Krylov subspace methods for eigenvalue problems

David S. Watkins
watkins@math.wsu.edu

Department of Mathematics
Washington State University
Problem: Linear Elasticity
Problem: Linear Elasticity

- Elastic Deformation
  (3D, anisotropic, composite materials)
Problem: Linear Elasticity

- Elastic Deformation
  (3D, anisotropic, composite materials)
- Singularities at cracks, interfaces
Problem: Linear Elasticity

- Elastic Deformation
  (3D, anisotropic, composite materials)
- Singularities at cracks, interfaces
- Lamé Equations (spherical coordinates)
Problem: Linear Elasticity

- Elastic Deformation (3D, anisotropic, composite materials)
- Singularities at cracks, interfaces
- Lamé Equations (spherical coordinates)
- Separate radial variable.
Problem: Linear Elasticity

- Elastic Deformation
  (3D, anisotropic, composite materials)
- Singularities at cracks, interfaces
- Lamé Equations (spherical coordinates)
- Separate radial variable.
- Get quadratic eigenvalue problem.
- \((\lambda^2 M + \lambda G + K)v = 0\)

\[ M^* = M > 0 \quad G^* = -G \quad K^* = K < 0 \]
Linear Elasticity, Continued

Discretize $\theta$ and $\varphi$ variables.
Linear Elasticity, Continued

Discretize $\theta$ and $\varphi$ variables.
(finite element method)
Linear Elasticity, Continued

- Discretize $\theta$ and $\varphi$ variables.
  (finite element method)

- $(\lambda^2 M + \lambda G + K)v = 0$

- $M^T = M > 0 \quad G^T = -G \quad K^T = K < 0$

- matrix quadratic eigenvalue problem
  (large, sparse)
Linear Elasticity, Continued

- Discretize $\theta$ and $\varphi$ variables.
  (finite element method)

- \((\lambda^2 M + \lambda G + K)v = 0\)

\[
M^T = M > 0 \quad G^T = -G \quad K^T = K < 0
\]

- matrix quadratic eigenvalue problem
  (large, sparse)

- Find few smallest eigenvalues (and corresponding eigenvectors).
Linear Elasticity, Continued

- Discretize $\theta$ and $\varphi$ variables.
  (finite element method)

- $$(\lambda^2 M + \lambda G + K)v = 0$$

- $$M^T = M > 0 \quad G^T = -G \quad K^T = K < 0$$

- matrix quadratic eigenvalue problem
  (large, sparse)

- Find few smallest eigenvalues (and corresponding eigenvectors).

- Respect the structure. (symmetric/skew-symmetric)
Hamiltonian Structure
Hamiltonian Structure
Reduction to First Order
Reduction to First Order

\[ \lambda^2 M v + \lambda G v + K v = 0 \]
Reduction to First Order

\[ \lambda^2 M v + \lambda G v + K v = 0 \]

\[ w = \lambda v, \]
Reduction to First Order

\[ \lambda^2 Mv + \lambda Gv + Kv = 0 \]

\[ w = \lambda v, \quad Mw = \lambda Mv \]
Reduction to First Order

\[ \lambda^2 M v + \lambda G v + K v = 0 \]

\[ w = \lambda v, \quad M w = \lambda M v \]

\[
\begin{bmatrix}
-K & 0 \\
0 & -M
\end{bmatrix}
\begin{bmatrix}
v \\
w
\end{bmatrix}
- \lambda
\begin{bmatrix}
G & M \\
-M & 0
\end{bmatrix}
\begin{bmatrix}
v \\
w
\end{bmatrix}
= 0
\]
Reduction to First Order

\[ \lambda^2 M v + \lambda G v + K v = 0 \]

\[ w = \lambda v, \quad M w = \lambda M v \]

\[
\begin{bmatrix}
-K & 0 \\
0 & -M
\end{bmatrix}
\begin{bmatrix}
v \\
w
\end{bmatrix}
- \lambda
\begin{bmatrix}
G & M \\
-M & 0
\end{bmatrix}
\begin{bmatrix}
v \\
w
\end{bmatrix}
= 0
\]

\[ Ax - \lambda B x = 0 \]
Reduction to First Order

\[ \lambda^2 M v + \lambda G v + K v = 0 \]

\[ w = \lambda v, \quad M w = \lambda M v \]

\[
\begin{bmatrix}
-K & 0 \\
0 & -M \\
\end{bmatrix}
\begin{bmatrix}
v \\
w \\
\end{bmatrix}
- \lambda
\begin{bmatrix}
G & M \\
-M & 0 \\
\end{bmatrix}
\begin{bmatrix}
v \\
w \\
\end{bmatrix}
= 0
\]

\[ A x - \lambda B x = 0 \]

symmetric/skew-symmetric
Reduction to Hamiltonian Matrix
Reduction to Hamiltonian Matrix

