Yokohama Publishers
ISSN 1349-8169 ONLINE JOURNAL

Pacific Journal of Optimization
Vol. 13, No. 2, pp. 243–253, April 2017

A SMOOTHING NEWTON METHOD FOR TENSOR EIGENVALUE COMPLEMENTARITY PROBLEMS*

Wenyu Hu, Laishui Lu, Cheng Yin and Gaohang Yu

Abstract: This paper investigates a smoothing Newton method for tensor eigenvalue complementarity problems (TEiCP) which are closely related to the optimality conditions for polynomial optimization and have wide applications. By introducing a smoothing approximation of NCP-function, the TEiCP problem can be reformulated as a system of smooth equations. Then, a smoothing Newton method is proposed for tensor eigenvalue complementarity problems (TEiCP). Its convergence and convergence rate could be guaranteed by existing results. Numerical experiments are reported to show that the proposed method is efficient and could detect more solutions than some existing optimization-based methods.

Key words: *Tensor eigenvalue complementarity problems, smoothing Newton method, NCP-function*

Mathematics Subject Classification: *15A18, 15A69, 90C33, 49M15*

1 Introduction

An mth-order n-dimensional real tensor $\mathcal{A}$ consists of n^m entries in real numbers:

$$\mathcal{A} = (a_{i_1 i_2 \cdots i_m}),\ a_{i_1 i_2 \cdots i_m} \in \mathbb{R},\ \text{for any}\ \ i_1, i_2, \ldots, i_m \in [n],$$

where $[n] = \{1, 2, \ldots, n\}$. $\mathcal{A}$ is called *symmetric* if the value of $a_{i_1 i_2 \cdots i_m}$ is invariant under any permutation of its indices $i_1, i_2, \ldots, i_m$. Let $\mathbb{T}^{[m,n]}$ denote the set of all real mth-order n-dimensional tensors and $\mathbb{S}^{[m,n]}$ denote the set of all real symmetric mth-order n-dimensional tensors.

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

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

and $\mathcal{A}x^m$ is a homogeneous polynomial function defined by

$$\mathcal{A}x^m = x^T(\mathcal{A}x^{m-1}) = \sum_{i_1, \ldots, i_m = 1}^{n} a_{i_1 \cdots i_m} x_{i_1} \cdots x_{i_m}.$$

*This work was supported in part by the National Natural Science Foundation of China (No.61262026, 11661007, 61502107), NCET Programm of the Ministry of Education (NCET 13-0738), science and technology programm of Jiangxi Education Committee (LDJH12088).

A tensor $\mathcal{A} \in \mathbb{S}^{[m,n]}$ is called *positive definite* if $\mathcal{A}x^m > 0$ for all $x \neq 0$. Clearly, when m is odd, there is no positive definite tensors.

In this paper, we consider the tensor eigenvalue complementarity problem (TEiCP): for given tensors $\mathcal{A}, \mathcal{B} \in \mathbb{T}^{[m,n]}$, finding a scalar $\lambda \in \mathbb{R}$, and $x \in \mathbb{R}^n \backslash \{0\}$ such that

$$x \geq 0, (\lambda\mathcal{B} - \mathcal{A})x^{m-1} \geq 0, \langle x, (\lambda\mathcal{B} - \mathcal{A})x^{m-1} \rangle = 0. \tag{1.1}$$

When $m = 2$, TEiCP is reduced to the classical matrix eigenvalue complementarity problem which has wide applications in mechanical systems [14]. TEiCP also has wide applications such in higher-order Markov chains [19], magnetic resonance imaging [20]. The solution of TEiCP (λ, x) is called Pareto eigenpair of $(\mathcal{A}, \mathcal{B})$. In pariticular, its solution could be called Pareto H-eigenpair or Pareto Z-eigenpair when the tensor $\mathcal{B}$ has some special form [21]. Replacing the nonnegative cones in (1.1) by a closed convex cone and its dual cone, Ling, He and Qi investigated the cone eigenvalue complementarity problem for higher-order tensor in [16]. As shown in [16], the TEiCP has at least one solution under the assumption that $\mathcal{B}x^m \neq 0$ for all $x \in \mathbb{R}^n_+ \backslash \{0\}$. Moreover, in [17], they studied the high-degree eigenvalue complementarity problem for tensors as a natural extension of quadratic eigenvalue complementarity problem for matrices. TEiCP is also closely related to the optimality conditions for polynomial optimization [21], a class of differential inclusions with noncovex processes [16], and a kind of nonlinear differential dynamical system [7]. The properties of Pareto eigenvalues and their connection to polynomial optimization are studied in [21].

