OPTIMIZING RATIONAL FILTERS FOR INTERIOR EIGENVALUE SOLVERS

October 22, 2019 | E. Di Napoli, Konrad Köllnig, Jan Winkelmann |
OUTLINE

From spectrum slicing to load balancing

A roadmap to filter optimization

From subspace to best worst-case convergence rate
From spectrum slicing to load balancing

A roadmap to filter optimization

From subspace to best worst-case convergence rate
The problem

\[ Au = \lambda Bu, \quad \lambda \in [a, b], \quad A, B \in \mathbb{C}^{n \times n} \quad (1) \]

The domain

The projector

\[ r(A, B) := \sum_{i=1}^{n} \beta_i (A - Bz_i)^{-1} B \approx \frac{1}{2\pi i} \oint_{\Gamma} (A - zB)^{-1} B \, dz \equiv \sum_{\lambda_j \in [a, b]} u_j u_j^T B \]
METHOD

Repeat Until convergence:

1. Filter a block of vectors $V \leftarrow r(A, B)V = \sum_{i=1}^{n} \beta_i (A - Bz_i)^{-1} BV$
2. Re-orthogonalize the vectors outputted by the filter; $V = QR$.
3. Compute the Rayleigh quotient $G = Q^\dagger \tilde{A} Q$.
4. Compute the primitive Ritz pairs $(\Lambda, Y)$ by solving for $GY = Y\Lambda$.
5. Compute the approximate Ritz pairs $(\Lambda, V \leftarrow QY)$.

End Repeat

Core elements

1. Relies on good estimates of number $\mu[a \ b]$ of eigenvalues in $[a \ b]$
2. Solve for multiple right-hand side linear systems $(A - Bz_i)W = \beta_i BV$ per complex pole $z_i$
3. Accuracy depends on the accuracy of the projector $r(A, B)$
WHY THIS METHOD?

\[ \lambda_{\text{min}} \rightarrow \text{few eigenvalues} \rightarrow \text{cluster} \rightarrow \lambda_{\text{max}} \]

Access to massively parallel computing clusters

Member of the Helmholtz Association

October 22, 2019
Access to **massively parallel computing clusters**
PARALLELISM

\[(A - Bz_1)W = \beta_1 BV \]

\[(A - Bz_1)w_1 = \beta_1 Bv_1 \]

\[\vdots\]

\[(A - Bz_1)w_k = \beta_1 Bv_k \]

\[(A - Bz_n)W = \beta_n BV \]

\[\vdots\]

\[(A - Bz_n)w_k = \beta_n Bv_k \]
**LOAD BALANCING**

1. **Level 1** is influenced by
   - the evenness in the distribution of number $\mu_{[a_j b_j]}$ of eigenvalues in each sub-interval $[a_j b_j]$;
   - the **effectiveness** of the projector $r(A, B)$ in filtering the subspace corresponding to each single interval $[a_j b_j]$.

2. **Level 2** is influenced by
   - the time to solution for linear systems $(A - Bz_i)W = \beta_i BV$ defined by the same matrices but distinct shifts (poles) $z_i$ and RHS coefficients $\beta_i$;
   - the **efficiency** of the projector $r(A, B)$ in regulating the number of subspace iterations until convergence;
   - the number $\mu_{[a_j b_j]}$ of eigenvalues in $[a_j b_j]$ which is directly related to the size of the RHS of the linear systems.

3. **Level 3** is influenced by
   - the time to solution for linear systems $(A - Bz_i)w_k = \beta_i Bv_k$ defined by the same matrices but distinct RHS $v_k$;
   - the **efficiency** of the projector $r(A, B)$ in regulating the number of subspace iteration until convergence.
THREE ISSUES

Eigenvalue distribution across sub-intervals

√ Kernel Polynomial Method or Lanczos DoS\textsuperscript{a} + Stochastic estimate\textsuperscript{b} are a good approach to address issue.

\textsuperscript{a}L. Lin et al. DOI:10.1137/130934283
\textsuperscript{b}E. Di Napoli et al. DOI:abs/10.1002/nla.2048

Predicting time to solution for linear solver

Ongoing work using supervised classification and linear solver + pre-conditioner matching\textsuperscript{a}

\textsuperscript{a}In collaboration with V. Ejikhout at TACC

Efficiency and robustness of rational filter

⇒ Optimize filter using Non-linear Least Squares for best worst-case convergence\textsuperscript{a}.

