

Newton's Method for M-Tensor Equations*

Dong-Hui Li Jie-Feng Xu and Hong-Bo Guan

School of Mathematical Sciences,
South China Normal University, Guangzhou, 510631, China.

Email: lidonghui@m.scnu.edu.cn

January 28, 2021

Abstract

We are concerned with the tensor equations whose coefficient tensor is an M-tensor. We first propose a Newton method for solving the equation with a positive constant term and establish its global and quadratic convergence. Then we extend the method to solve the equation with a nonnegative constant term and establish its convergence. At last, we do numerical experiments to test the proposed methods. The results show that the proposed method is quite efficient.

Keywords M-tensor equation, Newton method, quadratic convergence

1 Introduction

Newton's method is a famous iterative method for solving nonlinear equations. An attractive property of the method is its superlinear/quadratic convergence if the Jacobian of the residual function is nonsingular at the solution. However, when the Jacobian of the function is singular, the method may loss its fast convergence property or even fail to work. As an example, we consider the following system of nonlinear equations

$$F_i(x) = \sum_{j=1}^n a_{ij}x_j^{m-1} - b_i = 0, \quad i = 1, 2, \dots, n. \quad (1.1)$$

When matrix $A = (a_{ij})$ is nonsingular and $b = (b_1, \dots, b_n)^T \geq 0$, the equation has solutions satisfying $\bar{x}_i = (A^{-1}b)_i^{1/(m-1)}$, $i = 1, 2, \dots, n$. If there is some $\bar{x}_i = 0$, then the Jacobian $F'(\bar{x})$ is singular. As a result, the Newton method may loss its superlinear/quadratic convergence or even be failure.

The last equation (1.1) is a special tensor equation. In this paper, we will consider the following general tensor equation

$$F(x) = \mathcal{A}x^{m-1} - b = 0, \quad (1.2)$$

*Supported by the NSF of China grant No.11771157, 11801184, the NSF of Guangdong Province grant No.2020B1515310013 and the Education Department of Hunan Province grant No.20C0559.

where $x, b \in \mathbb{R}^n$ and \mathcal{A} is an m th-order n -dimensional tensor consists of n^m elements:

$$\mathcal{A} = (a_{i_1 i_2 \dots i_m}), \quad a_{i_1 i_2 \dots i_m} \in \mathbb{R}, \quad 1 \leq i_1, i_2, \dots, i_m \leq n,$$

and $\mathcal{A}x^{m-1} \in \mathbb{R}^n$ with elements

$$(\mathcal{A}x^{m-1})_i = \sum_{i_2, \dots, i_m=1}^n a_{ii_2 \dots i_m} x_{i_2} \cdots x_{i_m}, \quad i = 1, 2, \dots, n.$$

We will pay particular attention to the M-tensor equation (1.2) in which \mathcal{A} is an M-tensor. To give the definition of M-tensor, we first introduce some concepts. We refer two recent books [23, 24] for details.

We denote the set of all m th-order n -dimensional tensors by $\mathcal{T}(m, n)$ and $[n] = \{1, 2, \dots, n\}$.

A tensor $\mathcal{A} = (a_{i_1 i_2 \dots i_m}) \in \mathcal{T}(m, n)$ is called non-negative tensor, denoted by $\mathcal{A} \geq 0$, if all its elements are non-negative, i.e., $a_{i_1 i_2 \dots i_m} \geq 0, \forall i_1, \dots, i_m \in [n]$. \mathcal{A} is called the identity tensor, denoted by \mathcal{I} , if its diagonal elements are all ones and other elements are zeros, i.e., all $a_{i_1 i_2 \dots i_m} = 0$ except $a_{ii \dots i} = 1, \forall i, i_1, \dots, i_m \in [n]$.

If a real number λ and a nonzero real vector $x \in \mathbb{R}^n$ satisfy

$$\mathcal{A}x^{m-1} = \lambda x^{[m-1]},$$

then λ is called an H-eigenvalue of \mathcal{A} and x is an H-eigenvector of \mathcal{A} associated with λ . Here, for a real scalar α , $x^{[\alpha]} = (x_1^\alpha, x_2^\alpha, \dots, x_n^\alpha)$ whenever it is meaningful.

A tensor $\mathcal{A} = (a_{i_1 i_2 \dots i_m}) \in \mathcal{T}(m, n)$ is symmetric if its elements $a_{i_1 i_2 \dots i_m}$ are invariant under any permutation of their indices. The set of all m th-order n -dimensional symmetric tensors is denoted by $\mathcal{ST}(m, n)$. \mathcal{A} is called semi-symmetric if for any $i \in [n]$, the sub-tensor $\mathcal{A}_i := (a_{ii_2 \dots i_m})_{1 \leq i_2, \dots, i_m \leq n}$ is symmetric. In the case $\mathcal{A} \in \mathcal{ST}(m, n)$, we have

$$\nabla(\mathcal{A}x^m) = m\mathcal{A}x^{m-1}.$$

In the case $\mathcal{A} \in \mathcal{T}(m, n)$ is semi-symmetric, we have

$$\nabla(\mathcal{A}x^{m-1}) = (m-1)\mathcal{A}x^{m-2}.$$

The definition of M-tensor is introduced in [6, 24, 30].

Definition 1.1. A tensor $\mathcal{A} \in \mathcal{T}(m, n)$ is called an M-tensor, if it can be written as

$$\mathcal{A} = s\mathcal{I} - \mathcal{B}, \quad \mathcal{B} \geq 0, \quad s \geq \rho(\mathcal{B}), \quad (1.3)$$

where $\rho(\mathcal{B})$ is the spectral radius of tensor \mathcal{B} , that is

$$\rho(\mathcal{B}) = \max \{|\lambda| : \lambda \text{ is an eigenvalue of } \mathcal{B}\}.$$

If $s > \rho(\mathcal{B})$, then \mathcal{A} is called a strong or nonsingular M-tensor.

For $x, y \in \mathbb{R}^n$, we use $x \circ y$ to denote their Hadamard product defined by

$$x \circ y = (x_1 y_1, \dots, x_n y_n)^T.$$

We use \mathbb{R}_+^n and \mathbb{R}_{++}^n to denote the sets of all nonnegative vectors and positive vectors in \mathbb{R}^n . That is,

$$\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x \geq 0\} \quad \text{and} \quad \mathbb{R}_{++}^n = \{x \in \mathbb{R}^n \mid x > 0\}.$$

Tensor equation is also called multi-linear equation. It appears in many practical fields including data mining and numerical partial equations [4, 7, 8, 9, 13, 15, 16, 27].

The study in the numerical methods for solving tensor equation has begun only a few years ago. Most of them focus on solving the M-tensor equation (abbreviated as M-Teq). The existing methods for solving M-Teq focus on finding a positive solution under the restriction $b > 0$. Such as Jacobian and Gauss-Seidal methods [7], the homotopy method [11], tensor splitting method [20], Newton-type method [12], continuous time neural network method [25].

Recently, Bai, He, Ling and Zhou [2] proposed a nonnegativity preserving algorithm to solve M-Teq with $b \geq 0$. Li, Guan and Wang [18] proposed a monotone iterative method to solve the M-Teq with arbitrary b . Li, Guan and Xu [14] proposed an inexact Newton method with $b > 0$ and extended the method to solving the M-Teq with $b \geq 0$.

There are few methods to solve the tensor equation with other structure tensors or more general tensors. Li, Xie and Xu [15] extended the classic splitting methods for solving system of linear equations to solving tensor equations with symmetric tensor. Li, Dai and Gao [17] proposed a alternating projection method for solving tensor equations with a special 3-order tensor. Other related works can also be found in [3, 4, 5, 10, 16, 19, 21, 22, 26, 27, 28, 29].

