INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
I.C.T.P., P.O. BOX 586, 34100 TRIESTE, ITALY, CABLE: CENTRATOM TRIESTE

WINTER COLLEGE ON "MULTILEVEL TECHNIQUES IN COMPUTATIONAL PHYSICS"

***Physics and Computations with Multiple Scales of Lengths* (21 January - 1 February 1991)**

H4.SMR 539/3

Preconditioned Polynomial Iterative Methods

Tom Manteuffel
University of Colorado
Denver, USA

PRECONDITIONED POLYNOMIAL ITERATIVE METHODS

A Tutorial

Tom Manteuffel

University of Colorado
at Denver

Preconditioned Polynomial Iterative Methods for the linear system

$$A\underline{x} = \underline{b}$$

consists of two separate but interrelated processes:

- Preconditioning: The construction of a linear process $\mathcal{C}$ such that

$$\mathcal{C}A\underline{x} = \mathcal{C}\underline{b}$$

  is "easier to solve."

- Polynomial Acceleration: The construction of a polynomial $p(\lambda)$ such that

$$||p(\mathcal{C}A)||$$

  is "small in some sense."

Outline of Tutorial

I. Polynomial Iterative Methods I: Chebychev-like Methods

1. General Polynomial Methods
2. Chebychev-like Methods
3. Adaptive Strategies

II. Preconditioning

1. Preconditioning/Matrix Splitting Duality
2. Classical Matrix Splitting
3. Incomplete Factorization Preconditioning
4. Equivalent Operators

III. Polynomial Iterative Methods II: Conjugate Gradient-like Methods

1. Conjugate Gradient Methods
2. Projection Methods

Polynomial Methods: General Form

$$A\underline{x} = \underline{b} \quad (N \times N) \text{ nonsingular}$$

Given $\underline{x}_0$

$$\underline{r}_0 = \underline{b} - A\underline{x}_0$$
$$\underline{x}_1 = \underline{x}_0 + \eta_{00}\underline{r}_0$$
$$\underline{r}_1 = \underline{b} - A\underline{x}_1$$
$$\underline{x}_2 = \underline{x}_1 + \eta_{11}\underline{r}_1 + \eta_{00}\underline{r}_0$$

.

.

.

$$\underline{x}_k = \underline{x}_{k-1} + \sum_{j=0}^{k-1} \eta_{kj}\underline{r}_j$$

Polynomial Methods: Error Equation

$$\underline{x} = \underline{x}$$

$$\underline{x}_k = \underline{x}_{k-1} + \sum_{j=0}^{k-1} \eta_{kj} \underline{r}_j$$

$$\underline{e}_k = \underline{e}_{k-1} - \sum_{j=0}^{k-1} \eta_{kj} \underline{r}_j$$

$$\underline{r}_j = \underline{b} - A\underline{x}_j = A(\underline{x} - \underline{x}_j) = A\underline{e}_j$$

$$\underline{e}_k = \underline{e}_{k-1} - \sum_{j=0}^{k-1} \eta_{kj} A\underline{e}_j$$

Polynomial Methods: Error Equation

Result: If $\eta_{jj} \neq 0$ for $j = 0, \ldots, k$, then

$$\underline{e}_j = p_j(A)\underline{e}_0 \qquad p_j(0) = 1$$

Proof:

$$\underline{e}_k = \underline{e}_{k-1} - \sum_{j=0}^{k-1} \eta_{kj} A\underline{e}_j$$

Induction

$$\underline{e}_k = p_{k-1}(A)\underline{e}_0 - \sum_{j=0}^{k-1} \eta_{kj} A p_j(A)\underline{e}_0$$

$$\underline{e}_k = p_k(A)\underline{e}_0$$

Residual Polynomial: $p_k(A)$

$$\underline{r}_k = p_k(A)\underline{r}_0$$

CAYLEY HAMILTON THEOREM— CHARACTERISTIC POLYNOMIAL

$$\aleph_d(A) = \alpha_d A^d + \alpha_{d-1} A^{d-1} + \cdots + \alpha_1 A + I = 0$$

Where $d \le N$. Recall

$$\underline{e}_k = p_k(A)\underline{e}_0$$

Core problem: Find $p_k(\lambda)$ such that

$$\|p_k(A)\underline{e}_0\| \cong \|\aleph_d(A)\underline{e}_0\|$$

for $k \ll d$.

NORMS

$$\|\underline{e}\|_{l_2} = <\underline{e},\underline{e}>^{1/2}$$

$$\|\underline{e}\|_{l_1} = \left(\sum_{i=1}^{N} |e_i| \right)^{1/2}$$

$$\|\underline{e}\|_\infty = \max_i |e_i|$$

$$\|\underline{e}\|_B = <B\,\underline{e},\underline{e}>^{1/2} \qquad B - HPD$$

Conjugate Gradient-like Methods: Minimize

$$\|\underline{e}_k\|_B = \|p_k(A)\underline{e}_0\|_B$$

Chebychev-like Methods (Jordan form $A = SJS^{-1}$)

$$\|\underline{e}_k\| = \|p_k(A)\underline{e}_0\| \le \|p_k(A)\|\,\|\underline{e}_0\|$$

$$\le \|S\|\,\|S^{-1}\|\,\|p_k(J)\|\,\|\underline{e}_0\|$$

$$= \kappa(S)\|p_k(J)\|\,\|\underline{e}_0\|$$

Chebychev-like Methods

If J is diagonal

$$\|p(J)\|_{l_2} = \|p(J)\|_{l_1} = \|p(J)\|_\infty = \max_{\lambda \in \Sigma(A)} |p(\lambda)|$$

- Based on minimax polynomials (for $\Sigma(A) \subseteq H$)

$$p_k^H(\lambda) : \min_{p_k(0)=1} \left[\max_{\lambda \in H} |p_k(\lambda)| \right]$$

- Require a priori or adaptive estimates of $\Sigma(A)$
- Iteration is independent of $\underline{e}_0$

($\Sigma(A)$ = spectrum of A)

Conjugate Gradient-like Methods

- Based upon Optimization or Orthogonality
- Requires little or no a priori knowledge of $\Sigma(A)$
- Iteration depends upon $\underline{e}_0$

Preconditioning

Any linear transformation that yields an equivalent problem

$$A\underline{x} = \underline{b}$$

$$CA\underline{x} = C\underline{b}$$

For example:

Normal Equations

$$C = A^*$$

Matrix Splitting

$$\begin{aligned} A &= M - N \\ C &= M^{-1} \end{aligned}$$

Multigrid Cycle

Chebychev-like Methods: Outline

A. Stationary One-Step Methods

B. Nonstationary One-Step Methods

C. The Chebychev Iteration

D. General Methods for Nonsymmetric Systems

E. Adaptive Procedures

Chebychev-like Methods: General Formula

$$\underline{x}_k = \underline{x}_{k-1} + \sum_{j=0}^{k-1} \eta_{kj}\, \underline{r}_j$$

