Bull. Aust. Math. Soc. **94** (2016), 411–420
doi:10.1017/S0004972716000381

A MODIFIED FR CONJUGATE GRADIENT METHOD FOR COMPUTING Z-EIGENPAIRS OF SYMMETRIC TENSORS

MEILAN ZENG and GUANGHUI ZHOU✉

(Received 31 December 2015; accepted 7 April 2016; first published online 26 July 2016)

Abstract

This paper proposes improvements to the modified Fletcher–Reeves conjugate gradient method (FR-CGM) for computing Z-eigenpairs of symmetric tensors. The FR-CGM does not need to compute the exact gradient and Jacobian. The global convergence of this method is established. We also test other conjugate gradient methods such as the modified Polak–Ribière–Polyak conjugate gradient method (PRP-CGM) and shifted power method (SS-HOPM). Numerical experiments of FR-CGM, PRP-CGM and SS-HOPM show the efficiency of the proposed method for finding Z-eigenpairs of symmetric tensors.

2010 *Mathematics subject classification*: primary 15A18; secondary 65H10.

Keywords and phrases: conjugate gradient method, Z-eigenpair, symmetric tensor.

1. Introduction

Let $\mathbb{R}$ be the real field and let m and n be positive integers. An m-order n-dimensional tensor $\mathcal{A}$ is an array indexed by integer tuples $(i_1, \ldots, i_m)$ with $1 \le i_j \le n$. Let $\mathbb{R}^{[m,n]}$ denote the set of all m-order n-dimensional real tensors and represent $\mathcal{A} \in \mathbb{R}^{[m,n]}$ as

$$\mathcal{A} = (a_{i_1 \cdots i_m}), \quad a_{i_1 \cdots i_m} \in \mathbb{R}, 1 \le i_1, \ldots, i_m \le n.$$

A tensor is called symmetric if the value of $a_{i_1 \cdots i_m}$ is invariant under any permutation of its index $(i_1, \ldots, i_m)$. In this paper, all the tensors are real tensors.

To any n-vector $x = (x_1, \ldots, x_n)^T$, real or complex, we define an n-dimensional column vector

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

Tensor eigenvalues and eigenvectors have received much attention (see [1, 8, 18, 20]). The tensor eigen problem has applications in blind source separation [10], magnetic resonance imaging [23], higher-order Markov chains [15] and spectral

This work is supported by the Science and Technology Foundation of the Department of Education of Hubei Province (D20152701) and the Foundation of Education Department of Anhui Province (KJ2016A651, 2014jyxm161).

hypergraph theory [2, 9, 16]. There is more than one possible definition for a tensor eigenpair [19, 22]. In this paper, we use the following definition.

DEFINITION 1.1. Let $\mathcal{A} \in \mathbb{R}^{[m,n]}$. The pair $(\lambda, x) \in \mathbb{C} \times (\mathbb{C}^n \setminus \{0\})$ is called an E-eigenpair, where λ is the E-eigenvalue and x is the associated E-eigenvector of $\mathcal{A}$, if they satisfy the equations

$$\mathcal{A}x^{m-1} = \lambda x \quad \text{and} \quad x^T x = 1. \tag{1.1}$$

We call (λ, x) a Z-eigenpair if they are both real.

It is NP-hard to compute eigenvalues of higher-order tensors (that is, for $m \geq 3$). There is much recent work on calculating Z-eigenvalues of symmetric tensors. Qi *et al.* [21] proposed an elimination method for finding all the Z-eigenvalues, which is specific to third-order tensors. Kolda and Mayo [11, 12] provided a shifted power method (SS-HOPM) for computing Z-eigenpairs, but the choice of a suitable shift parameter may be crucial. More recently, Han [5] proposed an unconstrained optimisation approach for even-order symmetric tensors. Hu *et al.* [7] proposed a sequential semidefinite programming method for finding the extreme Z-eigenvalues of even-order symmetric tensors. The last two methods usually find one or two Z-eigenvalues of even-order symmetric tensors. Cui *et al.* [3] use Jacobian SDP relaxations in polynomial optimisation to compute all real eigenvalues of symmetric tensors sequentially. The computational complexity of this method increases rapidly with increasing relaxation order.

