Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 11, No. 2, pp. 279–290, April 2015

QUASI-NEWTON METHOD FOR COMPUTING Z-EIGENPAIRS OF A SYMMETRIC TENSOR*

Meilan Zeng and Qin Ni†

Abstract: In this paper, we propose a quasi-Newton method for computing Z-eigenpairs of a symmetric tensor. The iterative sequence generated by the quasi-Newton method is norm descent for the function corresponding to the eigenvalue equations. On the basis of the special structure of the system of eigenvalue equations, we can obtain a descent direction by solving a system of linear equations at each iteration. The global and superlinear convergence of the proposed method is established. The numerical results show the promising performance of the method.

Key words: *quasi-Newton method, Z-eigenpair, symmetric tensor*

Mathematics Subject Classification: *15A18, 65H10, 90C53*

1 Introduction

Let $\mathbb{R}$ be the real field, and $\mathcal{A} = (a_{i_1\cdots i_m})$ be a real m-order n-dimensional tensor. Denote the set of all real m-order n-dimensional tensors by $\mathbb{R}^{[m,n]}$. A tensor is called symmetric if the value of $a_{i_1\cdots i_m}$ is invariant under any permutation of its index $\{i_1, \dots, i_m\}$, and the Frobinious norm of $\mathcal{A}$ is defined by

$$\|\mathcal{A}\|_F = \sqrt{\sum_{i_1,\cdots,i_m=1}^{n} a^2_{i_1\cdots i_m}}. \tag{1.1}$$

In all cases, $\|\cdot\|$ denotes 2-norm. In the paper, all the tensors represent real tensors.

A symmetric tensor uniquely defines a real mth degree homogeneous polynomial function f and

$$f = \mathcal{A}x^m := \sum_{i_1,\dots,i_m=1}^{n} a_{i_1\cdots i_m} x_{i_1} \cdots x_{i_m},$$

where $x \in \mathbb{R}^n$, and $\mathcal{A}x^{m-1}$ is a vector in $\mathbb{R}^n$, with its ith component defined by

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

*This work was supported by the Natural Science Foundation of Jiangsu Province (BK20141409), and National Natural Science Foundation of China (11471159).

†Corresponding author

Definition 1.1. Let $\mathcal{A} \in \mathbb{R}^{[m,n]}$. The pair $(\lambda, x) \in \mathbb{R} \times \mathbb{R}^n$ is called Z-eigenpair, and λ is called Z-eigenvalue and x is the corresponding Z-eigenvector of $\mathcal{A}$ if they satisfy the equations

$$\mathcal{A}x^{m-1} = \lambda x \quad \text{and} \quad x^T x = 1. \tag{1.2}$$

It was proved in Qi [22] that Z-eigenvalues always exist for a symmetric tensor. The properties of Z-eigenvalues were studied in [20–22]. In [16, 20, 22] , the number of Z-eigenvalues of an m-order n-dimensional symmetric tensor was discussed. Z-eigenvalues of a given tensor have applications in numerical multilinear algebra [17], image processing [23, 25], higher order Markov chains [15], and spectral hypergraph theory [7], etc. In a series of their recent work [13–15], Ng et al. discovered interesting new connections of the Z-eigenvalue problem to the transition probability tensors of higher order Markov chains. They proposed a framework (HAR) that can be used in multi-relational data mining. Some other related works in this new front have been conducted in [2] and [8]. It is NP-hard to compute eigenvalues of higher order tensors (i.e. $m \geq 3$). Recently, Qi et al. [24] proposed a direct method for finding all the Z-eigenvalues in the case of order three and dimension three, by reducing a symmetric tensor to a pseudo-canonical form. Kolda and Mayo [10] provided a shifted power method (SS-HOPM) for computing a Z-eigenvalue and its associated eigenvector for a symmetric tensor. SS-HOPM is guaranteed to converge by a suitably modified function

$$\hat{f}(x) \equiv f(x) + \alpha(x^T x)^{m/2},$$

which is convex or concave on $\mathbb{R}^n$, however, only several Z-eigenpairs are generally found for a fixed α. More recently, Hu et al. [6] proposed a sequential semidefinite programming method for finding the extreme Z-eigenvalues of tensors, which can apply to even order tenors. Cui et al. [4] use Jacobian SDP relaxations in ploynomial optimization to compute all real eigenvalues of symmetric tensors sequentially. The goal of this paper is to compute Z-eigenpairs of symmetric tensors by another way.

The equations (1.2) are seen to be a system of nonlinear equations with respect to the variables (x, λ). Moreover, it is easy to prove that the Jacobian of the system is symmetric for a symmetric tensor. Gu et al. [5] proposed a norm descent quasi-Newton method for symmetric nonlinear equations. We mainly follow the quasi-Newton method in [5] to compute Z-eigenpairs of symmetric tensors. In this paper, we analyze the special structure of the system of nonlinear equations generated by (1.2), and determine an upper bound of a parameter where the decent direction can be obtained by solving a system of linear equations with a parameter less than that bound. Hence, we obtain a descent direction by solving only a system of linear equation each iteration. We prove that the proposed method for computing Z-eigenpairs of a symmetric tensor converges globally and superlinearly.