Stationary One-step Method

$$\begin{aligned} \underline{x}_k &= \underline{x}_{k-1} + \alpha\, \underline{r}_{k-1} \\ \underline{e}_k &= \underline{e}_{k-1} - \alpha A \underline{e}_{k-1} = (I - \alpha A)\underline{e}_{k-1} \\ \underline{e}_k &= (I - \alpha A)^k \underline{e}_0 \end{aligned}$$

Stationary One-step Methods:
Asymptotic Convergence Factor

$$||\underline{e}_k|| = ||(I - \alpha A)^k \underline{e}_0|| \le ||(I - \alpha A)^k||\; ||\underline{e}_0||$$

Convergence Factor

$$\rho_k = \left(\frac{||\underline{e}_k||}{||\underline{e}_0||} \right)^{1/k} \le ||(I - \alpha A)^k||^{1/k}$$

Asymptotic Convergence Factor

$$\rho = \lim_{k \to \infty} \rho_k = \lim_{k \to \infty} ||(I - \alpha A)^k||^{1/k} = \mathcal{S}(I - \alpha A)$$

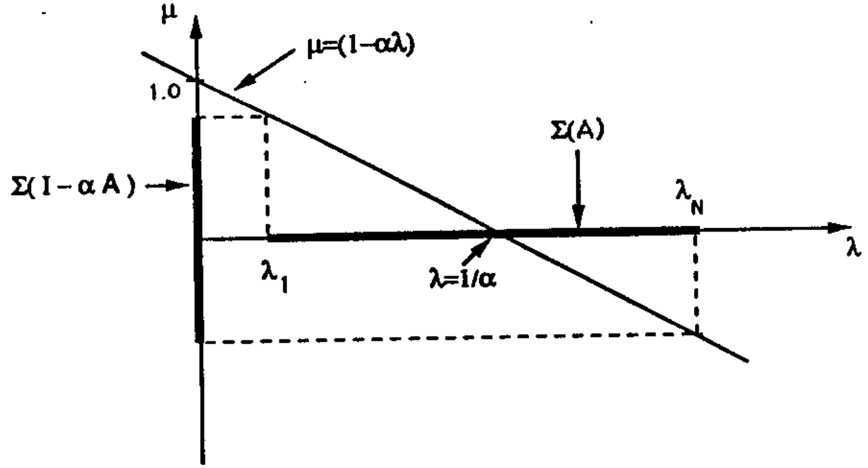
$(\mathcal{S}(I - \alpha A) =$ spectral radius of $(I - \alpha A))$

Stationary One-step Methods:
Optimal Parameter

Find α:

$$\min_{\alpha} \mathcal{S}(I-\alpha A) = \min_{\alpha} \max_{\lambda \in \Sigma(A)} |1-\alpha\lambda|$$

A-Symmetric Positive Definite (SPD)

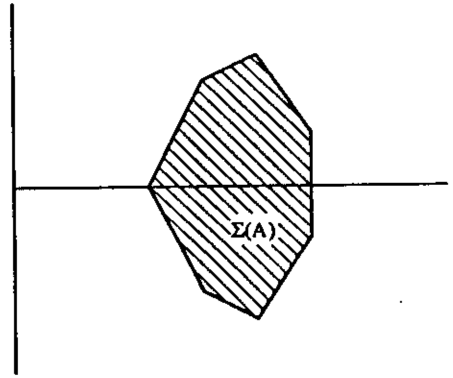


$$\mathcal{S}(I-\alpha A) = \max\{|1-\alpha\lambda_1|, |1-\alpha\lambda_N|\}$$

$$\alpha_{\text{opt}} = \frac{2}{\lambda_N+\lambda_1}, \ \rho = \frac{\lambda_N-\lambda_1}{\lambda_N+\lambda_1}$$

Stationary One-step Methods:
Optimal Parameter

A-Nonsymmetric



Find

$$\min_{\alpha} \mathcal{S}(I-\alpha A) = \min_{\alpha} \max_{\lambda \in \Sigma(A)} |1-\alpha\lambda|$$

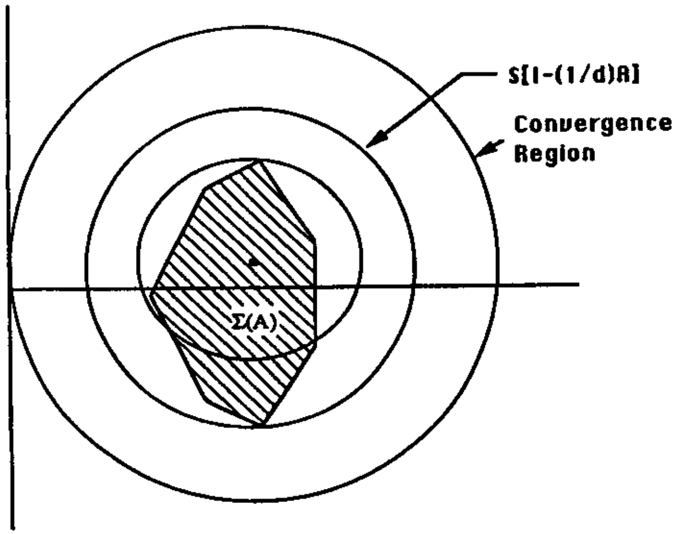
Let

$$\mu = 1-\alpha\lambda$$
$$\mu \in \Sigma(I-\alpha A) \Leftrightarrow \lambda \in \Sigma(A)$$

Stationary One-step Methods:

Optimal Parameter

Given $\alpha = \frac{1}{d}$, find $\mathcal{S}(I - \frac{1}{d}A)$



Level Lines of $\mu = (1 - \alpha\lambda) = (1 - \frac{1}{d}\lambda)$

$$\{\lambda / \, |\frac{d-\lambda}{d}| = r\}$$

Convergence Region

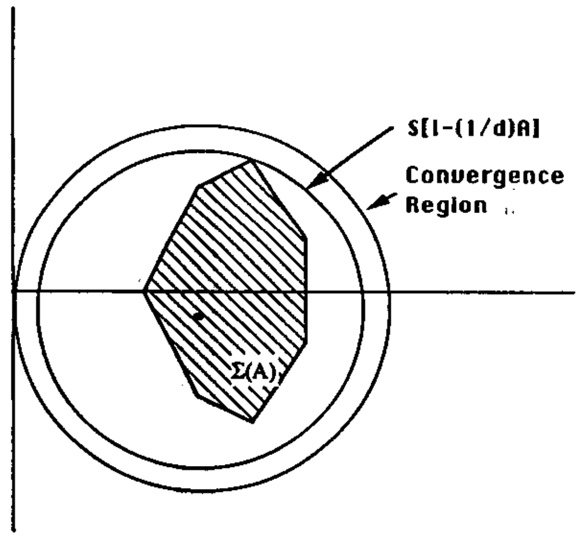
$$\{\lambda / \, |\frac{d-\lambda}{d}| < 1\}$$

Stationary One-step Methods:

Optimal Parameter

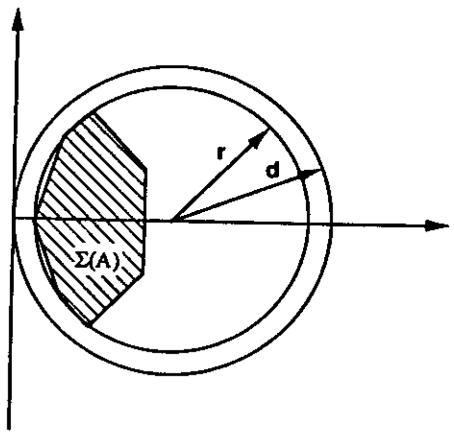
Let

$$H(A) = \{ \text{ convex hull of } \Sigma(A)\}$$



$$\mathcal{S}(I - \alpha A) = \max_{\lambda \in H(A)} |1 - \alpha\lambda|$$

Stationary One-step Methods



- Convergence possible $\Leftrightarrow \Sigma(A)$ can be contained in a circle that does not include the origin
- α_{opt} can be calculated from $H(A)$
- $\rho_{\text{opt}} = \frac{r}{d}$

Nonstationary One-step Methods

General Formula

$$\underline{x}_k = \underline{x}_{k-1} + \sum_{j=0}^{k-1} \eta_{kj}\, \underline{r}_j$$

Nonstationary One-step Method

$$\begin{aligned}
\underline{x}_k &= \underline{x}_{k-1} + \alpha_{k-1}\underline{r}_{k-1} \\
\underline{e}_k &= \underline{e}_{k-1} - \alpha_{k-1}A\underline{e}_{k-1} = (I - \alpha_{k-1}A)\underline{e}_{k-1} \\
\underline{e}_k &= \Big(\prod_{j=0}^{k-1}(I - \alpha_j A)\Big)\underline{e}_0 = p_k(A)\underline{e}_0
\end{aligned}$$

$$||\underline{e}_k|| \leq ||p_k(A)||\; ||\underline{e}_0||$$

Nonstationary One-step Methods

$$||\underset{\sim}{e}_k|| \leq ||p_k(A)||\, ||\underset{\sim}{e}_0||$$

Let $A = SJS^{-1}$ be the Jordan Decomposition

$$||p_k(A)|| = ||Sp_k(J)S^{-1}|| \leq ||S||\, ||S^{-1}||\, ||p_k(J)||$$

$$\kappa(S) = ||S||\, ||S^{-1}||$$

If J diagonal

$$||p_k(J)|| = \max_{\lambda \in \Sigma(A)} |p_k(\lambda)| = \mathcal{S}\,(p_k(A))$$

Choose $p_k(\lambda)$

$$\min_{p_k(0)=1} [\max_{\lambda \in \Sigma(A)} |p_k(\lambda)|]$$

Nonstationary One-step Methods

Convergence Factor

$$\rho_k = ||p_k(A)||^{1/k}$$

Result: Asymptotic Convergence Factor

$$\rho = \lim_{k \to \infty} \rho_k = \max_{\lambda \in \Sigma(A)} |p_k(\lambda)|^{1/k} = \mathcal{S}\,(p_k(A))^{1/k}$$

Proof:

$$\mathcal{S}\,(p_k(A)) \leq ||p_k(A)|| \leq \kappa(S)\, ||p_k(J)||$$

$$\mathcal{S}\,(p_k(A))^{1/k} \leq ||p_k(A)||^{1/k} \leq \kappa(S)^{1/k}\, ||p_k(J)||^{1/k}$$

$$\kappa(S)^{1/k} \to 1$$

$$||p_k(J)||^{1/k} \to \max_{\lambda \in \Sigma(A)} |p_k(\lambda)|^{1/k} = \mathcal{S}\,(p_k(A))^{1/k}$$

Nonstationary One-step Methods

Want to find $p_k(\lambda)$:

$$\min_{p_k(0)=1}\left[\max_{\lambda\in\Sigma(A)}|p_k(\lambda)|\right]$$

Use instead H such that $\Sigma(A)\subseteq H$:

$$p_k^H(\lambda):\ \min_{p_k(0)=1}\left[\max_{\lambda\in H}|p_k(\lambda)|\right]$$

Convergence Factor for the set H

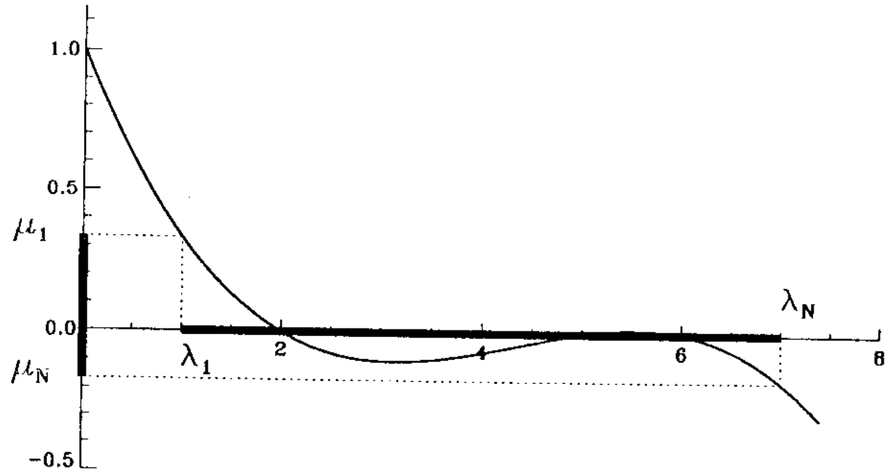
$$\rho_k(H)=\min_{p_k(0)=1}\left[\max_{\lambda\in H}|p_k(\lambda)|^{1/k}\right]$$

Asymptotic Convergence Factor for the set H

$$\rho_\infty(H)=\lim_{k\to\infty}\rho_k(H)$$

Nonstationary One-step Methods

A-Symmetric Positive Definite



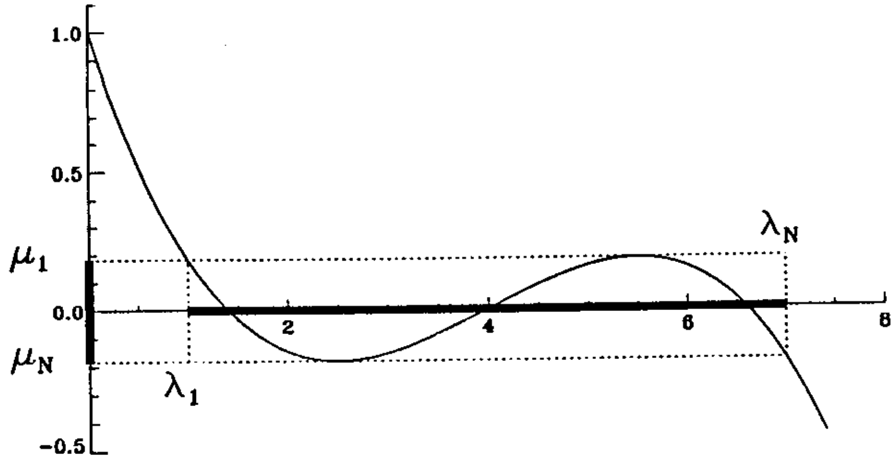
Minimax Polynomial for $H=[\lambda_1,\lambda_N]$

$$p_k^H(\lambda)=\frac{T_k\left[\frac{\lambda_N+\lambda_1-2\lambda}{\lambda_N-\lambda_1}\right]}{T_k\left[\frac{\lambda_N+\lambda_1}{\lambda_N-\lambda_1}\right]}$$