Recently, as a special type of nonlinear complementarity problems, the tensor complementarity problems is inspiring more and more research in the literature [3, 5–8, 10, 16, 18, 22, 23]. One popular approach is to reformulate the TEiCP equivalently as a polynomial optimization problem [7, 9, 16]. A shifted projected power method for TEiCP was proposed in [7], in which they need an adaptive shift to force the objective to be (locally) convex to guarantee the convergence of power method. In [16], Ling, He and Qi presented a scaling-and-projection algorithm (SPA) for TEiCP. One main shortcoming of SPA is the stepsize will approach to zero as the sequence gets close to a solution of TEiCP [16]. These methods are based on first order gradient information. So, these methods can only find a local stationary point for TEiCP. Recently, Fan, Nie and Zhou [9] proposed a Lasseree's hierarchy of semidefinite relaxation method for finding all solutions of TEiCP. On the other hand, by introducing an NCP-function, Chen and Qi [5] reformulated the TEiCP as a system of nonlinear equations. And then, they proposed a semismooth Newton method for solving the system of nonlinear equations [5]. Since the objection function is not necessary continuously differentiable, they need a subalgorithm to evaluate an element of the generalized Jacobian of objection function. This may bring some numerical troubles in practice.

In this paper, we will investigate a smoothing Newton method for computing TEiCP. Numerical experiments show that the proposed smoothing Newton method will be more robust than semismooth Newton method and it could detect more solutions than some existing optimization-based methods. The rest of this paper is organized as follows. In Section 2, we first reformulated the TEiCP as a system of smooth equations. Then, a smoothing Newton method was presented. Some numerical results are reported in Section 3. Finally, we have a conclusion section.

Throughout this paper, we assume that m is even and $\mathcal{B} \in \mathbb{S}^{[m,n]}$ is positive definite. Let small bold letters $x, y, \cdots$, denote vectors, calligraphic $\mathcal{A}, \mathcal{B}, \cdots$, denote tensors. All the tensor discussed here are real. $\mathbb{R}^n$ (respectively,$\mathbb{R}$) denotes the space of n dimensional real column vectors (respectively,real number), $\mathbb{R}^n_+$ and $\mathbb{R}^n_{++}$ denote the nonnegative and positive orthant of $\mathbb{R}^n$, $\mathbb{R}_+$ (respectively,$\mathbb{R}_{++}$) denotes the nonnegative (respectively,positive) line in $\mathbb{R}$.

2 Preliminaries and Algorithm

In this section, we present our smoothing Newton method and some preliminaries. For convenience, let $z = (x, y, \lambda) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$.

Suppose that m is even and $\mathcal{A} \in \mathbb{S}^{[m,n]}$. In [5], using the penalized Fischer-Burmeister NCP function

$$\phi_\tau(x, y) := \tau\phi_{FB}(x, y) + (1 - \tau)x_+y_+, \tag{2.1}$$

where $\tau \in (0, 1), \phi_{FB}(x, y) = (x + y) - \sqrt{x^2 + y^2}, a_+ = \max\{a, 0\}$ for $a \in R$,
TEiCP (1.1) can be reformulated as the following semismooth system of equations:

$$H(z) := H(x, y, \lambda) = \begin{pmatrix} \Phi_\tau(x, y) \\ (\lambda\mathcal{B} - \mathcal{A})x^{m-1} - y \\ x^T x - 1 \end{pmatrix} = 0, \tag{2.2}$$

where

$$\Phi_\tau(x, y) = \begin{pmatrix} \phi_\tau(x_1, y_1) \\ \phi_\tau(x_2, y_2) \\ \vdots \\ \phi_\tau(x_n, y_n) \end{pmatrix}. \tag{2.3}$$

Since the NCP function ϕ_τ in (2.1) are not continuously differentiable at $(0, 0)$, Chen and Qi [5] proposed a semismooth Newton method for (2.2). In this paper, we will introduce the smoothing approximation function of the Fischer-Burmeister NCP function

$$\phi(\mu, x, y) = (x + y) - \sqrt{x^2 + y^2 + 2\mu^2}, \quad \mu \in \mathbb{R}_{++}. \tag{2.4}$$

