|, \|F_{l(k)}\|\} = \|F_{l(k)}\|.$$

Thus, the sequence $\{\|F_{l(k)}\|\}$ is monotonically decreasing. By the definition of $F_{l(k)}$ and (2.4), there exists $k_0 \geq N_0$ such that for all $k \geq k_0$,

$$\|F_{l(k+1)}\| < \|F_{l(k)}\|.$$

That is, the sequence $\{\|F_{l(k)}\|\}$ is strictly decreasing for all $k \geq k_0$.
Since the sequence $\{\|F_{l(k)}\|\}$ is bounded below, the sequence $\{\|F_{l(k)}\|\}$ is convergent. □

Before discussing the global convergence of Algorithm 1, we need to establish the following assumptions.

Assumption 2.1. (a) $J(x)$ *is Hölderian continuous of order* $\nu \in (0, 1]$, *i.e., there exists a positive constant* c_h *such that*

$$\|J(x) - J(y)\| \leq c_h \|x - y\|^\nu, \forall x, y \in \mathbb{R}^n. \tag{2.5}$$

(b) $J(x)$ *is bounded, i.e., there exists a positive constant* c_J *such that*

$$\|J(x)\| \leq c_J, \forall x \in \mathbb{R}^n. \tag{2.6}$$

By (2.5) and (2.6), it holds that

$$\|F(y) - F(x) - J(x)(y - x)\| \leq \frac{c_h}{1+\nu} \|y - x\|^{1+\nu}, \tag{2.7}$$

$$\|F(y) - F(x)\| \leq c_J \|y - x\|. \tag{2.8}$$

It is not difficult to find that if $\nu = 1$, the Hölderian continuous condition becomes the Lipschitz continuity condition.

Theorem 2.1. *Under the conditions in Assumption 2.1, the sequence $\{x_k\}$ generated by Algorithm 1 satisfies*

$$\liminf_{k\to\infty} \|J_k^T F_k\| = 0. \tag{2.9}$$

Proof. Suppose that (2.9) does not hold. There exists a positive ε_0 such that

$$\left\|J_k^T F_k\right\| \geq \varepsilon_0, \quad \forall k. \tag{2.10}$$

From Step 4, we may obtain

$$\hat{r}_k \geq p_0, \quad \forall k. \tag{2.11}$$

By (1.9) and Lemma 2.1,

$$\|F_{l(k)}\|^2 - \|F_{k+1}\|^2 \geq p_0 \left\|J_k^T F_k\right\| \min\left\{\|d_k\|, \frac{\left\|J_k^T F_k\right\|}{\left\|J_k^T J_k\right\|}\right\}.$$

And according to Assumption 2.1(b) and (2.10),

$$\|F_{l(k)}\|^2 - \|F_{k+1}\|^2 \geq p_0\varepsilon_0 \min\left\{\|d_k\|, \frac{\varepsilon_0}{c_J^2}\right\}. \tag{2.12}$$

By substituting k with $l(k) - 1$, we obtain

$$\|F_{l(l(k)-1)}\|^2 - \left\|F_{l(k)}\right\|^2 \geq p_0\varepsilon_0 \min\left\{\left\|d_{l(k)-1}\right\|, \frac{\varepsilon_0}{{c_J}^2}\right\}. \tag{2.13}$$

According to Lemma 2.2,

$$\lim_{k\to\infty}(\|F_{l(l(k)-1)}\|^2 - \|F_{l(k)}\|^2) = 0.$$

Combined with (2.13),

$$\lim_{k\to\infty} \|d_{l(k)-1}\| = 0. \tag{2.14}$$

And by (2.8), we have

$$\lim_{k\to\infty} \|F_{l(k)}\| = \lim_{k\to\infty} \|F_{l(k)-1}\|. \tag{2.15}$$

Let $\hat{l}(k) = l(k + N_0 + 2)$. For any given $j \geq 1$, replace $l(k)$ with $\hat{l}(k) - j + 1$ in (2.13), thus

$$\|F_{l(\hat{l}(k)-j)}\|^2 - \left\|F_{\hat{l}(k)-j+1}\right\|^2 \geq p_0\varepsilon_0 \min\left\{\left\|d_{\hat{l}(k)-j}\right\|, \frac{\varepsilon_0}{{c_J}^2}\right\}. \tag{2.16}$$

According to Lemma 2.2, $\|F(x_{l(k)})\|$ is convergent, and any of its subsequences are also convergent. It implies that

$$\lim_{k\to\infty} \|F_{\hat{l}(k)-j+1}\| = \lim_{k\to\infty} \|F_{l_{\hat{l}(k)-j}}\|.$$

By (2.16),

$$\lim_{k\to\infty} \|d_{\hat{l}(k)-j}\| = 0. \tag{2.17}$$

From (2.8), it can be inferred that

$$\lim_{k\to\infty} \|F_{\hat{l}(k)-j}\| = \lim_{k\to\infty} \|F_{\hat{l}(k)}\|.$$

Since $\|F_{\hat{l}(k)}\|$ is a subsequence of $\|F_{l(k)}\|$, we have

$$\lim_{k\to\infty} \|F_{\hat{l}(k)-j}\| = \lim_{k\to\infty} \|F_{\hat{l}(k)}\| = \lim_{k\to\infty} \|F_{l(k)}\|. \tag{2.18}$$

Moreover,

$$x_{\hat{l}(k)} - x_{k+1} = \sum_{j=1}^{\hat{l}(k)-k-1} d_{\hat{l}(k)-j}, \quad \forall k.$$

By (2.17), considering that $\hat{l}(k) - j - 1 \le N_0 + 1$,

$$\lim_{k\to\infty} \|x_{k+1} - x_{\hat{l}(k)}\| = 0.$$

And according to (2.8) and (2.18) such that

$$\lim_{k\to\infty} \|F_{k+1}\| = \lim_{k\to\infty} \|F_{\hat{l}(k)}\| = \lim_{k\to\infty} \|F_{l(k)}\|, \tag{2.19}$$

hence, by (2.12),

$$\lim_{k\to\infty} \|d_k\| = 0. \tag{2.20}$$

Therefore, it follows from (1.3), (2.1), (2.8), and (2.10) that

$$\mu_k \to \infty, \quad as \quad k \to \infty. \tag{2.21}$$

On the other hand, by (2.7), we have

$$\Big| \|F(x_k + d_k)\| - \|F_k + J_k d_k\| \Big| \le \frac{c_h}{1+\nu} \|d_k\|^{1+\nu}. \tag{2.22}$$

If $\|F(x_k + d_k)\| \ge \|F_k + J_k d_k\|$ and by (2.22), we have

$$\begin{aligned} \|F(x_k + d_k)\| + \|F_k + J_k d_k\| &= \|F_k + J_k d_k\| + (\|F(x_k + d_k)\| - \|F_k + J_k d_k\|) + \|F_k + J_k d_k\| \\ &= 2\|F_k + J_k d_k\| + \|F(x_k + d_k)\| - \|F_k + J_k d_k\| \\ &\le 2\|F_k + J_k d_k\| + \frac{c_h}{1+\nu} \|d_k\|^{1+\nu}. \end{aligned} \tag{2.23}$$

Similarly, if $\|F(x_k + d_k)\| \le \|F_k + J_k d_k\|$, we obtain

$$\Big| \|F(x_k + d_k)\| + \|F_k + J_k d_k\| \Big| \le 2\|F(x_k + d_k)\| + \frac{c_h}{1+\nu} \|d_k\|^{1+\nu}. \tag{2.24}$$