Chebychev Polynomials of the First Kind

$$T_k(\lambda)=\cos\left(k\cos^{-1}(\lambda)\right)$$

Nonstationary One-step Methods



$$p_k^H(\lambda) = \frac{T_k[\frac{\lambda_N+\lambda_1-2\lambda}{\lambda_N-\lambda_1}]}{T_k[\frac{\lambda_N+\lambda_1}{\lambda_N-\lambda_1}]}$$

$$\max_{\lambda\in H}|p_k^H(\lambda)| \le 2\left(\frac{\sqrt{\lambda_N/\lambda_1}-1}{\sqrt{\lambda_N/\lambda_1}+1}\right)^k$$

$$\rho_\infty(H) = \left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)$$

$$\kappa(A) = \frac{\lambda_N}{\lambda_1} = \text{condition number of } A$$

Nonstationary One-step Methods

Comparison with stationary One-step Method for A SPD.

- Stationary

$$\frac{||\underline{e}_k||}{||\underline{e}_0||} \le \left(\frac{\lambda_N-\lambda_1}{\lambda_N+\lambda_1}\right)^k = \left(\frac{\kappa(A)-1}{\kappa(A)+1}\right)^k$$

Iterations

$$\varepsilon \ge \left(\frac{\kappa(A)-1}{\kappa(A)+1}\right)^k \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{2}\kappa(A)$$

- Nonstationary

$$\frac{||\underline{e}_k||}{||\underline{e}_0||} \le 2\left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)^k$$

Iterations

$$\varepsilon \ge 2\left(\frac{\sqrt{\kappa(A)}-1}{\sqrt{\kappa(A)}+1}\right)^k \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{2}\sqrt{\kappa(A)}$$

Nonstationary One-step Methods

Implementation $H = [\lambda_1, \lambda_N]$

$$p_k^H(\lambda) = \prod_{j=1}^{k}[1 - \frac{1}{\mu_j}\lambda]$$

$$\underline{x}_i = \underline{x}_{i-1} + \frac{1}{\mu_i}\underline{r}_{i-1}$$

$$\underline{e}_i = \underline{e}_{i-1} - \frac{1}{\mu_i}A\underline{e}_{i-1} = (I - \frac{1}{\mu_i}A)\underline{e}_{i-1}$$

$$\underline{e}_k = p_k^H(A)\underline{e}_0$$

- Only optimal at step k
- Order of roots important

Chebychev Iteration

Recursion for Chebychev Polynomials

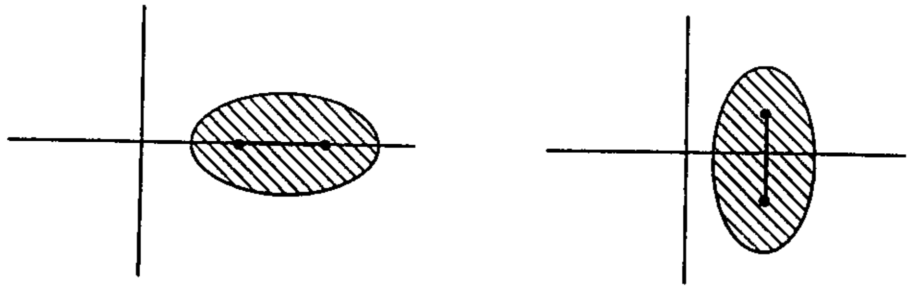
$$T_0(\lambda) = 1$$
$$T_1(\lambda) = \lambda$$
$$T_{k+1}(\lambda) = 2\lambda T_k(\lambda) - T_{k-1}(\lambda)$$

Two-step method

$$\underline{x}_k = \underline{x}_{k-1} + \underline{\Delta}_{k-1}$$
$$\underline{\Delta}_k = \alpha_k \underline{r}_k + \beta_k \underline{\Delta}_{k-1}$$
$$\underline{e}_k = p_k^{(H)}(A)\underline{e}_0 \text{ for every } k$$

- Three term recursion
- Optimal at every step

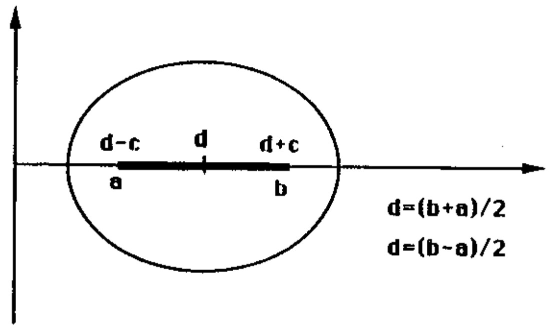
Chebychev Iteration: Nonsymmetric A



Real Foci　　　　Complex Foci

- For foci real or complex conjugate pair Chebychev polynomials are optimal for k sufficiently large
- For any ellipse E with $0 \notin E$ the Chebychev polynomials are asymptotically optimal

Chebychev Iteration: Nonsymmetric A



$$p_k(\lambda) = \frac{T_k[\frac{b+a-\alpha\lambda}{b-a}]}{T_k[\frac{b+a}{b-a}]} = \frac{T_k[\frac{d-\lambda}{c}]}{T_k[\frac{d}{c}]}$$

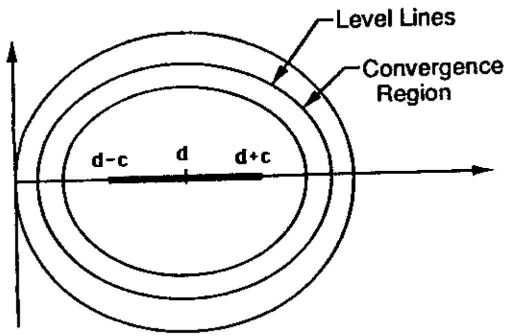
Asymptotic Form

$$p_k(\lambda) = R(\lambda)^k Q_k(\lambda)$$

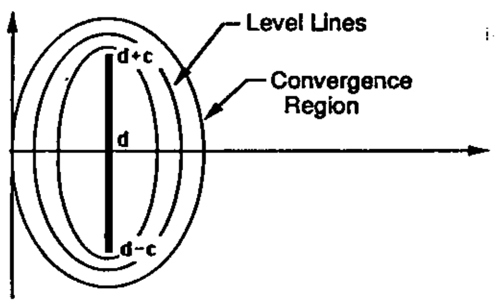
$$|Q_k(\lambda)| \leq 2,\ Q_k(\lambda) \to 1 \text{ quickly}$$

$$R(\lambda) = \frac{(d-\lambda) + \left((d-\lambda)^2 - c^2\right)^{1/2}}{d + (d^2 - c^2)^{1/2}}$$