$A - \lambda B$  (symmetric/skew-symmetric)
Reduction to Hamiltonian Matrix

- $A - \lambda B$ (symmetric/skew-symmetric)

- $B = R^T J R \quad \left( J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \right)$

sometimes easy, always possible
Reduction to Hamiltonian Matrix

1. $A - \lambda B$ (symmetric/skew-symmetric)

2. $B = R^T J R$ ($J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix}$)

sometimes easy, always possible

3. $A - \lambda R^T J R$
Reduction to Hamiltonian Matrix

- $A - \lambda B$ (symmetric/skew-symmetric)

- $B = R^T J R$
  
  
  \[
  J = \begin{bmatrix}
  0 & I \\
  -I & 0
  \end{bmatrix}
  \]

  sometimes easy, always possible

- $A - \lambda R^T J R$

- $R^{-T} A R^{-1} - \lambda J$
Reduction to Hamiltonian Matrix

- $A - \lambda B$ (symmetric/skew-symmetric)

- $B = R^T J R$ 
  \[ J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \]

sometimes easy, always possible

- $A - \lambda R^T J R$
- $R^{-T} A R^{-1} - \lambda J$
- $J^T R^{-T} A R^{-1} - \lambda I$
Reduction to Hamiltonian Matrix

- $A - \lambda B$ (symmetric/skew-symmetric)
- $B = R^T J R \quad \left( J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \right)$

sometimes easy, always possible

- $A - \lambda R^T J R$
- $R^{-T} A R^{-1} - \lambda J$
- $J^T R^{-T} A R^{-1} - \lambda I$
- $H = J^T R^{-T} A R^{-1}$ is Hamiltonian.
in our case ...
in our case ... 

\[ B = \begin{bmatrix} G & M \\ -M & 0 \end{bmatrix} \]
in our case . . .

\[ B = \begin{bmatrix} G & M \\ -M & 0 \end{bmatrix} \]

\[ B = R^T J R = \begin{bmatrix} I & -\frac{1}{2}G \\ 0 & M \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & M \end{bmatrix} \]
in our case ... 

\[
B = \begin{bmatrix}
G & M \\
-M & 0
\end{bmatrix}
\]

\[
B = R^TJR = \begin{bmatrix}
I & -\frac{1}{2}G \\
0 & M
\end{bmatrix} \begin{bmatrix}
0 & I \\
-I & 0
\end{bmatrix} \begin{bmatrix}
I & 0 \\
\frac{1}{2}G & M
\end{bmatrix}
\]

\[
H = \begin{bmatrix}
I & 0 \\
-\frac{1}{2}G & I
\end{bmatrix} \begin{bmatrix}
0 & M^{-1} \\
-K & 0
\end{bmatrix} \begin{bmatrix}
I & 0 \\
-\frac{1}{2}G & I
\end{bmatrix}
\]
in our case . . .

\[
B = \begin{bmatrix}
G & M \\
-M & 0
\end{bmatrix}
\]

\[
B = R^T J R = \begin{bmatrix}
I & -\frac{1}{2} G \\
0 & M
\end{bmatrix}
\begin{bmatrix}
0 & I \\
-I & 0
\end{bmatrix}
\begin{bmatrix}
I & 0 \\
\frac{1}{2} G & M
\end{bmatrix}
\]

\[
H = \begin{bmatrix}
I & 0 \\
-\frac{1}{2} G & I
\end{bmatrix}
\begin{bmatrix}
0 & M^{-1} \\
-K & 0
\end{bmatrix}
\begin{bmatrix}
I & 0 \\
-\frac{1}{2} G & I
\end{bmatrix}
\]

Do not compute \( H \) explicitly.
in our case . . .

