Introduction to Simulation - Lecture 7

Krylov-Subspace Matrix Solution Methods

Part II
Jacob White

Thanks to Deepak Ramaswamy, Michal Rewienski, and Karen Veroy
Outline

• Reminder about GCR
  – Residual minimizing solution
  – Krylov Subspace
  – Polynomial Connection

• Review Eigenvalues and Norms
  – Induced Norms
  – Spectral mapping theorem

• Estimating Convergence Rate
  – Chebychev Polynomials

• Preconditioners
  – Diagonal Preconditioners
  – Approximate LU preconditioners
Generalized Conjugate Residual Algorithm

\[ r^0 = b - Ax^0 \]

For \( j = 0 \) to \( k-1 \)

\[ p_j = r^j \]

Residual is next search direction

For \( i = 0 \) to \( j-1 \)

\[ p_j \leftarrow p_j - \left( M p_j \right)^T \left( M p_j \right) p_i \]

Orthogonalize Search Direction

\[ p_j \leftarrow \frac{1}{\sqrt{\left( M p_j \right)^T \left( M p_j \right)}} \]

Normalize

\[ x^{j+1} = x^j + \left( r^j \right)^T \left( M p_j \right) p_j \]

Update Solution

\[ r^{j+1} = r^j - \left( r^j \right)^T \left( M p_j \right) M p_j \]

Update Residual
1) orthogonalize the $M r^j$'s

2) compute the $r$ minimizing solution $x^k$
Generalized Conjugate Residual Algorithm

• First search direction
  \[ r^0 = b - Mx^0 = b, \quad p_0 = \frac{r^0}{\|Mr^0\|} \]

• Residual minimizing solution
  \[ x^1 = \left( \left( r^0 \right)^T M p_0 \right) p_0 \]

• Second Search Direction
  \[ r^1 = b - Mx^1 = r^0 - \gamma_1 Mr^0 \]
  \[ p_1 = \frac{r^1 - \beta_{1,0} p_0}{\|M(r^1 - \beta_{1,0} p_0)\|} \]
Generalized Conjugate Residual Algorithm

- Residual minimizing solution
  \[ x^2 = x^1 + \left( (r^1)^T M p_1 \right) p_1 \]

- Third Search Direction
  \[ r^2 = b - M x^2 = r^0 - \gamma_{2,1} M r^0 - \gamma_{2,0} M^2 r^0 \]
  \[ p_2 = \frac{r^1 - \beta_{2,0} p_0 - \beta_{2,1} p_1}{\| M (r^1 - \beta_{2,0} p_0 - \beta_{2,1} p_1) \|} \]
The kth step of GCR

\[
\tilde{p}_k = r^k - \sum_{j=0}^{k-1} (Mr^k)^T (Mp_j) p_j
\]

\[
P_k = \frac{\tilde{p}_k}{\|Mp_k\|}
\]

\[
\alpha_k = (r^k)^T (Mp_k)
\]

Orthogonalize and normalize search direction

Determine optimal stepsize in kth search direction

Update the solution and the residual

\[
x^{k+1} = x^k + \alpha_k P_k
\]

\[
r^{k+1} = r^k - \alpha_k Mp_k
\]
If $\alpha_j \neq 0$ for all $j \leq k$ in GCR, then

1) $\text{span} \{ p_0, p_1, \ldots, p_k \} = \text{span} \{ r^0, Mr^0, \ldots, Mr^k \}$

2) $x^{k+1} = \xi_k(M)r^0$, $\xi_k$ is the $k^{th}$ order poly

   minimizing $\|r^{k+1}\|^2$

3) $r^{k+1} = b - Mx^{k+1} = r^0 - M\xi_k(M)r^0$

   $= (I - M\xi_k(M))r^0 \equiv \varphi_{k+1}(M)r^0$

   where $\varphi_{k+1}(M)r^0$ is the $(k+1)^{th}$ order poly

   minimizing $\|r^{k+1}\|^2$ subject to $\varphi_{k+1}(0) = 1$
Krylov Methods

If \( x^{k+1} \in \text{span}\{r^0, Mr^0, \ldots, Mr^k\} \) minimizes \( \|r^{k+1}\|_2^2 \)

1) \( x^{k+1} = \xi_k(M)r^0 \), \( \xi_k \) is the \( k \)th order poly minimizing \( \|r^{k+1}\|_2^2 \)