Let $\omega = (\mu, z)$, TEiCP can be reformulated as the following smoothing system of equations:

$$H(\omega) = H(\mu, z) = H(\mu, x, y, \lambda) = \begin{pmatrix} \mu \\ \Phi(\mu, x, y) \\ (\lambda\mathcal{B} - \mathcal{A})x^{m-1} - y \\ x^T x - 1 \end{pmatrix} = 0, \tag{2.5}$$

where

$$\Phi(\mu, x, y) = \begin{pmatrix} \phi(\mu, x_1, y_1) \\ \vdots \\ \phi(\mu, x_n, y_n) \end{pmatrix}. \tag{2.6}$$

Define the following merit function associated with $H(\cdot)$ given in (2.5),

$$\Psi(\omega) = \frac{1}{2}\|H(\omega)\|^2, \tag{2.7}$$

TEiCP(1.1) is also equivalent to the following unconstrained optimization problem

$$\min \Psi(\omega), \tag{2.8}$$

with object function value zero.

Denote the Jacobian matrix of $H(\omega)$ by $H'(\omega)$, then we have following Lemma.

Lemma 2.1. *Let $H : \mathbb{R}^{2n+2} \to \mathbb{R}^{2n+2}$ be defined by (2.5), then*

H is continuously differentiable at any $w = (\mu, x, y, \lambda) \in \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$ with its Jacobian

$$H'(\omega) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \nu(\omega) & D_a(\omega) & D_b(\omega) & 0 \\ 0 & (m-1)(\lambda\mathcal{B} - \mathcal{A})x^{m-2} & -I & \mathcal{B}x^{m-1} \\ 0 & 2x^T & 0 & 0 \end{pmatrix}, \tag{2.9}$$

where

$$\nu(\omega) := (\nu_1(\omega), \cdots, \nu_n(\omega))^T,$$
$$D_a(\omega) := diag\{a_1(\omega), \cdots, a_n(\omega)\},$$
$$D_b(\omega) := diag\{b_1(\omega), \cdots, b_n(\omega)\},$$

with

$$\nu_i(\omega) = -\frac{2\mu}{\sqrt{x_i^2 + y_i^2 + 2\mu^2}},$$
$$a_i(\omega) = 1 - \frac{x_i}{\sqrt{x_i^2 + y_i^2 + 2\mu^2}},$$
$$b_i(\omega) = 1 - \frac{y_i}{\sqrt{x_i^2 + y_i^2 + 2\mu^2}},$$

the $i = 1, 2, \cdots n$.

Before giving smoothing Newton method, we define a nonnegative function $\beta(\omega)$. Let $\gamma \in (0, 1), \mu \in \mathbb{R}_{++}$ and $z := (x, y, \lambda) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$. Denote $\omega = (\mu, z)$ and a real-value function $\beta : \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}_+$ by

$$\beta(\omega) := \gamma \|H(\omega)\| \min\{1, \|H(\omega)\|\}. \tag{2.10}$$

The relationship between $H(\omega)$ and $\beta(\omega)$ can be seen in [13].

Lemma 2.2. *The following relations hold*

$$H(\omega) = 0 \Leftrightarrow \beta(\omega) = 0 \Leftrightarrow H(\omega) = \beta(\omega)\bar{\mu},$$

where $\bar{\mu} = (\mu, 0) \in \mathbb{R}_{++} \times \mathbb{R}^{2n+1}$.

In the following, we will show that functions (2.5) and (2.6) are all strongly semismooth.

Lemma 2.3. *The function Φ is strongly semismooth, where Φ is defined by (2.6). Moreover, the function $H(\omega)$ is strongly semismooth, where $H(\omega)$ is defined by (2.5).*

Proof. It is well known that the function Φ strongly semismooth. Note that for any $(x, y, \lambda) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$, the function $(\lambda\mathcal{B} - \mathcal{A})x^{m-1} - y$ is continuously differentiable and its Jacobian is locally Lipschitz continuous. It is clear that $x^T x - 1$ and u is continuously differentiable. It follows that $H(\omega)$ is strongly semismooth since all their components are strongly semismooth. □

Now we present a smoothing Newton method for tensor eigenvalue complementarity problem.

