

A numerical method for polynomial eigenvalue problems using contour integral

Junko Asakura^a Tetsuya Sakurai^b Hiroto Tadano^b
Tsutomu Ikegami^c Kinji Kimura^d

^a*Graduate School of Systems and Information Engineering, University of Tsukuba*

^b*Department of Computer Science, University of Tsukuba*

^c*Grid Technology Research Center, AIST*

^d*Department of Applied Mathematics and Physics, Kyoto University*

^aE-mail: asakura@mma.cs.tsukuba.ac.jp

Abstract

We propose a numerical method using contour integral to solve polynomial eigenvalue problems (PEPs). The method finds eigenvalues contained in a certain domain which is defined by a surrounding integral path. By evaluating the contour integral numerically along the path, the method reduces the original PEP into a small generalized eigenvalue problem, which has the identical eigenvalues in the domain. Error analysis indicates that the error of the eigenvalues is not uniform: inner eigenvalues are less erroneous. Four numerical examples are presented, which confirm the theoretical predictions.

Keywords: polynomial eigenvalue problem, matrix polynomial, contour integral, projection method

1 Introduction

The present paper is concerned with a numerical method to solve polynomial eigenvalue problems. The polynomial eigenvalue problem (PEP) [1,4,7] involves finding an eigenvalue λ and corresponding nonzero eigenvector $\mathbf{x}$ that satisfy $F(\lambda)\mathbf{x} = \mathbf{0}$, where $F(\lambda) = \sum_{i=0}^l \lambda^i A_i$ with real or complex coefficient matrices. The eigenvalue problem of the matrix polynomial is considered to be a generalization of linear eigenvalue problems.

Polynomial eigenvalue problems can be used in a variety of problems in science and engineering. For example, quadratic eigenvalue problems arise in oscillation

analysis with damping [9, 18] and stability problems in fluid dynamics [8], and the three-dimensional (3D) Schrödinger equation can result in a cubic eigenvalue problem [10]. Similarly, the study of higher-order systems of differential equations leads to a matrix polynomial of degree greater than one [5]. However, its applications are more complicated than standard and generalized eigenvalue problems.

One reason is in the difficulty in solving the PEPs. Polynomial eigenvalue problems are typically solved by linearization [6, 13, 15], which promotes the k -th order $n \times n$ matrix polynomial into the larger $kn \times kn$ linear eigenvalue problem. Other methods, such as Arnoldi shift and invert strategy [2], can be used when several eigenvalues are desired. A disadvantage of the shift-and-invert Arnoldi methods is that a change of the shift parameter requires a new Krylov subspace to be built. Another approach is a direct solution obtained by means of the Jacobi-Davidson method [17], although this method has been investigated far less extensively.

We herein propose an eigensolver, which prevents the inflation of the matrix dimension. Indeed, we can even reduce the dimension of the problem by focusing on only the eigenvalues of physical interest. For linear problems, the Sakurai-Sugiura (SS) method [16] can find certain eigenvalues in a given domain, is capable to solve non-Hermitian systems, and is suitable to modern distributed parallel computers. In the present paper, we propose a solution of polynomial eigenvalue problems by means of the SS method. The proposed method enables us to obtain the eigenvalues of the matrix polynomial by solving the generalized eigenvalue problem, which is derived by solving systems of linear equations. Since these linear systems are independent for each equation, they can be solved in parallel. The proposed method is discussed from the theoretical point of view. In order to give the theoretical proof of the present SS method, we have redefined PEP in terms of the Smith form which is sometimes called the canonical form for polynomial matrix. In addition, some results of numerical examples are reported to confirm our theoretical observations.

The remainder of the paper is organized as follows. In the next section, we introduce the Smith form for matrix polynomials, which is sometimes called a canonical form, and its related results. In Section 3, we present the numerical method for solving the PEP by means of the SS method and present theoretical results for the proposed method. In Section 4, we present the algorithm of the SS method for the case in which the domain is given by a circle and the numerical integration is evaluated via the trapezoidal rule. We also provide the preliminary error analysis for the approximated eigenvalues of matrix polynomials evaluated by the SS method. Some numerical examples are shown in Section 5. Finally, conclusions and suggestions for future study are presented in Section 6.

2 Matrix polynomials

In this section, we first define the term matrix polynomial. Then, we introduce the Smith form for matrix polynomials, which is sometimes called a canonical form, and its related results.

Definition 2.1 (matrix polynomial) If $A_0, A_1, \dots, A_l \in \mathbb{C}^{n \times n}$ and $A_l \neq O$, then the matrix-valued function defined on $\mathbb{C}$ by

$$F(z) = \sum_{i=0}^l z^i A_i, \quad z \in \mathbb{C}$$

is called a matrix polynomial of degree l .

The matrix polynomial $F(z)$ is called regular when the determinant of $F(z)$ is not identically zero for all values of z .

Definition 2.2 (vector polynomial) If $\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_l \in \mathbb{C}^n$ and $\mathbf{u}_l \neq 0$, then the vector-valued function defined on $\mathbb{C}$ by

$$\mathbf{v}(z) = \sum_{i=0}^l z^i \mathbf{u}_i, \quad z \in \mathbb{C}$$

is called a vector polynomial of degree l .

Theorem 2.3 (the Smith form [7]) Let $F(z)$ be an $n \times n$ regular matrix polynomial. Then, $F(z)$ admits the representation

$$P(z)F(z)Q(z) = D(z), \quad (1)$$

where

$$D(z) = \begin{pmatrix} d_1(z) & 0 & \cdots & 0 \\ 0 & d_2(z) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_n(z) \end{pmatrix}$$

is a diagonal polynomial matrix with monic scalar polynomials $d_j(z)$ such that $d_j(z)$ are divisible by $d_{j-1}(z)$ for $j = 2, \dots, n$. In addition, $P(z)$ and $Q(z)$ are $n \times n$ matrix polynomials with constant nonzero determinants.

Representation (1), as well as the diagonal matrix $D(z)$, is called the Smith form of matrix polynomial $F(z)$. $D(z)$ is defined uniquely.

The eigenpairs of the PEP $F(\lambda)\mathbf{x} = 0$ is formally derived from the Smith form. Let $\mathbf{q}_j(z)$ be the column vectors of $Q(z)$,

$$Q(z) = (\mathbf{q}_1(z) \dots \mathbf{q}_n(z)), \quad (2)$$

and $\mathbf{p}_j(z)$ be

$$P(z)^H = (\mathbf{p}_1(z) \dots \mathbf{p}_n(z)). \quad (3)$$

Let λ_i be a set of zero points of $d_n(z)$, where $i = 1, \dots, \tilde{l} \leq ln$. Because $d_j(z)$ is divisible by $d_{j-1}(z)$, the invariant polynomial $d_j(z)$ is represented in terms of λ_i as

$$d_j(z) = \prod_{i=1}^{\tilde{l}} (z - \lambda_i)^{\alpha_{ji}}, \quad j = 1, \dots, n,$$

where $\alpha_{ji} \in \mathbb{Z}^+$ (non-negative integer) and $\alpha_{ji} \leq \alpha_{j'i}$ for $j < j'$ (see [7, p. 320]). The eigenpairs of the PEP is defined by the the zero-points of $d_j(z)$ and $\mathbf{q}_j(z)$.

