# Filter Diagonalization Method by Using a Polynomial of a Resolvent as the Filter for a Real Symmetric-Definite Generalized Eigenproblem

**Hiroshi Murakami**

**Abstract** For a real symmetric-definite generalized eigenproblem of size $N$ matrices $A\mathbf{v} = \lambda B\mathbf{v}$ ($B > 0$), we solve those pairs whose eigenvalues are in a real interval $[a, b]$ by the filter diagonalization method.

In our present study, the filter which we use is a real-part of a polynomial of a resolvent: $F = \mathrm{Re} \sum_{k=1}^{n} \gamma_k \{R(\rho)\}^k$. Here $R(\rho) = (A - \rho B)^{-1} B$ is the resolvent with a non-real complex shift $\rho$, and $\gamma_k$ are coefficients. In our experiments, the (half) degree $n$ is 15 or 20.

By tuning the shift $\rho$ and coefficients $\{\gamma_k\}$ well, the filter passes those eigenvectors well whose eigenvalues are in a neighbor of $[a, b]$, but strongly reduces those ones whose eigenvalues are separated from the interval.

We apply the filter to a set of sufficiently many $B$-orthonormal random vectors $\{\mathbf{x}^{(\ell)}\}$ to obtain another set $\{\mathbf{y}^{(\ell)}\}$. From both sets of vectors and properties of the filter, we construct a basis which approximately spans an invariant subspace whose eigenvalues are in a neighbor of $[a, b]$. An application of the Rayleigh-Ritz procedure to the basis gives approximations of all required eigenpairs.

Experiments for banded problems showed this approach worked in success.

## 1 Introduction

We solve pairs of a real symmetric-definite generalized eigenproblem of size $N$ matrices $A$ and $B$ as:

$$A\,\mathbf{v} = \lambda B\,\mathbf{v} \tag{1}$$

whose eigenvalues are in the specified interval $[a, b]$ by the filter diagonalization method [16]. We define for this kind of eigenproblem, the resolvent with a

H. Murakami (✉)
Tokyo Metropolitan University, 1-1 Minami-Osawa, Hachi-Oji, Tokyo 192-0397, Japan
e-mail: mrkmhrsh@tmu.ac.jp



T. Sakurai et al. (eds.), *Eigenvalue Problems: Algorithms, Software and Applications in Petascale Computing*, Lecture Notes in Computational Science and Engineering 117, https://doi.org/10.1007/978-3-319-62426-6_14

complex-valued shift $\rho$ as:

$$R(\rho) = (A - \rho B)^{-1} B \,. \tag{2}$$

In our previous papers and reports [5–9] and papers of others [1–3], the filter studied or used was a real-part of a complex linear combination of resolvents with complex shifts:

$$F = c_\infty I + \mathrm{Re} \sum_{k=1}^{n} \gamma_k R(\rho_k) \,. \tag{3}$$

Here, $c_\infty$ is a real coefficient, $\gamma_k$, $k$=1, 2, …, $n$ are complex coefficients, and $I$ is the identity matrix. (The reason we take the real-part is to halve the cost of calculation by using the complex-conjugate symmetry). For example, from $n = 6$ to $n = 16$ (or more) resolvents were used.

For a given set of size $N$ column vectors $X$, the action of the resolvent $Y \leftarrow R(\rho)X$ reduces to solve an equation $CY = BX$ for a set of column vectors $Y$. Here, the coefficient matrix is $C = A - \rho B$. When $C$ is banded, the equation may be solved by some direct method using matrix factorization. When $C$ is random sparse, the equation is solved by some iterative method using incomplete matrix factorization. In the application of the filter, the matrix factorization is a large portion of the calculation. The total amount of memory to store the factor is also a very severe constraint in the calculations of large size problems. When many resolvents are used, the total amount of memory requirements is proportional to the number of resolvents applied concurrently.

There are also different but similar approaches and successful studies which are based on the contour integrals and moment calculations [4, 13–15], which also uses many resolvents whose shifts correspond to the integration points.

In this report of study (and in our several previous reports [10–12]), we used only a single resolvent with a complex shift and constructed the filter which is a real-part of a polynomial of the resolvent as:

$$F = c_\infty I + \mathrm{Re} \sum_{k=1}^{n} \gamma_k \, (R(\rho))^k \,. \tag{4}$$

We made some numerical experiments on a test problem to check if this approach is really applicable.

## 2 Present Approach: Filter is Real-Part of Polynomial of Resolvent

For the large eigenproblem, we assume the severest constraint is the amount of memory requirements. Thus, in our present study of the filter diagonalization method, we use a single resolvent in the filter rather than many ones. The filter we use is a real-part of a polynomial of the resolvent. In the filter operation, the same resolvent is applied as many times as the degree of the polynomial. Each time, the application of the resolvent to a set of vectors reduces to the set of solutions of simultaneous linear equations of the same coefficient matrix. To solve the set of simultaneous equations, the coefficient matrix is factored once and the factor is stored. The stored factor is used many times when the resolvent is applied. By the use of a single resolvent rather than many ones, even the transfer function of the filter cannot be made in good shape, but in exchange we obtain advantages of lower memory requirement and reduced number of matrix factorization.

### *2.1 Filter as a Polynomial of a Resolvent and Its Transfer Function*

We consider a real symmetric-definite generalized eigenproblem of size $N$ matrices $A$ and $B$:

$$A\mathbf{v} = \lambda B\mathbf{v}, \text{ where } B > 0\,. \tag{5}$$

The resolvent with a non-real complex shift $\rho$ is:

$$R(\rho) = (A - \rho B)^{-1} B\,. \tag{6}$$

For any pair of the eigenproblem $(\lambda, \mathbf{v})$, we have:

$$R(\rho)\,\mathbf{v} = \frac{1}{\lambda - \rho}\mathbf{v}\,. \tag{7}$$

The filter $F$ is a real-part of a degree $n$ polynomial of the resolvent:

$$F = c_\infty I + \mathrm{Re} \sum_{k=1}^{n} \gamma_k \{R(\rho)\}^k\ . \tag{8}$$

Here, $c_\infty$ is a real number, and $\gamma_k$ are complex numbers. This filter is a real linear operator. For any eigenpair $(\lambda, \mathbf{v})$, we have:

$$F\,\mathbf{v} = f(\lambda)\,\mathbf{v}\,. \tag{9}$$

Here, $f(\lambda)$ is the transfer function of the filter $F$ which is a real rational function of $\lambda$ of degree $2n$ as:

$$f(\lambda) = c_\infty + \mathrm{Re} \sum_{k=1}^{n} \frac{\gamma_k}{(\lambda - \rho)^k} \tag{10}$$

whose only poles are located at a non-real complex number $\rho$ and its complex conjugate (both poles are $n$-th order).

### 2.1.1 Transfer Function $g(t)$ in Normalized Coordinate $t$

We are to solve those pairs whose eigenvalues are in the specified real interval $[a, b]$. By the linear transformation which maps between $\lambda \in [a, b]$ and $t \in [-1, 1]$, the normalized coordinate $t$ of $\lambda$ is defined as $\lambda = \frac{a+b}{2} + \left(\frac{b-a}{2}\right) t$. We call the interval $t \in [-1, 1]$ as the *passband*, intervals $\mu \leq |t|$ as *stopbands*, and intervals $1 < |t| < \mu$ which are between the passband and stopbands as *transition-bands*.

The transfer function $g(t)$ in the normalized coordinate $t$ is defined by:

$$g(t) = f(\lambda)\,. \tag{11}$$

To the transfer function $g(t)$, we request the following conditions:

1. $|g(t)| \leq g_{\mathrm{stop}}$ when $t$ is in stopbands. Here, $g_{\mathrm{stop}}$ is a very small positive number.
2. $g_{\mathrm{pass}} \leq g(t)$ when and only when $t$ is in the passband. Here, $g_{\mathrm{pass}}$ is a number much larger than $g_{\mathrm{stop}}$. (The upper-bound of $g(t)$ is about unity. By re-normalization of the filter, which is the multiplication of a constant, we may set the upper-bound to unity later.)

For convenience, we also restrict $g(t)$ to an even function. Then the poles are pure imaginary numbers (Fig. 1).

We just placed the poles of $g(t)$ at pure imaginary numbers $t = \pm\sqrt{-1}$, and the expression of $g(t)$ is written as:

$$g(t) = c'_\infty + \mathrm{Re} \sum_{k=1}^{n} \frac{\alpha_k}{(1 + t\sqrt{-1}\,)^k}\,. \tag{12}$$

For this expression, to make $g(t)$ an even function, we restrict coefficients $\alpha_k$, $k$=1, 2, …, $n$ as real numbers. The real coefficients are so tuned to make the shape of $g(t)$ satisfies the following two conditions: (1) In the passband $|t| < 1$, the value of $g(t)$ is close to 1, (2) In stopbands $\mu < |t|$, the magnitude of $g(t)$ is very small. (See, Fig. 2). In our present study, coefficients are optimized by a method which is similar to the least-square method.

