Efficient parameter estimation and implementation of a contour integral-based eigensolver

Tetsuya Sakurai[†,‡], Yasunori Futamura[†] and Hiroto Tadano[†]
[†]Department of Computer Science, University of Tsukuba
[‡]CREST, Japan Science and Technology Agency

Abstract: We consider an eigensolver for computing eigenvalues in a given domain and the corresponding eigenvectors of large-scale matrix pencils. The Sakurai-Sugiura (SS) method is an eigensolver based on complex moments given by contour integrals of matrix inverses with several shift points. This method has good parallel scalability, and is suitable for massively parallel computing environments. The SS method has several parameters, and the choice of these parameters is crucial for achieving high accuracy and good parallel performance. We discuss some numerical properties of the method, and present efficient parameter estimation techniques. We demonstrate the efficiency of our method with numerical experiments.

1 INTRODUCTION

Large-scale eigenvalue problems arise in many scientific and engineering areas such as nano simulation, vibration analysis, data analysis, etc. Massively parallel computers are used to solve such large-scale problems, and they require efficient algorithms for parallel computing.

In this paper, we consider the problem of finding several eigenvalues in a given domain and their corresponding eigenvectors of the generalized eigenvalue problem

$$A\boldsymbol{x} = \lambda B\boldsymbol{x},$$

where $A, B \in \mathbb{C}^{n\times n}$. In [14], an eigensolver for generalized eigenvalue problems using complex moments is proposed. This method is called the Sakurai-Sugiura (SS) method. In the SS method, contour integrals with a source vector $\boldsymbol{v}$ are used to generate a subspace spanned by a set of eigenvectors with respect to the eigenvalues in a target domain. A large-scale eigenvalue problem is reduced to a small eigenvalue problem with Hankel matrices constructed from complex moments. In [16], an interpretation for filtering of spectrum is used to discuss numerical properties of a contour integral approximated by numerical quadrature. An influence of approximation by numerical quadrature is considered as a contamination of eigencomponents, and the choice of an appropriate subspace size provides accurate eigenpairs in a target domain.

A variant of the SS method that improves numerical accuracy by using the Rayleigh-Ritz procedure is presented in [15]. Ikegami, et al. [6, 7] presented a block version of the SS method that uses multiple source vectors instead of the single source vector for the contour integrals. The block SS method improves numerical stability when the target domain contains many eigenvalues. Moreover, this method can treat multiple eigenvalues. In [1, 2], the SS method is extended to nonlinear eigenvalue problems. As related works of eigensolvers using contour integrals, Polizzi [13] proposed an iterative refinement of a contour integral method for symmetric or Hermitian positive definite eigenvalue problems. Beyn [3] proposed a method for nonlinear eigenvalue problems using contour integrals with a singular value decomposition of a matrix with a Hankel type structure. Yokota, et al. [21] proposed a Rayleigh-Ritz type method using contour integrals for nonlinear eigenvalue problems. In this method, a subspace that includes target eigenvectors are generated by contour integrals, and a large-scale nonlinear eigenvalue problem is projected to a small nonlinear eigenvalue problem, and the projected problem is solved by Hankel type nonlinear eigensolver using contour integrals.

The SS method computes a set of eigenvalues by computing the solutions to systems of linear equations

$$(z_j B - A)Y_j = BV, \quad j = 1, \ldots, N, \tag{1}$$

where V is a matrix with L column vectors and z_j is a shift point on the complex plane. The method computes the desired eigenvalues inside of a border defined by the set of shifts $\{z_j\}$. The first step of the SS method is the construction of a subspace that includes the eigenvectors corresponding to the eigenvalues located inside the given domain. In this step, solutions of linear systems at several shift points are used. The second step is to solve the projected problem in the subspace and to extract the approximate eigenvalues and the corresponding eigenvectors for the original problem. Since the size of the projected subspace is assumed to be small compared with the original matrix size, the computational costs of the first step is dominant.

Krylov subspace methods for multiple right-hand sides are efficient for solving the linear systems (1). In [11, 18], methods to improve numerical stability and convergence for block Krylov subspace methods are presented. In the case of standard eigenvalue problems, the linear systems (1) are shifted linear systems, and a shift invariance of the Krylov subspace reduces computational costs to obtain solutions of linear systems at several shift points [12]. The application of the SS method with the shifted CG method for shell model calculations is reported in [10]. Futamura, et al. [5] presented an efficient implementation techniques for solving such shifted linear systems in parallel computing, and a report of application for band structure calculations with the real space density functional theory on the K computer is presented. Yamazaki, et al. [20] implemented a nonlinear version of the SS method, and evaluated parallel performances of the method.

Each of the linear systems is independent with respect to the other shifts, so each can be solved without any consideration of the nodes assigned to different shifts in distributed computing. Therefore, the method provides coarse-grained parallelism of computation. By employing a parallel linear solver for each shift point, the total

number of nodes is the product of the number of nodes assigned for each linear system and the number of shift points.

The SS method has several parameters, and the choice of these parameters is crucial for achieving high accuracy and good parallel performance. In this paper, we show some numerical properties of the method. The contour integral for a matrix inverse is regarded as a filter for an eigensubspace. When the contour integral is approximated by numerical quadrature, the quadrature error causes contamination of the eigencomponents corresponding to the eigenvalues located outside of the contour path. Based on these properties, we propose efficient parameter estimation techniques for the SS method.

In [4], a method for stochastic estimation of number of eigenvalues in a given domain is proposed. This estimation can be used for predicting appropriate parameters. Maeda, et al. [9] extended this eigenvalue count method to nonlinear eigenvalue problems.

The rest of this paper is organized as follows. In Section 2, we briefly introduce the SS method. In Section 3, the properties of numerical quadrature applied for a matrix inverse are discussed. In Section 4, efficient parameter estimation methods are presented. Some numerical experiments are shown in Section 5. The last section concludes the paper.

2 A CONTOUR INTEGRAL-BASED EIGENSOLVER

In this section, we briefly introduce the SS method. For matrices $A, B \in \mathbb{C}^{n\times n}$, let $\lambda_1, \ldots, \lambda_n$ be eigenvalues of the matrix pencil $A - \lambda B$, and let $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ be corresponding eigenvectors. Let Γ be a positively oriented closed Jordan curve in the complex plane, and let G be a domain for which the border is given by Γ. We will find the eigenvalues inside Γ and the corresponding eigenvectors by using contour integrals.

