Pacific Journal of Optimization
Vol. 18, No. 1, pp. 91–117, 2022

MODIFIED SPLITTING METHODS FOR SOLVING NON-HOMOGENOUS MULTI-LINEAR EQUATIONS WITH $\mathcal{M}$-TENSORS*

Ming-Yu Deng, Weiyang Ding and Xue-Ping Guo†

Abstract: In this paper, we concern with solving non-homogenous multi-linear equations with $\mathcal{M}$-tensors. We prove the existence and uniqueness of the positive solution to a non-homogenous $\mathcal{M}$-tensor equation with a positive right-hand side vector. In addition, we expand some classical splitting methods like the Jacobi-like, Gauss-Seidel-like, simplified Gauss-Seidel-like, and SOR-like methods to solve the tensor equations, further providing their convergence analyses. Moreover, numerical results show that generally, the SOR-like method performs the best in iteration steps. The Jacobi-like method is the worst but requires less CPU time, and the effects of the other methods are between the above two with narrow distinctions in most non-sparse cases. Furthermore, the SOR-like method for the sparse cases needs the least CPU time, while the Jacobi-like method needs the most.

Key words: *non-homogenous multi-linear systems, $\mathcal{M}$-tensors, classical splitting methods, existence and uniqueness, convergence analysis, sparse systems*

Mathematics Subject Classification: *15A48, 15A69, 65F10, 65H10, 65N22*

---

1 Introduction

Tensor research has attracted a wide range of interests due to some wide applications, such as medical engineering ([36]), the analysis of documents ([8, 30]) and high-order web link ([23, 24]), n-people noncooperative games ([21]), partial differential equations (PDEs, [12, 22]) and so on.

Solving multi-linear equations is an important problem in engineering and scientific computing ([29]). A homogenous multi-linear equation is often represented in a tensor form $\mathcal{A}\mathbf{x}^{m-1} - \mathbf{b} = \mathbf{0}$, where $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$, $\mathcal{A}$ is an mth-order n-dimensional real tensor ([9, 16, 25, 26, 27, 33, 34, 35]) that takes the form

$$\mathcal{A} = (\mathcal{A}_{i_1 i_2 i_3 \cdots i_m}),\ \mathcal{A}_{i_1 i_2 i_3 \cdots i_m} \in \mathbb{R},\ i_j = 1, \cdots, n \text{ for } j = 1, \cdots, m,$$

*W. Ding's research is supported by the National Natural Science Foundation of China under Grant 11801479, Shanghai Municipal Science and Technology Major Project (No. 2018SHZDZX01), ZJ Lab, and Shanghai Center for Brain Science and Brain-Inspired Technology, the 111 Project (No. B18015). X.-P. Guo's research is supported in part by National Natural Science Foundation of China (No. 12071149), Science and Technology Commission of Shanghai Municipality (No. 18dz2271000) and Shanghai Science and Technology Commission (No. 20511100200). M.-Y. Deng's research is supported by the Postgraduate Innovative Research Fund (No. 202104) of University of International Business and Economics.

†Corresponding author

denoted as $\mathcal{A} \in \mathbb{R}^{[m,n]}$, and the notation $\mathcal{A}\mathbf{x}^{m-1}$ is defined by

$$\left(\mathcal{A}\mathbf{x}^{m-1}\right)_i = \sum_{i_2,\cdots,i_m=1}^{n} \mathcal{A}_{ii_2i_3\cdots i_m} x_{i_2} x_{i_3} \cdots x_{i_m} \text{ for } i = 1, \cdots, n.$$

In 2015, Li and Ng proposed iterative methods (see [29]) for solving a set of sparse non-negative tensor equations and gave the convergence analysis under suitable conditions. Moreover, Ding and Wei [12] investigated the solution of $\mathcal{A}\mathbf{x}^{m-1} - \mathbf{b} = \mathbf{0}$ when the coefficient tensor $\mathcal{A}$ is an $\mathcal{M}$-tensor in 2016. In particular, the conditions implying a unique positive solution are "$\mathcal{A}$ is a nonsingular (or strong) $\mathcal{M}$-tensor" and "$\mathbf{b}$ is a positive vector" (see Theorem 3.2 in [12]). In the same year, Li et al. [28] used splitting methods for solving $\mathcal{A}\mathbf{x}^{m-1} - \mathbf{b} = \mathbf{0}$, such as the Jacobi, Gauss-Seidel, SOR, and Newton-Gauss-Seidel iteration methods, which are different from those in [12]. Furthermore, Han introduced a homotopy method in [17] for solving $\mathcal{M}$-equations and proved its convergence in 2017. In 2018, Lv and Ma [31] proposed a Levenberg-Marquardt method for solving semi-symmetric tensor equations and H-eigenvalue of a semi-symmetric tensor and gave the global convergence theorem. In the same year, Xie et al. proposed a new tensor method based on the rank-1 approximation ([5]) of the coefficient tensor for solving some $\mathcal{M}$-systems in [40].

With respect to (nonsingular) $\mathcal{M}$-tensors ([11]), we call a tensor $\mathcal{A} \in \mathbb{R}^{[m,n]}$ an $\mathcal{M}$-tensor, if there exist $s \in \mathbb{R}$ and an mth-order n-dimensional tensor $\mathcal{B} \geq 0$ such that $\mathcal{A} = s\mathcal{I} - \mathcal{B}$, where $s \geq \rho(\mathcal{B})$ with $\rho(\mathcal{B})$ being the spectral radius of $\mathcal{B}$,

$$\rho(\mathcal{B}) = \max\left\{|\lambda| : \mathcal{B}\xi^{m-1} = \lambda \cdot \xi^{[m-1]}\right\}, \xi^{[m-1]} = [\xi_1^{m-1}, \cdots, \xi_n^{m-1}]^\top \neq \mathbf{0}.$$

$\mathcal{B} \geq 0$ means that every entry of $\mathcal{B}$ is nonnegative. When $s > \rho(\mathcal{B})$, $\mathcal{A}$ is called a nonsingular $\mathcal{M}$-tensor. Some equivalent definitions have been proposed in [11, 42] and we just mention a few.

Proposition 1.1. *Suppose that $\mathcal{A}$ is a $\mathcal{Z}$-tensor, i.e., all its off-diagonal entries are non-positive. Then the following conditions are equivalent.*

① *$\mathcal{A}$ is a nonsingular $\mathcal{M}$-tensor.*

② *There exists $\mathbf{x} > \mathbf{0}$ with $\mathcal{A}\mathbf{x}^{m-1} > \mathbf{0}$.*

③ *There exists $\mathbf{x} \geq \mathbf{0}$ with $\mathcal{A}\mathbf{x}^{m-1} > \mathbf{0}$.*

Here $\mathbf{x} > \mathbf{0}$ and $\mathbf{x} \geq \mathbf{0}$ mean all its entries are positive and nonnegative, respectively.

In this paper, we consider the following system of non-homogenous multi-linear equations with $\mathcal{M}$-tensors

$$\mathcal{A}_m\mathbf{x}^{m-1} + \mathcal{A}_{m-1}\mathbf{x}^{m-2} + \cdots + \mathcal{A}_3\mathbf{x}^2 + \mathcal{A}_2\mathbf{x} = \mathbf{b}, \tag{1.1}$$

where $\{\mathcal{A}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$ is a given series of $\mathcal{M}$-tensors, $\mathcal{A}_m$ is a nonsingular $\mathcal{M}$-tensor, $\mathbf{b}$ is a given non-zero vector, and $\mathbf{x}$ contains the unknowns (see [10, 12, 13, 17, 20]).

One application from (1.1) is the one-dimension *nonlinear Poisson equation* ([15]) with Dirichlet ' s boundary condition

$$\begin{cases} -\Delta u = -u_{xx} = f(u) \text{ in } (0,1), \\ u(0) = c_0, u(1) = c_1, \end{cases} \tag{1.2}$$

where

$$f(u) = \frac{1}{1 + u + u^2 + \cdots + u^{m-2}}, \quad m = 2, 3, 4, \cdots.$$

Set the stepsize $h = \frac{1}{n-1}$ and let $x_i = u\left((i-1)\cdot h\right)$. The nonlinear Poisson equation (1.2) is discretized into

$$\begin{cases} x_1 = u(0) = c_0, \\ (2x_i - x_{i-1} - x_{i+1}) \cdot (1 + x_i + x_i^2 + \cdots + x_i^{m-2}) = \frac{1}{(n-1)^2},\ i = 2, 3, \cdots, n-1, \\ x_n = u(1) = c_1. \end{cases} \tag{1.3}$$

By noticing that $(2x_i - x_{i-1} - x_{i+1}) \cdot x_i^{k-2} = \mathcal{A}_k \mathbf{x}^{k-1}$ and $\mathcal{A}_k$ is an $\mathcal{M}$-tensor of kth-order with

$$\begin{cases} (\mathcal{A}_k)_{\underbrace{1,1,1,\cdots,1}_{k \text{ copies}}} = (\mathcal{A}_k)_{n,n,n,\cdots,n} = 1, \\ (\mathcal{A}_k)_{i,i,i,\cdots,i} = 2,\ i = 2, 3, \cdots, n-1, \\ (\mathcal{A}_k)_{i,i-1,i,\cdots,i} = (\mathcal{A}_k)_{i,i,i-1,\cdots,i} = \cdots = (\mathcal{A}_k)_{i,i,i,\cdots,i-1} = -\frac{1}{k-1},\ i = 2, 3, \cdots, n-1, \\ (\mathcal{A}_k)_{i,i+1,i,\cdots,i} = (\mathcal{A}_k)_{i,i,i+1,\cdots,i} = \cdots = (\mathcal{A}_k)_{i,i,i,\cdots,i+1} = -\frac{1}{k-1},\ i = 2, 3, \cdots, n-1 \end{cases} \tag{1.4}$$

for $k = 2, 3, \cdots, m$, hence Eq. (1.3) can be equivalently rewritten as a non-homogenous $\mathcal{M}$-equation with the right-hand side vector $\mathbf{b}$, whose elements are

$$\begin{cases} b_1 = \frac{1-c_0^{m-1}}{1-c_0} \text{ for } c_0 \neq 1 \text{ or } b_1 = m-1 \text{ for } c_0 = 1, \\ b_i = \frac{1}{(n-1)^2},\ i = 2, 3, \cdots, n-1, \\ b_n = \frac{1-c_1^{m-1}}{1-c_1} \text{ for } c_1 \neq 1 \text{ or } b_1 = m-1 \text{ for } c_1 = 1. \end{cases}$$

Generally, the storage cost of the tensor $\mathcal{A}$ is $O(n^k)$ and the computation complexity of $\mathcal{A}\mathbf{x}^{k-1}$ is $(k-1)n^k$ (i.e., $O(n^k)$) for any tensor $\mathcal{A} \in \mathbb{R}^{[k,n]}$. However, by some straightforward calculations, we can find the number of nonzero entries of $\mathcal{A}_k$ is only $2(k-1)(n-2)+n$ (i.e., $O(n)$), which indicates (1.3) is a sparse system in nature. At the same time, the computation complexity of $\mathcal{A}_k\mathbf{x}^{k-1}$ will decrease from $O(n^k)$ to $O(n)$, which improves greatly the computational efficiency. In addition, the sets of n-dimension nonnegative vectors and n-dimension positive vectors are denoted by $\mathbb{R}^n_+$ and $\mathbb{R}^n_{++}$, respectively. Moreover, the operator $\left[\{\mathcal{A}_k\}_{k=2}^m\right]^{-1}\mathbf{b}$ represents roots of the multi-polynomial equation (1.1), i.e.,

$$\left[\{\mathcal{A}_k\}_{k=2}^m\right]^{-1}\mathbf{b} = \left\{\mathbf{x} \in \mathbb{R}^n : \mathcal{A}_m\mathbf{x}^{m-1} + \mathcal{A}_{m-1}\mathbf{x}^{m-2} + \cdots + \mathcal{A}_3\mathbf{x}^2 + \mathcal{A}_2\mathbf{x} = \mathbf{b}\right\}, \tag{1.5}$$

where $\mathcal{A}_k \in \mathbb{R}^{[k,n]}$ and $\mathbf{b} \in \mathbb{R}^n$. Similarly,

$$\left[\{\mathcal{A}_k\}_{k=2}^m\right]_{++}^{-1}\mathbf{b} = \left\{\mathbf{x} \in \mathbb{R}^n_{++} : \mathcal{A}_m\mathbf{x}^{m-1} + \mathcal{A}_{m-1}\mathbf{x}^{m-2} + \cdots + \mathcal{A}_3\mathbf{x}^2 + \mathcal{A}_2\mathbf{x} = \mathbf{b}\right\}. \tag{1.6}$$

The rest of this paper is organized as follows. First we give the proof of the existence and uniqueness of a positive solution to non-homogenous multi-linear equations (1.1) in Section 2. Next, we present the Jacobi-like, (backward, simplified) Gauss-Seidel-like and SOR-like methods established to solve (1.1) in Section 3. In addition, we provide convergence analyses for the proposed algorithms in Section 4 and compare the effects of these methods applied to solve equations in Section 5. Finally, we draw some conclusions and raise several future directions in Section 6.