Lemma 2.4 Let $\mathbf{q}_j(z)$ be a vector polynomial of (2), and λ_i be a zero point of $d_j(z)$. Then,

$$F(\lambda_i)\mathbf{q}_j(\lambda_i) = 0,$$

that is, $\mathbf{q}_j(\lambda_i)$ is the eigenvector of $F(z)$ corresponding to eigenvalue λ_i .

Proof. Because $P(z)$ and $Q(z)$ are invertible,

$$\begin{aligned} F(\lambda_i)\mathbf{q}_j(\lambda_i) &= P(\lambda_i)^{-1}D(\lambda_i)Q(\lambda_i)^{-1}(Q(\lambda_i)\mathbf{e}_j) \\ &= P(\lambda_i)^{-1}D(\lambda_i)\mathbf{e}_j \\ &= d_j(\lambda_i)P(\lambda_i)^{-1}\mathbf{e}_j. \end{aligned}$$

Since $d_j(\lambda_i) = 0$, we have the result of the lemma. $\square$

Note that if the eigenvalue λ_i is simple and not degenerated, i.e., λ_i is the simple zero of $\det F(z)$, the invariant polynomials satisfy $\alpha_{ji} = 0$ for $j \neq n$ and $\alpha_{ni} = 1$.

3 An eigensolver using contour integral

In this section, we propose an eigensolver using contour integral. This method enables us to obtain the eigenvalues of the matrix polynomial by solving a generalized eigenvalue problem derived in this section.

Let $F(z)$ be an $n \times n$ regular matrix polynomial of degree l , and let $\lambda_1, \dots, \lambda_l$ be finite eigenvalues of the matrix polynomial $F(z)$. For nonzero vector $\mathbf{u}, \mathbf{v}$, we define

$$f(z) := \mathbf{u}^H F(z)^{-1} \mathbf{v}.$$

The existence of the Smith form allows us to prove the following.

Theorem 3.1 Let $D(z) = \text{diag}(d_1(z), \dots, d_n(z))$ be the Smith form for $F(z)$, and let $P(z)$ and $Q(z)$ be defined by (1). Let $\chi_j(z) = \mathbf{u}^H \mathbf{q}_j(z) \mathbf{p}_j(z)^H \mathbf{v}$, $1 \leq j \leq n$. Then, $f(z)$ admits the representation

$$f(z) = \sum_{j=1}^n \frac{\chi_j(z)}{d_j(z)}. \quad (4)$$

Proof. It follows from (1) that

$$\begin{aligned} f(z) &= \mathbf{u}^H F(z)^{-1} \mathbf{v} \\ &= \mathbf{u}^H Q(z) \begin{pmatrix} d_1^{-1}(z) & 0 & \cdots & 0 \\ 0 & d_2^{-1}(z) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & d_n^{-1}(z) \end{pmatrix} P(z) \mathbf{v} \\ &= \sum_{j=1}^n \frac{\mathbf{u}^H \mathbf{q}_j(z) \mathbf{p}_j(z)^H \mathbf{v}}{d_j(z)} \\ &= \sum_{j=1}^n \frac{\chi_j(z)}{d_j(z)}. \end{aligned}$$

This proves the theorem. $\square$

Let Γ be a positively oriented closed Jordan curve in the complex plane. Let $\lambda_1, \dots, \lambda_m$ be distinct eigenvalues that lie in the interior of Γ . Assume that these eigenvalues are simple and not degenerated. Then we can suppose that $\alpha_{jl} = 0$ and $\alpha_{nl} = 1$ for $j = 1, \dots, n-1$ and $l = 1, \dots, m$.

Definition 3.2 For a non-negative integer k , the moment μ_k is defined as

$$\mu_k := \frac{1}{2\pi i} \int_{\Gamma} z^k f(z) dz, \quad k = 0, 1, \dots \quad (5)$$

Definition 3.3 Two $m \times m$ Hankel matrices $H_m^<$ and H_m are defined as

$$H_m := \begin{pmatrix} \mu_0 & \mu_1 & \cdots & \mu_{m-1} \\ \mu_1 & \mu_2 & \cdots & \mu_m \\ \vdots & \vdots & & \vdots \\ \mu_{m-1} & \mu_m & \cdots & \mu_{2m-2} \end{pmatrix}$$

and

$$H_m^< := \begin{pmatrix} \mu_1 & \mu_2 & \cdots & \mu_m \\ \mu_2 & \mu_3 & \cdots & \mu_{m+1} \\ \vdots & \vdots & & \vdots \\ \mu_m & \mu_{m+1} & \cdots & \mu_{2m-1} \end{pmatrix}.$$

The following theorem is one of the main result of the present paper.

Theorem 3.4 The eigenvalues of the pencil $H_m^< - \lambda H_m$ are given by $\lambda_1, \dots, \lambda_m$, if $\chi_n(\lambda_l) = \mathbf{u}^H \mathbf{q}_n(\lambda_l) \mathbf{p}_n(\lambda_l)^H \mathbf{v} \neq 0$ for $1 \leq l \leq m$.

Proof. Let $D(z) = \text{diag}(d_1(z), \dots, d_n(z))$ be the Smith form of $F(z)$. Then, from Theorem 3.1, we have

$$\begin{aligned} \mu_k &= \frac{1}{2\pi i} \int_{\Gamma} z^k f(z) dz \\ &= \sum_{j=1}^n \frac{1}{2\pi i} \int_{\Gamma} \frac{\chi_j(z)}{d_j(z)} z^k dz \\ &= \sum_{l=1}^m \text{Res} \left(\frac{\chi_n(z)}{d_n(z)} z^k, \lambda_l \right) \\ &= \sum_{l=1}^m \frac{\chi_n(\lambda_l)}{\prod_{\substack{i \neq l \\ i \leq m}} (\lambda_l - \lambda_i) \prod_{i > m} (\lambda_l - \lambda_i)^{\alpha_{ni}}} \lambda_l^k \\ &= \sum_{l=1}^m \nu_l \lambda_l^k. \end{aligned}$$

Here, ν_l are given as follows:

$$\nu_l = \frac{\chi_n(\lambda_l)}{\prod_{\substack{i \neq l \\ i \leq m}} (\lambda_l - \lambda_i) \prod_{i > m} (\lambda_l - \lambda_i)^{\alpha_{ni}}},$$

for $l = 1, \dots, m$.