2.1 Eigensubspace obtained by contour integrals

Suppose that m eigenvalues $\lambda_1, \ldots, \lambda_m$ are located inside Γ, and other eigenvalues are located outside Γ. Define a sequence of matrices $F_0, F_1, \ldots$ as

$$F_k = \frac{1}{2\pi \mathrm{i}} \int_\Gamma z^k (zB - A)^{-1} B \, \mathrm{d}z, \quad k = 0, 1, \ldots. \tag{2}$$

For a matrix $V \in \mathbb{R}^{n\times L}$ with a positive integer L, let

$$S_k = F_k V = \frac{1}{2\pi \mathrm{i}} \int_\Gamma z^k (zB - A)^{-1} B \, \mathrm{d}z \, V, \quad k = 0, \ldots, M-1, \tag{3}$$

where M is chosen such that $LM \geq m$, and set

$$F = [F_0, F_1, \ldots, F_{M-1}]$$

and

$$S = [S_0, S_1, \ldots, S_{M-1}].$$

According to [14], the column vectors of S are given by linear combinations of the eigenvectors with respect to the eigenvalues located inside Γ, and thus

$$\mathrm{span}(S) = \mathrm{span}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m),$$

if the column space of V includes $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$. V is called a source matrix for the contour integral. In practice, the elements of V are set by a random number generator. The eigenvectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ are obtained from S when the maximum multiplicity of the eigenvalues in Γ is less than or equal to L.

Using the Rayleigh-Ritz procedure with S, we can extract the eigenpairs. Let the singular value decomposition of S be

$$S = U \Sigma W^{\mathrm{H}},$$

where $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_{LM})$, $U \in \mathbb{C}^{n \times LM}$ and $W \in \mathbb{C}^{LM \times LM}$. Since the rank of S is m, $\sigma_m \neq 0$ and $\sigma_{m+1} = \cdots = \sigma_{LM} = 0$. Setting $U_m = U(:, 1:m)$, we calculate the projected matrices as

$$A_m = U_m^{\mathrm{H}} A U_m, \quad B_m = U_m^{\mathrm{H}} B U_m. \tag{4}$$

Let $\omega_1, \ldots, \omega_m$ be the eigenvalues of the matrix pencil $A_m - \lambda B_m$, and let $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m$ be the corresponding eigenvectors. Then the eigenvalues inside Γ of the matrix pencil $A - \lambda B$ are given by

$$\lambda_i = \omega_i, \quad i = 1, \ldots, m,$$

and the corresponding eigenvectors are given by

$$\boldsymbol{x}_j = U_m \boldsymbol{r}_j, \quad j = 1, \ldots, m. \tag{5}$$

When the matrices are large, storage of S and computation of the singular value decomposition restrict the application size of the method. The use of Hankel matrices reduces the memory requirement and computational costs. Let $\mathcal{M}_k \in \mathbb{C}^{L \times L}$ be

$$\mathcal{M}_k = \frac{1}{2\pi \mathrm{i}} \int_{\Gamma} z^k V^{\mathrm{T}} (zB - A)^{-1} B V \, \mathrm{d}z. \tag{6}$$

Let the Hankel matrices $H_{LM}, H_{LM}^{<} \in \mathbb{C}^{LM \times LM}$ be

$$H_{LM} = \begin{bmatrix} \mathcal{M}_0 & \mathcal{M}_1 & \cdots & \mathcal{M}_{M-1} \\ \mathcal{M}_1 & \mathcal{M}_2 & \cdots & \mathcal{M}_M \\ \vdots & \vdots & & \vdots \\ \mathcal{M}_{M-1} & \mathcal{M}_M & \cdots & \mathcal{M}_{2M-2} \end{bmatrix}$$

and

$$H_{LM}^{<} = \begin{bmatrix} \mathcal{M}_1 & \mathcal{M}_2 & \cdots & \mathcal{M}_M \\ \mathcal{M}_2 & \mathcal{M}_3 & \cdots & \mathcal{M}_{M+1} \\ \vdots & \vdots & & \vdots \\ \mathcal{M}_M & \mathcal{M}_{M+1} & \cdots & \mathcal{M}_{2M-1} \end{bmatrix}.$$

Let the singular value decomposition of $\tilde{H}_m$ be

$$\tilde{H}_m = \tilde{U}\tilde{\Sigma}\tilde{W}^{\mathrm{H}}, \tag{7}$$

where $\tilde{H}_m = H_{LM}(1:m, 1:m)$ and $\tilde{H}_m^{<} = H_{LM}^{<}(1:m, 1:m)$. Let $\tilde{\omega}_1, \ldots, \tilde{\omega}_m$ and $\tilde{\boldsymbol{q}}_1, \ldots, \tilde{\boldsymbol{q}}_m$ be the eigenvalues and the corresponding eigenvectors such that

$$(\tilde{\Sigma}^{-1}\tilde{U}^{\mathrm{H}}\tilde{H}_m^{<}\tilde{W})\tilde{\boldsymbol{q}}_i = \tilde{\omega}_i\tilde{\boldsymbol{q}}_i, \quad i = 1, \ldots, m.$$

Then the eigenvalues of the matrix pencil $A - \lambda B$ in Γ are given by

$$\lambda_i = \tilde{\omega}_i.$$

The eigenvectors are given by

$$\boldsymbol{x}_i = S(:, 1:m)\tilde{W}\boldsymbol{q}_i, \quad i = 1, \ldots, m.$$

In this computation, the singular value decomposition of S does not required. A disadvantage using the Hankel matrices with the moment matrices $\mathcal{M}_k$ is numerical instability comparing with the Rayleigh-Ritz procedure in the case of numerical computation with large m.