According to (2.22), (2.23), and (2.24),

$$
\begin{aligned}
\Big| \|F(x_k+d_k)\|^2 - \|F_k + J_k d_k\|^2 \Big| &= \Big| \|F(x_k+d_k)\| - \|F_k + J_k d_k\| \Big| (\|F(x_k+d_k)\| + \|F_k + J_k d_k\|) \\
&\le \begin{cases} \frac{c_h}{1+\nu}\|d_k\|^{1+\nu}(2\|F_k + J_k d_k\| + \frac{c_h}{1+\nu}\|d_k\|^{1+\nu}), \text{if } \|F(x_k+d_k)\| \ge \|F_k + J_k d_k\|, \\ \frac{c_h}{1+\nu}\|d_k\|^{1+\nu}(2\|F(x_k+d_k)\| + \frac{c_h}{1+\nu}\|d_k\|^{1+\nu}), \text{otherwise}, \end{cases} \\
&= \frac{2c_h}{1+\nu}\|d_k\|^{1+\nu}\min\{\|F_k + J_k d_k\|, \|F(x_k+d_k)\|\} + \frac{{c_h}^2}{(1+\nu)^2}\|d_k\|^{2+2\nu} \\
&\le \frac{2c_h}{1+\nu}\|d_k\|^{1+\nu}\|F_k + J_k d_k\| + \frac{{c_h}^2}{(1+\nu)^2}\|d_k\|^{2+2\nu}. \qquad (2.25)
\end{aligned}
$$

Hence, according to Assumption 2.1, (2.25), and (2.5), we have

$$
\begin{aligned}
|r_k - 1| &= \left| \frac{Ared_k - Pred_k}{Pred_k} \right| \\
&= \left| \frac{(\|F_k\|^2 - \|F(x_k+d_k)\|^2) - (\|F_k\|^2 - \|F_k + J_k d_k\|^2)}{Pred_k} \right| \\
&= \left| \frac{\|F_k + J_k d_k\|^2 - \|F(x_k+d_k)\|^2}{Pred_k} \right| \\
&\le \frac{\frac{2c_h}{1+\nu}\|d_k\|^{1+\nu}\|F_k + J_k d_k\| + \frac{{c_h}^2}{1+\nu}\|d_k\|^{2+2\nu}}{\|{J_k}^T F_k\| \min\{\|d_k\|, \frac{\|{J_k}^T F_k\|}{\|{J_k}^T J_k\|}\}} \\
&\to 0. \qquad (2.26)
\end{aligned}
$$

By (1.7), (1.8), and (2.19),

$$
\frac{\bar{A}red_k}{Pred_k} = \frac{\|F_{l(k)}\|^2 - \|F(x_k+d_k)\|^2}{Pred_k} \ge \frac{\|F_k\|^2 - \|F(x_k+d_k)\|^2}{Pred_k} = r_k \to 1. \qquad (2.27)
$$

This implies that Step 5 in Algorithm 1 always executes $\mu_{k+1} = max\{\frac{\mu_k}{4}, m\}$, so there exists a positive number $u > m$ such that $\mu_k < u$ for sufficiently large k, which contradicts (2.21), and the proof is complete. □

Theorem 2.2. *Suppose that the sequence $\{x_k\}$ generated by Algorithm 1 satisfies Assumption 2.1, and for sufficiently large k, $\delta_k = 1 + \frac{1}{\ln(k+e)}$. Then,*

$$
\lim_{k\to\infty} \|F_k\| = 0. \qquad (2.28)
$$

Proof. By Lemma 2.2, the sequence $\{\|F_{l(k)}\|\}$ is convergent. Then, there exists a constant L^* such that

$$
\lim_{k\to\infty} \|F_{l(k)}\| = L^* \ge 0.
$$

Assume that $L^* \neq 0$, and let $L = L^*/2$. There exists a constant K_0 for all $k \ge K_0$ such that

$$
\|F_{l(k)}\| \ge L > 0. \qquad (2.29)
$$

By Theorem 2.1, there exists a subsequence $\{k_t\}$ that satisfies $k_t \geq K_0$ such that

$$\lim_{t\to\infty} \|J_{k_t}^T F_{k_t}\| = 0. \tag{2.30}$$

According to (2.13), for $k_t \geq K_0$, we obtain

$$\|F_{l(l(k_t)-1)}\|^2 - \left\|F_{l(k_t)}\right\|^2 \geq p_0\varepsilon_0 \min\left\{\left\|d_{l(k_t)-1}\right\|, \frac{\varepsilon_0}{{c_J}^2}\right\}.$$

Then,

$$\lim_{t\to\infty} \|d_{l(k_t)-1}\| = 0. \tag{2.31}$$

By (2.8), we have

$$\lim_{t\to\infty} \|F(x_{l(k_t)})\| = \lim_{t\to\infty} \|F(x_{l(k_t)-1})\| = L^*. \tag{2.32}$$

Additionally,

$$\bar{A}red_k = \|F_{l(k)}\|^2 - \|F(x_k + d_k)\|^2 > 0. \tag{2.33}$$

And let

$$S = \sum_{k=K_0}^{\infty} \bar{A}red_k.$$

Since S is convergent, we obtain

$$\lim_{k\to\infty} \bar{A}red_k = 0.$$

So,

$$\lim_{t\to\infty} \bar{A}red_{k_t} = 0. \tag{2.34}$$

On the other hand, it follows from (2.27) and (2.34) that

$$Pred_{k_t} \to 0. \tag{2.35}$$

According to equation (1.5), for $k_t \geq K_0$, $\delta_k \in (1, 2]$, we have

$$\lambda_{k_t} = \frac{\mu_{k_t}\|F_k\|^{\delta_{k_t}}}{1 + \|J_{k_t}^T F_{k_t}\|^{\delta_{k_t}}} \to \mu_{k_t}\|F_{k_t}\|. \tag{2.36}$$

For $k_t \geq K_0$, we have

$$\|F_{k_t}\| \geq \|F_{l(k_t)}\| - \|F_{k_t} - F_{l(k_t)}\| > \frac{L}{2}.$$

According to the above reasoning, there exists a constant η such that $Pred_{k_t} \geq \eta$. This leads to a contradiction, which means $\lim_{k\to\infty} \|F_{l(k)}\| = 0$. According to (1.7), (2.28) holds. The proof is complete. □

3. Local convergence of the new algorithm

In this section, we perform the convergence analysis of the algorithm under the Hölderian local error bound conditions of $F(x)$ and the Hölderian continuity of the Jacobian matrix.

We assume that the sequence $\{x_k\}$ generated by Algorithm 1 converges to the solution set X^* and lies in some neighborhood of $x^* \in X^*$. We denote the distance between x and X^* as $dist(x, X^*) := \|\bar{x} - x\|$, where $\bar{x} \in X^*$ and is the point closest to x.

Assumption 3.1. *$F(x)$ provides a Hölderian local error bound of order $\gamma \in (0, 1]$ in some neighborhood of $x^* \in X^*$, i.e., there exist constants $c > 0$ and $0 < b < 1$ such that*

$$cdist(x, X^*) \le \|F(x)\|^\gamma, \quad \forall x \in N(x^*, b), \tag{3.1}$$

where $N(x^*, b) = \{x \in \mathbb{R}^n \mid \|x - x^*\| \le b\}$.