In this paper, we further study numerical methods for solving M-Teq (1.2). Our purpose is to find a nonnegative solution of the equation with $b \geq 0$. As we know in [14], finding a nonnegative solution of the M-tensor equation can be done by finding a positive solution of a lower dimensional M-tensor equation with nonnegative constant term. It is noting that the constant term of that lower dimensional equation is still not guaranteed to be positive. So most of the existing methods are not able to be applied. We will propose a Newton method to get a positive solution of the equation and prove its global convergence and quadratic convergence. Our numerical results show that the proposed Newton method is very efficient.

In the next section, we propose a Newton method to find the unique positive solution to a M-tensor equation with positive constant term. We will also establish its global and quadratic convergence in Section 2. In section 3, we extend the idea of the method proposed in Section 2 to get a nonnegative solution of an M-tensor equation with nonnegative constant term and establish its convergence. It should be pointed out that such an extension is not trivial because the M-tensor equation with positive and nonnegative constant terms are quite different. At last, in Section 4, we do numerical experiments to test the proposed methods.

2 A Newton Method for M-Tensor Equation (1.2) with $b > 0$

In this section, we propose a Newton method to find the unique positive solution to (1.2) with $b > 0$. Throughout this section, without specification, we always suppose that the following assumption holds.

Assumption 2.1. *Tensor \mathcal{A} in (1.2) is a semi-symmetric and strong M-tensor, and $b > 0$.*

Since our purpose is to get a positive solution of the M-Teq (1.2), we restrict $x \in \mathbb{R}_{++}^n$. Making a variable transformation $y = x^{[m-1]}$, we formulate the M-Teq (1.2) as

$$f(y) = F\left(y^{[\frac{1}{m-1}]}\right) = \mathcal{A}\left(y^{[\frac{1}{m-1}]}\right)^{m-1} - b = 0. \quad (2.1)$$

A direct computation gives

$$f'(y) = \mathcal{A}\left(y^{[\frac{1}{m-1}]}\right)^{m-2} \text{diag}\left(y^{[\frac{1}{m-1}-1]}\right).$$

It follows that

$$f'(y)y = \mathcal{A}\left(y^{[\frac{1}{m-1}]}\right)^{m-2} \text{diag}\left(y^{[\frac{1}{m-1}-1]}\right)y = \mathcal{A}\left(y^{[\frac{1}{m-1}]}\right)^{m-1} = f(y) + b.$$

For $\epsilon \in (0, 1)$ and $\beta > 0$, define

$$\mathcal{F}_\epsilon = \left\{x \in \mathbb{R}_+^n : \mathcal{A}x^{m-1} \geq \epsilon b\right\} = \left\{y \in \mathbb{R}_+^n : \mathcal{A}\left(y^{[\frac{1}{m-1}]}\right)^{m-1} \geq \epsilon b\right\}$$

and

$$\Omega_\beta = \left\{x \in \mathbb{R}^n : \|\mathcal{A}x^{m-1} - b\| \leq \beta\right\} = \left\{y \in \mathbb{R}^n : \left\|\mathcal{A}\left(y^{[\frac{1}{m-1}]}\right)^{m-1} - b\right\| \leq \beta\right\}.$$

It is easy to see that for any $\epsilon \in (0, 1]$, the positive solution of the equation (1.2) is contained in \mathcal{F}_ϵ . In addition, the Jacobian matrices $F'(x)$ and $f'(y)$ are nonsingular M-matrices for any $x, y \in \mathcal{F}_\epsilon$.

Lemma 2.2. *The following statements are true.*

(i) *If $\mathcal{A} = (a_{i_1 \dots i_m})$ is a Z-tensor and $b > 0$, then for any $\epsilon > 0$, the set \mathcal{F}_ϵ is bounded away from zero. That is, there is a constant $\delta > 0$ such that*

$$x \geq \delta \mathbf{e}, \quad \forall x \in \mathcal{F}_\epsilon,$$

where $\mathbf{e} = (1, 1, \dots, 1)^T$.

(ii) *If \mathcal{A} is a strong M-tensor, then for any $\beta \in \mathbb{R}$, the level set Ω_β is bounded.*

Proof. We prove the lemma by contradiction.

(i) Suppose conclusion (i) is not true. Then, there is a sequence $\{x_k\} \subset \mathcal{F}_\epsilon$ and an index $i \in [n]$ such that $\{(x_k)_i\} \rightarrow 0$. Since $x_k \in \mathcal{F}_\epsilon$, it holds that

$$(\epsilon - 1)b_i \leq a_{i \dots i}(x_k)_i^{m-1} + \sum_{(i_2, \dots, i_m) \neq (i, \dots, i)} a_{ii_2 \dots i_m}(x_k)_{i_2} \dots (x_k)_{i_m} - b_i \leq a_{i \dots i}(x_k)_i^{m-1} - b_i.$$

Taking limits in both sides of the last inequality, we get $\epsilon b_i \leq 0$. It is a contradiction. Consequently, the set \mathcal{F}_ϵ is bounded away from zero.

(ii) Suppose that for some $\beta \in \mathbb{R}$, the level set Ω_β is not bounded. Then there is a sequence $\{x_k\} \subset \Omega_\beta$ satisfying $\|x_k\| \rightarrow \infty$, as $k \rightarrow \infty$. However, we obviously have

$$\frac{\beta}{\|x_k^{[m-1]}\|} \geq \frac{\|\mathcal{A}x_k^{m-1} - b\|}{\|x_k^{[m-1]}\|} \geq \left\| \mathcal{A}\left(\frac{x_k}{\|x_k\|}\right)^{m-1} \right\| - \frac{\|b\|}{\|x_k^{[m-1]}\|}.$$

Suppose that the subsequence $\{x_k / \|x_k\|\}_K$ converges to some $\bar{u} \neq 0$. Taking limits as $k \rightarrow \infty$ with $k \in K$ in both sides of the last inequality, we get $\mathcal{A}\bar{u}^{m-1} = 0$. Since \mathcal{A} is a strong M-tensor, from Theorem 2.3 in [14], we get a contradiction. \square

The idea to develop Newton's method is described as follows. Starting from some $y_0 = x_0^{[m-1]}$ satisfying $x_0 \in \mathcal{F}_\epsilon$ with some given small $\epsilon \in (0, 1)$, the method generates a sequence of iterates $\{x_k\} \subset \mathcal{F}_\epsilon$ by a damped Newton iteration such that the residual sequence $\{\|f(y_k)\|\}$ is decreasing.

We first show the following lemma.

Lemma 2.3. *Suppose that \mathcal{A} is a strong M-tensor and $b > 0$. Let d be the Newton direction that is the unique solution of the system of linear equations*

$$f'(y)d + f(y) = 0.$$

Then there is a constant $L > 0$ such that the inequality

$$\mathcal{A}\left[(y + \alpha d)^{[\frac{1}{m-1}]}\right]^{m-1} \geq \epsilon b + \alpha \left((1 - \epsilon)b - \frac{1}{2}L\alpha\|d\|^2 \mathbf{e} \right), \quad \forall \alpha > 0, \forall y \in \mathcal{F}_\epsilon \cap \Omega_\beta,$$

where $\mathbf{e} = (1, 1, \dots, 1)^T$.