In the case of the nonlinear eigenvalue problem $T(\lambda)\boldsymbol{x} = \boldsymbol{0}$ with a matrix valued function $T(\lambda)$, the integrand $V^{\mathrm{T}}(zB - A)^{-1}BV$ in (6) is replaced by $V^{\mathrm{T}}T(z)^{-1}V$ [1, 2]. Note that the derived eigenvalue problem with Hankel matrices are linear even if the original problem is nonlinear. In [3], the integrand in the contour integral (6) is replaced by $T(z)^{-1}V$ instead of $V^{\mathrm{T}}T(z)^{-1}V$.

2.2 Approximation by a numerical quadrature

The contour integral in (2) is approximated by an N-point numerical quadrature. Suppose that a Jordan curve Γ is represented by scaling and shifting from a Jordan curve Γ_0 with a scaling factor ρ and sa hift γ. Without any loss of generality, we assume that Γ_0 encloses the origin. Let $\zeta(\theta)$ be a point on Γ_0 with a parameter θ, $0 \le \theta \le 2\pi$, and let z on Γ be given by

$$z(\theta) = \gamma + \rho\zeta(\theta).$$

Then the contour integral of a function $f(z)$ is given by

$$\frac{1}{2\pi\mathrm{i}}\int_\Gamma f(z)\mathrm{d}z = \frac{1}{2\pi}\int_0^{2\pi} f(z)(-\mathrm{i}\rho\zeta'(\theta))\mathrm{d}\theta = \frac{1}{2\pi}\int_0^{2\pi} \rho f(z)w(\theta)\mathrm{d}\theta, \tag{8}$$

where $w(\theta) = -\mathrm{i}\zeta'(\theta)$. The integral (8) is approximated by the N-point quadrature rule

$$\frac{1}{2\pi\mathrm{i}}\int_\Gamma f(z)\mathrm{d}z \approx \sum_{j=1}^{N} \rho w_j f(z_j), \tag{9}$$

where $w_j = w(\theta_j)\Delta_j/(2\pi)$, $\zeta_j = \zeta(\theta_j)$ and $z_j = \gamma + \rho\zeta_j$ with appropriate θ_j and Δ_j, $j = 1, \ldots, N$.

Since

$$\frac{1}{2\pi \mathrm{i}} \int_{\Gamma_0} \zeta^k \mathrm{d}\zeta = \begin{cases} 1, & k = -1 \\ 0, & \text{otherwise} \end{cases}$$

for integer k, the quadrature points $\zeta_1, \ldots, \zeta_N$ on Γ_0 and the corresponding weights $w_1, \ldots, w_N$ are set to satisfy

$$\sum_{j=1}^{N} w_j \zeta_j^k = \begin{cases} \nu \neq 0, & k = -1 \\ 0, & k = 0, \ldots, N-2 \end{cases}, \tag{10}$$

where ν is a nonzero constant.

In particular, when Γ is a circle with center γ and radius ρ, and the quadrature points are set as

$$z_j = \gamma + \rho(\cos\theta_j + \mathrm{i}\sin\theta_j), \quad j = 1, \ldots, N,$$

where $\theta_j = (2\pi/N) \times (j - 1/2)$, $j = 1, \ldots, N$, then Γ_0 is the unit circle and the quadrature weights are given by

$$w_j = \cos\theta_j + \mathrm{i}\sin\theta_j, \quad j = 1, \ldots, N.$$

In the case that all the eigenvalues are located on the real axis, it might be better to put the quadrature points closer to the real axis as follows:

$$z_j = \gamma + \rho(\cos\theta_j + \mathrm{i}\alpha\sin\theta_j), \quad j = 1, \ldots, N \tag{11}$$

with a vertical scaling factor $0 < \alpha < 1$. The corresponding quadrature weights are given by

$$w_j = \alpha\cos\theta_j + \mathrm{i}\sin\theta_j, \quad j = 1, \ldots, N. \tag{12}$$

In [12], quadrature points are set on straight lines to reuse solutions of linear systems. The Gauss-Legendre quadrature rule on a circle is used for the numerical quadrature in [13].

Using the quadrature rule (9), F_k and S_k are approximated by

$$F_k \approx \hat{F}_k = \sum_{j=1}^{N} \rho w_j \zeta_j^k (z_j B - A)^{-1} B \tag{13}$$

and

$$\hat{S}_k = \hat{F}_k V = \sum_{j=1}^{N} \rho w_j \zeta_j^k (z_j B - A)^{-1} B V. \tag{14}$$

Matrices F and S are approximated by $\hat{F} = [\hat{F}_0, \ldots, \hat{F}_{M-1}]$ and $\hat{S} = [\hat{S}_0, \ldots, \hat{S}_{M-1}]$.

The Rayleigh-Ritz procedure for $\hat{S}$ gives the approximate eigenvalues $\hat{\lambda}_i$ and the eigenvectors $\hat{\boldsymbol{x}}_i$. Let the singular value decomposition of $\hat{S}$ be

$$\hat{S} = \hat{U}\hat{\Sigma}\hat{W}^{\mathrm{H}},$$

where $\hat{\Sigma} = \mathrm{diag}(\hat{\sigma}_1, \ldots, \hat{\sigma}_{LM})$. Let K be the number of singular values of $\hat{S}$ that satisfy $\hat{\sigma}_i \geq \delta$, $1 \leq i \leq K$ with small $\delta > 0$. We calculate the projected matrices as

$$\hat{A} = \hat{U}(:, 1:K)^{\mathrm{H}}(A - \gamma B)\hat{U}(:, 1:K), \quad \hat{B} = \hat{U}(:, 1:K)^{\mathrm{H}} B \hat{U}(:, 1:K). \tag{15}$$

Let $\hat{\omega}_1, \ldots, \hat{\omega}_K$ be the eigenvalues of the matrix pencil $\hat{A} - \lambda\hat{B}$, and let $\hat{\boldsymbol{r}}_1, \ldots, \hat{\boldsymbol{r}}_K$ be the corresponding eigenvectors. Then the approximate eigenvalues inside Γ are given by

$$\hat{\lambda}_i = \gamma + \hat{\omega}_i, \quad i = 1, \ldots, K,$$

and the corresponding eigenvectors are given by

$$\hat{\boldsymbol{x}}_j = \hat{U}(:, 1:K)\hat{\boldsymbol{r}}_j, \quad j = 1, \ldots, K. \tag{16}$$

3 FILTERING FOR A SUBSPACE

In this section, we discuss the properties of the subspace obtained by the numerical quadrature (14) from the view-point of a filter for a subspace.

