On the numerical solution of large scale algebraic linear systems

V. Simoncini

Dipartimento di Matematica, Università di Bologna

valeria.simoncini@unibo.it

Outline

- Algebraic linear systems - the problem
- Sparse matrices and sparse formats
- Symmetric vs nonsymmetric matrices
- State-of-the-art solvers. First steps

Lectures: see `https://www.dm.unibo.it/~simoncin/corso.html`

The Problem

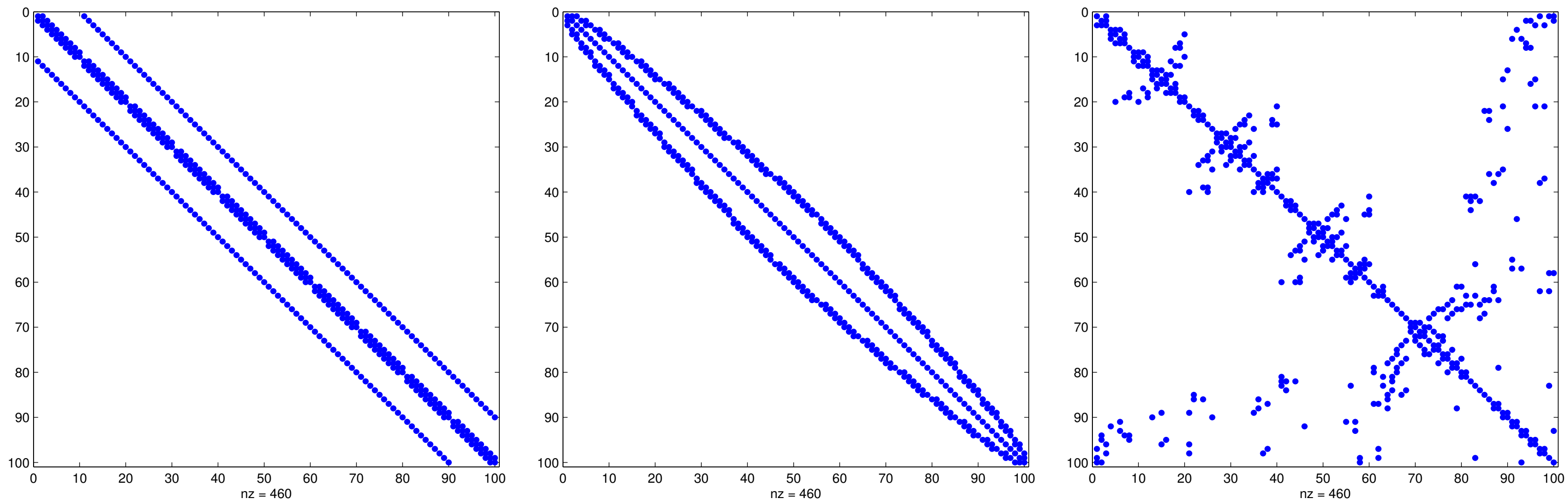
$$Ax = b \qquad \text{or} \quad AX = B, \quad B = [b_1, \ldots, b_s]$$

$A \in \mathbb{C}^{n\times n}$, B full column rank, $s \ll n$

- A large and sparse
- A large and structured: blocks, banded, ...
- A functional: $A = CS^{-1}D$, preconditioned, integral, ...
- ....

Sparse matrices. I

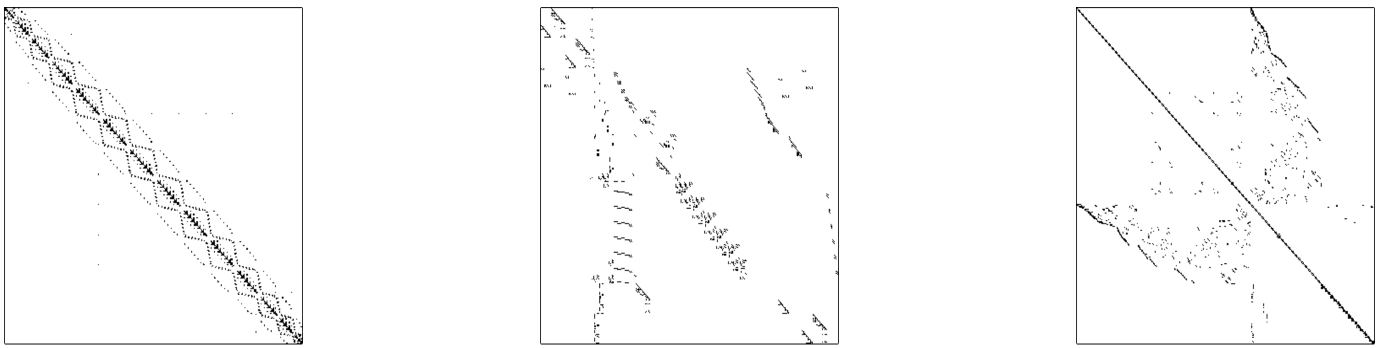
Matrices stemming from discretizations have special pattern:



Same matrix, different ordering of the unknowns

large dimensions, only low percentage of nonzero elements per row

Sparse matrices. Different applications



Nuclear reactor model / Chemical eng. plant model/ Hydroelectric Power System

Sparse matrices. II

Memory allocation of generic sparse matrices:

- Coordinate format
- Compressed sparse row format
- Compressed sparse column format
- ...

Sparse matrices. III

- Coordinate format (COO)

  `a(nnz), ia(nnz), ja(nnz)`, for $A(i,j)$, `nnz` # nonzeros

  simple, flexible. Often used to store on disk.

- Compressed sparse row format (CSR)

  `a(nnz), ia(n+1), ja(nnz)`, for $A(i,j)$, `n` matrix dimension

  (`ia(n+1)` contains the pointer to the first element of next row)

  very effective for matrix-vector multiplies

- Compressed sparse column format (CSC)

  Same as CSR but for the columns

- ...

Sparse matrices. II

$$y = Ax$$

Typical matrix-vector operation in *Compressed sparse row format*:

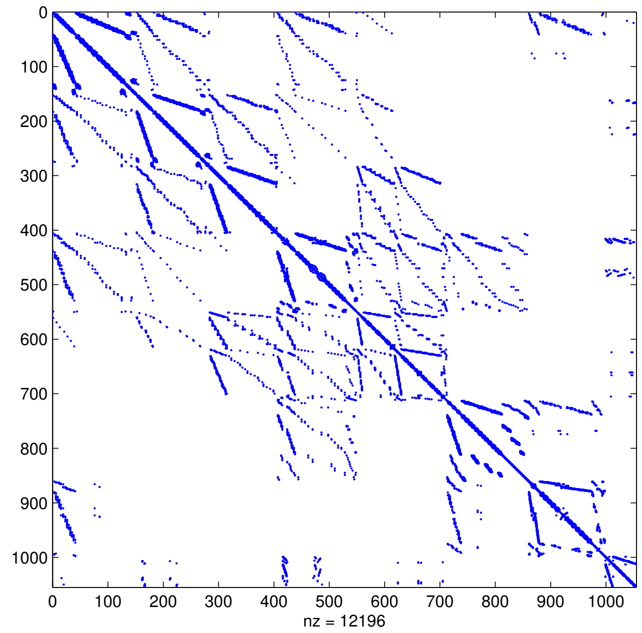
`a(nnz)`, `ia(n+1)`, `ja(nnz)`, for $A(i,j)$, `n` matrix dimension

```
      do 100 i = 1,n
c
c     compute the inner product of row i with vector x
c
         t = 0.0d0
         do 99 k=ia(i), ia(i+1)-1
            t = t + a(k)*x(ja(k))
 99      continue
c
c     store result in y(i)
c
         y(i) = t
 100  continue
```

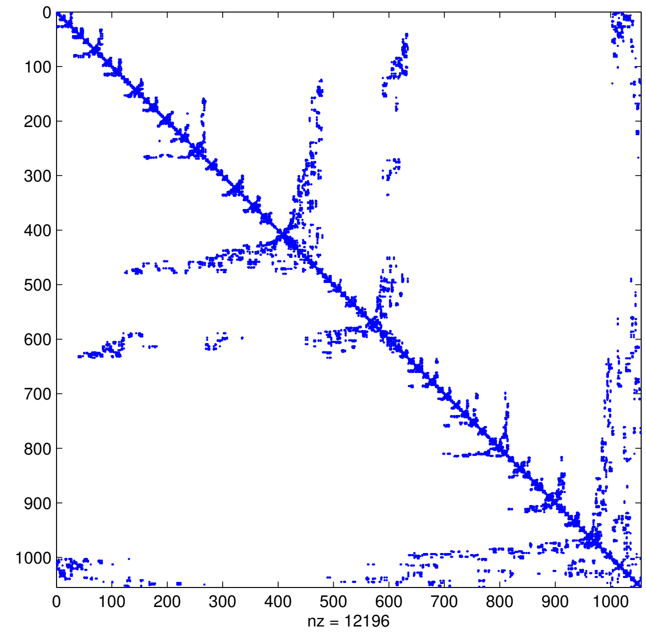
Sparse matrices. Reordering of the entries

Matrix market. matrix CAN_1072 (structure problem in aircraft design)