Proof. It follows from Lemma 2.2 that the set $\mathcal{D} = \mathcal{F}_\epsilon \cap \Omega_\beta$ has positive lower and upper bounds. It is also clear that function $f(y)$ is twice continuously differentiable on \mathcal{D} . Denote by L the bound of $\|f''(y)\|$ on \mathcal{D} . By the use of the mean-value theorem, we obtain for any $y \in \mathcal{D}$ and $\alpha > 0$,

$$\begin{aligned} \mathcal{A}[(y + \alpha d)^{[\frac{1}{m-1}]}]^{m-1} &= f(y + \alpha d) + b \\ &= f(y) + \alpha f'(y)d + \alpha \int_0^1 [f'(y + \alpha\tau d) - f'(y)] d \cdot d\tau + b \\ &\geq (1 - \alpha)f(y) + b - \frac{1}{2}L\alpha^2\|d\|^2 \mathbf{e} \\ &= (1 - \alpha)\mathcal{A}\left[(y)^{[\frac{1}{m-1}]}]\right]^{m-1} + \alpha b - \frac{1}{2}L\alpha^2\|d\|^2 \mathbf{e} \\ &\geq (1 - \alpha)\epsilon b + \alpha b - \frac{1}{2}L\alpha^2\|d\|^2 \mathbf{e} \\ &= \epsilon b + \alpha \left((1 - \epsilon)b - \frac{1}{2}L\alpha\|d\|^2 \mathbf{e} \right). \end{aligned} \tag{2.2}$$

The proof is complete. □

Denote

$$\bar{\alpha} = \min \left\{ \frac{2(1-\epsilon)b_i}{L\|d\|^2} : i \in [n] \right\}.$$

It follows from the last lemma that if $y \in \mathcal{D}$, then it holds that

$$\mathcal{A} \left[(y + \alpha d)^{\lfloor \frac{1}{m-1} \rfloor} \right]^{m-1} \geq \epsilon b, \quad \forall y \in (0, \bar{\alpha})$$

The steps of the Newton method are stated as follows.

Algorithm 2.4. (Newton's Method)

Initial. Given a small constant $\epsilon \in (0, 1)$ and constants $\sigma \in (0, \frac{1}{2}), \eta, \rho \in (0, 1)$. Select an initial point $x_0 \in \mathcal{F}_\epsilon$. Let $y_0 = x_0^{\lfloor m-1 \rfloor}$ and $k = 0$.

Step 1. Stop if $\|f(y_k)\| \leq \eta$.

Step 2. Solve the system of linear equations

$$f'(y_k)d_k + f(y_k) = 0 \tag{2.3}$$

to get d_k .

Step 3. Determine a steplength $\alpha_k = \max\{\rho^i : i = 0, 1, \dots\}$ such that $y_k + \alpha_k d_k \in \mathcal{F}_\epsilon$ and that the inequality

$$\|f(y_k + \alpha_k d_k)\|^2 \leq (1 - 2\sigma\alpha_k)\|f(y_k)\|^2 \tag{2.4}$$

is satisfied.

Step 4. Let $y_{k+1} = y_k + \alpha_k d_k$ and $x_{k+1} = y_{k+1}^{\lfloor \frac{1}{m-1} \rfloor}$. Go to Step 1.

Remark

- It is easy to see that the last method is very similar to the standard damped Newton method except the line search step where we need to ensure $x_{k+1} \in \mathcal{F}_\epsilon$. If $y_k + d_k \in \mathcal{F}_\epsilon$, then the last method is equivalent to the standard Newton method for solving nonlinear equation $f(y) = 0$.
- The steps of the last method ensure that the generated sequence of iterates $\{x_k\} \in \mathcal{F}_\epsilon$. As a result, $f'(y_k)$ is a strong M-matrix and hence the method is well defined. Moreover, the residual sequence $\{\|f(y_k)\|\}$ is decreasing. It then follows from Lemma 2.2 that there are positive constants $c \leq C$ such that

$$ce \leq y_k \leq Ce. \tag{2.5}$$

- It follows from Lemma 2.3 that if $y_k \in \mathcal{F}_\epsilon$, then

$$y_k + \alpha_k d_k \in \mathcal{F}_\epsilon, \forall \alpha_k \in (0, \bar{\alpha}_k)$$

where

$$\bar{\alpha}_k = \min \left\{ \frac{2(1-\epsilon)b_i}{L \|d_k\|^2} : i \in [n] \right\} \cap \left\{ -\frac{(y_k)_i}{(d_k)_i} : (d_k)_i < 0 \right\} \quad (2.6)$$

and L is the bound of $f''(y)$ on the set compact set $\mathcal{F}_\epsilon \cap \Omega_{\|f(y_0)\|}$.

Let x^* be the unique positive solution to the M-Teq and $y^* = (x^*)^{[m-1]}$. It is easy to see that for any $\epsilon \in (0, 1)$, $x^* \in \mathcal{F}_\epsilon$. Consequently, the matrix $f'(y^*)$ is a nonsingular M-matrix. As a result, the full step Newton method is locally quadratically convergent.

In what follows, we are going to show that Algorithm 2.4 is globally convergent and that after finitely many iterations, the method reduces to the full step Newton method. Consequently, it is quadratically convergent. We first show the following lemma.

Lemma 2.5. *Suppose that \mathcal{A} is a strong M-tensor and $b > 0$. Then the sequence $\{y_k\}$ and $\{d_k\}$ generated by Algorithm 2.4 are bounded. In addition, there is a positive constant $\bar{\alpha}$ such that*

$$y_k + \alpha_k d_k \in \mathcal{F}_\epsilon, \quad \forall \alpha_k \in (0, \bar{\alpha}). \quad (2.7)$$

Proof. By the steps of the algorithm, it is easy to see that the sequence $\{y_k\}$ is contained in the compact set $\mathcal{D} = \mathcal{F}_\epsilon \cap \Omega_{\|f(y_0)\|}$ and hence bounded. Since $f'(y_k)$ is a nonsingular M-matrix and \mathcal{D} is compact, the sequence $\{d_k\}_K$ is bounded too. Notice that $b > 0$ and $\{y_k\}$ has a positive lower bound, the scalar $\bar{\alpha}_k$ defined by (2.6) has a positive lower bound. This together with the boundedness of $\{\|d_k\|\}$ implies that $\bar{\alpha}_k$ has a positive lower bound. Consequently, (2.7) is satisfied with some positive $\bar{\alpha}$. \square

The following theorem establishes the global convergence of the proposed method.

Theorem 2.6. *Suppose that \mathcal{A} is a strong M-tensor and $b > 0$. Then the sequence of iterates $\{x_k\}$ generated by Algorithm 2.4 converges to the unique positive solution to the M-Teq (1.2).*

Proof. It suffices to show that there is an accumulation point \bar{y} of $\{y_k\}$ satisfying $f(\bar{y}) = 0$. Let the subsequence $\{y_k\}_K$ converge to \bar{y} . Without loss of generality, we suppose that the subsequence $\{d_k\}_K$ converges to some \bar{d} .

Denote $\tilde{\alpha} = \liminf_{k \rightarrow \infty, k \in K} \alpha_k$. If $\tilde{\alpha} > 0$, the inequality (2.4) implies $f(\bar{y}) = 0$. Consider the case $\tilde{\alpha} = 0$. By the line search rule, when $k \in K$ is sufficiently large, $\alpha'_k = \rho^{-1} \alpha_k$ will not satisfy (2.4), i.e.,

$$\|f(y_k + \alpha'_k d_k)\|^2 - \|f(y_k)\|^2 > -2\sigma \alpha'_k \|f(y_k)\|^2$$

Dividing both sides of the last inequality by α'_k and then taking limits as $k \rightarrow \infty$ with $k \in K$, we get

$$2f(\bar{y})^T f'(\bar{y}) \bar{d} \geq -2\sigma \|f(\bar{y})\|^2 \quad (2.8)$$

On the other hand, by taking limits in (2.3), we can obtain $f'(\bar{y})\bar{d} + f(\bar{y}) = 0$. It together with (2.8) and the fact $\sigma \in (0, 1)$ yields $f(\bar{y}) = 0$. The proof is complete. \square

The last theorem established the global convergence of the proposed Newton method. Moreover, we see from (2.2) that $x_k + d_k \in \mathcal{F}_\epsilon$ for all k sufficiently large because $\{d_k\} \rightarrow 0$. Consequently, the method locally reduces to a standard damped Newton method. Following a standard discussion as the proof of the quadratic convergence of a damped Newton method, it is not difficult to prove the quadratic convergence of the method. We give the result but omit the proof.