Here, for simplicity, we consider the case that all the eigenvalues inside Γ are simple, and the inverse of the matrix $zB - A$ is expanded as

$$(zB - A)^{-1} = \sum_{i=1}^{n} \frac{\boldsymbol{x}_i \boldsymbol{y}_i^{\mathrm{H}}}{z - \lambda_i}, \tag{17}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ are the right and left eigenvectors corresponding to the eigenvalue λ_i. This expansion can be generalized to non-Hermitian and nonlinear cases ([2, 3, 14]).

Let $P_i = \boldsymbol{x}_i \boldsymbol{y}_i^{\mathrm{H}} B$, $1 \leq i \leq n$. With the expansion (17), from the residue theorem, we have

$$\begin{aligned} F_k &= \frac{1}{2\pi \mathrm{i}} \int_{\Gamma} z^k (zB - A)^{-1} B \mathrm{d}z \\ &= \sum_{i=1}^{n} \left(\frac{1}{2\pi \mathrm{i}} \int_{\Gamma} \frac{z^k P_i}{z - \lambda_i} \mathrm{d}z \right) \\ &= \sum_{i=1}^{m} \lambda_i^k P_i, \end{aligned}$$

and

$$S_k = F_k V = \sum_{i=1}^{m} \lambda_i^k P_i V.$$

Define a function $\mathcal{F}_k(\lambda)$ as

$$\mathcal{F}_k(\lambda) = \frac{1}{2\pi \mathrm{i}} \int_{\Gamma} \frac{z^k}{z - \lambda} \, \mathrm{d}z.$$

Then

$$\mathcal{F}_k(\lambda_i) = \begin{cases} \lambda_i^k & \lambda_i \in G \\ 0, & \text{otherwise} \end{cases},$$

and S_k is represented as

$$S_k = \sum_{i=1}^{n} \mathcal{F}_k(\lambda_i) P_i V.$$

This equation shows that a projected component associated with P_i in V is filtered with the factor $\mathcal{F}_k(\lambda_i)$. Therefore the function $\mathcal{F}_k(\lambda)$ is regarded to give the factor of filtering with respect to λ.

For the case that the contour integral is approximated by the numerical quadrature, we define the corresponding filter function by

$$\hat{\mathcal{F}}_k(\lambda) = \sum_{j=1}^{N} \frac{\rho w_j \zeta_j^k}{z_j - \lambda}.$$

The following result is obtained.

Theorem 1 *Let λ be a complex number that is located outside Γ. Then the following holds:*

$$\hat{\mathcal{F}}_k(\lambda) = -\nu_{N-1} \eta^{-N+k} \left(1 + \eta^{-1} \sum_{p=0}^{\infty} \frac{\nu_{N+p}}{\nu_{N-1}} \eta^{-p} \right), \tag{18}$$

where $\eta = (\lambda - \gamma)/\rho$ and $\nu_p = \sum_{j=1}^{N} w_j \zeta_j^p$.

Proof. Since $|\eta| = |(\lambda - \gamma)/\rho| > |\zeta_j|$ for $1 \le j \le N$, we have

$$\begin{aligned}
\sum_{j=1}^{N} \frac{\rho w_j \zeta_j^k}{z_j - \lambda} &= \sum_{j=1}^{N} \frac{w_j \zeta_j^k}{(z_j - \gamma)/\rho - (\lambda - \gamma)/\rho} = \sum_{j=1}^{N} \frac{w_j \zeta_j^k}{\zeta_j - \eta} \\
&= \sum_{j=1}^{N} \left(\frac{-1}{\eta} \right) \frac{w_j \zeta_j^k}{1 - \zeta_j/\eta} \\
&= -\sum_{p=0}^{\infty} \left(\eta^{-p-1} \sum_{j=1}^{N} w_j \zeta_j^{p+k} \right).
\end{aligned}$$

Since the quadrature weights $w_1, \ldots w_N$ satisfy

$$\sum_{j=1}^{N} w_j \zeta_j^k = 0, 1, \quad k = 0, \ldots, N-2,$$

we have

$$\begin{aligned}
\hat{\mathcal{F}}_k(\lambda) = \sum_{j=1}^{N} \frac{\rho w_j \zeta_j^k}{z_j - \lambda} &= -\sum_{p=N-1-k}^{\infty} \left(\eta^{-p-1} \sum_{j=1}^{N} w_j \zeta_j^{p+k} \right) \\
&= -\left(\nu_{N-1} \eta^{-N+k} + \sum_{p=0}^{\infty} \nu_{N+p} \eta^{-N+k-1-p} \right).
\end{aligned}$$

Thus we have eqn (18). □

If $|(\lambda-\gamma)/\rho|$ is sufficiently large then the filter $\hat{\mathcal{F}}_k(\lambda)$ is approximated by

$$\hat{\mathcal{F}}_k(\lambda) = \sum_{j=1}^{N} \frac{\rho w_j \zeta_j^k}{z_j - \lambda} \approx -\nu_{N-1} \left(\frac{\lambda-\gamma}{\rho}\right)^{-N+k}. \tag{19}$$

This implies that the eigencomponents corresponding to the eigenvalues located outside Γ in each column vector of $\hat{S}_k = \hat{F}_k V$ are reduced in proportion to the $(-N+k)$-th power of magnitude of the scaled distance $|(\lambda-\gamma)/\rho|$.

Suppose that the integer m' is taken as

$$\left|\nu_{N-1}\left(\frac{\lambda_i-\gamma}{\rho}\right)^{-N+M-1}\right| \leq \delta, \quad m' < i \leq n \tag{20}$$

with small $\delta > 0$. Then, from eqn (18), we have

$$\hat{S}_k = \hat{F}_k V = \sum_{i=1}^{n} \hat{\mathcal{F}}_k(\lambda_i) P_i V = \sum_{i=1}^{m'} \hat{\mathcal{F}}_k(\lambda_i) P_i V + O(\delta).$$

4 EFFICIENT PARAMETER ESTIMATION AND IMPLEMENTATION

4.1 Selection of subspace size

The SS method has some parameters, and the choice of these parameters affects the accuracy and performance of the method. The number of quadrature points N determines the number of systems of linear equations to solve, and consequently N specifies the number of computing nodes to use in parallel computing. Therefore we assume that N is fixed in advance. In practice, N is chosen as $N = 16$ or 32 depending on the number of computing nodes or memory requirements, and it is not necessary to take a large N to reduce the quadrature error as was observed in the previous section.