Original sparsity pattern



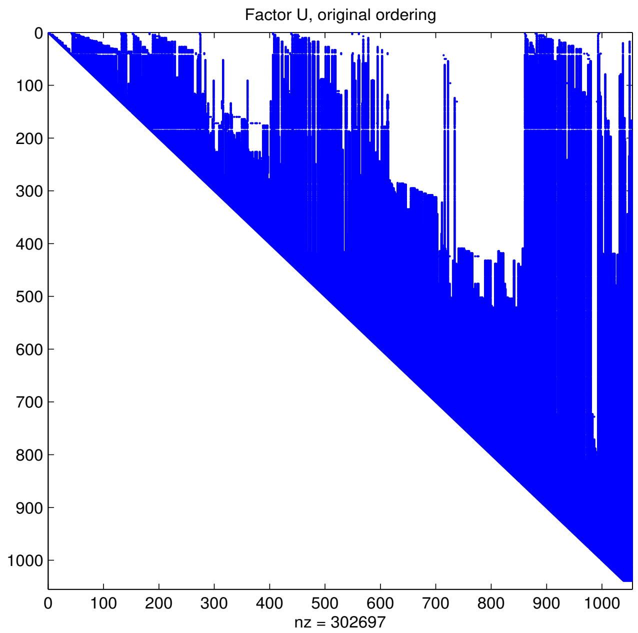
symamd reordering



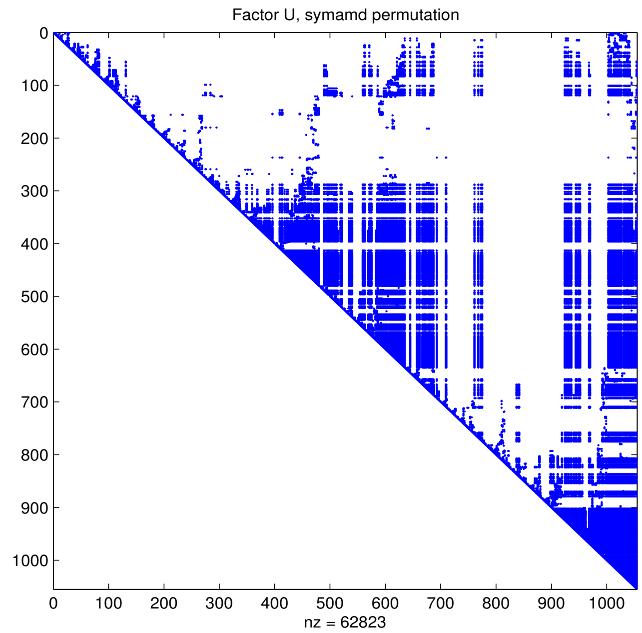
Sparse matrices. An Example

Factor U in LU factorization $A = LU$:

A with original sparsity pattern



A with `symamd` reordering



Solution methods for large matrices

Discretization of 2D and 3D problems leads to large matrices A

(size $O(10^k)$, $k = 5 - 8$)

$\Rightarrow$ (Optimized) LU decomposition too expensive

Alternatives do not rely on explicit factorizations !

- Iterative methods: Projection-type methods (*)
- Geometric multigrid methods
- Algebraic multigrid methods
- Problem-related optimized methods

Projection/Reduction methods for large scale linear systems

Outline

- Projection and polynomial -type methods
- **Coefficient matrix role in tailoring the solution strategy**
  - Real symmetric or complex Hermitian
  - Complex symmetric and H-symmetric
  - Complex/Real non-Hermitian
- Stopping criteria and inexactness

The Problem

$$Ax = b \qquad \text{or} \quad AX = B, \quad B = [b_1, \ldots, b_s]$$

$A \in \mathbb{C}^{n \times n}$, B full column rank, $s \ll n$

- A large and sparse
- A large and structured: blocks, banded, ...
- A functional: $A = CS^{-1}D$, preconditioned, integral, ...
- ....

The Problem

$$Ax = b \qquad \text{or} \quad AX = B, \quad B = [b_1, \ldots, b_s]$$

$A \in \mathbb{C}^{n\times n}$, B full column rank, $s \ll n$

- A large and sparse
- A large and structured: blocks, banded, ...
- A functional: $A = CS^{-1}D$, preconditioned, integral, ...
- ....

The solution approach. Generate sequence of approximate solutions:

$$\{x_0, x_1, x_2, \ldots\}, \qquad x_k \to_{k\to\infty} x$$

Occurrence of the problem

Very broad range of applications in Engineering and Scientific Computing

Original application context:

- Discretization of 2D and 3D PDEs
  (linear steady state, nonlinear, evolutive, etc.)
- Eigenvalue problems
- Approximation of matrix functions
- Workhorses of more advanced techniques
- ...

Relevant Bibliographic Pointers

YOUSEF SAAD

Iterative methods for sparse linear systems

SIAM, Society for Industrial and Applied Mathematics, 2003, 2nd edition.

VALERIA SIMONCINI AND DANIEL B. SZYLD

Recent developments in Krylov Subspace Methods for linear systems

Numerical Linear Algebra with Appl., v. 14, n.1 (2007), pp.1-59.

"Projection" methods (or, reduction methods)

- Approximation vector space K_m. At each iteration m

$$\{\mathbf{x}_m\} \text{ such that } \mathbf{x}_m \in K_m$$

K_m: dimension[a] m, with the "expansion" property:

$$K_m \subseteq K_{m+1}$$

- Computation of iterate. Galerkin condition:

$$\text{residual} \quad \mathbf{r}_m := \mathbf{b} - \mathbf{A}\mathbf{x}_m \quad \perp \quad K_m$$

$\Rightarrow$ This condition uniquely defines $\mathbf{x}_m \in K_m$

[a] At most

A well established code

Classical Conjugate Gradient:

Given x_0. Set $r_0 = b - Ax_0$, $p_0 = r_0$

for $i = 0, 1, \ldots$

$$\alpha_i = \frac{r_i^* r_i}{p_i^* A p_i}$$

$$x_{i+1} = x_i + p_i \alpha_i$$

$$r_{i+1} = r_i - A p_i \alpha_i$$

$$\beta_{i+1} = \frac{r_{i+1}^* A p_i}{p_i^* A p_i}$$

$$p_{i+1} = r_i + p_i \beta_{i+1}$$

end

* At each iteration: 1 `Mxv`, 3 `-axpys`, 2 `-dots`

* Short-term recurrence $\Rightarrow$: computational cost is constant at each iteration

* Implicit space generation, no explicit computation of the orthonormal basis!

The **Block** Conjugate Gradient

$R_0 = B - AX_0,\ P_0 = R_0 \in \mathbb{C}^{n\times s}$

for $i = 0, 1, \ldots$

$\quad \boldsymbol{\alpha}_i = (P_i^* A P_i)^{-1}(R_i^* R_i) \in \mathbb{C}^{s\times s}$

$\quad X_{i+1} = X_i + P_i \boldsymbol{\alpha}_i$

$\quad R_{i+1} = R_i - A P_i \boldsymbol{\alpha}_i$

$\quad \boldsymbol{\beta}_{i+1} = (P_i^* A P_i)^{-1}(R_{i+1}^* A P_i) \in \mathbb{C}^{s\times s}$

$\quad P_{i+1} = R_i + P_i \boldsymbol{\beta}_{i+1}$

end

Optimality property of Galerkin projection method

A symmetric and positive definite. Let $\mathbf{x}^\star$ be the true solution.
Galerkin property: Impose that

$$\text{residual} \quad \mathbf{r}_m := \mathbf{b} - \mathbf{A}\mathbf{x}_m \quad \perp \quad K_m$$

is equivalent to: Find

$$\mathbf{x}_m \quad \text{solution to} \quad \min_{\mathbf{x}\in K_m} \|\mathbf{x}^\star - \mathbf{x}\|_{\mathbf{A}}$$

where $\|\cdot\|_{\mathbf{A}}$ is the energy norm, namely $\|\mathbf{x}\|_{\mathbf{A}}^2 := \langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle$

Convergence and spectral properties

- In exact arithmetic (i.e., in theory), finite termination property
- A-priori bound for energy norm of the error:

  If $K_m = \mathrm{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \ldots, \mathbf{A}^{m-1}\mathbf{b}\}$, then

$$\|\mathbf{x}^\star - \mathbf{x}_m\|_{\mathbf{A}} \le 2 \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^m \|\mathbf{x}^\star - \mathbf{x}_0\|_{\mathbf{A}}$$

  where $\kappa = \frac{\lambda_{\max}(\mathbf{A})}{\lambda_{\min}(\mathbf{A})}$

(Conjugate Gradients, Hestenes & Stiefel, '52)

Convergence and spectral properties

- In exact arithmetic (i.e., in theory), finite termination property
- A-priori bound for energy norm of the error:

  If $K_m = \mathrm{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \ldots, \mathbf{A}^{m-1}\mathbf{b}\}$, then

$$\|\mathbf{x}^\star - \mathbf{x}_m\|_\mathbf{A} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^m \|\mathbf{x}^\star - \mathbf{x}_0\|_\mathbf{A}$$

  where $\kappa = \frac{\lambda_{\max}(\mathbf{A})}{\lambda_{\min}(\mathbf{A})}$

(Conjugate Gradients, Hestenes & Stiefel, '52)

Consequences:

- Convergence: The closer κ to 1 the faster
- Convergence depends on spectral properties, not directly on problem size!