The organization of this paper is as follows. Section 2 proposes quasi-Newton method for computing Z-eigenpairs of symmetric tensors. The convergence theory is presented in Section 3. Section 4 demonstrates preliminary numerical results on test problems.

2 Quasi-Newton Method

In this section, we give an algorithm for computing Z-eigenpairs of a symmetric tensor.

Let $\mathcal{A} \in \mathbb{R}^{[m,n]}$,

$$F(x, \lambda) = \begin{pmatrix} \mathcal{A}x^{m-1} - \lambda x \\ \frac{1}{2}(1 - x^T x) \end{pmatrix} = 0. \tag{2.1}$$

Then $F : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ is a nonlinear function, and is continuously differentiable. The tensor eigenvalue equations (1.2) for $m > 2$ amount to a system of nonlinear equations

(2.1). Any solution of (2.1) is a Z-eigenpair of the symmetric tensor $\mathcal{A}$. The following result is from [18].

Lemma 2.1. *If $\mathcal{A} \in \mathbb{R}^{[m,n]}$ is a symmetric tensor, then it follows*

$$F'(x,\lambda) = \begin{pmatrix} (m-1)\mathcal{A}x^{m-2} - \lambda I_n & -x \\ -x^T & 0 \end{pmatrix}, \tag{2.2}$$

where

$$(\mathcal{A}x^{m-2})_{ij} = \sum_{i_2,\cdots,i_{m-1}=1}^{n} a_{iji_2\cdots i_{m-1}} x_{i_2}\cdots x_{i_{m-1}},\ i,j=1,\cdots,n.$$

Hence, $F'(x,\lambda)$ is a symmetric matrix.

Since the Jacobian $F'(x,\lambda)$ is symmetric, then (2.1) is the symmetric nonlinear problem. The symmetric nonlinear equations has been studied by several authors. Li and Fukushima [11] proposed a globally and superlinearly convergent Gauss-Newton-based BFGS method for such problem. Gu et al. [5] extended this method to the norm descent case.

Let $\theta(\omega) = \frac{1}{2}\|F(\omega)\|^2$, where $\omega = (x,\lambda)$. Then the nonlinear equation problem (2.1) is equivalent to the following global optimization problem

$$\min\ \theta(\omega), \omega \in \mathbb{R}^{n+1}. \tag{2.3}$$

In the norm descent quasi-Newton method for solving (2.1), the subproblem to be solved on each iteration is the following linear equations:

$$B_k d + q_k(t) = 0, \tag{2.4}$$

where B_k is an approximation of matrix $F'(\omega)^2$, ω_k is the current iterate, and t is a parameter,

$$q_k(t) = (F(\omega_k + tF(\omega_k)) - F(\omega_k))/t. \tag{2.5}$$

For general symmetric nonlinear equations, it is proved in [5] that there exists a constant $\bar{t} > 0$ depending on k such that when $t \in (0,\bar{t})$, the solution $d(t)$ of (2.4) is a descent direction of $\theta(\omega)$ at ω_k. However, $\bar{t}$ is not explicitly obtained, and a descent direction is found by multiple tests.

Now we use this norm descent quasi-Newton method to solve (2.1). For (2.1), we can determine this bound $\bar{t}$, and obtain the following theorem.

Theorem 2.2. *Let $T_b \geq 1$ be a constant,*

$$F(\omega_k) = (\bar{F}_k, f_{n+1}^k)^T, \tag{2.6}$$

where f_{n+1}^k is the $(n+1)$-th component of $F(\omega_k) \in \mathbb{R}^{n+1}$, and B_k be a symmetric and positive definite matrix. If ω_k is not a stationary point of (2.3), then there is $\tilde{T}_k > 0$ such that when $t \in (0,\tilde{T}_k]$, the unique solution $d(t)$ of (2.4) satisfies

$$\nabla\theta(\omega_k)^T d(t) < 0, \tag{2.7}$$

where $\eta \in (0,1)$,

$$\tilde{T}_k = \min\Big\{T_b, \frac{\eta\nabla\theta(\omega_k)^T B_k^{-1}\nabla\theta(\omega_k)}{\sqrt{n}\|\nabla\theta(\omega_k)\|\|B_k^{-1}\|_\infty(\|f_{n+1}^k\bar{F}_k\| + \frac{1}{2}\|\bar{F}_k\|^2 + \frac{1}{2}(m-1)(m-2)\|\mathcal{A}\|_F(\|x_k\| + T_b\|\bar{F}_k\|)^{m-3})}\Big\}. \tag{2.8}$$