Let V_m be the Vandermonde matrix

$$V_m := \begin{pmatrix} 1 & 1 & \dots & 1 \\ \lambda_1 & \lambda_2 & \dots & \lambda_m \\ \vdots & \vdots & & \vdots \\ \lambda_1^{m-1} & \lambda_2^{m-1} & \dots & \lambda_m^{m-1} \end{pmatrix}.$$

Let $D_m := \text{diag}(\nu_1, \dots, \nu_m)$, $\Lambda_m := \text{diag}(\lambda_1, \dots, \lambda_m)$. One can easily verify that

$$H_m = V_m D_m V_m^T, \quad H_m^< = V_m D_m \Lambda_m V_m^T,$$

therefore,

$$H_m^< - \lambda_l H_m = V_m D_m (\Lambda_m - \lambda_l I) V_m^T. \quad (6)$$

If $\chi_n(\lambda_l) \neq 0$ for $1 \leq l \leq m$, then $\nu_l \neq 0$ for $1 \leq l \leq m$. Since $\lambda_1, \dots, \lambda_m$ are distinct, V_m is nonsingular. Since D_m is also nonsingular, we have the result of the theorem. $\square$

Therefore, we can obtain the eigenvalues $\lambda_1, \dots, \lambda_m$ of the matrix polynomial $F(z)$ by solving the generalized eigenvalue problem $H_m^< \mathbf{x} = \lambda H_m \mathbf{x}$.

Next, we consider a method to evaluate eigenvectors. For the nonzero vector $\mathbf{v} \in \mathbb{C}^n$, let

$$\mathbf{s}_k := \frac{1}{2\pi i} \int_{\Gamma} z^k F(z)^{-1} \mathbf{v} dz, \quad k = 0, 1, \dots,$$

and let

$$S := [\mathbf{s}_0, \dots, \mathbf{s}_{m-1}].$$

Note that $\mu_k = \mathbf{u}^H \mathbf{s}_k$. Then, we have the following theorem.

Theorem 3.5 *Let $(\lambda_l, \mathbf{w}_l)$ be the eigenpairs of the pencil $H_m^< - \lambda H_m$ and let*

$$\mathbf{x}_l := S \mathbf{w}_l, \quad l = 1, \dots, m.$$

Then $\mathbf{x}_l$ are the eigenvectors of $F(z)$ corresponding to eigenvalue λ_l for $l = 1, \dots, m$.

For the proof of Theorem 3.5, we need the following lemma.

Lemma 3.6 *Let $\mathbf{q}_n(z)$ be a vector polynomial of (2) and let $(\lambda_l, \mathbf{w}_l)$ be the eigenpairs of the pencil $H_m^< - \lambda H_m$. Then,*

$$\mathbf{q}_n(\lambda_l) = c_l S \mathbf{w}_l, \quad c_l \in \mathbb{C} \setminus \{0\}$$

for $l = 1, \dots, m$.

Proof. From equation(6),

$$\begin{aligned} 0 &= (H_m^< - \lambda_j H_m) \mathbf{w}_l \\ &= V_m D_m (\Lambda_m - \lambda_l I) V_m^T \mathbf{w}_l. \end{aligned}$$

Since V_m and D_m are nonsingular, and $\Lambda_m \mathbf{e}_l = \lambda_l \mathbf{e}_l$, $V_m^T \mathbf{w}_l$ admits the following representation:

$$V_m^T \mathbf{w}_l = \alpha_l \mathbf{e}_l, \quad \alpha_l \in \mathbb{C} \setminus \{0\}. \quad (7)$$

Here, $\mathbf{e}_l$ is the l -th unit vector. Let $\mathbf{p}_n(z)$ be a vector polynomial of (3) and let

$$\sigma_l := \frac{\mathbf{p}_n(\lambda_l)^H \mathbf{v}}{\prod_{\substack{i \neq l \\ l \leq m}} (\lambda_l - \lambda_i) \prod_{i > m} (\lambda_j - \lambda_i)^{\alpha_{ni}}}, \quad l = 1, \dots, m.$$

Note that $\sigma_l \neq 0$ if $\chi_n(\lambda_l) \neq 0$. As in the proof of Theorem 3.4, we can derive the following equation.

$$S = [\mathbf{s}_0, \dots, \mathbf{s}_{m-1}] = [\sigma_1 \mathbf{q}_n(\lambda_1), \dots, \sigma_m \mathbf{q}_n(\lambda_m)] V_m^T.$$

Therefore,

$$\mathbf{q}_n(\lambda_l) = \frac{1}{\sigma_l} S V_m^{-T} \mathbf{e}_l = \frac{1}{\sigma_l} S \frac{1}{\alpha_l} \mathbf{w}_l = c_l S \mathbf{w}_l.$$

Here, c_l are

$$c_l = \frac{1}{\sigma_l \alpha_l}$$

for $l = 1, \dots, m$. □

From Lemma 2.4 and Lemma 3.6, we obtain

$$0 = F(\lambda_l) \mathbf{q}_n(\lambda_l) = F(\lambda_l) (c_l S \mathbf{w}_l)$$

for $l = 1, \dots, m$. Hence $\mathbf{x}_l = c_l S \mathbf{x}_l$ is the eigenvector of $F(z)$ corresponding to λ_l . This proves Theorem 3.5.

4 Case in which Γ is given by a circle

In actual applications, the contour integrals of equation (5) is evaluated numerically. In this section, we will introduce a detailed algorithm where a trapezoidal rule is employed on a circular integral path.

Let γ and ρ be the center and the radius, respectively, of the given circle. Let N be a positive integer, and let

$$\omega_j = \gamma + \rho \exp(2\pi i(j + 1/2)/N), \quad j = 0, 1, \dots, N-1.$$

The integral of (5) is evaluated via the N -point trapezoidal rule, and the momental weight is shifted and scaled based on a numerical consideration. Therefore, we obtain the following approximations for μ_k .

$$\mu_k \approx \hat{\mu}_k := \frac{1}{N} \sum_{j=0}^{N-1} \left(\frac{\omega_j - \gamma}{\rho} \right)^{k+1} f(\omega_j), \quad k = 0, 1, \dots$$

Note that the resulting eigenvalues are also shifted and scaled.