- \( B = \begin{bmatrix} G & M \\ -M & 0 \end{bmatrix} \)

- \( B = R^TJR = \begin{bmatrix} I & -\frac{1}{2}G \\ 0 & M \end{bmatrix} \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & M \end{bmatrix} \)

- \( H = \begin{bmatrix} I & 0 \\ -\frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & M^{-1} \\ -K & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ -\frac{1}{2}G & I \end{bmatrix} \)

- Do not compute \( H \) explicitly. (nor \( M^{-1} \))
Working with $H$
Working with $H$

Krylov subspace methods
Working with $H$

- Krylov subspace methods
- just need to apply the operator:
Working with $H$

- Krylov subspace methods
- just need to apply the operator: $x \mapsto Hx$
Working with $H$

- Krylov subspace methods
- just need to apply the operator: $x \mapsto Hx$

\[
H = \begin{bmatrix}
I & 0 \\
-\frac{1}{2}G & I
\end{bmatrix}
\begin{bmatrix}
0 & M^{-1} \\
-K & 0
\end{bmatrix}
\begin{bmatrix}
I & 0 \\
-\frac{1}{2}G & I
\end{bmatrix}
\]
Working with $H$

- Krylov subspace methods
- just need to apply the operator: $x \mapsto Hx$

\[
H = \left[ \begin{array}{cc}
I & 0 \\
-\frac{1}{2}G & I \\
\end{array} \right] \left[ \begin{array}{cc}
0 & M^{-1} \\
-K & 0 \\
\end{array} \right] \left[ \begin{array}{cc}
I & 0 \\
-\frac{1}{2}G & I \\
\end{array} \right]
\]
Working with $H$

- Krylov subspace methods
- just need to apply the operator: $x \mapsto Hx$

$$H = \begin{bmatrix} I & 0 \\ -\frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & M^{-1} \\ -K & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ -\frac{1}{2}G & I \end{bmatrix}$$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$
Working with $H^{-1}$
Working with $H^{-1}$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$
Working with $H^{-1}$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$

$x \mapsto H^{-1}x$
Working with $H^{-1}$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$

- $x \mapsto H^{-1}x$
- Do not compute $(-K)^{-1}$
Working with $H^{-1}$

\[
H^{-1} = \begin{bmatrix}
I & 0 \\
\frac{1}{2}G & I
\end{bmatrix}
\begin{bmatrix}
0 & (-K)^{-1} \\
M & 0
\end{bmatrix}
\begin{bmatrix}
I & 0 \\
\frac{1}{2}G & I
\end{bmatrix}
\]

- $x \mapsto H^{-1}x$
- Do not compute $(-K)^{-1}$
- Cholesky decomposition:
Working with $H^{-1}$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$

- $x \mapsto H^{-1}x$
- Do not compute $(-K)^{-1}$
- Cholesky decomposition: $(-K) = R^T R$
- To compute $w = -K^{-1}v$, 
Working with $H^{-1}$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$

- $x \mapsto H^{-1}x$
- Do not compute $(-K)^{-1}$
- Cholesky decomposition: $(-K) = R^T R$
- To compute $w = -K^{-1}v$, Solve $(-K)w = v$. 
Working with $H^{-1}$

$$H^{-1} = \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix} \begin{bmatrix} 0 & (-K)^{-1} \\ M & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ \frac{1}{2}G & I \end{bmatrix}$$

- $x \mapsto H^{-1}x$
- Do not compute $(-K)^{-1}$
- Cholesky decomposition: $(-K) = R^T R$
- To compute $w = -K^{-1} v$, Solve $(-K)w = v$
- $R^T Rw = v$ Backsolve!
\[ K = \]
\[ K = \]
\[ R = \]
\[ R = \]
$K^{-1} =$
$K^{-1} =$
Second Application
Second Application

- Nonlinear Optics
Second Application

- Nonlinear Optics
- Schrödinger eigenvalue problem
Second Application

- Nonlinear Optics
- Schrödinger eigenvalue problem

\[ -\frac{\hbar^2}{2m} \nabla^2 \psi + V \psi = \lambda \psi \]
Second Application

- Nonlinear Optics
- Schrödinger eigenvalue problem

\[-\frac{\hbar^2}{2m} \nabla^2 \psi + V \psi = \lambda \psi\]

- Solve numerically (finite elements)
Second Application

- Nonlinear Optics
- Schrödinger eigenvalue problem

\[ -\frac{\hbar^2}{2m} \nabla^2 \psi + V \psi = \lambda \psi \]

- Solve numerically (finite elements)

\[ K v = \lambda M v \quad K = K^T > 0, \quad M = M^T > 0 \]
Second Application

- Nonlinear Optics
- Schrödinger eigenvalue problem
- \( -\frac{\hbar^2}{2m} \nabla^2 \psi + V \psi = \lambda \psi \)
- Solve numerically (finite elements)
- \( K \mathbf{v} = \lambda M \mathbf{v} \quad K = K^T > 0, \ M = M^T > 0 \)
- Matrices are large and sparse.
\[ K_u = \lambda M_u \]
$K u = \lambda M v$

Want few \textit{smallest} eigenvalues and associated eigenvectors.
$Ku = \lambda Mu$

Want few smallest eigenvalues and associated eigenvectors.