Algorithm1 : A Smoothing Newton algorithm for TEiCP, SNM

Step 0: Given $\varepsilon > 0$. Choose parameters $\delta, \sigma \in (0,1), \mu_0 > 0$. Let $\bar{u} := (\mu_0, 0, 0, 0) \in \mathbb{R}_{++} \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$ and $z_0 := (x_0, y_0, \lambda_0) \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}$ be an arbitrary initial point. Take $w_0 = (\mu_0, z_0)$ and choose parameter $\gamma \in (0,1)$ such that $\gamma\|H(w_0)\| < 1$ and $\gamma\mu_0 < 1$. Set $k = 0$.

Step 1: Stop, if $\|H(\omega_k)\| \leq \varepsilon$. Otherwise, compute $\beta_k := \beta(\omega_k)$, where the $\beta(\bullet)$ is defined by (2.10).

Step 2: Solve the following equation to obtain $\triangle\omega_k := (\triangle\mu_k, \triangle x_k, \triangle y_k, \triangle\lambda_k)$

$$H(\omega_k) + H'(\omega_k)\triangle\omega_k = \beta_k\bar{u}. \tag{2.11}$$

Let m_k be the smallest nonnegative integer m such that

$$\|H(w_k + \delta^m\triangle w_k)\| \leq [1 - \sigma(1 - \gamma\mu_0)\delta^m]\|H(\omega_k)\|. \tag{2.12}$$

Set $\alpha_k := \delta^{m_k}$ and go to Step 4.

Step 3: If the equation (2.11) is unsolvable, set $\triangle\omega_k = -\nabla\Psi(\omega_k)$. Let m_k be the smallest nonnegative integer m such that

$$\Psi(\omega_k + \delta^m\Delta\omega_k) - \Psi(\omega_k) \leq -\sigma\delta^m\|\Delta\omega_k\|^2. \tag{2.13}$$

Set $\alpha_k := \delta^{m_k}$ and go to Step 4.

Step 4: Set $\omega_{k+1} = \omega_k + \alpha_k\triangle\omega_k$, k :=k+1. Go to step 1.

When (2.11) is solvable, by the Theorem 6.5.9 of [13] or Theorem 2.5 of [24], we can obtain that there exists a suitable stepsize α such that $\|H(\omega_k)+\alpha\triangle\omega_k\| \leq [1-\sigma(1-\gamma\mu_0)\alpha]\|H(\omega_k\|$. In the case of that (2.11) is unsolvable, the negative gradient is used. It is well-known that (2.13) will be satisfied for some integer m.

The global convergence could be guaranteed by the following theorem.

Theorem 2.4. *Suppose that the solution set $\sigma(\mathcal{A}, \mathcal{B})$ is nonempty. Let $\omega_k = (\mu_k, x_k, y_k, \lambda_k)$ be generated by smoothing Newton. Then:*

(i) $\|H(\omega_{k+1})\| \leq \|H(\omega_k)\|$;

(ii) *Each accumulation point of the sequence $\{\omega_k\}$ generated by the Algorithm 1 is a stationary point of Ψ.*

Proof. First, by (2.12), $\|H(\omega_k)\|$ will satisfies (i). Next, we prove (ii).

Suppose that $\{\omega_k\}_K \to \omega^*$, $\{\omega_k\}_K$ is a subsequence of $\{\omega_k\}$, and $k = 1, 2, \ldots$. If there exists an infinite set of indices K such that $\triangle\omega_k = -\nabla\Psi(\omega_k)$ for all $k \in K$, then ω^* is a stationary point of Ψ by the Proposition 1.16 in [4]. Hence, without loss of generality, to prove the theorem we only need to consider the case in which the direction is always computed by (2.11). Then, by the Theorem 6.5.9 in [13], we obtain that any accumulation point ω^* of the sequence ω_k satisfies $H(\omega^*) = 0$. Thus, we have $\nabla\Psi(\omega^*) = 0$. We complete the proof. □

Furthermore, if the equation (2.11) is always solvable, i.e. this means that the system of linear equation (2.11) is well-posed with a good condition number, then we can establish the convergence rate of proposed algorithm by following theorem.

Theorem 2.5. *If the equation (2.11) is always solvable. Let $\omega_k = (\mu_k, x_k, y_k, \lambda_k)$ be generated by algorithm 1, then the $\{\omega_k\}$ converges to ω^* quadratically.*

Proof. By the lemma 3, we know that the function $H(\omega)$ defined by (2.5) is strongly semismooth. Then, by the Theorem 6.5.10 in [13], we have

$$\|\omega_{k+1} - \omega^*\| = O(\|\omega_k - \omega^*\|^2). \tag{2.14}$$

The proof is completed. □

3 Numerical experiments

In this section, we present some numerical results to verify the effectiveness of the smoothing Newton method (SNM). We compare it with Scaling-and-Projection Algorithm (SPA) proposed by Ling, He and Qi [16], shifted projected power (SPP) method in [7] and semismooth Newton method(SSNM) proposed by Chen, Qi [5]. All codes were written in Matlab (R2015b) and the Tensor Toolbox Version 2.1 [1].