To calculate the eigenvectors, let

$$\mathbf{y}_j := F(\omega_j)^{-1} \mathbf{v}, \quad j = 0, 1, \dots, N-1,$$

and approximate $\mathbf{s}_k$ as

$$\mathbf{s}_k \approx \hat{\mathbf{s}}_k := \frac{1}{N} \sum_{j=0}^{N-1} \left(\frac{\omega_j - \gamma}{\rho} \right)^{k+1} \mathbf{y}_j, \quad k = 0, 1, \dots \quad (8)$$

Note that $\hat{\mu}_k = \mathbf{u}^H \hat{\mathbf{s}}_k$, because

$$f(\omega_j) = \mathbf{u}^H F(\omega_j)^{-1} \mathbf{v} = \mathbf{u}^H \mathbf{y}_j.$$

Because of the numerical approximation, the derived Hankel matrices $H_m^<$ and H_m contain numerical errors. Let us start from a generalized eigenvalue problem $A\mathbf{w} = \lambda B\mathbf{w}$, where A and B are perturbed by εF and εG , respectively, where $\|F\|_2 \leq 1$, $\|G\|_2 \leq 1$ and $|\varepsilon| \ll 1$. The perturbed problem becomes

$$(A + \varepsilon F)\mathbf{w}(\varepsilon) = \lambda(\varepsilon)(B + \varepsilon G)\mathbf{w}(\varepsilon).$$

Again, we assume λ to be a simple eigenvalue.

Let vector $\mathbf{u}$ be the corresponding left eigenvectors. Then, the following estimation is given in [12].

$$\lambda(\varepsilon) - \lambda = \varepsilon \frac{\mathbf{u}^H (F - \lambda G) \mathbf{w}}{\mathbf{u}^H B \mathbf{w}} + \mathcal{O}(\varepsilon^2). \quad (9)$$

The first term of the right-hand side of equation (9) is evaluated as

$$\varepsilon \frac{|\mathbf{u}^H (F - \lambda G) \mathbf{w}|}{|\mathbf{u}^H B \mathbf{w}|} \leq \varepsilon (1 + |\lambda|) \frac{\|\mathbf{u}\|_2 \|\mathbf{w}\|_2}{|\mathbf{u}^H B \mathbf{w}|}. \quad (10)$$

From equation (9) and equation (10), we obtain

$$|\lambda(\varepsilon) - \lambda| \leq \varepsilon (1 + |\lambda|) \frac{\|\mathbf{u}\|_2 \|\mathbf{w}\|_2}{|\mathbf{u}^H B \mathbf{w}|} + |\mathcal{O}(\varepsilon^2)|. \quad (11)$$

From equation (11), we can define the following sensitivity factor for each eigenvalue.

Definition 4.1 Let λ_i be an eigenvalue of the matrix pencil $A - \lambda B$, and let $\mathbf{x}_i$ and $\mathbf{w}_i$ be the corresponding right and left eigenvectors. Then we define the sensitivity factor of the eigenvalue λ_i as

$$\tau_i := (1 + |\lambda_i|) \frac{\|\mathbf{u}_i\|_2 \|\mathbf{w}_i\|_2}{|\mathbf{u}_i^H B \mathbf{w}_i|}, \quad i = 1, 2, \dots \quad (12)$$

Now, we will investigate the sensitivity factor in the SS method. Let $A = H_m^<$ and $B = H_m$. From equation (7), the right eigenvector of the matrix pencil $H_m^< - \lambda H_m$ becomes

$$\mathbf{w}_i = \alpha_i V_m^{-T} \mathbf{e}_i$$

for the eigenvalue λ_i . Since the pencil is symmetric, the corresponding left eigenvector is $\mathbf{u}_i = \bar{\mathbf{w}}_i$. Then we can derive the following equation

$$\begin{aligned} \frac{\|\mathbf{u}_i\|_2 \|\mathbf{w}_i\|_2}{|\mathbf{u}_i^H H_m \mathbf{w}_i|} &= \frac{\|V_m^{-T} \mathbf{e}_i\|_2^2}{|\mathbf{e}_i^T V_m^{-1} (V_m D_m V_m^T) V_m^{-T} \mathbf{e}_i|} \\ &= \frac{\|V_m^{-T} \mathbf{e}_i\|_2^2}{|\mathbf{e}_i^T D_m \mathbf{e}_i|} = \frac{1}{|\nu_i|} \|V_m^{-T} \mathbf{e}_i\|_2^2, \end{aligned}$$

where $D_m = \text{diag}(\nu_1, \dots, \nu_m)$.

Therefore, we obtain the sensitivity factor of the eigenvalues of $H_m^< - \lambda H_m$ as

$$\tau_i = \frac{1}{|\nu_i|} \|V_m^{-T} \mathbf{e}_i\|_2^2 (1 + |\lambda_i|). \quad (13)$$

When μ_k is approximated numerically, eigenvalues on the periphery of Γ start to contaminate. Therefore, we usually take the dimension of the Hankel matrix $\hat{m}$ larger than m . Assuming that those peripheral eigenvalues are also simple, Then, we can write

$$\hat{\mu}_k = \sum_{i=1}^{\hat{m}} \hat{\nu}_i \lambda_i^k + \mathcal{O}(\varepsilon).$$

Here $\hat{\nu}_i$ for $i \leq m$, $\hat{\nu}_i$ for $i > m$, $\hat{\lambda}_i$ and ε approach to ν_i , 0, λ_i and 0 respectively, as the integrator is improved. Indeed, in the case of the N -point trapezoidal rule applied on the circular path treated here, we can show that

$$\hat{\nu}_i = \frac{c_i}{\rho} \cdot \frac{1}{1 + \eta_i^N}, \quad (14)$$

where $\eta_i = (\lambda_i - \gamma)/\rho$ and c_i is a constant determined by $F(z)$. From equation (13), we can see that the inner eigenvalues λ_i , $i = 1, \dots, m$ are less sensitive to the perturbation, while the peripheral eigenvalues may fluctuate largely.

Note that Hankel matrices tend to be very ill-conditioned [19]. Also in the present case, the condition number of the Hankel matrix can be enormous because of the trace $|\hat{\nu}_i|$ for $i > m$. As shown in the sensitivity factor analysis, however, the inner eigenvalues can be calculated accurately.

The algorithm of the SS method is shown below.

Algorithm 1 (SS method)

Input: $\mathbf{u}, \mathbf{v} \in \mathbb{C}^n$, $N, K, \delta, \gamma, \rho$

Output: $\hat{\lambda}_1, \dots, \hat{\lambda}_{\hat{m}}$, $\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_{\hat{m}}$

