Beyond AMLS: domain decomposition with rational filtering

Vassilis Kalantzis

Computer Science and Engineering Department
University of Minnesota - Twin Cities, USA

Argonne National Laboratory, Lemont, IL
11-07-2017
Acknowledgments

- Joint work with Y. Xi (UMN), and Y. Saad (UMN).
- Special thanks to the University of Minnesota Supercomputing Institute for providing us with computational resources to perform our experiments.
- Work supported by NSF and DOE (DE-SC0008877).
Contents

1 Introduction

2 The domain decomposition (DD) viewpoint and the AMLS approach

3 The Rational Filtering DD Eigenvalue Solver (RF-DDES)

4 Experiments
   • Comparisons against rational filtering Krylov
Our focus

- We consider the symmetric eigenvalue problem $Ax = \lambda Mx$, where $A$ and $M$ are sparse, and $M$ is SPD.
- We are interested in computing all $nev$ eigenvalues-eigenvectors located inside the real interval $[\alpha, \beta]$.
- In this talk: we combine domain decomposition with rational filtering

Contribution of this talk

We formulate an algorithm, abbreviated as RF-DDES, that features:

- Reduced orthogonalization costs compared to Krylov projection methods
- Enhanced accuracy compared to existing domain decomposition approaches
- Reduced complex arithmetic

Also: we discuss a parallel (PETSc) implementation of the proposed algorithm
Contents

1 Introduction

2 The domain decomposition (DD) viewpoint and the AMLS approach

3 The Rational Filtering DD Eigenvalue Solver (RF-DDES)

4 Experiments
   • Comparisons against rational filtering Krylov
The domain decomposition (DD) viewpoint and the AMLS approach

The main idea behind DD eigenvalue solvers (example for two subdomains)

DD decouples the original eigenvalue problem into two parts:
- The first part considers only interface (red) variables
- The second part considers only interior (green) variables
Reordering equations/unknowns ($p \geq 2$ subdomains)

$$A = \begin{pmatrix}
B_1 & & & E_1 \\
& B_2 & & E_2 \\
& & \ddots & \ddots \\
E_1^T & E_2^T & \cdots & E_p^T \\
& & & C
\end{pmatrix},$$

$$M = \begin{pmatrix}
M_B^{(1)} & M_B^{(2)} & & & M_E^{(1)} \\
& M_B & & & M_E^{(2)} \\
& & \ddots & \ddots \\
(M_E^{(1)})^T & (M_E^{(2)})^T & \cdots & (M_E^{(p)})^T & M_C
\end{pmatrix}.$$
Reordering equations/unknowns ($p \geq 2$ subdomains)

\[ A = \begin{pmatrix} B_1 & B_2 & \cdots & B_p \\ E_1 & E_2 & \cdots & E_p \\ E_1^T & E_2^T & \cdots & E_p^T \end{pmatrix}, \quad M = \begin{pmatrix} M_B^{(1)} & M_B^{(2)} & \cdots & M_B^{(p)} \\ M_E^{(1)} & M_E^{(2)} & \cdots & M_E^{(p)} \\ (M_E^{(1)})^T & (M_E^{(2)})^T & \cdots & (M_E^{(p)})^T \end{pmatrix}, \]

Notation: write as

\[ A = \begin{pmatrix} B & E \\ E^T & C \end{pmatrix}, \quad M = \begin{pmatrix} M_B & M_E \\ M_E^T & M_C \end{pmatrix}, \]

\[ x^{(i)} = \begin{pmatrix} u^{(i)} \\ y^{(i)} \end{pmatrix} = \begin{pmatrix} u_1^{(i)} \\ \vdots \\ u_p^{(i)} \\ y_1^{(i)} \\ \vdots \\ y_p^{(i)} \end{pmatrix}. \]
An example of the sparsity pattern of $A$ and $M$ for $p = 4$
Invariant subspaces from a Schur complement viewpoint

\[(A - \lambda_i M)x^{(i)} = \begin{pmatrix} B - \lambda_i M_B & E - \lambda_i M_E \\ E^T - \lambda_i M_E^T & C - \lambda_i M_C \end{pmatrix} \begin{pmatrix} u^{(i)} \\ y^{(i)} \end{pmatrix} = 0.\]
Invariant subspaces from a Schur complement viewpoint

\[(A - \lambda_i M)x^{(i)} = \begin{pmatrix} B - \lambda_i M_B & E - \lambda_i M_E \\ E^T - \lambda_i M_E^T & C - \lambda_i M_C \end{pmatrix} \begin{pmatrix} u^{(i)} \\ y^{(i)} \end{pmatrix} = 0.\]

A direct computation leads to:

\[S(\lambda_i)y^{(i)} = 0, \quad u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)},\]
Invariant subspaces from a Schur complement viewpoint

\[(A - \lambda_i M)x^{(i)} = \begin{pmatrix} B - \lambda_i M_B & E - \lambda_i M_E \\ E^T - \lambda_i M_E^T & C - \lambda_i M_C \end{pmatrix} \begin{pmatrix} u^{(i)} \\ y^{(i)} \end{pmatrix} = 0.\]

A direct computation leads to:

\[S(\lambda_i)y^{(i)} = 0, \quad u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)}, \]

\[S(\lambda_i) = C - \lambda_i M_C - (E - \lambda_i M_E)^T(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E).\]
Invariant subspaces from a Schur complement viewpoint

\[(A - \lambda_i M)x^{(i)} = \begin{pmatrix} B - \lambda_i M_B & E - \lambda_i M_E \\ E^T - \lambda_i M_E^T & C - \lambda_i M_C \end{pmatrix} \begin{pmatrix} u^{(i)} \\ y^{(i)} \end{pmatrix} = 0.\]