Chebychev Iteration: Nonsymmetric A



Real Foci

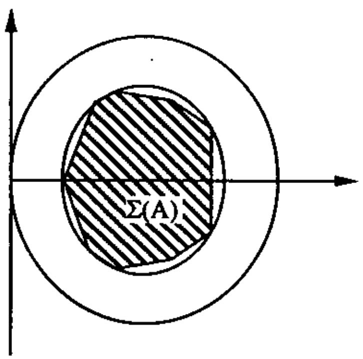


Complex Foci

The level lines of $|R(\lambda)|$ are the confocal family of ellipses with foci $d \pm c$

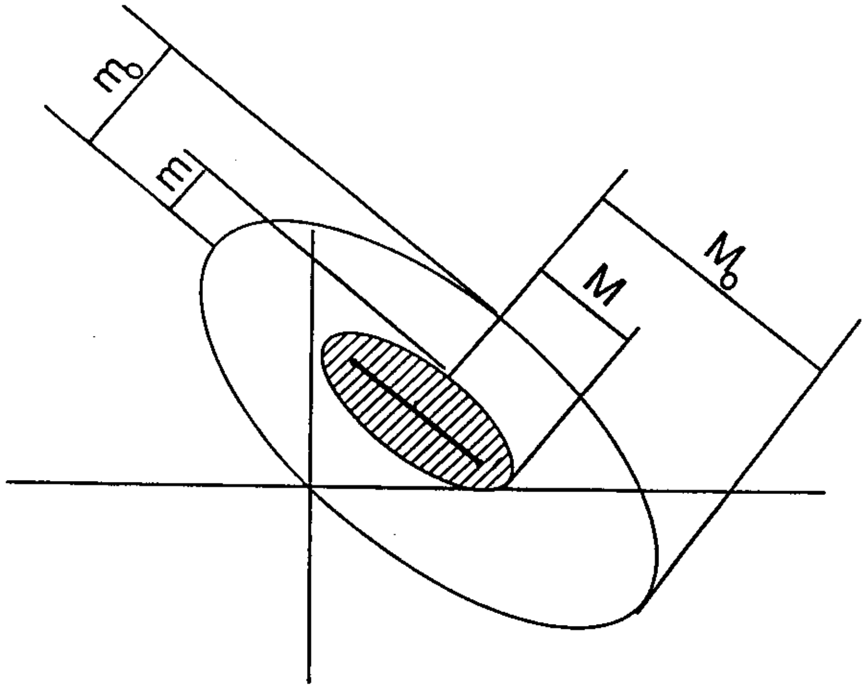
Chebychev Iteration: Nonsymmetric A

- If $\Sigma(A) = E$, then the corresponding Chebychev polynomials are asymptotically optimal
- If $\Sigma(A)$ not an ellipse, choose the "best" ellipse that encloses $\Sigma(A)$



NONSYMMETRIC A

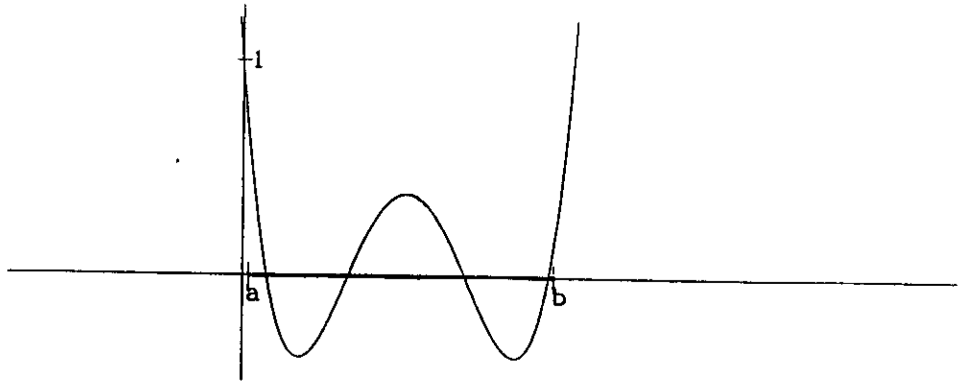
Ellipses



$$\rho_\infty(E) = \left[\frac{m + M}{m_0 + M_0}\right]$$

SYMMETRIC POSITIVE DEFINITE

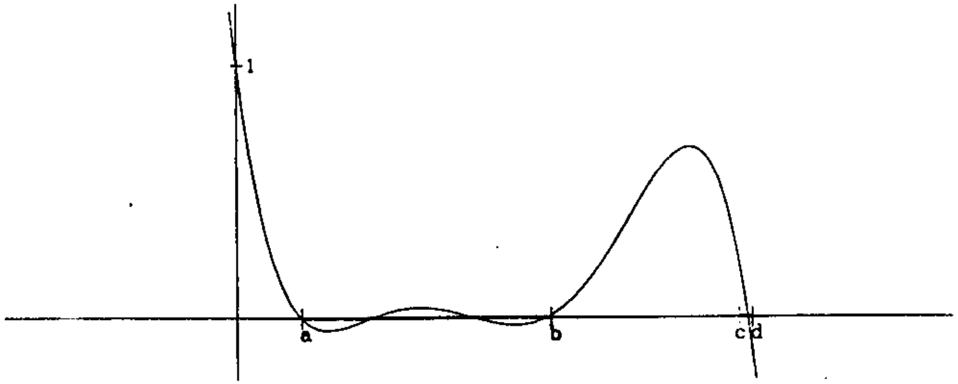
Single Interval



$$H = [a, b] \; , \quad \rho_\infty(H) = \left[\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}\right]$$

SYMMETRIC POSITIVE DEFINITE

Multiple Intervals

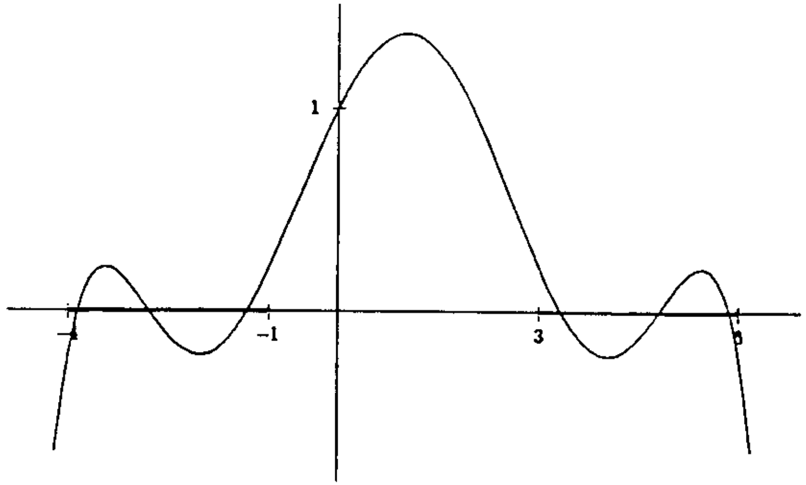


$$H = [a, b] \cup [c, d]$$

- $\rho_\infty(H)$ indirectly dependent on $\kappa(A)$

SYMMETRIC INDEFINITE A

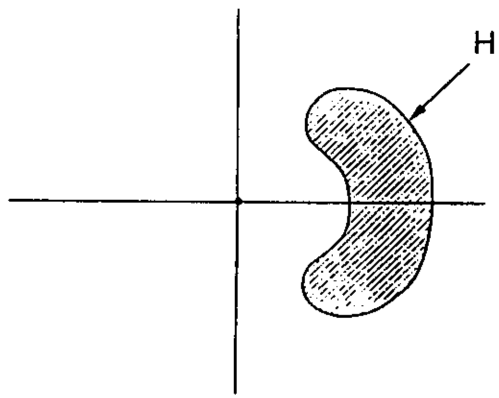
Multiply Connected H *(Roloff, Deboor/Rice, Grcar)*