Theorem 2.7. *Let the conditions in Theorem 2.6 hold. Then the convergence rate of the sequence $\{y_k\}$ generate by Algorithm 2.4 is quadratic.*

3 An Extension

In this section, we extend the Newton method proposed in the last section to the M-Teq (1.2) with $b \geq 0$. In the case b has zero elements, the M-Teq may have multiple nonnegative or positive solutions. Our purpose is to find one nonnegative or positive solution of the equation. By Theorem 2.6 in [14], a nonnegative solution of (1.2) has zero elements if and only if \mathcal{A} is reducible with respect to some $I \subseteq I_0$, where

$$I_0 = \{i \in [n] \mid b_i = 0\}.$$

Since justifying the reducibility is an easy task, without loss of generality, we suppose that the nonnegative solutions of the M-Teq are positive.

Assumption 3.1. *Suppose $b \geq 0$ and that tensor \mathcal{A} is a strong M -tensor and irreducible with respect to I_0 . Suppose further that for each $i \in I_0$, there is an element $a_{ii_2 \dots i_m} \neq 0$ with, $i_2, \dots, i_m \in I_+$.*

Under the conditions of Assumption 3.1, we have

$$\mathcal{A}^c(y^{[\frac{1}{m-1} \cdot]^{m-1}})_{I_0} < 0, \quad \forall y \in \mathbb{R}_{++}^n.$$

For the sake of convenience, we introduce some notations. Denote $I_+ = \{i : b_i > 0\}$ and $I_0 = \{i : b_i = 0\}$. For given constants $1 > \epsilon > \epsilon' > 0$, we define

$$\overline{\mathcal{F}}_{\epsilon, \epsilon'} = \overline{\mathcal{F}}_\epsilon^1 \cap \overline{\mathcal{F}}_{\epsilon'}^2$$

with

$$\overline{\mathcal{F}}_\epsilon^1 = \{y \in \mathbb{R}_{++}^n : (\mathcal{A}(y^{[\frac{1}{m-1} \cdot]^{m-1}}))_{I_+} \geq \epsilon b_{I_+}\}$$

and

$$\overline{\mathcal{F}}_{\epsilon'}^2 = \{y \in \mathbb{R}_{++}^n : (\mathcal{A}(y^{[\frac{1}{m-1} \cdot]^{m-1}}))_{I_0} \geq \epsilon' f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} b_{I_+}\}.$$

It is easy to see that every solution $\bar{x} \in \overline{\mathcal{F}}_\epsilon$.

For $y \in \mathbb{R}_{++}^n$, we split $f'(y)$ into

$$f'(y) = \begin{pmatrix} f'_{I_+ I_+}(y) & f'_{I_+ I_0}(y) \\ f'_{I_0 I_+}(y) & f'_{I_0 I_0}(y) \end{pmatrix}.$$

It is easy to see that $f'(y)$ is a Z-matrix.

The next theorem shows that for any $y \in \overline{\mathcal{F}}_\epsilon$, $f'(y)$ is a nonsingular M-matrix. As a result, the set $\overline{\mathcal{F}}_\epsilon$ is well defined.

Theorem 3.2. *Let $1 > \epsilon > \epsilon' > 0$. For any $y \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$, $f'(y)$ is a nonsingular M-matrix.*

Proof. By direct computation, we get $f'(y)y = \mathcal{A}(y^{\lfloor \frac{1}{m-1} \rfloor})^{m-1}$. The condition $y \in \overline{\mathcal{F}}_\epsilon^1$ yields

$$0 < \epsilon b_{I_+} \leq \left(\mathcal{A}(y^{\lfloor \frac{1}{m-1} \rfloor})^{m-1} \right)_{I_+} = f'(y)_{I_+ I_+} y_{I_+} + f'(y)_{I_+ I_0} y_{I_0} \leq f'(y)_{I_+ I_+} y_{I_+}.$$

Consequently, $f'(y)_{I_+ I_+}$ is a nonsingular M-matrix. We are going to show that the Schur complement

$$f'(y)_{I_0 I_0} - f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} f'(y)_{I_+ I_0}$$

is also a nonsingular M-matrix.

Observing $f'(y)_{I_0 I_+} \leq 0$, we get from the condition $y \in \overline{\mathcal{F}}_{\epsilon'}^2$

$$\begin{aligned} f'(y)_{I_0 I_0} y_{I_0} &\geq \epsilon' f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} b_{I_+} - f'(y)_{I_0 I_+} y_{I_+} \\ &\geq \epsilon' f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} b_{I_+} - f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} (\epsilon b_{I_+} - f'(y)_{I_+ I_0} y_{I_0}) \\ &= -(\epsilon - \epsilon') f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} b_{I_+} + f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} f'(y)_{I_+ I_0} y_{I_0}, \end{aligned}$$

which implies

$$(f'(y)_{I_0 I_0} - f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} f'(y)_{I_+ I_0}) y_{I_0} \geq -(\epsilon - \epsilon') f'(y)_{I_0 I_+} f'(y)_{I_+ I_+}^{-1} b_{I_+} > 0.$$

The last condition ensures that $f'(y)$ is a nonsingular M-matrix. □

Similar to Lemma 2.2, we have the following lemma.

Lemma 3.3. *If $\mathcal{A} = (a_{i_1 \dots i_m})$ is a Z-tensor and $b \geq 0$, then for any $\epsilon > 0$, the set $\overline{\mathcal{F}}_\epsilon$ is bounded away from zero. That is, there is a constant $\delta > 0$ such that*

$$y \geq \delta \mathbf{e}, \quad \forall y \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}.$$

Proof. First, following the same arguments as the proof of Lemma 2.2 (i), it is easy to show that the elements y_{I_+} has a positive lower bound. We only need to prove that y_{I_0} has a positive lower bound too.

It is easy to see by the definition of $\overline{\mathcal{F}}_\epsilon^1$ that each $y \in \overline{\mathcal{F}}_\epsilon^1$ satisfies

$$\epsilon b_{I_+} \leq \left(\mathcal{A}(y^{\lfloor \frac{1}{m-1} \rfloor})^{(m-1)} \right)_{I_+} = (f'(y)y)_{I_+} \leq f'(y)_{I_+ I_+} y_{I_+},$$

which implies

$$y_{I_+} \geq f'(y)_{I_+I_+}^{-1} b_{I_+} > \epsilon' f'(y)_{I_+I_+}^{-1} b_{I_+}.$$

The condition $y \in \overline{\mathcal{F}}_{\epsilon'}^2$ implies

$$0 \leq f'_{I_0I_+}(y)(y_{I_+} - \epsilon' f'(y)_{I_+I_+}^{-1} b_{I_+}) + f'_{I_0I_0}(y)y_{I_0}$$

By the condition of Assumption 3.1 and the fact that y_{I_+} has positive lower bound, we claim that the vector $f'_{I_0I_+}(y)(y_{I_+} - \epsilon' f'(y)_{I_+I_+}^{-1} b_{I_+})$ is bounded away from zero. Taking into account that $f'_{I_0I_0}(y)$ is a Z-matrix and $y_{I_0} > 0$, it is easy to see that y_{I_0} is bounded away from zero too.