2) \( r^{k+1} = b - Mx^{k+1} = (I - M\xi_k(M))r^0 = \wp_{k+1}(M)r^0 \)

where \( \wp_{k+1}(M)r^0 \) is the \((k+1)\)th order poly minimizing \( \|r^{k+1}\|_2^2 \) subject to \( \wp_{k+1}(0) = 1 \)

Polynomial Property only a function of solution space and residual minimization

SMA-HPC ©2003 MIT
Krylov Methods

“No-leak Example”

Insulated bar and Matrix

$T(0)$
Near End
Temperature

$T(1)$
Far End
Temperature

Discretization

$\begin{bmatrix}
2 & -1 \\
-1 & 2 \\
\vdots & -1 \\
-1 & 2
\end{bmatrix}$

Nodal Equation Form

Incoming Heat

$m$

$M$

$T(0)$

$T(1)$
Krylov Methods

“Leaky” Example

Conducting bar and Matrix

\[
\begin{bmatrix}
2.01 & -1 \\
-1 & 2.01 \\
\vdots & \ddots & \ddots \\
-1 & 2.01 \\
\end{bmatrix}
\]

Nodal Equation Form

Discretization

Near End Temperature

T(0)

Far End Temperature

T(1)
GCR Performance (Rhs = -1,+1,-1,+1,...)

Plot of \( \log(\text{residual}) \) versus Iteration

SMA-HPC ©2003 MIT
Residual Minimizing Optimality Property

$\|r^{k+1}\| \leq \|\tilde{\phi}_{k+1}(M)r^0\| \leq \|\tilde{\phi}_{k+1}(M)\|r^0\|

\tilde{\phi}_{k+1} \text{ is any } k^{th} \text{ order poly such that } \tilde{\phi}_{k+1}(0) = 1$

Therefore

Any polynomial which satisfies the constraints can be used to get an upper bound on

$\frac{\|r^{k+1}\|}{\|r^0\|}$
Suppose \( y = Mx \)

How much larger is \( y \) than \( x \)?

OR

How much does \( M \) magnify \( x \)?
**Induced Norms**

**Vector Norm Review**

\[ \| x \|_2 = \sqrt{\sum_{i=1}^{n} |x_i|^2} \quad \| x \|_2 < 1 \]

\[ \| x \|_1 = \sum_{i=1}^{n} |x_i| \quad \| x \|_1 < 1 \]

\[ \| x \|_\infty = \max_i |x_i| \quad \| x \|_\infty < 1 \]
**Induced Matrix Norms**

**Definition:**

\[
\|M\|_l = \max_x \frac{\|Mx\|_l}{\|x\|_l} = \max_{\|x\|_l = 1} \|Mx\|_l
\]

**Examples**

\[
\|M\|_1 = \max_i \sum_{j=1}^N |M_{ij}| \quad \text{Max Column Sum}
\]

\[
\|M\|_\infty = \max_j \sum_{i=1}^N |M_{ij}| \quad \text{Max Row Sum}
\]
As the algebra on the slide shows the relative changes in the solution $x$ is bounded by an $A$-dependent factor times the relative changes in $A$. The factor

$$\| A^{-1} \| \| A \|$$

was historically referred to as the condition number of $A$, but that definition has been abandoned as then the condition number is norm-dependent. Instead the condition number of $A$ is the ratio of singular values of $A$.

$$\text{cond}(A) = \frac{\sigma_{\text{max}}(A)}{\sigma_{\text{min}}(A)}$$

Singular values are outside the scope of this course, consider consulting Trefethen & Bau.
Given a polynomial
\[ f(x) = a_0 + a_1 x + \ldots + a_p x^p \]
Apply the polynomial to a matrix
\[ f(M) = a_0 + a_1 M + \ldots + a_p M^p \]
Then
\[ \text{spectrum}(f(M)) = f(\text{spectrum}(M)) \]
Krylov Methods

Convergence Analysis

Norm of matrix polynomials

\[ \| \varphi_k(M) \| = \| \begin{bmatrix} \ldots & \ldots & \ldots \\ \tilde{u}_1 & \tilde{u}_2 & \tilde{u}_N \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \varphi_k(\lambda_1) \\ \vdots \\ \varphi_k(\lambda_N) \end{bmatrix} \| \leq \text{Cond}(U) \begin{bmatrix} \varphi_k(\lambda_1) \\ \vdots \\ \varphi_k(\lambda_N) \end{bmatrix} \]