1. Set $\omega_j \leftarrow \gamma + \rho \exp(2\pi i(j + 1/2)/N)$, $j = 0, \dots, N - 1$
2. Compute $\mathbf{y}_j = F(\omega_j)^{-1} \mathbf{v}$, $j = 0, \dots, N - 1$
3. Compute $\hat{\mathbf{s}}_k$, $k = 0, \dots, 2K - 1$ by (8)
4. Form $\hat{\mu}_k = \mathbf{u}^H \hat{\mathbf{s}}_k$, $k = 0, \dots, 2K - 1$
5. Construct $\hat{H}_K = [\hat{\mu}_{i+j-2}]_{i,j=1}^K$ and $\hat{H}_K^< = [\hat{\mu}_{i+j-1}]_{i,j=1}^K$
6. Perform singular value decomposition of $\hat{H}_K$
7. Omit small singular value components $s < \delta$ so that
 $\hat{H}_{\hat{m}} = \hat{H}_K(1 : \hat{m}, 1 : \hat{m})$, $\hat{H}_{\hat{m}}^< = \hat{H}_K^<(1 : \hat{m}, 1 : \hat{m})$,
 where $\hat{m} \leq K$
8. Compute the eigenpairs $(\zeta_1, \mathbf{w}_1), \dots, (\zeta_{\hat{m}}, \mathbf{w}_{\hat{m}})$ of the pencil $\hat{H}_{\hat{m}}^< - \lambda \hat{H}_{\hat{m}}$
9. Construct $S = [\hat{\mathbf{s}}_0, \dots, \hat{\mathbf{s}}_{\hat{m}-1}]$
10. Form $\hat{\mathbf{x}}_j = S \mathbf{w}_j$, $j = 1, \dots, \hat{m}$
11. Set $\hat{\lambda}_j \leftarrow \gamma + \rho \zeta_j$, $j = 1, \dots, \hat{m}$

In practice, we choose vectors $\mathbf{u}$ and $\mathbf{v}$ as random vectors. The dimension of the Hankel matrices $\hat{m}$ is determined from the singular value decomposition of H_K with the singular value threshold of δ . Therefore, K must be taken large enough that $K \geq \hat{m}$, where $\hat{m}$ is the number of inner and peripheral eigenvalues.

The block version of the SS method for generalized eigenvalue problems was proposed in [11]. The numerical examples in [11] indicate that the block SS method has the potential to achieve higher accuracy.

Let U and V be $n \times L$ matrices, the column vectors of which are linearly independent and used as initial vectors. The block SS method is defined by replacing $f(z)$ in equation (5) with matrices $U^H F(z)^{-1} V$. For a non-negative integer k , the matrix M_k is defined as

$$M_k := \frac{1}{2\pi i} \int_{\Gamma} z^k U^H F(z)^{-1} V dz, \quad k = 0, 1, \dots \quad (15)$$

Then, $\tilde{m}L \times \tilde{m}L$ block Hankel matrices $H_{\tilde{m}L}^<$ and $H_{\tilde{m}L}$ are defined as

$$H_{\tilde{m}L} := \begin{pmatrix} M_0 & M_1 & \cdots & M_{\tilde{m}-1} \\ M_1 & M_2 & \cdots & M_{\tilde{m}} \\ \vdots & \vdots & & \vdots \\ M_{\tilde{m}-1} & M_K & \cdots & M_{2\tilde{m}-2} \end{pmatrix}$$

and

$$H_{\tilde{m}L}^< := \begin{pmatrix} M_1 & M_2 & \cdots & M_{\tilde{m}} \\ M_2 & M_3 & \cdots & M_{\tilde{m}+1} \\ \vdots & \vdots & & \vdots \\ M_{\tilde{m}} & M_{\tilde{m}+1} & \cdots & M_{2\tilde{m}-1} \end{pmatrix}.$$

Here, the order of the block Hankel matrices $\tilde{m}$ can be smaller than m , where $\tilde{m}L \geq m$.

Let

$$S_k := \frac{1}{2\pi i} \int_{\Gamma} z^k F(z)^{-1} V dz, \quad k = 0, 1, \dots,$$

and let $S := [S_0, \dots, S_{\tilde{m}-1}]$.

By approximating the integral of equation (15) via the N -point trapezoidal rule, we obtain the following approximations for M_k .

$$M_k \approx \hat{M}_k := \frac{1}{N} \sum_{j=0}^{N-1} \left(\frac{\omega_j - \gamma}{\rho} \right)^{k+1} U^H F(\omega_j)^{-1} V, \quad j = 0, 1, \dots$$

Similarly, we obtain the follow approximations for S_k .

$$S_k \approx \hat{S}_k := \frac{1}{N} \sum_{j=0}^{N-1} \left(\frac{\omega_j - \gamma}{\rho} \right)^{k+1} F(\omega_j)^{-1} V, \quad j = 0, 1, \dots \quad (16)$$

Note that $M_k = U^H S_k$. Therefore, the algorithm of the block SS method is shown below.

Algorithm 2 (Block SS method)

Input: $U, V \in \mathbb{C}^{n \times L}$, $N, K, L, \delta, \gamma, \rho$

Output: $\hat{\lambda}_1, \dots, \hat{\lambda}_{\hat{m}}$, $\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_{\hat{m}}$

1. Set $\omega_j \leftarrow \gamma + \rho \exp(2\pi i(j + 1/2)/N)$, $j = 0, \dots, N - 1$
2. Compute $F(\omega_j)^{-1} V$, $j = 0, \dots, N - 1$
3. Compute $\hat{S}_k$, $k = 0, \dots, 2K - 1$ by (16)
4. Form $\hat{M}_k = U^H \hat{S}_k$, $k = 0, \dots, 2K - 1$
5. Construct $\hat{H}_{KL} = [\hat{M}_{i+j-2}]_{i,j=1}^K$ and $\hat{H}_{KL}^< = [\hat{M}_{i+j-1}]_{i,j=1}^K \in \mathbb{C}^{KL \times KL}$
6. Perform singular value decomposition of $\hat{H}_{KL}$
7. Omit small singular value components $s < \delta$ so that
 $\hat{H}_{\hat{m}} = \hat{H}_{KL}(1 : \hat{m}, 1 : \hat{m})$, $\hat{H}_{\hat{m}}^< = \hat{H}_{KL}^<(1 : \hat{m}, 1 : \hat{m})$,
where $\hat{m} \leq KL$
8. Compute the eigenpairs $(\zeta_1, \mathbf{w}_1), \dots, (\zeta_{\hat{m}}, \mathbf{w}_{\hat{m}})$ of the pencil $\hat{H}_{\hat{m}}^< - \lambda \hat{H}_{\hat{m}}$
9. Construct $S = [\hat{S}_0, \dots, \hat{S}_{\hat{m}-1}]$
10. Compute $\mathbf{x}_j = S \mathbf{w}_j$, $j = 1, \dots, \hat{m}$
11. Set $\hat{\lambda}_j \leftarrow \gamma + \rho \zeta_j$, $j = 1, \dots, \hat{m}$

In practice, we choose matrices U and V as random matrices.

Note that Algorithm 2 with $L = 1$ is equivalent to Algorithm 1. At Step 2, we need to solve the systems of linear equations with L right hand side vectors. Practically, however, we can take smaller N and larger ρ in the block version, so that the total computational cost is similar to, or even smaller than the single case. The rigorous proof of the block SS method, which includes the PEP with degeneracy, is underway.