$$H = [-4, -1] \cup [3, 6]$$

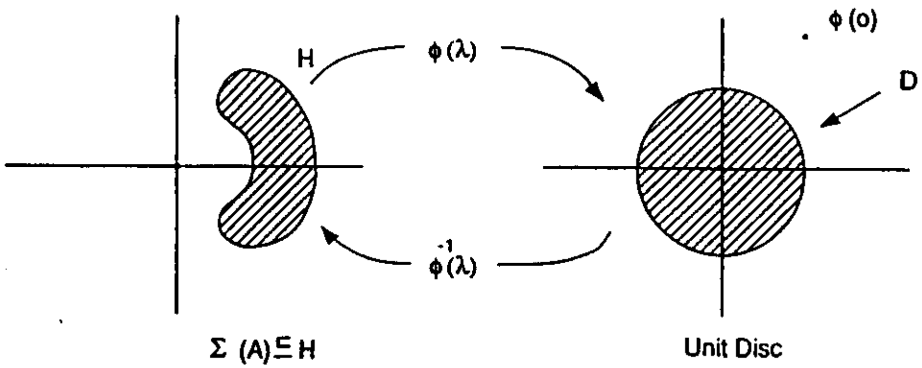
- $\rho_\infty(H)$ indirectly dependent on $\kappa(A)$

NONSYMMETRIC A



$\Sigma(A) \subseteq H$

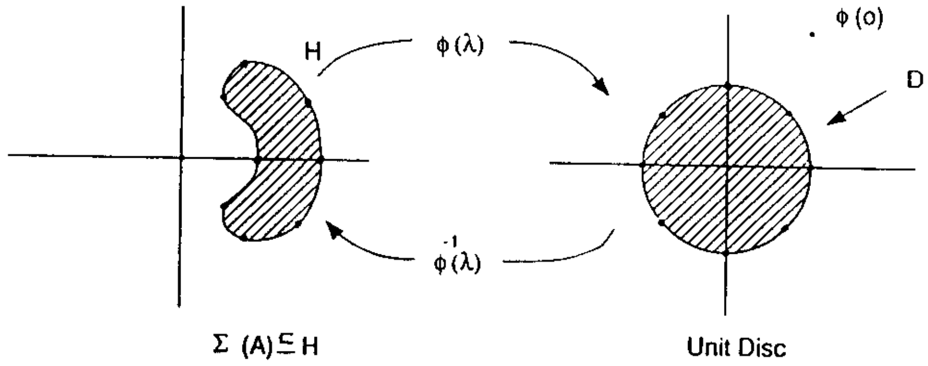
We seek $p_k^H(\lambda)$, Easier to find $\rho_\infty(H)$



$\phi(\lambda)$ conformal , $\phi(\infty) = \infty$

NONSYMMETRIC A

General Connected H



- Asymptotically Optimal Polynomials *(Reichel, Tel-Ezer,*

$$\tilde{p}_k(\lambda) = \prod_{j=1}^{k} \left[1 - \frac{1}{\mu_j} \lambda \right]$$

$$\mu_j = \phi^{-1}\left[e^{i\left(\frac{2\Pi j}{k}\right)} \right]$$

ALGORITHM: Given H

Choose k

Choose $\tilde{p}_k(\lambda) \cong p_k^H(\lambda)$

Write

$$\tilde{p}_k(\lambda) = \prod_{j=1}^{k} \left(1 - \frac{1}{\mu_j}\lambda\right)$$

Perform Steps

$$\underline{x}_i = \underline{x}_{i-1} + \frac{1}{\mu_j}\underline{r}_{j-1} \qquad j = 1, \ldots, k$$

After k steps

$$\underline{e}_k = \tilde{p}_k(A)\underline{e}_0$$

FABER POLYNOMIALS

- Laurent Expansion $\quad \lambda_0 \in H$

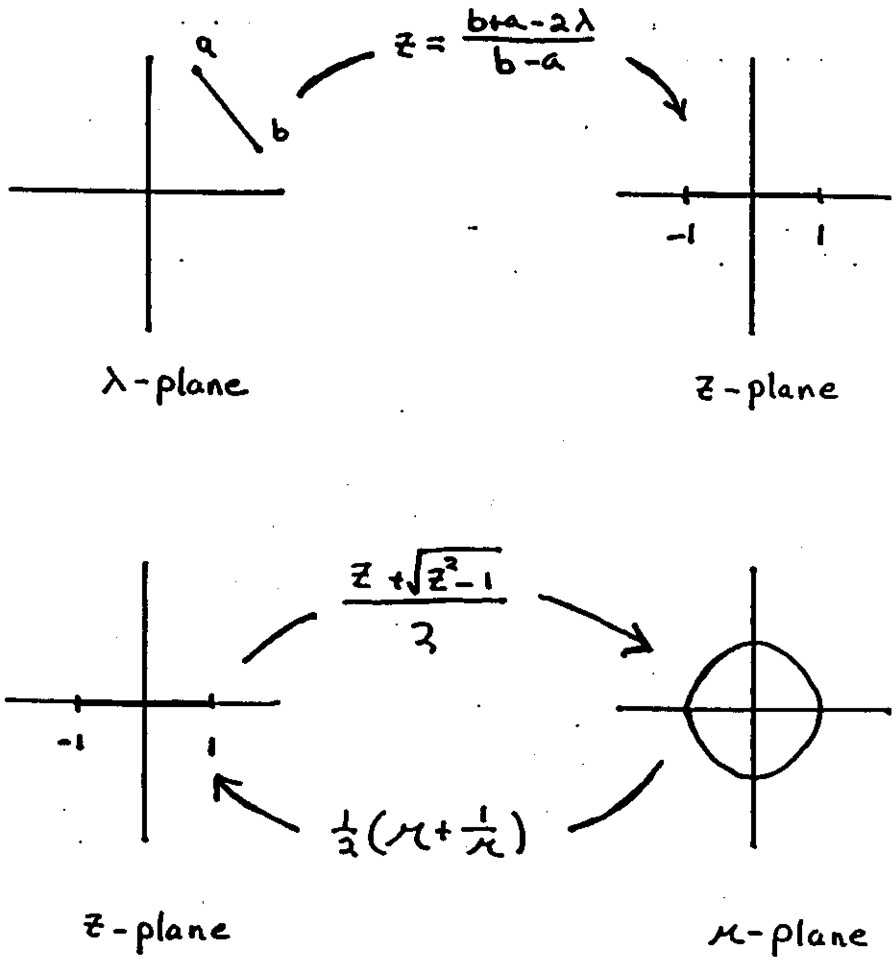
$$\phi(\lambda) = \frac{(\lambda - \lambda_0)}{c} + a_0 + \frac{a_1}{(\lambda - \lambda_0)} + \frac{a_2}{(\lambda - \lambda_0)^2} + \cdots$$

$$\phi(\lambda)^k = \left[\left[\frac{(\lambda - \lambda_0)}{c}\right]^k + \cdots + \gamma_0\right] + \frac{\gamma_1}{\lambda} + \frac{\gamma_2}{\lambda^2} + \cdots$$

$$F_k(\lambda) = \left[\left[\frac{(\lambda - \lambda_0)}{c}\right]^k + \cdots + \gamma_0\right]$$