- \( \| \cdot \| \) denotes the norm of the matrix polynomial
- \( \varphi_k(M) \) represents the polynomial of degree \( k \)
- \( \lambda_1, \lambda_2, \ldots, \lambda_N \) are the eigenvalues of \( M \)
- \( \tilde{u}_1, \tilde{u}_2, \ldots, \tilde{u}_N \) are the eigenvectors of \( M \)
- \( \text{Cond}(U) \) is the condition number of the eigenspace of \( M \)
Krylov Methods

Convergence Analysis

Norm of matrix polynomials

\[ \left\| \varphi_k (\lambda) \right\|_2 = \max_i \sqrt{\sum_i |\varphi_k (\lambda_i) x_i|^2} \]

\[ = \max_i |\varphi_k (\lambda_i)| \]

\[ \left\| \varphi_k (M) \right\| \leq \text{cond}(V) \max_i |\varphi_k (\lambda_i)| \]
1) A residual minimizing Krylov subspace algorithm converges to the exact solution in at most $n$ steps.

Proof: Let $\tilde{\phi}_n(x) = (x - \lambda_1)(x - \lambda_2)\ldots(x - \lambda_n)$

where $\lambda_i \in \lambda(M)$. Then, $\max_i |\tilde{\phi}_n(\lambda_i)| = 0$.

$\Rightarrow \|\tilde{\phi}_n(M)\| = 0$ and therefore $\|r^n\| = 0$

2) If $M$ has only $q$ distinct e-values, the residual minimizing Krylov subspace algorithm converges in at most $q$ steps.

Proof: Let $\tilde{\phi}_q(x) = (x - \lambda_1)(x - \lambda_2)\ldots(x - \lambda_q)$
Krylov Methods

If $M = M^T$ then

1) $M$ has orthonormal eigenvectors

$$\Rightarrow \text{cond}(V) = \begin{vmatrix} \overline{u}_1 & \cdots & \overline{u}_N \\ \overline{u}_1 & \cdots & \overline{u}_N \end{vmatrix} = 1$$

$$\Rightarrow \|\phi_k(M)\| = \max_i |\phi_k(\lambda_i)|$$

2) $M$ has real eigenvalues

If $M$ is positive definite, then $\lambda(M) > 0$
Residual Poly Picture for Heat Conducting Bar Matrix
No loss to air (n=10)

* = evals(M)
- = 5th order poly
- = 8th order poly
Keep $|\phi_k(\lambda_i)|$ as small as possible:
Strategically place zeros of the poly

Residual Poly Picture for Heat Conducting Bar Matrix
No loss to air (n=10)
Consider $\lambda(M) \in [\lambda_{\min}, \lambda_{\max}]$, $\lambda_{\min} > 0$

Then a good polynomial ($\|\tilde{p}_k(M)\|$ is small) can be found by solving the min-max problem

$$\min_{kth \ order \ polys \ s.t. \ \tilde{p}_k(0)=1} \ max_{x \in [\lambda_{\min}, \lambda_{\max}]} |\tilde{p}_k(x)|$$

The min-max problem is exactly solved by Chebyshev Polynomials.
The Chebyshev Polynomial

\[ C_k(x) \equiv \cos \left( k \cos^{-1}(x) \right) \quad x \in [-1,1] \]

\[ \min_{k\text{th order}} \max_{x \in [\lambda_{\min}, \lambda_{\max}]} |\tilde{\varphi}_k(x)| \]

\[ = \max_{x \in [\lambda_{\min}, \lambda_{\max}]} \left| \frac{C_k \left( 1 + 2 \frac{\lambda_{\min} - x}{\lambda_{\max} - \lambda_{\min}} \right) }{C_k \left( 1 + 2 \frac{\lambda_{\min}}{\lambda_{\max} - \lambda_{\min}} \right) } \right| \]
Chebyshev Polynomials minimizing over $[1,10]$
\[
\min_{k\text{th order}} \max_{x \in [\lambda_{\min}, \lambda_{\max}]} \left| \widetilde{v}_k(x) \right|
\]

\[
= \sqrt{k} \left( 1 - 2 \frac{\lambda_{\max}}{\lambda_{\max} - \lambda_{\min}} \right)
\]

\[
\leq 2 \left( \frac{\lambda_{\max}}{\sqrt{\lambda_{\min}}} - 1 \right)^{\frac{k}{2}} \left( \frac{\lambda_{\max}}{\sqrt{\lambda_{\min}}} + 1 \right)^{\frac{k}{2}}
\]
Krylov Methods