The aim of this paper is to provide a simple method for computing Z-eigenpairs of symmetric tensors. The equations (1.1) are regarded as a system of nonlinear equations with respect to the variables (x, λ). It is easy to see that the Jacobian of the system is symmetric for a symmetric tensor. We improve the line search technique of the modified FR-CGM proposed by Li and Wang [14] to symmetric nonlinear equations and propose a modified FR-CGM for calculating Z-eigenpairs of symmetric tensors. The FR-CGM need not compute the exact gradient and Jacobian. The global convergence of this method is established. We also test other conjugate gradient methods such as a modified PRP-CGM [25], and present numerical experiments of the FR-CGM, PRP-CGM and SS-HOPM for computing Z-eigenpairs of symmetric tensors. The results show that the proposed method is promising.

The paper is organised as follows. In Section 2, we describe the FR-CGM for finding Z-eigenpairs of symmetric tensors. In Section 3, the convergence theory of our algorithm is presented. Preliminary numerical results are demonstrated on test problems in Section 4.

2. An FR-CGM

In this section, we give a modified FR-CGM for computing Z-eigenpairs of symmetric tensors. Let $\mathcal{A} \in \mathbb{R}^{[m,n]}$ and consider the equation

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

Then $F : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ is a nonlinear continuously differentiable function. The tensor eigenvalue equation (1.1) for $m > 2$ amounts to a system of nonlinear equations (2.1). Any real solution of (2.1) is a Z-eigenpair of the symmetric tensor $\mathcal{A}$.

LEMMA 2.1 [17, 24]. *If $\mathcal{A} \in \mathbb{R}^{[m,n]}$ is a symmetric tensor, then*

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

is a symmetric matrix.

Since the Jacobian $F'(x, \lambda)$ is symmetric, (2.1) is a symmetric nonlinear problem. The symmetric nonlinear problem (2.1) has been studied by several authors. Li and Fukushima [13] proposed a globally and superlinearly convergent Gauss–Newton based BFGS method for such problems. Gu *et al.* [4] extended this method to the norm descent case. Li and Wang [14] introduced a modified FR-CGM for symmetric nonlinear equations.

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

$$\min \varphi(\omega), \qquad \omega \in \mathbb{R}^{n+1}. \tag{2.2}$$

Set $F_k = F(\omega_k)$, $J_k = F'(\omega_k)$ and

$$g_k(t) = (F(\omega_k + tF(\omega_k)) - F(\omega_k))/t, \tag{2.3}$$

so that $\lim_{t\to 0} g_k(t) = J_k F_k = \nabla\varphi(\omega_k)$. The modified FR-CGM in [14] for symmetric nonlinear equations defines $d_k(t)$ with parameter t as

$$d_k(t) = \begin{cases} -g_k(t) & \text{if } k = 0, \\ -\alpha_k(t) g_k(t) + \beta_k(t) d_{k-1} & \text{if } k \geq 1, \end{cases} \tag{2.4}$$

$$\alpha_k(t) = 1 + \frac{g_k(t)^T d_{k-1}}{\|g_{k-1}\|^2}, \quad \beta_k(t) = \frac{\|g_k(t)\|^2}{\|g_{k-1}\|^2}, \tag{2.5}$$

where g_{k-1} is an estimation to $\nabla\varphi(\omega_{k-1})$. By direct computation,

$$g_k(t)^T d_k(t) = -\|g_k(t)\|^2.$$

Procedures 1 and 2, below, provide a way of determining d_k and t_k.

PROCEDURE 1. Let $\sigma_1 > 0, \sigma_2 > 0, \sigma_3, r \in (0, 1)$ be constant. Let i_k be the smallest nonnegative integer such that the inequality

$$\begin{aligned} \varphi(\omega_k + t d_k(t)) - \varphi(\omega_k) \leq -\sigma_1\|t d_k(t)\|^2 - \sigma_2\|tF(\omega_k)\|^2 \\ + \sigma_3(F(\omega_k + tF(\omega_k)) - F(\omega_k))^T d_k(t) \end{aligned}$$

holds with $t = r^i, i = 0, 1, 2, \ldots, i_k$. Let $g_k = g_k(r^{i_k}), d_k = d_k(r^{i_k})$.