2 Existence and Uniqueness of Positive Solutions

In order to analyze the existence and uniqueness of the positive solution to a non-homogenous multi-linear equation with a positive right-hand side vector, we introduce two lemmas about polynomial equations as follows.

Lemma 2.1. *Given $b > 0$ and suppose that*

$$a_k \geq 0 \text{ for any } k = 2, 3, \ldots, m \text{ and } a_j > 0 \text{ for some } j \in \{2, 3, \ldots, m\}.$$

The following polynomial equation

$$a_m x^{m-1} + a_{m-1} x^{m-2} + \cdots + a_2 x = b \tag{2.1}$$

has a unique positive root. Furthermore, for any $b_2 \geq b_1 > 0$, we have $x_2 \geq x_1 > 0$, where x_1 and x_2 satisfy

$$a_m x_1^{m-1} + a_{m-1} x_1^{m-2} + \cdots + a_2 x_1 = b_1, \tag{2.2}$$

$$a_m x_2^{m-1} + a_{m-1} x_2^{m-2} + \cdots + a_2 x_2 = b_2. \tag{2.3}$$

Proof. Denote $f(x) = a_m x^{m-1} + a_{m-1} x^{m-2} + \cdots + a_2 x - b$. We can find $f'(x)$ is strictly increasing in $(0, +\infty)$. Notice that

$$f(0) = -b < 0 \text{ and } \lim_{x \to +\infty} f(x) = +\infty.$$

Hence, $f(x)$ has a unique zero point in $(0, +\infty)$. Taking difference between (2.2) and (2.3) yields

$$a_m(x_1^{m-1} - x_2^{m-1}) + a_{m-1}(x_1^{m-2} - x_2^{m-2}) + \cdots + a_2(x_1 - x_2) = b_1 - b_2 \leq 0,$$

which can be rewritten as

$$\begin{aligned} &(x_1 - x_2)\left[a_m \sum_{i=0}^{m-2} x_1^i x_2^{m-2-i} + a_{m-1} \sum_{i=0}^{m-3} x_1^i x_2^{m-3-i} + \cdots + a_2\right] \\ =&a_m(x_1 - x_2) \sum_{i=0}^{m-2} x_1^i x_2^{m-2-i} + a_{m-1}(x_1 - x_2) \sum_{i=0}^{m-3} x_1^i x_2^{m-3-i} + \cdots + a_2(x_1 - x_2) \leq 0 \end{aligned}$$

by using the formula that $a^{k-1} - b^{k-1} = (a - b)(a^{k-2} + a^{k-3}b + \cdots + ab^{k-3} + b^{k-2})$. Then $x_1 \leq x_2$, because

$$a_m \sum_{i=0}^{m-2} x_1^i x_2^{m-2-i} + a_{m-1} \sum_{i=0}^{m-3} x_1^i x_2^{m-3-i} + \cdots + a_2 > 0$$

can be derived from $x_1 > 0$, $x_2 > 0$ and the assumption about $\{a_k \in \mathbb{R}\}_{k=2}^m$. □

In addition, through this lemma, we can portray the following conclusion in the n-dimensional case of the $\mathcal{M}$-equation, whose diagonal elements are identical and the others are zeros.

Lemma 2.2. *Given $\mathbf{b} \in \mathbb{R}_{++}^n$ and suppose that*

$$s_k \geq 0 \text{ for any } k = 2, 3, \ldots, m \text{ and } s_j > 0 \text{ for some } j \in \{2, 3, \ldots, m\}.$$

The following system of polynomial equations

$$s_m \mathbf{x}^{[m-1]} + s_{m-1} \mathbf{x}^{[m-2]} + \cdots + s_3 \mathbf{x}^{[2]} + s_2 \mathbf{x} = \mathbf{b} \tag{2.4}$$

has a unique positive solution. Furthermore, for any $\mathbf{d} \geq \mathbf{c} > \mathbf{0}$, we have $\mathbf{y} \geq \mathbf{x} > \mathbf{0}$, where $\mathbf{x}$ and $\mathbf{y}$ are satisfying

$$s_m \mathbf{x}^{[m-1]} + s_{m-1} \mathbf{x}^{[m-2]} + \cdots + s_3 \mathbf{x}^{[2]} + s_2 \mathbf{x} = \mathbf{c},$$

$$s_m \mathbf{y}^{[m-1]} + s_{m-1} \mathbf{y}^{[m-2]} + \cdots + s_3 \mathbf{y}^{[2]} + s_2 \mathbf{y} = \mathbf{d}.$$

Proof. For any $i \in \{1, 2, \ldots, n\}$, Lemma 2.1 shows us that

$$s_m x_i^{m-1} + s_{m-1} x_i^{m-2} + \cdots + s_3 x_i^2 + s_2 x_i = b_i, \quad b_i > 0 \tag{2.5}$$

has a unique root in $(0, +\infty)$, denoted as x_i^*. This implies Eq. (2.4) has the unique positive root

$$\mathbf{x}_* = [x_1^*, x_2^*, \cdots, x_n^*]^\top.$$

In addition, Theorem 2.1 shows us that, for any $d_i \geq c_i > 0$, we have $y_i \geq x_i > 0$, where x_i and y_i are the roots of equation (2.5) when the right-hand side scalars are c_i and d_i, respectively. Therefore, let $\mathbf{x} = [x_1, \cdots, x_n]^\top$, $\mathbf{y} = [y_1, \cdots, y_n]^\top$ and $\mathbf{c} = [c_1, \cdots, c_n]^\top$, $\mathbf{d} = [d_1, \cdots, d_n]^\top$. Then we get $\mathbf{0} < \mathbf{x} \leq \mathbf{y}$ when $\mathbf{0} < \mathbf{c} \leq \mathbf{d}$. □

Moreover, we can obtain the theorem about the existence and uniqueness of a positive solution to Eq. (1.1) by the following two lemmas and a fixed-point theorem in [1].

Lemma 2.3. *Let $\{\mathcal{B}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$ be a series of nonnegative tensors and $\{s_k \in \mathbb{R}\}_{k=2}^m$ be also nonnegative with at least one $s_k > 0$. Given $\mathbf{b}, \mathbf{x}_0 \in \mathbb{R}^n_{++}$, and define $\mathbf{x}_{l+1}$ as the solution to*

$$s_m \mathbf{x}^{[m-1]} + s_{m-1} \mathbf{x}^{[m-2]} + \cdots + s_3 \mathbf{x}^{[2]} + s_2 \mathbf{x} = \mathcal{B}_m \mathbf{x}_l^{m-1} + \mathcal{B}_{m-1} \mathbf{x}_l^{m-2} + \cdots + \mathcal{B}_2 \mathbf{x}_l + \mathbf{b}, \quad l = 0, 1, \cdots. \tag{2.6}$$

Then each vector of the sequence $\{\mathbf{x}_l\}_{l=0}^\infty$ is positive.

Proof. First, we know that

$$s_m \mathbf{x}^{[m-1]} + s_{m-1} \mathbf{x}^{[m-2]} + \cdots + s_3 \mathbf{x}^{[2]} + s_2 \mathbf{x} = \mathcal{B}_m \mathbf{x}_0^{m-1} + \mathcal{B}_{m-1} \mathbf{x}_0^{m-2} + \cdots + \mathcal{B}_2 \mathbf{x}_0 + \mathbf{b}$$

has a unique positive solution under given conditions according to Lemma 2.2, i.e., $\mathbf{x}_1$ is positive. Second, we assume that $\mathbf{x}_l > \mathbf{0}$ which indicates

$$\mathcal{B}_m \mathbf{x}_l^{m-1} + \mathcal{B}_{m-1} \mathbf{x}_l^{m-2} + \cdots + \mathcal{B}_2 \mathbf{x}_l + \mathbf{b} > \mathbf{0}, \quad l \geq 1.$$

Finally, we can similarly get $\mathbf{x}_{l+1}$ is positive according to Lemma 2.2. □

Lemma 2.4. *Let $\mathcal{A}_m \in \mathbb{R}^{[m,n]}$ be a nonsingular $\mathcal{M}$-tensor and $\{\mathcal{B}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^{m-1}$ be a series of nonnegative tensors, and $\mathbf{b} \in \mathbb{R}^n_{++}$. Then there exists a vector $\mathbf{x} \in \mathbb{R}^n_{++}$ such that*

$$\mathcal{A}_m \mathbf{x}^{m-1} > \mathcal{B}_{m-1} \mathbf{x}^{m-2} + \cdots + \mathcal{B}_2 \mathbf{x} + \mathbf{b}. \tag{2.7}$$

Proof. Since $\mathcal{A}_m$ is a non-singular $\mathcal{M}$-tensor and by ② in Prop. 1.1, there exists a vector $\mathbf{z} \in \mathbb{R}^n_{++}$ such that $\mathcal{A}_m \mathbf{z}^{m-1} > \mathbf{0}$. For any $\alpha > 1$ we have

$$\mathcal{A}_m (\alpha \mathbf{z})^{m-1} = \alpha^{m-1} \cdot \mathcal{A}_m \mathbf{z}^{m-1} > \mathbf{0}$$

and

$$\begin{aligned} \mathbf{0} < \mathcal{B}_{m-1} (\alpha \mathbf{z})^{m-2} + \cdots + \mathcal{B}_2 (\alpha \mathbf{z}) + \mathbf{b} &= \alpha^{m-2} \cdot \mathcal{B}_{m-1} \mathbf{z}^{m-2} + \cdots + \alpha \cdot \mathcal{B}_2 \mathbf{z} + \mathbf{b} \\ &< \alpha^{m-2} \cdot \left[\mathcal{B}_{m-1} \mathbf{z}^{m-2} + \cdots + \mathcal{B}_2 \mathbf{z} + \mathbf{b} \right]. \end{aligned}$$

Therefore, we get

$$\begin{aligned} &\mathcal{A}_m (\alpha \mathbf{z})^{m-1} - \left[\mathcal{B}_{m-1} (\alpha \mathbf{z})^{m-2} + \cdots + \mathcal{B}_2 (\alpha \mathbf{z}) + \mathbf{b} \right] \\ &> \alpha^{m-1} \cdot \mathcal{A}_m \mathbf{z}^{m-1} - \alpha^{m-2} \cdot \left[\mathcal{B}_{m-1} \mathbf{z}^{m-2} + \cdots + \mathcal{B}_2 \mathbf{z} + \mathbf{b} \right] \\ &= \alpha^{m-2} \cdot \left[\alpha \mathcal{A}_m \mathbf{z}^{m-1} - \left(\mathcal{B}_{m-1} \mathbf{z}^{m-2} + \cdots + \mathcal{B}_2 \mathbf{z} + \mathbf{b} \right) \right]. \end{aligned}$$

It follows that there a large enough α like

$$\alpha = \frac{\max\limits_{i=1,\cdots,n} \left\{(\mathcal{B}_{m-1}\mathbf{z}^{m-2} + \cdots + \mathcal{B}_2\mathbf{z} + \mathbf{b})_i\right\}}{\min\limits_{i=1,\cdots,n} \{(\mathcal{A}_m\mathbf{z}^{m-1})_i\}} + 1$$

satisfying $\alpha\mathcal{A}_m\mathbf{z}^{m-1} - \left(\mathcal{B}_{m-1}\mathbf{z}^{m-2} + \cdots + \mathcal{B}_2\mathbf{z} + \mathbf{b}\right) > \mathbf{0}$, which indicates the inequality (2.7) taking $\mathbf{x} = \alpha\mathbf{z}$. ☐

Theorem 2.5 ([1, 12, 39])**.** *Let $\mathbb{P}$ be a regular cone in an ordered Banach space $\mathbb{E}$ and $[\mathbf{u}, \mathbf{v}] \subset \mathbb{E}$ be a bounded order interval. Suppose that $T : [\mathbf{u}, \mathbf{v}] \to \mathbb{E}$ is an increasing map*[1] *which satisfies*

$$\mathbf{u} \le T(\mathbf{u}) \text{ and } \mathbf{v} \ge T(\mathbf{v}). \tag{2.8}$$

Then T has at least one fixed point in $[\mathbf{u}, \mathbf{v}]$. Moreover, there exist a minimal fixed point $\mathbf{x}_*$ and a maximal fixed point $\mathbf{x}^*$ in the sense that every fixed point $\overline{\mathbf{x}}$ satisfies $\mathbf{x}_* \le \overline{\mathbf{x}} \le \mathbf{x}^*$. Finally, the iteration method

$$\mathbf{x}_{l+1} = T(\mathbf{x}_l), \quad l = 0, 1, 2, \cdots$$

converges to $\mathbf{x}_*$ from below if $\mathbf{x}_0 = \mathbf{u}$, i.e.,

$$\mathbf{u} = \mathbf{x}_0 \le \mathbf{x}_1 \le \cdots \le \mathbf{x}_*,$$

and converges to $\mathbf{x}^*$ from above if $\mathbf{x}_0 = \mathbf{v}$, i.e.,

$$\mathbf{v} = \mathbf{x}_0 \ge \mathbf{x}_1 \ge \cdots \ge \mathbf{x}^*.$$

Based on the above lemmas and theorem, we have the following theorem.