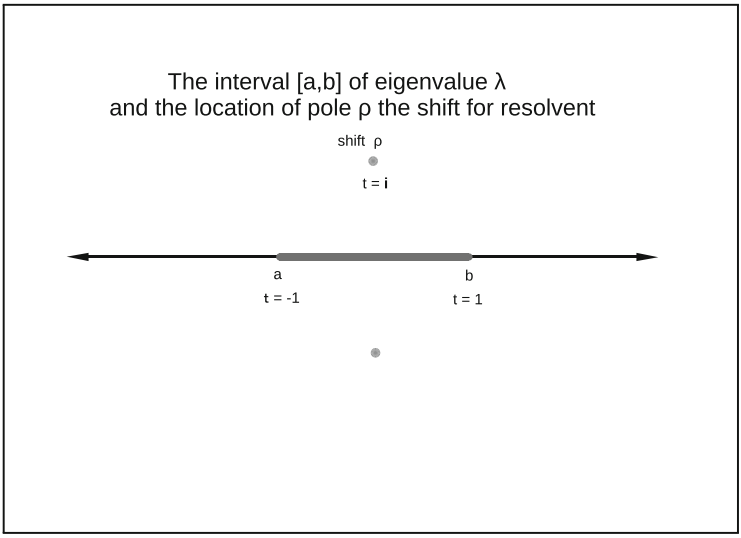


**Fig. 1** Specified interval of eigenvalue and location of poles (shifts)

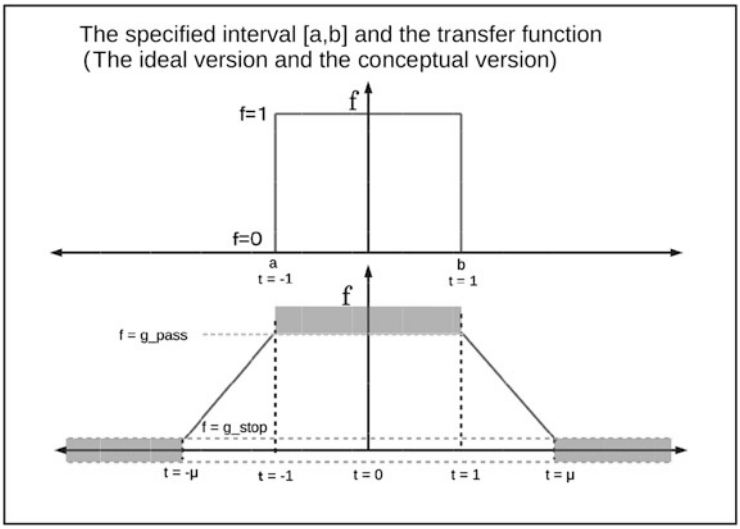


**Fig. 2** Shapes of transfer functions (ideal and conceptual)

## 2.2 Construction of Filter from Transfer Function

We construct the filter operator $F$ from the transfer function $g(t)$.

Since

$$g(t) = c'_{\infty} + \mathrm{Re} \sum_{k=1}^{n} \frac{\alpha_k}{\left(1 + t\sqrt{-1}\right)^k}, \tag{13}$$

$$f(\lambda) = c_{\infty} + \mathrm{Re} \sum_{k=1}^{n} \frac{\gamma_k}{(\lambda - \rho)^k}, \tag{14}$$

and also from both two relations: $f(\lambda) = g(t)$, $\lambda = \dfrac{a+b}{2} + \left(\dfrac{b-a}{2}\right) t$,
we have relations between coefficients and the value of the shift.

$$c'_{\infty} = c_{\infty}, \tag{15}$$

$$\gamma_k = \alpha_k \left(-\sqrt{-1}\right)^k \left(\frac{b-a}{2}\right)^k \text{ , } k=1, 2, \ldots, n \text{ ,} \tag{16}$$

$$\rho = \frac{a+b}{2} + \frac{b-a}{2}\sqrt{-1}\,. \tag{17}$$

Here after, we simply set the transfer rate at infinity $c_{\infty}$ to zero, thus our filter operator is:

$$F = \mathrm{Re} \sum_{k=1}^{n} \gamma_k \, \{R(\rho)\}^k \,. \tag{18}$$

When the half-width of the interval $\frac{b-a}{2}$ is a large or a small number, then coefficients $\gamma_k = \left(-\frac{b-a}{2}\sqrt{-1}\right)^k \alpha_k$ for higher $k$-th terms might get unnecessary floating point number overflows or underflows, which can be avoided by changing the expression of the filter as:

$$F = \mathrm{Re} \sum_{k=1}^{n} \alpha_k \left\{ \left(-\sqrt{-1}\right) \frac{b-a}{2} R(\rho) \right\}^k \,. \tag{19}$$

## 2.3 Procedure of Filter Operation

Here, we show the procedure of the filter operation, the action of the filter to a given set of vectors.

**Fig. 3** Procedure of filter operation $Y \leftarrow F\,X$

```
W ← X ;
Y ← 0 ;
for k := 1 to n do begin
  Z ← R(ρ) W ;
  W ← (−√−1) (b−a)/2 Z ;
  Y ← Y + α_k Re W
end
```

Our filter $F$ is specified by the degree $n$, a complex shift $\rho = \frac{a+b}{2} + \frac{b-a}{2}\sqrt{-1}$ and real coefficients $\alpha_k$, $k$=1, 2, . . ., $n$ as the above expression (19).

Let $X$ and $Y$ are sets of $m$ real column vectors of size $N$ which are represented as real $N\times m$ matrices. Then, the filter operation $Y \leftarrow F\,X$ can be calculated by a procedure shown in Fig. 3. (In the procedure, $W$ and $Z$ are complex $N\times m$ matrices for work.)

### *2.4 Implementation of Resolvent*

To calculate the action of a resolvent $Z \leftarrow R(\rho)\,W$, we first calculate $B\,W$ from $W$, then solve the equation $C\,Z = B\,W$ for $Z$. Here, the coefficient is $C = A - \rho B$. Since both matrices $A$ and $B$ are real symmetric, $C$ is complex symmetric ($C^T = C$). When both matrices $A$ and $B$ are banded, $C$ is also banded. In our present experiments, the complex modified-Cholesky method without pivoting for banded system is used for the banded complex symmetric matrix $C$, even there might be a potential risk of numerical instability.

In the calculation of Rayleigh quotient inverse-iteration which refines approximated eigenpairs, the shifted matrix is real symmetric but indefinite and very close to singular, therefore the simultaneous linear equation is solved carefully by the banded $LU$ decomposition with partial pivoting without using the symmetry.

## 3 Filter Design

Our filter is a real-part of a polynomials of a resolvent. The coefficients of the polynomial are determined by a kind of least-square method, which is the minimization of the weighted sum of definite integrals in both passband and stopbands. The definite integrals are weighted integrations of square of errors of the transfer function from ideal one. The minimization condition gives a system of linear equation with a symmetric positive definite matrix. However, the equations is numerically highly ill-conditioned. Therefore, to determine accurate coefficients in double precision, we have to make quadruple precision calculation both in the

generation of the system of linear equations and in the solution of the generated system by using regularization.

## *3.1 Design by LSQ Method*

We show a method to tune $\alpha_k$, the coefficients of $g(t)$, by least square like method. First, we make the change of variable from $t$ to $\theta$ as $t \equiv \tan\theta$, and let $h(\theta) \equiv g(t)$.

$$h(\theta) = \sum_{k=1}^{n} \alpha_k \cos(k\theta) (\cos\theta)^k . \tag{20}$$

Since $h(\theta)$ is an even function, it is sufficient to consider only in $\theta \in [0, \infty)$. The condition of passband is also considered in $\theta \in [0, \frac{\pi}{4}]$.

### 3.1.1 Method-I

$J_{\text{stop}}$ and $J_{\text{pass}}$ are the integrals in the stopband and in the passband (with weight 1) of the square of difference of the transfer function $h(\theta)$ from the ideal one.

We choose a positive small number $\eta$ and minimize $J \equiv J_{\text{stop}} + \eta J_{\text{pass}}$.