Assumption 3.2. *(a) $J(x)$ is Hölderian continuous of order $\nu \in (0, 1]$, i.e., there exists a positive constant c_h such that*

$$\|J(x) - J(y)\| \le c_h\|x - y\|^\nu, \forall x \in N(x^*, b). \tag{3.2}$$

(b) $J(x)$ is bounded, i.e., there exists a positive constant c_J such that

$$\|J(x)\| \le c_J, \forall x \in N(x^*, b). \tag{3.3}$$

By (3.2) and (3.3), we have

$$\|F(y) - F(x) - J(x)(y - x)\| \le \frac{c_h}{1 + \nu}\|y - x\|^{1+\nu}, \forall x \in N(x^*, \frac{b}{2}), \tag{3.4}$$

$$\|F(y) - F(x)\| \le c_J\|y - x\|, \forall x \in N(x^*, \frac{b}{2}). \tag{3.5}$$

In the following, we describe the relationship between $\|d_k\|$ and $dist(x_k, X^*)$, then we give the boundedness of the parameter. Without loss of generality, we assume that x_k lies in $N(x^*, \frac{b}{4})$.

Lemma 3.1. *Under Assumption 3.1 and Assumption 3.2, for all sufficiently large k, we have*

$$\|d_k\| \le O(\|\bar{x}_k - x_k\|^{min\{1,1+\nu-\frac{1}{2\gamma}\}}). \tag{3.6}$$

Proof. Since $x_k \in N(x^*, \frac{b}{4})$, we have

$$\|\bar{x}_k - x^*\| \le \|\bar{x}_k - x_k\| + \|x_k - x^*\| \le 2\|x_k - x^*\| \le \frac{b}{2}. \tag{3.7}$$

Then, $\bar{x}_k \in N(x^*, \frac{b}{2})$. According to (3.1), Step 5 in Algorithm 1, (3.3), and (3.5), we obtain

$$\begin{aligned}
\lambda_k &= \frac{\mu_k \|F_k\|^{\delta_k}}{1 + \left\|J_k^T F_k\right\|^{\delta_k}} \\
&\geq \frac{m c^{\frac{\delta_k}{\gamma}} \|\bar{x}_k - x_k\|^{\frac{\delta_k}{\gamma}}}{1 + c_J^{\delta_k} \|F_k\|^{\delta_k}} \\
&\geq \frac{m c^{\frac{\delta_k}{\gamma}} \|\bar{x}_k - x_k\|^{\frac{\delta_k}{\gamma}}}{1 + c_J^{\delta_k} c^{\frac{\delta_k}{\gamma}} \|\bar{x}_k - x_k\|^{\frac{\delta_k}{\gamma}}}.
\end{aligned}$$

By Theorem 2.2, as k increases, $\delta_k = 1 + \frac{1}{ln(k+e)}$ tends to 1, which gives

$$\lambda_k \to \frac{m c^{\frac{1}{\gamma}} \|\bar{x}_k - x_k\|^{\frac{1}{\gamma}}}{1 + c_J c^{\frac{1}{\gamma}} \|\bar{x}_k - x_k\|^{\frac{1}{\gamma}}}. \tag{3.8}$$

Define

$$\varphi_k(d) = \|F_k + J_k d\|^2 + \lambda_k \|d\|^2. \tag{3.9}$$

Obviously, d_k is the minimizer of $\varphi_k(d)$. By (3.4) and (3.8), we have

$$\begin{aligned}
\|d_k\|^2 &\leq \frac{\varphi_k(d_k)}{\lambda_k} \\
&\leq \frac{\varphi_k(\bar{x}_k - x_k)}{\lambda_k} \\
&= \frac{\|F_k + J_k(\bar{x}_k - x_k)\|^2 + \lambda_k \|\bar{x}_k - x_k\|^2}{\lambda_k} \\
&\leq \frac{c_h^2}{\lambda_k (1+\nu)^2} \|\bar{x}_k - x_k\|^{2+2\nu} + \|\bar{x}_k - x_k\|^2 \\
&\leq \frac{{c_h}^2}{m(1+\nu)^2} \left(\frac{1}{c^{\frac{1}{\gamma}}} \|\bar{x}_k - x_k\|^{2+2\nu-\frac{1}{\gamma}} + (1 + c_J) \|\bar{x}_k - x_k\|^{2+2\nu} \right) + \|\bar{x}_k - x_k\|^2 \\
&\leq \frac{{c_h}^2 (1 + c^{\frac{1}{\gamma}} + c^{\frac{1}{\gamma}} c_J)}{m c^{\frac{1}{\gamma}} (1+\nu)^2} \|\bar{x}_k - x_k\|^{2+2\nu-\frac{1}{\gamma}} + \|\bar{x}_k - x_k\|^2 \\
&= O(\|\bar{x}_k - x_k\|^{2\min\{1, 1+\nu-\frac{1}{2\gamma}\}}).
\end{aligned} \tag{3.10}$$

The proof is complete. □

Lemma 3.2. *Under Assumption 3.1 and Assumption 3.2, if* $\nu > \max\left\{\frac{1}{\gamma} - 1, \frac{2}{2\gamma(1+\nu)-1} - 1, \frac{\nu(1+\gamma)}{2\nu(1+\gamma)-1} - 1\right\}$, *which for all large k, there exists a constant* $\tilde{\mu} > 0$ *such that*

$$\mu_k \leq \tilde{\mu}. \tag{3.11}$$

Proof. Clearly, d_k is also the solution of (1.5). Next, we discuss the following two cases.

Case 1: $\|\bar{x}_k - x_k\| \leq \|d_k\|$. According to (3.1), (3.4), (3.6), and $\nu > \frac{1}{\gamma} - 1$,

$$\|F_k\| - \|F_k + J_k d_k\| \geq \|F_k\| - \|F_k + J_k (\bar{x}_k - x_k)\|$$

$$\geq c^{\frac{1}{\gamma}}\|\bar{x}_k - x_k\|^{\frac{1}{\gamma}} - \frac{c_h}{1+\nu}\|\bar{x}_k - x_k\|^{1+\nu}$$
$$\geq \left(c^{\frac{1}{\gamma}} - \frac{c_h}{1+\nu}\right)\|\bar{x}_k - x_k\|^{\frac{1}{\gamma}}$$
$$\geq c_1\|d_k\|^{\max\{\frac{1}{\gamma}, \frac{2}{2\gamma(1+\nu)-1}\}}, \tag{3.12}$$

where $c_1 > 0$.

Case 2: $\|\bar{x}_k - x_k\| \geq \|d_k\|$. We have

$$\begin{aligned}\|F_k\| - \|F_k + J_k d_k\| &\geq \|F_k\| - \left\|F_k + \frac{\|d_k\|}{\|\bar{x}_k - x_k\|}J_k(\bar{x}_k - x_k)\right\| \\ &\geq \|F_k\| - \left\|\left(1 - \frac{\|d_k\|}{\|\bar{x}_k - x_k\|}\right)F_k + \frac{\|d_k\|}{\|\bar{x}_k - x_k\|}(F_k + J_k(\bar{x}_k - x_k))\right\| \\ &\geq \frac{\|d_k\|}{\|\bar{x}_k - x_k\|}(\|F_k\| - \|F_k + J_k(\bar{x}_k - x_k)\|) \\ &\geq \left(c^{\frac{1}{\gamma}} - \frac{c_h}{1+\nu}\right)\|d_k\|\,\|\bar{x}_k - x_k\|^{\frac{1}{\gamma}-1} \\ &\geq c_2\|d_k\|^{\max\left\{\frac{1}{\gamma}, \frac{\nu(1+\gamma)}{2\nu(1+\gamma)-1}\right\}},\end{aligned} \tag{3.13}$$

where $c_2 > 0$.