The parameter M specifies the upper bound of the degree of moments. Increasing M gives a larger subspace size LM. However, the decay factor of the filter depends on $-N+k$ with $0 \leq k \leq M-1$, and a large M diminishes the performance of the filter. Considering a performance of the filer and computational costs, we set $M = N/4$.

The number of column vectors LM of $\hat{S}$ should be taken such that the minimum singular value of $\hat{S}$ becomes sufficiently small. Since M depends on N, we shall extend the number of column vectors of $\hat{S}$ by increasing the number of source vectors L. Since m' is larger than or equal to m, an approximation for m can be used as a lower bound of m'. To predict m, we use the stochastic estimation method proposed in [4, 9]. The number of eigenvalues inside Γ is given by

$$m = \frac{1}{2\pi \mathrm{i}} \int_{\Gamma} \mathrm{trace}((zB-A)^{-1}B)\mathrm{d}z.$$

However, the cost of computing the trace of the matrix inverse is large if the matrices are large. The trace of $(zB-A)^{-1}B$ is approximated by

$$\mathrm{trace}((zB-A)^{-1}B) \approx \left(\frac{1}{L_0}\right)\sum_{i=1}^{L_0} \boldsymbol{v}_i^{\mathrm{T}}(zB-A)^{-1}B\boldsymbol{v}_i,$$

with some integer L_0, where the elements of the sample vectors $\boldsymbol{v}_i \in \mathbb{R}^n$ are taken as -1 or 1 with equal probability. The contour integral is approximated by the N-point numerical quadrature. Thus, the estimated number of eigenvalues $\tilde{m}$ is given by

$$\begin{aligned} m &\approx \sum_{j=1}^{N} \rho w_j \left(\mathrm{trace}((z_jB-A)^{-1}B)\right) \\ &\approx \sum_{j=1}^{N} \rho w_j \left(\left(\frac{1}{L_0}\right)\sum_{i=1}^{L_0} \boldsymbol{v}_i^{\mathrm{T}}(z_jB-A)^{-1}B\boldsymbol{v}_i\right) = \tilde{m}. \end{aligned}$$

Note that

$$\tilde{m} = \sum_{j=1}^{N} \rho w_j \left(\left(\frac{1}{L_0}\right)\sum_{i=1}^{L_0} \boldsymbol{v}_i^{\mathrm{T}}(z_jB-A)^{-1}B\boldsymbol{v}_i\right) = \mathrm{trace}(V^{\mathrm{T}}\hat{S}_0)/L_0,$$

and $\hat{S}_0$ can be used for the stochastic estimation of the number of eigenvalues by setting $V = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{L_0}]$ in the SS method.

Using $\tilde{m}$, we set the approximation of m' as $\kappa\tilde{m}$ with a parameter $\kappa \geq 1$, and consequently we set $L = \lceil m'/M \rceil \approx \lceil \kappa\tilde{m}/M \rceil$, where $\lceil x \rceil$ returns the smallest integer not less than x. When the subspace size LM is not sufficiently large, the minimum singular value $\sigma_{\min}$ of $\hat{S}$ is not small. In this case, we increment L until $\sigma_{\min}$ satisfies the condition $\sigma_{\min} \leq \delta \times \sigma_1$ with small $\delta > 0$. The computation of the singular values of $\hat{S}$ is rather expensive, so we may use the Hankel matrix $\hat{H}$ instead of $\hat{S}$.

4.2 Iterative refinement of a subspace

After setting appropriate L, we apply the Rayleigh-Ritz procedure with $\hat{S}$. The increase of L causes an increase in the size of the projected subspace. It causes an increase in the cost for computing the singular value decomposition of $\hat{S}$ and the solution of the projected eigenvalue problem with matrices $\hat{A}$ and $\hat{B}$. To avoid increasing the size of the projected space, we restrict the size of L, and apply the recurrence refinement described below.

Setting $\hat{S}_0^{(0)} = \hat{S}_0$, and recurrently applying $\hat{F}_0$, we have

$$\hat{S}_0^{(r-1)} = \hat{F}_0\hat{S}_0^{(r-2)} = \cdots = (\hat{F}_0)^{r-1}\hat{S}_0^{(0)}. \tag{21}$$

Using $\hat{S}_0^{(r-1)}$, the output matrix with r refinements is given by

$$\hat{S}_k^{(r)} = \hat{F}_k\hat{S}_0^{(r-1)}, \quad k = 0, \ldots, M-1, \tag{22}$$

and $\hat{S}^{(r)} = [\hat{S}_0^{(r)}, \ldots, \hat{S}_{M-1}^{(r)}]$. The corresponding filter is given by $(\mathcal{F}_k(\lambda))^r$ and is approximated by

$$(\hat{\mathcal{F}}_k(\lambda))^r \approx (-\nu_{N-1})^r \left(\frac{\lambda - \gamma}{\rho}\right)^{-r(N-k)}.$$

Therefore the recurrence application of the filter process makes the decay factor of the filter smaller. The refinement is terminated if the smallest singular value of $\hat{S}^{(r)}$ becomes sufficiently small with a threshold $\delta > 0$.

In the case that some residuals of the obtained approximate eigenpairs are not small enough for a given tolerance, we can brush up the resulting approximate eigenpairs by setting the source matrix of the SS method as

$$V = [\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_{\hat{m}}]C,$$

where $C \in \mathbb{R}^{\hat{m} \times L}$ for which the elements are given by random numbers, and $\hat{\boldsymbol{x}}_1, \ldots, \hat{\boldsymbol{x}}_{\hat{m}}$ are the selected eigenvectors that are regarded as the approximate eigenvectors with respect to the eigenvalues inside Γ. This refinement technique using approximate eigenvectors for the source matrix V is used in [13].