For intervals $[0, 1]$ and $[\mu, \infty)$ of $t$ correspond to intervals $[0, \pi/4]$ and $[\tan^{-1}\mu, \pi/2)$ of $\theta$, respectively (these give endpoints of definite integrals for (half of) passband and a stopband). Then we have:

$$J_{\text{stop}} \equiv \int_{\tan^{-1}\mu}^{\pi/2} \{h(\theta)\}^2 d\theta = \sum_{p,q=1}^{n} \alpha_p \mathscr{A}_{p,q} \alpha_q , \tag{21}$$

$$J_{\text{pass}} \equiv \int_{0}^{\pi/4} \{1 - h(\theta)\}^2 d\theta = \sum_{p,q=1}^{n} \alpha_p \mathscr{B}_{p,q} \alpha_q - 2\sum_{p=1}^{n} \alpha_p \mathscr{B}_{p,0} + \text{const}. \tag{22}$$

Here,

$$\mathscr{A}_{p,q} \equiv \int_{\tan^{-1}\mu}^{\pi/2} \cos(p\theta) \cos(q\theta) (\cos\theta)^{p+q} d\theta , \tag{23}$$

$$\mathscr{B}_{p,q} \equiv \int_{0}^{\pi/4} \cos(p\theta) \cos(q\theta) (\cos\theta)^{p+q} d\theta . \tag{24}$$

We calculated numerical values of these definite integrals by analytic closed formulae.

We can easily show that $\cos(p\theta)\cos(q\theta)(\cos\theta)^{p+q}$

$$= 2^{-(p+q+2)}(1+e^{-2ip\theta})(1+e^{-2iq\theta})(1+e^{2i\theta})^{p+q}$$

$$= 2^{-(p+q+2)}(1+e^{-2ip\theta})(1+e^{-2iq\theta})\sum_{k=0}^{p+q}\binom{p+q}{k}e^{2ik\theta}$$

$$= 2^{-(p+q+2)}\sum_{k=0}^{p+q}\binom{p+q}{k}\{e^{2ik\theta}+e^{2i(k-p)\theta}+e^{2i(k-q)\theta}+e^{2i(k-p-q)\theta}\},$$

where $i$ denotes the imaginary unit $\sqrt{-1}$.

We define for an integer $\ell$ and real number $a$ and $b$:

$$T_\ell \equiv \int_a^b \cos 2\ell\theta\, d\theta$$

$$= \begin{cases} b-a & (\ell = 0) \\ (\sin 2\ell b - \sin 2\ell a)/(2\ell) = \{\sin\ell(b-a)\cdot\cos\ell(b+a)\}/\ell & (\textit{otherwise}) \end{cases}$$

Then for integers $p$ and $q$, we have:

$$\int_a^b \cos(p\theta)\cos(q\theta)(\cos\theta)^{p+q}\,d\theta = \frac{1}{2^{p+q+2}}\sum_{k=0}^{p+q}\binom{p+q}{k}\{T_k + T_{k-p} + T_{q-k} + T_{p+q-k}\},$$

which has a symmetry for $p \leftrightarrow q$. We can use another symmetry that $T_k + T_{k-p}$ and $T_{q-k} + T_{p+q-k}$ is interchanged when $k \to p+q-k$. Since $|T_k| \le 1/k$, we calculate the sum so that terms are added in ascending order of magnitudes of binomial coefficients so to reduce rounding errors. Let $w_k \equiv T_k + T_{k-p}$ and $m \equiv p+q$, $c_k \equiv \binom{m}{k}$, the value of integral $v$ is calculated as in Fig. 4.

The minimization condition of $J \equiv J_{\text{stop}} + \eta J_{\text{pass}}$ is, if we set $b_p \equiv \mathscr{B}_{p,0}$, reduces to a simultaneous linear equations whose coefficient matrix is real symmetric positive definite:

$$(\mathscr{A} + \eta\mathscr{B})\,\boldsymbol{\alpha} = \eta\,\mathbf{b}\,. \tag{25}$$

For this linear equation, $\mathscr{A}$ and $\mathscr{B}$ are size $n$ matrices whose elements are $\mathscr{A}_{p,q}$ and $\mathscr{B}_{p,q}$, $p, q$=1, 2, …, $n$, respectively, and also $\boldsymbol{\alpha}$ and $\mathbf{b}$ are column vectors $\boldsymbol{\alpha} \equiv [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$ and $\mathbf{b} \equiv [b_1, b_2, \ldots, b_n]^T$, respectively. We solve this linear equation to obtain the coefficients $\alpha_k$, $k$=1, 2, …, $n$.

**Fig. 4** Procedure to calculate definite integral

```
integral (p, q, a, b) :=
 begin
 m ← p + q ;
 c_0 ← 1 ;
 for j := 1 to m do c_j ← c_{j-1} * (m−j+1)/j ;
 for j := 0 to m do w_j ← T_j(a,b) + T_{j−p}(a,b) ;
 s := 0.0 ;
 for j := 0 to m do begin
   if (j < m − j) then
     s ← s + c_j * (w_j + w_{m−j})
   else if (j == m − j) then
     s ← s + c_j * w_j
   else
    exit for
   end if
 end;
 return v ← s/2^{m+1} ;
end
```

### 3.1.2 Method-II

We assume $\boldsymbol{\alpha}$ as a vector whose 2-norm is a constant, and we first minimize the definite integral in the stopband:

$$J_{\text{stop}} = \sum_{p,q=1}^{n} \alpha_p \mathscr{A}_{p,q} \alpha_q = \boldsymbol{\alpha}^T \mathscr{A} \boldsymbol{\alpha} . \tag{26}$$

If we choose $\boldsymbol{\alpha}$ to the eigenvector of the smallest eigenvalue of the matrix $\mathscr{A}$, then $J_{\text{stop}}$ is the minimum. But if we did so, there is no more freedom left to tune the approximation in the passband. Thus, we introduce the following modification. We choose a suitable small positive number $\epsilon$. If there are $\ell$ eigenvectors whose eigenvalues are under the threshold $\epsilon$, let $S^{(\ell)}$ be the subspace which is spanned by those $\ell$ eigenvectors. Then, it holds $J_{\text{stop}} \leq \epsilon \, ||\boldsymbol{\alpha}||_2^2$ whenever $\boldsymbol{\alpha} \in S^{(\ell)}$.

The minimization condition of $J_{\text{pass}}$ under the constraint $\boldsymbol{\alpha} \in S^{(\ell)}$ reduces to a simultaneous linear equations whose coefficient matrix is of size $\ell$ and symmetric positive definite.

When we extend the subspace (by increasing $\ell$), then $J_{\text{pass}}$ decreases and the approximation in passband become better, however $J_{\text{stop}}$ increases and the approximation in stopband become worse. On the other hand, when we shrink the subspace (by decreasing $\ell$), then $J_{\text{stop}}$ decreases and the approximation in the stopband become better, however $J_{\text{pass}}$ increases and the approximation in passband goes worse.

We have to find a good choice of threshold $\epsilon$ (or $\ell$) considering the balance of both contradicting conditions of approximations in the passband and the stopband.

## 3.2 *Examples of Designed Filters*

We show in Table 1 (See Tables 2, 3, 4 and Figs. 5, 6, 7), three filters (No.1), (No.2) and (No.3) which are determined by a least-square type method (Method-II). The good thresholds in the method are determined by trials. For the filter (No.1), $\epsilon = 10^{-30}$ is used, which gives $\ell = 2$. For the filter (No.2), $\epsilon = 10^{-25}$ is used, which gives $\ell = 2$. For the filter (No.3), $\epsilon = 10^{-30}$ is used, which gives $\ell = 5$. When $n$ and $\mu$ are given, the result depends only on $\ell$ the rank of subspace. The value of threshold $\epsilon$ is used to obtain the appropriate $\ell$. If about 15-digits reduction ratio in stopbands ($g_{\text{stop}} \approx 10^{-15}$) is desired, we need 30-digits accuracy to calculate the least-square type method. Therefore, we used quadruple precision calculation only in this step to obtain coefficients $\alpha_k$, $k$=1, 2, ..., $n$ in double precision.) It seems the coefficients $\alpha_k$ themselves are numerically very sensitive even the calculation is made in quadruple precision, however it does not matter as long as the shape of the obtained transfer function is good. For the filter (No.2), the value of $\mu$ is set smaller

**Table 1** Filters used in experiments

| Filter | $n$ | $\mu$ | $g_{\text{pass}}$ | $G_{\text{stop}}$ | Coefficients | Graph |
|---|---|---|---|---|---|---|
| (No.1) | 15 | 2.0 | $2.37975\times10^{-4}$ | $1.1\times10^{-15}$ | Table 2 | Figure 5 |
| (No.2) | 15 | 1.5 | $5.46471\times10^{-5}$ | $5.8\times10^{-13}$ | Table 3 | Figure 6 |
| (No.3) | 20 | 2.0 | $1.27268\times10^{-2}$ | $2.6\times10^{-15}$ | Table 4 | Figure 7 |

**Table 2** Filter (No.1): coefficients $\alpha_k$

| $k$ | $\alpha_k$ |
|---|---|
| 1 | 3.10422 91727 23495 E−1 |
| 2 | 3.10422 91727 25609 E−1 |
| 3 | 2.85453 67519 83506 E−1 |
| 4 | 2.35515 19113 67395 E−1 |
| 5 | 1.64913 99494 59607 E−1 |
| 6 | 8.22631 58940 55446 E−2 |
| 7 | −6.57520 79352 44120 E−4 |
| 8 | −7.11802 27019 60262 E−2 |
| 9 | −1.18756 19212 14338 E−1 |
| 10 | −1.37828 28527 33139 E−1 |
| 11 | −1.29654 88587 73316 E−1 |
| 12 | −1.01680 66293 50991 E−1 |
| 13 | −6.60360 83956 00963 E−2 |
| 14 | −3.26587 11429 62141 E−2 |
| 15 | −1.19174 53737 97113 E−2 |