5 Numerical examples

In this section, we investigate the theory using several examples of polynomial eigenvalue problems. The algorithm was implemented in MATLAB 7.4. We generated a matrix $V := [\mathbf{v}_1, \dots, \mathbf{v}_L]$ using the MATLAB function `rand` and set $U = V$. The MATLAB command `mldivide\` was used to evaluate $F(z)^{-1}V$ numerically. We compared the eigenvalues $\{\hat{\lambda}_j\}$ that were obtained by the Block SS method and the eigenvalues $\{\lambda_j^*\}$ that were computed by calling the MATLAB function `polyeig`. We define the residual of λ_j as $\|F(\hat{\lambda}_j)\hat{\mathbf{x}}_j\|_2$, where $\hat{\mathbf{x}}_j$ is the corresponding eigenvector and $\|\hat{\mathbf{x}}_j\|_2 = 1$.

Example 1

In Example 1, we confirm the error estimation of the approximated eigenvalues in Section 4. We consider the quadratic eigenvalue problem

$$F(\lambda) = \lambda^2 A_2 + \lambda(A_1 + \beta I_n) + A_0 + \beta A_1$$

given in [3, p.10]. Here, A_i are matrices of order $n = 10$ and A_0 is positive definite. We set $\beta = 0.8$.

We took the integral path Γ as

$$\Gamma = \gamma + \rho e^{i\theta} \quad (\gamma = -0.8, \rho = 7).$$

so that 4 eigenvalues lie in Γ . The distribution of eigenvalues and the integral path are shown in Figure 1. We took $N = 32$, $K = 16$, $L = 1$ and $\delta = 10^{-12}$.

We obtained 14 eigenvalues of $F(z)$ and computed approximation of their sensitivity factors $\hat{\tau}_k$. Note that from equation (13) and equation (14), we can approximate sensitivity factors τ_k by

$$\hat{\tau}_k = \rho(1 + |\eta_k|^N)(1 + |\lambda_k|)\|V_m^{-T}\mathbf{e}_k\|_2^2,$$

where we set $\lambda_k \leftarrow \hat{\lambda}_k$ and $m \leftarrow \hat{m}$.

The numerical results are shown in Table 1. In Table 1, underlines denote disagreement of digits between $\hat{\lambda}_k$ and λ_k^* . As shown in Table 1, the sensitivity factors $\hat{\tau}_k$ are exponentially amplified if λ_k is outside Γ . We can also see that the relative errors for the outer eigenvalues actually increase exponentially. Meanwhile, the inner eigenvalues were calculated accurately, even though the condition number of the Hankel matrix was very large such that $\text{cond}(H_{\hat{m}}) = 8.25 \times 10^{12}$.

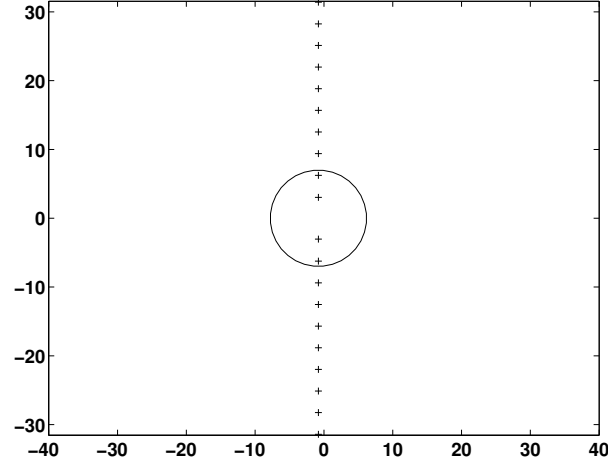


Figure 1: Eigenvalues in Example 1.

Table 1: Relative errors and sensitivity factors in Example 1.

k	$\hat{\lambda}_k$	$ \hat{\lambda}_k - \lambda_k^* / \hat{\lambda}_k $	$\hat{\tau}_k$
1	$-0.799999999999723 + 3.037712186344210 \text{ i}$	1.09×10^{-12}	1.54×10^3
2	$-0.8000000000000111 - 3.037712186344154 \text{ i}$	1.07×10^{-12}	1.54×10^3
3	$-0.799999999999710 + 6.231420704050912 \text{ i}$	7.44×10^{-13}	3.03×10^3
4	$-0.799999999999999 - 6.231420704050882 \text{ i}$	7.38×10^{-13}	3.03×10^3
5	$-0.799999988132865 + 9.389825194259766 \text{ i}$	2.98×10^{-8}	1.46×10^7
6	$-0.799999997431621 - 9.389825196314405 \text{ i}$	3.00×10^{-8}	1.46×10^7
7	$-0.799979432271814 + 12.540406718026578 \text{ i}$	6.12×10^{-5}	2.01×10^{10}
8	$-0.799992884762234 - 12.540413682454103 \text{ i}$	6.18×10^{-5}	2.00×10^{10}
9	$-0.799811460749113 + 15.699437889062162 \text{ i}$	8.54×10^{-4}	8.82×10^{11}
10	$-0.799917135958713 - 15.699515527393906 \text{ i}$	8.59×10^{-4}	8.78×10^{11}
11	$-0.771833428737543 + 21.756907623843826 \text{ i}$	9.98×10^{-3}	2.49×10^{13}
12	$-0.784076104391534 - 21.770287407471734 \text{ i}$	9.32×10^{-3}	2.50×10^{13}
13	$-0.788213898295595 + 28.952844300096686 \text{ i}$	2.45×10^{-2}	7.76×10^{13}
14	$-0.792216122017860 - 28.958853070430731 \text{ i}$	2.47×10^{-2}	7.79×10^{13}

Table 2: Relative errors and residuals in Example 2.

k	$\hat{\lambda}_k$	$ \hat{\lambda}_k - \lambda_k^* / \hat{\lambda}_k $	$\ F(\hat{\lambda}_k)\hat{\mathbf{x}}_k\ _2$
1	0.333333333333372	1.05×10^{-13}	1.78×10^{-14}
2	0.499999999999953	8.24×10^{-14}	1.41×10^{-14}
3	1.000000000000012	9.10×10^{-15}	1.53×10^{-14}
4	1.000000000000001 i	1.02×10^{-15}	1.94×10^{-14}
5	- 1.000000000000001 i	1.02×10^{-15}	1.49×10^{-14}

Example 2

In Example 2, we show that the SS method can be used to solve polynomial eigenvalue problems. We consider the quadratic eigenvalue problem with

$$F(\lambda) = \begin{pmatrix} \lambda + 1 & 6\lambda^2 - 6\lambda & 0 \\ 2\lambda & 6\lambda^2 - 7\lambda + 1 & 0 \\ 0 & 0 & \lambda^2 + 1 \end{pmatrix},$$

given in [18, p.250]. There are six eigenvalues $\lambda_1 = 1/3, \lambda_2 = 1/2, \lambda_3 = 1, \lambda_4 = i, \lambda_5 = -i, \lambda_6 = \infty$.