Proof. Since ω_k is not a stationary point of (2.3), we have $\nabla\theta(\omega_k) \neq 0$. By direct computation, we have

$$F(\omega_k+tF(\omega_k))-F(\omega_k) = \begin{pmatrix} \mathcal{A}(x_k + t\bar{F}_k)^{m-1} - \mathcal{A}x_k^{m-1} - t(\lambda_k \bar{F}_k + f_{n+1}^k x_k) - t^2 f_{n+1}^k \bar{F}_k \\ -t x_k^T \bar{F}_k - \frac{1}{2}t^2\|\bar{F}_k\|^2 \end{pmatrix}. \tag{2.9}$$

From (2.2), (2.6) and the symmetry of $F'(\omega_k)$, it follows that

$$\begin{aligned} & \mathcal{A}(x_k + t\bar{F}_k)^{m-1} - \mathcal{A}x_k^{m-1} \\ = \; & (m-1)t\mathcal{A}x_k^{m-2}\bar{F}_k + \frac{1}{2}(m-1)(m-2)t^2\mathcal{A}(x_k + \xi t\bar{F}_k)^{m-3}\bar{F}_k^2, \end{aligned} \tag{2.10}$$

$$\nabla\theta(\omega_k) = F'(\omega_k)F(\omega_k) = \begin{pmatrix} (m-1)\mathcal{A}x_k^{m-2}\bar{F}_k - \lambda_k\bar{F}_k - f_{n+1}^k x_k \\ -x_k^T\bar{F}_k \end{pmatrix}, \tag{2.11}$$

where $\xi \in (0,1)$.

From (2.9)-(2.11), we get

$$\begin{aligned} q_k(t) &= \frac{F(\omega_k + tF(\omega_k)) - F(\omega_k)}{t} \\ &= \nabla\theta(\omega_k) + t\begin{pmatrix} -f_{n+1}^k\bar{F}_k + \frac{1}{2}(m-1)(m-2)\mathcal{A}(x_k + \xi t\bar{F}_k)^{m-3}\bar{F}_k^2 \\ -\frac{1}{2}\|\bar{F}_k\|^2 \end{pmatrix}. \end{aligned} \tag{2.12}$$

Since B_k is positive definite together with (2.4), (2.12) and $\|B_k^{-1}\| \le \sqrt{n}\|B_k^{-1}\|_\infty$, then we have

$$\begin{aligned} \nabla\theta(\omega_k)^T d(t) &= -\nabla\theta(\omega_k)^T B_k^{-1}\nabla\theta(\omega_k) \\ &\quad + t\nabla\theta(\omega_k)^T B_k^{-1}\begin{pmatrix} -f_{n+1}^k\bar{F}_k + \frac{1}{2}(m-1)(m-2)\mathcal{A}(x_k + \xi t\bar{F}_k)^{m-3}\bar{F}_k^2 \\ -\frac{1}{2}\|\bar{F}_k\|^2 \end{pmatrix} \\ &\le -\nabla\theta(\omega_k)^T B_k^{-1}\nabla\theta(\omega_k) + t\sqrt{n}\|\nabla\theta(\omega_k)\|\|B_k^{-1}\|_\infty \Big[\|f_{n+1}^k\bar{F}_k\| + \frac{1}{2}\|\bar{F}_k\|^2 \\ &\quad + \frac{1}{2}(m-1)(m-2)\|\mathcal{A}\|_F(\|x_k\| + T_b\|\bar{F}_k\|)^{m-3}\|\bar{F}_k\|^2\Big] \\ &\le -(1-\eta)\nabla\theta(\omega_k)^T B_k^{-1}\nabla\theta(\omega_k) < 0, \end{aligned} \tag{2.13}$$

for $T_b \ge 1, t \in (0, \tilde{T}_k]$, where $\|\mathcal{A}\|_F$ is defined in (1.1). □

Now we represent a norm descent quasi-Newton algorithm for solving (2.1) in the following.

Algorithm 2.1.

Step 0. Choose $\epsilon > 0, \omega_0 \in \mathbb{R}^{n+1}$. Let $B_0 \in \mathbb{R}^{(n+1)\times(n+1)}$ be symmetric and positive definite. Set $k := 0$.

Step 1. Evaluate F_k, $F_k = F(\omega_k)$, if $\|F_k\| \le \epsilon$, terminate.

Step 2. Let T_k be

$$T_k = \begin{cases} \tilde{T}_k, & \text{if } k = 0, \\ \min\{t_{k-1}, \tilde{T}_k\}, & \text{if } k \ge 1, \end{cases} \tag{2.14}$$

where $\tilde{T}_k$ is defined in (2.8), and find d_k by (2.4) where $t = T_k$. Let $t_k = \max\{1, \rho, \rho^2, \ldots\}$ satisfy

$$\theta(\omega_k + t_k d_k) - \theta(\omega_k) \le -\sigma_1\|t_k d_k\|^2 - \sigma_2\|t_k F_k\|^2, \tag{2.15}$$

where σ_1 and σ_2 are positive constants. Set $\omega_{k+1} = \omega_k + t_k d_k$.