**Table 3** Filter (No.2): coefficients $\alpha_k$

| $k$ | $\alpha_k$ |
|---|---|
| 1 | 2.96820 21545 20158 E−1 |
| 2 | 2.96820 21559 16071 E−1 |
| 3 | 2.75088 15974 67332 E−1 |
| 4 | 2.31624 08572 14527 E−1 |
| 5 | 1.69794 56003 35121 E−1 |
| 6 | 9.63363 20742 38457 E−2 |
| 7 | 2.05451 20416 48405 E−2 |
| 8 | −4.71689 11183 01840 E−2 |
| 9 | −9.79849 96401 27541 E−2 |
| 10 | −1.24548 37945 26314 E−1 |
| 11 | −1.29956 87350 27408 E−1 |
| 12 | −1.07402 42743 15133 E−1 |
| 13 | −8.79229 80353 17280 E−2 |
| 14 | −4.04631 99059 03723 E−2 |
| 15 | −3.14108 29390 18306 E−2 |

**Table 4** Filter (No.3): coefficients $\alpha_k$

| $k$ | $\alpha_k$ |
|---|---|
| 1 | 4.83711 51618 67720 E−1 |
| 2 | 4.83711 51618 86980 E−1 |
| 3 | 3.89953 63967 72419 E−1 |
| 4 | 2.02437 88771 47818 E−1 |
| 5 | −4.47810 12123 49263 E−2 |
| 6 | −2.83593 39733 50968 E−1 |
| 7 | −4.27656 83262 24258 E−1 |
| 8 | −4.04019 57859 22469 E−1 |
| 9 | −1.91913 38100 55309 E−1 |
| 10 | 1.49822 57109 15564 E−1 |
| 11 | 4.82023 35016 46190 E−1 |
| 12 | 6.49169 20356 99877 E−1 |
| 13 | 4.90263 91392 85137 E−1 |
| 14 | 1.69552 59243 54134 E−1 |
| 15 | −5.78530 56855 20654 E−1 |
| 16 | −3.72065 97434 12249 E−1 |
| 17 | −1.44479 33647 97014 E−0 |
| 18 | 7.85556 89830 56973 E−1 |
| 19 | −1.07607 55888 09609 E−0 |
| 20 | 1.70217 32211 25582 E−0 |

than that of filter (No.1), but in exchange the value of $g_{\text{pass}}$ becomes smaller and the value of $g_{\text{stop}}$ becomes larger. For the filter (No.3), the value of $g_{\text{pass}}$ is closer to 1 than that of filter (No.1), which is attained with larger degree $n = 20$.

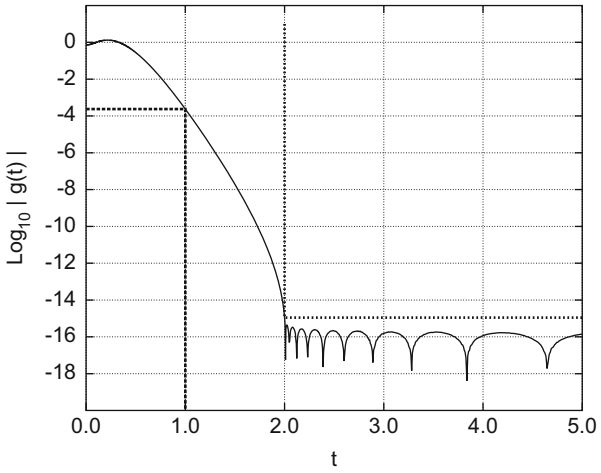


**Fig. 5** Filter (No.1): magnitude of transfer function $|g(t)|$

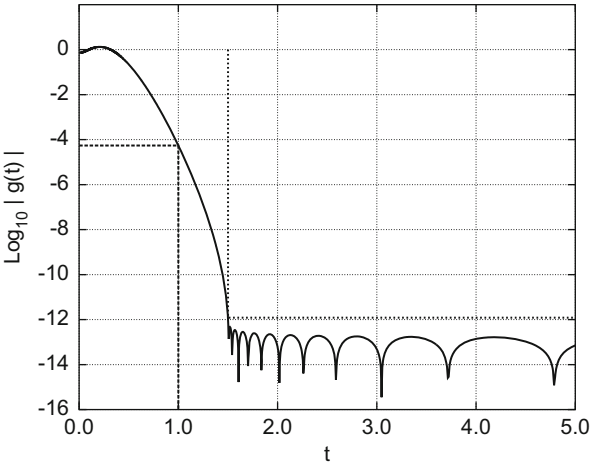


**Fig. 6** Filter (No.2): magnitude of transfer function $|g(t)|$

About the shape parameters of the transfer function $\mu$, $g_{\text{pass}}$ and $g_{\text{stop}}$:

- If $\mu$ is increased, the transition-bands become wider, then it is likely that the number of eigenvectors whose eigenvalues are in the transition-bands increases. The more eigenvectors exist whose eigenvalues are in transition-bands, then the more vectors are required to be filtered to construct an approximation of the basis of the invariant subspace.
- When the max-min ratio of the transfer rate in the passband (related to the reciprocal of $g_{\text{pass}}$) is a large number, the ununiformity of accuracies of approximated pairs tends to be low.

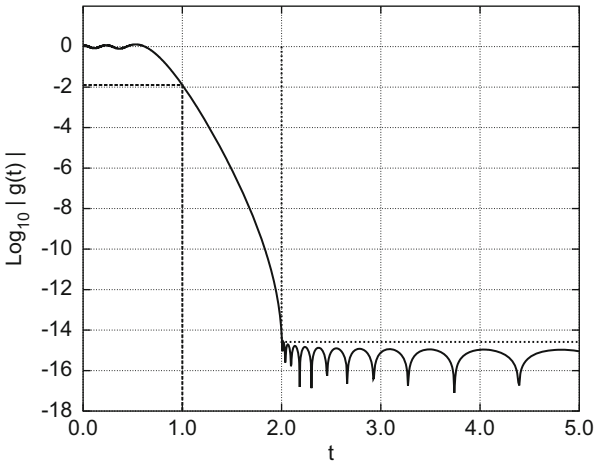


**Fig. 7** Filter (No.3): magnitude of transfer function $|g(t)|$

- When $g_{\text{stop}}$, the upper-bound of magnitude of transfer rate in stopbands, is not very small, the approximation of the invariant subspace will not be good and approximated pairs will be poor.

# 4 Experiments of Filter Diagonalization

## *4.1 Test Problem: 3D-Laplacian Discretized by FEM*

Our test problem for a real symmetric-definite generalized eigenproblem:

$$A\,\mathbf{v} = \lambda B\,\mathbf{v} \tag{27}$$

is originated from a discretization by the finite element method approximation of an eigenproblem of the Laplace operator in three dimensions:

$$(-\nabla^2)\,\Psi(x, y, z) = \lambda\,\Psi(x, y, z)\,. \tag{28}$$

The region of the partial differential operator is a cube $[0, \pi]^3$, and the boundary condition is zero-Dirichlet.

For the discretization by the finite element method approximation, each direction of the edge of the cube is equi-divided into $N_1 + 1$, $N_2 + 1$, $N_3 + 1$ sub-intervals. In each finite element, basis functions are tri-linear functions which are products of piece-wise linear function in each direction. The discretization by the finite element method gives a real symmetric-definite generalized eigenproblem of

matrices (In this case, both matrices $A$ and $B$ are positive definite, and all eigenvalues are positive real numbers).

The size of both matrices $A$ and $B$ is $N = N_1N_2N_3$. The lower bandwidth of matrices is $1 + N_1 + N_1N_2$ by a good numbering of basis functions. (Although $A$ and $B$ are quite sparse inside their bands, in our calculation they are treated as if dense inside their bands).

We solve only those eigenpairs $(\lambda, \mathbf{v})$ whose eigenvalues are in a specified interval $[a, b]$. Exact eigenvalues can be calculated by a simple formula. When the numbers of sub-intervals in directions are all different, all eigenvalues are distinct.

**Computer System Environment**

Our calculation is made on a high end class PC system. The CPU is intel Core i7-5960X (3.0 GHz, 8cores, 20 MB L3 cache). Both the turbo mode and the hyper-threading mode of the CPU are disabled from the BIOS menu. The theoretical peak performance of the CPU is 384 GFLOPS in double precision. The memory bus is quad-channel and the total main memory size is 128 GB (8 pieces of DDR4-2133 MHz (PC4-17000) 16 GB memory module). The operating system is CentOS 7 (64bit address). We used intel Fortran compiler ver.15.0.0. for Linux x86_64 with compile options: `-fast`, `-openmp`.

