Preconditioned Locally Harmonic Residual Methods for Interior Eigenvalue Computations

Eugene Vecharynski
Lawrence Berkeley National Laboratory
Computational Research Division

SIAM Conference on Applied Linear Algebra, October 29, 2015
Atlanta, GA
The Preconditioned Locally Harmonic Residual methods

A new class of robust block preconditioned iterative eigensolvers for computing interior eigenpairs

▶ **Hermitian** eigenproblems
  The PLHR algorithm (EV, A. Knyazev)

▶ **Non-Hermitian** eigenproblems
  The Generalized PLHR (GPLHR) algorithm (EV, C. Yang, F. Xue)
Problem

Compute a subset of $k$ right eigenpairs $(\lambda, x)$ of a non-Hermitian matrix pair $(A, B)$ that are closest to a given shift $\sigma \in \mathbb{C}$

$$Ax = \lambda Bx, \quad A, B \in \mathbb{C}^{n \times n}$$

$A, B$ can be extremely large

$A, B$ may not be explicitly given

$(A, B)$ is regular

$\sigma$ can point to the spectrum’s interior or exterior
Many origins of the problem

Study of excited states of molecules (complex scaling, EOM-CC)

Chemical reactions, stability analysis in fluid dynamics, crystal growth simulations, power systems, Markov chains, plasmas, ETC.
Available eigensolvers

- (Inexact) Inverse subspace iteration
  Sufficiently accurate \((A - \sigma B)^{-1}\) is needed

- Arnoldi (ARPACK)
  Sufficiently accurate \((A - \sigma B)^{-1}\) is needed

- Block Generalized Davidson (Morgan-92)
  Robustness issues, may need a larger search subspace

- JDQR/JDQZ (Fokkema et. al. SISC-98)
  “One-by-one” eigenpair computation, lack of BLAS3
Need an eigensolver that ...

- Avoids “shift-and-invert”
- Takes advantage of preconditioning, \( T \approx (A - \sigma B)^{-1} \)
- Performs block iteration (parallel performance, BLAS3)
- Is robust and reliable if memory is limited
- Handles standard and generalized, interior and exterior, eigenproblems in a uniform manner
- Is simple (to the extent a non-Hermitian solver can be ;-)
An example from the “Hermitian world”: LOBPCG

The LOBPCG algorithm (Knyazev SISC-01):

$$X^{(i+1)} \leftarrow \text{col} \left\{ X^{(i)}, \ T(AX^{(i)} - BX^{(i)}\Lambda^{(i)}), \ X^{(i-1)} \right\}$$

- Block iterations
- No inversions
- Short-term recurrence
- The Rayleigh–Ritz procedure over low-dimensional subspaces
An example from the “Hermitian world”: LOBPCG

The LOBPCG algorithm (Knyazev SISC-01):

\[ X^{(i+1)} \leftarrow \text{col} \left\{ X^{(i)}, T(AX^{(i)} - BX^{(i)}\Lambda^{(i)}), X^{(i-1)} \right\} \]

- Block iterations
- No inversions
- Short-term recurrence
- The Rayleigh–Ritz procedure over low-dimensional subspaces

Does NOT extend to non-Hermitian eigenproblems
Difficulties

1. The eigendecomposition $AX = BX \Lambda$ may not exist or $X$ can be ill-conditioned. Partial generalized Schur form should be computed

$$\begin{align*}
AV &= QR_A \\
BV &= QR_B
\end{align*}$$
$$\lambda_j = R_A(j,j)/R_B(j,j)$$

2. A correct definition of a block residual is not clear

3. The LOBPCG-like trial subspace is “too small”

4. The standard Rayleigh–Ritz may not be appropriate for interior eigenvalue computations
The “Q-free” partial generalized Schur form

- Partial generalized Schur form

\[
\begin{align*}
AV &= QR_A, \\
BV &= QR_B, \quad \lambda_j = R_A(j,j)/R_B(j,j)
\end{align*}
\]

- The “Q-free” partial generalized Schur form

\[
AVM_B = BVM_A, \quad \lambda_j = M_A(j,j)/M_B(j,j),
\]

where \( M_A, M_B \) are upper triangular.