PDE discretization and linear system solves

$$-\Delta u = f, \quad u|_{\partial\Omega} = u_0.$$

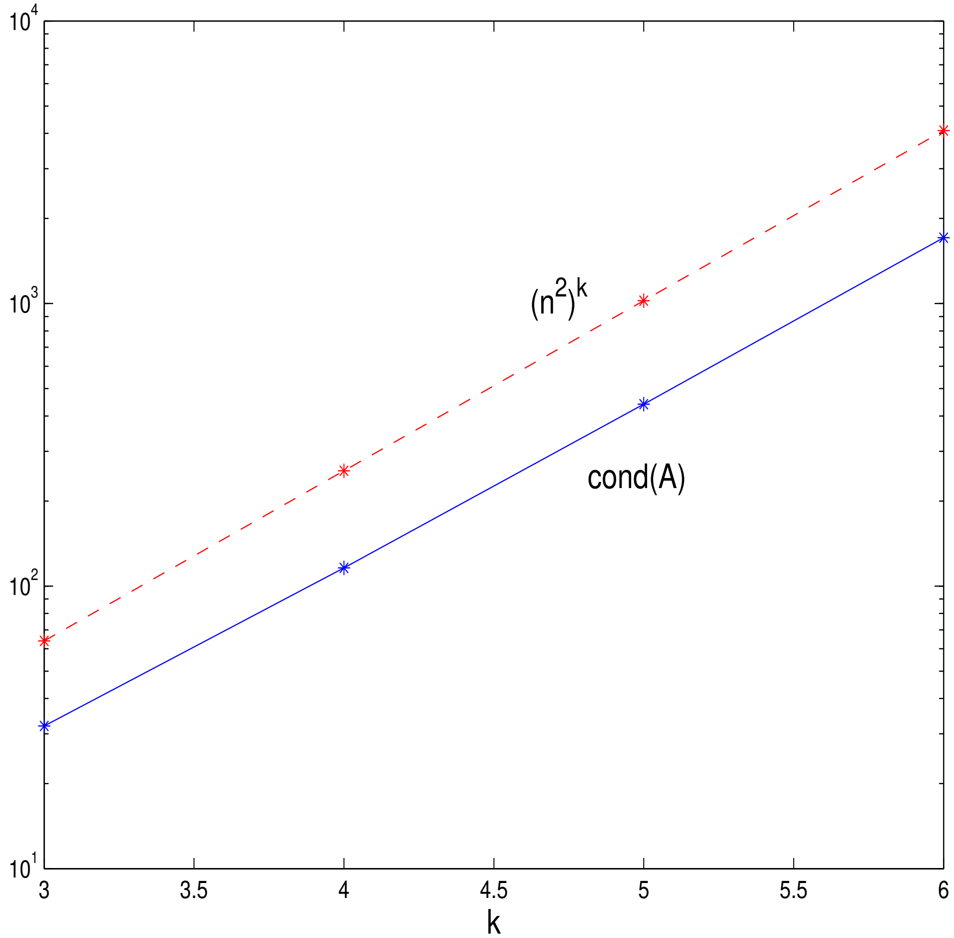
A 2D Poisson operator $\Rightarrow$ A symmetric and positive definite

CG: Number of iterations k depends on cond(A) $:= \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)}$

number of nodes n per dimension	cond(A)	# its tol $= 10^{-10}$
2^3	32.16	10
2^4	116.46	31
2^5	440.69	66
2^6	1711.17	132

Stopping criterion: $r_k := b - Ax_k$ small enough in some norm

Discretization and linear system solves



For fine discretizations, convergence is slow !

A more general picture. Nonsymmetric problems

- A normal, $AA^* = A^*A$
- A (highly) non-normal, $\|AA^* - A^*A\| \gg 0$
- A "Hermitian" in disguise:

A more general picture. Nonsymmetric problems

- A normal, $AA^* = A^*A$
- A (highly) non-normal, $\|AA^* - A^*A\| \gg 0$
- A "Hermitian" in disguise:
  - ⋆ $A = M + \sigma I,\ \sigma \in \mathbb{C}$, $M \in \mathbb{R}^{n\times n}$ symmetric

A more general picture. Nonsymmetric problems

- A normal, $AA^* = A^*A$
- A (highly) non-normal, $\|AA^* - A^*A\| \gg 0$
- A "Hermitian" in disguise:
  - $\star$ $A = M + \sigma I,\ \sigma \in \mathbb{C}$, $M \in \mathbb{R}^{n\times n}$ symmetric
  - $\star$ $A = M + \sigma H,\ \sigma \in \mathbb{C}$, $M, H \in \mathbb{R}^{n\times n}$ symmetric

A more general picture. Nonsymmetric problems

- A normal, $AA^* = A^*A$
- A (highly) non-normal, $\|AA^* - A^*A\| \gg 0$
- A "Hermitian" in disguise:
  - ⋆ $A = M + \sigma I, \; \sigma \in \mathbb{C}, \; M \in \mathbb{R}^{n\times n}$ symmetric
  - ⋆ $A = M + \sigma H, \; \sigma \in \mathbb{C}, \; M, H \in \mathbb{R}^{n\times n}$ symmetric
  - ⋆ There exists nonsing. Herm. $H \in \mathbb{C}^{n\times n}$ such that $HA = A^*H$, e.g. M, C Hermitian

$$A = \begin{bmatrix} M & B \\ -B^* & C \end{bmatrix}, \quad H = \begin{bmatrix} I & \\ & -I \end{bmatrix},$$

A more general picture. Nonsymmetric problems

- A normal, $AA^* = A^*A$
- A (highly) non-normal, $\|AA^* - A^*A\| \gg 0$
- A "Hermitian" in disguise:
  - ⋆ $A = M + \sigma I,\ \sigma \in \mathbb{C},\ M \in \mathbb{R}^{n\times n}$ symmetric
  - ⋆ $A = M + \sigma H,\ \sigma \in \mathbb{C},\ M, H \in \mathbb{R}^{n\times n}$ symmetric
  - ⋆ There exists nonsing. Herm. $H \in \mathbb{C}^{n\times n}$ such that $HA = A^*H$, e.g. M, C Hermitian

$$A = \begin{bmatrix} M & B \\ -B^* & C \end{bmatrix}, \quad H = \begin{bmatrix} I & \\ & -I \end{bmatrix},$$

  - ⋆ $Ax = b \quad \Leftrightarrow \quad A^*Ax = A^*b$ (not recommended in general...)

Outline

- What is the added difficulty with A non-Hermitian ?
- How to handle "Symmetry in disguise"
- Non-normal (non-Hermitian) case
  - ⋆ Long-term recurrences and their problems
  - ⋆ Coping with them ⇒ Restarted, truncated, flexible
  - ⋆ Making it without ⇒ short-term recurrences
- Tricks for all trades

What goes "wrong" with A non-Hermitian. I

$$\{x_k\}, \quad \text{with} \quad x_k \in x_0 + K_k(A, r_0) = \text{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}$$

Let $V_k = [v_1, \ldots, v_k]$ be a (orthogonal) basis of $K_k(A, r_0)$. Then

$$x_k = x_0 + V_k y_k, \qquad y_k \in \mathbb{C}^k$$

A condition is required to specify y_k.

What goes "wrong" with A non-Hermitian. I

$$\{x_k\}, \quad \text{with} \quad x_k \in x_0 + K_k(A, r_0) = \text{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}$$

Let $V_k = [v_1, \ldots, v_k]$ be a (orthogonal) basis of $K_k(A, r_0)$. Then

$$x_k = x_0 + V_k y_k, \qquad y_k \in \mathbb{C}^k$$

A condition is required to specify y_k. For instance:

$$r_k := b - Ax_k = r_0 - AV_k y_k \quad \perp \quad K_k(A, r_0) \qquad V_k^* r_k = 0$$

(Galerkin condition, again!) so that

$$0 = V_k^* r_k = V_k^* r_0 - V_k^* AV_k y_k \quad \Leftrightarrow \quad y_k \ s.t. \ (V_k^* AV_k) y_k = V_k^* r_0$$

What goes "wrong" with A non-Hermitian. I

$$\{x_k\}, \quad \text{with} \quad x_k \in x_0 + K_k(A, r_0) = \text{span}\{r_0, Ar_0, \dots, A^{k-1} r_0\}$$

Let $V_k = [v_1, \dots, v_k]$ be a (orthogonal) basis of $K_k(A, r_0)$. Then

$$x_k = x_0 + V_k y_k, \qquad y_k \in \mathbb{C}^k$$

A condition is required to specify y_k. For instance:

$$r_k := b - Ax_k = r_0 - AV_k y_k \quad \perp \quad K_k(A, r_0) \qquad V_k^* r_k = 0$$

(Galerkin condition, again!) so that

$$0 = V_k^* r_k = V_k^* r_0 - V_k^* A V_k y_k \quad \Leftrightarrow \quad y_k \ s.t. \ (V_k^* A V_k) y_k = V_k^* r_0$$

Hence

$$x_k = x_0 + V_k (V_k^* A V_k)^{-1} V_k^* r_0 \qquad \text{with} \quad V_k^* r_0 = e_1 \|r_0\|$$

And: $V_k^* A V_k$ upper Hessenberg (Gram-Schmidt procedure to build V_k)

What goes "wrong" with A non-Hermitian. II

If A were Hpd $\quad\Rightarrow\quad V_k^* A V_k$ also Hpd $\quad\Rightarrow\quad$ tridiagonal

$$V_k^* A V_k = L_k L_k^* \qquad L_k \text{ bidiagonal}$$

$$\begin{aligned} x_k &= x_0 + V_k L_k^{-*} L_k^{-1} e_1 \|r_0\| \\ &= x_0 + V_{k-1} L_{k-1}^{-*} L_{k-1}^{-1} e_1 \|r_0\| + p_k \alpha_k \\ &= x_{k-1} + p_k \alpha_k \end{aligned}$$

with $p_k \in \text{span}\{v_{k-1}, v_k\}$

(development underlying Conjugate Gradient)

What goes "wrong" with A non-Hermitian. II

If A were Hpd $\Rightarrow$ $V_k^* A V_k$ also Hpd $\Rightarrow$ tridiagonal

$$V_k^* A V_k = L_k L_k^* \qquad L_k \text{ bidiagonal}$$

$$\begin{aligned} x_k &= x_0 + V_k L_k^{-*} L_k^{-1} e_1 \|r_0\| \\ &= x_0 + V_{k-1} L_{k-1}^{-*} L_{k-1}^{-1} e_1 \|r_0\| + p_k \alpha_k \\ &= x_{k-1} + p_k \alpha_k \end{aligned}$$

with $p_k \in \text{span}\{v_{k-1}, v_k\}$

(development underlying Conjugate Gradient)

A non-Hermitian $\Rightarrow$ $V_k^* A V_k$ only upper Hessenberg

$$p_k \in \text{span}\{v_1, \ldots, v_k\}$$

What goes "wrong" with A non-Hermitian. III

$p_k \in \text{span}\{v_1, \ldots, v_k\}$, with $\{v_1, \ldots, v_k\}$ orthogonal basis

Alternatives

- Give up orthogonal basis, $V_k^* V_k = I_k$
- Give up optimality condition, e.g. $r_k \perp K_k(A, r_0)$
- Resume symmetry

Symmetry in disguise. Complex symmetric shifted systems. 1.