Theorem 2.6. *Given $\mathbf{b} \in \mathbb{R}^n_{++}$. The non-homogeneous $\mathcal{M}$-equation* (1.1) *has a unique positive solution, i.e., there exists only one $\mathbf{x}_* \in \mathbb{R}^n_{++}$ such that*

$$\mathcal{A}_m\mathbf{x}_*^{m-1} + \mathcal{A}_{m-1}\mathbf{x}_*^{m-2} + \cdots + \mathcal{A}_3\mathbf{x}_*^2 + \mathcal{A}_2\mathbf{x}_* = \mathbf{b}.$$

Proof. According to the definition of (nonsingular) $\mathcal{M}$-tensors, there exists a series of nonnegative tensors $\{\mathcal{B}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$ and scalars $\{s_k \in \mathbb{R}\}_{k=2}^m$ satisfying

$$\mathcal{A}_k = s_k\mathcal{I}_k - \mathcal{B}_k, \text{ and } s_k \ge \rho(\mathcal{B}_k), \quad k = 2, \cdots, m.$$

Particularly, $s_m > \rho(\mathcal{B}_m) \ge 0$. Consider the fixed-point iteration

$$\mathbf{x}_{l+1} = \mathcal{P}_{s,\mathcal{B}}(\mathbf{x}_l), \quad l = 0, 1, \cdots,$$

where $\mathbf{x}_0 \in \mathbb{R}^n_{++}$ is given and $\mathcal{P}_{s,\mathcal{B}}(\mathbf{x}_l)$ represents the unique positive solution of the multi-polynomial equation (2.6) (proved in Lemma 2.3). Apparently $\mathcal{P}_{s,\mathcal{B}}(\cdot)$ is a mapping from $\mathbb{R}^n_{++}$ to $\mathbb{R}^n_{++}$ and $\mathbf{0} < \mathcal{P}_{s,\mathcal{B}}(\mathbf{0})$.

Next, we know from Lemma 2.4 there exist $\mathbf{z} \in \mathbb{R}^n_{++}$ and a scalar $\alpha > 0$ such that

$$\mathcal{A}_m(\alpha\mathbf{z})^{m-1} > \mathcal{B}_{m-1}(\alpha\mathbf{z})^{m-2} + \cdots + \mathcal{B}_2(\alpha\mathbf{z}) + \mathbf{b}.$$

[1] T is an increasing continuous map if for any $\mathbf{x}$, $\mathbf{y}$ in its domain and $\mathbf{x} \le \mathbf{y}$ (i.e., $\mathbf{x} - \mathbf{y} \le \mathbf{0}$), we have $T(\mathbf{x}) \le T(\mathbf{y})$ (i.e., $T(\mathbf{x}) - T(\mathbf{y}) \le \mathbf{0}$).

Furthermore,

$$\mathcal{A}_m(\alpha\mathbf{z})^{m-1} + s_{m-1}(\alpha\mathbf{z})^{[m-2]} + \cdots + s_3(\alpha\mathbf{z})^{[2]} + s_2(\alpha\mathbf{z}) > \mathcal{B}_{m-1}(\alpha\mathbf{z})^{m-2} + \cdots + \mathcal{B}_2(\alpha\mathbf{z}) + \mathbf{b}$$

holds, which can be rewritten as

$$s_m(\alpha\mathbf{z})^{[m-1]} + \cdots + s_3(\alpha\mathbf{z})^{[2]} + s_2(\alpha\mathbf{z}) > \mathcal{B}_m(\alpha\mathbf{z})^{m-1} + \mathcal{B}_{m-1}(\alpha\mathbf{z})^{m-2} + \cdots + \mathcal{B}_2(\alpha\mathbf{z}) + \mathbf{b}. \quad (2.9)$$

Meanwhile, the definition of $\mathcal{P}_{s,\mathcal{B}}(\cdot)$ implies that $\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})$ is the solution of following equation

$$s_m\mathbf{x}^{[m-1]} + s_{m-1}\mathbf{x}^{[m-2]} + \cdots + s_3\mathbf{x}^{[2]} + s_2\mathbf{x} = \mathcal{B}_m(\alpha\mathbf{z})^{m-1} + \mathcal{B}_{m-1}(\alpha\mathbf{z})^{m-2} + \cdots + \mathcal{B}_2(\alpha\mathbf{z}) + \mathbf{b},$$

in which $\mathbf{x}$ contains the unknowns. It follows that

$$\begin{aligned} s_m\left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)^{[m-1]} + \cdots + s_3\left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)^{[2]} + s_2\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z}) = \mathcal{B}_m(\alpha\mathbf{z})^{m-1} + \mathcal{B}_{m-1}(\alpha\mathbf{z})^{m-2} \\ + \cdots + \mathcal{B}_2(\alpha\mathbf{z}) + \mathbf{b}. \end{aligned} \quad (2.10)$$

Taking (2.9) and (2.10) together, it yields

$$s_m(\alpha\mathbf{z})^{[m-1]} + \cdots + s_3(\alpha\mathbf{z})^{[2]} + s_2(\alpha\mathbf{z}) > s_m\left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)^{[m-1]} + \cdots + s_3\left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)^{[2]} + s_2\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z}),$$

i.e.,

$$\sum_{k=2}^{m} s_k\left[(\alpha\mathbf{z})^{[k-1]} - \left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)^{[k-1]}\right] > \mathbf{0}.$$

This indicates for any $i = 1, \ldots, n$,

$$\begin{aligned} \sum_{k=2}^{m} s_k \Big[\left((\alpha\mathbf{z})_i - \left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i\right)\Big((\alpha\mathbf{z})_i^{k-2} + (\alpha\mathbf{z})_i^{k-3}\left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i \\ + \cdots + (\alpha\mathbf{z})_i\left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i^{k-3} + \left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i^{k-2}\Big)\Big] > 0 \end{aligned}$$

or equivalent form

$$\left[(\alpha\mathbf{z})_i - \left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i\right]\sum_{k=2}^{m} s_k\left[(\alpha\mathbf{z})_i^{k-2} + \cdots + \left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i^{k-2}\right] > 0$$

by using the formula $a^{k-1} - b^{k-1} = (a-b)(a^{k-2} + a^{k-3}b + \cdots + ab^{k-3} + b^{k-2})$. Obviously, there must be $(\alpha\mathbf{z})_i > \left(\mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z})\right)_i$. Otherwise the above inequality is wrong. Thus,

$$\alpha\mathbf{z} > \mathcal{P}_{s,\mathcal{B}}(\alpha\mathbf{z}).$$

For any $\mathbf{y} \geq \mathbf{x} > \mathbf{0}$, we can verify that

$$\mathcal{B}_m\mathbf{x}^{m-1} + \cdots + \mathcal{B}_2\mathbf{x} + \mathbf{b} \leq \mathcal{B}_m\mathbf{y}^{m-1} + \cdots + \mathcal{B}_2\mathbf{y} + \mathbf{b},$$

which yields $\mathcal{P}_{s,\mathcal{B}}(\mathbf{x}) \leq \mathcal{P}_{s,\mathcal{B}}(\mathbf{y})$ based on Theorem 2.2. Therefore, $\mathcal{P}_{s,\mathcal{B}}(\cdot)$ is an increasing continuous map. Note that $\mathbb{R}^n_+$ is a regular cone. There exists at least one fixed point $\mathbf{x}_*$ of $\mathcal{P}_{s,\mathcal{B}}(\cdot)$ with $\mathbf{0} < \mathbf{x}_* < \alpha\mathbf{z}$ according to Theorem 2.5, which is also a positive vector.

Furthermore, we can prove that the positive fixed point is unique when $\mathbf{b}$ is positive. Assume that there are two positive fixed points $\mathbf{x}_*$ and $\mathbf{y}_*$, i.e.,

$$\mathcal{P}_{s,\mathcal{B}}(\mathbf{x}_*) = \mathbf{x}_* > \mathbf{0} \text{ and } \mathcal{P}_{s,\mathcal{B}}(\mathbf{y}_*) = \mathbf{y}_* > \mathbf{0},$$

and let $\gamma = \min_{i=1,2,\cdots,n} \frac{(\mathbf{y}_*)_i}{(\mathbf{x}_*)_i} > 0$. Then $\mathbf{y}_* \geq \gamma \mathbf{x}_*$ and $(\mathbf{y}_*)_j = \gamma(\mathbf{x}_*)_j$ for some j. If $\gamma < 1$, we obtain

$$\mathcal{A}_m(\gamma \mathbf{x}_*)^{m-1} + \cdots + \mathcal{A}_2(\gamma \mathbf{x}_*) < \mathcal{A}_m \mathbf{x}_*^{m-1} + \cdots + \mathcal{A}_2 \mathbf{x}_* = \mathbf{b}.$$

From the above discussion, we know that

$$\gamma \mathbf{x}_* < \mathcal{P}_{s,\mathcal{B}}(\gamma \mathbf{x}_*).$$

However, since $\mathcal{P}_{s,\mathcal{B}}$ is positive and increasing, we have

$$\left(\mathcal{P}_{s,\mathcal{B}}(\gamma \mathbf{x}_*)\right)_j \leq \left(\mathcal{P}_{s,\mathcal{B}}(\mathbf{y}_*)\right)_j = (\mathbf{y}_*)_j = \gamma(\mathbf{x}_*)_j.$$

This forms a contradiction. If $\gamma \geq 1$, it implies $\mathbf{y}_* \leq \gamma \mathbf{x}_*$. Similarly, we can also show that $\mathbf{x}_* \leq \gamma \mathbf{y}_*$, so $\mathbf{x}_* = \gamma \mathbf{y}_*$. Therefore, the positive fixed point of $\mathcal{P}_{s,\mathcal{B}}(\cdot)$ is unique, i.e., the positive solution to the non-homogeneous $\mathcal{M}$-equation (1.1) is unique. □

However, the inverse of Theorem 2.6 is not correct. For example, after taking $\mathbf{x} = \mathbf{1} \triangleq [1, \cdots, 1]_n^\top$ in the Test One of Example 5.1 and by direct calculation on Matlab, we find

$$\mathbf{b} = \mathcal{A}_3 \mathbf{1}^2 + \mathcal{A}_2 \mathbf{1} \approx [-687.6, \cdots, 1367.2]^\top \notin \mathbb{R}^n_{++}.$$

3 Generalization of Classical Methods

Just like the definitions of the *diagonal*, *(strictly) lower* and *(strictly) upper triangular matrixs* in matrix theory, Ding and Wei define the *diagonal*, *(strictly) lower* and *(strictly) upper triangular tensors* (see Pages 692, 693 in [12]) as follows.

Definition 3.1 ([12])**.** For any tensor $\mathcal{A} \in \mathbb{R}^{[m,n]}$,

- The *diagonal* of $\mathcal{A}$ contains the entries $\mathcal{A}_{i,\cdots,i}$ with $i = 1, 2, \ldots, n$, and other entries are called *off-diagonal* (see [33]). A tensor is called diagonal if all its off-diagonal entries are zeros.
- The *lower triangular part* of $\mathcal{A}$ contains the entries $\mathcal{A}_{i_1 i_2 \ldots i_m}$ with $i_1 = 1, 2, \ldots, n$ and $i_2, \ldots, i_m \leq i_1$, and other entries are said to be in the *off-lower triangular part*. The *strictly lower part* consists of the entries $\mathcal{A}_{i_1 i_2 \cdots i_m}$ with $i_1 = 1, 2, \ldots, n$ and $i_2, \ldots, i_m < i_1$. A tensor is called *lower triangular* if all its entries in the off-lower triangular part are zeros.
- Similarly, the *upper triangular part* of $\mathcal{A}$ contains the entries $\mathcal{A}_{i,\ldots,i}$ with $i_1 = 1, 2, \ldots, n$ and $i_2, \ldots, i_m \geq i_1$, and other entries are said to be in the *off-upper triangular part*. The *strictly upper part* consists of the entries $\mathcal{A}_{i_1 i_2 \cdots i_m}$ with $\mathcal{A}_{i,\ldots,i}$ with $i_1 = 1, 2, \ldots, n$ and $i_2, \ldots, i_m > i_1$. A tensor is called *upper triangular* if all its entries in the off-upper triangular part are zeros.