Invert the problem.
\[ K v = \lambda M v \]

Want few **smallest** eigenvalues and associated eigenvectors.

Invert the problem.

\[ K = R^T R \]
$Kv = \lambda Mv$

Want few smallest eigenvalues and associated eigenvectors.

Invert the problem.

$K = R^T R \quad R^T Rv = \lambda Mv$
$Kv = \lambda Mv$

Want few **smallest** eigenvalues and associated eigenvectors.

Invert the problem.

$K = R^T R \quad R^T Rv = \lambda Mv$

$R^{-T} M R^{-1}(Rv) = \lambda^{-1}(Rv)$
\[ Kv = \lambda M v \]

Want few **smallest** eigenvalues and associated eigenvectors.

Invert the problem.

\[ K = R^T R \quad R^T R v = \lambda M v \]
\[ R^{-T} M R^{-1} (R v) = \lambda^{-1} (R v) \]
\[ A = R^{-T} M R^{-1}, \quad A^T = A > 0 \]
$Kv = \lambda Mv$

Want few smallest eigenvalues and associated eigenvectors.

Invert the problem.

$K = R^T R \quad R^T Rv = \lambda Mv$

$R^{-T} M R^{-1}(Rv) = \lambda^{-1}(Rv)$

$A = R^{-T} M R^{-1}, \quad A^T = A > 0$

$x \mapsto Ax$
\[ K v = \lambda M v \]

Want few \textit{smallest} eigenvalues and associated eigenvectors.

Invert the problem.

\[ K = R^T R \quad R^T R v = \lambda M v \]

\[ R^{-T} M R^{-1}(Rv) = \lambda^{-1}(Rv) \]

\[ A = R^{-T} M R^{-1}, \quad A^T = A > 0 \]

\[ x \mapsto Ax \quad \text{backsolve} \]
\[ K v = \lambda M v \]

- Want few \textit{smallest} eigenvalues and associated eigenvectors.
- Invert the problem.
- \[ K = R^T R \quad R^T R v = \lambda M v \]
- \[ R^{-T} M R^{-1} (R v) = \lambda^{-1} (R v) \]
- \[ A = R^{-T} M R^{-1}, \quad A^T = A > 0 \]
- \[ x \mapsto A x \quad \text{backsolve} \]
- Do not form \( A \) explicitly.
What our applications have in common
What our applications have in common

- large, sparse matrices
What our applications have in common

- large, sparse matrices
- use of matrix factorization (Cholesky decomposition)
What our applications have in common

- large, sparse matrices
- use of matrix factorization (Cholesky decomposition)
- some kind of structure
What our applications have in common

- large, sparse matrices
- use of matrix factorization (Cholesky decomposition)
- some kind of structure
- ...not enough time to discuss this
Classification of Eigenvalue Problems
Classification of Eigenvalue Problems

small
Classification of Eigenvalue Problems

- small
- medium
Classification of Eigenvalue Problems

- small
- medium
- large
Small Matrices
Small Matrices

store conventionally
Small Matrices

- store conventionally
- similarity transformations
Small Matrices

- store conventionally
- similarity transformations
- $QR$ algorithm
Small Matrices

- store conventionally
- similarity transformations
- $QR$ algorithm
- get all eigenvalues/vectors
Small Matrices

- store conventionally
- similarity transformations
- $QR$ algorithm
- get all eigenvalues/vectors
- $n \approx 10^3$
Medium Matrices

store as sparse matrix
Medium Matrices

- store as sparse matrix
- no similarity transformations
Medium Matrices

- store as sparse matrix
- no similarity transformations
- matrix factorization okay
Medium Matrices

- store as sparse matrix
- no similarity transformations
- matrix factorization okay
- shift and invert
Medium Matrices, Continued
Medium Matrices, Continued

- store matrix factor as sparse matrix
Medium Matrices, Continued

- store matrix factor as sparse matrix
- \( n \approx 10^5 \)
Medium Matrices, Continued