PROCEDURE 2. Let i_k and d_k be determined by Procedure 1. If $i_k = 0$, let $t_k = 1$. Otherwise, let j_k be the largest positive integer $j = \{0, 1, 2, \ldots, i_k - 1\}$ satisfying

$$\varphi(\omega_k + r^{i_k-j}d_k) - \varphi(\omega_k) \le -\sigma_1\|r^{i_k-j}d_k\|^2 - \sigma_2\|r^{i_k-j}F(\omega_k)\|^2 + \sigma_3(F(\omega_k + r^{i_k-j}F(\omega_k)) - F(\omega_k))^T d_k.$$

Let $t_k = r^{i_k-j_k}$.

We improve the line search technique of the above modified FR-CGM and use it to calculate Z-eigenpairs of symmetric tensors. To this end, we give the following lemma which shows that, for $t > 0$ sufficiently small, every solution of (2.4) is a descent direction of φ at ω_k.

LEMMA 2.2. *Let σ_1 and σ_2 be positive constants. If ω_k is not a stationary point of (2.2), then there exists a constant $\bar{t} > 0$ depending on k such that, for $t \in (0, \bar{t})$, the solution $d_k(t)$ of (2.4) satisfies*

$$\nabla\varphi(\omega_k)^T d_k(t) < 0. \tag{2.6}$$

Moreover, the inequality

$$\varphi(\omega_k + td_k(t)) - \varphi(\omega_k) \le -\sigma_1\|td_k(t)\|^2 - \sigma_2\|tF(\omega_k)\|^2 \tag{2.7}$$

holds for all $t > 0$ sufficiently small.

PROOF. By direct computation, we get

$$\lim_{t\to 0^+} \nabla\varphi(\omega_k)^T d_k(t) = -\|\nabla\varphi(\omega_k)\|^2 < 0.$$

Therefore, inequality (2.6) holds for all $t > 0$ sufficiently small. Next

$$\lim_{t\to 0^+} (\varphi(\omega_k + td_k(t)) - \varphi(\omega_k))/t = \lim_{t\to 0^+} \nabla\varphi(\omega_k)^T d_k(t) < 0.$$

However, the right-hand side of (2.7) is $o(t)$. Therefore, inequality (2.7) holds for all $t > 0$ sufficiently small. □

The following procedure gives a way of determining a search direction d_k and a step size t_k, simultaneously.

PROCEDURE 3. Let $\sigma_1 > 0, \sigma_2 > 0, r \in (0, 1)$ be constant. Let $t_k = \max\{1, r, r^2, \ldots\}$ satisfying

$$\varphi(\omega_k + t_k d_k) - \varphi(\omega_k) \le -\sigma_1\|t_k d_k\|^2 - \sigma_2\|t_k F(\omega_k)\|^2. \tag{2.8}$$

Then let $g_k = g_k(t_k), d_k = d_k(t_k)$.

Procedure 3 ensures that the value of φ at $\omega_k + t_k d_k$ is less than that of φ at ω_k.

Now we present a modified FR-CGM for computing Z-eigenpairs of symmetric tensors. The algorithm is given as follows.

Algorithm FR-CGM
Initial. Choose $\epsilon > 0, \omega_0 \in R^{n+1}$. Set $k := 0$.
Step 1. Evaluate F_k. If $\|F_k\| \le \epsilon$, terminate.
Step 2. Determine t_k and d_k by (2.3)–(2.5) and Procedure 3.
Step 3. Set $\omega_{k+1} = \omega_k + t_k d_k$. Let $k := k+1$ and go to Step 1.

As in the proof of [24, Lemma 2.2], we can derive the following lemma from Algorithm FR-CGM.

LEMMA 2.3. *The sequence $\{\varphi(\omega_k)\}$ is strictly decreasing. If $s_k = \omega_{k+1} - \omega_k = t_k d_k$, then*

$$\lim_{k\to\infty} \|s_k\| = 0, \quad \lim_{k\to\infty} \|t_k F_k\| = 0. \tag{2.9}$$

3. Convergence analysis

In this section, we prove the global convergence of Algorithm FR-CGM. In order to prove the global convergence, we make the following assumption.