Then *the non-homogenous diagonal equation* (the simplest non-homogenous multi-linear equations)

$$\mathcal{D}_m \mathbf{x}^{m-1} + \mathcal{D}_{m-1} \mathbf{x}^{m-2} + \cdots + \mathcal{D}_2 \mathbf{x} = \mathbf{b}, \tag{3.1}$$

the non-homogenous lower triangle equation

$$\mathcal{L}_m \mathbf{x}^{m-1} + \mathcal{L}_{m-1} \mathbf{x}^{m-2} + \cdots + \mathcal{L}_2 \mathbf{x} = \mathbf{b}, \tag{3.2}$$

and *the non-homogenous upper triangle equation*

$$\mathcal{U}_m\mathbf{x}^{m-1} + \mathcal{U}_{m-1}\mathbf{x}^{m-2} + \cdots + \mathcal{U}_2\mathbf{x} = \mathbf{b}, \tag{3.3}$$

corresponding to *diagonal* ($\{\mathcal{D}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$), *lower* ($\{\mathcal{L}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$), and *upper triangular tensors* ($\{\mathcal{L}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$), respectively, are defined automatically (similar to [2, 3, 6]). At the same time, we establish a **direct method** named *forward substitution* (*back substitution*) for solving the *the lower triangle equation* (3.2) (*the upper triangle equation* (3.3)) as follows, and another is omitted because it is similar.

Algorithm 1: forward substitution

Input: The set of coefficient tensors $\{\mathcal{L}_k \in \mathbb{R}^{[k,n]}\}_{k=2}^m$ and the right hand vector $\mathbf{b} \in \mathbb{R}^n$ for the lower triangle equation (3.2).

Output: $\mathbf{x}_*$, the root set to the above lower triangle equation.

x_1 = one root of $(\mathcal{L}_m)_{1,\cdots,1}x_1^{m-1} + \cdots + (\mathcal{L}_3)_{1,1,1}x_1^2 + (\mathcal{L}_2)_{1,1}x_1 = b_1$;

for $i = 2 : n$ **do**

  for $j = 1 : m$ **do**

$$p_j = \sum_{k=j}^{m} \sum_{i_2,\cdots,i_k=1}^{i} \left\{ \frac{1}{x_i^{j-1}} (\mathcal{L}_k)_{i,i_2,\cdots,i_k} \prod_{l=2}^{k} x_{i_l} : \text{there only exist } (j-1) \text{ indices in } \{i_2,\cdots,i_k\} \text{ equal to } i \right\}$$

  end

  x_i = one of the roots of the polynomial equation: $p_1 + p_2t + \cdots + p_mt^{m-1} = b_i$;

end

return $\mathbf{x}$;

For different choices of $\mathcal{M}_k$ in

$$\mathcal{A}_k = \mathcal{M}_k - \overline{\mathcal{M}}_k,\ k = 2,\cdots,m, \tag{3.4}$$

we obtain different methods for solving (1.1). Similar to [12], we mainly establish four alternatives corresponding to different iterative methods as follows, based on the requirement that the tensor equation $\sum_{k=2}^{m} \mathcal{M}_k\mathbf{y}^{k-1} = \mathbf{g}$ is easy to solve and the fact that $\mathcal{A}_k\mathbf{x}^{k-1} = \mathcal{M}_k\mathbf{x}^{k-1} - \overline{\mathcal{M}}_k\mathbf{x}^{k-1}$.

Similar to the Jacobi method for solving linear equations $A\mathbf{x} = \mathbf{b}$, we can take $\mathcal{M}_k = \mathcal{D}_k$ in (3.4) for getting the following fixed point iteration

$$\mathbf{x}_{l+1} = \mathbf{J}_{\mathcal{D},\overline{\mathcal{D}}}(\mathbf{x}_l) \triangleq \mathbf{J}(\mathbf{x}_l) := [\{\mathcal{D}_k\}_{k=2}^m]^{-1}\left[\sum_{k=2}^{m} \overline{\mathcal{D}}_k\mathbf{x}_l^{k-1} + \mathbf{b}\right], \quad l = 0, 1, \cdots, \tag{3.5}$$

where $\mathcal{D}_k$ is the *diagonal part* of tensor $\mathcal{A}_k$ and the operator $[\{\mathcal{D}_k\}_{k=2}^m]^{-1}\left[\sum_{k=2}^{m} \overline{\mathcal{D}}_k\mathbf{x}_l^{i-1} + \mathbf{b}\right]$ has been defined as in (1.5). This *non-homogenous diagonal equation* can be solved easily. Therefore, we call (3.5) **the Jacobi-like method** for solving non-homogenous $\mathcal{M}$-equations.

Similarly, after taking $\mathcal{M}_k = \mathcal{L}_k$ in (3.4) we establish the following **Gauss-Seidel-like** method to solve non-homogenous $\mathcal{M}$-equations.

$$\mathbf{x}_{l+1} = \mathbf{G}_{\mathcal{L},\overline{\mathcal{L}}}(\mathbf{x}_l) \triangleq \mathbf{G}(\mathbf{x}_l) := [\{\mathcal{L}_k\}_{k=2}^m]^{-1} \left[\sum_{k=2}^m \overline{\mathcal{L}}_k \mathbf{x}_l^{k-1} + \mathbf{b}\right], \quad l = 0, 1, \cdots, \tag{3.6}$$

where $\mathcal{L}_k$ is the *lower triangular part* of tensor $\mathcal{A}_k$ and the operator $[\{\mathcal{L}_k\}_{k=2}^m]^{-1} \left[\sum_{k=2}^m \overline{\mathcal{L}}_k \mathbf{x}_l^{k-1} + \mathbf{b}\right]$ has been defined as in (1.5). It refers to *the non-homogenous lower triangle equation* (3.2) which can be solved by the algorithm of forward substitution.

Note 3.2.
- If $\mathcal{L}_k$ in (3.6) represents sum of the strictly *lower* and *diagonal parts* of tensor $\mathcal{A}_k$, (3.6) becomes a **simplified Gauss-Seidel-like** method.
- In contrast, if $\mathcal{M}_k = \mathcal{U}_k$ in (3.4) is the *upper part* (or the sum of *strictly upper* and *diagonal parts*) of tensor $\mathcal{A}_k$, we obtain **(simplified) backward Gauss Seidel-like** method to solve non-homogenous $\mathcal{M}$-equations.
- In each iteration (set as l) of these methods, $\{(\mathbf{x}_l)_i\}_{i=1}^n$, n components of the vector $\mathbf{x}_l$, can't be calculated independently. This indicates we must know the information about the first $j-1$ components $\{(\mathbf{x}_l)_i\}_{i=1}^{j-1}$ before calculating $(\mathbf{x}_l)_j$, $j = 1, \ldots, n$. This will cause these iterations to run longer than the Jacobi-like iteration.

With the introduction of successive over-relaxation parameter ω and based on

$$\mathcal{A}_k = \overbrace{\left[\frac{1}{\omega}\mathcal{D}_k + \mathcal{L}_k\right]}^{\mathcal{M}_k} - \overbrace{\left[(\frac{1}{\omega} - 1)\mathcal{D}_k + \mathcal{T}_k\right]}^{\overline{\mathcal{M}}_k},$$

we can establish the following **SOR-like** method to solve non-homogenous $\mathcal{M}$-equations,

$$\mathbf{x}_{l+1} = \left[\left\{\frac{1}{\omega}\mathcal{D}_k + \mathcal{L}_k\right\}_{k=2}^m\right]^{-1} \left[\sum_{k=2}^m \left[(\frac{1}{\omega} - 1)\mathcal{D}_k + \mathcal{T}_k\right] \mathbf{x}_l^{k-1} + \mathbf{b}\right], \quad l = 0, 1, \cdots, \tag{3.7}$$

where $\mathcal{D}_i$ and $\mathcal{L}_i$ are the *diagonal* and *strictly lower triangular parts* of tensor $\mathcal{A}_k$, respectively, and the operator of the above right-hand side is defined as in (1.5). It is *the non-homogenous lower triangle equation* (3.2) which can be solved by the algorithm of forward substitution.

Note 3.3. We need to solve a system of n one-dimensional polynomial equations about these proposed methods at each iteration. Indeed, like in [14] we transform this problem into finding the eigenvalues of its corresponding m-dimension companion matrix, which can be solved stably.[2]

4 Convergence Analysis

We now establish the convergence for these above methods. Consider a mapping $\phi : \mathbb{R}^n \to \mathbb{R}^n$. Let $\mathbf{x}_*$ be a fixed point of $\phi(\mathbf{x})$. $\mathbf{x}_*$ is called an *attracting fixed point* if there exists $\delta > 0$ such that $\{\mathbf{x}_l\}$ defined by $\mathbf{x}_{l+1} = \phi(\mathbf{x}_l)$ converges to $\mathbf{x}_*$ for any $\mathbf{x}_0 \in \{\mathbf{x} \in \mathbb{R}^n : \|\mathbf{x} - \mathbf{x}_*\| \leq \delta\}$.

[2] We solve these equations in Matlab by using the code: `roots()`.

Lemma 4.1 (Theorem 3.5 in [37])**.** *Suppose that $\phi : E \subset \mathbb{R}^n \to \mathbb{R}^n$ has a fixed point $\mathbf{x}_* \in int(E)$, where ϕ is differentiable. If $\sigma := \rho\left(J_{\phi(\mathbf{x}_*)}\right) < 1$, then $\mathbf{x}_*$ is a point of attraction of the iteration*

$$\mathbf{x}_{l+1} = \phi(\mathbf{x}_l), \quad l = 0, 1, \cdots .$$

Further, if $\sigma > 0$, then the convergence to $\mathbf{x}_*$ is linear with rate σ, where $J_{\phi(\mathbf{x}_*)}$ represents the Jacobian matrix of $\phi(\mathbf{x})$ at point $\mathbf{x}_*$ and $\rho\left(J_{\phi(\mathbf{x}_*)}\right)$ is its spectral radius.

By the Lemma 4.1, we can establish the following local convergence theory.

Theorem 4.2. *Assume $\mathbf{x}_*$ is the unique positive solution to the non-homogenous $\mathcal{M}$-equation (1.1) with $\mathbf{b} > \mathbf{0}$. Suppose that $\mathbf{x}_0 \in \mathbb{R}^n_{++}$ and given the following tensor splitting*

$$\mathcal{A}_k = \mathcal{M}_k - \overline{\mathcal{M}}_k, \ \overline{\mathcal{M}}_k \geq 0, \ k = 2, \cdots, m.$$

Then $\mathbf{x}_*$ is an attracting fixed point of the following iteration scheme

$$\mathbf{x}_{l+1} = \phi(\mathbf{x}_l) = [\{\mathcal{M}_k\}_{k=2}^m]_{++}^{-1} \left(\overline{\mathcal{M}}_m \mathbf{x}_l^{m-1} + \overline{\mathcal{M}}_{m-2} \mathbf{x}_l^{m-1} + \cdots + \overline{\mathcal{M}}_2 \mathbf{x}_l + \mathbf{b}\right), \quad l = 0, 1, 2, \cdots \tag{4.1}$$

provided that $\sum_{k=2}^{m} (k-1) \cdot \mathcal{A}_k \mathbf{x}_*^{k-1} > \mathbf{b}$.

Proof. To begin with, for any $k = 2, \cdots, m$, there always exist semi-symmetric tensors[3] $\widetilde{\mathcal{M}}_k$ and $\widehat{\mathcal{M}}_k$ such that for any $\mathbf{x} \in \mathbb{R}^n$,

$$\widetilde{\mathcal{M}}_k \mathbf{x}^{k-1} = \mathcal{M}_k \mathbf{x}^{k-1}, \quad \widehat{\mathcal{M}}_k \mathbf{x}^{k-1} = \overline{\mathcal{M}}_k \mathbf{x}^{k-1} \tag{4.2}$$

and

$$\frac{\partial(\mathcal{M}_k \mathbf{x}^{k-1})}{\partial \mathbf{x}} = (k-1)\widetilde{\mathcal{M}}_k \mathbf{x}^{k-2}, \quad \frac{\partial(\overline{\mathcal{M}}_k \mathbf{x}^{k-1})}{\partial \mathbf{x}} = (k-1)\widehat{\mathcal{M}}_k \mathbf{x}^{k-2} \tag{4.3}$$

hold, where the matrix $\widehat{\mathcal{M}}_k \mathbf{x}^{k-2}$ is defined as

$$\left(\widehat{\mathcal{M}}_k \mathbf{x}^{k-2}\right)_{ij} = \sum_{i_3, \cdots, i_k = 1}^{n} (\widehat{\mathcal{M}}_k)_{i,j,i_3,\cdots,i_k} x_{i_3} \cdots x_{i_k},$$

which is similar to the definition of $\widetilde{\mathcal{M}}_k \mathbf{x}^{k-2}$.