A direct computation leads to:

\[S(\lambda_i)y^{(i)} = 0, \quad u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)},\]

\[S(\lambda_i) = C - \lambda_i M_C - (E - \lambda_i M_E)^T(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E).\]

To recover the exact eigenpairs \((\lambda_i, x^{(i)})_{i=1,...,nev}\)

Perform a Rayleigh-Ritz projection on \(Z = U \oplus Y:\)

\[Y = \text{span} \left\{ y^{(i)} \right\}_{i=1,...,nev},\]

\[U = \text{span} \left\{ -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} \right\}_{i=1,...,nev}.\]
The Automated Multi-Level Substructuring (AMLS) approach

Truncation of the interface eigenvalue problem

- AMLS considers a first-order approximation of $S(\lambda_i)$, $i = 1, \ldots, \text{nev}$ around a fixed $\sigma \in \mathbb{R}$
- $\mathcal{V}$ is approximated by $\text{span}\left\{\hat{y}^{(1)}, \ldots, \hat{y}^{(k)}\right\}$, where $\hat{y}^{(1)}, \ldots, \hat{y}^{(k)}$ denote the eigenvectors associated with the $k$ smallest (in modulus) eigenvalues of $(S(\sigma), -S'(\sigma))$.
- **Pros**: reduced orthogonalization costs
- **Cons**: only moderate accuracy

Approximation of the solution associated with the interior variables

- Similarly, $\mathcal{U}$ is approximated by $\text{span}\left\{(B - \sigma M_B)^{-1}(E - \sigma M_E)\left[\hat{y}^{(1)}, \ldots, \hat{y}^{(k)}\right]\right\}$
- This step is trivially parallel among the subdomains
The Rational Filtering DD Eigenvalue Solver (RF-DDES)

Contents

1 Introduction

2 The domain decomposition (DD) viewpoint and the AMLS approach

3 The Rational Filtering DD Eigenvalue Solver (RF-DDES)

4 Experiments
   • Comparisons against rational filtering Krylov
Let

\[ I_{[\alpha,\beta]}(\zeta) = \frac{1}{2\pi i} \int_{\Gamma_{[\alpha,\beta]}} \frac{1}{\nu - \zeta} d\nu. \]
Let

$$I_{[\alpha, \beta]}(\zeta) = \frac{1}{2\pi i} \int_{\Gamma_{[\alpha, \beta]}} \frac{1}{\nu - \zeta} d\nu.$$ 

We approximate $-I_{[\alpha, \beta]}(\zeta)$ by

$$\rho(\zeta) = \sum_{\ell=1}^{2N_c} \frac{\omega_{\ell}}{\zeta - \zeta_{\ell}}.$$
Let
\[ I_{[\alpha, \beta]}(\zeta) = \frac{1}{2\pi i} \int_{\Gamma_{[\alpha, \beta]}} \frac{1}{\nu - \zeta} d\nu. \]

We approximate \(-I_{[\alpha, \beta]}(\zeta)\) by
\[ \rho(\zeta) = \sum_{\ell=1}^{2N_c} \frac{\omega_\ell}{\zeta - \zeta_\ell}. \]

Applying the filter to the matrix pencil \((A, M)\) gives:
\[ \rho(M^{-1}A) = \sum_{\ell=1}^{2N_c} \omega_\ell (A - \zeta_\ell M)^{-1} M. \]
Let
\[ I_{[\alpha, \beta]}(\zeta) = \frac{1}{2\pi i} \int_{\Gamma_{[\alpha, \beta]}} \frac{1}{\nu - \zeta} d\nu. \]

We approximate \(-I_{[\alpha, \beta]}(\zeta)\) by
\[ \rho(\zeta) = \sum_{\ell=1}^{2N_c} \frac{\omega_\ell}{\zeta - \zeta_\ell}. \]

Applying the filter to the matrix pencil \((A, M)\) gives:
\[ \rho(M^{-1}A) = \sum_{\ell=1}^{2N_c} \omega_\ell (A - \zeta_\ell M)^{-1} M. \]

Note that if \((\omega_\ell, \zeta_\ell) = (\omega_\ell + N_c, \zeta_\ell + N_c)\):
\[ \rho(M^{-1}A) = 2\Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell (A - \zeta_\ell M)^{-1} M \right\}. \]
How to approximate $\text{span}\{y^{(1)}, \ldots, y^{(nev)}\}$

Let $\zeta \in \mathbb{C}$ and define

$$B_\zeta = B - \zeta M_B, \quad E_\zeta = E - \zeta M_E,$$
$$C_\zeta = C - \zeta M_C, \quad S_\zeta = C_\zeta - E_\zeta^T B_\zeta^{-1} E_\zeta.$$
Let $\zeta \in \mathbb{C}$ and define

\begin{align*}
B_\zeta &= B - \zeta M_B, & E_\zeta &= E - \zeta M_E, \\
C_\zeta &= C - \zeta M_C, & S_\zeta &= C_\zeta - E_\zeta^T B_\zeta^{-1} E_\zeta.
\end{align*}

Then,

$$(A - \zeta M)^{-1} = \begin{pmatrix}
B_\zeta^{-1} + B_\zeta^{-1} E_\zeta S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & -B_\zeta^{-1} E_\zeta S_\zeta^{-1} \\
-S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & S_\zeta^{-1}
\end{pmatrix}.$$
How to approximate \( \text{span}\{y^{(1)}, \ldots, y^{(nev)}\} \) \( (I) \)

Let \( \zeta \in \mathbb{C} \) and define

\[
B_\zeta = B - \zeta M_B, \quad E_\zeta = E - \zeta M_E,
\]

\[
C_\zeta = C - \zeta M_C, \quad S_\zeta = C_\zeta - E_\zeta^T B_\zeta^{-1} E_\zeta.
\]