Step 3. Let $s_k = \omega_{k+1} - \omega_k = t_k d_k, \delta_k = F_{k+1} - F_k, y_k = F(\omega_k + \delta_k) - F(\omega_k)$. If $y_k^T s_k \leq 0$, then $B_{k+1} = B_k$ and go to Step 4. Otherwise, update B_k to get B_{k+1} by the BFGS formula:

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}. \tag{2.16}$$

Step 4. Let $k := k + 1$ and go to Step 1.

Remark 2.1. Step 3 of Algorithm 2.1 ensures that B_k is always symmetric and positive definiteness.

We get the following lemma from Algorithm 2.1.

Lemma 2.3. *The sequence $\{\theta(\omega_k)\}$ is strictly decreasing. In addition, we have*

$$\lim_{k\to\infty} \|s_k\| = 0, \quad \lim_{k\to\infty} \|t_k F_k\| = 0. \tag{2.17}$$

Proof. From (2.15), it is clear that the sequence $\{\theta(\omega_k)\}$ is strictly decreasing. Moreover, it follows

$$\sigma_1 \|t_k d_k\|^2 + \sigma_2 \|t_k F_k\|^2 \leq \theta(\omega_k) - \theta(\omega_{k+1}),$$

then

$$\sigma_1 \sum_{k=0}^{n} \|s_k\|^2 + \sigma_2 \sum_{k=0}^{n} \|t_k F_k\|^2 \leq \theta(\omega_0) - \theta(\omega_{n+1}) \leq \theta(\omega_0).$$

Hence, equalities (2.17) hold. □

3 Convergence Analysis

In this section, we prove the global and superlinear convergence of Algorithm 2.1. In order to prove the global convergence of Algorithm 2.1, we make the following assumption.

Assumption 3.1. The level set $\Omega = \{\omega \in R^{n+1} | \theta(\omega) \leq \theta(\omega_0)\}$ is bounded.

Since the level set Ω is bounded on which F is continuously differentiable together with (2.2), it is not difficult to get that the Jacobian $F'(\omega)$ is bounded on Ω, namely, there is a constant $M > 0$ such that $\|F'(\omega)\| \leq M$ for all $\omega \in \Omega$. Then the following inequalities hold

$$\|\delta_k\| \leq M\|s_k\|, \|y_k\| \leq M\|\delta_k\| \leq M^2\|s_k\|, \tag{3.1}$$

for $\{\omega_k\} \subset \Omega$, where $\delta_k = F_{k+1} - F_k, y_k = F(\omega_k + \delta_k) - F(\omega_k)$. By $\|s_k\| \to 0$, then there is a constant $m > 0$ such that (see Lemma 2.2 in [11])

$$m\|s_k\|^2 \leq y_k^T s_k \leq M^2 \|s_k\|^2. \tag{3.2}$$

From (3.1) and (3.2), we get the following Lemma 3.2, whose proof is similar to that of Theorem 2.1 in [1] and is omitted.

Lemma 3.2. *Let $\{B_k\}$ be generated by Algorithm 2.1. If ω_k is not a stationary point of* (2.3)*, then there exist positive constants $\beta_1, \beta_2, \beta_3$, such that for any positive integer k, inequalities*

$$\beta_2\|s_i\| \leq \|B_i s_i\| \leq \beta_1\|s_i\|, \quad \beta_2\|s_i\|^2 \leq s_i^T B_i s_i \leq \beta_3\|s_i\|^2 \tag{3.3}$$

hold for at least $[k/2]$ values of $i \leq k$.

The following theorem shows a global convergence of Algorithm 2.1, its proof is similar to that of Theorem 3.3 in [5], and is omitted.

Theorem 3.3. *Let Assumption 3.1 hold and $\{\omega_k\}$ be generated by Algorithm 2.1. Then*

$$\liminf_{k\to\infty} \|\nabla\theta(\omega_k)\| = 0 \tag{3.4}$$

holds.

Theorem 3.3 shows that the iterative sequence has an accumulation point which is a stationary point of (2.3). It may not be a solution of (2.1) if the Jacobian matrix $F'(\omega)$ is singular at that point. The next theorem shows a strong global convergence of Algorithm 2.1.

Theorem 3.4. *Let Assumption 3.1 hold. Suppose that the sequence $\{\omega_k\}$ be generated by Algorithm 2.1 has a subsequence converging to a stationary ω^* at which $F'(\omega^*)$ is nonsingular. Then ω^* is a solution of (2.1), i.e., (x^*, λ^*) is a Z-eigenpair of the symmetric tensor $\mathcal{A}$. Moreover, the whole sequence $\{\omega_k\}$ converges to ω^*.*

Proof. This theorem follows from Theorem 3.4 in [5]. □

It is easy to prove that the Jacobian $F'(x, \lambda)$ of $F(x, \lambda)$ in (2.1) is Lipschitz continuous.