In addition, define a mapping $\mathbf{F} : \mathbb{R}^n_{++} \times \mathbb{R}^n_{++} \to \mathbb{R}^n; (\mathbf{x}, \mathbf{y}) \mapsto \mathbf{F}(\mathbf{x}, \mathbf{y})$ with

$$\mathbf{F}(\mathbf{x}, \mathbf{y}) := \sum_{k=2}^{m} \mathcal{M}_k \mathbf{y}^{k-1} - \left(\sum_{k=2}^{m} \overline{\mathcal{M}}_k \mathbf{x}^{k-1} + \mathbf{b}\right).$$

We can find that $\mathbf{F}(\cdot)$ is differentiable in its domain and $\mathbf{F}(\mathbf{x}_*, \mathbf{x}_*) = \mathbf{0}$. By direct computation, we have

$$\begin{aligned}
\mathbf{F}_{\mathbf{x}}|_{\mathbf{x}=\mathbf{x}_*} &= -\frac{\partial}{\partial \mathbf{x}} \left(\sum_{k=2}^{m} \overline{\mathcal{M}}_k \mathbf{x}^{k-1}\right) |_{\mathbf{x}=\mathbf{x}_*} \\
&= -\frac{\partial}{\partial \mathbf{x}} \left(\sum_{k=2}^{m} \widehat{\mathcal{M}}_k \mathbf{x}^{k-1}\right) |_{\mathbf{x}=\mathbf{x}_*} \\
&= -\sum_{k=2}^{m} (k-1) \widehat{\mathcal{M}}_k \mathbf{x}_*^{k-2}
\end{aligned}$$

[3] A tensor $\mathcal{A} \in \mathbb{R}^{[k,n]}$ is called semi-symmetric, if $\mathcal{A}_{i_1 i_2 \cdots i_m} = \mathcal{A}_{i_1 \pi'(i_2 \cdots i_m)}$ for any $i_1 \in \{1, \cdots, n\}$ and $\forall \pi' \in \Pi_{m-1}$, where Π_{m-1} is the permutation group of $m-1$ indices $\{i_2, \cdots, i_m\}$.

and

$$\mathbf{F}_{\mathbf{y}}|_{\mathbf{y}=\mathbf{x}_*} = \frac{\partial}{\partial \mathbf{y}} \left(\sum_{k=2}^{m} \mathcal{M}_k \mathbf{y}^{k-1} \right) |_{\mathbf{y}=\mathbf{x}_*} = \frac{\partial}{\partial \mathbf{y}} \left(\sum_{k=2}^{m} \widetilde{\mathcal{M}}_k \mathbf{y}^{k-1} \right) |_{\mathbf{y}=\mathbf{x}_*} = \sum_{k=2}^{m} (k-1) \widetilde{\mathcal{M}}_k \mathbf{x}_*^{k-2}$$

since (4.2) and (4.3). From $[\mathbf{F}_{\mathbf{y}}|_{\mathbf{y}=\mathbf{x}_*}]$ is a Z-matrix and

$$\begin{aligned} [\mathbf{F}_{\mathbf{y}}|_{\mathbf{y}=\mathbf{x}_*}] \cdot \mathbf{x}_* &= \sum_{k=2}^{m} (k-1) \widetilde{\mathcal{M}}_k \mathbf{x}_*^{k-2} \cdot \mathbf{x}_* \\ &= \sum_{k=2}^{m} (k-1) \mathcal{M}_k \mathbf{x}_*^{k-1} \\ &> \sum_{k=2}^{m} (k-1) \overline{\mathcal{M}}_k \mathbf{x}_*^{k-1} + \mathbf{b} > \mathbf{0} \end{aligned}$$

we obtain $\mathbf{F}_{\mathbf{y}}|_{\mathbf{y}=\mathbf{x}_*}$ is a nonsingular M-matrix and $[\mathbf{F}_{\mathbf{y}}|_{\mathbf{y}=\mathbf{x}_*}]^{-1} > \mathbf{0}$ (see Ref. [38] for details). Therefore, there exist open neighborhoods $U, V \subset \mathbb{R}^n_{++}$ and a continuous mapping $\phi : U \to V; \mathbf{x} \mapsto \mathbf{y} = \phi(\mathbf{x})$ such that $\phi(\mathbf{x})$ is the solution to the following polynomial equations

$$\sum_{k=2}^{m} (k-1) \mathcal{M}_k \mathbf{z}^{k-2} = \sum_{k=2}^{m} (k-1) \overline{\mathcal{M}}_k \mathbf{x}^{k-1} + \mathbf{b} \quad (\mathbf{x} \text{ is given and } \mathbf{z} \text{ is the variable}),$$

and $\phi(\mathbf{x})$ is differentiable at $\mathbf{x}_*$ with the Jacobian matrix at the point $\mathbf{x}_*$ being

$$J_{\phi(\mathbf{x}_*)} = -[\mathbf{F}_{\mathbf{y}}|_{\mathbf{y}=\mathbf{x}_*}]^{-1} [\mathbf{F}_{\mathbf{x}}|_{\mathbf{x}=\mathbf{x}_*}] = \left[\sum_{k=2}^{m} (k-1) \cdot \widetilde{\mathcal{M}}_k \mathbf{x}_*^{k-2} \right]^{-1} \left(\sum_{k=2}^{m} (k-1) \cdot \widehat{\mathcal{M}}_k \mathbf{x}_*^{k-2} \right),$$

which is a nonnegative matrix as $\widehat{\mathcal{M}}_k$ is nonnegative, according to Theorem 5.2.4 in [32].

Finally, since

$$\begin{aligned} \mathbf{0} &< \sum_{k=2}^{m} (k-1) \cdot \widehat{\mathcal{M}}_k \mathbf{x}_*^{k-2} \cdot \mathbf{x}_* = \sum_{k=2}^{m} (k-1) \cdot \overline{\mathcal{M}}_k \mathbf{x}_*^{k-1} \\ &< \sum_{k=2}^{m} (k-1) \cdot \mathcal{M}_k \mathbf{x}_*^{k-1} - \mathbf{b} \\ &\leq \theta \sum_{k=2}^{m} (k-1) \cdot \mathcal{M}_k \mathbf{x}_*^{k-1} = \theta \sum_{k=2}^{m} (k-1) \cdot \widetilde{\mathcal{M}}_k \mathbf{x}_*^{k-1} \\ &= \left(\sum_{k=2}^{m} (k-1) \cdot \widetilde{\mathcal{M}}_k \mathbf{x}_*^{k-2} \right) (\theta \mathbf{x}_*) \end{aligned}$$

with $0 \leq \theta < 1$, $J_{\phi(\mathbf{x}_*)} \cdot \mathbf{x}_* \leq \theta \mathbf{x}_*$. Therefore, the spectral radius $\rho(J_{\phi(\mathbf{x}_*)}) \leq \theta < 1$ based on Corollary 8.1.29 in [19]. From *Lemma 4.1* we know the proof is completed. □

Similarly, $\overline{\mathcal{D}}_k$, $\overline{\mathcal{L}}_k$ and $\left[(\frac{1}{\omega} - 1)\mathcal{D}_k + \mathcal{T}_k\right]$ ($\omega \in (0, 1]$) are constructed as nonnegative tensors, hence we have the following corollary.

Corollary 4.3. *Assume $\mathbf{x}_*$ is the unique positive solution to the non-homogenous $\mathcal{M}$-equation* (1.1) *with $\mathbf{b} > \mathbf{0}$ and provided that $\sum_{k=2}^{m}(k-1)\cdot\mathcal{A}_k\mathbf{x}_*^{k-1} > \mathbf{b}$. Then $\mathbf{x}_*$ is an attracting fixed point of the Jacobi-like, Gauss-Seidel-like and simplified Gauss-Seidel-like iterations, respectively.*

For choosing a proper over-relaxation factor ω, we have some constraints like

- $\omega > 0$,
- $(\frac{1}{\omega}\mathcal{D}_k + \mathcal{L}_k)$ are $\mathcal{M}$-tensors for $k = 2, \cdots, m$,
- $\sum_{k=2}^{m}\left[(\frac{1}{\omega} - 1)\mathcal{D}_k + \mathcal{T}_k\right]\mathbf{x}_l^{k-1} + \mathbf{b} > \mathbf{0}$ $(l = 1, 2, \cdots)$.

Apparently we can get the following theorem.

Theorem 4.4. *For any $\omega \in (0, 1]$, the SOR-like method is convergent with $\mathbf{b} > 0$ for the non-homogenous $\mathcal{M}$-equation* (1.1) *provided that $\sum_{k=2}^{m}(k-1)\cdot\mathcal{A}_k\mathbf{x}_*^{k-1} > \mathbf{b}$.*

Nevertheless, Theorem 4.4 does not mean

- the optimal $\omega \in (0, 1]$,
- and the SOR-like method does not converge in any interval $D \nsubseteq (0, 1]$ for Eq. (1.1).

Indeed, it's highly possible that we get the optimal ω in an interval $D \nsubseteq (0, 1]$. E.g., according to Test Two of Example 5.1 in section 5, we can see $\omega_{\text{opt}} = 2.11 \notin (0, 1]$ for the cases of $m = 3$ and $n = 16$ according to Fig. 3(d).

5 Numerical Experiments

In this section, all experiments are implemented in MATLAB R2016a with a machine precision 10^{-16} on a personal computer with 2.20 GHz central processing unit (Intel(R) Core(TM) i5-5200U), 4GB memory and windows 10.1903 operating system.

The iteration stopping criterion is that the lth iterative residual satisfies

$$\left\|\mathbf{b} - \sum_{k=2}^{m}\mathcal{A}_k\mathbf{x}_l^{k-1}\right\|_2 < \eta,$$

where η is set to be different values in the following two examples. We compare the Jacobi-like (J-like), Gauss-Seidel-like (G-S-like), backward Gauss-Seidel-like (backward G-S-like), simplified Gauss-Seidel-like (Sim. G-S-like), backward simplified Gauss-Seidel-like (backward Sim. G-S-like), and SOR-like methods. The number of iteration steps and total CPU time (unit: seconds) are abbreviated as IT and CPU, respectively.

Example 5.1. Similar to Examples 4.1 and 4.2 in [12] and Example 6.1 in [40], we construct nonsingular $\mathcal{M}$-tensors $\mathcal{A}_k = s_k\mathcal{I}_k - \mathcal{B}_k$ $(k = 2, 3, \cdots, m)$ as follows. Moreover, we set $\eta = \texttt{1e-12}$ and use tensor toolbox (see Ref. [4]) for computing $\mathcal{A}\mathbf{x}^{m-1}$, i.e.,

$$\mathcal{A}\mathbf{x}^{m-1} = \texttt{ttv(A, } \underbrace{\{x, \cdots, x\}}_{(m-1)\ \texttt{copies}} \texttt{, } [m, m-1, \cdots, 2]\texttt{)}.$$

- **Test One**
  The initial iteration guess $\mathbf{x}_0$ is chosen from $\{\mathbf{x}_0^{(1)}, \mathbf{x}_0^{(2)}, \mathbf{x}_0^{(3)}, \mathbf{x}_0^{(4)}\} = \{0 \cdot \mathbf{1}, 0.5 \cdot \mathbf{1}, 5 \cdot \mathbf{1}, 10 \cdot \mathbf{1}\}$. Let $m = 3$, $\mathbf{b} = \mathbf{1}$, $\mathcal{B}_k \in \mathbb{R}^{[k,10]}$ be a nonnegative tensor with

$$(\mathcal{B}_k)_{i_1 \cdots i_k} = |\tan(i_1 + \cdots + i_k)|, \quad k = 2, 3,$$

  and define $\mathcal{A}_3 = 1500\mathcal{I}_3 - \mathcal{B}_3$. Example 4.2 in [12] shows that it is a symmetric nonsingular $\mathcal{M}$-tensor. In addition, $\mathcal{A}_2 = 260I - \mathcal{B}_2$ is an M-matrix under $\rho(\mathcal{B}_2) \approx 241.4184$.

- **Test Two**
  The initial iteration guess is set to be $\mathbf{x}_0 = 0 \cdot \mathbf{1} = \mathbf{0} = [0, \cdots, 0]_n^\top$. Let $m = 3,\ 4$. First, we generate 3 nonnegative tensors $\mathcal{B}_k \in \mathbb{R}_n^{[k,n]}$ containing random values drawn from the standard uniform distribution on $(0, 1)$. Next, set the scalar

$$s_k = (1 + \varepsilon) \cdot \max_{i=1,2,\cdots,n} (\mathcal{B}_k \mathbf{1}^{k-1})_i, \varepsilon > 0, \quad k = 2, 3, 4.$$

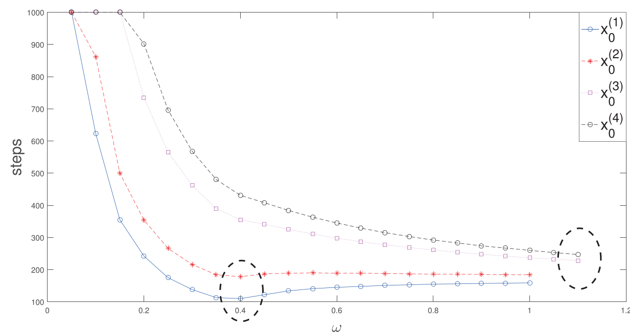
  Here $\mathcal{A}_k$ is a nonsingular $\mathcal{M}$-tensor (see Example 4.2 in [12]). During the procession of Test Two, we take $n \in \{3, 4, 6, 8, 12, 16, 24, 32\}$ and $\varepsilon = 0.01$, which are basically same as in Ref. [12]. In addition, we choose the right hand-side vector $\mathbf{b} = \texttt{rand(n,1)}$.