- Gives a numerically stable analogue of \( AX\Lambda_B = BX\Lambda_A \)
- Allows defining a block residual \( AVM_B - BVM_A \)
The “Q-free” partial generalized Schur form

- Partial generalized Schur form

\[
\begin{align*}
AV &= QR_A, \\
BV &= QR_B, \\
\lambda_j &= R_A(j,j)/R_B(j,j)
\end{align*}
\]

- The “Q-free” partial generalized Schur form

\[
AVM_B = BVM_A, \quad \lambda_j = M_A(j,j)/M_B(j,j),
\]

where \(M_A, M_B\) are upper triangular.

- Gives a numerically stable analogue of \(AX\Lambda_B = BX\Lambda_A\)
- Allows defining a block residual \(AVM_B - BVM_A\)

Does the “Q-free” form exist for any regular pair \((A, B)\)?
The “$Q$-free” partial generalized Schur form

For any regular pair $(A, B)$, there exist

$$AVM_B = BV M_A,$$
$$\lambda_j = M_A(j,j)/M_B(j,j)$$

where

$$M_A = G_2 G^{-1} R_A, \quad M_B = I - G_1 G^{-1} R_A,$$

with an upper triangular

$$G = R_A G_1 + R_B G_2, \quad G(j,j) = 1,$$

and $G_1, G_2$ diagonal

$$G_1(j,j) = \begin{cases} 0, & |R_A(j,j)| < |R_B(j,j)| \\ \frac{1 - R_B(j,j)}{R_A(j,j)}, & \text{otherwise} \end{cases}$$

$$G_2(j,j) = \begin{cases} 1/R_B(j,j), & |R_A(j,j)| < |R_B(j,j)| \\ 1, & \text{otherwise} \end{cases}$$
The simplest GPLHR trial subspace

Given approximations $V^{(i)}$, $R^{(i)}_A$, and $R^{(i)}_B$, the “Q-free” partial generalized Schur form allows

- Defining a **preconditioned residual**

  \[ W^{(i)} = T(AV^{(i)}M^{(i)}_B - BV^{(i)}M^{(i)}_A), \]

  where $T$ is a preconditioning operator.

- Constructing a **LOBPCG-like trial subspace**

  \[ \mathcal{Z} = \text{col}\{V^{(i)}, W^{(i)}, P^{(i)}\} \]

  where $P^{(i)}$ is a block of additional search directions.
The simplest GPLHR trial subspace

Given approximations $V^{(i)}$, $R_A^{(i)}$, and $R_B^{(i)}$, the “Q-free” partial generalized Schur form allows

- Defining a preconditioned residual

$$W^{(i)} = T(AV^{(i)} M_B^{(i)} - BV^{(i)} M_A^{(i)}),$$

where $T$ is a preconditioning operator.

- Constructing a LOBPCG-like trial subspace

$$\mathcal{Z} = \text{col}\{V^{(i)}, W^{(i)}, P^{(i)}\}$$

where $P^{(i)}$ is a block of additional search directions.

How to construct a larger trial subspace?
The GPLHR trial subspace

- Find the corrections, such that

\[ A(V(i) + C(i))(M_B^{(i)} + \Delta_B^{(i)}) = B(V(i) + C(i))(M_A^{(i)} + \Delta_A^{(i)}), \quad V(i)^* C^{(i)} = 0 \]
The GPLHR trial subspace

Find the corrections, such that

\[ A(V^{(i)} + C^{(i)})(M_B^{(i)} + \Delta_B^{(i)}) = B(V^{(i)} + C^{(i)})(M_A^{(i)} + \Delta_A^{(i)}), \quad V^{(i)*}C^{(i)} = 0 \]

The correction \( C^{(i)} \) can be approximated by \( C \) that satisfies the generalized Sylvester equation

\[ L(C) = F, \quad V^{(i)*}C = 0, \]

where

\[ L(C) \equiv (P_{Q\perp} A P_{V\perp}) C M_B^{(i)} - (P_{Q\perp} B P_{V\perp}) C M_A^{(i)} \]

and

\[ F \equiv -P_{Q\perp}(AV^{(i)} M_B^{(i)} - BV^{(i)} M_A^{(i)}). \]

