A practical method for computing the largest M-eigenvalue of a fourth-order partially symmetric tensor

Yiju Wang\textsuperscript{1} \textsuperscript{*}† Liqun Qi\textsuperscript{2}‡ Xinzhen Zhang\textsuperscript{2}§

\textsuperscript{1}School of Operations Research and Management Sciences, Qufu Normal University, Rizhao Shandong, China, 276800
\textsuperscript{2}Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong.

SUMMARY

In this paper, we consider a quartic homogenous polynomial optimization problem over two unit spheres arising in nonlinear elastic material analysis and in entanglement studies in quantum physics. The problem is equivalent to computing the largest M-eigenvalue of a fourth-order tensor. To solve the problem, we propose a practical method whose validity is guaranteed theoretically. To make the sequence generated by the method converge to a good solution of the problem, we also develop an initialization scheme. The given numerical experiments show the effectiveness of the proposed method.

Copyright © 2007 John Wiley & Sons, Ltd.

key words: M-eigenvalue; power method; initialization scheme

1. INTRODUCTION

Consider the following optimization problem:

\[
\max f(x, y) = \sum_{i,k=1}^{m} \sum_{j,l=1}^{n} A_{ijkl} x_i y_j x_k y_l \\
\text{s.t. } \quad x^\top x = 1, \quad y^\top y = 1, \quad x \in \mathbb{R}^m, \quad y \in \mathbb{R}^n, \tag{1}
\]

where the coefficients $A_{ijkl}$ satisfy the following symmetric property:

\[
A_{ijkl} = A_{kjil} = A_{ilkj} = A_{klij}, \quad \text{for } i, k = 1, 2, \ldots, m; \quad j, l = 1, 2, \ldots, n. \tag{2}
\]
Certainly, the coefficients $A_{ijkl}$ form a fourth-order partially symmetric tensor, and we denote it by $A$.

This problem arises from the nonlinear elastic materials analysis and entanglement studies in quantum physics.

In the nonlinear elastic materials analysis, one approach is to consider an elastic material in terms of a fourth-order three-dimensional elastic module tensor which satisfies the symmetric property stated above [1]. It is well known that both the strong ellipticity and ordinary ellipticity play an important role in nonlinear elastic material analysis, especially when a material is required to satisfy a number of important statical and dynamical properties [2, 3]. Based on tensor expression of elastic material, the strong ellipticity condition can be mathematically characterized by the positiveness of the following function over two unit spheres in $R^3$:

$$f(x, y) = \sum_{i,j,k,l=1}^{3} A_{ijkl} x_i y_j x_k y_l$$

and the ordinary ellipticity condition is characterized by the nonnegativeness of this function. That is, the strong ellipticity condition holds if and only if the optimal value of the following optimization problem is positive

$$\min \ f(x, y) = \sum_{i,k=1}^{3} \sum_{j,l=1}^{3} A_{ijkl} x_i y_j x_k y_l$$

s.t. $x^\top x = 1, \ y^\top y = 1$ $x, y \in R^3$ (3)

and the ordinary ellipticity condition holds if and only if the global optimal value is nonnegative. Certainly, this problem can readily be transformed into problem (1) with $m = n = 3$.

In the study of entanglement in quantum physics, determining whether a quantum state is separable or inseparable (entangled) is a fundamental problem [4, 5, 6], which is an NP-hard problem [5]. To solve the entanglement problem, Dahl et al. [7] applied the Frank-Wolfe minimizing method [8], in which problem (1) is involved as a subproblem.