- **Test Three**
  The initial iteration guess is set to be $\mathbf{x}_0 = \mathbf{0}$ and Let $m = 3, 4, 5$. We take an $\mathcal{M}$-tensor like in Example 6.1 of [40], i.e., $s_k = n^{k-1}$ and each entry of $\mathcal{B}_k$ is

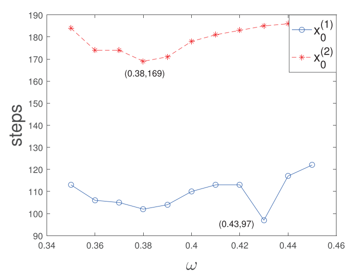
$$(\mathcal{B}_k)_{i_1 i_2 \cdots i_k} = |\sin(i_1 + i_2 + \cdots + i_k)|.$$

  $\mathcal{A}_k$ is an $\mathcal{M}$-tensor, which is proved in [41]. Meanwhile, we choose the right-hand side vector $\mathbf{b} = 10 \cdot \mathbf{1}$.

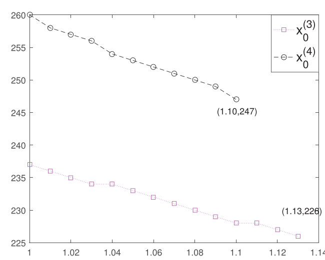
In Test One, we take the value of over-relaxation factor ω from 0 to 1.2 with stepsize 0.05 when using the SOR-like method for each initial iteration guess, and the result is displayed in Fig. 1(a). We can find the optimal parameters ω_{opt} should be in the intervals $[0.35, 0.45]$ and $[1.00, 1.20]$, approximately. Therefore, we search the value ω_{opt} from 0.3 to 0.4, and from 1.00 to 1.20 with stepsize 0.01 again. In Fig. 1(b) and (c) we can see that ω_{opt} equals to $0.43, 0.38, 1.13, 1.10$, respectively, with respect to each initial iteration vector $\mathbf{x}_0$. Particularly, the SOR-like method is invalid when $\omega > 1.13$ for $\mathbf{x}_0 = \mathbf{x}_0^{(3)}$, and ditto for $\mathbf{x}_0 = \mathbf{x}_0^{(4)}$ when $\omega > 1.10$. Therefore, there are only parts of curves in Fig. 1(c).



(a) $\omega \in (0, 1.2)$ with stepsize= 0.05.

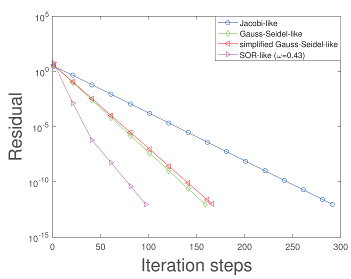


(b) $\omega \in (0.35, 0.45)$ with stepsize= 0.01.

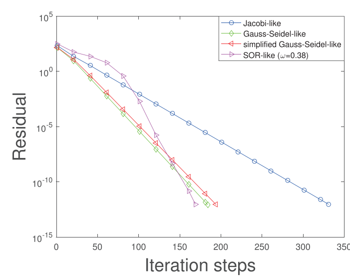


(c) $\omega \in (1, 1.14)$ with stepsize= 0.01.

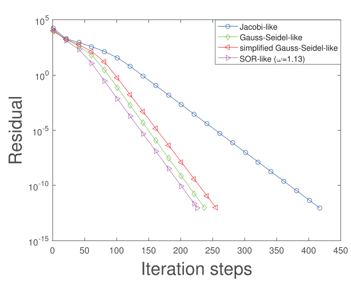
Figure 1: Optimal ω for the SOR-like method in Test One for each initial guess.



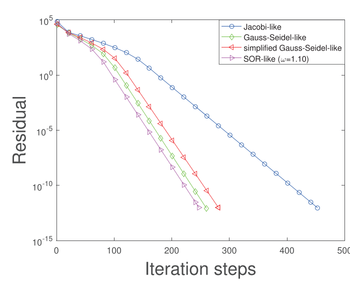
(a) $\mathbf{x}_0 = \mathbf{x}_0^{(1)}$.



(b) $\mathbf{x}_0 = \mathbf{x}_0^{(2)}$.



(c) $\mathbf{x}_0 = \mathbf{x}_0^{(3)}$.



(d) $\mathbf{x}_0 = \mathbf{x}_0^{(4)}$.

Figure 2: Results for Test One

The relationships between residual and IT for the Jacobi-like, Gauss-Seidel-like, simplified Gauss-Seidel-like, and SOR-like methods with different initial iteration vectors are displayed in Fig. 2. It shows that the SOR-like method is the best with $\omega_{\rm opt}$ in terms of IT among these methods, while the Jacobi-like method's performance is not very well. Unexceptionably, the effects of the other methods, between those of the SOR-like and Jacobi-like methods, are better than that of the Jacobi-like method but not beyond the SOR-like method.

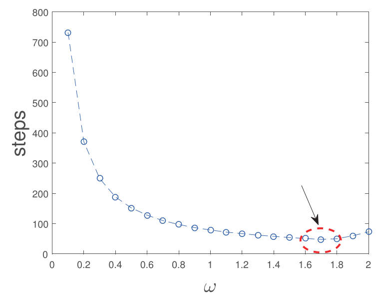
In Test Two, we first search the value of $\omega_{\rm opt}$ like in Test One when using the SOR-like method for the cases of $m \in \{3,4\}$, $n \in \{3,4,6,8,12,16,24,32\}$. The results are showed in Figs. 3-4 and Tab. 2. The relationships among residual, CPU, IT for the Jacobi-like, (backward) Gauss-Seidel-like, (backward) simplified Gauss-Seidel-like, and SOR-like methods are displayed in the Figs. 5-6 and Tab. 1, respectively. According to these figures and tables, we can obtain the following conclusions.

- From the perspective of IT, the conclusion is similar to Test One, and the main reason is discussed scrupulously in the 3rd item of *Note* 3.2.

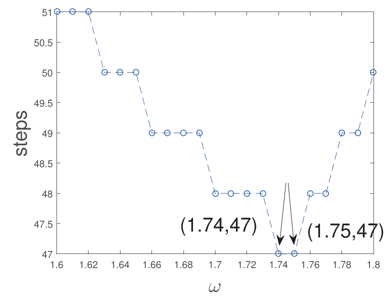
- From the perspective of CPU, the Jacobi-like method takes the least although it has large numbers of IT. Meanwhile, we can see from Figs. 5-6 and Tab. 1 that as the dimension and order (i.e., n changed from 3 to 32, and $m \in \{3,4\}$) get larger, the other methods need more CPU times. This is mainly because we can use parallel computation in the Jacobi-like method but can not in other methods.

Table 1: Results for Test Two

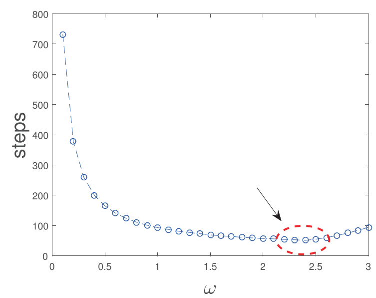
		$m=3$				$m=4$			
methods	n	4	8	16	32	4	8	16	32
J-like	IT	114	141	248	356	226	264	468	798
	CPU	0.41	0.57	1.01	1.75	1.27	1.33	2.57	8.90
G-S-like	IT	66	84	155	227	136	171	323	565
	CPU	0.51	1.98	23.61	276.50	2.31	22.22	576.07	13430.46
Sim. G-S-like	IT	79	93	166	235	184	197	348	587
	CPU	0.43	1.69	22.95	246.52	1.96	16.89	500.45	14115.91
backward G-S-like	IT	67	86	156	226	138	176	321	563
	CPU	0.45	1.84	19.61	208.37	2.44	23.86	629.88	14286.67
backward Sim. G-S-like	IT	82	97	167	235	181	200	346	586
	CPU	0.37	1.35	18.26	197.49	1.83	16.58	506.21	13411.29
SOR-like ($\omega_{\rm opt}$)	IT	47	51	74	88	72	82	152	309
	CPU	0.278	3.17	14.03	102.39	1.74	8.47	174.19	12895.84



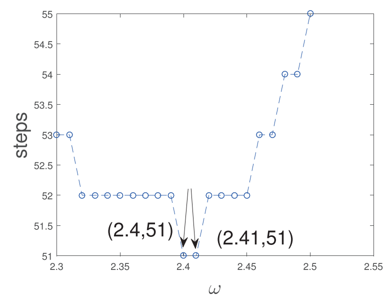
(a) $n = 4, \omega \in (0, 2)$ with stepsize= 0.1.



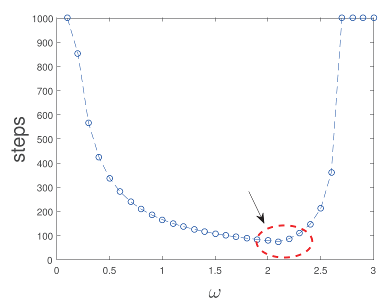
(b) $n = 4, \omega \in (1.6, 1.8)$ with stepsize= 0.01.



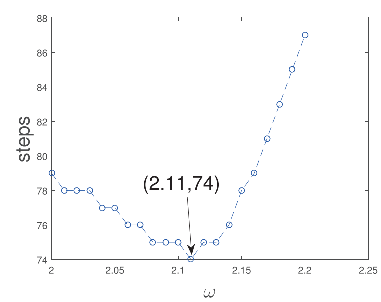
(c) $n = 8, \omega \in (0, 3)$ with stepsize= 0.1.



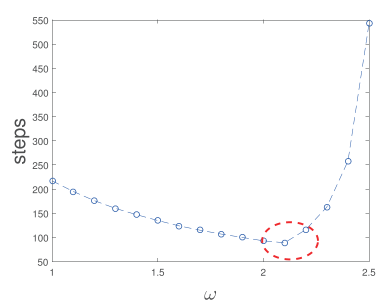
(d) $n = 8, \omega \in (2.3, 2.5)$ with stepsize= 0.01.



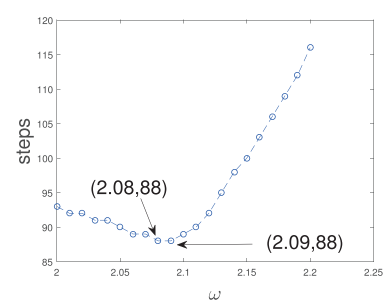
(e) $n = 16, \omega \in (0, 3)$ with stepsize= 0.1.



(f) $n = 16, \omega \in (2, 2.2)$ with stepsize= 0.01.

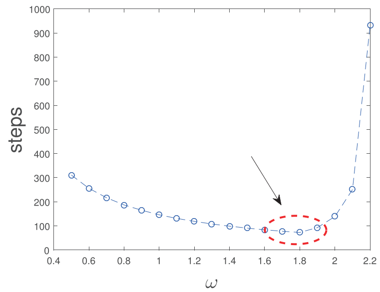


(g) $n = 32, \omega \in (1, 2.5)$ with stepsize= 0.1.

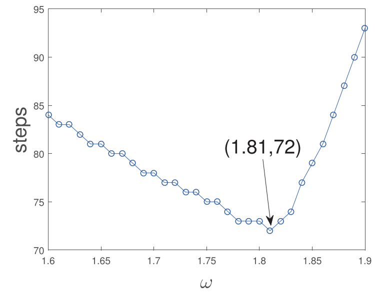


(h) $n = 32, \omega \in (2, 2.2)$ with stepsize= 0.01.

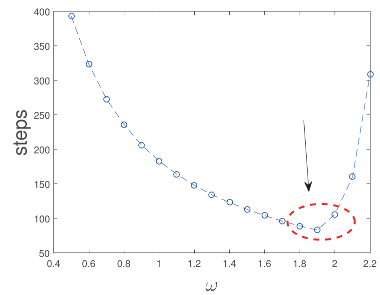
Figure 3: Optimal ω for the SOR-like method in Test Two with $m = 3$.



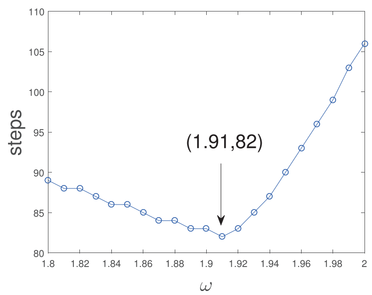
(a) $n = 4$,$\omega \in (0.4, 2.2)$ with stepsize= 0.1.