Here \( P_{V\perp} = I - V^{(i)}V^{(i)*}, P_{Q\perp} = I - Q^{(i)}Q^{(i)*} \), and \( Q^{(i)} \) is an orthonormal basis of \( \text{col}\{(A - \sigma B)V^{(i)}\} \).
The GPLHR trial subspace

Define a preconditioner

\[ T_L = (I - VV^*) T (I - QQ^*) \], \quad T \approx (A - \sigma B)^{-1} \]

for the generalized Sylvester’s operator

\[ L(C) \equiv (P_{Q \perp} A P_{V \perp}) CM_B^{(i)} - (P_{Q \perp} B P_{V \perp}) CM_A^{(i)} \]

Apply \( m \) steps of a preconditioned block Arnoldi to \( L(C) = F \)

1: Set \( V \leftarrow V^{(i)}, M_A \leftarrow M_A^{(i)}, \) and \( M_B \leftarrow M_B^{(i)} \);
2: Set \( W \leftarrow (I - VV^*) T (I - QQ^*) (AVM_B - BVM_A);(\equiv T_L(F)) \)
3: \( W \leftarrow \text{orth}(W); S_0 \leftarrow W; S \leftarrow [\ ]; \)
4: for \( l = 1 \rightarrow m \) do
5: \( S_l \leftarrow (I - VV^*) T (I - QQ^*) (AS_{l-1} M_B - BS_{l-1} M_A); (\equiv T_L(L(S_{l-1}))) \)
6: \( S_l \leftarrow S_l - W(W^* S_l); S_l \leftarrow S_l - S(S^* S_l); \)
7: \( S_l \leftarrow \text{orth}(S_l); S \leftarrow [S S_l]; \)
8: end for

\( \Rightarrow \) The GPLHR trial subspace

\[ \mathcal{Z} = \text{col}\{V^{(i)}, W^{(i)}, S_1^{(i)}, S_2^{(i)}, \ldots, S_m^{(i)}, P^{(i)}\}. \]
The harmonic Schur–Rayleigh–Ritz (SRR)

- Find $V$, $M_B$, and $M_A$, such that (Petrov–Galerkin)

$$AVM_B - BVM_A \perp (A - \sigma B)Z, \quad \lambda_j \approx M_A(j,j)/M_B(j,j)$$

where columns of $V$ are in $Z$.
The harmonic Schur–Rayleigh–Ritz (SRR)

- Find $V$, $M_B$, and $M_A$, such that (Petrov–Galerkin)

$$AVM_B - BVM_A \perp (A - \sigma B)Z, \quad \lambda_j \approx M_A(j,j)/M_B(j,j)$$

where columns of $V$ are in $Z$.

- Solve the projected problem (the ordered QZ algorithm)

$$(U^*AZ)YM_B = (U^*BZ)YM_A,$$

where $Z = \text{orth}\{Z\}$ and $U = \text{orth}\{(A - \sigma I)Z\}$. 
The harmonic Schur–Rayleigh–Ritz (SRR)

- Find $V$, $M_B$, and $M_A$, such that (Petrov–Galerkin)

$$AVM_B - BVM_A \perp (A - \sigma B)Z, \quad \lambda_j \approx M_A(j,j)/M_B(j,j)$$

where columns of $V$ are in $Z$.

- Solve the projected problem (the ordered QZ algorithm)

$$(U^*AZ)YM_B = (U^*BZ)YM_A,$$

where $Z = \text{orth}\{Z\}$ and $U = \text{orth}\{(A - \sigma I)Z\}$.