4.3 Linear solvers for a complex shift

When A and B are real symmetric, the shifted matrix $C = zB - A$ with a complex shift z is complex symmetric. Therefore, a linear solver for complex symmetric systems is used to solve the system

$$(zB - A)Y = BV.$$

For a direct solver, the modified Cholesky factorization saves computational costs for factorization. For an iterative solver, Krylov subspace methods for complex symmetric systems, such as the COCG method, can be used.

When Γ is symmetric with respect to the real axis, the quadrature points are set as $z_{N-j+1} = \overline{z}_j$, $j = 1, \ldots, N/2$. Then, for real matrices A and B, the solutions at z_{N-j+1} are obtained by

$$Y_{N-j+1} = (z_{N-j+1}B - A)^{-1}BV = \overline{Y}_j$$

without any computations on z_{N-j+1}.

When A and B are Hermitian, we use the property

$$(z_j B - A)^{\mathrm{H}} = \overline{z}_j B^{\mathrm{H}} - A^{\mathrm{H}} = \overline{z}_j B - A.$$

If the LU factorization at z_j is calculated as $z_j B - A = LU$ then we have

$$Y_{N-j+1} = (\overline{z}_j B - A)^{-1}V = (U^{\mathrm{H}}L^{\mathrm{H}})^{-1}V.$$

Therefore the LU factorization at z_j can be used for the calculation at z_{N-j+1}.

5 NUMERICAL EXAMPLES

In this section we show some numerical examples. The computations are performed in MATLAB 8.0.0. in double precision arithmetic. Random numbers are generated by the function `rand`, and the projected small eigenvalue problems are solved by `eig`. The systems of linear equations are solved by `lu`. The factorized matrices are held during the computation, and only triangular solves are applied in the recurrence refinements.

In the following examples, the quadrature points are set by eqn (11) and the corresponding weights are set by eqn (12) with $\alpha = 0.1$. The relative residual for the eigenpair $(\hat{\lambda}_i, \hat{\boldsymbol{x}}_i)$ is calculated by

$$\mathrm{res}_i = \frac{\|A\hat{\boldsymbol{x}}_i - \hat{\lambda} B\hat{\boldsymbol{x}}_i\|_2}{\|A\hat{\boldsymbol{x}}_i\|_2 + |\hat{\lambda}_i|\|B\hat{\boldsymbol{x}}_i\|_2}.$$

We removed the eigenvalues with $\mathrm{res}_i \geq 10^{-2}$ inside Γ as spurious eigenvalues.

Example 1. The matrices A and B are taken from BCSSTK11 and BCSSTM11 of the BCS Structural Engineering Matrices in Matrix Market [8]. A and B are real symmetric and B is positive definite. The matrix dimension is $n = 1,473$ with $34,241$ nonzero entries. The parameters are set as $N = 16$ and $L = 16$. The domain is set as $\gamma = 10^3$ and $\rho = 5 \times 10^2$. In this example, L is fixed, and the iterative refinement is not applied.

The results are shown in Table 1. The number of singular values that are greater than $\delta = 10^{-12}$ is $K = 18$. Therefore 18 eigenvalues are obtained from the projected problem, of which 7 eigenvalues are located inside Γ. The residuals of the eigenvalues located inside Γ are small, however the residuals of the eigenvalues located outside Γ are related to the scaled distance $|\eta_i| = |(\lambda_i - \gamma)/\rho|$.

Example 2. In this example, we apply the iterative refinement defined by (21) and (22). The matrices A and B are same as Example 1. The parameters are set as $N = 16$, $L = 16$, and the domain is set as $\gamma = 2 \times 10^5$ and $\rho = 2 \times 10^4$.

In Figure 1, The singular values of $\hat{S}^{(r)}$ at r-th refinement are shown. We can see that the ratio of the minimum singular value and the maximum singular value increases by the iterative refinement. After two refinement, the minimum singular value becomes small enough. Table 2 shows the residuals of the calculated eigenvalues located inside Γ. In the table, the notation $\mathrm{mean}(\mathrm{res}_i)$ is given by the geometric mean of the residuals defined by

$$\mathrm{mean}(\mathrm{res}_i) = \left(\prod_{i=1}^{\hat{m}} \mathrm{res}_i \right)^{1/\hat{m}},$$

where $\hat{m}$ is the number of calculated eigenvalues located inside Γ.

Example 3. In this example, we use the stochastic estimation of the number of eigenvalues in Γ to set the initial L, and the iterative refinement of $\hat{S}$ is also used. The matrices A and B are same as Example 1. The parameters are set as $N = 16$

Table 1: Results of Example 1.

i	$\hat{\lambda}_i$	res_i	$\lvert\eta_i\rvert$
1	2345.08723030540	3.0×10^{-01}	3.3
2	2398.81729572773	3.3×10^{-01}	3.2
3	2628.94468521146	4.9×10^{-02}	2.7
4	2723.54384863656	1.4×10^{-02}	2.6
5	3383.97540832681	3.8×10^{-08}	1.2
6	3501.25383608303	9.0×10^{-11}	–
7	3561.62085364923	2.7×10^{-11}	–
8	3629.33212408543	4.0×10^{-11}	–
9	3796.50112783802	4.8×10^{-11}	–
10	4022.39762561787	3.1×10^{-11}	–
11	4100.71462746484	1.5×10^{-11}	–
12	4175.86741050601	3.4×10^{-11}	–
13	4770.43635520514	5.1×10^{-06}	1.5
14	5071.04303115872	1.6×10^{-04}	2.1
15	5185.64239506030	3.5×10^{-03}	2.4
16	5325.06302301902	1.6×10^{-02}	2.7
17	5608.24863853754	1.0×10^{-01}	3.2
18	5874.78406307974	6.6×10^{-01}	3.8

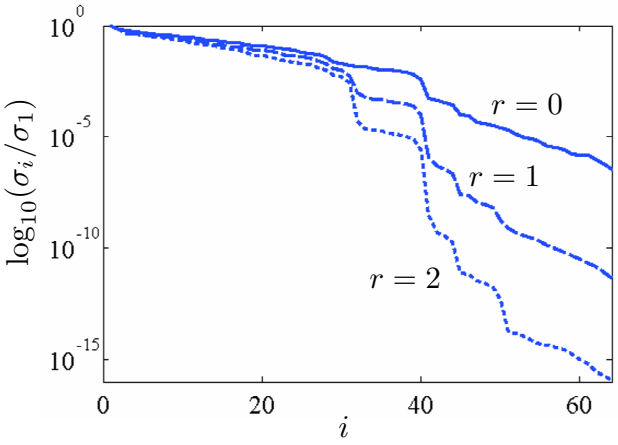

Figure 1: Singular values in r-th iterative refinement.