In the implementation of our algorithm, we choose parameters $\varepsilon = 10^{-6}, \mu_0 = 0.05, \delta = 0.6, \sigma = 0.15, \rho = 0.4$ and $\gamma = 0.01$ and using the stop condition $\|H(\omega)\| \leq \varepsilon$ and $k < 300$ for SNM and SSNM methods. On the other hand, the stop condition is $\|x_{k+1} - x_k\| \leq \varepsilon$ for SPA and SPP.

3.1 Numerical experiments for computing Pareto H-eigenpairs

In this subsection, we test SSNM, SNM, SPP and SPA methods for finding Pareto H-eigenpairs of irreducible nonnegative $\mathcal{A}$. The tensor $\mathcal{A} \in S_{6,4}$ is described in Table 4 of [5]. As shown in [7], there exists a unique solution for TEiCP when $\mathcal{A} \in T_{m,n}$ is irreducible nonnegative and $\mathcal{B} = \mathcal{I} \in S_{m,n}$, where $\mathcal{I}$ is the diagonal tensor with diagonal entries 1 and 0 otherwise. The initial point is chosen as $x_0 = e/\|e\|, \lambda_0 = \frac{\mathcal{A}x_0^m}{\mathcal{B}x_0^m}, y_0 = (\lambda\mathcal{B} - \mathcal{A})x_0^{m-1}$, and $e = (1, 1, \ldots, 1) \in \mathbb{R}^n$. Numerical results are listed in Table 1, in which **Its** denotes the average number of iteration for each solution, **Time** denotes the average time of iteration. As we can see from Table 1, all of the methods could reach the unique H-eigenpair. SNM is competitive to SSNM and they are much faster than SPP method.

Table 1. Comparison results for computing Pareto H-eigenvalues of $\mathcal{A}$ from [5]

Alg.	λ	Eigenvector	Its.	Time (sec.)
SSNM	515.5227	(0.4995 0.4988 0.5006 0.5011)	5	0.1025
SNM	515.5227	(0.4995 0.4988 0.5006 0.5011)	5	0.0860
SPP	515.5227	(0.4995 0.4988 0.5006 0.5011)	13	0.2872

Table 2. SNM for computing Parteo-H eigenvalue of 100 random nonnegative tensors

m	n	Suc	$Its.$	Time(sec.)	λ^*
4	3	100	4.10	0.0324	0.1353+e02
4	10	100	5.26	0.0471	0.4997+e03
4	20	100	5.92	0.1152	3.9992+e03
4	30	100	6.00	0.3542	1.3499+e04
4	40	100	6.00	0.9548	3.2000+e04
4	50	100	6.00	2.3075	6.2490+e04
6	3	100	4.04	0.0324	0.1214+e03
6	4	100	4.24	0.0456	0.5115e+03
6	6	100	4.62	0.0597	3.8877e+03
6	8	100	4.87	0.2004	1.6380e+04
6	10	100	5.05	0.5015	4.9996e+04
8	3	100	6.00	0.0549	1.0941e+03
8	4	100	7.00	0.1015	8.1916e+03
8	6	100	8.00	1.4298	0.1399e+06

Table 3. SSNM for computing Parteo-H eigenvalue of 100 random nonnegative tensors

m	n	Suc	$Its.$	Time(sec.)	λ^*
4	3	100	5.00	0.0398	0.1353+e02
4	10	100	6.28	0.0612	0.4997+e03
4	20	100	7.00	0.1773	3.9992+e03
4	30	100	8.00	0.6601	1.3499+e04
4	40	100	9.00	1.9694	3.2000+e04
4	50	100	9.00	4.9866	6.2490+e04
6	3	100	5.00	0.0324	0.1214+e03
6	4	100	5.01	0.0554	0.5115e+03
6	6	100	6.00	0.0863	3.8877e+03
6	8	100	7.00	0.3710	1.6380e+04
6	10	100	7.00	0.9256	4.9996e+04
8	3	100	3.93	0.0450	1.0941e+03
8	4	100	4.09	0.0778	8.1916e+03
8	6	100	4.50	0.6053	0.1399e+06