ASSUMPTION 3.1. The level set $\Omega = \{\omega \in R^{n+1} \mid \varphi(\omega) \le \varphi(\omega_0)\}$ is bounded.

Assumption 3.1 shows that there exist positive constants M_1, M_2 such that

$$\|F(x)\| \le M_1, \quad \|J(x)\| \le M_2, \quad \forall x \in N.$$

THEOREM 3.2. *Let Assumption 3.1 hold and $\{\omega_k\}$ be generated by Algorithm FR-CGM. Then*

$$\liminf_{k\to\infty} \|\nabla\varphi(\omega_k)\| = 0. \tag{3.1}$$

PROOF. If $\limsup_{k\to\infty} \alpha_k > 0$, then, from (2.9), it is easy to see that

$$\liminf_{k\to\infty} \|F_k\| = 0,$$

which implies (3.1).

We need only to show (3.1) for the case $\lim_{k\to\infty} \alpha_k = 0$. We do it by assuming that

$$\liminf_{k\to\infty} \|\nabla\varphi(\omega_k)\| > 0, \tag{3.2}$$

to deduce a contradiction. Suppose that (3.2) holds. Then there is a constant $\eta_1 > 0$ such that $\|F(\omega_k)\| \ge \eta_1$ for all k. Since $\{\omega_k\} \subset \Omega$ is bounded, it is clear that the sequence $\{d_k\}$ is bounded. Then there exists a set K of nonnegative integers and subsequences $\{\omega_k\}_{k\in K}$ and $\{d_k\}_{k\in K}$, respectively, that converge to ω^* and d^*. From

$$\lim_{t\to 0} g_k(t) = J_k F_k = \nabla\varphi(\omega_k),$$

we get

$$\lim_{k\in K, k\to\infty} g_k = \nabla\varphi(\omega^*).$$

https://doi.org/10.1017/S0004972716000381 Published online by Cambridge University Press

Since $\lim_{k\to\infty} t_k = 0$, we see that $t'_k = t_k/r$ does not satisfy (2.8), namely,

$$\varphi(\omega_k + t'_k d_k) - \varphi(\omega_k) > -\sigma_1 \|t'_k d_k\|^2 - \sigma_2 \|t'_k F_k\|^2.$$

Dividing both sides by α'_k and then taking limits as $k \to \infty$ with $k \in K$, it follows that

$$\nabla\varphi(\omega^*)^T d^* \geq 0. \tag{3.3}$$

Since $g_k(t)^T d_k(t) = -\|g_k(t)\|^2$, $\lim_{t\to 0^+} g_k(t)^T d_k(t) = -\|\nabla\varphi(\omega_k)\|^2$ and

$$\lim_{k\in K, k\to\infty} g_k^T d_k = -\|\nabla\varphi(\omega^*)\|^2.$$

From

$$\lim_{k\in K, k\to\infty} g_k = \nabla\varphi(\omega^*), \qquad \lim_{k\in K, k\to\infty} d_k = d^*$$

and (3.3) we get

$$-\|\nabla\varphi(\omega^*)\|^2 \geq 0,$$

which implies that

$$\|\nabla\varphi(\omega^*)\| = 0.$$

But this contradicts (3.2). This completes the proof. □

Theorem 3.2 shows that the iterative sequence $\{\omega_k\}$ has an accumulation point which is a stationary point of the problem $\min \varphi(\omega) = \frac{1}{2}\|F(\omega)\|^2$. It may not be a solution of (2.1) if the Jacobian matrix $J(\omega)$ is singular at that point.

As in the proof of [24, Theorem 3.3], it is easy to derive the following theorem from Algorithm FR-CGM, showing that Algorithm FR-CGM is globally convergent.