- Set $M_A^{(i+1)} \leftarrow M_A$, $M_B^{(i+1)} \leftarrow M_B$, and $V^{(i+1)} \leftarrow ZY$. 
The GPLHR algorithm

**Input:** \( A, B, \sigma, V^{(0)}, T \approx (A - \sigma I)^{-1}, m. \)

**Output:** \( k \) eigenpairs of \((A, B)\) closest to \( \sigma \)

1. Use \( V^{(0)} \) to construct initial \( V, M_A, \) and \( M_B; P \leftarrow [ \ ]; \)
2. **while** convergence not reached **do**
3. \( W \leftarrow (I - VV^*)T(I - QQ^*)(AVM_B - BVM_A); W \leftarrow \text{orth}(W); \)
4. Apply \( m \) block Arnoldi steps to generate blocks \( S_1, \ldots, S_m; \)
5. Set \( Z \leftarrow \text{orth}([V, W, S_1, \ldots, S_m, P]); \)
6. Set \( U \leftarrow \text{orth}((A - \sigma B)Z); \)
7. \([\bar{R}_A, \bar{R}_B, \bar{Y}_L, \bar{Y}_R] \leftarrow \text{ordqz}(U^* AZ, U^* BZ, \sigma); \)
8. \( Y \leftarrow \bar{Y}_R(:, 1:k), Y_L \leftarrow \bar{Y}_L(:, 1:k); \)
9. \( R_A \leftarrow \bar{R}_A(1:k, 1:k), R_B \leftarrow \bar{R}_B(1:k, 1:k); \)
10. \( V \leftarrow ZY; Q \leftarrow UY_L; P \leftarrow Z\bar{Y}_R(:, k+1:2k); \)
11. \( G \leftarrow R_A G_1 + R_B G_2; M_A \leftarrow G_2 G^{-1} R_A; M_B \leftarrow I - G_1 G^{-1} R_A; \)
12. **end while**
13. Extract wanted eigenpairs \((X, \Lambda)\) using RR for \((A, B)\) over col\{\(V\}\).
The GPLHR algorithm

**Input:** $A$, $B$, $\sigma$, $V^{(0)}$, $T \approx (A - \sigma I)^{-1}$, $m$.

**Output:** $k$ eigenpairs of $(A, B)$ closest to $\sigma$

1. Use $V^{(0)}$ to construct initial $V$, $M_A$, and $M_B$; $P \leftarrow [ ]$;
2. **while** convergence not reached **do**
3. $W \leftarrow (I - VV^*)T(I - QQ^*)(AVM_B - BVM_A)$; $W \leftarrow \text{orth}(W)$;
4. Apply $m$ block Arnoldi steps to generate blocks $S_1, \ldots, S_m$;
5. Set $Z \leftarrow \text{orth}([V, W, S_1, \ldots, S_m, P])$;
6. Set $U \leftarrow \text{orth}((A - \sigma B)Z)$;
7. $[\tilde{R}_A, \tilde{R}_B, \tilde{Y}_L, \tilde{Y}_R] \leftarrow \text{ordqz}(U^*AZ, U^*BZ, \sigma)$;
8. $Y \leftarrow \tilde{Y}_R(:,1:k)$, $Y_L \leftarrow \tilde{Y}_L(:,1:k)$;
9. $R_A \leftarrow \tilde{R}_A(1:k,1:k)$, $R_B \leftarrow \tilde{R}_B(1:k,1:k)$;
10. $V \leftarrow ZY$; $Q \leftarrow UY_L$; $P \leftarrow Z\tilde{Y}_R(:,k+1:2k)$;
11. $G \leftarrow R_AG_1 + R_BG_2$; $M_A \leftarrow G_2G^{-1}R_A$; $M_B \leftarrow I - G_1G^{-1}R_A$;
12. **end while**
13. Extract wanted eigenpairs $(X, \Lambda)$ using RR for $(A, B)$ over col$\{V\}$.
Choice of the additional search directions $P$

<table>
<thead>
<tr>
<th>Problem</th>
<th>$P_{\text{thick}}^{(i)}$</th>
<th>$P_{\text{LOBPCG}}^{(i)}$</th>
<th>W/o $P^{(i)}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A15428</td>
<td>12</td>
<td>14</td>
<td>16</td>
</tr>
<tr>
<td>AF23560</td>
<td>10</td>
<td>DNC</td>
<td>153</td>
</tr>
<tr>
<td>CRY10000</td>
<td>27</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>DW8192</td>
<td>118</td>
<td>55</td>
<td>329</td>
</tr>
<tr>
<td>LSTAB_NS</td>
<td>38</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>MHD4800</td>
<td>137</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>PDE2961</td>
<td>13</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>QH882</td>
<td>15</td>
<td>DNC</td>
<td>30</td>
</tr>
<tr>
<td>RDB3200L</td>
<td>10</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>UTM1700</td>
<td>16</td>
<td>23</td>
<td>21</td>
</tr>
</tbody>
</table>

**Table:** Numbers of iterations performed by GPLHR to compute $k = 5$ eigenpairs with different choices of the search directions $P^{(i)}$; $m = 1$. 