To solve problem (1), we first consider its optimal condition following the tensor product notation used in [9, 10, 11]. Denote $A \cdot yxy$ as a vector with its $i$th component being $\sum_{j,l=1}^{n} A_{ijkl} y_j x_k y_l$, and $A_{xy}x$ as a vector with its $l$th component being $\sum_{i,k=1}^{m} A_{ijkl} x_i y_j x_k$. For any minimizer $(x, y)$ of (1), by the optimality theory [8, 12], there exist $\lambda, \mu \in R$ such that

$$\begin{align*}
A \cdot yxy &= \lambda x \\
A_{xy}x &= \mu y \\
x^\top x &= 1 \\
y^\top y &= 1
\end{align*}$$

Certainly, the optimal condition can further be simplified as

$$\begin{align*}
A \cdot yxy &= \lambda x \\
A_{xy}x &= \mu y \\
x^\top x &= 1 \\
y^\top y &= 1
\end{align*}$$

(4)
If $\lambda$, $x$, and $y$ are real solutions of (4), $\lambda$ is said to be an M-eigenvalue of tensor $A$, and $x$ and $y$ are said to be a left M-eigenvector and a right M-eigenvector of tensor $A$, associated with the M-eigenvalue $\lambda$, respectively \[13\]. Here, the letter $M$ is borrowed from the word material.

By the discussion above, we know that problem (1) is equivalently transformed into computing the largest M-eigenvalue of tensor $A$, i.e., solving (4). For this system, it seems not difficult to solve. However, this is not the case due to that neither equation of system (4) is linear. Recently, this problem is also shown to be NP-hard \[14\]. As this problem is a subproblem of the entanglement problem, in practice, we may aim to find a “good” solution of the problem. In the next section, we will propose a practical method to compute the largest M-eigenvalue of tensor $A$ based on the power method for computing the largest eigenvalue in magnitude of a matrix \[15\]. Compared with the alternating eigenvalue maximization method for solving (1) proposed in \[7\], the computation cost of our method is less. As for the validity of this method, it is guaranteed theoretically for the convex case. To make the conclusion hold generally, we introduce a translation technique into the method. Furthermore, to make the generated sequence converge to a good solution of the problem, we also develop an initialization scheme in Section 3. The given numerical experiments in Section 4 show that the proposed method could generate a well approximated point to the global maximizer of our concerned problem.

2. PRACTICAL POWER METHOD AND ITS CONVERGENCE

It is well known that the power method is an efficient method for computing the largest eigenvalue in magnitude of a matrix \[15\]. This method was successfully extended to compute the best rank-1 approximations to higher-order tensors \[16, 17\], i.e., the largest Z-eigenvalue in magnitude of higher-order tensors \[9\]. Motivated by this, we propose a modified power method to compute the largest M-eigenvalue of a fourth-order partially symmetric tensor.

Algorithm 2.1

Initialization Step: Take initial points $x_0 \in R^m, y_0 \in R^n$, and let $k = 0$;

Iterative Step: Execute the following procedures alternatively until convergence:

\[
\bar{x}_{k+1} = A \cdot y_k x_k y_k, \quad x_{k+1} = \frac{\bar{x}_{k+1}}{\|\bar{x}_{k+1}\|};
\]

\[
\bar{y}_{k+1} = A x_{k+1} y_k x_{k+1}, \quad y_{k+1} = \frac{\bar{y}_{k+1}}{\|\bar{y}_{k+1}\|};
\]

\[k = k + 1.\]

Now, we give a theoretical analysis to the method. For the objective function $f(x, y)$, from (1), we know that it is a bi-quadratic function with respect to $x, y$, respectively. That is, the function can be written as

\[
f(x, y) = Axyxy = x^\top B(y)x = y^\top C(x)y,
\]

where $B(y)$ and $C(x)$ are respectively symmetric matrices in $R^{m \times m}$ and $R^{n \times n}$ with entries

\[
B_{ik}(y) = \sum_{j,l=1}^n A_{ijkl}y_j y_l, \quad C_{jl}(x) = \sum_{i,k=1}^m A_{ijkl}x_i x_k.
\]
Based on this analysis, we have the following conclusion for Algorithm 2.1.

**Theorem 2.1.** Suppose that for any \( x \in \mathbb{R}^m, y \in \mathbb{R}^n \), the matrices \( B(y) \) and \( C(x) \) are both positive definite. Then the generated sequence \( \{ f(x_k, y_k) \} \) by Algorithm 2.1 is nondecreasing.