THEOREM 3.3. *Let Assumption 3.1 hold. Suppose that the sequence $\{\omega_k\}$ generated by Algorithm FR-CGM has a subsequence converging to a stationary point ω^* at which $J(\omega^*)$ is nonsingular. Then ω^* is a solution of (2.1), that is, (x^*, λ^*) is a Z-eigenpair of the symmetric tensor $\mathcal{A}$. Moreover, the whole sequence $\{\omega_k\}$ converges to ω^*.*

4. Numerical experiments

In this section, we report some numerical experiments of the proposed algorithm (FR-CGM) for computing Z-eigenpairs of symmetric tensors. In addition, we also give the comparative numerical results for the FR-CGM, PRP-CGM and SS-HOPM for Examples 4.1 and 4.2.

The parameters are specified as follows. We take $r = 0.5$ in Procedure 3 and $\sigma_1 = \sigma_2 = 10^{-5}$ in (2.8). We stop the iteration process if $\|F(\omega_k)\| \leq 10^{-4}$. We also stop the program if the iteration number is larger than 10 000. The program is coded in MATLAB 7.8. For any odd-order tensor (that is, m odd), if (x, λ) is an eigenpair, then $(-x, -\lambda)$ is also an eigenpair. For any even-order tensor (that is, m even), if (x, λ) is an eigenpair, then $(-x, \lambda)$ is also an eigenpair. Therefore, we only give their positive Z-eigenvalues and the corresponding Z-eigenvectors for odd-order tensors. *Iters* is the total number of iterations and *Time* is the CPU time in seconds.

https://doi.org/10.1017/S0004972716000381 Published online by Cambridge University Press

TABLE 4.1. Numerical results of the FR-CGM and the PRP-CGM for Example 4.1.

FR-CGM						PRP-CGM					
λ	x^T			*Iters*	*Time*	λ	x^T			*Iters*	*Time*
0.8730	[−0.3921	0.7249	0.5663]	13	0.0032	0.8729	[−0.3921	0.7248	0.5663]	12	0.0232
0.4306	[−0.7187	−0.1244	−0.6841]	10	0.0015	—					
0.2294	[−0.8448	0.4384	−0.3068]	14	0.0019	0.2294	[−0.8448	0.4384	−0.3068]	38	0.0067
0.0180	[0.7127	0.5097	−0.4820]	29	0.0039	0.0180	[0.7126	0.5097	−0.4820]	169	0.0151
0.0033	[0.4484	0.7740	−0.4471]	27	0.0032	0.0033	[0.4483	0.7741	−0.4470]	107	0.0119
0.0018	[0.3309	0.6309	−0.7018]	34	0.0067	0.0018	[0.3305	0.6308	−0.7020]	178	0.0208
0.0006	[0.2899	0.7359	−0.6119]	42	0.0058	0.0006	[0.2906	0.7354	−0.6122]	226	0.0176

TABLE 4.2. Numerical results of the FR-CGM and the SS-HOPM for Example 4.1.

FR-CGM						SS-HOPM					
λ	x^T			*Iters*	*Time*	λ	x^T			*Iters*	*Time*
0.8730	[−0.3921	0.7249	0.5663]	13	0.0032	0.8730	[−0.3922	0.7249	0.5664]	39	0.0018
0.4306	[−0.7187	−0.1244	−0.6841]	10	0.0015	0.4306	[−0.7187	−0.1245	−0.6840]	74	0.0014
0.2294	[−0.8448	0.4384	−0.3068]	14	0.0019	0.4306	[−0.7187	−0.1245	−0.6840]	92	0.0036
0.0180	[0.7127	0.5097	−0.4820]	29	0.0039	0.0180	[0.7132	0.5093	−0.4817]	179	0.0067
0.0033	[0.4484	0.7740	−0.4471]	27	0.0032	0.8730	[0.3922	0.7249	0.5664]	70	0.0031
0.0018	[0.3309	0.6309	−0.7018]	34	0.0067	0.4306	[−0.7187	−0.1245	−0.6840]	123	0.0049
0.0006	[0.2899	0.7359	−0.6119]	42	0.0058	0.0180	[0.7132	0.5093	−0.4817]	282	0.0106

EXAMPLE 4.1 [11, Example 3.6]. Consider an odd-order symmetric tensor $\mathcal{A} \in R^{[3,3]}$ defined by

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

From [11, Theorem 5.3], $\mathcal{A}$ has at most seven Z-eigenpairs. Under the same initial conditions, the FR-CGM found all the Z-eigenpairs, the PRP-CGM found six Z-eigenpairs, while the SS-HOMP only found three Z-eigenpairs with the shift parameter $\alpha = 1$. The numerical results are shown in Tables 4.1 and 4.2.