- store matrix factor as sparse matrix
- $n \approx 10^5$
- get selected eigenvalues/vectors
Medium Matrices, Continued

- store matrix factor as sparse matrix
- $n \approx 10^5$
- get selected eigenvalues/vectors
- Krylov subspace methods
Medium Matrices, Continued

- store matrix factor as sparse matrix
- $n \approx 10^5$
- get selected eigenvalues/vectors
- Krylov subspace methods
- Jacobi-Davidson methods
Large Matrices
Large Matrices

store as sparse matrix
Large Matrices

- store as sparse matrix
- no similarity transformations
Large Matrices

- store as sparse matrix
- no similarity transformations
- no shift-and-invert
Large Matrices

- store as sparse matrix
- no similarity transformations
- no shift-and-invert
- $n \approx 10^7$
Large Matrices

- store as sparse matrix
- no similarity transformations
- no shift-and-invert
- $n \approx 10^7$
- get selected eigenvalues/vectors
Large Matrices

- store as sparse matrix
- no similarity transformations
- no shift-and-invert
- \( n \approx 10^7 \)
- get selected eigenvalues/vectors
- Krylov subspace methods
Large Matrices

- store as sparse matrix
- no similarity transformations
- no shift-and-invert
- $n \approx 10^7$
- get selected eigenvalues/vectors
- Krylov subspace methods
- Jacobi-Davidson methods
Krylov Subspace Methods
Krylov Subspace Methods

\[ x \mapsto Ax \]
Krylov Subspace Methods

$x \rightarrow Ax$

Example: $A = R^{-T}MR^{-1}$
Krylov Subspace Methods

\[ x \mapsto Ax \]

Example: \[ A = R^{-T} M R^{-1} \]

Pick a vector \( v \)
Krylov Subspace Methods

- $x \rightarrow Ax$
- Example: $A = R^{-T}MR^{-1}$
- Pick a vector $v$
- $v, Av, A^2v, A^3v, \ldots$
Krylov Subspace Methods

- $x \rightarrow Ax$

- Example: $A = R^{-T} M R^{-1}$

- Pick a vector $v$

- $v, Av, A^2v, A^3v, \ldots$

- Krylov subspace:

$$\mathcal{K}_j(A, v) = \text{span}\{v, Av, A^2v, \ldots, A^{j-1}v\}$$
Krylov Subspace Methods

- $x \mapsto Ax$

Example: $A = R^{-T} M R^{-1}$

- Pick a vector $v$

- $v, Av, A^2v, A^3v, \ldots$

Krylov subspace:

$$\mathcal{K}_j(A, v) = \text{span}\{v, Av, A^2v, \ldots, A^{j-1}v\}$$

- Look in here for approximate eigenvectors.
Krylov Subspace Methods

- \( x \mapsto Ax \)
- Example: \( A = R^{-T} M R^{-1} \)
- Pick a vector \( v \)
- \( v, Av, A^2v, A^3v, \ldots \)
- Krylov subspace:
  \[
  \mathcal{K}_j(A, v) = \text{span}\{v, Av, A^2v, \ldots, A^{j-1}v\}
  \]
- Look in here for approximate eigenvectors.
- …but need better basis
Arnoldi Process
Arnoldi Process

build orthonormal basis $v_1, v_2, v_3, \ldots$
Arnoldi Process

- build orthonormal basis $v_1, v_2, v_3, \ldots$

- $j$th step: $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} v_i h_{ij}$
Arnoldi Process

- build orthonormal basis $v_1, v_2, v_3, \ldots$
- $j$th step: $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} v_i h_{ij}$
- $h_{ij} = \langle Av_j, v_i \rangle$ (Gram-Schmidt)
Arnoldi Process

- build orthonormal basis $v_1, v_2, v_3, \ldots$
- $j$th step: $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} v_i h_{ij}$
- $h_{ij} = \langle Av_j, v_i \rangle$ (Gram-Schmidt)
- Normalization: $h_{j+1,j} = \|\hat{v}_{j+1}\|$, $v_{j+1} = \hat{v}_{j+1}/h_{j+1,j}$
Arnoldi Process

- build orthonormal basis $v_1, v_2, v_3, \ldots$

- $j$th step: $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} v_i h_{ij}$

- $h_{ij} = \langle Av_j, v_i \rangle$ (Gram-Schmidt)