The integral path Γ was taken as follows:

$$\Gamma = \gamma + \rho e^{i\theta} \quad (\gamma = 0, \rho = 1.5).$$

We took $N = 10, K = 5, L = 1$ and $\delta = 10^{-12}$.

The numerical results are shown in Table 2. The SS method found five eigenvalues inside the circle. Therefore, the SS method can be used for polynomial eigenvalue problems.

Example 3

Next, we consider the connected damped mass-spring system [18, p.258]

$$F(\lambda) = \lambda^2 A_2 + \lambda A_1 + A_0,$$

where

$$A_2 = I_n, \quad A_1 = \kappa_1 \text{ tridiag}(-1, 3, -1), \quad A_0 = \kappa_0 \text{ tridiag}(-1, 3, -1).$$

We took $n = 50$ degrees of freedom and chose $\kappa_0 = 5, \kappa_1 = 3$.

The integral path Γ was taken as follows:

$$\Gamma = \gamma + \rho e^{i\theta} \quad (\gamma = -2 + 1.5i, \rho = 1).$$

All of the eigenvalues lie in the left half-plane shown in Figure 2. The circle shows Γ . A total of 16 eigenvalues lie in Γ .

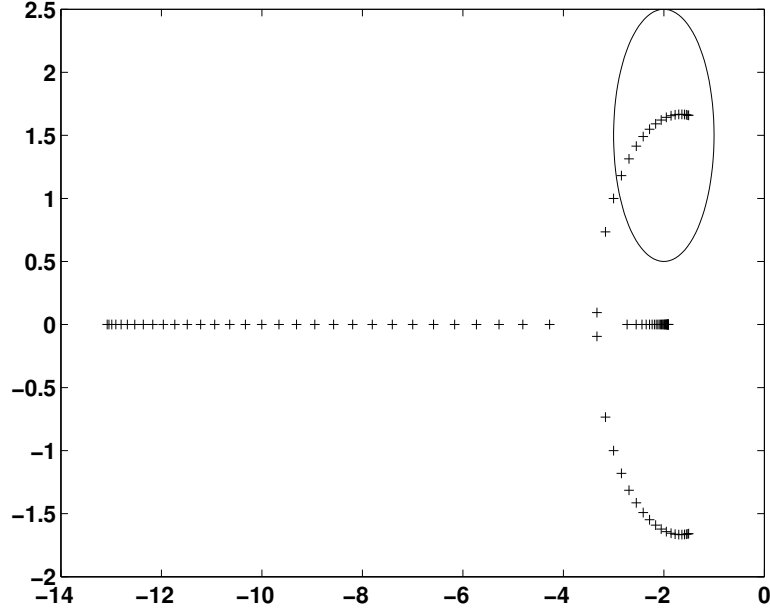


Figure 2: Eigenvalues in Example 3.

We set $N = 32$, $K = 8$, $L = 24$ and $\delta = 10^{-12}$.

The numerical results are shown in Table 3. As shown in Table 3, we obtained all of the eigenvalues in Γ . The largest residual of the computed eigenvalues is 2.72×10^{-12} .

Example 4

In Example 4, we show that the SS method can be applied to higher-order matrix polynomials. We consider the quartic eigenvalue problem

$$F(\lambda) = \lambda^4 A_4 + \lambda^3 A_3 + \lambda^2 A_2 + \lambda A_1 + A_0,$$

given in [14]. Here, A_i are matrices of order $n = 64$, obtained by a tensor product construction. We constructed A_i by calling the MATLAB function `kron`.

The integral path Γ was taken as follows:

$$\Gamma = \gamma + \rho e^{i\theta} \quad (\gamma = 1 - i, \rho = 0.5).$$

The distribution of eigenvalues and the integral path are shown in Figure 3. A total of 13 eigenvalues lies in Γ . We set $N = 64$, $K = 8$, $L = 24$ and $\delta = 10^{-12}$.

The numerical results are shown in Table 4. As shown in Table 4, we obtained all of the eigenvalues in Γ . The largest residual of the computed eigenvalues is 2.32×10^{-12} .

Table 3: Relative errors and residuals in Example 3.

k	$\hat{\lambda}_k$	$ \hat{\lambda}_k - \lambda_k^* / \hat{\lambda}_k $	$\ F(\hat{\lambda}_k)\hat{\mathbf{x}}_k\ _2$
1	$-1.505690013788871 + 1.658874406037429 i$	3.86×10^{-15}	4.45×10^{-13}
2	$-1.522738470984195 + 1.660440439240554 i$	5.31×10^{-16}	1.94×10^{-13}
3	$-1.551080700948297 + 1.662653801098332 i$	2.42×10^{-15}	2.30×10^{-13}
4	$-1.590609191894973 + 1.664930340377379 i$	5.05×10^{-15}	4.41×10^{-13}
5	$-1.641173998718531 + 1.666471692426448 i$	3.08×10^{-15}	5.42×10^{-13}
6	$-1.702583311786932 + 1.666279620106145 i$	5.20×10^{-15}	2.22×10^{-13}
7	$-1.774604184541429 + 1.663167841804553 i$	1.13×10^{-15}	3.80×10^{-13}
8	$-1.856963417142646 + 1.655767170992366 i$	1.33×10^{-15}	6.00×10^{-13}
9	$-1.949348592811159 + 1.642519012495447 i$	8.58×10^{-16}	6.12×10^{-13}
10	$-2.051409262931337 + 1.621650675206375 i$	4.42×10^{-15}	4.51×10^{-13}
11	$-2.162758276422983 + 1.591122525925358 i$	6.99×10^{-15}	7.57×10^{-13}
12	$-2.282973248338001 + 1.548529617142127 i$	9.00×10^{-15}	2.72×10^{-12}
13	$-2.411598162111222 + 1.490924157988098 i$	2.07×10^{-15}	1.42×10^{-12}
14	$-2.548145099387273 + 1.414487027309264 i$	1.07×10^{-15}	9.79×10^{-14}
15	$-2.692096090862231 + 1.313876810729122 i$	1.06×10^{-15}	9.84×10^{-14}
16	$-2.842905081118472 + 1.180779814421682 i$	8.54×10^{-15}	3.94×10^{-13}

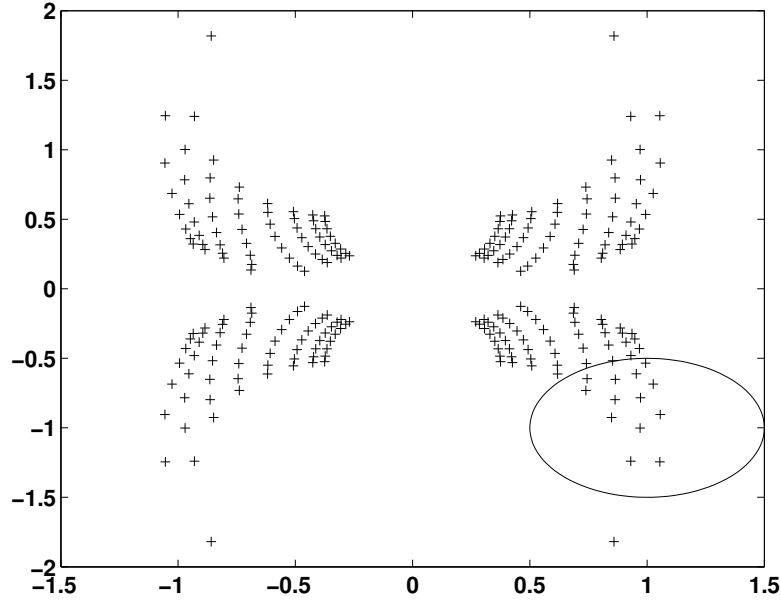


Figure 3: Eigenvalues in Example 4.