According to (3.12) and (3.13),

$$\begin{aligned}Pred_k &= \|F_k\|^2 - \|F_k + J_k d_k\|^2 \\ &= (\|F_k\| + \|F_k + J_k d_k\|)(\|F_k\| - \|F_k + J_k d_k\|) \\ &\geq \|F_k\|(\|F_k\| - \|F_k + J_k d_k\|) \\ &\geq c_3\,\|F_k\|\,\|d_k\|^{\max\left\{\frac{1}{\gamma}, \frac{2}{2\gamma(1+\nu)-1}, \frac{\nu(1+\gamma)}{2\nu(1+\gamma)-1}\right\}},\end{aligned} \tag{3.14}$$

where $c_3 = min\{c_1, c_2\}$.

By (2.25) and (3.14), we have

$$\begin{aligned}|r_k - 1| &= \left|\frac{Ared_k - Pred_k}{Pred_k}\right| \\ &= \left|\frac{(\|F_k\|^2 - \|F(x_k + d_k)\|^2) - (\|F_k\|^2 - \|F_k + J_k d_k\|^2)}{\|F_k\|^2 - \|F_k + J_k d_k\|^2}\right| \\ &= \left|\frac{\|F_k + J_k d_k\|^2 - \|F(x_k + d_k)\|^2}{\|F_k\|^2 - \|F_k + J_k d_k\|^2}\right| \\ &\leq \frac{\frac{2c_h}{1+\nu}\|d_k\|^{1+\nu}\|F_k + J_k d_k\| + \frac{{c_h}^2}{1+\nu}\|d_k\|^{2+2\nu}}{c_5\,\|F_k\|\,\|d_k\|^{\max\left\{\frac{1}{\gamma}, \frac{2}{2\gamma(1+\nu)-1}, \frac{\nu(1+\gamma)}{2\nu(1+\gamma)-1}\right\}}} \to 0.\end{aligned}$$

By (1.7) and (1.8),

$$\frac{\bar{A}red_k}{Pred_k} = \frac{\|F_{l(k)}\|^2 - \|F(x_k + d_k)\|^2}{Pred_k} \geq \frac{\|F_k\|^2 - \|F(x_k + d_k)\|^2}{Pred_k} = r_k \to 1. \tag{3.15}$$

According to Algorithm 1, we know that there exists a positive constant $\bar{\mu} > m$ such that $\mu_k \leq \bar{\mu}$ for all large k. The proof is complete. □

Next, we deduce the convergence rate of Algorithm 1 using the singular value decomposition (SVD) and matrix perturbation theory. According to the conclusions given in [19], without loss of generality, it is assumed that $rank(J(\bar{x})) = r$ for all $\bar{x} \in N(x^*, b) \bigcap X^*$. Suppose the SVD of $J(\bar{x}_k)$ is

$$J(\bar{x}_k) = \bar{U}_k\bar{\Sigma}_k\bar{V}_k^T = (\bar{U}_{k,1}, \bar{U}_{k,2})\begin{pmatrix} \bar{\Sigma}_{k,1} & \\ & 0 \end{pmatrix}\begin{pmatrix} \bar{V}_{k,1}^T \\ \bar{V}_{k,2}^T \end{pmatrix} = \bar{U}_{k,1}\bar{\Sigma}_{k,1}\bar{V}_{k,1}^T,$$

where $\bar{\Sigma}_1 = diag\,(\bar{\sigma}_1, \bar{\sigma}_2, \ldots, \bar{\sigma}_r)$ with $\bar{\sigma}_1 \geq \bar{\sigma}_2 \geq \ldots \geq \bar{\sigma}_r > 0$ and $\bar{U}_k, \bar{V}_k$ are two orthogonal matrices.

Correspondingly, we consider the SVD of J_k by

$$J_k = U_k\Sigma_kV_k^T = (U_{k,1}, U_{k,2})\begin{pmatrix} \Sigma_{k,1} & \\ & \Sigma_{k,2} \end{pmatrix}\begin{pmatrix} V_{k,1}^T \\ V_{k,2}^T \end{pmatrix} = U_{k,1}\Sigma_{k,1}V_{k,1}^T + U_{k,2}\Sigma_{k,2}V_{k,2}^T,$$

where $\Sigma_{k,1} = diag\,(\sigma_{k,1}, \sigma_{k,2}, \ldots, \sigma_{k,r})$ with $\sigma_{k,1} \geq \sigma_{k,2} \geq \ldots \geq \sigma_{k,r} > 0$ and $\Sigma_{k,2} = diag\,(\sigma_{k,r+1}, \sigma_{k,r+2}, \ldots, \sigma_{k,n})$ with $\sigma_{k,r+1} \geq \sigma_{k,r+2} \geq \ldots \geq \sigma_{k,n} \geq 0$.

In the following, if the context is clear, we suppress the subscript k in $U_{k,i}, \Sigma_{k,i}$, and $V_{k,i}$, and we write

$$J_k = U_1\Sigma_1V_1^T + U_2\Sigma_2V_2^T. \tag{3.16}$$

Lemma 3.3. *Under the conditions of Assumption 3.1 and Assumption 3.2, for all sufficiently large k, we have*
(a)$\left\|U_1U_1^TF_k\right\| \leq O\left(\|\bar{x}_k - x_k\|\right)$;
(b)$\left\|U_2U_2^TF_k\right\| \leq O\left(\|x_k - \bar{x}_k\|^{1+\nu}\right)$;
(c)$\|F_k + J_kd_k\| \leq O(\|\bar{x}_k - x_k\|^{\min\{2,1+\nu\}})$.

Proof. The proof of (a) and (b) is similar to the proof of Lemma 3.4 in [15], so we omit it here and only prove (c).

According to the definition of d_k and (3.16), we may obtain

$$d_k = -V_1\left(\Sigma_1^2 + \lambda_kI\right)^{-1}\Sigma_1^TU_1^TF_k - V_2\left(\Sigma_2^2 + \lambda_kI\right)^{-1}\Sigma_2^TU_2^TF_k, \tag{3.17}$$

and

$$\begin{aligned} F_k + J_kd_k &= F_k - U_1\Sigma_1\left(\Sigma_1^2 + \lambda_kI\right)^{-1}\Sigma_1^TU_1^TF_k - U_2\Sigma_2\left(\Sigma_2^2 + \lambda_kI\right)^{-1}\Sigma_2^TU_2^TF_k \\ &= \lambda_kU_1\left(\Sigma_1^2 + \lambda_kI\right)^{-1}U_1^TF_k + \lambda_kU_2\left(\Sigma_2^2 + \lambda_kI\right)^{-1}\Sigma_2^TU_2^TF_k. \end{aligned} \tag{3.18}$$