Table 2: Results in Example 2.

♯refinement	$\min(\mathrm{res}_i)$	$\mathrm{mean}(\mathrm{res}_i)$	$\max(\mathrm{res}_i)$
0	1.8×10^{-07}	9.1×10^{-06}	1.7×10^{-04}
1	3.8×10^{-12}	1.1×10^{-10}	1.1×10^{-09}
2	1.9×10^{-14}	7.2×10^{-13}	1.2×10^{-11}

and $\delta = 10^{-12}$. The domain is set as $\gamma = 2 \times 10^5$ and $\rho = 2 \times 10^4$. The number of sample vectors for the stochastic estimation of the number of eigenvalues in Γ is set as $L_0 = 16$. The initial guess of the number of column vectors of V is given by $L = \lceil 2\tilde{m}/M \rceil$, i.e. $\kappa = 2$.

In Table 3, we show the residuals of the eigenvalues located inside Γ. The number of eigenvalues in Γ is $m = 30$ and the estimated number of eigenvalues is $\tilde{m} = 32.7$. The number of iterative refinement is 2.

Example 4. The matrices A and B are taken from BCSSTK13 and BCSSTM13. A and B are real symmetric and B is positive semi-definite. The matrix dimension is $n = 2,003$ with $83,883$ nonzero entries. The parameters are the same as in Example 3. The domain is set as $\gamma = 10^6$ and $\rho = 4 \times 10^5$.

In Table 4, we show the residuals of the eigenvalues located inside Γ. The number of eigenvalues in Γ is $m = 73$ and the estimated number of eigenvalues is $\tilde{m} = 77.7$. The number of column vectors of V is $L = 55$ and the number of iterative refinement is 2. The maximum, mean and minimum residuals are 2.1×10^{-10}, 8.6×10^{-12} and 2.7×10^{-13}, respectively. We can obtain the eigenpairs in the given domain with the same initial parameters.

Example 5. The matrices A and B are derived from molecular orbital calculations for a model DNA [19]. A and B are real symmetric and B is positive definite. The matrix dimension is $n = 1,980$ with $728,080$ nonzero entries. The parameters are the same as in Example 3 and 4. The domains are given by the intervals $[-0.20, -0.15]$, $[-0.25, -0.15]$, $[-0.30, -0.15]$, $[-0.35, -0.15]$, $[-0.40, -0.15]$, $[-0.45, -0.15]$ and $[-0.50, -0.15]$.

In Table 5, we show the number of eigenvalues in the given interval (♯ev), the estimated number of eigenvalues (Est. ♯ev), the number of column vectors of V (L), the number of iterative refinement (♯refinement) and the maximum residuals of eigenvalues in the interval ($\max(\mathrm{res}_i)$). In the results, the maximum residuals are sufficiently small by estimating appropriate L and the number of iterative refinement for each domain.

6 CONCLUSIONS

In this paper, we considered an eigensolver for computing the eigenvalues in a given domain and the corresponding eigenvectors of large-scale matrix pencils. The

Table 3: Results in Example 3.

i	$\hat{\lambda}_i$	res_i	i	$\hat{\lambda}_i$	res_i
1	181301.355856	3.0×10^{-12}	16	206423.180896	2.2×10^{-12}
2	181353.297523	8.2×10^{-13}	17	207887.176182	4.5×10^{-12}
3	185810.063953	3.1×10^{-12}	18	209720.799807	1.2×10^{-12}
4	185856.309721	2.2×10^{-12}	19	211359.608331	1.6×10^{-12}
5	189076.069885	1.3×10^{-12}	20	211525.005509	1.2×10^{-12}
6	190580.274469	1.7×10^{-12}	21	211778.728062	1.0×10^{-12}
7	191916.768828	4.6×10^{-12}	22	211798.736010	1.4×10^{-12}
8	192249.997887	6.2×10^{-12}	23	214623.208612	1.7×10^{-12}
9	192450.352262	8.8×10^{-12}	24	215071.649241	1.2×10^{-12}
10	195110.875562	8.9×10^{-13}	25	216638.323804	1.1×10^{-12}
11	195362.147280	1.6×10^{-12}	26	216782.856683	5.0×10^{-13}
12	195522.864186	2.1×10^{-12}	27	216875.914785	4.9×10^{-13}
13	196453.465229	9.5×10^{-13}	28	217120.082795	1.4×10^{-12}
14	196779.318796	1.1×10^{-12}	29	217475.120411	1.3×10^{-13}
15	203358.448118	5.6×10^{-13}	30	217803.381541	5.8×10^{-13}

Table 4: Results in Example 4.

i	$\hat{\lambda}_i$	res_i	i	$\hat{\lambda}_i$	res_i
1	602514.527692	1.2×10^{-12}	38	964884.799128	2.7×10^{-11}
2	605178.148251	2.1×10^{-11}	39	971058.404128	3.0×10^{-11}
3	616657.672408	5.2×10^{-12}	40	973436.179279	9.5×10^{-12}
4	623758.141144	2.2×10^{-11}	41	981630.285398	3.0×10^{-11}
5	641859.031825	1.3×10^{-12}	42	985027.771304	6.4×10^{-11}
⋮	⋮	⋮	⋮	⋮	⋮
33	924036.280859	7.0×10^{-11}	70	1332026.80482	7.5×10^{-12}
34	927854.750782	4.3×10^{-12}	71	1348423.99041	4.7×10^{-13}
35	941218.254886	9.8×10^{-12}	72	1372139.51897	8.4×10^{-12}
36	942132.221466	1.5×10^{-12}	73	1379152.51378	1.1×10^{-12}
37	960716.560772	1.0×10^{-11}			

Table 5: Results in Example 5.