The proof is complete. \square

In what follows, we propose a Newton method for finding a positive solution to the M-Teq (1.2) with $b \geq 0$ as follows

Algorithm 3.4. (Extended Newton Method for (1.2) with $b \geq 0$)

Initial. Given constants $\epsilon, \epsilon', \rho, \eta, \sigma \in (0, 1)$ satisfying $\epsilon' < \epsilon$. Select an initial point $y_0 \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$. Let $k = 0$.

Step 1. Stop if $\|f(y_k)\| \leq \eta$.

Step 2. Solve the system of linear equations

$$f'(y_k)d_k + f(y_k) = 0. \quad (3.1)$$

to get d_k .

Step 3. Determine a steplength $\alpha_k = \max\{\rho^i : i = 0, 1, \dots\}$ such that $y_k + \alpha_k d_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$ and that the inequality

$$\|f(y_k + \alpha_k d_k)\|^2 \leq (1 - 2\sigma\alpha_k)\|f(y_k)\|^2 \quad (3.2)$$

is satisfied.

Step 4. Let $y_{k+1} = y_k + \alpha_k d_k$. Go to Step 1.

In what follows, we show that the algorithm above is well-defined.

Proposition 3.5. Let the conditions in Assumption 3.1 hold. Then Algorithm 3.4 is well defined.

Proof. It suffices to verify that the relation $y_k + \alpha_k d_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'_{k+1}}$ is satisfied for all $\alpha > 0$ sufficiently small. Indeed, we have

$$\begin{aligned} \mathcal{A}\left((y_k + \alpha d_k)^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1} &= f(y_k + \alpha d_k) + b \\ &= f(y_k) + \alpha f'(y_k)d_k + O(\|\alpha d_k\|^2) + b \\ &= (1 - \alpha)f(y_k) + b + O(\|\alpha d_k\|^2) \\ &= (1 - \alpha)\mathcal{A}\left(y_k^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1} + \alpha b + O(\|\alpha d_k\|^2). \end{aligned}$$

Since $y_k \in \overline{\mathcal{F}}_\epsilon^1$ and $b_{I_+} > 0$, we get from the last equality

$$\left(\mathcal{A}\left((y_k + \alpha d_k)^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1}\right)_{I_+} \geq \epsilon b_{I_+} + \alpha[(1 - \epsilon)b_{I_+} + O(\alpha\|d_k\|^2)], \quad (3.3)$$

which implies $y_k + \alpha d_k \in \overline{\mathcal{F}}_\epsilon^1$ for all $\alpha > 0$ sufficiently small.

Similarly, we have by the fact $y_k \in \overline{\mathcal{F}}_{\epsilon'}^2$ and $b_{I_0} = 0$

$$\left(\mathcal{A}\left((y_k + \alpha d_k)^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1}\right)_{I_0} \geq \epsilon' f'(y_k)_{I_0 I_+} f'(y_k)_{I_+ I_+}^{-1} b_{I_+} + \alpha \left(-\epsilon' f'(y_k)_{I_0 I_+} f'(y_k)_{I_+ I_+}^{-1} b_{I_+} + O(\alpha\|d_k\|^2) \right). \quad (3.4)$$

By the condition of Assumption 3.1, it is not difficult to see from the last inequality that we claim that the inequality

$$\left(\mathcal{A}\left((y_k + \alpha d_k)^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1}\right)_{I_0} \geq \epsilon' f'(y_k)_{I_0 I_+} f'(y_k)_{I_+ I_+}^{-1} b_{I_+}$$

is satisfied for all $\alpha > 0$ sufficiently small. \square

It is easy to show that Lemma 2.2 holds true for the case $b \geq 0$. As a result, the sequence generated by Algorithm 3.4 is bounded. Consequently, the inequalities (3.3) and (3.4) ensure that there is a positive constant $\bar{\alpha} > 0$ such that $x_k + \alpha d_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$. $\forall \alpha \in (0, \bar{\alpha}]$.

Similar to the proof of Theorem 2.6, we can prove the global convergence of Algorithm 3.4.

Theorem 3.6. *Let the conditions in Assumption 3.1 hold. Then the sequence of iterates $\{y_k\}$ generated by Algorithm 3.4 is bounded. Moreover, every accumulation point of the iterates $\{y_k\}$ is a positive solution to the M-tensor equation $f(y) = 0$.*

The remainder of this section is devoted to the proof of the quadratic convergence of Algorithm 3.4. It should be pointed out that the unit steplength may not be acceptable due to the existence of zero elements in b . To ensure the quadratic convergence of the method, we need to make a slight modification to Step 3 of the algorithm. Specifically, we use the following Step 3' instead of Step 3 in Algorithm 3.4.

Step 3'. If $\alpha_k = 1$ satisfies $y_k + \alpha_k d_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$ and (3.2), then we let $\alpha_k = 1$. Otherwise, for given constant $c > 0$, we let $\beta_k = 1 - c\|f(y_k)\|$. If $\beta_k \leq 0$, we let $\beta_k = 1$. Determine a steplength $\alpha_k = \max\{\beta_k \rho^i : i = 0, 1, \dots\}$ such that $y_k + \alpha_k d_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$ and that the inequality (3.2) is satisfied.

It is not difficult to see that the global convergence still remains true if Step 3 is replaced by Step 3'. Moreover, since $\{d_k\} \rightarrow 0$, it is easy to prove from (3.3), (3.4) and (3.2) that for all k sufficiently large, the step $\alpha_k = \beta_k = 1 - c\|f(y_k)\|$ will be accepted. In this case, the sequence of iterates $\{y_k\}$ satisfies $y_{k+1} = y_k + \bar{d}_k$, with $\bar{d}_k = \beta_k d_k = (1 - c\|f(y_k)\|)d_k$ satisfying

$$f'(y_k)\bar{d}_k + f(y_k) = f'(y_k)d_k + f(y_k) - c\|f(y_k)\|d_k = -c\|f(y_k)\|d_k.$$

If $y_{k+1} = y_k + \bar{d}_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}$, then when k is sufficiently, y_k can be regarded as the sequence generated by a full step inexact Newton method. Consequently, the quadratic convergence becomes well-known.

Theorem 3.7. *Let the conditions in Assumption 3.1 hold. Suppose that the sequence of iterates $\{y_k\}$*

generated by Algorithm 3.4 converges to a positive solution y^* to the M -tensor equation $f(y) = 0$. Then the convergence rate of $\{y_k\}$ is quadratic.

Proof. We only need to verify

$$y_{k+1} = y_k + \bar{d}_k = y_k + (1 - c\|f(y_k)\|)d_k \in \overline{\mathcal{F}}_{\epsilon, \epsilon'}. \quad (3.5)$$

It is not difficult to show from (3.1) that

$$\|d_k\| = O(\|f(y_k)\|) = O(\|f(y_k)\|) = O(\|x - x^*\|).$$

Similar to the proof of (3.3), we can derive

$$\left(\mathcal{A}\left((y_k + \beta_k d_k)^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1}\right)_{I_+} \geq \epsilon b_{I_+} + \beta_k[(1 - \epsilon)b_{I_+} + O(\beta_k \|d_k\|^2)].$$

Since $\{\beta_k\} \rightarrow 1$ and $\{d_k\} \rightarrow 0$, the last inequality implies $y_k + \beta_k d_k \in \overline{\mathcal{F}}_{\epsilon}^1$.