Lemma 3.5. *Let Assumption 3.1 hold. Then the Jacobian $F'(x, \lambda)$ is Lipschitz continuous in Ω, namely, there exists a constant $L > 0$ such that*

$$\|F'(x,\lambda) - F'(\bar{x},\bar{\lambda})\| \le L\|(x,\lambda) - (\bar{x},\bar{\lambda})\|, \forall (x,\lambda), (\bar{x},\bar{\lambda}) \in \Omega. \tag{3.5}$$

Proof.

$$F'(x,\lambda) - F'(\bar{x},\bar{\lambda}) = \begin{pmatrix} (m-1)(\mathcal{A}x^{m-2} - \mathcal{A}\bar{x}^{m-2}) - (\lambda - \bar{\lambda})I_n & \bar{x} - x \\ (\bar{x} - x)^T & 0 \end{pmatrix}. \tag{3.6}$$

By straightforward calculation, we have

$$\begin{aligned} \mathcal{A}x^{m-2} - \mathcal{A}\bar{x}^{m-2} &= \mathcal{A}x^{m-2} - \mathcal{A}x^{m-3}\bar{x} + \mathcal{A}x^{m-3}\bar{x} - \mathcal{A}x^{m-4}\bar{x}^2 \\ &\quad + \cdots + \mathcal{A}x\bar{x}^{m-3} - \mathcal{A}\bar{x}^{m-2} \\ &= [\mathcal{A}x^{m-3} + \mathcal{A}x^{m-4}\bar{x} + \cdots + \mathcal{A}\bar{x}^{m-3}](x - \bar{x}). \end{aligned} \tag{3.7}$$

Substituting into (3.6) with (3.7), it follows the conclusion of this lemma from basic property of matrix and vector norm. □

It is not difficult to prove the superlinear convergence of Algorithm 2.1.

Theorem 3.6. *Let the conditions of Theorem 3.4 hold. Since $F'(x, \lambda)$ is Lipschitz continuous in Ω, then*

$$\lim_{k\to\infty} \frac{\|(B_k - F'(\omega^*)^2)s_k\|}{\|s_k\|} = 0. \tag{3.8}$$

Moreover, $\{\omega_k\}$ is superlinearly convergent.

Proof. The proof of (3.8) is similar to that of the Dennis-Moré condition in [12]. In a way similar to the proof of Lemma 3.5 in [11], we obtain that $\{\omega_k\}$ converges superlinearly. □

4 Numerical Experiments

In this section, we discuss numerical test results for Algorithm 2.1 and SS-HOPM.

The parameters are specified as follows. We take $\rho = 0.5$ in Algorithm 2.1 and $\sigma_1 = \sigma_2 = 10^{-5}$ in (2.15). The initial quasi-Newton matrix is set to be $B_0 = A$ [11], where A is an $n \times n$ tridiagonal matrix given by

$$\begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}.$$

For Step 3 of Algorithm 2.1, we update B_k by (2.16) if $y_k^T s_k > 10^{-5}$, otherwise, we set $B_{k+1} = B_k$. We stop the iteration process if $\|F(\omega_k)\| \le 10^{-4}$, and stop the program if the iteration number is larger than 5000 for Examples 4.1 and 4.2. The program is coded in MATLAB 7.8. For any odd-order tensor (i.e., m odd), if (x, λ) is an eigenpair, $(-x, -\lambda)$ is also an eigenpair. For any even-order tensor (i.e., m even), if (x, λ) is an eigenpair, $(-x, \lambda)$ is also an eigenpair. Thus for Examples 4.1 and 4.3, we only give their positive real Z-eigenvalues and the corresponding Z-eigenvectors.

Example 4.1 ([10], Example 3.6). Let a symmetric tensor $\mathcal{A} \in R^{[3,3]}$ be defined by

$$\begin{array}{llll} a_{111} = -0.1281, & a_{112} = 0.0516, & a_{113} = -0.0954, & a_{122} = -0.1958, \\ a_{123} = -0.1790, & a_{133} = -0.2676, & a_{222} = 0.3251, & a_{223} = 0.2513, \\ a_{233} = 0.1773, & a_{333} = 0.0338. & & \end{array}$$

From Theorem 5.3 [10], we know that $\mathcal{A}$ has at most 7 eigenpairs. Using Algorithm 2.1, we ran 39 trials to get all the eigenpairs of the tensor with different random starting points w_0 chosen from a uniform distribution on $[-1, 1]^{n+1}$. The occurrences of each eigenpair are summarized in Table 4.1. The comparative numerical results of Example 4.1 are summarized in Table 4.2, Iters is the total number of iterations, Time(s) is the CPU time in seconds. Under the same initial conditions, 7 eigenpairs are obtained by Algorithm 2.1, only 3 eigenpairs are found by SS-HOMP with the same parameter $\alpha = 1$, and other eigenpairs may be obtained by SS-HOMP with different parameter α (see [10]).