Interval	♯ev	Est. ♯ev	L	♯refinement	$\max(\mathrm{res}_i)$
$[-0.20, -0.15]$	22	23.9	16	1	2.8×10^{-13}
$[-0.25, -0.15]$	78	80.0	40	2	2.1×10^{-12}
$[-0.35, -0.15]$	198	196.3	99	2	8.5×10^{-12}
$[-0.40, -0.15]$	262	270.1	136	2	1.7×10^{-12}
$[-0.45, -0.15]$	333	327.9	164	2	9.0×10^{-12}
$[-0.50, -0.15]$	406	410.5	206	2	9.4×10^{-12}

Sakurai-Sugiura (SS) method is an eigensolver based on complex moments given by the contour integrals of the matrix inverses with several shift points.

Some numerical properties of the method were presented from the view-point of a filter for a subspace. According to the results, efficient parameter estimation techniques were shown. The contour integral for a matrix inverse is regarded as a filter for an eigensubspace. When the contour integral is approximated by a numerical quadrature, the quadrature error causes contamination of the eigencomponents corresponding to the eigenvalues located outside of the contour path. We demonstrated the efficiency of our method with numerical experiments.

In the numerical experiments, we used a sparse direct solver. The use of iterative linear solvers for multiple right-hand sides such as block Krylov subspace solvers are useful because our eigensolver requires very small number of iterative refinement. The combination of block iterative linear solvers and performance evaluation with such solvers in practical problems remaining as future work. We are implementing a software package of the SS method for large-scale parallel computing. Applications of practical problems and performance evaluations of the presented method in a massively parallel computing environment also remain as our future work.

ACKNOWLEDGMENT

This research was partially supported by the Core Research for Evolutional Science and Technology (CREST) Project of JST and the Grant-in-Aid for Scientific Research of Ministry of Education, Culture, Sports, Science and Technology, Japan, Grant number: 23105702.

References

[1] J. Asakura, T. Sakurai, H. Tadano, T. Ikegami and K. Kimura, A numerical method for nonlinear eigenvalue problems using contour integrals, *JSIAM Letters* **1** (2009) 52–55.

[2] J. Asakura, T. Sakurai, H. Tadano, T. Ikegami and K. Kimura, A numerical method for polynomial eigenvalue problems using contour integral, *Japan J. Indust. Appl. Math.* **27** (2010) 73–90.

[3] W. J. Beyn, An integral method for solving nonlinear eigenvalue problems, *Lin. Alg. Appl.* **436** (2012) 3839–3863.

[4] Y. Futamura, H. Tadano and T. Sakurai, Parallel stochastic estimation method of eigenvalue distribution, *JSIAM Letters* **2** (2011) 27–30.

[5] Y. Futamura, T. Sakurai, S. Furuya and J.-I. Iwata, Efficient algorithm for linear systems arising in solutions of eigenproblems and its application to electronic-structure calculations, Proc. 10th International Meeting on High-Performance Computing for Computational Science (VECPAR 2012) (accepted).

[6] T. Ikegami, T. Sakurai and U. Nagashima, A filter diagonalization for generalized eigenvalue problems based on the Sakurai-Sugiura projection method, *J. Comput, Appl. Math.* **233** (2010) 1927–1936.

[7] T. Ikegami and T. Sakurai, Contour integral eigensolver for non-Hermitian systems: a Rayleigh-Ritz-type approach, *Taiwanese J. Math.* **14** (2010) 825–837.

[8] Matrix Market, `http://math.nist.gov/MatrixMarket/`.

[9] Y. Maeda, Y. Futamura, T. Sakurai, Stochastic estimation method of eigenvalue density for nonlinear eigenvalue problem on the complex plane, *JSIAM Letters* **3** (2011) 61–64.

[10] T. Mizusaki, K. Kaneko, M. Honma and T. Sakurai, Filter diagonalization of shell-model calculations, *Phys. Rev. C* **82** (2010) 024310.

[11] M. Naito, H. Tadano, T. Sakurai, A modified block IDR(s) method for computing high accuracy solutions, *JSIAM Letters* **4** (2012) 25–28.

[12] H. Ohno, Y. Kuramashi, H. Tadano and T. Sakurai, A quadrature-based eigensolver with a Krylov subspace method for shifted linear systems for Hermitian eigenproblems in lattice QCD, *JSIAM Letters* **2** (2010) 115–118.

[13] E. Polizzi, Density-matrix-based algorithms for solving eigenvalue problems, *Phys. Rev. B* **79** (2009) 115112.

[14] T. Sakurai and H. Sugiura, A projection method for generalized eigenvalue problems, *J. Comput. Appl. Math.* **159** (2003) 119–128.

[15] T. Sakurai and H. Tadano, CIRR: a Rayleigh-Ritz type method with contour integral for generalized eigenvalue problems, *Hokkaido Math. J.* **36** (2007) 745–757.

[16] T. Sakurai, J. Asakura, H. Tadano and T. Ikegami, Error analysis for a matrix pencil of Hankel matrices with perturbed complex moments, *JSIAM Letters* **1** (2009) 76–79.

[17] T. Sakurai, H. Tadano and Y. Kuramashi, Application of block Krylov subspace algorithms to the Wilson-Dirac equation with multiple right-hand sides in lattice QCD, *Comput. Phys. Commun.* **181** (2010) 113–117.

[18] H. Tadano, T. Sakurai and Y. Kuramashi, Block BiCGGR: A new block Krylov subspace method for computing high accuracy solutions, *JSIAM Letters* **1** (2009) 44–47.

[19] T. Watanabe, Y. Inadomi, H. Umeda, K. Fukuzawa, S. Tanaka, T. Nakano and U. Nagashima, Fragment molecular orbital (FMO) and FMO-MO calculations of DNA: accuracy validation of energy and interfragment interaction energy, *J. Comput. Theo. Nanoscience* **6** (2009) 1328–1337.

[20] I. Yamazaki, H. Tadano, T. Sakurai and T. Ikegami, Performance comparison of parallel eigensolvers based on a contour integral method and a Lanczos method, *Parallel Computing* (in press).

[21] S. Yokota and T. Sakurai, A projection method for nonlinear eigenvalue problems using contour integrals, JSIAM Letters (accepted).