## *4.2 Experiment Results*

We solve an eigenproblem of large size whose discretization manner is $(N_1, N_2, N_3) = (50, 60, 70)$. In this case, the size of matrices is $N = 50 \times 60 \times 70 = 210,000$, and the lower bandwidth of matrices is $w_L = 1 + 50 + 50 \times 60 = 3051$.

We solved those pairs whose eigenvalues are in the interval $[200, 210]$ (The true count of such pairs is 91). We chose $m = 200$ for the number of vectors to be filtered. In the calculation of the action of the resolvent, the modified Cholesky factorization for the complex symmetric banded matrix is used. In experiments, three filters (No.1), (No.2) and (No.3) are tested and elapse times are measured in seconds (Table 5). For an approximated eigenpair $(\lambda, \mathbf{v})$ of the generalized eigenproblem,

**Table 5** Elapse times (in s) (matrix size $N$=210, 000)

| Kind of filter | (No.1) | (No.2) | (No.3) |
|---|---|---|---|
| Total filter diagonalization procedure | 2659.68 | 2658.44 | 3318.02 |
| − Generation of random vectors | 0.16 | 0.16 | 0.16 |
| − $B$-Orthonormalization of inputs | 90.83 | 90.82 | 90.87 |
| − Application of the filter | 2273.86 | 2272.92 | 2931.39 |
| − Construction of invariant-subspace | 213.38 | 213.11 | 214.20 |
| − Rayleigh-Ritz procedure | 81.45 | 81.42 | 81.40 |
| Calculation of norms of residuals | 220.22 | 220.45 | 219.93 |
| Memory usage (in GB)(virtual, real) | 21.5(20) | 21.5(20) | 21.5(20) |

the residual of the pair is a vector $\mathbf{r} = (A - \lambda B)\mathbf{v}$. We assume the vector $\mathbf{v}$ of every approximated pair is already normalized in $B$-norm such that $\mathbf{v}^T B\mathbf{v} = 1$. We use $B^{-1}$-norm for the norm to the residual of an approximated pair. Therefore, the norm of residual is $\Delta = \sqrt{\mathbf{r}^T B^{-1}\mathbf{r}}$, where $\mathbf{r} = (A - \lambda B)\mathbf{v}$ and $\mathbf{v}$ is $B$-normalized is assumed. The errors of eigenvalues are calculated by comparisons from exact values by using the formula for this special test problem made by the FEM discretization of Laplace operator in a cube with zero-Dirichlet boundary condition.

- Case of filter (No.1) : The graph in Fig. 8 plots the norm of the residual of each approximated pair. In the middle of the interval of eigenvalue the norm of the residual is about $10^{-10}$, and near the both ends of the interval it is about $10^{-6}$. Their ratio is about $10^4$, which corresponds to the ununiformity of transfer rate of the filter (No.1) in the passband.

  The graph in Fig. 9 plots the absolute error of eigenvalue of each approximated pair. The errors of approximated eigenvalues are less than $10^{-12}$, and approximated eigenvalues are accurate to about 14 digits or more.
- Case of filter (No.2) : The graph in Fig. 10 plots the norm of the residual of each approximated pair. In the middle of the interval of eigenvalue the norm of residual is about $10^{-10}$, and near the both ends of the interval and it is about $10^{-6}$. Their ratio is about $10^4$, which corresponds to the ununiformity of transfer rate of the filter (No.2) in the passband.

  The graph in Fig. 11 plots the absolute error of eigenvalue of each approximated pair. The absolute errors of approximated eigenvalues are less than $10^{-12}$, and approximated eigenvalues are accurate to about 14 digits or more.
- Case of filter (No.3) : The graph in Fig. 12 plots the norm of the residual of each approximated pair. In the middle of the interval of eigenvalue the norm of the residual is about $10^{-10}$, and near the both ends of the interval it is about $10^{-8}$.

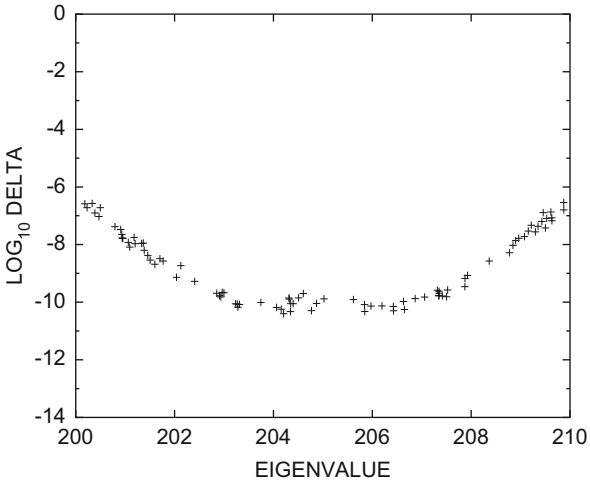


**Fig. 8** Filter (No.1): norm of residual (matrix size $N$=210, 000)

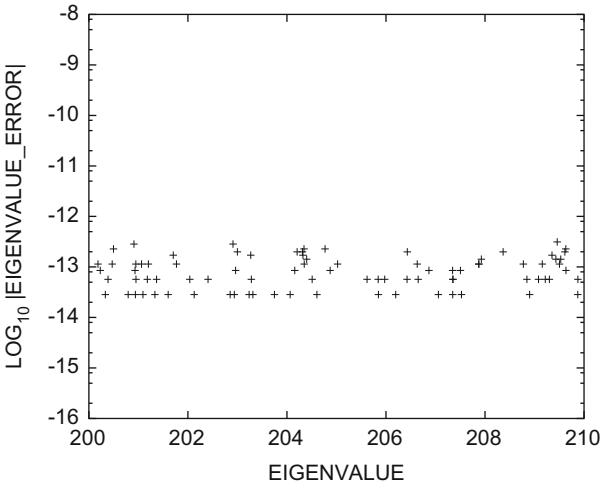


**Fig. 9** Filter (No.1): error of eigenvalue (matrix size $N$=210, 000)

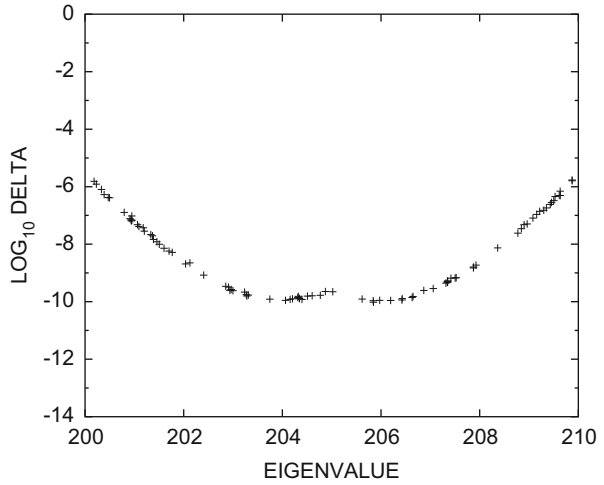


**Fig. 10** Filter (No.2): norm of residual (matrix size $N$=210, 000)

Their ratio is about $10^2$, which corresponds to the ununiformity of transfer rate of the filter (No.3) in the passband.

The graph in Fig. 13 plots the absolute error of eigenvalue of each approximated pair. The absolute errors of approximated eigenvalues are less than $10^{-12}$, and approximated eigenvalues are accurate to about 14 digits or more.