Table 4.1: Eigenpairs for $\mathcal{A} \in R^{[3,3]}$ from Example 4.1 computed by Algorithm 2.1 with 39 random starts.

Occurrences	λ	x^T
5	0.8729	[-0.3922 0.7248 0.5664]
7	0.4306	[-0.7128 -0.1243 -0.6842]
9	0.2294	[-0.8448 0.4384 -0.3068]
9	0.0180	[0.7126 0.5097 -0.4820]
1	0.0033	[0.4470 0.7740 -0.4485]
6	0.0018	[0.3300 0.6319 -0.7013]
2	0.0006	[0.2914 0.7359 -0.6112]

Table 4.2: Numerical results for $\mathcal{A} \in R^{[3,3]}$ from Example 4.1

Algorithm 2.1						SS-HOPM					
λ	x^T			Iters	Time(s)	λ	x^T			Iters	Time(s)
0.8729	[-0.3922	0.7248	0.5664]	18	0.0040	0.8730	[-0.3922	0.7249	0.5664]	39	0.0018
0.4306	[-0.7128	-0.1243	-0.6842]	53	0.0119	0.4306	[-0.7187	-0.1245	-0.6840]	74	0.0014
0.2294	[-0.8448	0.4384	-0.3068]	124	0.0170	0.4306	[-0.7187	-0.1245	-0.6840]	92	0.0036
0.0180	[0.7126	0.5097	-0.4820]	491	0.0502	0.0180	[0.7132	0.5093	-0.4817]	179	0.0067
0.0033	[0.4470	0.7740	-0.4485]	599	0.0905	0.8730	[0.3922	0.7249	0.5664]	70	0.0031
0.0018	[0.3300	0.6319	-0.7013]	204	0.0336	0.4306	[-0.7187	-0.1245	-0.6840]	123	0.0049
0.0006	[0.2914	0.7359	-0.6112]	610	0.0496	0.0180	[0.7132	0.5093	-0.4817]	282	0.0106

Example 4.2 ([10], Example 3.5). As a second illustrative example, $\mathcal{A} \in R^{[4,3]}$ is the symmetric tensor defined by

$$\begin{array}{llll} a_{1111} = \ \ 0.2883, & a_{1112} = -0.0031, & a_{1113} = \ \ 0.1973, & a_{1122} = -0.2485, \\ a_{1123} = -0.2939, & a_{1133} = \ \ 0.3847, & a_{1222} = \ \ 0.2972, & a_{1223} = \ \ 0.1862, \\ a_{1133} = \ \ 0.0919, & a_{1333} = -0.3619, & a_{2222} = \ \ 0.1241, & a_{2223} = -0.3420, \\ a_{2233} = \ \ 0.2127, & a_{2333} = \ \ 0.2727, & a_{3333} = -0.3054. & \end{array}$$

From Theorem 5.3 [10], this problem has at most 13 eigenpairs. Using Algorithm 2.1, we ran 70 trials to get all real eigenpairs of the tensor as described for Example 4.1, where 7 trials are failure. The occurrences of each eigenpair are summarized in Table 4.3. The comparative numerical results of Example 4.2 are summarized in Table 4.4, where Iters and Time(s) are the same as those in Table 4.2. Under the same initial conditions, 11 real eigenpairs are obtained by Algorithm 2.1, only 3 real eigenpairs are found by SS-HOMP with the same parameter $\alpha = 2$.

Table 4.3: Eigenpairs for $\mathcal{A} \in R^{[4,3]}$ from Example 4.2 computed by Algorithm 2.1 with 70 random starts.

Occurrences	λ	x^T		
1	0.8893	[0.6672	0.2471	-0.7027]
6	0.8169	[0.8412	-0.2635	0.4722]
8	0.5104	[0.3598	-0.7779	0.5150]
10	0.3633	[0.2675	0.6448	0.7160]
10	0.2682	[0.6099	0.4362	0.6616]
5	0.2428	[0.1318	-0.4425	-0.8870]
2	0.2434	[0.9896	0.0947	-0.1088]
7	0.1734	[0.3357	0.9073	0.2532]
8	-0.0451	[0.7798	0.6136	0.1250]
5	-0.5630	[0.1762	-0.1796	0.9679]
1	-1.0953	[0.5915	-0.7467	-0.3043]

Example 4.3 Let $\mathcal{A}$ be the m-order n-dimensional diagonal tensor whose entries are

$$a_{ii\cdots i} = \begin{cases} i, & \text{if} \quad i \in \{1, 2, \cdots, n\}, \\ 0, & \text{otherwise}. \end{cases} \tag{4.1}$$