**Proof**
From the assumption, we know that for any fixed \( x \in \mathbb{R}^m \), the function \( f(x, y) \) is strictly convex with respect to \( y \in \mathbb{R}^n \), and similarly, the function \( f(x, y) \) is strictly convex with respect to \( x \in \mathbb{R}^m \) for any fixed \( y \in \mathbb{R}^n \). Thus, for any \( k \geq 0 \),

\[
f(x_{k+1}, y_k) - f(x_k, y_k) \geq \langle x_{k+1} - x_k, \nabla_x f(x_k, y_k) \rangle.
\]

Consider the inner product term in the right-hand side. Since \( \nabla_x f(x_k, y_k) = 2A \cdot y_k x_k y_k \), from Algorithm 2.1, one has

\[
x_{k+1} = \frac{\nabla_x f(x_k, y_k)}{\| \nabla_x f(x_k, y_k) \|}.
\]

Recalling the Cauchy-Schwartz inequality and the choice of \( x_{k+1} \), we conclude from (5) that

\[
f(x_{k+1}, y_k) \geq f(x_k, y_k) \quad \text{and it holds strictly if } x_{k+1} \neq x_k.
\]

Similarly, we can conclude that

\[
f(x_{k+1}, y_{k+1}) \geq f(x_{k+1}, y_k) \quad \text{and it holds strictly if } y_{k+1} \neq y_k.
\]

Combining these two cases, we obtain the desired result. \( \square \)

Before giving an analysis to Theorem 2.1, we first give an explanation of the condition of Theorem 2.1 by introducing the following definition.

**Definition 2.1.** A fourth-order partially symmetric tensor \( A \) is said to be positive definite on \( \mathbb{R}^m \times \mathbb{R}^n \) if for any nonzero vectors \( x \in \mathbb{R}^m \) and \( y \in \mathbb{R}^n \), it holds that

\[
A_{ijkl} x_i y_j x_k y_l > 0.
\]

From the definition, we know that the matrices \( B(y) \) and \( C(x) \) are both positive definite for any nonzero vectors \( x \in \mathbb{R}^m \) and \( y \in \mathbb{R}^n \), it holds that

\[
A_{ijkl} x_i y_j x_k y_l > 0.
\]

From Theorem 2.1, we may conclude that the generated sequence \( \{ x_k, y_k \} \) converges to a stationary point of problem (1) in the “convex” case [8]. Now, one key problem is posed naturally: how about the algorithm for the general case? That is, for the case that either matrix \( B(y) \) or matrix \( C(x) \) is not positive definite. To solve this problem, we now make a translation to the corresponding tensor in the objective function by introducing the following “identity” tensor \( I \in \mathbb{R}^{m \times n \times m \times n} :\)

\[
I_{ijkl} = \begin{cases} 
1, & \text{if } i = k \text{ and } j = l \\
0, & \text{otherwise}
\end{cases}
\]

Take \( \tau \in \mathbb{R} \) such that \( \tau > \max \{ |\lambda| \mid \lambda \text{ is an M-eigenvalue of tensor } A \} \) and set

\[
f(x, y) = \tau I_{ijkl} + A_{ijkl},
\]

\[
\hat{f}(x, y) = \tau I_{ijkl} + A_{ijkl}.
\]
It is easy to see that tensor $\mathcal{A}$ is positive definite on $R^m \times R^n$ and has the same symmetry property as tensor $\mathcal{A}$. Furthermore, if $x$ and $y$ constitute a pair of M-eigenvectors of tensor $\mathcal{A}$ associated with M-eigenvalue $\lambda$, then they are also a pair of M-eigenvectors of tensor $\mathcal{A}$ associated with M-eigenvalue $(\lambda - \tau)$. Since function $f(x, y)$ satisfies the assumptions in Theorem 2.1, we can apply Algorithm 2.1 to compute the largest M-eigenvalue of tensor $\mathcal{A}$ and hence we can obtain the largest M-eigenvalue of tensor $\mathcal{A}$.

Now, one more question rises accordingly: How to choose a suitable $\tau$? In fact, this can be solved based on the estimation of the largest eigenvalue of the unfolded matrix of tensor $\mathcal{A}$ defined below.