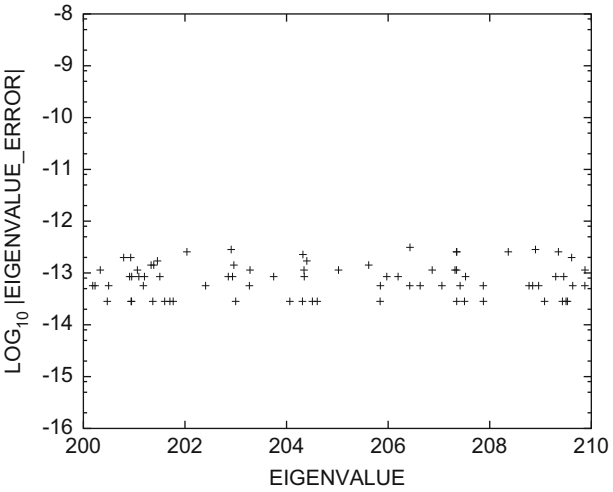


**Fig. 11** Filter (No.2): error of eigenvalue (matrix size $N=210,000$)

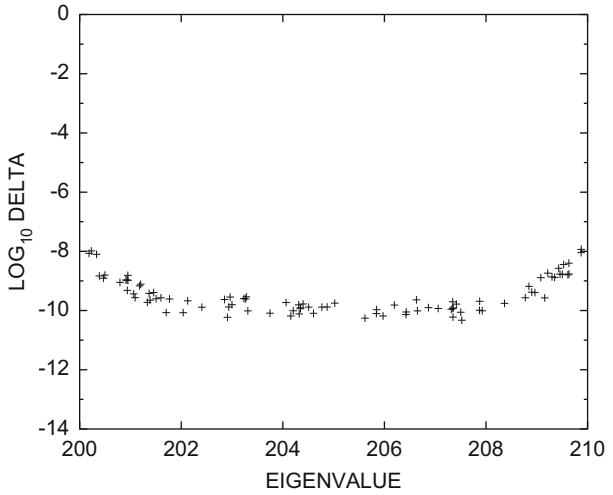


**Fig. 12** Filter (No.3): norm of residual (matrix size $N=210,000$)

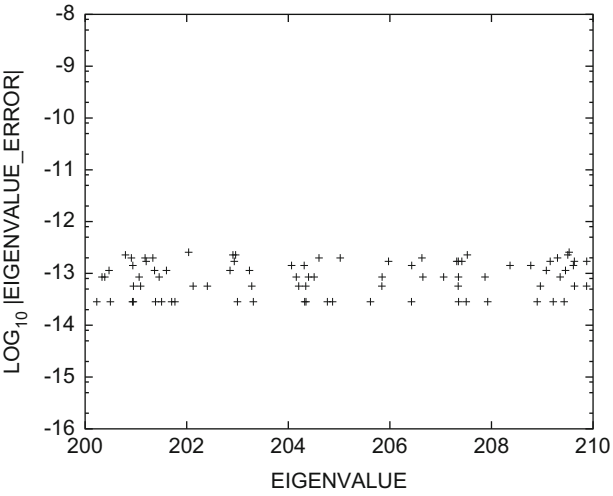


**Fig. 13** Filter (No.3): error of eigenvalue (matrix size $N$=210, 000)

## 5 Timing Comparisons with Elliptic Filters

We compared our present filter to the elliptic filter. Our present filter is a real-part of a polynomial of a resolvent. The elliptic filter is a typical filter which is a real-part of a linear combination of resolvents.

### *5.1 Filter Which is a Real-Part of a Linear Combination of Resolvents*

The filter $\mathscr{F}$ which is a real-part of a linear combination of resolvents is written as:

$$\mathscr{F} = c_\infty I + \mathrm{Re} \sum_{i=1}^{k} \gamma_i R(\rho_i) \,. \tag{29}$$

The coefficient $c_\infty$ is real, coefficients $\gamma_i$ and shifts $\rho_i$ $i$=1, 2, . . ., $k$ are complex numbers. We assume shifts are not real numbers and their imaginary parts are positive. An application of $\mathscr{F}$ to a real vector gives a real vector. For any eigenpair $(\lambda, \mathbf{v})$ of the original eigenproblem, we have $\mathscr{F}\mathbf{v} = f(\lambda)\mathbf{v}$. Here $f(\lambda)$ is the transfer function of $\mathscr{F}$, which is the following real rational function of $\lambda$ of degree $2k$:

$$f(\lambda) = c_\infty + \mathrm{Re} \sum_{i=1}^{k} \frac{\gamma_i}{\lambda - \rho_i} \,. \tag{30}$$

The normal coordinate $t$ of $\lambda$ is introduced by $\lambda = \mathscr{L}(t) \equiv (a+b)/2+(b-a)/2\cdot t$, which is a linear map between $t \in [-1, 1]$ and $\lambda \in [a, b]$. We define $g(t)$, the transfer function in the normal coordinate $t$, by the relation $f(\lambda) \equiv g(t)$. Then the form of the transfer function $g(t)$ is:

$$g(t) = c_\infty + \mathrm{Re} \sum_{i=1}^{k} \frac{c_i}{t - \tau_i}, \tag{31}$$

which is a real rational function of degree $2k$. Here, $\rho_i = \mathscr{L}(\tau_i)$, $\gamma_i = \mathscr{L}' \cdot c_i$, where $\mathscr{L}' \equiv (b - a)/2$ is a constant. In reverse, when a real rational function $g(t)$ is given which can be represented in the form of expression (31), then from the real coefficient $c_\infty$, complex coefficients $c_i$, $i$=1, 2, …, $k$ and also complex poles $\tau_i$, $i$=1, 2, …, $k$, the function $f(\lambda)$ is determined. Thus the filter $\mathscr{F}$ which is a real-part of a linear combination of resolvents is also determined.

We impose conditions for the shape of $g(t)$ on the real axis:

- $|g(t)| \le g_{\mathrm{stop}}$ when $\mu \le |t|$,
- $g_{\mathrm{pass}} \le g(t)$ when $|t| \le 1$, and $\max g(t) = 1$,
- $g_{\mathrm{stop}} < g(t) < g_{\mathrm{pass}}$ when $1 < |t| < \mu$.

We give $\mu$, $g_{\mathrm{pass}}$ and $g_{\mathrm{stop}}$ ($\mu > 1$ and $0 < g_{\mathrm{stop}} \ll g_{\mathrm{pass}} < 1$), then the (half) degree $k$ and the coefficient $c_\infty$ and coefficients and poles $c_i$, $\tau_i$, $i$=1, 2, …, $k$ are so determined to satisfy the shape conditions.

## 5.2 *Elliptic Filters for Comparisons*

For comparisons, we choose the *elliptic filter* (also called *Cauer filter* or *Zolotarev filter*) as the filter which is a real-part of a linear combination of resolvents, which comes from the theory of best approximation by rational function. The elliptic filter is so powerful that it can choose the value of $\mu$ any close to unity, $g_{\mathrm{p}}$ any close to unity, and also the value of $g_{\mathrm{s}}$ any small if (half) degree $k$ is raised. But in our experiments to make comparisons, we choose three elliptic filters (No.E1), (No.E2) and (No.E3), which have similar shape parameters ($\mu$, $g_{\mathrm{pass}}$, $g_{\mathrm{stop}}$) to our present filters (No.1), (No.2) and (No.3) respectively (Table 6). We set the same values of $\mu$ and $g_{\mathrm{pass}}$ between the present filters and the corresponding elliptic filters, but since the (half) degree $n$ of the elliptic filter must be an integer, the true values $g_{\mathrm{stop}}$ for the elliptic filters are not the same but chosen smaller (better in shape). For each elliptic filters we used, complex poles in the upper half complex plane and their complex coefficients of $g(t)$ are tabulated in Tables 7, 8 and 9, respectively. Figures 14, 15 and 16 plot graphs of transfer functions $g(t)$ of elliptic filters for only $t \ge 0$ since they are even functions.

**Table 6** Elliptic filters used in comparisons

| Filter | $k$ | $\mu$ | $g_{\text{pass}}$ | $g_{\text{stop}}$ | Coefficients | Graph |
|---|---|---|---|---|---|---|
| (No.E1) | 8 | 2.0 | $2.37975\times10^{-4}$ | $4.15\times10^{-17}$ | Table 7 | Figure 14 |
| (No.E2) | 7 | 1.5 | $5.46471\times10^{-5}$ | $7.80\times10^{-14}$ | Table 8 | Figure 15 |
| (No.E3) | 8 | 2.0 | $1.27268\times10^{-2}$ | $2.25\times10^{-15}$ | Table 9 | Figure 16 |

## *5.3 Elapse Times of Filtering*

For an elliptic filter which is a real-part of a linear combination of resolvents, the applications of resolvents to a set of vectors can be made in parallel, however when the applications are made in parallel, the larger memory space is required especially when the applications of resolvents are made by solving linear equations by direct method (matrix factorization method). Therefore, in this experiment, the applications of resolvents are made *one by one* to keep the memory requirement low.

In both present filters and elliptic filters, in an application of a resolvent $R(\rho_i) \equiv (A-\rho_i B)^{-1}B$ to a set of vectors, the linear equation with complex symmetric matrix $C = A - \rho_i B$ is solved by the complex version of modified Cholesky method $C = LDL^T$ for banded matrix $C$ and it is calculated by the same program code. For the elliptic filters, in the calculation of $R(\rho_i)X$ for the set of $m$ vectors $X$, we make a matrix multiplication $X' = BX$ once, since $X'$ is the common right-hand-sides of the set of linear equations for every $\rho_i$.

We are to solve the same eigenproblem from FEM discretization of Laplacian problem as before whose discretization manner is $(N_1, N_2, N_3) = (50, 60, 70)$. The size of matrices of the eigenproblem is $N = 50\times60\times70 = 210,000$, and the lower bandwidth of matrices is $w_L = 1 + 50 + 50\times60 = 3051$.

The only difference from the previous experiment is the kind of filters used, therefore elapse times are compared for the filtering procedure.