EXAMPLE 4.2 [11, Example 3.5]. Let $\mathcal{A} \in \mathbb{R}^{[4,3]}$ be an even-order symmetric tensor defined by

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

From [11, Theorem 5.3], this problem has at most 13 E-eigenpairs. In fact, $\mathcal{A}$ has 11 Z-eigenpairs. Under the same initial conditions, the FR-CGM found all the Z-eigenpairs. However, the PRP-CGM only found three Z-eigenpairs. The SS-HOMP also found three Z-eigenpairs with the shift parameter $\alpha = 2$. The numerical results are shown in Tables 4.3 and 4.4.

https://doi.org/10.1017/S0004972716000381 Published online by Cambridge University Press

TABLE 4.3. Numerical results of the FR-CGM and the PRP-CGM for Example 4.2.

FR-CGM					PRP-CGM			
λ	x^T		*Iters*	*Time*	λ	x^T	*Iters*	*Time*
0.8895	[0.6672	0.2470 −0.7028]	70	0.0124	—			
0.8170	[0.8412	−0.2636 0.4722]	21	0.0095	—			
0.5104	[0.3597	−0.7780 0.5150]	23	0.0062	—			
0.3633	[0.2676	0.6447 0.7160]	10	0.0017	0.3633	[0.2675 0.6447 0.7161]	12	0.0019
0.2683	[0.6100	0.4362 0.6616]	6	0.0012	0.2682	[0.6099 0.4363 0.6617]	8	0.0016
0.2628	[0.1318	−0.4425 −0.8870]	20	0.0038	—			
0.2433	[0.9895	0.0946 −0.1088]	10	0.0017	—			
0.1735	[0.3357	0.9073 0.2531]	6	0.0012	0.1734	[0.3357 0.9073 0.2531]	8	0.0015
−0.0452	[0.7797	0.6135 0.1250]	6	0.0013	—			
−0.5630	[0.1761	−0.1796 0.9679]	21	0.0077	—			
−1.0955	[0.5915	−0.7467 −0.3043]	42	0.0098	—			

TABLE 4.4. Numerical results of the FR-CGM and the SS-HOPM for Example 4.2.

FR-CGM				SS-HOPM			
λ	x^T	*Iters*	*Time*	λ	x^T	*Iters*	*Time*
0.8895	[0.6672 0.2470 −0.7028]	70	0.0124	0.8893	[0.6672 0.2471 −0.7027]	44	0.0158
0.8170	[0.8412 −0.2636 0.4722]	21	0.0095	0.8169	[0.8412 −0.2635 0.4722]	41	0.0025
0.5104	[0.3597 −0.7780 0.5150]	23	0.0062	0.8893	[−0.6672 −0.2471 0.7027]	158	0.0078
0.3633	[0.2676 0.6447 0.7160]	10	0.0017	0.3633	[0.2676 0.6447 0.7160]	55	0.0030
0.2683	[0.6100 0.4362 0.6616]	6	0.0012	0.8169	[0.8412 −0.2635 0.4722]	63	0.0034
0.2628	[0.1318 −0.4425 −0.8870]	20	0.0038	0.3633	[−0.2676 −0.6447 −0.7160]	76	0.0033
0.2433	[0.9895 0.0946 −0.1088]	10	0.0017	0.8893	[0.6672 0.2471 −0.7027]	54	0.0016
0.1735	[0.3357 0.9073 0.2531]	6	0.0012	0.8893	[0.6672 0.2471 −0.7027]	130	0.0064
−0.0452	[0.7797 0.6135 0.1250]	6	0.0013	0.8893	[0.6672 0.2471 −0.7027]	83	0.0043
−0.5630	[0.1761 −0.1796 0.9679]	21	0.0077	0.8169	[0.8412 −0.2635 0.4722]	49	0.0014
−1.0955	[0.5915 −0.7467 −0.3043]	42	0.0098	0.8893	[0.6672 0.2471 −0.7027]	50	0.0014

TABLE 4.5. Z-eigenpairs for $\mathcal{A} \in \mathbb{R}^{[3,5]}$ from Example 4.3.