Define the following index mapping from four indices $i, j, k, l$ to two indices $s, t$:

$$s = n(i - 1) + j, \quad t = n(k - 1) + l.$$ 

Using this mapping, we may unfold tensor $\mathcal{A}$ into a matrix $A \in R^{mn \times mn}$. From the partial symmetry of tensor $\mathcal{A}$, we know that the unfolded matrix $A$ is symmetric. Based on this representation, the objective function $f(x, y)$ can be written as a quadratic form:

$$f(x, y) = (x \otimes y)^\top A(x \otimes y),$$

where $x \otimes y$ denotes the Kronecker product of vectors $x$ and $y$ which is a vector in $R^{mn}$. Based on this, we can immediately obtain the following conclusion.

**Proposition 2.1.** Suppose matrix $A$ is the unfolded matrix of tensor $\mathcal{A}$. Then tensor $\mathcal{A}$ is positive definite on $R^m \times R^n$ provided that matrix $A$ is positive definite on $R^{mn}$. Furthermore, all the M-eigenvalues of tensor $\mathcal{A}$ lie in the interval composed by the smallest eigenvalue and the largest eigenvalue of matrix $A$.

From the Geršgorin Disc Theorem [15], we know that the magnitude of any eigenvalue of matrix $A$ must be less than or equal to

$$\max_{1 \leq i \leq mn} \sum_{j=1}^{mn} |A_{ij}|.$$ 

This can easily be computed. In fact, since matrix $A$ is symmetric, $\tau = \sum_{1 \leq i \leq j \leq mn} |A_{ij}|$ is sufficient to guarantee that $(\tau I + A)$ satisfies the assumption in Theorem 2.1.

To end this section, we give an example to show that a fourth-order partially symmetric tensor $\mathcal{A}$ may be positive definite on $R^m \times R^n$ but the corresponding unfolded matrix $A$ is not positive definite on $R^{mn}$.

**Example 2.1.** Consider the following fourth-order two-dimensional tensor $\mathcal{A}$ with entries

$$\begin{align*}
A_{1111} &= 12, & A_{1112} &= 1, & A_{1121} &= 1, & A_{1122} &= 5, & A_{1212} &= 2, \\
A_{1222} &= 1, & A_{2121} &= 2, & A_{2122} &= 1, & A_{2222} &= 12.
\end{align*}$$

For this symmetric tensor, the unfolded matrix is:

$$A = \begin{pmatrix}
12 & 1 & 1 & 5 \\
1 & 2 & 5 & 1 \\
1 & 5 & 2 & 1 \\
5 & 1 & 1 & 12
\end{pmatrix}.$$
It is easy to verify that for \( x, y \in \mathbb{R}^2 \),
\[
f(x, y) = A_{xy} = \sum_{i,j,k,l=1}^{2} A_{ijkl} x_i y_j x_k y_l = (x_1 y_1 + x_1 y_2)^2 + (x_1 y_2 + x_2 y_2)^2 + (x_2 y_1 + x_1 y_1)^2 + (x_2 y_1 + x_2 y_2)^2 + 10(x_1 y_1 + x_2 y_2)^2,
\]
and it is positive for any nonzero vectors \( x, y \in \mathbb{R}^2 \). However, for \( w = (0, 1, -1, 0)^\top \), \( w^\top A w = -6 < 0 \), i.e., the unfolded matrix \( A \) is not positive definite on \( \mathbb{R}^4 \).

3. INITIALIZATION TECHNIQUE

From the discussion in the last section, we know that the given power method can generate a stationary point of problem (1) generally. However, whether an accumulation point of the generated sequence is a “good” maximizer of problem (1) or not strongly depends on the initialization. In this sense, we claim that Algorithm 2.1 only partially solves problem (1).

To obtain a good maximizer of problem (1), we will introduce an initialization technique into Algorithm 2.1 inspired by the initialization strategy for computing the best rank-1 supersymmetric approximation to a supersymmetric tensor in [18]. This initialization technique is based on the basic fact that the unit eigenvector corresponding to the largest eigenvalue of positive definite and symmetric matrix \( G \) is a maximizer of the function \( g(x) = x^\top G x \) over the unit sphere.