- Normalization: $h_{j+1,j} = ||\hat{v}_{j+1}||$, $v_{j+1} = \hat{v}_{j+1} / h_{j+1,j}$

- Collect coefficients $h_{ij}$
Arnoldi Process

- build orthonormal basis $v_1, v_2, v_3, \ldots$

- $j$th step: $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} v_i h_{ij}$

- $h_{ij} = \langle Av_j, v_i \rangle$ (Gram-Schmidt)

- Normalization: $h_{j+1,j} = \|\hat{v}_{j+1}\|$, $v_{j+1} = \hat{v}_{j+1}/h_{j+1,j}$

- Collect coefficients $h_{ij}$

- $H_4 = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{32} & h_{33} & h_{34} \\ h_{43} & h_{44} \end{bmatrix}$
Arnoldi Process

- build orthonormal basis $v_1, v_2, v_3, \ldots$
- $j$th step:  $\hat{v}_{j+1} = Av_j - \sum_{i=1}^{j} v_i h_{ij}$
- $h_{ij} = \langle Av_j, v_i \rangle$ (Gram-Schmidt)
- Normalization:  $h_{j+1,j} = \|\hat{v}_{j+1}\|$,  $v_{j+1} = \hat{v}_{j+1}/h_{j+1,j}$
- Collect coefficients $h_{ij}$

$$H_4 = \begin{bmatrix}
  h_{11} & h_{12} & h_{13} & h_{14} \\
  h_{21} & h_{22} & h_{23} & h_{24} \\
  h_{32} & h_{33} & h_{34} \\
  h_{43} & h_{44}
\end{bmatrix}$$

- eigenvalues are Ritz values
Example: 479 × 479 matrix
Example: $479 \times 479$ matrix
Example: $479 \times 479$ matrix
Example: \(479 \times 479\) matrix
For better accuracy ...
For better accuracy . . .

. . . take more steps.
For better accuracy . . .

. . . take more steps.

but,
For better accuracy …
… take more steps.
but, vectors take up space.
For better accuracy …

…take more steps.

but, vectors take up space.

Alternate plan:
For better accuracy . . .

. . . take more steps.

but, vectors take up space.

Alternate plan:

Start over with a better vector.
Implicit Restarts
Implicit Restarts

Take, say, 30 steps.
Implicit Restarts

- Take, say, 30 steps.
- Get 30 Ritz values.
Implicit Restarts

- Take, say, 30 steps.
- Get 30 Ritz values.
- Keep the best ones (e.g. 10) ...
Implicit Restarts

- Take, say, 30 steps.
- Get 30 Ritz values.
- Keep the best ones (e.g. 10) . . .
- . . . and associated invariant subspace.
Implicit Restarts

- Take, say, 30 steps.
- Get 30 Ritz values.
- Keep the best ones (e.g. 10) . . .
- . . . and associated invariant subspace.
- Restart at step 11.
Implicit Restarts

- Take, say, 30 steps.
- Get 30 Ritz values.
- Keep the best ones (e.g. 10) . . .
- . . . and associated invariant subspace.
- Restart at step 11.
- (neat details omitted)
Implicit Restarts

- Take, say, 30 steps.
- Get 30 Ritz values.
- Keep the best ones (e.g. 10) . . .
- . . . and associated invariant subspace.
- Restart at step 11.
- (neat details omitted)
- Build back up to 30 and then restart again.
Example
Example

460 × 460 Hamiltonian matrix (toy problem)
Example

- $460 \times 460$ Hamiltonian matrix (toy problem)
- asking for 24 eigenpairs
Example

- $460 \times 460$ Hamiltonian matrix (toy problem)
- asking for 24 eigenpairs
- building to dimension 84
Example

- 460 × 460 Hamiltonian matrix (toy problem)
- asking for 24 eigenpairs
- building to dimension 84
- restarting with 28
Example

- $460 \times 460$ Hamiltonian matrix (toy problem)
- asking for 24 eigenpairs
- building to dimension 84
- restarting with 28
- Hamiltonian Lanczos process
Concluding Remarks
Concluding Remarks

I used these methods to solve my “medium” sized eigenvalue problems.
Concluding Remarks

- I used these methods to solve my “medium” sized eigenvalue problems.
- Simple ideas lead to powerful methods.
Concluding Remarks

- I used these methods to solve my “medium” sized eigenvalue problems.
- Simple ideas lead to powerful methods.
- Thank you for your attention.