Then,

\[
(A - \zeta M)^{-1} = \begin{pmatrix}
B_\zeta^{-1} + B_\zeta^{-1} E_\zeta S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & -B_\zeta^{-1} E_\zeta S_\zeta^{-1} \\
-S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & S_\zeta^{-1}
\end{pmatrix}.
\]

Recall the partitioning \( x^{(i)} = [(u^{(i)})^T, (y^{(i)})^T]^T \):
How to approximate $\text{span}\{y^{(1)}, \ldots, y^{(nev)}\}$ (I)

Let $\zeta \in \mathbb{C}$ and define

$$
B_\zeta = B - \zeta M_B, \quad E_\zeta = E - \zeta M_E, \\
C_\zeta = C - \zeta M_C, \quad S_\zeta = C_\zeta - E_\zeta^T B_\zeta^{-1} E_\zeta.
$$

Then,

$$(A - \zeta M)^{-1} = \begin{pmatrix}
B_\zeta^{-1} + B_\zeta^{-1} E_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & -B_\zeta^{-1} E_\zeta^{-1} S_\zeta^{-1} \\
-S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & S_\zeta^{-1}
\end{pmatrix}.
$$

Recall the partitioning $x^{(i)} = [(u^{(i)})^T, (y^{(i)})^T]^T$:

$$
\rho(M^{-1}A) = 2\Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell \begin{pmatrix}
B_{\zeta_\ell}^{-1} + B_{\zeta_\ell}^{-1} E_{\zeta_\ell}^{-1} E_{\zeta_\ell}^T B_{\zeta_\ell}^{-1} & -B_{\zeta_\ell}^{-1} E_{\zeta_\ell}^{-1} S_{\zeta_\ell}^{-1} \\
-S_{\zeta_\ell}^{-1} E_{\zeta_\ell}^T B_{\zeta_\ell}^{-1} & S_{\zeta_\ell}^{-1}
\end{pmatrix} \right\} M
$$
How to approximate \( \text{span}\{y^{(1)}, \ldots, y^{(nev)}\} \) (I)

Let \( \zeta \in \mathbb{C} \) and define

\[
B_\zeta = B - \zeta M_B, \quad E_\zeta = E - \zeta M_E, \\
C_\zeta = C - \zeta M_C, \quad S_\zeta = C_\zeta - E_\zeta^T B_\zeta^{-1} E_\zeta.
\]

Then,

\[
(A - \zeta M)^{-1} = \begin{pmatrix}
B_\zeta^{-1} + B_\zeta^{-1} E_\zeta S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & -B_\zeta^{-1} E_\zeta S_\zeta^{-1} \\
-S_\zeta^{-1} E_\zeta^T B_\zeta^{-1} & S_\zeta^{-1}
\end{pmatrix}.
\]

Recall the partitioning \( x^{(i)} = [(u^{(i)})^T, (y^{(i)})^T]^T \):

\[
\rho(M^{-1}A) = 2\Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell \begin{bmatrix}
B_{\zeta_\ell}^{-1} + B_{\zeta_\ell}^{-1} E_{\zeta_\ell} S_{\zeta_\ell}^{-1} E_{\zeta_\ell}^T B_{\zeta_\ell}^{-1} & -B_{\zeta_\ell}^{-1} E_{\zeta_\ell} S_{\zeta_\ell}^{-1} \\
-S_{\zeta_\ell}^{-1} E_{\zeta_\ell}^T B_{\zeta_\ell}^{-1} & S_{\zeta_\ell}^{-1}
\end{bmatrix} \right\} M
\]

\[
= \sum_{i=1}^{n} \rho(\lambda_i) \begin{bmatrix}
(u^{(i)})^T & (u^{(i)})^T \\
(y^{(i)})^T & (y^{(i)})^T
\end{bmatrix} M.
\]
How to approximate \( \text{span}\{y^{(1)}, \ldots, y^{(\text{nev})}\} \) (II)

Equating blocks leads to:

\[
2\Re\left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S_{\xi_{\ell}}^{-1} \right\} = \sum_{i=1}^{n} \rho(\lambda_i) y^{(i)}(y^{(i)})^T.
\]

Since \( \rho(\lambda_1), \ldots, \rho(\lambda_{\text{nev}}) \neq 0\):

\[
\text{span}\{y^{(1)}, \ldots, y^{(\text{nev})}\} \subseteq \text{range} \left( 2\Re\left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S_{\xi_{\ell}}^{-1} \right\} \right).
\]
How to approximate \( \text{span}\{y^{(1)}, \ldots, y^{(nev)}\} \) (II)

Equating blocks leads to:

\[
2\Re\left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S_{\zeta_{\ell}}^{-1} \right\} = \sum_{i=1}^{n} \rho(\lambda_i)y^{(i)}(y^{(i)})^T.
\]

Since \( \rho(\lambda_1), \ldots, \rho(\lambda_{nev}) \neq 0 \):

\[
\text{span}\{y^{(1)}, \ldots, y^{(nev)}\} \subseteq \text{range}\left( 2\Re\left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S_{\zeta_{\ell}}^{-1} \right\} \right).
\]

Capture \( \text{range}\left( \Re\left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S_{\zeta_{\ell}}^{-1} \right\} \right) \) by a Krylov projection scheme!
Algorithm 3.1: Krylov restricted to the interface variables

**Algorithm**

0. **Start with** $q^{(1)} \in \mathbb{R}^s$, *s.t.* $\|q^{(1)}\|_2 = 1$, $q_0 := 0$, $b_1 = 0$, $\text{tol} \in \mathbb{R}$

1. **For** $\mu = 1, 2, \ldots$

2. **Compute** $w = \Re \left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S_{\ell}^{-1} q^{(\mu)} \right\} - b_{\mu} q^{(\mu-1)}$