Case 1: $A = M + \sigma I, \quad M \in \mathbb{R}^{n\times n}, \sigma \in \mathbb{C}$

E.g.: Helmholtz equation (wave problems such as vibrating strings and membranes)

Trick: replace $*$ (conj. transp.) with $\top$ (transp.)

$$A = A^\top \qquad \text{complex symmetric}$$

Apply CG with $\top$

Given x_0. Set $r_0 = b - Ax_0$, $p_0 = r_0$

for $i = 0, 1, \ldots$

$$\alpha_i = \frac{r_i^\top r_i}{p_i^\top A p_i}$$

$$x_{i+1} = x_i + p_i \alpha_i$$

$$r_{i+1} = r_i - A p_i \alpha_i$$

$$\beta_{i+1} = \frac{r_{i+1}^\top A p_i}{p_i^\top A p_i}$$

$$p_{i+1} = r_i + p_i \beta_{i+1}$$

end

Symmetry in disguise. Complex symmetric shifted systems. 2.

$A = M + \sigma I$: Apply CG with $\top$

Properties:

- V_k real: $K_k(A, r_0) = K_k(A + \sigma I, r_0)$
- $\top$ does not define an inner product!
- $V_k^\top A V_k = V_k^\top M V_k + \sigma I$

  If $\Im(\sigma) \neq 0$ then $V_k^\top A V_k$ is nonsingular $\Rightarrow$ No breakdown

The same code applies in case of any A complex symmetric ($A = A^\top$)

H-symmetry

A is H-Hermitian if there exists $H \in \mathbb{C}^{n \times n}$ Hermitian, nonsingular s.t.

$$HA = A^* H$$

(H-symmetric if $HA = A^\top H$ with H is symmetric)

H-symmetry

A is H-Hermitian if there exists $H \in \mathbb{C}^{n\times n}$ Hermitian, nonsingular s.t.

$$HA = A^* H$$

(H-symmetric if $HA = A^\top H$ with H is symmetric)

If H is Hpd (and HA is also Hpd), use CG in the H-inner product:

Given x_0. Set $r_0 = b - Ax_0$, $p_0 = r_0$

for $i = 0, 1, \ldots$

$$\alpha_i = \frac{r_i^* H r_i}{p_i^* H A p_i}$$

$$x_{i+1} = x_i + p_i \alpha_i$$

$$r_{i+1} = r_i - A p_i \alpha_i$$

$$\beta_{i+1} = \frac{r_{i+1}^* H A p_i}{p_i^* H A p_i}$$

$$p_{i+1} = r_i + p_i \beta_{i+1}$$

end

(H not Hpd $\Rightarrow$ see later)

First Summary

Symmetry in disguise:

- Shifted matrices, $A = M + \sigma I$, M real symmetric
- Complex symmetric matrices
- H-symmetric or H-Hermitian matrices

Long-term recurrences

$$K_k(A, r_0) = \text{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}, \quad V_k \text{ orth. basis}$$

1. Arnoldi process : $v_{k+1} \leftarrow Av_k - \sum_{j=1}^{k} v_j h_{j,k}$, that is

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^* = V_{k+1} \underline{H}_k \qquad (H_k = V_k^* A V_k)$$

2. $x_k = x_0 + V_k y_k$

Long-term recurrences

$$K_k(A, r_0) = \text{span}\{r_0, Ar_0, \ldots, A^{k-1}r_0\}, \quad V_k \text{ orth. basis}$$

1. Arnoldi process : $v_{k+1} \leftarrow Av_k - \sum_{j=1}^{k} v_j h_{j,k}$, that is

$$AV_k = V_k H_k + h_{k+1,k} v_{k+1} e_k^* = V_{k+1} \underline{H}_k \qquad (H_k = V_k^* A V_k)$$

2. $x_k = x_0 + V_k y_k$

- GMRES. Particular Petrov-Galerkin condition:

$$r_k \perp AK_k \Rightarrow \quad y_k \text{ s.t. } \min_y \|r_0 - AV_k y\|$$

- FOM. Galerkin condition: (H_k nonsingular)

$$r_k \perp K_k \Rightarrow \quad y_k \text{ s.t. } H_k y = e_1 \|r_0\|$$

GMRES

$$AV_k = V_{k+1}\underline{H}_k, \qquad r_0 = V_{k+1}e_1\beta_0$$

Crucial property:

$$\begin{aligned}
\min_y \|r_0 - AV_k y\| &= \\
&= \min_y \|V_{k+1}(e_1\beta_0 - \underline{H}_k y)\| \\
&= \min_y \|e_1\beta_0 - \underline{H}_k y\|
\end{aligned}$$

Least squares problem expands at each iteration.

QR decomposition of $\underline{H}_k$ only updated, not recomputed from scratch.

Block GMRES

$$R_0 = B - AX_0, \qquad K_k(A, R_0) = \text{span}\{R_0, AR_0, \ldots, A^{k-1}R_0\},$$

$\mathcal{U}_k$ orth. basis, $\mathcal{U}_k = [U_1, U_2, \ldots, U_k] \in \mathbb{C}^{n\times ks}$

Block Arnoldi process (s MxV + Gram-Schmidt)

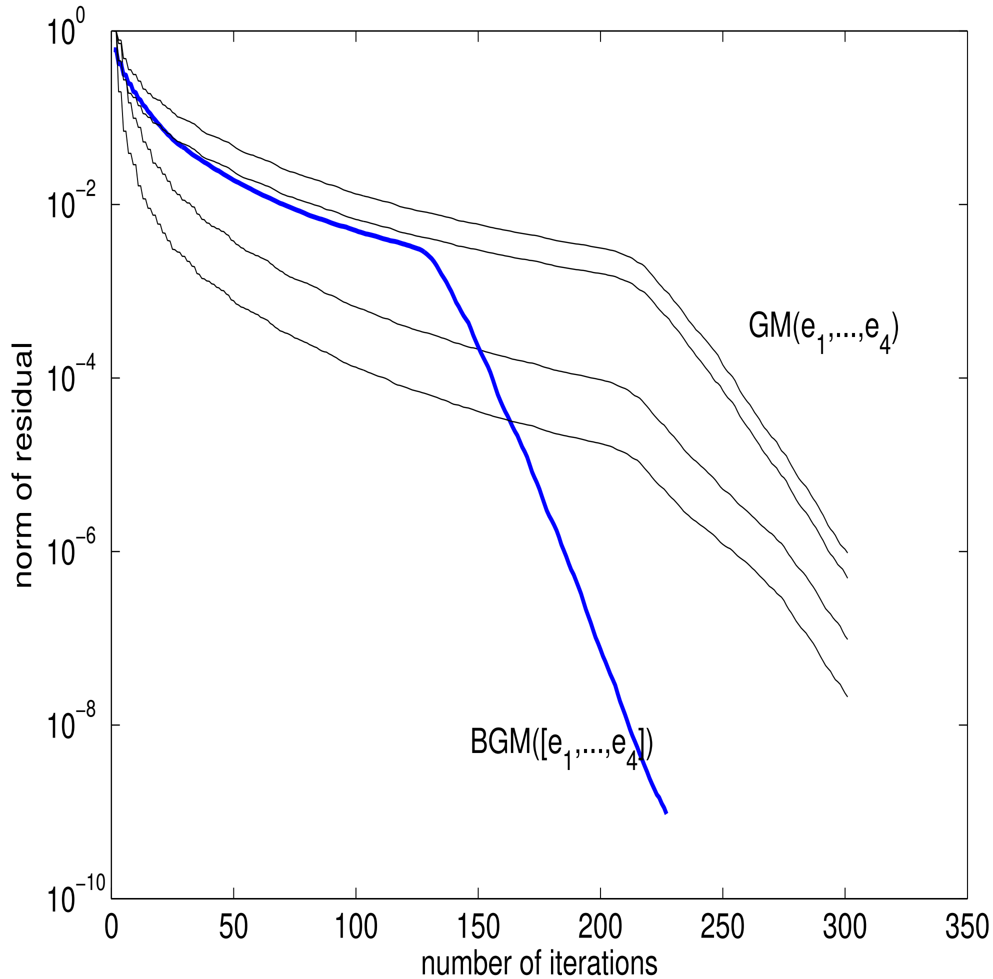
$\Rightarrow \quad A\mathcal{U}_k = \mathcal{U}_k\mathcal{H}_k + U_{k+1}\chi_{k+1,k}E_k^* = \mathcal{U}_{k+1}\underline{\mathcal{H}}_k \qquad (\mathcal{H}_k = \mathcal{U}_k^* A\mathcal{U}_k)$

$$\min_Y \|R_0 - A\mathcal{U}_k Y\| = \min_Y \| E_1\boldsymbol{\rho} - \underline{\mathcal{H}}_k Y \| \quad R_0 = U_1\boldsymbol{\rho}$$

$$\underline{\mathcal{H}}_k = \begin{bmatrix} \Box & \Box & \cdots & \Box \\ \Box & \Box & \cdots & \Box \\ O & \Box & \cdots & \Box \\ O & O & \ddots & \Box \\ O & O & O & \Box \end{bmatrix}$$

Block GMRES

$A \in \mathbb{R}^{6400\times 6400}$: FD discretiz. of $\mathcal{L}(u) = -\Delta u + \frac{1000}{x+y} u_x$ in $[-1,1]^2$



Coping with long-term recurrences

Restarted, Truncated, etc variants.

Coping with long-term recurrences

Restarted, Truncated, etc variants.

Restarted: Choose $m_{\max}$.