λ	x^T				
9.9972	[−0.7312	−0.1375	−0.4674	−0.2365	−0.4146]
4.2872	[−0.1858	0.7158	0.2149	0.5655	0.2950]
0.0000	[0.5213	−0.1043	0.4170	−0.7298	−0.1043]

EXAMPLE 4.3 [3, Example 4.11]. Consider the odd-order symmetric tensor $\mathcal{A} \in \mathbb{R}^{[3,5]}$ such that

$$a_{ijk} = \frac{(-1)^i}{i} + \frac{(-1)^j}{j} + \frac{(-1)^k}{k}.$$

This problem has three Z-eigenpairs. Using the FR-CGM, we find all the Z-eigenpairs, which are listed in Table 4.5.

EXAMPLE 4.4 [3, Example 4.12]. Consider the even-order symmetric tensor $\mathcal{A} \in \mathbb{R}^{[4,5]}$ such that $a_{ijkl} = \sin(i + j + k + l)$. This problem has five Z-eigenpairs. Using the FR-CGM, we find all the Z-eigenpairs, which are listed in Table 4.6.

https://doi.org/10.1017/S0004972716000381 Published online by Cambridge University Press

TABLE 4.6. Z-eigenpairs for $\mathcal{A} \in \mathbb{R}^{[4,5]}$ from Example 4.4.

λ	x^T				
7.2591	[0.2686	0.6150	0.3959	−0.1872	−0.5982]
4.6410	[−0.5056	0.1228	0.6382	0.5669	−0.0256]
0.0000	[0.6294	0.3328	−0.0809	0.5550	0.4226]
−3.9207	[−0.1785	0.4847	0.7023	0.2742	−0.4060]
−8.8478	[−0.5810	−0.3563	0.1959	0.5680	0.4179]

5. Conclusions

The numerical results show that the FR-CGM is more effective than the PRP-CGM for computing Z-eigenpairs of symmetric tensors. For an m-order n-dimensional tensor $\mathcal{A}$, the computational complexity of each iteration of the FR-CGM is $O(n^m)$, which is the work of computing $\mathcal{A}x^{m-1}$. The computational complexity of each iteration of the SS-HOPM is also $O(n^m)$. The numerical results illustrate that the FR-CGM may compute all the Z-eigenpairs while the SS-HOPM may only find some Z-eigenpairs for a fixed shift parameter. The polynomial optimisation method [3], implemented by the software Gloptiploy 3 [6] and SEDUMI, can compute all the Z-eigenpairs of symmetric tensors. However, the calculation and implementation of the polynomial optimisation method are complex, while our method is simple.

References

[1] K.-C. Chang and T. Zhang, 'On the uniqueness and non-uniqueness of the positive z-eigenvector for transition probability tensors', *J. Math. Anal. Appl.* **408**(2) (2013), 525–540.

[2] J. Cooper and A. Dutle, 'Spectra of uniform hypergraphs', *Linear Algebra Appl.* **436**(9) (2012), 3268–3292.

[3] C. Cui, Y. Dai and J. Nie, 'All real eigenvalues of symmetric tensors', *SIAM J. Matrix Anal. Appl.* **35**(4) (2014), 1582–1601.

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

[5] L. Han, 'An unconstrained optimization approach for finding real eigenvalues of even order symmetric tensors', *Numer. Algebra Control Optim.* **3**(3) (2012), 583–599.

[6] D. Henrion, J.-B. Lasserre and J. Lofberg, 'Gloptipoly 3: moments, optimization and semidefinite programming', *Optim. Methods Softw.* **24**(4–5) (2009), 761–779.

[7] S. Hu, Z. Huang and L. Qi, 'Finding the extreme z-eigenvalues of tensors via a sequential semidefinite programming method', *Numer. Linear Algebra Appl.* **20**(6) (2013), 972–984.