We also can obtain

$$\left(\mathcal{A}\left((y_k + \beta_k d_k)^{\lfloor \frac{1}{m-1} \rfloor}\right)^{m-1}\right)_{I_0} \geq \epsilon' f'(y_k)_{I_0 I_+} f'(y_k)_{I_+ I_+}^{-1} b_{I_+} + \beta_k \left(-\epsilon' f'(y_k)_{I_0 I_+} f'(y_k)_{I_+ I_+}^{-1} b_{I_+} + O(\beta_k \|d_k\|^2)\right).$$

By the condition of Assumption 3.1, it is clear that $f'(y_k)_{I_0 I_+} f'(y_k)_{I_+ I_+}^{-1} b_{I_+} < 0$. Consequently, the last inequality implies $y_k + \beta_k d_k \in \overline{\mathcal{F}}_{\epsilon}^2$. The proof is complete. \square

4 Numerical Results

In this section, we do numerical experiments to test the effectiveness of the proposed methods. We implemented our methods in Matlab R2019a and ran the codes on a computer with Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz and 16.0 GB RAM. We used a tensor toolbox [1] to proceed some tensor computation.

The test problems are from [7, 14, 15, 27].

Problem 1. We solve tensor equation (1.2) where \mathcal{A} is a symmetric strong M -tensor of order m ($m = 3, 4, 5$) in the form $\mathcal{A} = s\mathcal{I} - \mathcal{B}$, where tensor \mathcal{B} is symmetric tensor whose entries are uniformly distributed in $(0, 1)$, and

$$s = (1 + 0.01) \cdot \max_{i=1,2,\dots,n} (\mathcal{B}\mathbf{e}^{m-1})_i,$$

where $\mathbf{e} = (1, 1, \dots, 1)^T$.

Problem 2. We solve tensor equation (1.2) where \mathcal{A} is a symmetric strong M -tensor of order m ($m = 3, 4, 5$) in the form $\mathcal{A} = s\mathcal{I} - \mathcal{B}$, and tensor \mathcal{B} is a nonnegative tensor with

$$b_{i_1 i_2 \dots i_m} = |\sin(i_1 + i_2 + \dots + i_m)|,$$

and $s = n^{m-1}$.

Problem 3. Consider the ordinary differential equation

$$\frac{d^2 x(t)}{dt^2} = -\frac{GM}{x(t)^2}, \quad t \in (0, 1),$$

with Dirichlet's boundary conditions

$$x(0) = c_0, \quad x(1) = c_1,$$

where $G \approx 6.67 \times 10^{-11} Nm^2/kg^2$ and $M \approx 5.98 \times 10^{24}$ is the gravitational constant and the mass of the earth.

Discretize the above equation, we have

$$\begin{cases} x_1^3 = c_0^3, \\ 2x_i^3 - x_i^2 x_{i-1} - x_i^2 x_{i+1} = \frac{GM}{(n-1)^2}, \quad i = 2, 3, \dots, n-1, \\ x_n^3 = c_1^3. \end{cases}$$

It is a tensor equation, i.e.,

$$\mathcal{A}x^3 = b,$$

where \mathcal{A} is a 4-th order M tensor whose entries are

$$\begin{cases} a_{1111} = a_{nnnn} = 1, \\ a_{iiii} = 2, \quad i = 2, 3, \dots, n-1, \\ a_{i(i-1)ii} = a_{ii(i-1)i} = a_{iii(i-1)} = -1/3, \quad i = 2, 3, \dots, n-1, \\ a_{i(i+1)ii} = a_{ii(i+1)i} = a_{iii(i+1)} = -1/3, \quad i = 2, 3, \dots, n-1, \end{cases}$$

and b is a positive vector with

$$\begin{cases} b_1 = c_0^3, \\ b_i = \frac{GM}{(n-1)^2}, \quad i = 2, 3, \dots, n-1, \\ b_n = c_1^3. \end{cases}$$

Problem 4. We solve tensor equation (1.2) where \mathcal{A} is a non-symmetric strong M-tensor of order m ($m = 3, 4, 5$) in the form $\mathcal{A} = s\mathcal{I} - \mathcal{B}$, and tensor \mathcal{B} is nonnegative tensor whose entries are uniformly distributed in $(0, 1)$. The parameter s is set to

$$s = (1 + 0.01) \cdot \max_{i=1,2,\dots,n} (\mathcal{B}\mathbf{e}^{m-1})_i.$$

Problem 5. We solve tensor equation (1.2) where \mathcal{A} is a lower triangle strong M-tensor of order m ($m = 3, 4, 5$) in the form $\mathcal{A} = s\mathcal{I} - \mathcal{B}$, and tensor \mathcal{B} is a strictly lower triangular nonnegative tensor whose

entries are uniformly distributed in $(0, 1)$. The parameter s is set to

$$s = (1 - 0.5) \cdot \max_{i=1,2,\dots,n} (\mathcal{B}\mathbf{e}^{m-1})_i.$$

For Problem 1, 2 4 and 5, similar to [11, 12], we solved the tensor equation

$$\hat{F}(x) = \hat{\mathcal{A}}x^{m-1} - \hat{b} = 0$$

instead of the tensor equation (1.2), where $\hat{\mathcal{A}} := \mathcal{A}/\omega$ and $\hat{b} := b/\omega$ with ω is the largest value among the absolute values of components of \mathcal{A} and b . The stopping criterion is set to

$$\|\hat{F}(x_k)\| \leq 10^{-10}.$$

And for Problem 3, the stopping criterion is set to

$$\frac{\|\mathcal{A}x^{m-1} - b\|}{\|b\|} \leq 10^{-10}.$$

We also stop the tested algorithms if the number of iteration reaches to 300, which means that the method is failure for the problem.

Remark 4.1. Since \mathcal{A} is a strong M -tensor, there exists a positive vector u such that $\mathcal{A}u^{m-1} > 0$. This vector u can be obtained in a certain iteration of solving $\mathcal{A}x^{m-1} = \mathbf{e}$ by the existing methods proposed in [7, 11, 12]. Then we can get an initial point of Algorithm 2.4 or Algorithm 3.4 by letting $x_0 = tu$ and $y_0 = x_0^{[m-1]}$ with sufficient large constant t . Particularly, if \mathcal{A} is a diagonally dominant M -tensor we can simply let $u = \mathbf{e}$.

Note that strong M -tensor constructed in Problems 1, 2 and 4 are diagonally dominant M -tensor. Whereas Problem 3 and 5 both are non-diagonally dominant M -tensor.

We first test the performance of Algorithm 2.4 (Newton's Method denoted by 'NM'). In order to test the effectiveness of the proposed method, we compare the Newton method with Inexact Newton Method (denoted by 'INM') proposed in [14]. We take the parameter of NM be $\epsilon = 0.1, \sigma = 0.1$ and $\rho = 0.5$. And let parameters of INM be $\sigma = 0.1, \rho = 0.4$. We set the initial point for INM as in [14], i.e., $y_0 = t\mathbf{e} > 0$ such that $f(y_0) \leq b$, where t is a sufficient small positive constant. We use INM to find an initial point of NM for Problem 3 and 5, i.e., in the iterative of INM, if $\mathcal{A}x_k^{m-1} > 0$, let the vector $u = x_k$ in Remark 4.1.

For the stability of numerical results, we test the problems of different sizes. For each pair (m, n) , we randomly generate 50 tensors \mathcal{A} and $b \in (0, 1)$. The results are listed in Tables 1 to 5, thereinto 'Pro' represent the test problem; 'Iter' represents the average number of iterations; 'Time-Int' denotes the average time to find an initial point of NM; 'Time' denotes the computing time (in seconds) including initial time to find an approximate solution. And the ratio signs are denoted bellow.