\textsuperscript{a}J. Winkelmann et al. DOI:10.3389/fams.2019.00005 & K. Köllnig et al. TBS to SISC
TOPIC

From spectrum slicing to load balancing

A roadmap to filter optimization

From subspace to best worst-case convergence rate
SETTING UP THE PROBLEM

Ideal filter

\[ \mathbb{1}_{(a,b)}(x) = \begin{cases} 1, & \text{if } x \in [a, b], \\ 0, & \text{otherwise} \end{cases} \]  \hspace{1cm} (2)

(Symmetric) Rational filter

\[ r_{\beta,z}(x) := \sum_{i=1}^{m} \frac{\beta_i}{x - z_i} + \frac{\overline{\beta_i}}{x - \overline{z_i}} - \frac{\beta_i}{x + z_i} - \frac{\overline{\beta_i}}{x + \overline{z_i}}, \quad x \in \mathbb{R}, \quad \text{with } \beta \in \mathbb{C}^m, z \in (\mathbb{H}^+ \mathbb{R})^m \]  \hspace{1cm} (3)

Objective function

\[ f_\omega(\beta, z) := \int_{-\infty}^{\infty} \omega(x) \left( \mathbb{1}_{(a,b)}(x) - r_{\beta,z}(x) \right)^2 dx, \]  \hspace{1cm} (4)

Minimization problem

\[ \arg\min_{\beta \in \mathbb{C}^m, z \in (\mathbb{H}^+ \mathbb{R})^m} f_\omega(\beta, z). \]  \hspace{1cm} (5)
MINIMIZATION APPROACHES

1. **First approach:** Gradient descent

   \[ x^{(k+1)} = x^{(k)} + s \cdot \Delta x^{(k)} = x^{(k)} - s \cdot \nabla f_\omega(x) \bigg|_{x=x^{(k)}} , \quad s \geq 0 \quad x \equiv (\beta, z). \]  

   Slow (linear) convergence  
   Dependence of starting positions \( x^{(0)} \) and weight function \( \omega \)

2. **Second approach:** Levenberg-Mardquardt

   \( \xi(\beta, z) = 1_{(a, b)} - r_{\beta, z} \Rightarrow f_\omega(x) \equiv ||\xi(x)||_2^2 \)

   1. Set: \( H := \langle \nabla \xi(x^{(k)}), \nabla \xi(x^{(k)}) \rangle \)
   2. Solve: \( H \cdot \Delta x_{GN}^{(k)} = \langle \xi(x^{(k)}), \nabla \xi(x^{(k)}) \rangle = -\frac{1}{2} \nabla f_\omega(x^{(k)}) \)
   3. Update: \( x^{(k+1)} = x^{(k)} + s \cdot \Delta x_{GN}^{(k)} \).

   Faster convergence  
   Starting position: existing filters (e.g. Gauss-Legendre)

3. **Third approach:** Broyden-Fletcher-Goldfarb-Shanno (BFGS)
SLISE FILTERS

… using the BFGS algorithm

- Supports only real-valued objective functions $f_\omega : \mathbb{C}^m \times \mathbb{H}^+ \rightarrow \mathbb{R} \Rightarrow \tilde{f}_\omega : \mathbb{R}^4m \rightarrow \mathbb{R}$

  \[
  \tilde{f}(\begin{pmatrix}
  \Re(\beta^T) \\
  \Re(z^T) \\
  \Im(\beta^T) \\
  \Im(z^T)
  \end{pmatrix}) := f(\Re(\beta) + i\Im(\beta), \Re(z) + i\Im(z)).
  \]  
  \hspace{1cm} (7)

- The inverse Hessian of $\tilde{f}_\omega$ is recursively defined as

  \[
  H_0 := I_{4m}, \quad H_{k+1} := (I_{4m} - \frac{s_k y_k^T}{y_k^T s_k}) H_k (I_{4m} - \frac{y_k s_k^T}{y_k^T s_k}) + \frac{s_k s_k^T}{y_k^T s_k},
  \]  
  \hspace{1cm} (8)

  with

  \[
  s_k := x_{k+1} - x_k, \quad y_k := \nabla \tilde{f}(x_{k+1}) - \nabla \tilde{f}(x_k),
  \]  
  \hspace{1cm} (9)