3. $a_{\mu} = w^T q^{(\mu)}$

4. **For** $\kappa = 1, \ldots, \mu$

5. $w = w - q^{(\kappa)} (w^T q^{(\kappa)})$

6. **End**

7. $b_{\mu+1} := \|w\|_2$

8. **If** $b_{\mu+1} = 0$

9. **generate a unit-norm** $q^{(\mu+1)}$ orthogonal to $q^{(1)}, \ldots, q^{(\mu)}$

10. **Else**

11. $q^{(\mu+1)} = w / b_{\mu+1}$

12. **EndIf**

13. **If** the sum of eigenvalue of $T_\mu$ remains unchanged (up to $\text{tol}$) during the last few iterations; **BREAK**; **EndIf**

14. **End**

15. **Return** $Q_\mu = [q^{(1)}, \ldots, q^{(\mu)}]$
How to approximate $\text{span}\{y^{(1)}, \ldots, y^{(nev)}\}$ (III)

Figure: Leading singular values of $\Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell S(\zeta_\ell)^{-1} \right\}$ ($[\alpha, \beta] = [\lambda_1, \lambda_{100}]$).
How to approximate $\text{span}\{y^{(1)}, \ldots, y^{(nev)}\}$ (III)

**Figure**: Leading singular values of $\Re \left\{ \sum_{\ell=1}^{N_c} \omega_{\ell} S(\zeta_{\ell})^{-1} \right\}$ $([\alpha, \beta] = [\lambda_1, \lambda_{100}])$.

- Only vectors of length $s$ (\# of interface variables) need be orthonormalized.
- Moreover, $\text{solve}(A, M, \zeta_{\ell}) \approx \text{solve}(S(\zeta_{\ell})) + 2 \times \text{solve}(B, M_B, \zeta_{\ell})$
How to approximate \( \text{span}\{y^{(1)}, \ldots, y^{(nev)}\} \) (III)

**Figure:** Leading singular values of \( \Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell S(\zeta_\ell)^{-1} \right\} \) \( ([\alpha, \beta] = [\lambda_1, \lambda_{100}]\)).

- Only vectors of length \( s \) (\# of interface variables) need be orthonormalized
- Moreover, \( \text{solve}(A, M, \zeta_\ell) \approx \text{solve}(S(\zeta_\ell)) + 2 \times \text{solve}(B, M_B, \zeta_\ell) \)
- What if \( nev > s \), or rank\( [y^{(1)}, \ldots, y^{(nev)}] < nev \)?
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(\text{nev})}\}$ (I)

Standard approach

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} = -B\lambda_i^{-1}E\lambda_i y^{(i)}$, $i = 1, \ldots, \text{nev}$. 

Issue #1: Needs access to both $\lambda_i y^{(i)}$

Issue #2: Impractical for large values of $\text{nev}$

The alternative: approximate the action of $B^{-1}\lambda_i E\lambda_i$

Assume that $y^{(i)}$ is known:

Let $\sigma \in \mathbb{R}$ and start with a "basic" approximation:

$\hat{u}^{(i)} = -B^{-1}\sigma E\sigma y^{(i)}$

The error is of the form:

$u^{(i)} - \hat{u}^{(i)} = -\left[ B^{-1} \lambda_i - B^{-1} \sigma \right] E\sigma y^{(i)} + (\lambda_i - \sigma) B^{-1} \lambda_i M_E y^{(i)}$.
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(nev)}\}$ (I)

**Standard approach**

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} = -B_{\lambda_i}^{-1}E_{\lambda_i}y^{(i)}$, $i = 1, \ldots, nev$.

- **Issue #1**: Needs access to both $\lambda_i$, $y^{(i)}$
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(nev)}\}$ (I)

Standard approach

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} = -B_{\lambda_i}^{-1}E_{\lambda_i}y^{(i)}$, $i = 1, \ldots, nev$.

- Issue #1: Needs access to both $\lambda_i$, $y^{(i)}$
- Issue #2: Impractical for large values of $nev$

The alternative: approximate the action of $B_{\lambda_i}^{-1}$, $E_{\lambda_i}$
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(nev)}\}$ (I)

**Standard approach**

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} = -B^{-1}_{\lambda_i}E_{\lambda_i}y^{(i)}$, $i = 1, \ldots, \text{nev}$.

- Issue #1: Needs access to both $\lambda_i$, $y^{(i)}$
- Issue #2: Impractical for large values of $\text{nev}$

**The alternative:** approximate the action of $B_{\lambda_i}^{-1}$, $E_{\lambda_i}$

Assume that $y^{(i)}$ is known:
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(nev)}\}$ \(I\)

**Standard approach**

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1} (E - \lambda_i M_E) y^{(i)} = -B_{\lambda_i}^{-1} E_{\lambda_i} y^{(i)}$, $i = 1, \ldots, nev$.

- Issue #1: Needs access to both $\lambda_i$, $y^{(i)}$
- Issue #2: Impractical for large values of $nev$

**The alternative: approximate the action of $B_{\lambda_i}^{-1}$, $E_{\lambda_i}$**

Assume that $y^{(i)}$ is known:

- Let $\sigma \in \mathbb{R}$ and start with a “basic” approximation:
  $$\hat{u}^{(i)} = -B_{\sigma}^{-1} E_{\sigma} y^{(i)}$$
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(nev)}\}$ (I)

**Standard approach**

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} = -B_{\lambda_i}^{-1}E\lambda_i y^{(i)}$, $i = 1, \ldots, nev$.