In the second numerical experiment, our object is to use some randomly generated tensors to test the performance of SNM with comparisons to SSNM. We first select random entries from the interval $(0, 1)$, then symmetrize it by using $\mathcal{A} = symmetrize(\mathcal{A})$. The initial point is chosen as same as in the first numerical experiment. The numerical results are reported in Table 2 and Table 3. For each case, we use a sample of 100 random tensors to record the the number of success (Suc) and the average value of H-eigenvalues (λ^*). As we can see, both SNM and SSNM can converge to the unique solution, and SNM is competitive to SSNM.

3.2 Numerical experiments for computing Pareto Z-eigenpairs

In this subsection, our goal is to compute Pareto Z-eigenpairs for the following Example 3 and Example 4.

Example 3. Let $\mathcal{A} \in \mathbb{S}^{[4,3]}$ be the symmetric tensor defined by: Firstly, set $A = tensor(zeros(3, 3, 3, 3))$, and

$$a_{1222} = 1, \quad a_{1333} = 1,$$
$$a_{2111} = 1, \quad a_{3111} = 1,$$

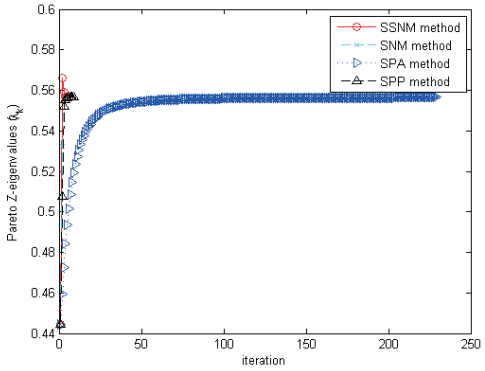

Figure 1: Comparison with SPA, SPP, SSNM,SPP algorithms for computing Pareto Z-eigenvalues of $\mathcal{A}$ in *Example 3*, and the starting point is $x_0 = [1.0; 1.0; 1.0]$

and then using $\mathcal{A} = symmetrize(\mathcal{A})$ to symmetrize it.

Table 4 lists the numerical results for computing Pareto Z-eigenvalues of $\mathcal{A}$ in *Example 3* from the starting point is $x_0 = [1.0; 1.0; 1.0]$. In this case, all of the test algorithms can reach the same Pareto Z-eigenvalue 0.5566. SNM need to run 5 iterations in 0.1328 seconds while SPA method need 229 iterations in 2.0233 seconds. SPP method need to run 9 iterations in 0.2817 seconds. According to the CPU time, SSNM cost much more time than SNM.

Table 4. Comparison results for computing Pareto Z-eigenvalues of $\mathcal{A}$ from *Example 3*.

Alg.	λ	Eigenvector	Its.	Time (sec.)
SSNM	0.5566	(0.8002,0.4240,0.4240)	5	0.5437
SNM	0.5566	(0.8002,0.4240,0.4240)	5	0.1328
SPA	0.5566	(0.7948,0.4291,0.4291)	229	2.0233
SPP	0.5566	(0.8004 0.4239 0.4239)	9	0.2817

The fourth numerical example is originally from [15] and was used in evaluating the SS-HOPM algorithm in [12] and the GEAP algorithm in [11].

Example 4 (Kofidis and Regalia [15]). Let $\mathcal{A} \in \mathbb{S}^{[4,3]}$ be the symmetric tensor defined by

$$\begin{aligned} a_{1111} = 0.2883, \quad a_{1112} = -0.0031, \quad a_{1113} = 0.1973, \quad a_{1122} = -0.2485, \\ a_{1223} = 0.1862, \quad a_{1133} = 0.3847, \quad a_{1222} = 0.2972, \quad a_{1123} = -0.2939, \\ a_{1233} = 0.0919, \quad a_{1333} = -0.3619, \quad a_{2222} = 0.1241, \quad a_{2223} = -0.3420, \\ a_{2233} = 0.2127, \quad a_{2333} = 0.2727, \quad a_{3333} = -0.3054. \end{aligned}$$