[8] S. Hu, G. Li, L. Qi and Y. Song, 'Finding the maximum eigenvalue of essentially nonnegative symmetric tensors via sum of squares programming', *J. Optim. Theory Appl.* **158**(3) (2013), 717–738.

[9] S. Hu and L. Qi, 'Algebraic connectivity of an even uniform hypergraph', *J. Comb. Optim.* **24**(4) (2012), 564–579.

[10] E. Kofidis and P. A. Regalia, 'On the best rank-1 approximation of higher-order supersymmetric tensors', *SIAM J. Matrix Anal. Appl.* **23**(3) (2002), 863–884.

[11] T. G. Kolda and J. R. Mayo., 'Shifted power method for computing tensor eigenpairs', *SIAM J. Matrix Anal. Appl.* **32**(4) (2011), 1095–1124.

[12] T. G. Kolda and J. R. Mayo, 'An adaptive shifted power method for computing generalized tensor eigenpairs', *SIAM J. Matrix Anal. Appl.* **35**(4) (2014), 1563–1581.

[13] D. Li and M. Fukushima, 'A globally and superlinearly convergent Gauss–Newton-based BFGS method for symmetric nonlinear equations', *SIAM J. Numer. Anal.* **37**(1) (1999), 152–172.
[14] D. Li and X. Wang, 'A modified Fletcher–Reeves-type derivative-free method for symmetric nonlinear equations', *Numer. Algebra Control Optim.* **1**(1) (2011), 71–82.
[15] X. Li, M. Ng and Y. Ye, 'Finding stationary probability vector of a transition probability tensor arising from a higher-order Markov chain', *Matrix* **1** (2011), 1.
[16] G. Li, L. Qi and Y. Yu, 'The z-eigenvalues of a symmetric tensor and its application to spectral hypergraph theory', *Numer. Linear Algebra Appl.* **20**(6) (2013), 1001–1029.
[17] Q. Ni and L. Qi, 'A quadratically convergent algorithm for finding the largest eigenvalue of a nonnegative homogeneous polynomial map', *J. Global Optim.* **61**(4) (2015), 627–641.
[18] Q. Ni, L. Qi and F. Wang, 'An eigenvalue method for the positive definiteness identification problem', *IEEE Trans. Automat. Control* **53** (2008), 1096–1107.
[19] L. Qi, 'Eigenvalues of a real supersymmetric tensor', *J. Symbolic Comput.* **40**(6) (2005), 1302–1324.
[20] L. Qi, 'Rank and eigenvalues of a supersymmetric tensor, the multivariate homogeneous polynomial and the algebraic hypersurface it defines', *J. Symbolic Comput.* **41**(12) (2006), 1309–1327.
[21] L. Qi, F. Wang and Y. Wang, 'Z-eigenvalue methods for a global polynomial optimization problem', *Math. Program.* **118**(2) (2009), 301–316.
[22] L. Qi, Y. Wang and E. X. Wu, 'D-eigenvalues of diffusion kurtosis tensors', *J. Comput. Appl. Math.* **221**(1) (2008), 150–157.
[23] L. Qi, G. Yu and E. X. Wu, 'Higher order positive semidefinite diffusion tensor imaging', *SIAM J. Imaging Sci.* **3**(3) (2010), 416–433.
[24] M. Zeng and Q. Ni, 'Quasi-Newton method for computing z-eigenpairs of a symmetric tensor', *Pac. J. Optim.* **11**(2) (2015), 279–290.
[25] W. Zhou and D. Shen, 'An inexact PRP conjugate gradient method for symmetric nonlinear equations', *Numer. Funct. Anal. Optim.* **35**(3) (2014), 370–388.

MEILAN ZENG, Nanjing University of Aeronautics and Astronautics,
Nanjing, Jiangsu 210016, China
and
Hubei Engineering University, Xiaogan, Hubei 432000, China
e-mail: zml571@126.com

GUANGHUI ZHOU, Nanjing University of Aeronautics and Astronautics,
Nanjing, Jiangsu 210016, China
and
School of Mathematical Sciences, Information College,
Huaibei Normal University, Huaibei, Anhui 235000, China
e-mail: 163zgh@163.com