- Issue #1: Needs access to both $\lambda_i$, $y^{(i)}$
- Issue #2: Impractical for large values of $nev$

The alternative: approximate the action of $B_{\lambda_i}^{-1}$, $E\lambda_i$

Assume that $y^{(i)}$ is known:

- Let $\sigma \in \mathbb{R}$ and start with a “basic” approximation:
  \[
  \hat{u}^{(i)} = -B_{\sigma}^{-1}E\sigma y^{(i)}
  \]
  
- The error is of the form:
  \[
  u^{(i)} - \hat{u}^{(i)} = -[B_{\lambda_i}^{-1} - B_{\sigma}^{-1}]E\sigma y^{(i)} + (\lambda_i - \sigma)B_{\lambda_i}^{-1}M_E y^{(i)}.
  \]
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(\text{nev})}\}$ (I)

Standard approach

Compute $u^{(i)} = -(B - \lambda_i M_B)^{-1}(E - \lambda_i M_E)y^{(i)} = -B_{\lambda_i}^{-1}E\lambda_i y^{(i)}$, $i = 1, \ldots, \text{nev}$.

- Issue #1: Needs access to both $\lambda_i$, $y^{(i)}$
- Issue #2: Impractical for large values of $\text{nev}$

The alternative: approximate the action of $B_{\lambda_i}^{-1}$, $E\lambda_i$

Assume that $y^{(i)}$ is known:

- Let $\sigma \in \mathbb{R}$ and start with a “basic” approximation:
  $$\hat{u}^{(i)} = -B_{\sigma}^{-1}E\sigma y^{(i)}$$

  The error is of the form:
  $$u^{(i)} - \hat{u}^{(i)} = -[B_{\lambda_i}^{-1} - B_{\sigma}^{-1}]E\sigma y^{(i)} + (\lambda_i - \sigma)B_{\lambda_i}^{-1}M_E y^{(i)}.$$

- To improve accuracy: extract $\hat{u}^{(i)}$ from a subspace, i.e. $\hat{u}^{(i)} \in \mathcal{U}$
How to approximate \( \text{span}\{u^{(1)}, \ldots, u^{(nev)}\} \) (II)

Let \((\delta_\ell, \nu^{(\ell)})\), \(\ell = 1, \ldots, d\), denote the eigenpairs of \((B, M_B)\).

**Higher-order resolvent expansions**

- Exploit \(\psi \geq 1\) terms of the formula \(B^{-1}_\lambda = B^{-1}_\sigma \sum_{\theta=0} B_M B^{-1}_\sigma \theta \): \[
\|u^{(i)} - \hat{u}^{(i)}\|_{M_B} \leq \left\| \sum_{\ell=1}^{\ell=d} \gamma_\ell (\lambda - \sigma)^{\psi+1} - \epsilon_\ell (\lambda - \sigma)^{\psi} \nu^{(\ell)} \right\|_{M_B} \]

\[
= \sum_{\ell=1}^{\ell=d} \frac{\gamma_\ell (\lambda - \sigma)^{\psi+1} - \epsilon_\ell (\lambda - \sigma)^{\psi}}{(\delta_\ell - \lambda)(\delta_\ell - \sigma)^{\psi}} \nu^{(\ell)} \]
How to approximate $\text{span}\{u^{(1)}, \ldots, u^{(nev)}\}$ (II)

Let $(\delta_\ell, v^{(\ell)})$, $\ell = 1, \ldots, d$, denote the eigenpairs of $(B, M_B)$.

**Higher-order resolvent expansions**

- Exploit $\psi \geq 1$ terms of the formula $B_\lambda^{-1} = B_\sigma^{-1} \sum_{\theta=0}^{\psi} [(\lambda - \sigma)M_B B_\sigma^{-1}]^\theta$:

$$
\|u^{(i)} - \hat{u}^{(i)}\|_{M_B} \leq \left\| \sum_{\ell=1}^{\ell=d} \gamma_\ell (\lambda - \sigma)^{\psi+1} - \epsilon_\ell (\lambda - \sigma)^\psi \frac{v^{(\ell)}}{(\delta_\ell - \lambda)(\delta_\ell - \sigma)^\psi} \right\|_{M_B}
$$

**Include eigenvectors of $(B, M_B)$ in $U$**

- If we also include the eigenvectors associated with the $\kappa$ eigenvalues of $(B, M_B)$ lying the closest to $\sigma$:

$$
\|u^{(i)} - \hat{u}^{(i)}\|_{M_B} \leq \left\| \sum_{\ell=\kappa+1}^{\ell=d} \frac{\gamma_\ell (\lambda - \sigma)^{\psi+1} - \epsilon_\ell (\lambda - \sigma)^\psi}{(\delta_\ell - \lambda)(\delta_\ell - \sigma)^\psi} v^{(\ell)} \right\|_{M_B}
$$
The RF-DDES scheme

RF-DDES is a RR approach on a basis of the subspace \( \mathcal{Z} = \mathcal{U} \oplus \mathcal{Y} \)

- \( \mathcal{Y} = \text{range}\{Q\} \), where \( Q \) is the Krylov basis formed by applying Lanczos to \( \Re\left\{ \sum_{\ell=1}^{Nc} \omega_\ell S_{\zeta_\ell}^{-1} \right\} \).
The RF-DDES scheme

RF-DDES is a RR approach on a basis of the subspace $\mathcal{Z} = \mathcal{U} \oplus \mathcal{Y}$

- $\mathcal{Y} = \text{range}\{Q\}$, where $Q$ is the Krylov basis formed by applying Lanczos to $\Re\{\sum_{\ell=1}^{N_c} \omega_{\ell} S_{\zeta_{\ell}}^{-1}\}$.
- $\mathcal{U} = \text{range}\{\bar{V}, U_1, U_2\}$ where

$$U_1 = -\left[ B_\sigma^{-1} EQ, \ldots, (B_{\sigma} M_B)^{\psi-1} B_\sigma^{-1} EQ \right],$$

$$U_2 = \left[ B_\sigma^{-1} M_E Q, \ldots, (B_{\sigma} M_B)^{\psi-1} B_\sigma^{-1} M_E Q \right].$$
The RF-DDES scheme

RF-DDES is a RR approach on a basis of the subspace $\mathcal{Z} = \mathcal{U} \oplus \mathcal{Y}$

- $\mathcal{Y} = \text{range}\{Q\}$, where $Q$ is the Krylov basis formed by applying Lanczos to $\Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell S_{\zeta_\ell}^{-1} \right\}$.