The GPLHR algorithm

**Input:** \( A, B, \sigma, V^{(0)}, T \approx (A - \sigma I)^{-1}, m. \)

**Output:** \( k \) eigenpairs of \((A, B)\) closest to \( \sigma \)

1. Use \( V^{(0)} \) to construct initial \( V, M_A, \) and \( M_B; P \leftarrow [ ] \);
2. **while** convergence not reached **do**
3. \( W \leftarrow (I - VV^*) T(1 - QQ^*)(AVM_B - BVM_A); W \leftarrow \text{orth}(W); \)
4. Apply \( m \) block Arnoldi steps to generate blocks \( S_1, \ldots, S_m; \)
5. Set \( Z \leftarrow \text{orth}([V, W, S_1, \ldots, S_m, P]); \)
6. Set \( U \leftarrow \text{orth}((A - \sigma B)Z); \)
7. \([\tilde{R}_A, \tilde{R}_B, \tilde{Y}_L, \tilde{Y}_R] \leftarrow \text{ordqz}(U^*AZ, U^*BZ, \sigma); \)
8. \( Y \leftarrow \tilde{Y}_R(:,1:k), Y_L \leftarrow \tilde{Y}_L(:,1:k); \)
9. \( R_A \leftarrow \tilde{R}_A(1:k,1:k), R_B \leftarrow \tilde{R}_B(1:k,1:k); \)
10. \( V \leftarrow ZY; Q \leftarrow UY_L; P \leftarrow Z\tilde{Y}_R(:,k+1:2k); \)
11. \( G \leftarrow R_A G_1 + R_B G_2; M_A \leftarrow G_2 G^{-1} R_A; M_B \leftarrow I - G_1 G^{-1} R_A; \)
12. **end while**
13. Extract wanted eigenpairs \((X, \Lambda)\) using RR for \((A, B)\) over col\{\( V \}\).
Influence of the projectors

<table>
<thead>
<tr>
<th>Problem</th>
<th>With proj.</th>
<th>W/o proj.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A15428</td>
<td>14</td>
<td>DNC</td>
</tr>
<tr>
<td>AF23560</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>CRY10000</td>
<td>13</td>
<td>17</td>
</tr>
<tr>
<td>DW8192</td>
<td>44</td>
<td>45</td>
</tr>
<tr>
<td>LSTAB_NS</td>
<td>49</td>
<td>84</td>
</tr>
<tr>
<td>MHD4800</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>PDE2961</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>RDB3200L</td>
<td>12</td>
<td>21</td>
</tr>
<tr>
<td>UTM1700</td>
<td>25</td>
<td>26</td>
</tr>
</tbody>
</table>

Table: Numbers of iterations performed by GPLHR with and without the projection at the preconditioning step to compute $k = 10$ eigenpairs; $m = 1$. 
Few algorithmic details

- The preconditioner $T \approx (A - \sigma B)^{-1}$ can be given by a general procedure, e.g., by several steps of a preconditioned iterative linear solver applied to $(A - \sigma B)w = r$.

- The dimension of the search subspace is $(m + 3)k$, where $k$ is the block size. In practice, $m$ is between 1 and 5.

- The algorithms can be implemented with $m + 1$ matrix-block multiplications and $m + 1$ block preconditioning operations per iteration.

- Converged Schur vectors should be locked.