Without loss of generality, the sequence $\{x_k\}$ converges to X^*, and we assume that $c_h\|\bar{x}_k - x_k\| \leq \frac{\bar{\sigma_k}^2}{2}$ and

$$\left\|\left(\Sigma_1^2 + \lambda_kI\right)^{-1}\right\| \leq \left\|\Sigma_1^{-2}\right\| \leq \frac{1}{(\bar{\sigma}_k - c_h\|x_k - \bar{x}\|)^2} < \frac{4}{\bar{\sigma_k}^2}, \quad \left\|\left(\Sigma_2^2 + \lambda_kI\right)^{-1}\right\| \leq \frac{1}{\lambda_k}. \tag{3.19}$$

For sufficiently large k, there is $\delta_k = 1 + \frac{1}{\ln(k+e)}, \delta_k \in (1, 2]$. By (3.5), (3.11), and $F(\bar{x}_k) = 0$, we may obtain

$$\lambda_k = \frac{\mu_k\|F_k\|^{\delta_k}}{1 + \left\|J_k^TF_k\right\|^{\delta_k}} \leq \mu_k\|F_k\|^{\delta_k} \leq \tilde{\mu}c_J^{\delta_k}\|\bar{x}_k - x_k\|^{\delta_k}. \tag{3.20}$$

According (3.18), (3.19), (3.20), and Lemma 3.3, we may obtain

$$\begin{aligned} \|F_k + J_k d_k\| &\le \frac{4}{\bar{\sigma}_k}\lambda_k \left\|U_1 U_1^T F_k\right\| + \left\|U_2 U_2^T F_k\right\| \\ &\le \frac{4\bar{\mu}\max\{c_J, c_J^2\}}{\bar{\sigma}_k}\|\bar{x}_k - x_k\| O(\|\bar{x}_k - x_k\|) + O(\|\bar{x}_k - x_k\|^{1+\nu}) \\ &= O(\|\bar{x}_k - x_k\|^{\min\{2,1+\nu\}}). \end{aligned} \tag{3.21}$$

That completes the proof. □

Theorem 3.1. *Under Lemma 3.1, Assumption 3.1, and Assumption 3.2, if* $\nu \ge \frac{1}{2\gamma}$ *and* $\nu > \max\left\{\frac{1}{\gamma} - 1, \frac{2}{2\gamma(1+\nu)-1} - 1, \frac{\nu(1+\gamma)}{2\nu(1+\gamma)-1} - 1\right\}$ *, the sequence* x_k *generated by Algorithm 1 converges to the solution of (1.1) with order* $\gamma(1+\nu)$*.*

Proof. According to (3.1),

$$c\|\bar{x}_{k+1} - x_{k+1}\| \le \|F(x_k + d_k)\|^{\gamma}.$$

From (3.4), (3.21), and (3.6), we have

$$\begin{aligned} \|F(x_k + d_k)\| &\le \|F(x_k + d_k) - (F_k + J_k d_k)\| + \|F_k + J_k d_k\| \\ &\le \frac{c_h}{1+\nu}\|d_k\|^{1+\nu} + \|F_k + J_k d_k\| \\ &\le O\left(\|\bar{x}_k - x_k\|^{min\{1+\nu,(1+\nu)(1+\nu-\frac{1}{2\gamma})\}}\right) + O\left(\|\bar{x}_k - x_k\|^{min\{2,1+\nu\}}\right) \\ &\le O\left(\|\bar{x}_k - x_k\|^{\min\{2,1+\nu,(1+\nu)(1+\nu-\frac{1}{2\gamma})\}}\right). \end{aligned}$$

Thus,

$$c^{\frac{1}{\gamma}}\|\bar{x}_{k+1} - x_{k+1}\|^{\frac{1}{\gamma}} \le O\left(\|\bar{x}_k - x_k\|^{\min\{2,1+\nu,(1+\nu)(1+\nu-\frac{1}{2\gamma})\}}\right). \tag{3.22}$$

Consider $\nu \in (0,1]$, $\gamma \in (0,1]$, and $\nu > \frac{1}{2\gamma}$. We have

$$2 - (1+\nu) = 1 - \nu > 0$$

and

$$(1+\nu) - \left((1+\nu)(1+\nu-\frac{1}{2\gamma})\right) = (1+\nu)(\frac{1}{2\gamma} - \nu) < 0.$$

Therefore,

$$\|\bar{x}_{k+1} - x_{k+1}\| \le O(\|\bar{x}_k - x_k\|^{\gamma(1+\nu)}). \tag{3.23}$$

This means that when $\nu > \frac{1}{\gamma} - 1$, the sequence x_k converges to the solution of (1.1) with the rate of $\gamma(1+\nu) > 1$.

Since $\|\bar{x}_k - x_k\| \le \|\bar{x}_{k+1} - x_k\| \le \|\bar{x}_{k+1} - x_{k+1}\| + \|d_k\|$ and (3.23) holds, there exists a constant C such that

$$\|\bar{x}_k - x_k\| \le C\|d_k\|$$

for all sufficiently large k. It is clear that

$$\|d_{k+1}\| \le O(\|d_k\|^{\gamma(1+\nu)}). \tag{3.24}$$

Therefore, when $\nu > \frac{1}{\gamma} - 1$, Algorithm 1 is convergent with order $\gamma(1+\nu) > 1$. The proof is complete. □

Remark 3.1. *The parameter $\gamma \in (0,1]$ is denoted as the order of the Hölderian local error bound of $F(x)$, and the parameter $\nu \in (0,1]$ is denoted as the order of the Hölderian continuity of the Jacobian matrix $J(x)$. Under the condition of Theorem 3.1, when $\nu \ge \frac{1}{2\gamma}$ and $\nu > \max\left\{\frac{1}{\gamma} - 1, \frac{2}{2\gamma(1+\nu)-1} - 1, \frac{\nu(1+\gamma)}{2\nu(1+\gamma)-1} - 1\right\}$, the algorithm converges with order $\gamma(1+\nu)$. In particular, the convergence rates can be described as follows:*

$$\|d_{k+1}\| \le \begin{cases} O\left(\|d_k\|^{1+\nu}\right), & \text{if } \gamma = 1, \\ O\left(\|d_k\|^{2\gamma}\right), & \text{if } \nu = 1, \\ O\left(\|d_k\|^{2}\right), & \text{if } \nu = 1 \text{ and } \gamma = 1. \end{cases}$$

It is easy to see that if $\gamma = 1$, the sequence $\{x_k\}$ converges superlinearly with order $1+\nu$. If $\gamma \in [\frac{1}{2}, 1)$ and $\nu = 1$, the convergence rate is 2γ. If $\nu = 1$ and $\gamma = 1$, the convergence rate is quadratic.

4. Numerical experiments

In this section, we test the nonmonotone modified Levenberg–Marquardt method (NMLM) with some numerical experiments. All tests are performed on a computer with an Intel Core i5-8250U CPU with 12.0 GB RAM and MATLAB R2023b (64-bit). We compare the NMLM algorithm with the modified Levenberg–Marquardt algorithm for nonlinear equations (MLM, Algorithm 2.2) in [11], the new Levenberg–Marquardt method (NLM, Algorithm 2.1) in [14], and the modified efficient Levenberg–Marquardt method (MELM, Algorithm 1) in [17].