- $\mathcal{U} = \text{range}\{\bar{V}, U_1, U_2\}$ where
  
  \[
  U_1 = - \left[ B_{\sigma}^{-1} EQ, \ldots, (B_{\sigma} M_B)^{\psi-1} B_{\sigma}^{-1} EQ \right],
  \]
  
  \[
  U_2 = \left[ B_{\sigma}^{-1} M_E Q, \ldots, (B_{\sigma} M_B)^{\psi-1} B_{\sigma}^{-1} M_E Q \right],
  \]

- $\bar{V}$ includes the eigenvectors associated with the $nev_B$ eigenvalues lying the closest to $\sigma$ for each $(B^{(j)}_\sigma, M_B^{(j)})$, $j = 1, \ldots, p$
The RF-DDES scheme

RF-DDES is a RR approach on a basis of the subspace \( \mathcal{Z} = \mathcal{U} \oplus \mathcal{Y} \)

- \( \mathcal{Y} = \text{range}\{Q\} \), where \( Q \) is the Krylov basis formed by applying Lanczos to \( \Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell S^{-1}_\ell \right\} \).
- \( \mathcal{U} = \text{range}\{\bar{V}, U_1, U_2\} \) where
  \[
  U_1 = - \left[ B^{\sigma^{-1}}_E Q, \ldots, (B_\sigma M_B)^{\psi^{-1}} B^{\sigma^{-1}}_E Q \right], \\
  U_2 = \left[ B^{\sigma^{-1}} M_E Q, \ldots, (B_\sigma M_B)^{\psi^{-1}} B^{\sigma^{-1}} M_E Q \right],
  \]
- \( \bar{V} \) includes the eigenvectors associated with the \text{nev}_B eigenvalues lying the closest to \( \sigma \) for each \( (B^{(j)}_\sigma, M^{(j)}_B) \), \( j = 1, \ldots, p \)
- The subspace \( \mathcal{U} \) is formed independently in each one of the \( p \) subdomains
**The RF-DDES scheme**

RF-DDES is a RR approach on a basis of the subspace $\mathcal{Z} = \mathcal{U} \oplus \mathcal{Y}$

- $\mathcal{Y} = \text{range}\{Q\}$, where $Q$ is the Krylov basis formed by applying Lanczos to $\Re e \left\{ \sum_{\ell=1}^{N_c} \omega_\ell S^{-1}_{\zeta_\ell} \right\}$.

- $\mathcal{U} = \text{range}\{\bar{V}, U_1, U_2\}$ where

  $$U_1 = -\left[ B_{\sigma}^{-1} EQ, \ldots, (B_{\sigma} M_B)^{\psi-1} B_{\sigma}^{-1} EQ \right],$$

  $$U_2 = \left[ B_{\sigma}^{-1} M_E Q, \ldots, (B_{\sigma} M_B)^{\psi-1} B_{\sigma}^{-1} M_E Q \right],$$

- $\bar{V}$ includes the eigenvectors associated with the $nev_B$ eigenvalues lying the closest to $\sigma$ for each $(B^{(j)}_{\sigma}, M^{(j)}_B)$, $j = 1, \ldots, p$

- The subspace $\mathcal{U}$ is formed independently in each one of the $p$ subdomains

- Only real arithmetic need be exploited to form $\mathcal{U}$
The RF-DDES scheme

RF-DDES is a RR approach on a basis of the subspace $\mathcal{Z} = U \oplus \mathcal{Y}$

- $\mathcal{Y} = \text{range}\{Q\}$, where $Q$ is the Krylov basis formed by applying Lanczos to $\Re\left\{\sum_{\ell=1}^{N_c} \omega_\ell S_{\zeta_\ell}^{-1}\right\}$.
- $U = \text{range}\{\bar{V}, U_1, U_2\}$ where
  
  $$U_1 = - \left[ B_{\sigma}^{-1} EQ, \ldots, (B_{\sigma} M_B)^{\psi-1} B_{\sigma}^{-1} EQ \right],$$
  $$U_2 = \left[ B_{\sigma}^{-1} M_E Q, \ldots, (B_{\sigma} M_B)^{\psi-1} B_{\sigma}^{-1} M_E Q \right],$$

- $\bar{V}$ includes the eigenvectors associated with the $\text{nev}_B$ eigenvalues lying the closest to $\sigma$ for each $(B_{\sigma}^{(j)}, M_B^{(j)}), \ j = 1, \ldots, p$
- The subspace $U$ is formed independently in each one of the $p$ subdomains
- Only real arithmetic need be exploited to form $U$
- When $\psi$ resolvent terms are kept, we will write RF-DDES($\psi$)
Contents

1 Introduction

2 The domain decomposition (DD) viewpoint and the AMLS approach

3 The Rational Filtering DD Eigenvalue Solver (RF-DDES)

4 Experiments
   • Comparisons against rational filtering Krylov
Experiments

Implementation and computing environment

Hardware
- Experiments performed at the mesabi linux cluster at Minnesota Supercomputing Institute
- 741 two-socket nodes, each socket equipped with an Intel Haswell E5-2680v3 processor and 32 GB of memory

Software
- All methods were implemented in C++ and on top of PETSc (MPI)
- Linked to METIS, PARDISO, MUMPS, and MKL
- Compiled with mpiicpc (-O3)