- The parameter $m$ can be adaptively chosen, e.g., increased after a number of eigenpairs vectors has converged.
Numerical examples: benchmark systems

Experiments in Q-Chem 4.2: compare GPLHR and Davidson with the EOM-IP method for calculation of ionized potentials

Benchmark systems:

- Hydrated photoactive yellow protein chromophore \textit{PYP\textsubscript{a-W}}\textsubscript{p} (left). The 6-31+G(d,p) basis set (292 basis functions).
- Dihydrated 1,3-dimethyluracil \((mU)_{2-}(H\textsubscript{2}O)_{2}\) (right). The 6-311+G(d,p) basis set (336 basis functions).
Davidson vs GPLHR: iteration count

The GPLHR iteration count decreases as \( m \) grows

GPLHR requires less iterations than Davidson

Each GPLHR iterations requires more \( (\text{nev} \times (m + 1)) \) matvecs per iteration

Figure: **Left:** PYPa-\( W_p/6-31+G(d,p) \), convergence for the eigenvalues 4.11 and 4.20 eV; **Right:** \( (mU)_2-(H_2O)_2/6-311+G(d,p) \), convergence for eigenvalues 8.89 and 10.04 eV
Davidson vs GPLHR: low-lying eigenvalues ($\sigma = 0\, a.u.$)

$$(mU)_2 - (H_2O)_2/6-311+G(d,p)$$

### Davidson

<table>
<thead>
<tr>
<th>nev</th>
<th>niters</th>
<th>Max. # of stored vectors</th>
<th># matvec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>48</td>
<td>24</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>76</td>
<td>38</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
<td>120</td>
<td>99</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>120</td>
<td>118</td>
</tr>
</tbody>
</table>

### GPLHR

<table>
<thead>
<tr>
<th>nev</th>
<th>niters</th>
<th>m</th>
<th>Max. # of stored vectors</th>
<th># matvec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>1</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>1</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>1</td>
<td>40</td>
<td>43</td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>1</td>
<td>80</td>
<td>118</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>2</td>
<td>120</td>
<td>151</td>
</tr>
</tbody>
</table>
Davidson vs GPLHR: the interior eigenvalues ($\sigma = 11 \text{a.u.}$)

**PYPa-W\textsubscript{$p$/6-31+G(d,p)}**

<table>
<thead>
<tr>
<th>nev</th>
<th>niter</th>
<th>Max. # of stored vectors</th>
<th># matvec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DNC</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>2</td>
<td>DNC</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>3</td>
<td>DNC</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>5</td>
<td>DNC</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>nev</th>
<th>niter</th>
<th>m</th>
<th>Max. # of stored vectors</th>
<th># matvec</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>1</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>16</td>
<td>18</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>1</td>
<td>24</td>
<td>27</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>1</td>
<td>40</td>
<td>63</td>
</tr>
</tbody>
</table>
Figure: $\sigma = 8$ for CRY10000 (left) and $\sigma = 0$ for LSTAB_NS (right)
## JDQZ vs GPLHR

<table>
<thead>
<tr>
<th>Problem</th>
<th>$k$</th>
<th>$i t_G$</th>
<th>#it</th>
<th>#mv</th>
<th>#prec</th>
<th>#it</th>
<th>#mv</th>
<th>#prec</th>
</tr>
</thead>
<tbody>
<tr>
<td>MHD4800</td>
<td>1</td>
<td>0</td>
<td>20</td>
<td>41</td>
<td>40</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0</td>
<td>15</td>
<td>79</td>
<td>76</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>0</td>
<td>12</td>
<td>101</td>
<td>96</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>0</td>
<td>9</td>
<td>165</td>
<td>155</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>LSTAB_NS</td>
<td>1</td>
<td>25</td>
<td>5</td>
<td>261</td>
<td>260</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>25</td>
<td>49</td>
<td>5307</td>
<td>5304</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>25</td>
<td>28</td>
<td>4893</td>
<td>4888</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>25</td>
<td>36</td>
<td>12984</td>
<td>12974</td>
<td>DNC</td>
<td>DNC</td>
<td>DNC</td>
</tr>
<tr>
<td>UTM1700</td>
<td>1</td>
<td>15</td>
<td>5</td>
<td>161</td>
<td>160</td>
<td>MCV</td>
<td>MCV</td>
<td>MCV</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>15</td>
<td>13</td>
<td>963</td>
<td>960</td>
<td>16</td>
<td>263</td>
<td>280</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>15</td>
<td>13</td>
<td>1669</td>
<td>1664</td>
<td>31</td>
<td>511</td>
<td>545</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>15</td>
<td>21</td>
<td>5034</td>
<td>5024</td>
<td>166</td>
<td>2691</td>
<td>2865</td>
</tr>
</tbody>
</table>
Twenty eigenvalues of MHD4800 computed by GPLHR