Set $x = x_0$, $r_0 = b - Ax_0$

for $i = 1, 2, \ldots$

$z \leftarrow$ GMRES($A, r_0, m_{\max}$) (or other method)

$x \leftarrow x + z, \quad r_0 = b - Ax$

Check Convergence

Pros and Cons

Pros:

- Shorter dependencies
- Lower and fixed memory requirements

Pros and Cons

Pros:

- Shorter dependencies
- Lower and fixed memory requirements

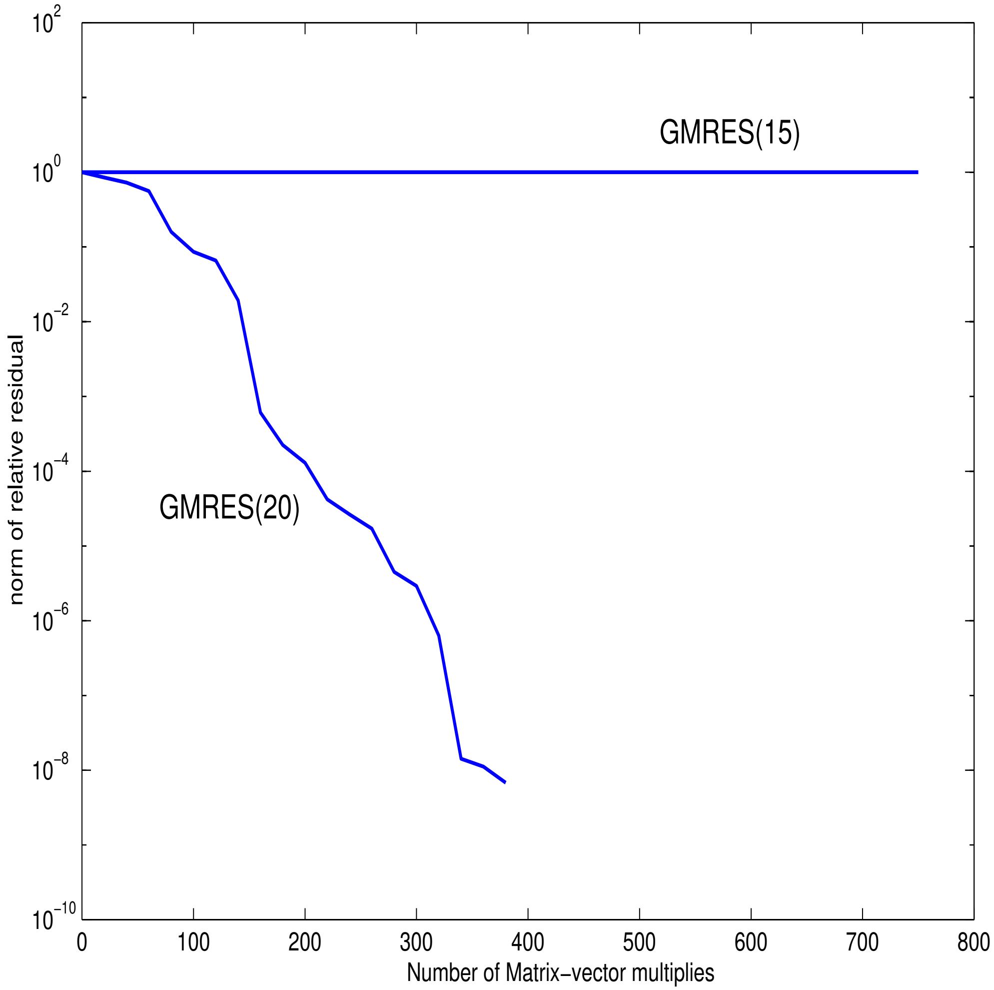
Cons:

- All optimality properties are lost

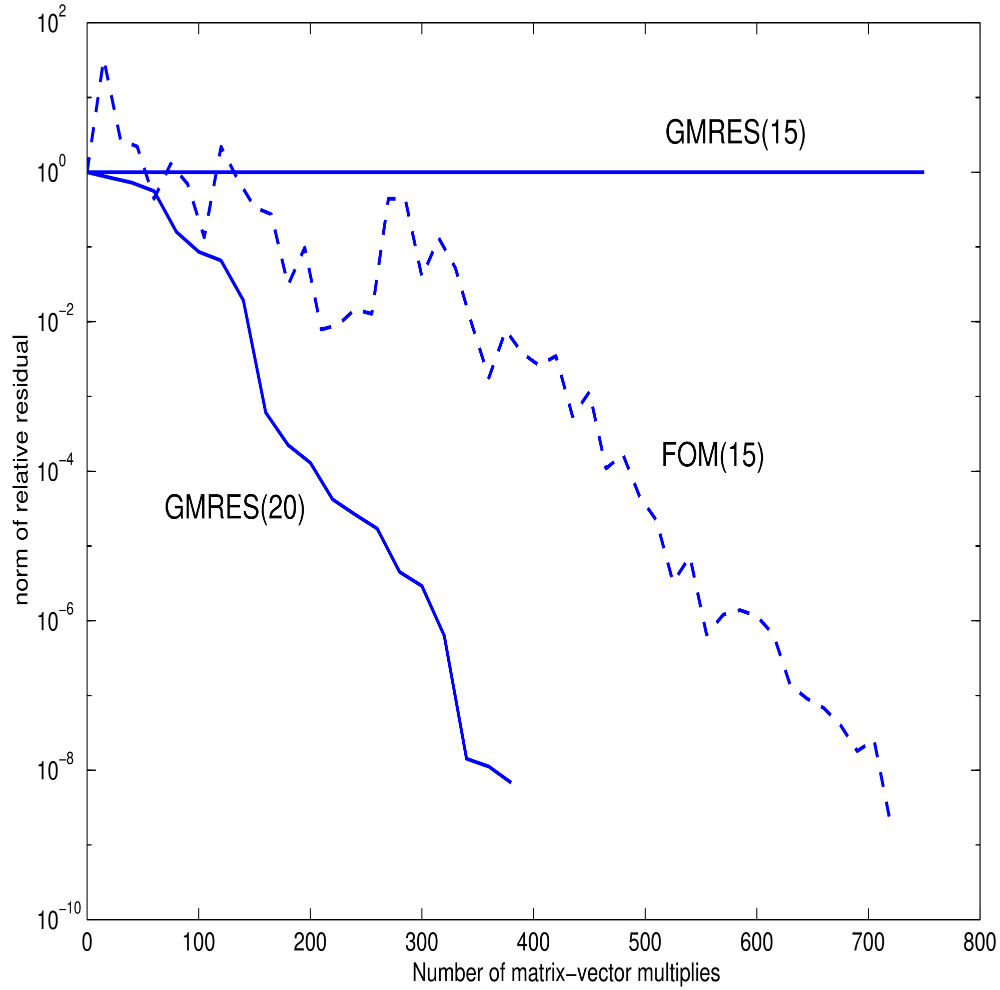
$$K_{m_{\max}}(A, r_0^{(0)}) + K_{m_{\max}}(A, r_0^{(1)}) + \ldots K_{m_{\max}}(A, r_0^{(k)}) + \ldots$$

- Additional parameter. What value for $m_{\max}$??

A problem with the restarting parameter? ...



A problem with the restarting parameter? ... or with the method?



Explanation

$$K_{m_{\max}}(A, r_0^{(0)}) + K_{m_{\max}}(A, r_0^{(1)}) + \ldots K_{m_{\max}}(A, r_0^{(k)}) + \ldots$$

GMRES: $r_0^{(k)} \in \mathrm{range}(V_{m_{\max}+1}^{(k-1)})$. Almost stagnation: $\rightarrow r_0^{(k)} \propto v_1^{(k-1)}$

Explanation

$$K_{m_{\max}}(A, r_0^{(0)}) + K_{m_{\max}}(A, r_0^{(1)}) + \ldots K_{m_{\max}}(A, r_0^{(k)}) + \ldots$$

GMRES: $r_0^{(k)} \in \mathrm{range}(V_{m_{\max}+1}^{(k-1)})$. Almost stagnation: $\rightarrow r_0^{(k)} \propto v_1^{(k-1)}$

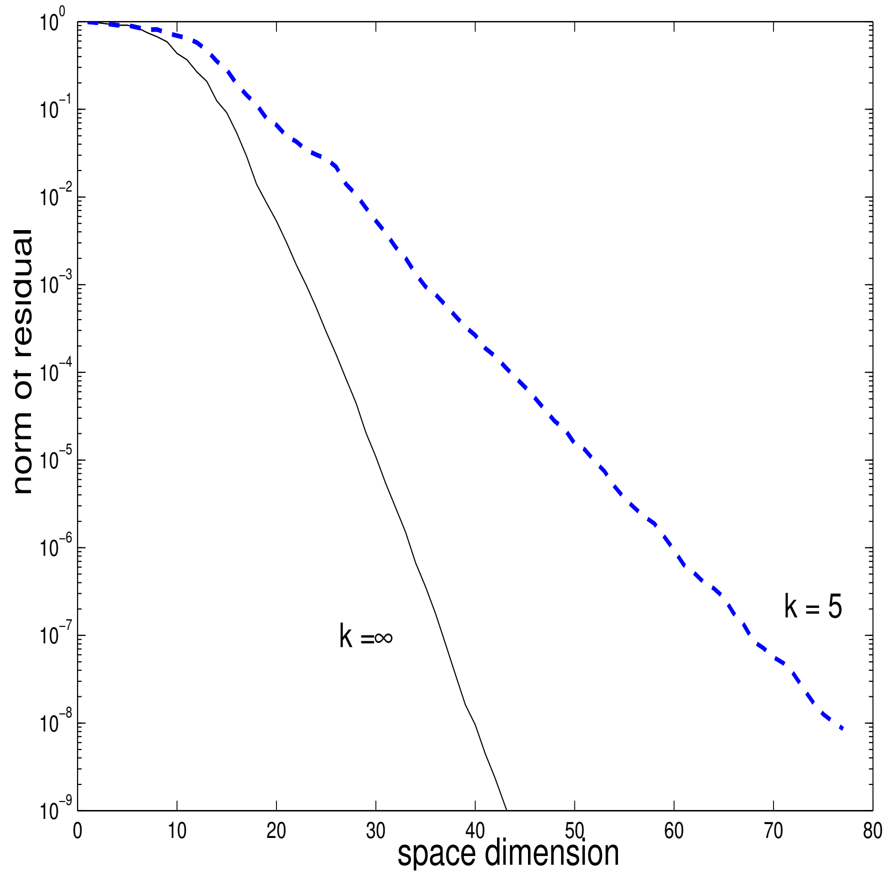
FOM: $r_0^{(k)} \propto v_{m_{\max}+1}^{(k-1)}$ Subspace keeps growing