We choose some standard singular test equations that were created by Frank et al. in [20],

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

where the function $F(x)$ is the standard nonsingular function in [21]. Here, x^* is the solution of $\|F(x)\| = 0$, $A \in \mathbb{R}^{n\times k}$ and has full column rank with $1 \le k \le n$. It is not difficult to see that the Jacobian matrix of $\hat{F}(x)$ at x^* is

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

and its rank is $n-k$. It is obvious that $A = [1, 1, \cdots, 1]^T \in \mathbb{R}^{n\times 1}$ such that $rank(\hat{J}(x^*)) = n-1$. For the convenience of numerical comparison, we adopt the same test problems as in [11, 14, 17].

The values for p_0, p_1, p_2, and μ_0 are taken as the same empirical values as the compared algorithms. We set the parameters $p_0 = 10^{-4}$, $p_1 = 0.25$, $p_2 = 0.75$, $\mu_0 = 1$, and $m = 10^{-8}$ for the test. The termination condition is $\left\|J_k^T F_k\right\| \le 10^{-6}$ and $k \ge 1000$, where 1000 is set as the maximum tolerance for iteration, and $k \ge 1000$ indicates the algorithm failed.

The results of the numerical experiments are listed in Tables 1 and 2. In the two tables, the first

column shows the names of the test functions. The second column lists the five initial points $-10x_0$, $-x_0$, x_0, $10x_0$, and $100x_0$, where x_0 is the standard initial point recommended in the literature [20] and [21]. For brevity, the five initial points are written as -10, -1, 1, 10, and 100. The third column lists the dimension of the function. The other columns list the numerical results of the NMLM, MLM, NLM, and MELM algorithms. Specially, the meanings of NF, NJ, NT, Iter, and CPU are as follows:
NF: the number of function evaluations
NJ: the number of Jacobian evaluations
NT: NT = NF + NJ$^*n$
Iter: the number of iterations
CPU: CPU time
-: Indicates that the number of iterations is more than 1000 and the algorithm fails.

Table 1. Numerical experiments for singular problems with standard values n.

			NMLM					MLM					NLM					MELM				
Function	x_0	n	NF	NJ	NT	Iter	CPU	NF	NJ	NT	Iter	CPU	NF	NJ	NT	Iter	CPU	NF	NJ	NT	Iter	CPU
Rosenbrock	-10	2	18	18	54	17	0.031	167	167	501	166	0.063	16	16	48	15	0	16	16	48	15	0.016
	-1	2	16	16	48	15	0	154	154	462	153	0.016	17	17	51	16	0	76	76	228	75	0.063
	1	2	17	17	51	16	0	176	176	528	175	0.031	18	18	54	17	0	79	79	237	78	0
	10	2	19	19	57	18	0	107	107	321	106	0	19	19	57	18	0	19	19	57	18	0
	100	2	22	22	66	21	0	38	38	114	37	0	22	22	66	21	0.016	22	22	66	21	0
Powell singular	-10	4	21	21	105	20	0	21	21	105	20	0	21	21	105	20	0	21	21	105	20	0
	-1	4	17	17	85	16	0	17	17	85	16	0	17	17	85	16	0	17	17	85	16	0
	1	4	17	17	85	16	0	17	17	85	16	0	17	17	85	16	0	17	17	85	16	0
	10	4	21	21	105	20	0.063	21	21	105	20	0	21	21	105	20	0	21	21	105	20	0
	100	4	24	24	120	23	0	24	24	120	23	0	24	24	120	23	0	24	24	120	23	0
Wood function	-10	4	20	20	100	19	0	21	21	105	20	0	20	20	100	19	0	20	20	100	19	0
	-1	4	17	17	85	16	0	17	17	85	16	0	17	17	85	16	0	17	17	85	16	0
	1	4	18	18	90	17	0	18	18	90	17	0	18	18	90	17	0	18	18	90	17	0
	10	4	20	20	100	19	0	20	20	100	19	0	20	20	100	19	0	20	20	100	19	0
	100	4	24	24	120	23	0	22	22	110	21	0	24	24	120	23	0.016	24	24	120	23	0
Variable dimensioned	-10	10	18	18	198	17	0	18	18	198	17	0	18	18	198	17	0	18	18	198	17	0.016
	-1	10	16	16	176	15	0	16	16	176	15	0	16	16	176	15	0	16	16	176	15	0
	1	10	15	15	165	14	0	15	15	165	14	0.016	15	15	165	14	0.047	15	15	165	14	0
	10	10	17	17	187	16	0	17	17	187	16	0	17	17	187	16	0.031	17	17	187	16	0
	100	10	21	21	231	20	0	21	21	231	20	0	21	21	231	20	0	21	21	231	20	0
Brown almost_linear	-10	10	23	23	253	22	0	23	23	253	22	0	23	23	253	22	0	23	23	253	22	0
	-1	10	9	9	99	8	0	11	11	121	10	0.016	10	10	110	9	0	10	10	110	9	0
	1	10	9	9	99	8	0	10	10	110	9	0	9	9	99	8	0	9	9	99	8	0
	10	10	24	24	264	23	0	24	24	264	23	0	24	24	264	23	0	24	24	264	23	0.031
	100	10	76	45	526	44	0.125	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Discrete boundary value	-10	10	13	13	143	12	0.141	14	14	154	13	0	16	16	176	15	0	13	13	143	12	0
	-1	10	39	25	289	24	0.047	12	12	132	11	0	14	14	154	13	0	13	13	143	12	0
	1	10	47	28	327	27	0	12	12	132	11	0	16	16	176	15	0	14	14	154	13	0
	10	10	10	10	110	9	0	12	12	132	11	0	11	11	121	10	0	12	12	132	11	0
	100	10	12	12	132	11	0	13	13	143	12	0	13	13	143	12	0.016	-	-	-	-	-

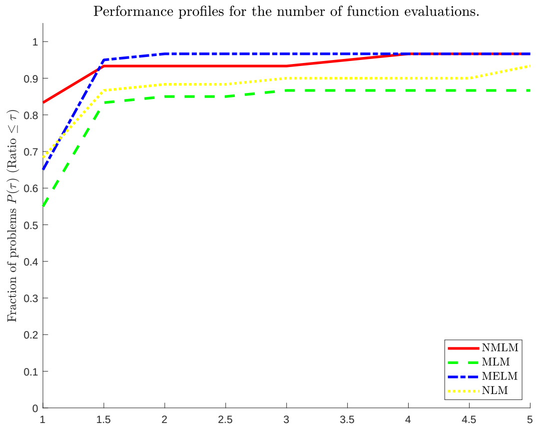
We compare the numerical results for the low-dimensional problems in Table 1. For example, the third line lists the numerical results of the Rosenbrock problem with the initial point $-x_0$ and two dimensions. The indexes NF, NJ, NT, Iter, and CPU of NMLM are 16, 16, 48, 15, and 0, respectively. That of MLM are 154, 154, 462, 153, and 0.016. That of NLM are 17, 17, 51, 16, and 0. That of MELM are 76, 76, 228, 75, and 0.063. For the Rosenbrock problem, the Brown almost linear problem and the discrete boundary value problem, the NMLM algorithm wins in terms of NF, NJ, NT, and Iter, compared with the MLM, NLM, and MELM algorithms. For the Brown function with $100x_0$, the NMLM algorithm runs successfully, and its indexes NF, NJ, NT, Iter, and CPU of NMLM are 76, 45, 526, 44, and 0.125. The other algorithms terminate at $k = 1000$ and fail. For the other problems, the NMLM algorithm performs a similar number of iterations to the other algorithms.