In order to get all possible solutions, 100 random initial points are used for all methods. The initial point is generated by the following rule: firstly generate a random vector $\xi \in R^n$ with uniform distribution in $[0, 1]^n$, and then set $x_0 = \xi/\|\xi\|, \lambda_0 = \mathcal{A}x^m/\mathcal{B}x^m, y_0 = (\lambda_0\mathcal{B} - \mathcal{A})x^{m-1}$. The numerical results are reported in Table 5, in which **No** denotes number of each solution detected by the method within 100 random initial points. As we can see, SNM and SSNM are able to detect 7 Pareto Z-eigenpairs and 5 Pareto Z-eigenpairs, respectively,

whereas the SPA method and SPP method can only find 3 Pareto Z-eigenpairs. Especially, the SNM is much faster than the SPA method. We also find that SSNM has 18% points which can't detect the Pareto Z-eigenvalue and SNM only has 4% points which can't find the Pareto Z-eigenvalue.

On the other hand, in order to compare the convergence in terms of the number of the iterations. Table 6 presents the results for computing Pareto Z-eigenvalues of $\mathcal{A}$ in *Example 4*, and the starting point is $x_0 = [1.0; 1.0; 1.0]$. In this case, both SSNM and SNM can reach the Pareto Z-eigenvalue $\lambda = 0.2682$ while the other two methods reach the Z-eigenvalue $\lambda = 0.3632$. SSNM, SNM need 6/5 iterations while SPA need to run 260 iterations, and SPP need 7 iterations in this case.

Table 5. All possible solutions detected by SSNM, SNM, SPA, SPP for tensor $\mathcal{A}$ in *Example 4* from 100 random initial points.

Alg.	No	λ^*	x^*	Its.	Time(sec)
SSNM	31	0.3633	(0.2676,0.6447,0.7160)	4.9677	0.0991
	25	0.2682	(0.6099,0.4362,0.6616)	5.0004	0.1602
	9	0.1735	(0.3357,0.9073,0.2531)	4.0333	0.0871
	11	0.6798	(0.8843,0.0000,0.4669)	5.1818	0.1034
	6	0.2938	(0.2742,0.9617,0.0000)	5.333	0.1046
	18	failure			
SNM	22	0.3633	(0.2676,0.6447,0.7160)	5.3181	0.1156
	38	0.2682	(0.6099,0.4362,0.6616)	4.7894	0.1022
	16	0.1735	(0.3357,0.9073,0.2531)	4.8750	0.1053
	7	0.6798	(0.8843,0.0000,0.4669)	5.0000	0.1167
	6	-0.0451	(0.7797,0.6135,0.1250)	5.0000	0.1071
	5	-0.0077	(0.8003,0.5995,0.0000)	5.0000	0.1120
	2	0.2938	(0.2742,0.9617,0.0000)	5.4000	0.1251
	4	failure			
SPA	57	0.3632	(0.2771,0.6461,0.7112)	233.4150	2.0934
	12	0.2938	(0.2742,0.9617,0.0000)	40.25	0.3711
	31	0.6798	(0.8845,0.0000,0.4666)	20.74	0.2016
SPP	57	0.3633	(0.2679,0.6448,0.7159)	9.1578	0.1748
	17	0.2938	(0.2742,0.9617,0.0000)	4.6470	0.0958
	26	0.6798	(0.8843,0.0000,0.4669)	4.6470	0.0958

Table 6. Comparison results for computing Pareto Z-eigenvalues of $\mathcal{A}$ in *Example 4* from the initial point $x_0 = [1.0; 1.0; 1.0]$.

Alg.	λ	Eigenvector	Its.	Time (sec.)
SSNM	0.2682	(0.6099,0.4362,0.6616)	6	0.0860
SNM	0.2682	(0.6099,0.4362,0.6616)	5	0.0832
SPA	0.3632	(0.2771,0.6461,0.7112)	260	2.0137
SPP	0.3632	(0.2648,0.6445,0.7162)	7	0.1120

4 Conclusion

In this paper, we study the numerical behaviors of the smoothing Newton method for solving tensor eigenvalue complementarity problem (TEiCP). By introducing the smoothing approximation fuction of the Fischer-Burmeister NCP function, the tensor eigenvalue complementarity problem can be converted into an equivalent system of smooth equations. Then, a smoothing Newton method is investigated for TEiCP. Numerical experiments show that smoothing Newton method is efficient and competitive to some existing methods.

Acknowledgements

The authors would like to thank two anonymous referees for their comments and suggestions, which lead to great improvements of the presentation.