Truncating

Only local orthogonalization (k-term recurrence, H_m banded)

Truncating

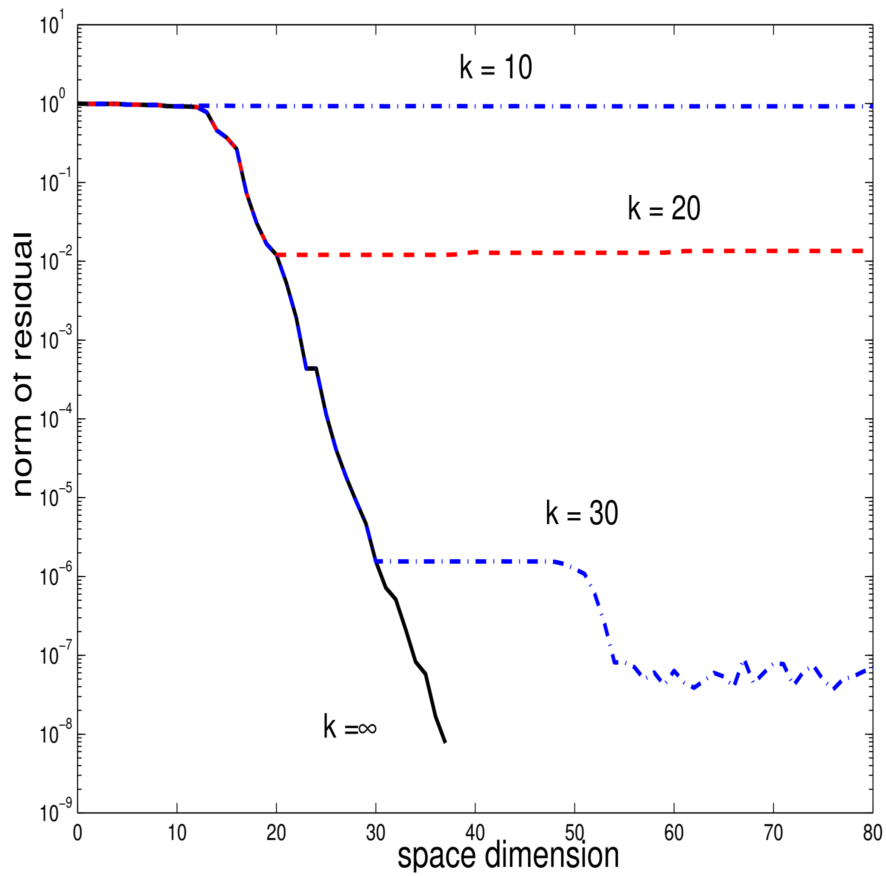
Only local orthogonalization (k-term recurrence, H_m banded)



a reasonable strategy

Truncating

...but not always good



Making it without long-term recurrences: short-term recurrences for A non-Hermitian

- Non-Hermitian Lanczos
- BiCGStab(ℓ): ℓ iterations of GMRES at every step
- IDR(s): $r_k \in \mathcal{G}_k$, where $\mathcal{G}_{k+1} \subset \mathcal{G}_k$

Stopping criterion: Problem dependence

Choice of tolerance:

- Direct method accurate up to machine precision (likely)
- Iterative method accurate up to what is wanted (hopefully)

Stopping criterion: Problem dependence

Choice of tolerance:

- Direct method accurate up to machine precision (likely)
- Iterative method accurate up to what is wanted (hopefully)

Algebraic problem: Discretization of PDEs

$$\text{error} \quad \to O(h)$$

h discretization parameter...

Stopping criterion: Problem dependence

Choice of criterion and norm:

$$\|b - Ax_k\|_2 \qquad \text{vs.} \qquad \|b - Ax_k\|_*$$

Stopping criterion: Problem dependence

Choice of criterion and norm:

$$\|b - Ax_k\|_2 \qquad \text{vs.} \qquad \|b - Ax_k\|_*$$

For instance, CG optimal: ($\|x\|_A^2 = x^* A x$)

$$\min_{x_k \in x_0 + K_k(A, r_0)} \|b - Ax_k\|_{A^{-1}} = \min_{x_k \in x_0 + K_k(A, r_0)} \|x - x_k\|_A$$

Available: Cheap, reliable estimates of $\|x - x_k\|_A$

Stopping criterion: Problem dependence

Choice of criterion and norm:

$$\|b - Ax_k\|_2 \qquad \text{vs.} \qquad \|b - Ax_k\|_*$$

For instance, CG optimal: ($\|x\|_A^2 = x^* A x$)

$$\min_{x_k \in x_0 + K_k(A, r_0)} \|b - Ax_k\|_{A^{-1}} = \min_{x_k \in x_0 + K_k(A, r_0)} \|x - x_k\|_A$$

Available: Cheap, reliable estimates of $\|x - x_k\|_A$

For instance, matrix G associated with FE error measure:

$$\min_{x_k} \|b - Ax_k\|_G$$

Matrix dependence

A may be very ill-conditioned

$\Rightarrow$ small residual does not necessarily imply small error

$$\frac{1}{\kappa(A)}\frac{\|b - Ax_k\|}{\|b\|} \leq \frac{\|x^\star - x_k\|}{\|x^\star\|} \leq \kappa(A)\frac{\|b - Ax_k\|}{\|b\|}$$

Well-known fact, but often not used

$$\frac{\|b - Ax_k\|}{\|b\|} \quad \text{vs} \quad \frac{\|b - Ax_k\|}{\|b\| + \|A\|_\star \|x_k\|}$$

(here $x_0 = 0$)

Matrix dependence

Inner-outer methods. e.g. Solve

$$BM^{-1}B^\top x = b$$

Each multiplication with $A = BM^{-1}B^\top$ requires solving a system with M

$$u = Av \quad \Leftrightarrow \quad \begin{array}{l} \tilde{u} = B^\top v \\ \tilde{\tilde{u}} \text{ solves } M\tilde{\tilde{u}} = \tilde{u} \\ u = B\tilde{\tilde{u}} \end{array}$$

How accurately should one solve with M?

Matrix dependence

Inner-outer methods. e.g. Solve

$$BM^{-1}B^\top x = b$$

Each multiplication with $A = BM^{-1}B^\top$ requires solving a system with M

$$u = Av \quad \Leftrightarrow \quad \begin{aligned} &\tilde{u} = B^\top v \\ &\tilde{\tilde{u}} \text{ solves } M\tilde{\tilde{u}} = \tilde{u} \\ &u = B\tilde{\tilde{u}} \end{aligned}$$

How accurately should one solve with M?

Note: True residual $r_k = b - BM^{-1}B^\top x_k$ not available!

How accurately should one solve with M?

Typically: Inner tolerance $<$ Outer tolerance

But: if optimal Krylov method is used to solve $BM^{-1}B^{\top}x = b$ then:

$$\text{Inner tolerance} = \text{c} \cdot \frac{\text{Outer tolerance}}{\text{current outer residual}}$$

Conclusions on methods

- Computational issues for Krylov solvers well understood
- Other tricks can be used (but not usually in black-box routines)
- Many ideas have wider applicability
- Theory is still under development

http://www.dm.unibo.it/~simoncin
valeria.simoncini@unibo.it

Preconditioning techniques

Determine matrix P such that

$$(PA)x = Pb$$

is "easier" to solve than $Ax = b$, that is

- Takes less CPU time
- P is cheap to construct
- P is reasonably cheap to apply

Note: Typically, P used in operators such as $y \leftarrow Pv$

Preconditioning techniques

Determine matrix P such that

$$(PA)x = Pb$$

is "easier" to solve than $Ax = b$, that is

- Takes less CPU time
- P is cheap to construct
- P is reasonably cheap to apply

Note: Typically, P used in operators such as $y \leftarrow Pv$

Choice criteria :

- P s.t. $PA \approx \alpha I$, with I identity matrix

Preconditioning techniques

Determine matrix P such that

$$(PA)x = Pb$$

is "easier" to solve than $Ax = b$, that is

- Takes less CPU time
- P is cheap to construct
- P is reasonably cheap to apply

Note: Typically, P used in operators such as $y \leftarrow Pv$

Choice criteria :

- P s.t. $PA \approx \alpha I$, with I identity matrix
- P s.t. P spectral properties similar to those of A^{-1}

Preconditioning techniques

Determine matrix P such that

$$(PA)x = Pb$$

is "easier" to solve than $Ax = b$, that is

- Takes less CPU time
- P is cheap to construct
- P is reasonably cheap to apply

Note: Typically, P used in operators such as $y \leftarrow Pv$

Choice criteria :

- P s.t. $PA \approx \alpha I$, with I identity matrix
- P s.t. P spectral properties similar to those of A^{-1}
- P "mimicks" the operator behind A
- ...