Let $P = I - 2uu^T$ be an $n \times n$ Household matrix, u is a unit vector in $\mathbb{R}^n$. Define $\mathcal{B} = P^m\mathcal{A}$

Table 4.4: Numerical results for $\mathcal{A} \in R^{[4,3]}$ from Example 4.2

Algorithm 2.1						SS-HOPM					
λ	x^T			Iters	Time(s)	λ	x^T			Iters	Time(s)
0.8893	[0.6672	0.2471	-0.7027]	14	0.0064	0.8893	[0.6672	0.2471	-0.7027]	44	0.0038
0.8169	[0.8412	-0.2635	0.4722]	11	0.0037	0.8169	[0.8412	-0.2635	0.4722]	41	0.0025
0.5104	[0.3598	-0.7779	0.5150]	16	0.0042	0.8893	[-0.6672	-0.2471	0.7027]	158	0.0078
0.3633	[0.2675	0.6448	0.7160]	14	0.0036	0.3633	[0.2676	0.6447	0.7160]	55	0.0030
0.2682	[0.6099	0.4362	0.6616]	8	0.0023	0.8169	[0.8412	-0.2635	0.4722]	63	0.0034
0.2428	[0.1318	-0.4425	-0.8870]	11	0.0049	0.3633	[-0.2676	-0.6447	-0.7160]	76	0.0033
0.2434	[0.9896	0.0947	-0.1088]	8	0.0024	0.8893	[0.6672	0.2471	-0.7027]	54	0.0016
0.1734	[0.3357	0.9073	0.2532]	7	0.0021	0.8893	[0.6672	0.2471	-0.7027]	130	0.0064
-0.0451	[0.7798	0.6136	0.1250]	9	0.0029	0.8893	[0.6672	0.2471	-0.7027]	83	0.0043
-0.5630	[0.1762	-0.1796	0.9679]	10	0.0032	0.8169	[0.8412	-0.2635	0.4722]	49	0.0014
-1.0953	[0.5915	-0.7467	-0.3043]	10	0.0033	0.8893	[0.6672	0.2471	-0.7027]	50	0.0014

as an m-order n-dimensional tensor with its entries as

$$b_{i_1,\cdots,i_m} = \sum_{j_1,\cdots,j_m=1}^{n} p_{i_1 j_1} \cdots p_{i_m j_m} a_{j_1,\cdots,j_m}.$$

Let $m = 11, n = 5, u = \frac{1}{\sqrt{2}}(1,1,0,0,0)^T$. Using Algorithm 2.1, we find 5 real Z-eigenpairs of $\mathcal{B}$, where we stop the iteration process if $\|F(\omega_k)\| \leq 10^{-2}$, and stop the program if the iteration number is larger than 5000. The five real Z-eigenvalues and the corresponding Z-eigenvectors are shown in Table 4.5.

Table 4.5: Eigenpairs for $\mathcal{B} \in R^{[11,5]}$ from Example 4.3

λ	x^T				
1.0131	[-0.0005	-1.0009	0.0002	-0.0012	0.0005]
1.9347	[-0.9963	0.0003	0.0002	0.0001	-0.0001]
3.0003	[-0.0001	0.0002	1.0000	-0.0001	0.0002]
4.0253	[0.0001	-0.0017	-0.0005	1.0008	-0.0010]
4.8576	[0.0000	0.0001	0.0001	0.0001	0.9968]

5 Comparison to Other Methods

From Tables 4.2 and 4.4, we see that Algorithm 2.1 is able to find eigenpairs not found by SS-HOPM. For fixed m and large n, the computational complexity of each iteration of Algorithm 2.1 is $O(n^m)$. The work for computing $\|B_k^{-1}\|_\infty$ is $O(n^3)$, and for $m \geq 3$ it doesn't affect the complexity scale, which remains $O(n^m)$. The computational complexity of each iteration of SS-HOPM is $O(n^m)$, which is the work of computing $\mathcal{A}x^{m-1}$. The computational cost of Algorithm 2.1 is higher than that of SS-HOPM in each iteration, which may be balanced by the superlinear convergence of Algorithm 2.1.

The sequential semidefinite programming method in [6] can compute the largest or smallest Z-eigenvalues for even order tensors. Jacobian SDP relaxations method in [4] can find all real eigenvalues theoretically, and may become computationally expensive for large m and n. Algorithm 2.1 is a simple method for computing a Z-eigenpair of a symmetric tensor. For small tensor, Algorithm 2.1 may compute out all the Z-eigenpairs by choosing the different initial points. For large m and n, Algorithm 2.1 cannot guarantee to compute out all the Z-eigenpairs. If we combine the sequential technique and some constrained conditions in [4] and [6] to Algorithm 2.1, then an efficient method may be obtained. This can be further studied in the future.

Acknowledgement

The authors are very grateful to the editors and the two referees for their valuable suggestions and constructive comments, which have considerably improved the presentation of the paper.