We also compare the numerical experiments on 30 problems with $n = 500$ in Table 2. The MLM algorithm fails to run in the Broyden banded problem. For the discrete boundary problem, the extended Rosenbrock problem, and the Broyden banded problem, the NMLM algorithm wins in terms of NF, NJ, NT, and Iter. For the other problems, the numerical results of the NMLM algorithm are similar to those of the other algorithms. Moreover, NMLM can successfully solve some problems in which the other algorithms fail. These results demonstrate the effectiveness of the proposed algorithm.

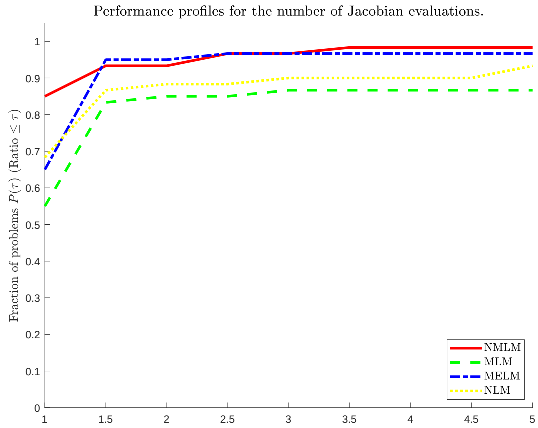
Table 2. Numerical experiments for singular problems of $n = 500$.

			NMLM					MLM					NLM					MELM				
Function	x_0	n	NF	NJ	NT	Iter	CPU	NF	NJ	NT	Iter	CPU	NF	NJ	NT	Iter	CPU	NF	NJ	NT	Iter	CPU
Variable dimensioned	-10	500	32	32	16032	31	2.172	32	32	16032	31	3.094	37	37	18537	36	4.203	32	32	16032	31	2.797
	-1		30	30	15030	29	1.922	30	30	15030	29	1.891	30	30	15030	29	2.750	30	30	15030	29	2.422
	1		29	29	14529	28	1.859	29	29	14529	28	2.125	29	29	14529	28	2.984	29	29	14529	28	2.406
	10		31	31	15531	30	1.938	31	31	15531	30	2.422	31	31	15531	30	4.203	31	31	15531	30	2.516
	100		43	35	17543	34	3.266	35	35	17535	34	2.406	42	42	21042	41	5.469	35	35	17535	34	2.859
Discrete boundary	-10	500	9	9	4509	8	0.375	10	10	5010	9	0.672	10	10	5010	9	0.828	10	10	5010	9	0.781
	-1		5	5	2505	4	0.188	7	7	3507	6	0.422	7	7	3507	6	0.453	7	7	3507	6	0.313
	1		5	5	2505	4	0.188	7	7	3507	6	0.375	7	7	3507	6	0.516	7	7	3507	6	0.469
	10		15	15	7515	14	2.828	12	12	6012	11	2.094	11	11	5511	10	2.828	11	11	5511	10	2.031
	100		17	17	8517	16	2.813	24	24	12024	23	4.141	18	18	9018	17	4.406	26	26	13026	25	4.891
Extended Rosenbrock	-10	500	20	20	10020	19	0.906	24	24	12024	23	1.734	18	18	9018	17	3.094	18	18	9018	17	1.125
	-1		19	19	9519	18	0.953	208	208	104208	207	20.656	20	20	10020	19	1.516	141	141	70641	140	8.984
	1		20	20	10020	19	0.938	220	220	110220	219	20.906	20	20	10020	19	1.719	55	55	27555	54	3.234
	10		21	21	10521	20	1.063	57	57	28557	56	3.375	21	21	10521	20	3.328	21	21	10521	20	1.359
	100		24	24	12024	23	1.188	33	33	16533	32	2.109	24	24	12024	23	3.734	24	24	12024	23	1.594
Extended Powell singular	-10	500	15	15	7515	14	0.625	16	16	8016	15	1.031	15	15	7515	14	2.531	15	15	7515	14	0.875
	-1		12	12	6012	11	0.516	13	13	6513	12	1.563	12	12	6012	11	0.844	12	12	6012	11	0.750
	1		12	12	6012	11	0.563	13	13	6513	12	1.484	12	12	6012	11	0.984	12	12	6012	11	0.750
	10		15	15	7515	14	0.672	16	16	8016	15	0.875	15	15	7515	14	2.359	15	15	7515	14	0.906
	100		19	19	9519	18	1.031	19	19	9519	18	1.203	19	19	9519	18	3.719	19	19	9519	18	1.109
Trigonometric	-10	500	11	11	5511	10	0.906	11	11	5511	10	1.203	11	11	5511	10	3.656	11	11	5511	10	1.219
	-1		9	9	189	8	0	9	9	189	8	0	9	9	189	8	0	9	9	189	8	0
	1		8	8	4008	7	0.563	8	8	4008	7	1.594	8	8	4008	7	1.234	8	8	4008	7	0.844
	10		336	197	98836	196	22.203	19	19	9519	18	2.094	149	149	74649	148	57.516	22	22	11022	21	2.953
	100		194	110	55194	109	27.203	35	35	17535	34	4.234	48	48	24048	47	16.359	40	40	20040	39	7.453
Broyden banded function	-10	500	18	18	9018	17	0.750	18	18	9018	17	21.484	18	18	9018	17	45.094	18	18	9018	17	15.750
	-1		18	15	7518	14	0.734	-	-	-	-	-	34	34	17034	33	81.781	970	970	485970	969	1214.750
	1		9	9	4509	8	0.281	9	9	4509	8	12.047	9	9	4509	8	20.734	9	9	4509	8	7.422
	10		14	14	7014	13	0.500	13	13	6513	12	12.031	14	14	7014	13	36.453	14	14	7014	13	17.000
	100		20	20	10020	19	1.500	20	20	10020	19	25.547	20	20	10020	19	54.922	20	20	10020	19	25.578

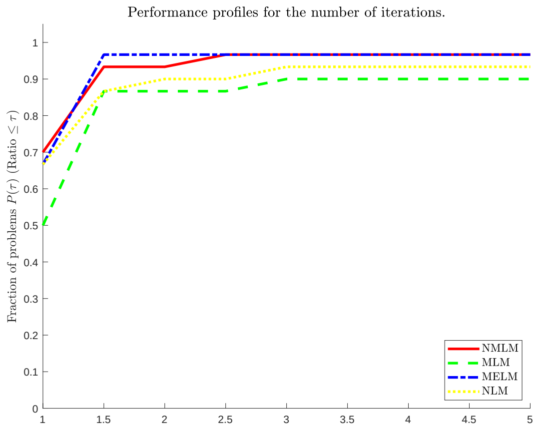
Based on Table 1 and Table 2, Figure 1 shows the performance curves in terms of NF, NJ, Iter, and CPU. Its vertical axis represents the proportion of problems solved , which is denoted as $P(\tau)$, and the horizontal axis represents the performance ratio τ. When $\tau = 1$, a larger value of $P(\tau)$ indicates that the algorithm wins a higher proportion of the test problems. Based on Figure 1(a), we observe that the NMLM algorithm wins on 83.33% of the problems at $\tau = 1$, achieving the minimal NF. From Figure 1(b), (c), and (d), we note that the NMLM algorithm requires the fewest NJ in 85% of the test problems, 70% using the fewest Iter, and 73.33% using the least CPU. The numerical results show that the NMLM algorithm achieves better performance, requiring less time and fewer computations.