Suppose the unfolded matrix \( A \) of tensor \( A \) is positive definite, \( w \in \mathbb{R}^{mn} \) is an unit eigenvector of matrix \( A \) associated with the largest eigenvalue \( \mu \), and \( (x^*, y^*) \) is a solution of problem (1). Then \( f(x^*, y^*) \leq \mu \) and the equality holds only when \( x^* \otimes y^* \) coincide with \( \pm w \). Motivated by this, we may take the initial point \( (x_0, y_0) \) in Algorithm 2.1 that maximizes the inner product \( \langle x \otimes y, w \rangle \) over unit spheres.

To solve the subproblem, we need to fold vector \( w \) into a matrix in \( \mathbb{R}^{m \times n} \) in the following way: for \( k = 1, 2, \cdots, mn \), set \( i = \lceil \frac{k}{n} \rceil \), \( j = (k - 1) \bmod n + 1 \) and
\[
W_{ij} = w_k.
\]

Then the inner product \( \langle x \otimes y, w \rangle \) can be expressed as a bi-linear function \( x^\top W y \) and the involved subproblem is as follows
\[
\begin{align*}
\max & \quad x^\top W y \\
\text{s.t.} & \quad x^\top x = 1, \ y^\top y = 1
\end{align*}
\]
which is also equivalent to the following minimization problem [16]:
\[
\begin{align*}
\min & \quad \| W - \mu x y^\top \|_F^2 \\
\text{s.t.} & \quad \mu \in \mathbb{R} \\
& \quad x^\top x = 1, \ y^\top y = 1
\end{align*}
\]
(6)
where \( \| \cdot \|_F \) denotes the Frobenius norm of a matrix. This optimization problem can easily be solved via singular eigenvalue decomposition of matrix \( W \) [15]. That is, if matrix \( W \) has the following singular eigenvalue decomposition
\[
W = U^\top \Sigma V = \sum_{i=1}^{r} \sigma_i u_i v_i^\top
\]
where \( \sigma_1 \geq \sigma \geq \cdots \sigma_r > 0 \) and \( r \) is the rank of the matrix, then \( u_1 \) and \( v_1 \) constitute the solution of problem (6). Thus, the initialization subproblem can be solved by letting \( x_0 = u_1, y_0 = v_1 \).

4. NUMERICAL EXPERIMENTS AND SIMULATIONS

Combining discussions in Sections 2 and 3, we can give a complete algorithm to compute a good approximation of the largest \( M \)-eigenvalue of a fourth-order partially symmetric tensor.

Algorithm 4.1

**Initial Step:** Input \( A \) and unfold it to obtain matrix \( A \).

**Substep 1:** Take \( \tau = \sum_{1 \leq i \leq j \leq mn} |A_{ij}| \), set \( \bar{A} = \tau I + A \) and unfold \( \bar{A} \) to matrix \( \bar{A} \).

**Substep 2:** Compute the eigenvector \( w \) of matrix \( \bar{A} \) associated with the largest eigenvalue and fold it into the matrix \( W \).

**Substep 3:** Compute the singular vectors \( u_1 \) and \( v_1 \) corresponding to the largest singular value of the matrix \( W \).

**Substep 4:** Take \( x_0 = u_1, y_0 = v_1 \), and let \( k = 0 \).

**Iterative Step:** Execute the following procedures alternatively until certain convergence criterion is satisfied and output \( x^*, y^* \):

\[
\begin{align*}
\bar{x}_{k+1} &= \bar{A} \cdot y_k x_k y_k, \quad x_{k+1} = \frac{\bar{x}_{k+1}}{\|\bar{x}_{k+1}\|}; \\
\bar{y}_{k+1} &= \bar{A} x_{k+1} y_k x_{k+1}, \quad y_{k+1} = \frac{\bar{y}_{k+1}}{\|\bar{y}_{k+1}\|}; \\
k &= k + 1.
\end{align*}
\]

**Final Step:** Output the largest \( M \)-eigenvalue of tensor \( A \): \( \lambda = f(x^*, y^*) - \tau \), and the associated \( M \)-eigenvectors: \( x^*, y^* \).