Convergence for $M = M^T$

Chebychev Result

If $\lambda(M) \in [\lambda_{\min}, \lambda_{\max}]$, $\lambda_{\min} > 0$

$$\|r^k\| \leq 2 \left( \frac{\sqrt{\lambda_{\max}} - 1}{\sqrt{\lambda_{\min}} + 1} \right)^k \|r^0\|$$
Krylov Methods

Diagonal Example

\[
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 1 & \ddots & \\
\vdots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & 1
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 0 & \cdots & 0 \\
0 & 2 & \ddots & \\
\vdots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & N
\end{bmatrix}
\]

For which problem will GCR Converge Faster?
Let $M = D + M_{nd}$

Apply GCR to $\left( D^{-1}M \right)x = \left( I + D^{-1}M_{nd} \right)x = D^{-1}b$

- The Inverse of a diagonal is cheap to compute
- Usually improves convergence
Heat Conducting Bar example

Discretized system

\[
\begin{bmatrix}
2 + \gamma & -1 \\
-1 & 2 + \gamma \\
\vdots & \ddots & \ddots & \ddots \\
-1 & 1 + \gamma + 100 & -100 & 1 + \gamma + 100 & 1 \\
-100 & -100 & 1 + \gamma + 100 & 1 & \ddots & \ddots & \ddots & \ddots & \ddots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
-1 & -1 & 2 + \gamma & -1 & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
\end{bmatrix} \begin{bmatrix}
\hat{u}_1 \\
\hat{u}_2 \\
\vdots \\
\hat{u}_{n-1} \\
\hat{u}_n
\end{bmatrix} = 
\begin{bmatrix}
\gamma \\
\gamma \\
\vdots \\
\gamma \\
\gamma
\end{bmatrix} = \begin{bmatrix}
f(x_1) \\
f(x_2) \\
\vdots \\
f(x_{n-1}) \\
f(x_n)
\end{bmatrix}
\]

\[\frac{\lambda_{\text{max}}}{\lambda_{\text{min}}} > 100\]
Which Convergence Curve is GCR?
Residual Minimizing Krylov-subspace Algorithm can eliminate outlying eigenvalues by placing polynomial zeros directly on them.
<table>
<thead>
<tr>
<th>Dimension</th>
<th>Dense GE</th>
<th>Sparse GE</th>
<th>GCR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$O(m^3)$</td>
<td>$O(m)$</td>
<td>$O(m^2)$</td>
</tr>
<tr>
<td>2</td>
<td>$O(m^6)$</td>
<td>$O(m^3)$</td>
<td>$O(m^3)$</td>
</tr>
<tr>
<td>3</td>
<td>$O(m^9)$</td>
<td>$O(m^6)$</td>
<td>$O(m^4)$</td>
</tr>
</tbody>
</table>

GCR faster than banded GE in 2 and 3 dimensions
Could be faster, 3-D matrix only $m^3$ nonzeros.
GCR converges too slowly!
Let $M \approx \tilde{L} \tilde{U}$

Applying GCR to $\left((\tilde{L} \tilde{U})^{-1} M\right)x = (\tilde{L} \tilde{U})^{-1} b$

Use an Implicit matrix representation!

Forming $y = \left((\tilde{L} \tilde{U})^{-1} M\right)x$ is equivalent to

solving $\tilde{L} \tilde{U} y = M x$
Krylov Methods

Preconditioning
Approximate LU
Preconditioners Continued

Nonzeros in an exact LU Factorization

Filled-in LU factorization
Too expensive.

Ignore the fillin!
Factoring 2-D Grid Matrices

Generated Fill-in Makes Factorization Expensive
THROW AWAY FILL-INS!

Throw away all fill-ins
Throw away only fill-ins with small values
Throw away fill-ins produced by other fill-ins
Throw away fill-ins produced by fill-ins of other fill-ins, etc.
Summary

• Reminder about GCR
  – Residual minimizing solution
  – Krylov Subspace
  – Polynomial Connection

• Review Norms and Eigenvalues
  – Induced Norms
  – Spectral mapping theorem

• Estimating Convergence Rate
  – Chebychev Polynomials

• Preconditioners
  – Diagonal Preconditioners
  – Approximate LU preconditioners