Parameters and details
- Default values: $p = 2$, $N_c = 2$, $nev_B = 100$, and $\sigma = 0$
- All times are listed in seconds
- All experiments are performed in 64-bit arithmetic
Approximation of the $nev = 100$ algebraically smallest eigenvalues of matrix bcsstk39
Approximation of the $\text{nev} = 100$ algebraically smallest eigenvalues of pencil $qa8fk/qafm$

<table>
<thead>
<tr>
<th>Eigenvalue index</th>
<th>Relative error $\text{nev}_B=50$</th>
<th>Relative error $\text{nev}_B=100$</th>
<th>Relative error $\text{nev}_B=200$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-14}$</td>
<td>RF-DDES(1)</td>
<td>RF-DDES(2)</td>
<td>RF-DDES(3)</td>
</tr>
<tr>
<td>$10^{-12}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-10}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-8}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-4}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{-2}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$10^{0}$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Graphs showing the relative error for different values of $\text{nev}_B$.
Figure: Matrix: “FDmesh1” (2D Laplacian of size $n = 160 \times 150$). Results are reported for all different combinations of $p = 2, 4, 8$ and $p = 16$, and $N_c = 1, 2, 4, 8$ and $N_c = 16$. Interval: $[\alpha, \beta] = [\lambda_1, \lambda_{200}]$. 
Figure: The leading 250 singular values of $\Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell S(\zeta_\ell)^{-1} \right\}$ for matrix “FDmesh1”. Left: $p = 2$. Right: $p = 8$. 
Rational Filtering Krylov (RF-KRYLOV)

Algorithm

0. **Start with** $q^{(1)} \in \mathbb{R}^n$ \textit{s.t.} $\|q^{(1)}\|_2 = 1$
1. For $\mu = 1, 2, \ldots$
2. Compute $w = \Re \left\{ \sum_{\ell=1}^{N_c} \omega_\ell (A - \zeta_\ell M)^{-1} M q^{(\mu)} \right\}$
3. For $\kappa = 1, \ldots, \mu$
4. $h_{\kappa,\mu} = w^T q^{(\kappa)}$, $w = w - h_{\kappa,\mu} q^{(\kappa)}$
5. End
6. $h_{\mu+1,\mu} = \|w\|_2$
7. If $h_{\mu+1,\mu} \neq 0$
8. $q^{(\mu+1)} = w / h_{\mu+1,\mu}$
9. EndIf
10. Check convergence
11. End
12. Return Ritz values located inside $[\alpha, \beta]$ and associated Ritz vectors
A comparison of RF-KRYLOV and RF-DDES (I)

Table: Wall-clock times of RF-KRYLOV and RF-DDES using $\tau = 2$, 4, 8, 16 and $\tau = 32$ computational cores. RFD(2) and RFD(4) denote RF-DDES with $p = 2$ and $p = 4$ subdomains, respectively.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>$nev = 100$</th>
<th></th>
<th>$nev = 200$</th>
<th></th>
<th>$nev = 300$</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RFK</td>
<td>RFD(2)</td>
<td>RFD(4)</td>
<td>RFK</td>
<td>RFD(2)</td>
<td>RFD(4)</td>
</tr>
<tr>
<td>shipsec8($\tau = 2$)</td>
<td>114</td>
<td>195</td>
<td>-</td>
<td>195</td>
<td>207</td>
<td>-</td>
</tr>
<tr>
<td>($\tau = 4$)</td>
<td>76</td>
<td>129</td>
<td>93</td>
<td>123</td>
<td>133</td>
<td>103</td>
</tr>
<tr>
<td>($\tau = 8$)</td>
<td>65</td>
<td>74</td>
<td>56</td>
<td>90</td>
<td>75</td>
<td>62</td>
</tr>
<tr>
<td>($\tau = 16$)</td>
<td>40</td>
<td>51</td>
<td>36</td>
<td>66</td>
<td>55</td>
<td>41</td>
</tr>
<tr>
<td>($\tau = 32$)</td>
<td>40</td>
<td>36</td>
<td>28</td>
<td>62</td>
<td>41</td>
<td>30</td>
</tr>
<tr>
<td>boneS01($\tau = 2$)</td>
<td>94</td>
<td>292</td>
<td>-</td>
<td>194</td>
<td>356</td>
<td>-</td>
</tr>
<tr>
<td>($\tau = 4$)</td>
<td>68</td>
<td>182</td>
<td>162</td>
<td>131</td>
<td>230</td>
<td>213</td>
</tr>
<tr>
<td>($\tau = 8$)</td>
<td>49</td>
<td>115</td>
<td>113</td>
<td>94</td>
<td>148</td>
<td>152</td>
</tr>
<tr>
<td>($\tau = 16$)</td>
<td>44</td>
<td>86</td>
<td>82</td>
<td>80</td>
<td>112</td>
<td>109</td>
</tr>
<tr>
<td>($\tau = 32$)</td>
<td>51</td>
<td>66</td>
<td>60</td>
<td>74</td>
<td>86</td>
<td>71</td>
</tr>
</tbody>
</table>
A comparison of RF-KRYLOV and RF-DDES (II)

Table: Wall-clock times of RF-KRYLOV and RF-DDES using $\tau = 2, 4, 8, 16$ and $\tau = 32$ computational cores. RFD(2) and RFD(4) denote RF-DDES with $p = 2$ and $p = 4$ subdomains, respectively.