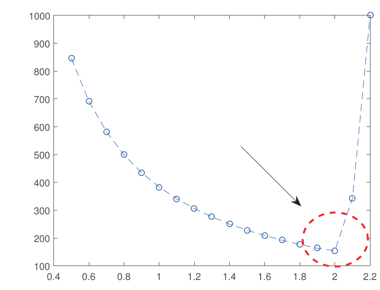
(b) $n = 4$,$\omega \in (1.6, 1.8)$ with stepsize= 0.01.



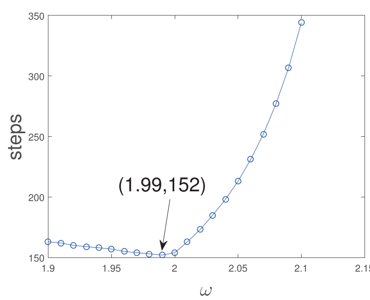
(c) $n = 8$,$\omega \in (0.4, 2.2)$ with stepsize= 0.1.



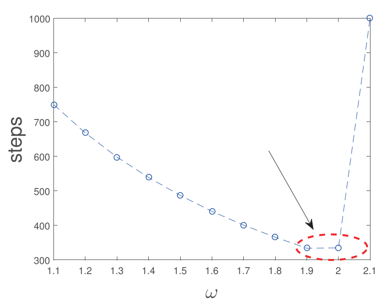
(d) $n = 8$,$\omega \in (1.8, 2)$ with stepsize= 0.01.



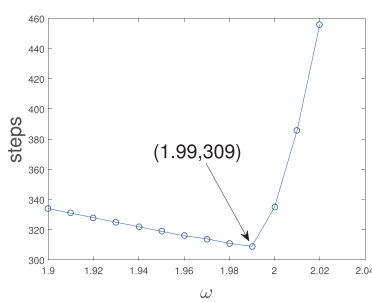
(e) $n = 16$,$\omega \in (0.4, 2.2)$ with stepsize= 0.1.



(f) $n = 16$,$\omega \in (1.9, 2.1)$ with stepsize= 0.01.

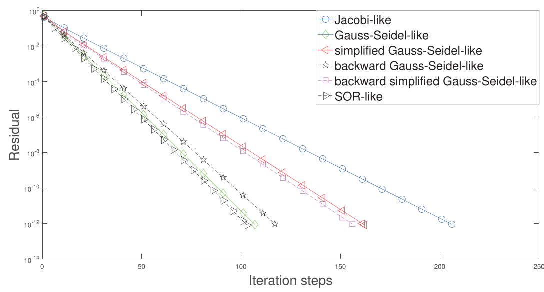


(g) $n = 32$,$\omega \in (1, 2.5)$ with stepsize= 0.1.

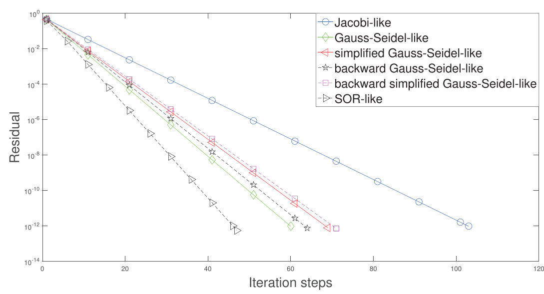


(h) $n = 32$,$\omega \in (1.9, 2.02)$ with stepsize= 0.01.

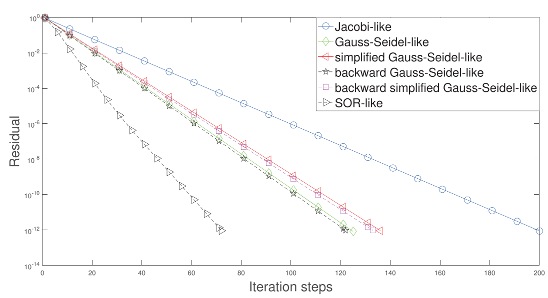
Figure 4: Optimal ω for the SOR-like method in Test Two with $m = 4$.



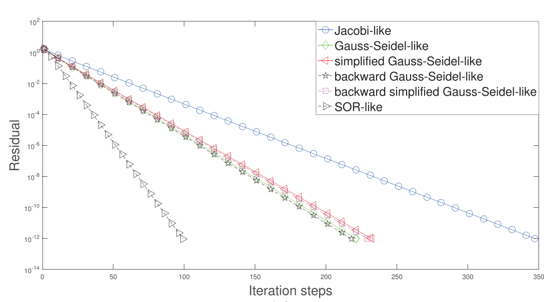
(a) $m = 3$, $n = 3$.



(b) $m = 3$, $n = 6$.

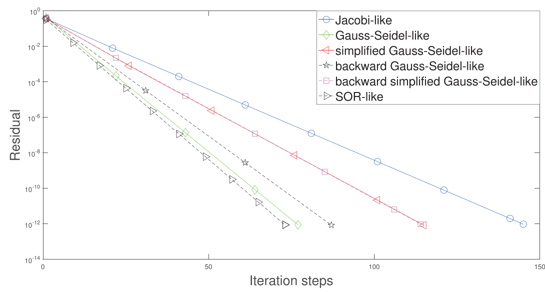


(c) $m = 3$, $n = 12$.

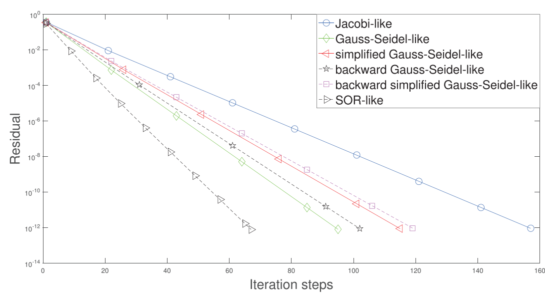


(d) $m = 3$, $n = 24$.

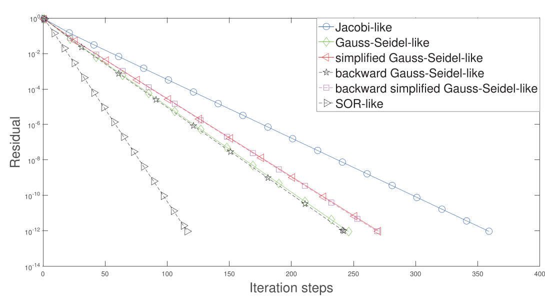
Figure 5: Partial outer iteration history of six methods in Test Two.



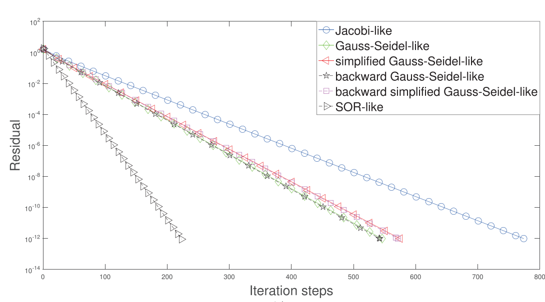
(a) $m = 4$, $n = 3$.



(b) $m = 4$, $n = 6$.



(c) $m = 4$, $n = 12$.



(d) $m = 4$, $n = 24$.

Figure 6: Partial outer iteration history of six methods in Test Two.

Table 2: Partial optimal parameters ω for the SOR-like method in Test Two

	$m=3$				$m=4$			
n	3	6	12	24	3	6	12	24
ω_{opt}	1.81	1.83	2.15	2.04	1.65	2.00	1.95	1.98

Tab. 3 lists these values of ω_{opt} in the last column for all the cases in Test Three. The conclusion in Test Three is almost identical to that in Test Two. Here we only need to explain that, when the order is 4 or 5, all the methods except the Jacobi-like method need more CPU times as the dimension gets larger. Thus, we let $n \in \{2, 4, 8, 12, 16, 20\}$ and $n \in \{2, 4, 8, 12\}$ corresponding to the cases of $m = 4$ and $m = 5$, respectively.

Table 3: Results for Test Three

	methods	J-like		G-S-like		Sim. G-S-like		SOR-like		
	n	IT	CPU	IT	CPU	IT	CPU	IT	CPU	ω_{opt}
$m=3$	5	72	0.22	45	0.41	56	0.40	29	0.30	1.39
	20	70	0.32	47	13.77	50	12.88	27	5.43	1.31
	40	71	0.31	49	128.00	50	118.56	27	47.34	1.33
	60	71	0.51	49	425.47	50	389.94	27	161.07	1.31
	80	72	0.62	50	878.63	51	880.12	27	389.52	1.32
	100	72	0.80	51	1760.30	51	1667.88	27	736.32	1.31
$m=4$	2	57	1.20	35	0.35	51	0.37	28	0.25	1.41
	4	72	0.40	48	0.96	62	0.70	34	0.55	1.43
	8	68	0.48	49	6.79	56	4.83	30	3.73	1.39
	12	69	0.47	50	30.25	55	22.72	30	17.48	1.37
	16	70	0.47	52	87.99	55	79.18	30	56.55	1.38
	20	70	0.59	52	209.18	55	180.60	30	114.08	1.37
$m=5$	2	66	0.64	38	1.04	63	0.76	39	0.45	1.39
	4	73	0.66	51	2.10	66	2.49	37	1.22	1.44
	8	67	3.59	51	45.60	57	45.16	32	24.29	1.40
	12	69	23.19	54	334.40	59	363.25	32	154.13	1.42

Note 5.2. In these three tests of Example 5.1, we need to make the following explanation.

- About ω. On the one hand, we find that the value of ω_{opt} seems to change as the initial iteration vector changes according to Test One. For every iteration step, these optimal parameters ω in each table are computed according to the least iteration steps before the iterations start, hence the CPU time does not include the time spent on searching ω_{opt}. Moreover, we need to find the optimal parameter again if problems are changed.

  On the other hand, we give a numerical strategy to find the optimal ω. We first search in a longer interval like $[0, 3]$ with a larger stepsize 0.05 so that the ω_{opt} is located in a narrow interval. Then the numerical optimal parameter is obtained by reducing the search stepsize (such as 0.01) again in this narrow interval.

- About symmetric. We call the non-homogenous $\mathcal{M}$-equation (1.1) a symmetric system if all the coefficient tensors are symmetric ([5, 7, 33]). There is no doubt that Test Two is not a symmetric system but the other two are. In the case of symmetric systems, the Gauss-Seidel-like and simplified Gauss-Seidel-like methods are equivalent to the backward Gauss-Seidel-like and backward simplified Gauss-Seidel-like methods, respectively. Therefore, we do not consider the backward Gauss-Seidel-like and backward simplified Gauss-Seidel-like methods in Tests One and Three.

Example 5.3 (The nonlinear Poisson equation in (1.2))**.** In this example, we set $\eta = \texttt{1e-4}$ which is larger than in *Example* 5.1, because the CPU time and iteration steps will be more as the dimension and order get larger. Therefore, we choose `1e-4` through a lot of experiments. In addition, we set $m = 3, 4, 5, 6$, and all the numerical results are reported in Tabs. 4–7. The initial iteration guess is set to be $\mathbf{x}_0 = \mathbf{0}$.

The conclusion in terms of the iteration steps is also similar to that in *Example* 5.1. However, from the perspective of the CPU time, we find (i) the SOR-like method takes the least CPU time, and (ii) the CPU times in the Gauss-Seidel-like, simplified Gauss-Seidel-like methods are not so much but a little more than that in the SOR-like method. On the contrary, the Jacobi-like method takes the most. This is mainly because the nonlinear Poisson equation in (1.2) is a sparse system. Moreover, the computation complexity of $\mathcal{A}_k\mathbf{x}^{k-1}$ is reduced to $O(n)$, which significantly improves the computational efficiency.

Furthermore, we can see that when $n = 400$ in Tabs. 4–7, both the CPU time and IT are smaller than those of the case of $n = 200$. Indeed, when $n = 400$ in Eq. (1.1), some entries of the right-hand vector are

$$b_i = \frac{1}{(n-1)^2} = \frac{1}{399^2} \approx \texttt{9.92e-6},\ i = 2, \ldots, 399,$$

which implies the exact solution $\mathbf{x}_*$ must be close to zero.