Table 4: Relative errors and residuals in Example 4.

k	$\hat{\lambda}_k$	$ \hat{\lambda}_k - \lambda_k^* / \hat{\lambda}_k $	$\ F(\hat{\lambda}_k)\hat{\mathbf{x}}_k\ _2$
1	0.744283786331637 - 0.646541111813926 i	3.76×10^{-14}	9.57×10^{-13}
2	0.738844823886407 - 0.731658534005984 i	3.87×10^{-14}	1.20×10^{-12}
3	0.864617980453669 - 0.651815654480539 i	1.23×10^{-14}	1.43×10^{-12}
4	0.994127888031147 - 0.535135868221425 i	8.85×10^{-15}	9.69×10^{-13}
5	0.953854040217050 - 0.611439886362259 i	6.30×10^{-15}	5.88×10^{-13}
6	0.863349700394677 - 0.797929809342592 i	1.68×10^{-14}	8.96×10^{-13}
7	1.026189973208218 - 0.685703044215538 i	5.96×10^{-15}	9.48×10^{-13}
8	0.971854722649304 - 0.783539836463599 i	1.18×10^{-14}	2.32×10^{-12}
9	0.848570953056571 - 0.925677807336431 i	2.05×10^{-14}	1.33×10^{-12}
10	1.056265535074977 - 0.904134007343116 i	7.36×10^{-15}	1.45×10^{-12}
11	0.970370449857825 - 1.001776965449533 i	4.73×10^{-15}	6.59×10^{-13}
12	0.930660687304600 - 1.240183199928941 i	8.54×10^{-15}	1.08×10^{-12}
13	1.054414864515329 - 1.244513158205443 i	1.53×10^{-14}	1.86×10^{-12}

6 Conclusions

In the present paper, we have proposed a novel method to solve polynomial eigenvalue problems. By using contour integral, the method converts the original problem into a small linear eigenvalue problem that has identical eigenvalues in the domain defined by the integral path. The numerical evaluation of the contour integral can be performed highly in parallel, which makes the method suitable to the modern cluster computer architecture. The present method is closely related to the contour integral method proposed for generalized eigenvalue problems [16], though the theories behind them are not identical. Relevance and irrelevance between them are still under study.

From an application point of view, the potential applicability of the polynomial eigenvalue problem has not been scrutinized yet, especially for the higher order polynomials. It is probably due to the lack of effective numerical methods to solve high-dimensional and high-order polynomial eigenvalue problems. The proposed method can deal with high-order problems without pain, which may derive quite a new algorithm in the field of science and engineering. We are now under collaboration with physicists and chemists in this direction.

References

- [1] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. Van Der Vorst, Templates for the solution of algebraic eigenvalue problems: A practical guide, SIAM, Philadelphia (2000).
- [2] Z. Bai and Y. Su, SOAR: A second order Arnoldi method for the solution of the quadratic eigenvalue problem, SIAM J. Matrix Anal. Appl., **26** (2005) 640–659.

- [3] T. Betcke, N. J. Higham, V. Mehrmann, C. Schröder, and F. Tisseur, NLEVP: A collection of nonlinear eigenvalue problem, MIMS EPrint 2008.40 (2008).
- [4] F. R. Gantmacher, The theory of matrices, vol. 1-2, Chelsea Publishing Company (1977).
- [5] I. M. Gel'fand, Lecture on linear algebra, Dover Publications, INC (1961).
- [6] I. Gohberg, M. A. Kaashoek and P. Lancaster, General theory of regular matrix polynomials and band toeplitz operators, Integr. Eq. Oper. Theory, **11** (1988) 776–882.
- [7] I. Gohberg, P. Lancaster, F. Rodman, Matrix polynomials, Academic Press, New York (1982).
- [8] R. S. Heeg, Stability and transition of attachment-line flow, Ph.D.thesis, Universiteit Twente, Enschede, the Netherlands (1998).
- [9] N. J. Higham and F. Tisseur, Bounds for eigenvalues of matrix polynomials, Lin. Alg. Appl., **358** (2003) 5–22.
- [10] T. Hwang, W. Lin, J. Liu and W. Wang, Jacobi-Davidson methods for cubic eigenvalue problems, Numer. Lin. Alg. Appl., **12** (2005) 605–624.
- [11] T. Ikegami, T. Sakurai, U. Nagashima, A filter diagonalization for generalized eigenvalue problems based on the Sakurai-Sugiura projection method, Technical Report CS-TR-08-13, Tsukuba (2008).
- [12] P. Kravanja, T. Sakurai, H. Sugiura, M. Van Barel, A perturbation result for generalized eigenvalue problems and its application to error estimation in a quadrature method for computing zeros of analytic functions, J. Comput. Appl. Math., **161** (2003) 339–347.
- [13] D. S. Mackey, N. Mackey, C. Mehl and V. Mehrmann, Vector spaces of linearizations for matrix polynomials, SIAM J. Matrix Anal. Appl., **28** (2006) 971–1004.
- [14] V. Mehrmann and D. Watkins, Polynomial eigenvalue problems with Hamiltonian structure, Electron. Trans. Numer. Anal., **13** (2002) 106–118.
- [15] E. Pereira, On solvents of matrix polynomials, Appl. Numer. Math., **47** (2003) 197–208.
- [16] T. Sakurai, H. Sugiura, A projection method for generalized eigenvalue problems, J. Comput. Appl. Math., **159** (2003) 119–128.
- [17] G. L. G. Sleijpen, A. G. L. Booten, D. R. Fokkema and H. A. Van Der Vorst, Jacobi-Davidson type methods for generalized eigenproblems and polynomial eigenproblems, BIT, **36** (1996) 595–633.

- [18] F. Tisseur, K. Meerbergen, The quadratic eigenvalue problem, SIAM Rev., **43** (2001) 235–286.
- [19] E. E. Tyrtyshnikov, How bad are Hankel matrices?, Numer. Math., **67** (1994) 261–269.