References

[1] R.H. Byrd and J. Nocedal, A tool for the analysis of quasi-Newton methods with application to unconstrained minimization, *SIAM J. Numer. Anal.* 26 (1989) 727–739.

[2] K.C. Chang and T. Zhang, On the uniqueness and non-uniqueness of the positive Z-eigenvector for transition probability tensors, School of Mathematical Sciences, Peking University, 2012.

[3] J. Cooper and A. Dutle, Spectra of uniform hypergraphs, *Linear Algebra Appl.* 436 (2012) 3268–3292.

[4] C. Cui, Y. Dai and J. Nie, All real eigenvalues of symmetric tensors, arXiv. 1403.3720.

[5] G.Z. Gu, D.H. Li and L. Qi et al., Descent directions of quasi-Newton methods for symmetric nonlinear equations, *SIAM J. Numer. Anal.* 40 (2002) 1763–1774.

[6] S. Hu, Z.H. Huang and L. Qi, Finding the extreme Z-eigenvalues of tensors via a sequential semidefinite programming method, *Numerical Linear Algebra with Applications* 20 (2013) 972–984.

[7] S. Hu and L. Qi, Algebraic connectivity of an even uniform hypergraph, *J. Combinatorial Optim.* 24 (2012) 564–579.

[8] S. Hu, L. Qi, Convergence of a second order Markov chain, *Applied Mathematics and Computation* 241 (2014) 183–192.

[9] E. Kofidis and P.A. Regalia, On the best rank-1 approximation of higher-order supersymmetric tensors, *SIAM J. Matrix Anal. Appl.* 23 (2002) 863–884.

[10] T.G. Kolda and J.R. Mayo, Shifted power method for computing tensor eigenpairs, *SIAM J. Matrix Anal. Appl.* 32 (2011) 1095–1124.

[11] D.H. Li and M. Fukushima, A global and superlinear convergent Gauss-Newton-based BFGS method for symmetric nonlinear equations, *SIAM J. Numer. Anal.* 37 (1999) 152–172.

[12] D.H. Li and M. Fukushima, A modified BFGS method and its global convergence in nonconvex minimization, *J. Comput. Appl. Math.* 129 (2001) 15–35.

[13] W. Li and M. Ng, Existence and uniqueness of stationary probability vector of a transition probability tensor, Department of Mathematics, The Hong Kong Baptist University, 2011.

[14] W. Li and M. Ng, On linear convergence of power method for computing stationary probability vector of a transition probability tensor, Department of Mathematics, The Hong Kong Baptist University, 2011.

[15] X. Li, M. Ng and Y. Ye, Finding stationary probability vector of a transition probability tensor arising from a higher-order Markov chain, Department of Mathematics, The Hong Kong Baptist University, 2011.

[16] G. Ni, L. Qi and F. Wang et al., The degree of the E-characteristic polynomial of an even order tensor, *J. Math. Anal. Appl.* 329 (2007) 1218–1229.

[17] Q. Ni, L. Qi and F. Wang, An eigenvalue method for the positive definiteness identification problem, *IEEE Trans. Autom. Control* 53 (2008) 1096–1107.

[18] Q. Ni and L. Qi, A quadratically convergent algorithm for finding the largest eigenvalue of a nonnegative homogeneous polynomial map, *Journal of Global Optimization* (2014). To appear.

[19] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables,* SIAM, Philadelphia, 2000.

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

[21] L. Qi, Rank and eigenvalues of a supersymmetric tensor, a multivariate homogeneous polynomial and an algebraic surface defined by them, *J. Symb. Comput.* 41 (2006) 1309–1327.

[22] L. Qi, Eigenvalues and invariants of tensors, *J. Math. Anal. Appl.* 325 (2007) 1363-1377.

[23] L. Qi and K.L. Teo, Multivariate polynomial minimization and its application in signal processing, *J. Global Optim.* 46 (2003) 419–433.

[24] L. Qi, F. Wang and Y. Wang, Z-eigenvalue methods for a global polynomial optimization problem, *Math. Program.* 118 (2009) 301–316.

[25] L. Qi, G. Yu and E.X. Wu, Higher order positive semi-definite diffusion tensor imaging, *SIAM J. Imaging Sci.* 3 (2010) 416–433.

[26] L. Zhang and L. Qi, Linear convergence of an algorithm for computing the largest eigenvalue of a nonnegative tensor, *Numerical Linear Algebra with Applications* 19 (2012) 830–841.

Manuscript received 14 April 2014
revised 28 August 2014, 12 October 2014
accepted for publication 12 October 2014

Meilan Zeng
College of Economics and Management
NUAA, Nanjing 210016, PRC

College of Mathematics and Statistics
Hubei Engineering University
Xiaogan Hubei 432000, PRC
E-mail address: zml571@126.com

Qin Ni
College of Economics and Management
NUAA, Nanjing 210016, PRC
E-mail address: niqfs@nuaa.edu.cn