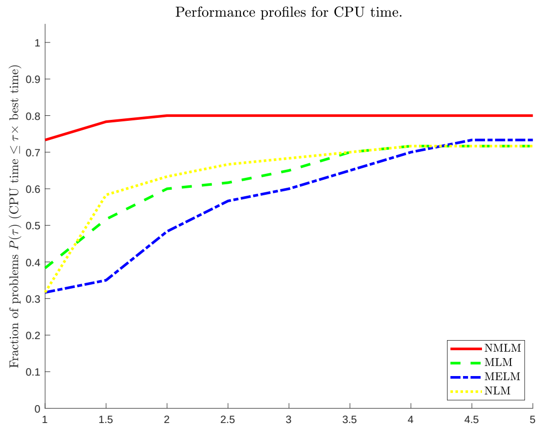
(a) Performance profiles for the number of function evaluations.



(b) Performance profiles for the number of Jacobian evaluations.



(c) Performance profiles for the number of iterations.



(d) Performance profiles for the CPU time.

Figure 1. Performance profiles of the numerical results.

5. Conclusions

In this paper, a modified LM method with new LM parameters $\lambda_k = \frac{\mu_k \|F_k\|^{\delta_k}}{1+\|J_k^T F_k\|^{\delta_k}}$ is proposed, and a nonmonotone technique is applied. Under the Hölderian continuity and the Hölderian local error bounds conditions, we prove the global convergence of the algorithm and show that it converges with order of $\gamma(1+\nu) > 1$. We perform numerical experiments on 60 problems and compare the NMLM algorithm with the NLM, MELM, and MLM algorithms. The numerical results demonstrate that our algorithm is competitive.

Author contributions

Yuya Zheng: conceptualization, writing–original draft, software, methodology, writing–review and editing; Yueting Yang: writing–original draft, supervision, funding acquisition, methodology, project administration, writing–review and editing; Mingyuan Cao: writing–original draft, supervision, funding acquisition, methodology, project administration, writing–review and editing. All authors have read and approved the final version of the manuscript for publication.

Use of AI tools declaration

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

Acknowledgments

The authors would like to thank the Youth Science and Technology Talent Cultivation Project of Jilin Province (20250602005RC); and the Graduate Innovation Project of Beihua University (2024005, 2024040).

Conflict of interest

The authors declare no conflicts of interest.

References

1. Y. L. Shang, D. Q. Qu, J. X. Li, R. X. Zhang, A new parameter-free continuously differentiable filled function algorithm for solving nonlinear equations and data fitting problems, *J. Comput. Appl. Math.*, **454** (2025), 116198. https://doi.org/10.1016/j.cam.2024.116198
2. M. C. Huang, Y. N. Zhang, S. Li, Mobile localization based on online solution of linear matrix-vector equations using inverse-free acceleration-layer Zhang neurodynamics, *Expert Syst. Appl.*, **278** (2025), 127361. https://doi.org/10.1016/j.eswa.2025.127361
3. M. Liu, Y. F. Hu, J. C. Li, L. Jin, Characteristics-capturing neural dynamics for solving time-dependent nonlinear equations with periodic noise, *IEEE Trans. Cognit. Dev. Syst.*, **16** (2024), 984–992. https://doi.org/10.1109/TCDS.2023.3316776
4. T. Y. Song, Z. X. Liu, An efficient inertial subspace minimization CG algorithm with convergence rate analysis for constrained nonlinear monotone equations, *J. Comput. Appl. Math.*, **446** (2024), 115873. https://doi.org/10.1016/j.cam.2024.115873
5. M. Yuan, L. Du, F. Jiang, J. C. Bai, G. R. Chen, Reweighted alternating direction method of multipliers for DNN weight pruning, *Neural Networks*, **179** (2024), 106534. https://doi.org/10.1016/j.neunet.2024.106534
6. K. Z. Zheng, S. Li, Y. N. Zhang, Low-computational-complexity zeroing neural network model for solving systems of dynamic nonlinear equations, *IEEE Trans. Autom. Control*, **69** (2024), 4368–4379. https://doi.org/10.1109/TAC.2023.3319132

7. J. Nocedal, S. J. Wright, Numerical optimization, 2nd Ed., New York: Springer, 2006. https://doi.org/10.1007/978-0-387-40065-5
8. D. W. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *J. Soc. Ind. Appl. Math.*, **11** (1963), 431–441. https://doi.org/10.1137/0111030
9. K. Levenberg, A method for the solution of certain non-linear problems in least squares, *Quart. Appl. Math.*, **2** (1944), 164–168. https://doi.org/10.1090/qam/10666
10. N. Yamashita, M. Fukushima, On the rate of convergence of the Levenberg-Marquardt method, in *Topics in Numerical Analysis*, Vienna: Springer, 2001, 239–249. https://doi.org/10.1007/978-3-7091-6217-0_18
11. J. Y. Fan, A modified Levenberg-Marquardt algorithm for singular system of nonlinear equations, *J. Comput. Math.*, **21** (2003), 625–636.
12. J. Y. Fan, Y. X. Yuan, On the quadratic convergence of the Levenberg-Marquardt method without nonsingularity assumption, *Computing*, **74** (2005), 23–39. https://doi.org/10.1007/s00607-004-0083-1
13. K. Amini, F. Rostami, A modified two steps Levenberg–Marquardt method for nonlinear equations, *Int. J. Comput. Appl. Math.*, **288** (2015), 341–350. https://doi.org/10.1016/j.cam.2015.04.040
14. K. Amini, F. Rostami, G. Caristi, An efficient Levenberg–Marquardt method with a new LM parameter for systems of nonlinear equations, *Optimization*, **67** (2018), 637–650. https://doi.org/10.1080/02331934.2018.1435655
15. H. Y. Wang, J. Y. Fan, Convergence rate of the Levenberg–Marquardt method under Hölderian local error bound, *Optim. Methods. Softw.*, **35** (2020), 767–786. https://doi.org/10.1080/10556788.2019.1694927
16. L. Chen, Y. F. Ma, A modified Levenberg–Marquardt method for solving system of nonlinear equations, *J. Appl. Math. Comput.*, **69** (2023), 2019–2040. https://doi.org/10.1007/s12190-022-01823-x
17. M. L. Zeng, G. H. Zhou, Improved convergence results of an efficient Levenberg–Marquardt method for nonlinear equations, *J. Appl. Math. Comput.*, **68** (2022), 3655–3671. https://doi.org/10.1007/s12190-021-01599-6
18. M. J. D. Powell, Convergence properties of a class of minimization algorithms, in *Nonlinear Programming 2,* 1975, 1–27. https://doi.org/10.1016/B978-0-12-468650-2.50005-5
19. R. Behling, A. Iusem, The effect of calmness on the solution set of systems of nonlinear equations, *Math. Program.*, **137** (2013), 155–165. https://doi.org/10.1007/s10107-011-0486-7
20. R. B. Schnabel, P. D. Frank, Tensor methods for nonlinear equations, *SIAM J. Numer. Anal.*, **21** (1984), 815–843. https://doi.org/10.1137/0721054
21. J. J. Moré, B. S. Garbow, K. E. Hillstrom, Testing unconstrained optimization software, *ACM Trans. Math. Softw.*, **7** (1981), 17–41. https://doi.org/10.1145/355934.355936

AIMS Press