For present filter (No.1), (No.2) and (No.3), the count of matrix decomposition is only once, however $n$ the number of repeats of a matrix multiplication by $B$ followed by solution of a set of simultaneous linear equations by using the matrix factors is 15, 15 and 20, respectively. For elliptic filter (No.E1), (No.E2) and (No.E3), the count of matrix decompositions $k$ is 8, 7 and 8, respectively.

We measured elapse times to filter a set of $m$ vectors for the cases $m = 30$ and $m = 200$, by using present filters (No.1), (No.2) and (No.3) and elliptic filters (No.E1), (No.E2) and (No.E3), which are shown in Table 10. In the case of $m = 30$, the elapse times are about 3 times less for present filters compared with elliptic ones, therefore, when $m = 30$ the use of present filter reduces the elapse time for filtering than elliptic filter. But in the case of $m = 200$, the elapse times were not so much different between elliptic filters and present filters.

When the size of matrices $A$ and $B$ of the eigenproblem is $N$ and its bandwidth is $w$, the amount of computation to factor a symmetric banded matrix $C$ is $O(Nw^2)$, and it is $O(Nwm)$ to solve the set of simultaneous linear equations with $m$ right-

**Table 7** Elliptic filter (No.E1) for $\mu = 2.0$, $g_{\text{pass}} = 2.3797\times10^{-4}$, $g_{\text{stop}} = 1.1\times10^{-15}$ (The half degree $k = 8$ and the true $g_{\text{stop}} = 4.1457\times10^{-17}$)

| $c_\infty$=0.41457 27735 04193 0E−16 | | |
|---|---|---|
| $i$ | $\tau_i$ | $c_i$ |
| 1 | (−0.98333 59161 66002 6E+0, 0.32764 47565 36014 5E−3) | (0.32282 04928 71600 5E−5, −0.32761 86707 18423 1E−3) |
| 2 | (−0.84933 21816 52866 9E+0, 0.98905 43009 62692 8E−3) | (0.32349 46453 11294 0E−5, −0.98897 60764 00351 2E−3) |
| 3 | (−0.58304 77825 30795 7E+0, 0.16083 78800 99375 6E−2) | (0.26969 42336 27235 4E−5, −0.16082 52907 59052 8E−2) |
| 4 | (−0.20878 57279 01429 9E+0, 0.20129 18707 11943 9E−2) | (0.10982 56974 97660 9E−5, −0.20127 62426 40295 4E−2) |
| 5 | (0.20878 57279 01427 2E+0, 0.20129 18707 11944 1E−2) | (−0.10982 56974 97659 5E−5, −0.20127 62426 40295 5E−2) |
| 6 | (0.58304 77825 30793 6E+0, 0.16083 78800 99375 9E−2) | (−0.26969 42336 27234 6E−5, −0.16082 52907 59053 1E−2) |
| 7 | (0.84933 21816 52865 9E+0, 0.98905 43009 62696 0E−3) | (−0.32349 46453 11294 0E−5, −0.98897 60764 00354 3E−3) |
| 8 | (0.98333 59161 66002 2E+0, 0.32764 47565 36016 7E−3) | (−0.32282 04928 71598 6E−5, −0.32761 86707 18425 4E−3) |

**Table 8** Elliptic filter (No.E2) for $\mu = 1.5$, $g_{\text{pass}} = 5.4647\times10^{-5}$, $g_{\text{stop}} = 5.8\times10^{-13}$ (The half degree $k = 7$ and the true $g_{\text{stop}} = 7.7975\times10^{-14}$)

| $c_\infty = 0.0$ | | |
|---|---|---|
| $i$ | $\tau_i$ | $c_i$ |
| 1 | (−0.98131 05913 82855 9E+0, 0.17707 38116 11888 8E−3) | (0.85480 75525 93772 8E−6, −0.17707 04878 81648 8E−3) |
| 2 | (−0.82397 85781 93720 0E+0, 0.57606 38840 41885 7E−3) | (0.10256 96227 55537 0E−5, −0.57605 32864 32075 0E−3) |
| 3 | (−0.48589 61141 89152 9E+0, 0.10060 52142 56393 2E−2) | (0.88797 41935 99554 4E−6, −0.10060 34220 85259 1E−2) |
| 4 | (0.00000 00000 00000 0E+0, 0.12166 69145 31958 6E−2) | (0.00000 00000 00000 0E+0, −0.12166 47849 65604 5E−2) |
| 5 | (0.48589 61141 89152 2E+0, 0.10060 52142 56393 3E−2) | (−0.88797 41935 99553 8E−6, −0.10060 34220 85259 1E−2) |
| 6 | (0.82397 85781 93719 6E+0, 0.57606 38840 41886 9E−3) | (−0.10256 96227 55537 2E−5, −0.57605 32864 32076 1E−3) |
| 7 | (0.98131 05913 82855 9E+0, 0.17707 38116 11888 6E−3) | (−0.85480 75525 93770 0E−6, −0.17707 04878 81648 6E−3) |

**Table 9** Elliptic filter (No.E3) for $\mu = 2.0$, $g_{\text{pass}} = 1.2727\times10^{-2}$, $g_{\text{stop}} = 2.6\times10^{-15}$ (The half degree $k = 8$ and the true $g_{\text{stop}} = 2.2452\times10^{-15}$)

| $c_\infty$= 0.22451 63375 99703 6E−14 | | |
|---|---|---|
| $i$ | $\tau_i$ | $c_i$ |
| 1 | (−0.98342 13559 84046 3E+0, 0.24060 91027 92044 1E−2) | (0.17336 66188 99784 1E−3, −0.23958 11873 77108 1E−2) |
| 2 | (−0.84941 78015 80027 0E+0, 0.72633 17228 26009 7E−2) | (0.17373 54957 94905 3E−3, −0.72324 92487 29854 6E−2) |
| 3 | (−0.58311 91649 98342 8E+0, 0.11811 70538 57957 5E−1) | (0.14484 96328 39890 6E−3, −0.11762 09595 25342 0E−1) |
| 4 | (−0.20881 47970 76505 9E+0, 0.14782 83729 13307 9E−1) | (0.58988 50525 82562 5E−4, −0.14721 25294 03648 0E−1) |
| 5 | (0.20881 47970 76503 1E+0, 0.14782 83729 13308 0E−1) | (−0.58988 50525 82555 1E−4, −0.14721 25294 03648 1E−1) |
| 6 | (0.58311 91649 98340 7E+0, 0.11811 70538 57957 8E−1) | (−0.14484 96328 39890 2E−3, −0.11762 09595 25342 3E−1) |
| 7 | (0.84941 78015 80026 0E+0, 0.72633 17228 26012 2E−2) | (−0.17373 54957 94905 3E−3, −0.72324 92487 29857 0E−2) |
| 8 | (0.98342 13559 84046 0E+0, 0.24060 91027 92045 8E−2) | (−0.17336 66188 99783 1E−3, −0.23958 11873 77109 8E−2) |

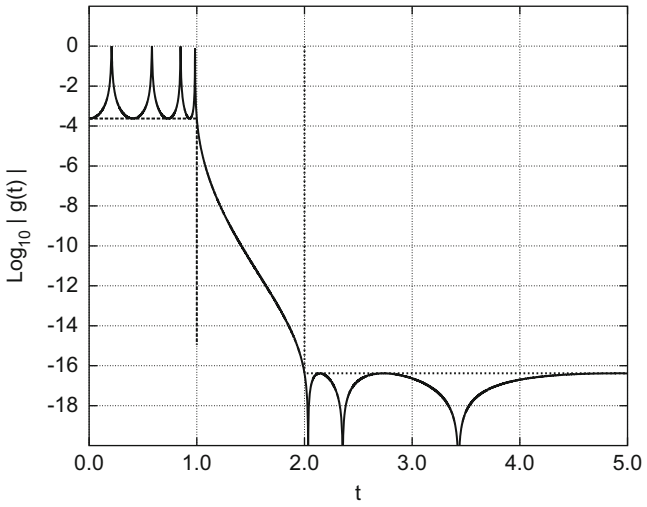


**Fig. 14** Filter (No.E1): magnitude of transfer function $|g(t)|$

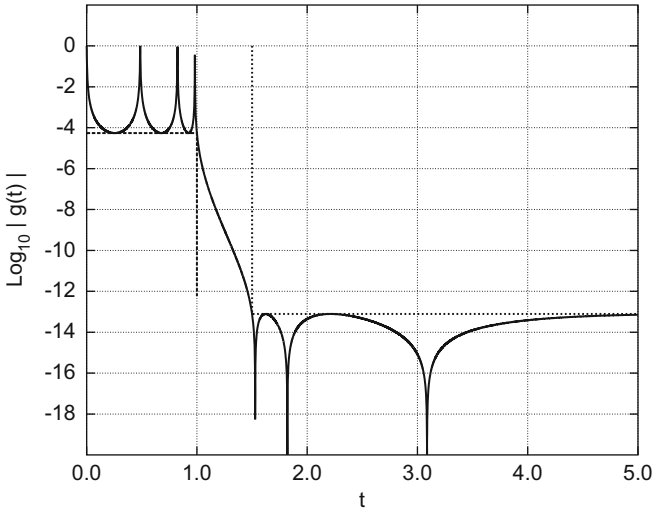


**Fig. 15** Filter (No.E2): magnitude of transfer function $|g(t)|$

hand-sides after the matrix is factored. The amount of computation to multiply a symmetric banded matrix $B$ to a set of $m$ vectors is also $O(Nwm)$.