References

[1] B.W. Bader, T.G. Kolda and others. MATLAB Tensor Toolbox Version 2.6, Available online, February 2015. http://www.sandia.gov/ tgkolda/TensorToolbox/.

[2] B. Chen, X. Chen and C. Kanzow, A penalized Fischer-Burmeister NCP-function, *Math. Program*, 88 (2000) 211–216.

[3] X. Bai, Z. Huang and Y. Wang, Global uniqueness and solvability for tensor complementarity problems, *J. Optim. Theory Appl*, 170 (2016) 72–84.

[4] D.P. Bertsekas, *Constrained Optimization and Largrange Multiplier Methods*, Academic Press, New York,1982.

[5] Z. Chen and L. Qi, A semismooth Newton method for tensor eigenvalue complementarity problem, *Comput. Optim. Appl* 65 (2016) 109–126.

[6] M. Che, L. Qi, Y. Wei, Positive definite tensors to nonlinear complementarity problems, *J. Optim. Theory Appl* 168 (2016) 475–487.

[7] Z. Chen, Q. Yang and L. Ye, Generalized eigenvalue complementarity problem for tensors, *arXiv: 1505.02494v1*, May 2015.

[8] W. Ding, Z. Luo, L. Qi, P-tensors, P_0-tensors, and tensor complementarity problem,*arXiv:1507.06731*, July 2015.

[9] J. Fan, J. Nie and A. Zhou, Tensor eigenvalue complementarity problems, *arXiv:1601.05370v1*, Jan. 2016.

[10] M.S. Gowda, Z. Luo, L. Qi, N. Xiu, Z-tensors and complementarity problems, *arXiv:1510.07933*, Oct. 2015.

[11] T.G. Kolda and J.R. Mayo, An adaptive shifted power methods for computing generalized tensor eigenpairs, *SIAM J. Matrix Anal. Appl* 35 (2014) 1563–1581.

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

[13] J.-Y. Han, N.-H. Xiu, H.-D. Qi, *The Theory and Algorithms for the Nonlinear Complementarity Problems*, Shanghai Science and Technology Press, 2006.

[14] J. J. Judice, I. M. Ribeiro and H. D. Sherali, The eigenvalue complementarity problem, *Comput. Optim. Appl*, 37 (2007) 139–156.

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

[16] C. Ling, H. He and L. Qi, On the cone eigenvalue complementarity problem for higher-order tensors, *Comput. Optim. Appl* 63 (2016) 143–168.

[17] C. Ling, H. He and L. Qi, Higher-degree eigenvalue complementarity problem for tensors, *Comput. Optim. Appl*, 64 (2016) 149–176.

[18] Z. Luo, L. Qi, N. Xiu, The sparsest solutions to Z-tensor complementarity problems, *Optim. Lett* 11 (2017) 471–482.

[19] M. Ng, L. Qi and G. Zhou, Finding the largest eigenvalue of a nonnegative tensor, *SIAM J. Matrix Anal. Appl* 31 (2009) 1090–1099.

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

[21] Y. Song and L. Qi, Eigenvalue analysis of constrained minimization problem for homogeneous polynomial, *J. Global Optim* 64 (2016) 563–575.

[22] Y. Song and L. Qi, Tensor complementarity problem and semi-positive tensors, *J. Optim. Theory Appl* 169 (2016) 1069–1078.

[23] Y. Wang, Z. Huang, X. Bai, Exceptionally regular tensors and tensor complementarity problems, *Optim. Methods Softw* 31 (2016) 815–828.

[24] L. Zhang, J. Han, Z. Huang, Superlinear/quadratic one-step smoothing Newton method for P0-NCP, *Acta Math. Sinica* 26 (2005) 117–128.

Manuscript received 28 Augusut 2016
revised 15 March 2016
accepted for publication 4 April 2017

Wenyu Hu
School of Mathematics and Computer Sciences
Gannan Normal University, Ganzhou, 341000, China
E–mail address: 18270746281@163.com

Laishui Lu
School of Mathematics and Computer Sciences
Gannan Normal University, Ganzhou, 341000, China
E–mail address: malslv@163.com

Cheng Yin
School of Science, Jiangxi University of Science and Technology
Ganzhou, 341000, China
E–mail address: 83623729@qq.com

Gaohang Yu
School of Mathematics and Computer Sciences
Gannan Normal University, Ganzhou, 341000, China
E–mail address: maghyu@163.com