Very fast convergence
(Still) dependent on weight function $\omega$
TOPIC

From spectrum slicing to load balancing

A roadmap to filter optimization

From subspace to best worst-case convergence rate
CONVENTIONS

In the rest of the slides we maintain the following notations:

- Standard interval $[a, b] \rightarrow [-1, 1]$,
- Active subspace size $M_0 = C \times \mu_{[a,b]}$ and $C \geq 1$,
- Gap parameter $G \in (0, 1)$ such that $G < 1 < G^{-1}$ ($-G^{-1} < -1 < -G$).

SINGLE TEST

- CNT matrix, $N = 12,450$ with $86,808$ nnz
- Interval $[a, b] = [-65.0, 4.96]$
- $M = 100$

(LARGE) BENCHMARK SET

- 2116 intervals defining the corresponding interior eigenproblem,
- Each interval contains between 5 and 20% of the total spectrum of Si$_2$ problem,
- Interval are selected based on “feature points”: neighborhood of an identifiable spectral feature, such as a spectral gap or a cluster.
SLISE FILTER EFFICIENCY

Convergence rate for subspace iteration solver (e.g. FEAST)

\[ \tau = \frac{|r(\lambda_{M_0+1})|}{|r(\lambda_{in})|}, \text{ with } |r(\lambda_{in})| = \min_{\lambda \in [-1,1]} |r(\lambda)| \]  

(10)

Performance profile: given a point \( x \) on the abscissa, the corresponding value \( \phi_r(x) \) of the graph indicates that for \( 100 \cdot \phi_r(x) \) percent of the benchmark problems the filter \( r \) is at most a factor of \( x \) worse than the fastest of all filters.
Best Worst-Case Convergence Rate (WCR)

Given a rational filter $r$ and some fixed gap parameter $G \in (0, 1)$, a filtered subspace iteration converges linearly, with probability one, at a convergence rate no larger than

$$w_G(r) = \frac{\max_{x \in [-\infty, -G^{-1}] \cup [G^{-1}, \infty]} |r(x)|}{\min_{x \in [-G, G]} |r(x)|},$$

as long as no eigenvalues lie within $[-G^{-1}, -G] \cup [G, G^{-1}]$.

New minimization problem

$$\begin{cases} 
\beta', z' & \leftarrow \arg\min_{\beta, z} f_{\omega'}(\beta, z) \\
\omega' & \leftarrow \arg\min_{\omega} w_G(r_{\beta', z'}[\omega]).
\end{cases}$$ \hspace{1cm} (11)

- Minimize WCR instead of Subspace Iteration convergence rate.
- Nested minimization: requires thousands of SLiSe “minimizations”.
- Derivative-free minimization: Nelder-Mead algorithm.
- Eliminate parameter dependence on weight functions.
- Best worst-case convergence of FEAST strongly correlates with WCR of filter,
- Size of the active subspace $M_0$ is a confounding factor: big values of $C$ mask the correlation between WCR and $\tau$,
- WiSe filters performance hardly depends on number of poles $m$. 
BENCHMARK SET WITH FEAST

- Number of poles fixed to $m = 4$,
- Confirms that FEAST with WiSe filter only influenced by WCR,
- For larger active subspaces Gauss-Legendre is competitive with WiSe but costs more FLOPs.
SUMMARY AND OUTLOOK

- WiSe filters depend almost exclusively on gap parameter $G$,
- WiSe filters offer a competitive edge when compared to the same solver using Gauss-Legendre and Zolotarev filters,
- WiSe filters are quite stable with respect to the convergence rate of the solver independently of the active subspace or the degree of the filter function,
- WiSe filters almost always minimize the total FLOP count required by FEAST to reach convergence.

Future work

- Integrating rational filters in the ChASE library
- Prediction of time to solution for linear systems solves,
- Filters for general complex eigenproblems.
THANK YOU

https://github.com/SimLabQuantumMaterials/SLiSeFilters.jl

https://doi.org/10.3389/fams.2019.00005

e.di.napoli@fz-juelich.de

http://www.fz-juelich.de/ias/jsc/slqm