Thus, the elapse time to factor a symmetric banded matrix $C$ of size $N$ with bandwidth $w$ is $T_{\text{decompose}} \approx t_1 N w^2$, the elapse time to solve a set of $m$ of simultaneous linear equations using the matrix factor is $T_{\text{solve}} \approx t_2 N w m$ and the elapse time to multiply a set of $m$ vectors to a symmetric banded matrix $B$ of size $N$ with bandwidth $w$ is $T_{\text{mulB}} \approx t_3 N w m$. Here, $t_1$, $t_2$ and $t_3$ are some coefficients. Then,

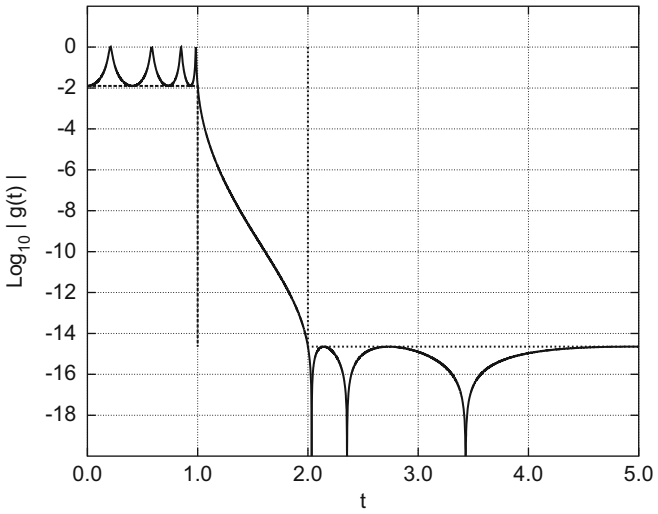


**Fig. 16** Filter (No.E3): magnitude of transfer function $|g(t)|$

**Table 10** Elapse times (in s) for filtering $m$ vectors

| $m$ | (No.1) | (No.2) | (No.3) | (No.E1) | (No.E2) | (No.E3) |
|---|---|---|---|---|---|---|
| 30 | 621.8 | 622.3 | 728.3 | 2516.7 | 2210.2 | 2517.9 |
| 200 | 2273.9 | 2272.9 | 2931.4 | 3352.1 | 3023.7 | 3352.0 |

the elapse time of present filter can be written as:

$$T_{\text{present}} = T_{\text{decompose}} + n \times (T_{\text{mulB}} + T_{\text{solve}} + O(Nm)) \,, \tag{32}$$

and the elapse time of elliptic filter by using $k$ resolvents is written as:

$$T_{\text{elliptic}} = T_{\text{mulB}} + k \times (T_{\text{decompose}} + T_{\text{solve}} + O(Nm)) \,. \tag{33}$$

We have

$$\begin{aligned} T_{\text{present}}/N &\approx t_1 w^2 + (t_2 + t_3) nwm, \\ T_{\text{elliptic}}/N &\approx t_1 k w^2 + (t_2 k + t_3) wm \,. \end{aligned} \tag{34}$$

The coefficients $t_1$, $t_2$ and $t_3$ depends on the system and the manner of calculation. Since the calculation of matrix decomposition has much higher locality of data references than the matrix-vector multiplication or the solution of linear equation using the matrix factors, the coefficient $t_1$ must be smaller than $t_2$ or $t_3$. From the above expression (34), when $m$ is small and ignorable, the elapse time of present filter could be nearly $k$ times faster than the elliptic filter (with no parallel resolvent calculation) since it makes just one matrix decomposition. However, as $m$ increases, the advantage of the present approach seems reduced.

## 6 Conclusion

For a real symmetric-definite generalized eigenproblem, the filter diagonalization method solves those eigenpairs whose eigenvalues are in the specified interval. In this present study, the filter we used is a real-part of a polynomial of a single resolvent rather than a real-part of a linear combination of many resolvents to take advantages of reductions of the amount of memory requirement and also computation. In numerical experiments, we obtained good results.

When the filter is a real-part of a linear combination of many (8 to 16) resolvents, for each resolvents the applications to a set of vectors can be made in parallel. For our present method, when the filter is a real-part of a polynomial of a resolvent, applications of the resolvent as many times as the degree of the polynomial can be made only in sequential. However, even the potential parallelism is reduced, the present method has an advantage that it requires only single resolvent, therefore the amount of storage requirement is low and also the total amount of computation can be reduced. Once a single matrix which corresponds to the resolvent is decomposed and factors are stored, each application of the resolvent can be calculated easily and fast. Another difficulty of the present type of filter is that the shape of the transfer function is not so good as the shape of the filter which is a real-part of a linear combination of many resolvents such as elliptic filter.

In this paper, three filters are constructed and they are shown with their polynomial coefficients and shape parameters. By using these three filters, we made some experiments of the filter diagonalization. For a generalized eigenproblem which is derived from FEM discretization of the Laplace operator over a cubic region with zero Dirichlet boundary condition, we solved some internal eigenpairs whose eigenvalues are in the specified interval. We compared eigenvalues of approximated pairs with exact ones, and found their agreements were good, which showed our approach worked as expected.

## References

1. Austin, A.P., Trefethen, L.N.: Computing eigenvalues of real symmetric matrices with rational filters in real arithmetic. SIAM J. Sci. Comput. **37**(3), A1365–A1387 (2015)
2. Galgon, M., Krämer, L., Lang, B.: The FEAST algorithm for large eigenvalue problems. Proc. Appl. Math. Mech. **11**, 747–748 (2011)
3. Güttel, S., Polizzi, E., Tang, P.T.P., Viaud, G.: Zolotarev quadrature rules and load balancing for the FEAST eigensolver. SIAM J. Sci. Comput. **37**(4), A2100–A2122 (2015)
4. Ikegami, T., Sakurai, T., Nagashima, U.: A filter diagonalization for generalized eigenvalue problems based on the Sakurai–Sugiura projection method. J. Comput. Appl. Math. **233**(8), 1927–1936 (2010)
5. Murakami, H.: A filter diagonalization method by the linear combination of resolvents. IPSJ Trans. ACS-21 **49**(SIG2), 66–87 (2008, written in Japanese)
6. Murakami, H.: Filter designs for the symmetric eigenproblems to solve eigenpairs whose eigenvalues are in the specified interval. IPSJ Trans. ACS-31 **3**(3), 1–21 (2010, written in Japanese)

7. Murakami, H.: Construction of the approximate invariant subspace of a symmetric generalized eigenproblem by the filter operator. IPSJ Proc. SACSIS 2011, 332–339 (2011, written in Japanese).
8. Murakami, H.: Construction of the approximate invariant subspace of a symmetric generalized eigenproblem by the filter operator. IPSJ Trans. ACS-35 **4**(4), 51–64 (2011, written in Japanese).
9. Murakami, H.: Filter diagonalization method for a Hermitian definite generalized eigenproblem by using a linear combination of resolvents as the filter. IPSJ Tans. ACS-45 **7**(1), 57–72 (2014, written in Japanese)
10. Murakami, H.: On filter diagonalization method. In: Abstracts of JSIAM 2014 annual meeting, pp. 329–330 (August 2014, written in Japanese)
11. Murakami, H.: An experiment of filter diagonalization method for symmetric definite generalized eigenproblem which uses a filter constructed from a resolvent. IPSJ SIG Technical Reports, vol. 2015-HPC-149(7), pp. 1–16 (June, 2015, written in Japanese)
12. Murakami, H.: Filter diagonalization method for real symmetric definite generalized eigenproblem whose filter is a polynomial of a resolvent. In: Abstracts of EPASA 2015 at Tsukuba, p. 28 (single page poster abstract) (September 2015)
13. Polizzi, E.: A density matrix-based algorithm for solving eigenvalue problems. Phys. Rev. B **79**(1), 115112(6pages) (2009)
14. Sakurai, T., Sugiura, H.: A projection method for generalized eigenvalue problems using numerical integration. J. Comput. Appl. Math. **159**, 119–128 (2003)
15. Sakurai, T., Tadano, H.: CIRR: a Rayleigh-Ritz type method with contour integral for generalized eigenvalue problems. Hokkaido Math. J. **36**(4), 745–757 (2007)
16. Toledo, S., Rabani, E.: Very large electronic structure calculations using an out-of-core filter-diagonalization method. J. Comput. Phys. **180**(1), 256–269 (2002)