It can be seen from Table 1 that the proposed NM has an advantage over the INM in [14] both in iteration and CPU time. Paticularly, for Problem 1, 2 and 4, the coefficient tensor are diagonally dominant,

R-Int		RI		RT		RT1	
$\frac{\text{Time-Int}}{\text{Time of NM}}$		$\frac{\text{Iter of NM}}{\text{Iter of INM}}$		$\frac{\text{Time of NM}}{\text{Time of INM}}$		$\frac{\text{Time of NM minus Time-Int}}{\text{Time of INM}}$	
(m, n)	NM		INM		Rate		
	Iter	Time	Iter	Time	RI	RT	
(3,200)	2	0.01163	11	0.03634	18.1%	32.0%	
(3,401)	2	0.08120	12	0.39976	16.9%	20.3%	
(3,650)	2	0.39719	13	1.79962	15.8%	22.1%	
(4, 40)	2	0.00366	8.4	0.00727	23.8%	50.4%	
(4, 71)	2	0.02786	9.2	0.10398	21.7%	26.8%	
(4,100)	2	0.11595	9.6	0.45303	20.9%	25.6%	
(4,130)	2	0.38220	10	1.34639	19.6%	28.4%	
(5, 30)	2	0.02681	7.6	0.08171	26.4%	32.8%	
(5, 48)	2	0.37926	8.3	1.27908	24.0%	29.7%	

Table 1: Comparison on Problem 1.

(m, n)	NM		INM		Rate	
	Iter	Time	Iter	Time	RI	RT
(3,200)	3	0.01214	11	0.03416	28.2%	35.5%
(3,401)	3	0.10887	12	0.40037	25.3%	27.2%
(3,650)	3	0.48992	12	1.94355	24.9%	25.2%
(4, 40)	3	0.00353	9.4	0.00726	31.8%	48.6%
(4, 71)	3	0.03599	9.3	0.11561	32.2%	31.1%
(4,100)	2.7	0.14569	9.1	0.52564	29.4%	27.7%
(4,130)	2	0.35740	9.7	1.59151	20.9%	22.5%
(5, 30)	2.4	0.02937	8.4	0.09164	29.0%	32.1%
(5, 48)	2	0.38996	8	1.61044	24.9%	24.2%

Table 2: Comparison on Problem 2.

(m, n)	NM				INM		Rate		
	Iter	Time	Time-Int	R-Int	Iter	Time	RI	RT	RT1
(4, 40)	1	0.00263	0.00166	63.1%	5	0.00495	20.0%	53.1%	19.6%
(4, 71)	1	0.03952	0.02627	66.5%	5	0.07993	20.0%	49.4%	16.6%
(4,100)	1	0.16575	0.11177	67.4%	5	0.33503	20.0%	49.5%	16.1%
(4,130)	1	0.45219	0.29926	66.2%	5	0.91583	20.0%	49.4%	16.7%

Table 3: Comparison on Problem 3.

and thus NM is easy to get an initial point, while for Problem 3 and 5, although the number of iterations in NM was significantly less than in INM, the reduction in the required CPU time was not significant since it take much CPU time to find an initial point.

We then test the performance of Algorithm 3.4 (the Extended Newton Method denoted by 'ENM') by comparing with the Regularized Newton Method (denoted by 'RNM') proposed in [14]. The parameter

(m, n)	NM		INM		Rate	
	Iter	Time	Iter	Time	RI	RT
(3,200)	2	0.00886	11	0.03518	17.9%	25.2%
(3,401)	2	0.08087	12	0.39426	17.2%	20.5%
(3,650)	2	0.36240	12	1.97655	16.4%	18.3%
(4, 40)	2	0.00272	8.7	0.00662	22.9%	41.1%
(4, 71)	2	0.02710	9.4	0.11227	21.3%	24.1%
(4,100)	2	0.11558	9.4	0.54760	21.4%	21.1%
(4,130)	2	0.35507	9.9	1.57429	20.3%	22.6%
(5, 30)	2	0.02590	7.7	0.08085	25.9%	32.0%
(5, 48)	2	0.38484	8.4	1.68508	23.7%	22.8%

Table 4: Comparison on Problem 4.

(m, n)	NM				INM		Rate		
	Iter	Time	Time-Int	R-Int	Iter	Time	RI	RT	RT1
(3,200)	2.9	0.02955	0.02151	72.8%	12	0.03493	24.7%	84.6%	23.0%
(3,401)	2.9	0.35773	0.25802	72.1%	12	0.44801	24.2%	79.8%	22.3%
(3,650)	2.9	1.83363	1.38006	75.3%	13	2.21081	22.5%	82.9%	20.5%
(4, 40)	2.9	0.00673	0.00474	70.4%	9.1	0.00640	32.2%	105.1%	31.1%
(4, 71)	2.8	0.09155	0.06100	66.6%	9.6	0.14487	29.1%	63.2%	21.1%
(4,100)	2.9	0.48084	0.31653	65.8%	10	0.63916	28.3%	75.2%	25.7%
(4,130)	2.9	1.53593	1.08875	70.9%	11	1.90284	25.9%	80.7%	23.5%
(5, 30)	2.8	0.07459	0.04873	65.3%	8.7	0.10647	32.3%	70.1%	24.3%
(5, 48)	2.8	1.42363	0.92151	64.7%	9.1	1.87335	30.3%	76.0%	26.8%

Table 5: Comparison on Problem 5.

of ENM are set to $\epsilon = 0.1$, $\epsilon' = 0.05$, $\sigma = 0.1$ and $\rho = 0.5$. And the parameters in RNM are the same as in [14], i.e., $\sigma = 0.1$, $\rho = 0.8$, $\gamma = 0.9$ and $\bar{t} = 0.01$.

For Problem 1, 2, 4 and 5, we generated $b \in R_{++}^n$ randomly and then let some but not all elements of b be 0 randomly, but we let $b_1 \neq 0$ in Problem 5. Then it is easy to see that the strong M-tensor \mathcal{A} and b constructed in our tested problem satisfy the conditions in Assumption 3.1. For Problem 5, we also use INM to find an initial point of ENM. The parameter and starting point of INM are the same as before. And the initial point of RNM is the same as INM. The results are summarized in Table 6 to 9.

It can be seen from the data that the proposed ENM is not only effective for solving the M-Teq with $b \geq 0$, but also more efficient than RNM to a certain extent.

References

- [1] B.W. Bader, T.G. Kolda and others, MATLAB Tensor Toolbox Version 2.6, 2015.
- [2] X. Bai, H. He, C. Ling and G. Zhou, An efficient nonnegativity preserving algorithm for multilinear systems with nonsingular M-tensors, arXiv:1811.09917v1, 2018.

(m, n)	ENM		RNM		Rate	
	Iter	Time	Iter	Time	RI	RT
(3,200)	2.4	0.01286	7.4	0.03194	32.3%	40.3%
(3,350)	2.3	0.05659	7.2	0.18505	32.0%	30.6%
(3,500)	2.2	0.16599	7.7	0.60120	28.1%	27.6%
(3,650)	2.2	0.43180	7.8	1.38419	27.8%	31.2%
(4, 40)	2.3	0.00467	7.4	0.00732	31.1%	63.8%
(4, 90)	2.1	0.07785	8	0.27286	26.4%	28.5%
(4,130)	2	0.38707	8.1	1.20770	25.2%	32.1%
(5, 30)	2.1	0.02927	7.9	0.09237	26.2%	31.7%
(5, 48)	2	0.38698	8.1	1.42453	24.6%	27.2%

Table 6: Comparison on Problem 1.