Twenty eigenvalues of MHD4800 computed by JDQZ
ARPACK vs GPLHR

<table>
<thead>
<tr>
<th>Problem</th>
<th>$k$</th>
<th>GPLHR</th>
<th>ARPACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>A15876</td>
<td>10</td>
<td>194</td>
<td>2895</td>
</tr>
<tr>
<td>A15428</td>
<td>10</td>
<td>335</td>
<td>3006</td>
</tr>
<tr>
<td>AF23560</td>
<td>5</td>
<td>2</td>
<td>8</td>
</tr>
</tbody>
</table>

Table: Time required by GPLHR with the preconditioner $T$ given by ILU($10^{-3}$) and ARPACK with the shift-and-invert operator computed through the LU decomposition of $A - \sigma I$ to find $k$ eigenvalues closest to $\sigma$ and their associated eigenvectors.
The PLHR algorithm for Hermitian problems

- A “shorter” trial subspace
- Requires SPD preconditioning
- Can incorporate preconditioning directly into the harmonic RR

![Graphs showing convergence for Si2H4, σ = 0.5 and Si2H4, σ = 0.7](image)

EV and A. Knyazev, to appear in SISC Copper Mountain Special Issue
Conclusions

- GPLHR: a new block preconditioned eigensolver for non-Hermitian problems;

- Robust and efficient under limited memory;

- Solves standard and generalized eigenproblems in a uniform manner;

- Well suited for high-performance parallel computations;

- A pilot version implemented in Q-Chem
Thank you!


## Test problems

<table>
<thead>
<tr>
<th>Problem</th>
<th>Type</th>
<th>$n$</th>
<th>$\sigma$</th>
<th>Preconditioner</th>
<th>Spectr. reg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>A15428</td>
<td>stand.</td>
<td>15428</td>
<td>$-40.871$</td>
<td>ILU($10^{-3}$)</td>
<td>interior</td>
</tr>
<tr>
<td>AF23560</td>
<td>stand.</td>
<td>23560</td>
<td>$-40 + 300i$</td>
<td>ILU($10^{-3}$)</td>
<td>largest mod.</td>
</tr>
<tr>
<td>CRY10000</td>
<td>stand.</td>
<td>10000</td>
<td>8.0</td>
<td>ILU($10^{-3}$)</td>
<td>largest Re</td>
</tr>
<tr>
<td>DW8192</td>
<td>stand.</td>
<td>8192</td>
<td>1.0</td>
<td>ILU($10^{-3}$)</td>
<td>rightmost</td>
</tr>
<tr>
<td>LSTAB_NS</td>
<td>gen.</td>
<td>12619</td>
<td>0</td>
<td>GMRES(25)+LSC</td>
<td>rightmost</td>
</tr>
<tr>
<td>MHD4800</td>
<td>gen.</td>
<td>4800</td>
<td>$-0.1 + 0.5i$</td>
<td>$(A - \sigma B)^{-1}$</td>
<td>interior</td>
</tr>
<tr>
<td>PDE2961</td>
<td>stand.</td>
<td>2961</td>
<td>10.0</td>
<td>ILU($10^{-3}$)</td>
<td>largest Re</td>
</tr>
<tr>
<td>QH882</td>
<td>stand.</td>
<td>882</td>
<td>$-150 + 180i$</td>
<td>GMRES(5)+ILU($10^{-5}$)</td>
<td>interior</td>
</tr>
<tr>
<td>RDB3200L</td>
<td>stand.</td>
<td>3200</td>
<td>2i</td>
<td>ILU($10^{-3}$)</td>
<td>rightmost</td>
</tr>
<tr>
<td>UTM1700</td>
<td>gen.</td>
<td>1700</td>
<td>0</td>
<td>GMRES(15)+ILU($10^{-4}$)</td>
<td>leftmost</td>
</tr>
</tbody>
</table>

**Table:** Test problems.