- Asymptotically Optimal
- Recursion (not short)

CHEBYCHEV POLYNOMIALS



CHEBYCHEV POLYNOMIALS

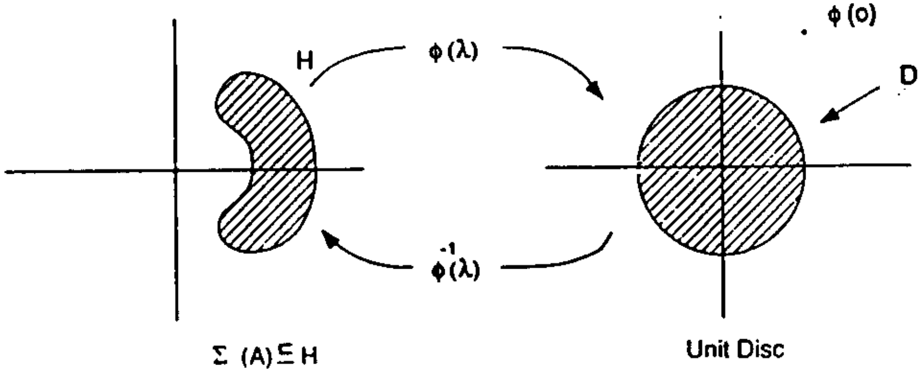
- Chebychev Polynomials are Faber Polynomials for Ellipses
- Three Term Recursion

$$\underline{x}_i = \underline{x}_{i-1} + \underline{\Delta}_{i-1}$$

$$\underline{r}_i = \underline{b} - A\,\underline{x}_i$$

$$\underline{\Delta}_i = \alpha_i \underline{r}_i + \beta_i \underline{\Delta}_{i-1}$$

SHORT RECURSION



Laurent Expansion

$$\phi^{-1}(\lambda) = \beta_{-1}\lambda + \beta_0 + \frac{\beta_1}{\lambda} + \frac{\beta_2}{\lambda} + \cdots$$

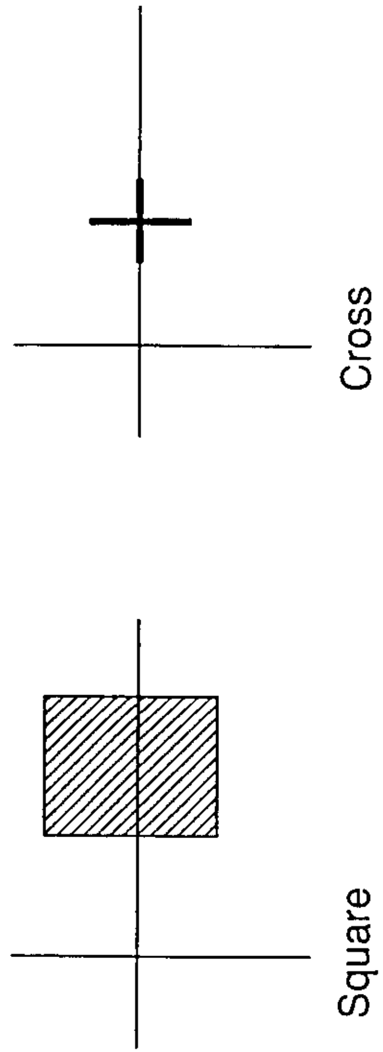
Finite Expansion Yields Finite Recursion *(Curtiss)*

S-STEP METHODS *(Eiermann/Niethammer/Varga)*

$$\underline{x}_i = \underline{x}_{i-1} + \underline{\Delta}_{i-1}$$

$$\underline{\Delta}_i = \alpha_i \underline{r}_i + \sum_{j=1}^{S} \beta_{ij} \underline{\Delta}_{i-j}$$

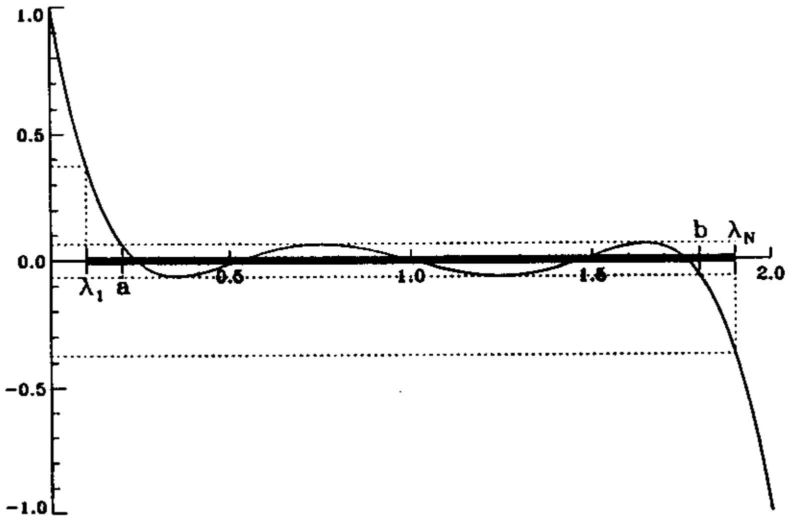
Asymptotically Optimal Over Regions Bounded by Lemniscates



Adaptive Strategy: Chebychev Iteration

- $\Sigma(A)$ seldom known
- Use information from iteration to estimate $\Sigma(A)$

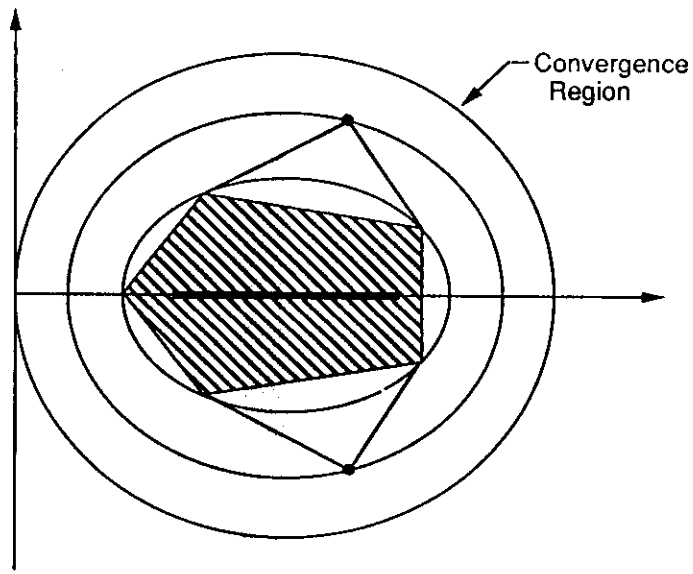
A Symmetric Positive Definite



- Residual becomes rich in eigen components of eigenvalues outside $[a, b]$

Adaptive Procedure: Chebychev Iteration

A Nonsymmetric



Residual becomes rich in eigen components associated with eigenvalues on outer-most ellipses