Table 4: Results for Example Two with $m = 3$

methods	J-like		G-S-like		Sim. G-S-like		SOR-like		
n	IT	CPU	IT	CPU	IT	CPU	IT	CPU	ω_{opt}
5	34	0.22	19	0.38	22	0.24	11	0.05	1.35
20	383	1.67	193	2.02	221	1.05	44	0.27	1.84
40	1181	4.98	592	11.01	670	3.25	100	0.84	1.93
60	2200	11.96	1101	29.47	1236	7.72	160	1.49	1.96
80	3338	23.15	1670	70.22	1863	14.82	221	3.24	1.97
100	4523	41.26	2263	120.14	2508	23.54	280	3.88	1.98
150	7355	109.31	3679	336.39	4020	63.57	385	13.52	1.98
200	9527	242.51	4765	592.14	5135	133.08	426	19.98	1.98
400	4225	4853.71	2113	7627.86	2139	401.74	95	27.46	1.95

Table 5: Results for Example Two with $m = 4$

methods	J-like		G-S-like		Sim. G-S-like		SOR-like		
n	IT	CPU	IT	CPU	IT	CPU	IT	CPU	ω_{opt}
5	51	0.67	27	2.40	34	0.06	13	1.26	1.60
20	426	5.80	214	0.66	252	0.59	41	4.43	1.96
40	1248	33.77	625	2.12	720	2.58	87	10.19	2.03
60	2284	8.94	1143	4.53	1302	5.31	143	17.80	2.03
80	3435	16.28	1719	8.34	1941	9.99	207	27.50	2.02
100	4630	25.97	2316	13.42	2596	15.80	260	35.88	2.02
150	7477	47.44	3740	27.22	4120	29.50	368	2.82	2.01
200	9651	90.24	4827	47.81	5232	53.81	391	3.81	2.01
400	4284	78.46	2142	39.48	2169	41.14	97	1.60	1.95

Table 6: Results for Example Two with $m = 5$

methods	J-like		G-S-like		Sim. G-S-like		SOR-like		
n	IT	CPU	IT	CPU	IT	CPU	IT	CPU	ω_{opt}
5	51	0.09	27	0.04	35	0.06	13	0.03	1.59
20	426	1.06	214	0.53	252	0.72	41	0.10	1.97
40	1248	4.42	625	2.24	721	2.69	88	0.30	2.03
60	2284	9.59	1143	5.01	1304	5.73	145	0.63	2.03
80	3435	17.11	1719	9.28	1943	11.05	210	1.19	2.02
100	4631	27.70	2317	14.30	2599	16.83	263	1.63	2.02
150	7477	64.49	3740	31.17	4123	38.08	371	4.42	2.01
200	9652	100.91	4827	53.63	5235	59.18	393	4.45	2.01
400	4284	81.27	2142	42.56	2169	43.54	97	1.95	1.95

Table 7: Results for Example Two with $m = 6$

methods	J-like		G-S-like		Sim. G-S-like		SOR-like		
n	IT	CPU	IT	CPU	IT	CPU	IT	CPU	ω_{opt}
5	51	0.10	27	0.06	35	0.07	13	0.03	1.60
20	426	1.10	214	0.60	253	0.72	41	0.11	1.97
40	1248	4.93	625	2.34	721	2.55	88	0.31	2.03
60	2284	10.07	1143	5.28	1304	6.16	145	0.65	2.03
80	3435	19.31	1719	9.94	1944	10.53	210	1.11	2.02
100	4631	31.68	2317	15.38	2599	18.90	263	2.08	2.02
150	7478	70.49	3740	46.41	4124	54.46	371	4.47	2.01
200	9652	116.50	4827	57.86	5235	58.75	393	4.46	2.01
400	4284	90.38	2142	43.28	2169	45.08	97	2.03	1.95

6 Conclusions

In this paper, we first prove the existence and uniqueness of a positive solution to the non-homogenous $\mathcal{M}$-equation

$$\mathcal{A}_m\mathbf{x}^{m-1} + \mathcal{A}_{m-1}\mathbf{x}^{m-2} + \cdots + \mathcal{A}_3\mathbf{x}^2 + \mathcal{A}_2\mathbf{x} = \mathbf{b},$$

with a positive right-hand side vector. In addition, we expand some classical splitting methods to obtain the Jacobi-like, Gauss-Seidel-like, simplified Gauss-Seidel-like, and SOR-like methods for solving the tensor equations and give their convergence analyses. Next, we find the SOR-like method with the optimal over-relaxation factor ω in terms of iteration steps performs the best among these methods. The Jacobi-like method needs the most, while the other methods do not differ much and are between the SOR-like and Jacobi-like methods. From the perspective of the CPU time, the Jacobi-like method takes the least CPU time in the non-sparse case, while SOR-like method needs the most. The SOR-like method takes the least CPU time in the sparse case, and the CPU times of the other methods are also not so much. The Jacobi-like method takes the most CPU time.

Finally, we give a numerical strategy in Note 5.2 about searching the optimal parameter ω in the SOR-like method. How to obtain the optimal parameter is an interesting topic and worth studying. Hadjidimos and other scholars have studied the optimal ω for saddle point problems, e.g. see [18] and references therein. Nevertheless, whether those techniques for the saddle point problems can be generalized to tensor problems, and how to calculate the optimal ω also need to be further investigated. That is one direction in our future study.

Acknowledgements

The authors are indebted to Prof. Zhenghai Huang for his kind help. We are also very grateful to anonymous reviewers for their valuable comments and suggestions, which have important guiding significance to our research.

References

[1] H. Amann, Fixed point equations and nonlinear eigenvalue problems in ordered banach spaces, *SIAM Rev.* 18 (1976) 620–709.

[2] P. Aubry, D. Lazard and M.M. Maza, On the theories of triangular sets, *J. Symb. Comput.* 28 (1999) 105–124.

[3] P. Aubry and M.M. Maza, Triangular sets for solving polynomial systems: a comparative implementation of four methods, *J. Symb. Comput.* 28 (1999) 125–154.

[4] B.W. Bader and T.G. Kolda, Algorithm 862: Matlab tensor classes for fast algorithm prototyping, *ACM Trans. Math. Software* 32 (2006) 635–653.

[5] K. Batselier and N. Wong, Symmetric tensor decomposition by an iterative eigendecomposition algorithm, *ACM Trans. Math. Software* 308 (2016) 69–82.

[6] C. Chen and M.M. Maza, Algorithms for computing triangular decomposition of polynomial systems, *J. Symb. Comput.* 47 (2012) 610–642.

[7] P. Comon, G.H. Golub, L.H. Lim and B. Mourrain, Symmetric tensors and symmetric tensor rank, *SIAM J. Matrix Anal. Appl.* 30 (2008) 1254–1279.

[8] C. Deng, X.F. He, and J.W. Han, Tensor space model for document analysis, in *International Conference on Research and Development in Information Retrieval*, 2006, pp. 625–626.

[9] M.Y. Deng and X.P. Guo, On hss-based iteration methods for two classes of tensor equations, *East Asian J. Appl. Math.* 10 (2020) 381–398.

[10] W. Ding, J.J. Liu, L. Qi and H. Yan, Elasticity $\mathcal{M}$-tensors and the strong ellipticity condition, *Appl. Math. Comput.* 373 (2020) 1–11.

[11] W. Ding and Y. Wei, $\mathcal{M}$-tensors and nonsingular $\mathcal{M}$-tensors, *Linear Algebra Appl.* 439 (2013) 3264–3278.

[12] W. Ding and Y. Wei, Solving multi-linear systems with $\mathcal{M}$-tensors, *J. Sci. Comput.* 68 (2016) 689–715.

[13] W. Ding and Y. Wei, *Theory and computation of tensors-multi-dimensional arrays*, Ph.D. thesis, Fudan University, 2016.

[14] A. Edelman and H. Murakami, Polynomial roots from companion matrix eigenvalues, *Math. Comp.* 64 (1995) 763–776.

[15] L.C. Evans, *Partial Differential Equations, 2th edn.*, The Johns Hopkins University Press, Berkeley, 2013.

[16] G.H. Golub and C.F. Loan, *Matrix Computations, 4th edn.*, The Johns Hopkins University Press, Baltimore, 2013.

[17] L.X. Han, A homotopy method for solving multilinear systems with $\mathcal{M}$-tensors, *Appl. Math. Lett.* 69 (2017) 49–54.

[18] A. Hadjidimos, On equivalence of optimal relaxed block iterative methods for the singular nonsymmetric saddle point problem, *Linear Algebra Appl.* 522 (2017) 175-202.

[19] R.A. Horn and C.R. Johnson, *Matrix Analysis, 2nd edition*, CAmbrisge University Press, Cambridge, 2013.

[20] J. He and T.Z. Huang, Inequalities for $\mathcal{M}$-tensors, *J. Inequal. Appl.* 2014 (2014) 114–122.

[21] Z.H. Huang and L. Qi, Formulating an n-person noncooperative game as a tensor complementarity problemm, *Comput. Optim. Appl.* 66 (2017) 557–576.

[22] B.N. Khoromskij, Tensor numerical methods for multidimensional pdes: theoretical analysis and initial applications, *ESAIM Proc. Surveys* 81 (2015) 22–47.

[23] T.G. Kolda and B.W. Bader, The TOPHITS model for higher-order web link analysis, *Workshop on Link Analysis Counterterrorism and Security* (2006) 26–29.

[24] T.G. Kolda and B.W. Bader, Tensor decompositions and applications, *SIAM Rev.* 51 (2009) 455–500.

[25] C.Q. Li, Z. Chen and Y.T. Li, A new eigenvalue inclusion set for tensors and its application, *Linear Algebra Appl.* 481 (2015) 36–53.

[26] C.Q. Li and Y.T. Li, An eigenvalue localization set for tensors with application to determine the positive (semi-)definiteness of tensors, *Linear Multilinear Algebra* 64 (2015) 587–601.

[27] C.Q. Li, Y.T. Li and X. Kong, New eigenvalue inclusion set for tensors, *Numer. Linear Algebra Appl.* 21 (2014) 39–50.

[28] D.H. Li, S.L. Xie and H.R. Xu, Splitting methods for tensor equations, *Numer. Linear Algebra Appl.* 24 (2017), `https://doi.org/10.1002/nla.2102`.

[29] X.T. Li and M.K. NG, Solving sparse non-negative tensor equations: algorithms and applications, *Front Math China* 10 (2015) 649–680.

[30] N. Liu, B.Y. Zhang, J. Yan, Z. Chen, W.Y. Liu, F.S. Bai and L.F. Chien, Text representation: from vector to tensor, in *Proceedings of the IEEE Internationnal Confernece on Data Mining*, IEEE, Texas, 2005, pp. 725-728.

[31] C.Q. Lv and C.F. Ma, A Levenberg-Marquardt method for solving semi-symmetric tensor equations, *J. Comput. Appl. Math.* 332 (2018) 13–25.

[32] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[33] L. Qi, Eigenvalues of a real supersymmetric tensor. *J. Symbolic Comput.* 40 (2005) 1302–1324.

[34] L. Qi, H.B. Chen and Y.N. Chen, *Tensor Eigenvalues and Their Applications*, Springer Nature Singapore Pte Ltd, SingaPore, 2018.

[35] L. Qi and Z.Y. Luo, *Tensor Analysis (Spectral Theory and Special Tensors)*, Society for Industial and Applied Mathematics (SIAM), Philadephia, 2017.

[36] L. Qi, Y.J. Wang and E.X. Wu, **D**-eigenvalues of diffusion kurtosis tensors, *J. Comput. Appl. Math.* 221 (2008) 150–157.

[37] W.C. Rheinboldt, *Methods for Solving Systems of Nonlinear Equations, vol. 70, 2nd edn.*, Society for Industrial and Applied Mathematics (SIAM), Philadephia, 1998.

[38] R.S. Varge, *Matrix Iterative Analysis, 2th edn.*, Springer Heidelberg Dordrecht London New York, Berlin, 2000.

[39] X.Z. Wang, M.L. Che and Y. Wei, Existence and uniqueness of positive solution for $\mathcal{H}^+$-tensor equations, *Appl. Math. Letters* 98 (2019) 191–198.

[40] Z.J. Xie, X.Q. Jin and Y. Wei, Tensor methods for solving symmetric $\mathcal{M}$-tensor systems, *J. Sci. Comput.* 74 (2018) 412–425.

[41] Y.N. Yang, Q.Z. Yang, Further results for perron-frobenius theorem for nonnegative tensors, *SIAM J.Matrix Anal. Appl.* 31 (2010) 2517–2530.

[42] L.P. Zhang, L. Qi and G.L. Zhou, $\mathcal{M}$-tensors and some applications, *SIAM J. Matrix Anal. Appl.* 35 (2014) 437–452.

Manuscript received 26 May 2021
revised 25 July 2021
accepted for publication 19 October 2021

Ming-Yu Deng
School of International Trade and Economics
University of International Business and Economics
Beijing 100029, P.R. China
E-mail address: my_deng@stu.ecnu.edu.cn

Weiyang Ding
Institute of Science and Technology for Brain-Inspired Intelligence
Fudan University, Shanghai, China;
Shanghai Center for Brain Science and Brain-Inspired Technology
Shanghai, China;
Key Laboratory of Computational Neuroscience and
Brain-Inspired Intelligence (Fudan University)
Ministry of Education, China;
MOE Frontiers Center for Brain Science, Fudan University
Shanghai, China;
Zhangjiang Fudan International Innovation Center
Shanghai 200433, P.R. China
E-mail address: dingwy@fudan.edu.cn

Xue-Ping Guo
School of Mathematical Sciences, East China Normal University
Shanghai 200241, P.R. China
E-mail address: xpguo@math.ecnu.edu.cn