Preconditioning. 2

$$(PA)x = Pb$$

Classical strategy:

Determine P as $P = \mathcal{P}^{-1}$ con $\mathcal{P} \approx A$

$$\mathcal{P}^{-1}Ax = \mathcal{P}^{-1}b$$

Preconditioning. 2

$$(PA)x = Pb$$

Classical strategy:

Determine P as $P = \mathcal{P}^{-1}$ con $\mathcal{P} \approx A$

$$\mathcal{P}^{-1}Ax = \mathcal{P}^{-1}b$$

hoping that:

$\Rightarrow$ $\mathcal{P} \approx A$ then $\mathcal{P}^{-1} \approx A^{-1}$ so that $\mathcal{P}^{-1}A \approx I$

$\Rightarrow$ $\mathcal{P}^{-1}$ cheap to apply (via $y \leftarrow \mathcal{P}^{-1}v$), that is, solving

$$\mathcal{P}y = v$$

is far less expensive than $Ax = b$

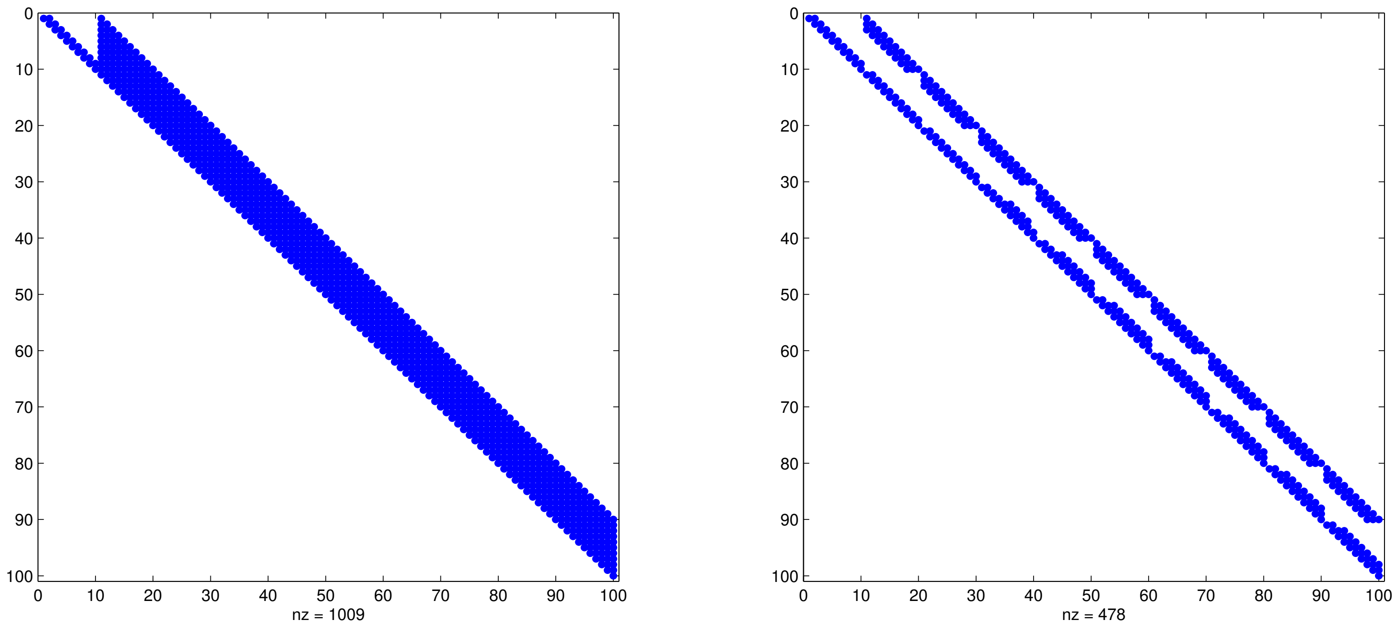
$\star$ Example: $\mathcal{P} = \mathrm{diag}(A)$: cheap, but little effective....

An example: Cholesky incomplete decomposition

A sym.pos.def. $\qquad A = LL^T \approx L_0 L_0^T$

L_0 obtained from L by threshold chopping (element values below `tol` zeroed out)

L Original $\qquad\qquad$ approximation L_0



A corresponds to the Poisson operator, and $\texttt{tol} = 10^{-2}$

A possible strategy for incomplete LU

(ILUT, Algorithm 10.6, Saad)

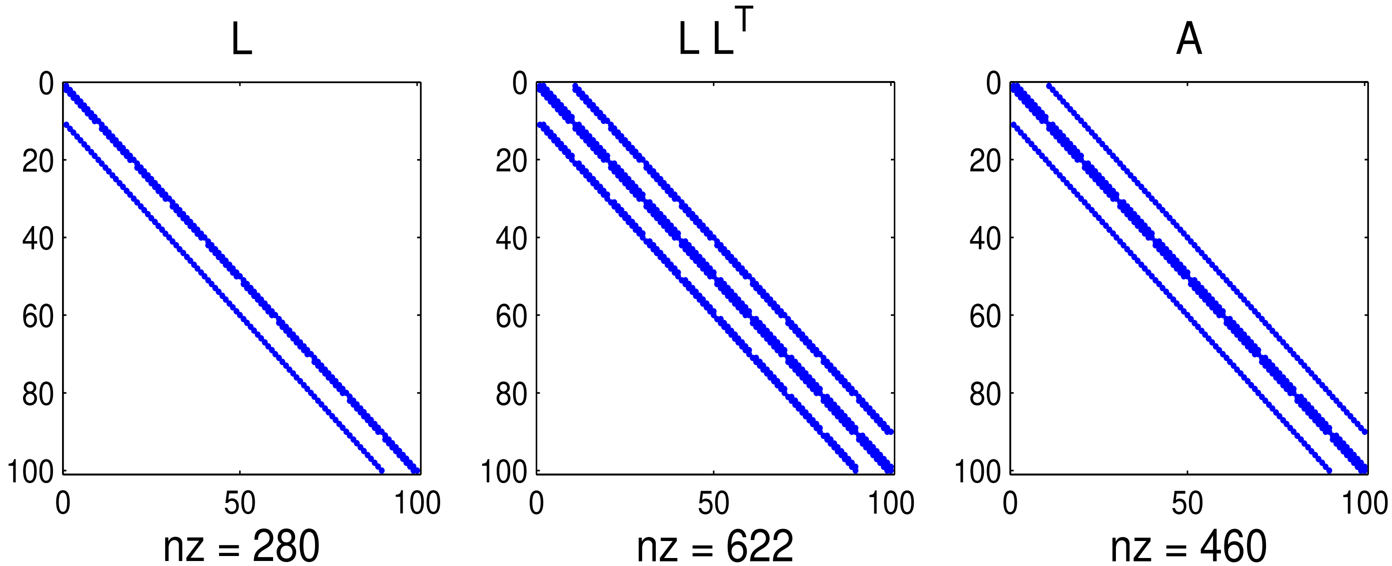
A $n \times n$, "threshold dropping" strategy

```
1.  for i = 1...n do
2.      w = a_{i,:} (with w = (w_1, ..., w_n))
3.      for k = 1....i-1 and w_k ≠ 0 do
4.          w_k := w_k / a_{k,k}
5.          Apply the ``dropping rule'' to w_k
6.          If w_k ≠ 0, w := w - w_k u_{k,:},end
7.      endfor
8.      Apply the ``dropping rule'' to the row w
9.      l_{i,1:i-1} = w_{1:i-1}, u_{i,i:n} = w_{i:n}
10. endfor
```

zero threshold: ILU(0) and CHOLINC(0)

$A \approx LU$ such that L and U have the same sparsity pattern as A

(`nnz(L + U - speye(size(A))) = nnz(A)`)



...also other strategies...

THEOREM. If A is a P-matrix, then there exists an incomplete factorization of A with fixed zero sparsity pattern, such that $A = LU - R$ with LU non-singular

PCG, maintaing symmetry

For A sym pos.def., $A \approx P = LL^T$. The preconditioned problem:

$$Ax = b \quad \Rightarrow \quad \underbrace{L^{-1}AL^{-T}}_{\widetilde{A}}\underbrace{L^T x}_{\tilde{x}} = \underbrace{L^{-1}b}_{\tilde{b}},$$

PCG, maintaing symmetry

For A sym pos.def., $A \approx P = LL^T$. The preconditioned problem:

$$Ax = b \quad \Rightarrow \quad \underbrace{L^{-1}AL^{-T}}_{\widetilde{A}}\underbrace{L^{T}x}_{\tilde{x}} = \underbrace{L^{-1}b}_{\tilde{b}},$$

For $\tilde{p}^{(0)} = \tilde{r}^{(0)} = \tilde{b} - \widetilde{A}\tilde{x}^{(0)} = L^{-1}(b - Ax^{(0)}) = L^{-1}r^{(0)}$, we have

$\tilde{x}^{(j+1)} = \tilde{x}^{(j)} + \alpha_j \tilde{p}^{(j)}$, with $\qquad \alpha_j = \frac{(\tilde{r}^{(j)}, \tilde{r}^{(j)})}{(\widetilde{A}\tilde{p}^{(j)}, \tilde{p}^{(j)})}$

$\tilde{r}^{(j+1)} = \tilde{r}^{(j)} - \alpha_j \widetilde{A}\tilde{p}^{(j)}$

$\tilde{p}^{(j+1)} = \tilde{r}^{(j+1)} + \beta_j \tilde{p}^{(j)}$, con $\qquad \beta_j = \frac{(\tilde{r}^{(j+1)}, \tilde{r}^{(j+1)})}{(\tilde{r}^{(j)}, \tilde{r}^{(j)})}$

PCG, maintaing symmetry

For A sym pos.def., $A \approx P = LL^T$. The preconditioned problem:

$$Ax = b \quad \Rightarrow \quad \underbrace{L^{-1}AL^{-T}}_{\widetilde{A}}\underbrace{L^Tx}_{\tilde{x}} = \underbrace{L^{-1}b}_{\tilde{b}},$$

For $\tilde{p}^{(0)} = \tilde{r}^{(0)} = \tilde{b} - \widetilde{A}\tilde{x}^{(0)} = L^{-1}(b - Ax^{(0)}) = L^{-1}r^{(0)}$, we have

$\tilde{x}^{(j+1)} = \tilde{x}^{(j)} + \alpha_j \tilde{p}^{(j)}$, with $\quad \alpha_j = \frac{(\tilde{r}^{(j)}, \tilde{r}^{(j)})}{(\widetilde{A}\tilde{p}^{(j)}, \tilde{p}^{(j)})}$

$L^T x^{(j+1)} = L^T x^{(j)} + \alpha_j L^{-1} p^{(j)}$, with $\quad \alpha_j = \frac{(L^{-1}r^{(j)}, L^{-1}r^{(j)})}{(L^{-1}AL^{-T}L^{-1}p^{(j)}, L^{-1}p^{(j)})}$