(m, n)	ENM		RNM		Rate	
	Iter	Time	Iter	Time	RI	RT
(3,200)	3.5	0.01525	9.6	0.04179	36.7%	36.5%
(3,350)	3.2	0.07255	10	0.23803	32.4%	30.5%
(3,500)	3.2	0.22577	10	0.75133	32.3%	30.0%
(3,650)	3.3	0.53518	10	1.82259	32.6%	29.4%
(4, 40)	3.4	0.00394	9.6	0.00838	35.3%	47.0%
(4, 90)	3.3	0.10645	10	0.40368	31.7%	26.4%
(4,130)	3.2	0.48901	10	1.70984	30.9%	28.6%
(5, 30)	3.3	0.03658	10	0.12137	32.7%	30.1%
(5, 48)	3	0.52140	11	2.20545	28.1%	23.6%

Table 7: Comparison on Problem 2.

(m, n)	ENM		RNM		Rate	
	Iter	Time	Iter	Time	RI	RT
(3,200)	2.6	0.01099	7.5	0.03303	34.5%	33.3%
(3,350)	2.2	0.05440	7.6	0.19408	29.1%	28.0%
(3,500)	2.2	0.16818	8	0.63763	27.4%	26.4%
(3,650)	2.1	0.40086	8.1	1.64549	25.8%	24.4%
(4, 40)	2.3	0.00477	7.4	0.01114	30.6%	42.8%
(4, 90)	2	0.09399	7.7	0.28420	26.6%	33.1%
(4,130)	2	0.35984	8.1	1.39153	24.9%	25.9%
(5, 30)	2	0.02607	7.8	0.09134	26.0%	28.5%
(5, 48)	2	0.37042	8.4	1.82102	24.1%	20.3%

Table 8: Comparison on Problem 4.

[3] A. Berman, R. Plemmons, Nonnegative Matrices in the Mathematical Sciences. SIAM, Philadelphia 1994.

[4] M. Brazell, N. Li, C. Navasca and C. Tamon, Solving multilinear systems via tensor inversion, SIAM

(m, n)	ENM				RNM		Rate		
	Iter	Time	Time-Int	R-Int	Iter	Time	RI	RT	RT1
(3,200)	4.3	0.02343	0.01179	50.3%	10	0.03949	42.8%	59.3%	29.5%
(3,350)	4.2	0.14555	0.06743	46.3%	10	0.24752	40.5%	58.8%	31.6%
(3,500)	4.3	0.48284	0.21440	44.4%	11	0.80104	39.7%	60.3%	33.5%
(3,650)	4.1	1.10155	0.49533	45.0%	11	1.87385	38.3%	58.8%	32.4%
(4, 40)	4.4	0.00624	0.00328	52.5%	8.3	0.00636	52.5%	98.2%	46.6%
(4, 90)	4.6	0.25343	0.10136	40.0%	9.2	0.39225	49.5%	64.6%	38.8%
(4,130)	4.4	1.13300	0.48959	43.2%	9.5	1.64542	46.6%	68.9%	39.1%
(5, 30)	4.4	0.07540	0.03641	48.3%	8	0.10592	54.4%	71.2%	36.8%
(5, 48)	4.5	1.40943	0.60182	42.7%	8.5	1.86548	53.2%	75.6%	43.3%

Table 9: Comparison on Problem 5.

Journal on Matrix Analysis & Applications, 34 (2013) 542-570.

- [5] K.C. Chang, K. Pearson and T. Zhang, Primitivity, the convergence of the NQZ method, and the largest eigenvalue for nonnegative tensors. SIAM. J. Matrix Anal. Appl. 32 (2011) 806-819.
- [6] W. Ding, L. Qi and Y. Wei, M-tensors and nonsingular M-tensors, Linear Algebra & Its Applications, 10 (2013) 3264-3278.
- [7] W. Ding and Y. Wei, Solving multi-linear systems with M-tensors, Journal of Scientific Computing, 68 (2016) 1-27.
- [8] F. Facchinei and C. Kanzow, Beyond mootonicity in regularization methods for nonlinear complementarity problems, SIAM Journal on Control and Optimimization, 37 (1999) 1150-1161.
- [9] H.Y. Fan, L. Zhang, E.K. Chu and Y. Wei, Numerical solution to a linear equation with tensor product structure, Numerical Linear Algebra with Applications, 24 (2017) 1-22.
- [10] M.S. Gowda, Z. Luo, L. Qi and N. Xiu, Z-tensor and complementarity problems, arXiv:1510.07933, 2015.
- [11] L. Han, A homotopy method for solving multilinear systems with M-tensors, Applied Mathematics Letters, 69 (2017) 49-54.
- [12] H. He, C. Ling, L. Qi and G. Zhou, A globally and quadratically convergent algorithm for solving multilinear systems with M-tensors, Journal of Scientific Computing, 76 (2018) 1718-1741.
- [13] D. Kressner and C. Tobler, Krylov subspace methods for linear systems with tensor product structure, SIAM Journal on Matrix Analysis & Applications, 31 (2010) 1688-1714.
- [14] D.H. Li, H.B. Guan, J.X. Feng, Inexact Newton method for M-Tensor equations. manuscript. arXiv:2007.13324, 2020.

- [15] D.H. Li, S. Xie and H.R. Xu, Splitting methods for tensor equations, *Numerical Linear Algebra with Applications*, <https://doi.org/10.1002/nla.2102>, 2017.
- [16] X. Li and M.K. Ng, Solving sparse non-negative tensor equations: algorithms and applications, *Frontiers of Mathematics in China*, 10 (2015) 649-680.
- [17] Z. Li, Y.-H. Dai and H. Gao, Alternating projection method for a class of tensor equations, *Journal of Computational & Applied Mathematics*, 346 (2019) 490-504.
- [18] D.H. Li, H.B. Guan and X.Z. Wang, Finding a nonnegative solution to an M-tensor equation, *Pacific Journal of Optimization*, 16 (2020) 419-440.
- [19] L.H. Lim, Singular values and eigenvalues of tensors, a variational approach. In: *Proceedings of the 1st IEEE International Workshop on Computational Advances of Multi-tensor Adaptive Processing*, 1 (2005) 129-132.
- [20] D. Liu, W. Li and S.W. Vong, The tensor splitting with application to solve multi-linear systems, *Journal of Computational & Applied Mathematics*, 330 (2018) 75-94.
- [21] C.Q. Lv and C.F. Ma, A Levenberg-Marquardt method for solving semi-symmetric tensor equations, *Journal of Computational & Applied Mathematics*, 332 (2018) 13-25.
- [22] L. Qi. Eigenvalues of a real supersymmetric tensor. *Journal of Symbolic Computation*, 40 (2005) 1302-1324.
- [23] L. Qi, H. Chen and Y. Chen, *Tensor Eigenvalues and Their Applications*, Springer, 2018.
- [24] L. Qi and Z. Luo, *Tensor Analysis, Spectral Theory and Special Tensors*, SIAM Philadelphia, 2017.
- [25] X. Wang, M. Che and Y. Wei, Neural networks based approach solving multi-linear systems with M-tensors, *Neurocomputing*, 351 (2019) 33-42.
- [26] Z.J. Xie, X.Q. Jin and Y.M. Wei, A fast algorithm for solving circulant tensor systems, *Linear & Multilinear Algebra*, 65 (2017) 1894-1904.
- [27] Z.J. Xie, X.Q. Jin and Y.M. Wei, Tensor methods for solving symmetric M-tensor systems, *Journal of Scientific Computing*, 74 (2018) 412-425.
- [28] Y. Xu, W. Gu and Z.H. Huang, Properties of the nonnegative solution set of multi-linear equations, *Pacific Journal of Optimization*, 15 (2019) 441-456.
- [29] W. Yan, C. Ling, L. Ling and H. He, Generalized tensor equations with leading structured tensors, *Applied Mathematics and Computation*, 361 (2019) 311-324.
- [30] L. Zhang, L. Qi and G. Zhou, M-tensor and some applications. *SIAM Journal on Matrix Analysis & Applications*, 35 (2014) 437-452.