<table>
<thead>
<tr>
<th>Matrix</th>
<th>$nev = 100$</th>
<th></th>
<th></th>
<th>$nev = 200$</th>
<th></th>
<th></th>
<th>$nev = 300$</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RFK</td>
<td>RFD(2)</td>
<td>RFD(4)</td>
<td>RFK</td>
<td>RFD(2)</td>
<td>RFD(4)</td>
<td>RFK</td>
<td>RFD(2)</td>
<td>RFD(4)</td>
</tr>
<tr>
<td>FDmesh2($\tau = 2$)</td>
<td>241</td>
<td>85</td>
<td>-</td>
<td>480</td>
<td>99</td>
<td>-</td>
<td>731</td>
<td>116</td>
<td>-</td>
</tr>
<tr>
<td>($\tau = 4$)</td>
<td>159</td>
<td>34</td>
<td>63</td>
<td>305</td>
<td>37</td>
<td>78</td>
<td>473</td>
<td>43</td>
<td>85</td>
</tr>
<tr>
<td>($\tau = 8$)</td>
<td>126</td>
<td>22</td>
<td>23</td>
<td>228</td>
<td>24</td>
<td>27</td>
<td>358</td>
<td>27</td>
<td>31</td>
</tr>
<tr>
<td>($\tau = 16$)</td>
<td>89</td>
<td>16</td>
<td>15</td>
<td>171</td>
<td>17</td>
<td>18</td>
<td>256</td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td>($\tau = 32$)</td>
<td>51</td>
<td>12</td>
<td>12</td>
<td>94</td>
<td>13</td>
<td>14</td>
<td>138</td>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td>FDmesh3($\tau = 2$)</td>
<td>1021</td>
<td>446</td>
<td>-</td>
<td>2062</td>
<td>502</td>
<td>-</td>
<td>3328</td>
<td>564</td>
<td>-</td>
</tr>
<tr>
<td>($\tau = 4$)</td>
<td>718</td>
<td>201</td>
<td>281</td>
<td>1281</td>
<td>217</td>
<td>338</td>
<td>1844</td>
<td>237</td>
<td>362</td>
</tr>
<tr>
<td>($\tau = 8$)</td>
<td>423</td>
<td>119</td>
<td>111</td>
<td>825</td>
<td>132</td>
<td>126</td>
<td>1250</td>
<td>143</td>
<td>141</td>
</tr>
<tr>
<td>($\tau = 16$)</td>
<td>355</td>
<td>70</td>
<td>66</td>
<td>684</td>
<td>77</td>
<td>81</td>
<td>1038</td>
<td>88</td>
<td>93</td>
</tr>
<tr>
<td>($\tau = 32$)</td>
<td>177</td>
<td>47</td>
<td>49</td>
<td>343</td>
<td>51</td>
<td>58</td>
<td>706</td>
<td>62</td>
<td>82</td>
</tr>
</tbody>
</table>
**Table:** Number of iterations performed by RF-KRYLOV (denoted as RFK) and RF-DDES (denoted as RFD(p)).

<table>
<thead>
<tr>
<th></th>
<th>$nev = 100$</th>
<th>$nev = 200$</th>
<th>$nev = 300$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RFK</td>
<td>RFD(2)</td>
<td>RFD(4)</td>
</tr>
<tr>
<td>shipsec8</td>
<td>280</td>
<td>170</td>
<td>180</td>
</tr>
<tr>
<td>boneS01</td>
<td>240</td>
<td>350</td>
<td>410</td>
</tr>
<tr>
<td>FDmesh2</td>
<td>200</td>
<td>100</td>
<td>170</td>
</tr>
<tr>
<td>FDmesh3</td>
<td>280</td>
<td>150</td>
<td>230</td>
</tr>
</tbody>
</table>

**Table:** Maximum relative errors returned by RF-DDES.

<table>
<thead>
<tr>
<th></th>
<th>$nev = 100$</th>
<th>$nev = 200$</th>
<th>$nev = 300$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$nev_B$</td>
<td>25</td>
<td>50</td>
</tr>
<tr>
<td>shipsec8</td>
<td>1.4e-3</td>
<td>2.2e-5</td>
<td>2.4e-6</td>
</tr>
<tr>
<td>boneS01</td>
<td>5.2e-3</td>
<td>7.1e-4</td>
<td>2.2e-4</td>
</tr>
<tr>
<td>FDmesh2</td>
<td>4.0e-5</td>
<td>2.5e-6</td>
<td>1.9e-7</td>
</tr>
<tr>
<td>FDmesh3</td>
<td>6.2e-5</td>
<td>8.5e-6</td>
<td>4.3e-6</td>
</tr>
</tbody>
</table>
Amount of time spent on orthonormalization

**Figure:** Left: “FDmesh2” \((n = 250,000)\). Right: “FDmesh3” \((n = 1,000,000)\).
Runtimes for MPI-only implementation (nev = 300)

Figure: Left: “shipsec8”. Right: “FDmesh2”.

[Bar charts showing runtimes for different numbers of processes (p) and interfaces (Interface, Interior, Total).]
Conclusion

Summary

The main features of RF-DDES:

- No estimation of \( nev \) is needed
- Orthogonalization is applied to vectors whose length is equal to the number of interface variables
- The part of the solution associated with the interior variables is computed in real arithmetic
- Ability to exploit a possible low-rank of \( y^{(1)}, \ldots, y^{(nev)} \)
- Typically, not as accurate as RF-KRYLOV (do we always need high accuracy?)

Considerations

- RF-DDES is well-suited for 2D problems. What about 3D?
- Multi-MPI implementations are possible
Technical reports related to this talk

Main reference:
- V. Kalantzis, Y. Xi, and Y. Saad, "Domain decomposition Krylov rational filtering techniques for symmetric generalized eigenvalue problems".

See also:
- V. Kalantzis, J. Kestyn, E. Polizzi, and Y. Saad, "Domain Decomposition Approaches for Accelerating Contour Integration Eigenvalue Solvers for Symmetric Eigenvalue Problems".

http://www-users.cs.umn.edu/kalantzi/