$\tilde{r}^{(j+1)} = \tilde{r}^{(j)} - \alpha_j \widetilde{A}\tilde{p}^{(j)}$

$L^{-1}r^{(j+1)} = L^{-1}r^{(j)} - \alpha_j L^{-1}AL^{-T}L^{-1}p^{(j)}$

$\tilde{p}^{(j+1)} = \tilde{r}^{(j+1)} + \beta_j \tilde{p}^{(j)}$, with $\quad \beta_j = \frac{(\tilde{r}^{(j+1)}, \tilde{r}^{(j+1)})}{(\tilde{r}^{(j)}, \tilde{r}^{(j)})}$

$L^{-1}p^{(j+1)} = L^{-1}r^{(j+1)} + \beta_j L^{-1}p^{(j)}$, with $\quad \beta_j = \frac{(L^{-1}r^{(j+1)}, L^{-1}r^{(j+1)})}{(L^{-1}r^{(j)}, L^{-1}r^{(j)})}$

PCG, maintaing symmetry

For A sym pos.def., $A \approx P = LL^T$. The preconditioned problem:

$$Ax = b \quad \Rightarrow \quad \underbrace{L^{-1}AL^{-T}}_{\widetilde{A}} \underbrace{L^T x}_{\tilde{x}} = \underbrace{L^{-1}b}_{\tilde{b}},$$

For $\tilde{p}^{(0)} = \tilde{r}^{(0)} = \tilde{b} - \widetilde{A}\tilde{x}^{(0)} = L^{-1}(b - Ax^{(0)}) = L^{-1}r^{(0)}$, we have

$\tilde{x}^{(j+1)} = \tilde{x}^{(j)} + \alpha_j \tilde{p}^{(j)}$, with $\quad \alpha_j = \frac{(\tilde{r}^{(j)}, \tilde{r}^{(j)})}{(\widetilde{A}\tilde{p}^{(j)}, \tilde{p}^{(j)})}$

$x^{(j+1)} = x^{(j)} + \alpha_j L^{-T}L^{-1}p^{(j)}$, with $\quad \alpha_j = \frac{(r^{(j)}, L^{-T}L^{-1}r^{(j)})}{(AL^{-T}L^{-1}p^{(j)}, L^{-T}L^{-1}p^{(j)})}$

$\tilde{r}^{(j+1)} = \tilde{r}^{(j)} - \alpha_j \widetilde{A}\tilde{p}^{(j)}$

$r^{(j+1)} = r^{(j)} - \alpha_j AL^{-T}L^{-1}p^{(j)}$

$\tilde{p}^{(j+1)} = \tilde{r}^{(j+1)} + \beta_j \tilde{p}^{(j)}$, with $\quad \beta_j = \frac{(\tilde{r}^{(j+1)}, \tilde{r}^{(j+1)})}{(\tilde{r}^{(j)}, \tilde{p}^{(j)})}$

$L^{-T}L^{-1}p^{(j+1)} = L^{-T}L^{-1}r^{(j+1)} + \beta_j L^{-T}L^{-1}p^{(j)}$, with $\quad \beta_j = \frac{(r^{(j+1)}, L^{-T}L^{-1}r^{(j+1)})}{(r^{(j)}, L^{-T}L^{-1}r^{(j)})}$

PCG, maintaing symmetry

For A sym pos.def., $A \approx P = LL^T$. The preconditioned problem:

$$Ax = b \quad \Rightarrow \quad \underbrace{L^{-1}AL^{-T}}_{\widetilde{A}} \underbrace{L^T x}_{\tilde{x}} = \underbrace{L^{-1}b}_{\tilde{b}},$$

For $\tilde{p}^{(0)} = \tilde{r}^{(0)} = \tilde{b} - \widetilde{A}\tilde{x}^{(0)} = L^{-1}(b - Ax^{(0)}) = L^{-1}r^{(0)}$, we have

With $\hat{p}^{(0)} = L^{-T}L^{-1}p^{(0)} = P^{-1}p^{(0)}$ and $z^{(j)} = L^{-T}L^{-1}r^{(j)} = P^{-1}r^{(j)}$:

$x^{(j+1)} = x^{(j)} + \alpha_j \hat{p}^{(j)}$ with $\qquad \alpha_j = \frac{(r^{(j)}, z^{(j)})}{(A\hat{p}^{(j)}, \hat{p}^{(j)})}$

$r^{(j+1)} = r^{(j)} - \alpha_j A\hat{p}^{(j)}$

$\hat{p}^{(j+1)} = z^{(j+1)} + \beta_j \hat{p}^{(j)}$, with $\beta_j = \frac{(r^{(j+1)}, z^{(j+1)})}{(r^{(j)}, z^{(j)})}$

Practical preconditioning strategies

- LU-type approx decomposition of A: $\rightarrow Pv = U^{-1}L^{-1}v$
- Algebraic multigrid (approximate representation of A on smaller version of the matrix - recursive procedure)
- Geometric multigrid (operator and domain dependent)
- Functional approximation of the underlying operator

A comparison :
Incomplete Cholesky and Algebraic Multigrid

Poisson, 2D problem on $[0,1]^2$. Matrices of dim $n = 2^k \times 2^k$

grid	incomplete Chol		AMG	
nodes per dim	# it's	CPU time	# it's	CPU time
2^4	11	0.008	6	0.18
2^5	18	0.007	6	0.20
2^6	33	0.04	7	0.22
2^7	58	0.29	7	0.32
2^8	106	2.27	8	0.71

For 2^8, dim$(A) = 65536 \times 65536$

!! Preconditioned CG with AMG gives grid independent # it's !!

A comparison : Incomplete Cholesky and Algebraic Multigrid

Poisson, 2D problem on $[0,1]^2$. Matrices of dim $n = 2^k \times 2^k$

grid	incomplete Chol		AMG	
nodes per dim	# it's	CPU time	# it's	CPU time
2^4	11	0.008	6	0.18
2^5	18	0.007	6	0.20
2^6	33	0.04	7	0.22
2^7	58	0.29	7	0.32
2^8	106	2.27	8	0.71

For 2^8, dim$(A) = 65536 \times 65536$

!! Preconditioned CG with AMG gives grid independent # it's !!

Remark: For 2^8, `tic;`$A\backslash b$`;toc`, gives: `Elapsed time is 0.58 secs`

Algebraic Multigrid (AMG)

Consider the original system

$$A_h u^h = f^h \qquad (\star)$$

The error vector is split in two parts: an *oscillatory* component (high freq.) and a *regular* component (smooth, low freq.)

A Multigrid (or multilevel) type method for a linear system is made of two ingredients:

- A smoothing step of the oscillatory portion:

  usually a few iterations of a classical method (e.g., Jacobi, Gauss-Seidel)

- A correction on a coarser grid for the smooth part

  The system $(\star)$ is approximated by a system on a coarser grid:

A^H, f^H such that

$$A_H = I_h^H A_h I_H^h, \qquad f^H = I_h^H f^h$$

Conceptually similar to a Galerkin projection type procedure:

I_h^H: restriction operator, full rank

I_H^h: prolongation operator, rull rank

with

$$I_h^H = (I_H^h)^T \qquad \text{(transposition)}$$

Remark: *Geometric* Multigrid uses the physical grid. *Algebraic* Multigrid use the matrix elements

(matrix indexes $\equiv$ grid nodes)

Algebraic Multigrid (AMG)

General procedure (on two grids):

1. Perform n_1 steps of smoothing (e.g., Jacobi) on $A_h u^h = f^h$
2. Compute the residual $r^h = f^h - A_h u^h \equiv A e^h$
3. Project (restrict) to the coarse grid $r^H = I_h^H r^h$
4. Solve on coarse grid: $A_H e^H = r^H$
5. Add (prolong) $u^h := u^h + I_H^h e^H$
6. Take n_2 steps of smoothing on $A_h u^h = f^h$

Algebraic Multigrid (AMG). The coarse grid

Determine A_H from A_h, A_H is a subset of the rows/columns of A_h

(strong connection among the elements of A_H)

DEF. Let $\theta \in (0,1]$ be a fixed threshold. The variable u_i *strongly* depends on the variable u_j if

$$-a_{ij} \geq \theta \max_{k \neq i} \{-a_{ik}\}$$

$\Rightarrow$ non-diagonal positive elements have a weak connection

The following steps should be taken (where: node= pair of indexes)

1. Define a "strength" matrix (A_f) by eliminating the weak connections
2. Choose an independent set of strong nodes of A_f
3. Add possible nodes to have a correct proloungation operator

Spectral equivalence

Under particular conditions[a] on the matrix A, it can be proved that the AMG preconditioner is spectrally equivalent to A, that is:

There exist $\alpha_1, \alpha_2 > 0$ independent of the dimension of A such that

$$\alpha_1(x, Px) \leq (x, Ax) \leq \alpha_2(x, Px), \qquad \forall x \neq 0$$

[a]e.g., if A is Hpd is an M-matrix, that is with $a_{ii} > 0\ \forall i$ and $a_{ij} \leq 0\ \forall i \neq j$, with non-negative inverse - the usual discretization of the Laplacian.

Spectral equivalence

Under particular conditions[a] on the matrix A, it can be proved that the AMG preconditioner is spectrally equivalent to A, that is:

There exist $\alpha_1, \alpha_2 > 0$ independent of the dimension of A such that

$$\alpha_1(x, Px) \leq (x, Ax) \leq \alpha_2(x, Px), \qquad \forall x \neq 0$$

In our context:

$$P^{-1}Av = \lambda v \qquad \Leftrightarrow \qquad Av = \lambda Pv$$

so that

$$\lambda = \frac{(v, Av)}{(v, Pv)}, \qquad \min_{x \neq 0} \frac{(x, Ax)}{(x, Px)} \leq \lambda \leq \max_{x \neq 0} \frac{(x, Ax)}{(x, Px)}$$

$\Rightarrow$ The spectral interval of the preconditioned problems **does not** depend on the problem dimension (or on the grid!)

[a] e.g., if A is Hpd is an M-matrix, that is with $a_{ii} > 0 \ \forall i$ and $a_{ij} \leq 0 \ \forall i \neq j$, with non-negative inverse - the usual discretization of the Laplacian.