Certainly, the algorithm contains two parts: the initial step and the iterative step. In fact, the initial step, i.e., computing the largest eigenvalue and the corresponding eigenvector of a matrix, is also an iterative scheme [15]. For Algorithm 4.1, the computation complexity at each iterative step is of order \( O(m^2 n + mn^2) \). Thus, if the largest \( M \)-eigenvalue of tensor \( A \) can be generated within few steps, this algorithm can be said to be practical. To check this, we first make the numerical experiments of Algorithm 4.1 on two forth-order three-dimensional partially symmetric tensors, and then compute the global optimal values of the objective functions by the projected gradient method [19] combined with the uniform grid method in high order accuracy.

**Example 4.1.** Consider the tensor whose entries are uniformly generated in \((-1,1)\):

\[
A(:,:,1,1) = \begin{pmatrix}
-0.9727 & 0.3169 & -0.3437 \\
-0.6332 & -0.7866 & 0.4257 \\
-0.3350 & -0.9896 & -0.4323
\end{pmatrix}
\]
For this tensor, its largest M-eigenvalue 2.3227 which is marked in Figure 1 by the horizontal line.

**Example 4.2.** Consider the tensor whose entries are uniformly generated in $(0,5)$:

$$A(\cdot,1,1) = \begin{pmatrix}
1.9832 & 1.0023 & 4.2525 \\
2.6721 & 3.2123 & 2.8761 \\
4.6384 & 2.9484 & 4.0319 \\
\end{pmatrix}$$

$$A(\cdot,2,1) = \begin{pmatrix}
2.6721 & 3.2123 & 2.8761 \\
3.0871 & 0.1393 & 4.4704 \\
1.7450 & 3.0394 & 4.6836 \\
\end{pmatrix}$$

$$A(\cdot,3,1) = \begin{pmatrix}
4.6384 & 2.9484 & 4.0319 \\
1.7450 & 3.0394 & 4.6836 \\
0.3741 & 1.6947 & 2.7677 \\
\end{pmatrix}$$
For this tensor, its largest M-eigenvalue is 26.1187 which is marked in Figure 2 by the horizontal line.

From Figures 1 and 2, we can see that the largest M-eigenvalue can be highly approximated within few steps especially for the second example. In fact, we have done many numerical experiments of Algorithm 4.1 on tensors whose entries are uniformly generated in (0, L) for some positive number L, and the numerical results show that this algorithm has a particularly good performance for this kind of tensor.

We also do two numerical experiments on tensors with high dimensions. Here, we would not compute the largest M-eigenvalue of the tensors using the globalization method since its computation cost is extremely high.

Figure 3 shows the numerical result of Algorithm 4.1 on a tensor whose entries are uniformly generated in (-5,5) with $m = 12$, $n = 18$, and Figure 4 shows the numerical result on a tensor whose entries are uniformly generated in (-5,5) with $m = 30$, $n = 18$.

From the given numerical experiments, we can claim that the numerical result of Algorithm 4.1 is good although the iterative procedure is at most linearly convergent in theory [15]. Since the computing cost at the iterative step of the algorithm is very small, the designed algorithm is really efficient in practice especially for the large scale problem.

Acknowledgement: The authors are in debt to two anonymous referees for numerous insightful comments and suggestions, which have greatly improved the paper.
Figure 1. Numerical Result of Example 4.1

Figure 2. Numerical Result of Example 4.2
Figure 3. Numerical Result for a Tensor with $m = 12, n = 18$

Figure 4. Numerical Result for a Tensor with $m = 30, n = 18$
14. Ling C., Nie J., Qi L. and Ye Y. Bi-quadratic optimization over unit spheres and semidefinite programming relaxations, preprint, Department of Applied Mathematics, The Hong Kong Polytechnic University, 2008.