ADAPTIVE PROCEDURES

- $\Sigma(A)$ Seldom known
- Use information from iteration

Field of Values

$$\mathbf{F}(A) = \left\{ \lambda : \lambda = \frac{<A\underline{x},\underline{x}>}{<\underline{x},\underline{x}>} \text{ some } \underline{x} \right\}$$

Convex Hull

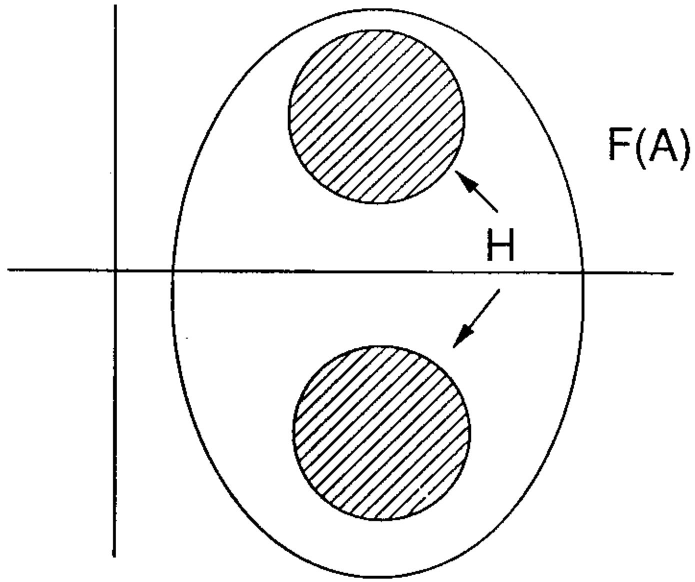
$$\mathbf{H}(A) = \{\text{Convex Hull of } \Sigma(A)\}$$

Result:

$$\mathbf{H}(A) \subseteq F(A)$$

$$\mathbf{F}(A) \backslash \mathbf{H}(A) \text{ measure of Normality}$$

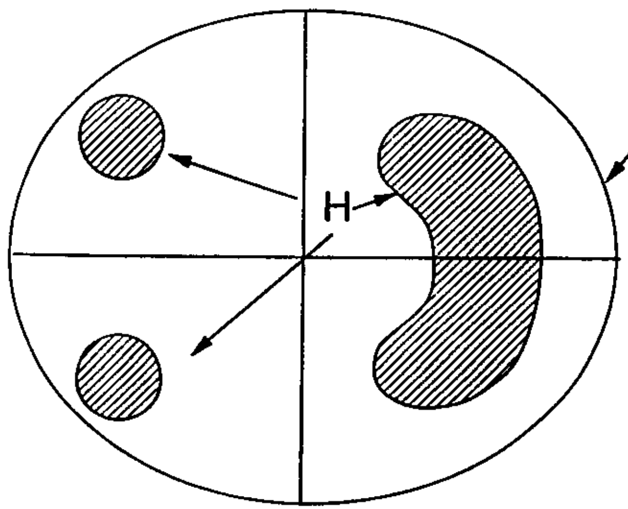
ADAPTIVE PROCEDURES: Definite Problems



$$\Sigma(A) \subseteq H \subseteq F(A)$$

Eigenvalue estimates $\lambda_e \in F(A)$

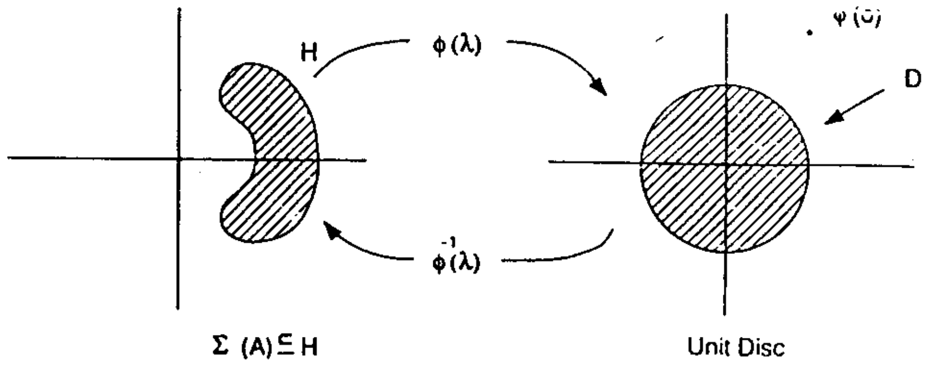
ADAPTIVE PROCEDURES: Indefinite Problems



$\Sigma(A) \subseteq H \subseteq F(A)$

Eigenvalue estimates $\lambda_e \in F(A)$

ADAPTIVE PROCEDURES



$$\hat{p}_k(\lambda) = \frac{F_k(\lambda)}{F_0(\lambda)} = (\phi(\lambda))^k Q_k(\lambda)$$

$$|Q_k(\lambda)| \to 1$$

Thus,

$$\underline{r}_k = \hat{p}_k(A)\underline{r}_0 \cong (\phi(A))^k \underline{r}_0$$

$$\{\underline{r}_k, \underline{r}_{k+1}, \ldots, \underline{r}_{k+s}\} \cong \{\underline{r}_k, \phi(A)\underline{r}_k, \ldots, \phi(A)^{s-1}\underline{r}_k\}$$

Chebychev-like Methods: Summary

- Based on Polynomials on $\sum(A)$
- One-Step Methods: Choose Optimal Circle
- Chebychev Method: Choose Optimal Ellipse
- Faber Polynomials: Asymptotically Optimal
- Adaptive Procedures: Field of Values

Preconditioned Polynomial Iterative Methods

II. Preconditioning

TOM MANTEUFFEL
UNIVERSITY OF COLORADO AT DENVER

Preconditioning

Given the system

$$A\underline{x} = \underline{b}$$

A preconditioning is any nonsingular linear process C such that the equivalent system

$$CA\underline{x} = c\underline{b}$$

is in some sense easier to solve.

Preconditioning

Outline

A. Preconditioning/Matrix Splitting

B. Model Problem

C. Classical Splittings

1. Jacobi
2. Gauss-Seidel
3. SOR
4. SSOR

D. Incomplete Factorization

1. IC (Incomplete Cholesky)
2. MIC (Modified Incomplete Cholesky)

E. Equivalent Operators

Preconditioning/Matrix Splitting

Given

$$A\underline{x} = \underline{b}$$

Matrix splitting

$$A = M - N$$

Write

$$M\underline{x} = N\underline{x} + \underline{b}$$

$$M\underline{x}_k = N\underline{x}_{k-1} + \underline{b}$$

Error equation

$$M\underline{e}_k = N\underline{e}_{k-1}$$

$$\underline{e}_k = M^{-1}N\underline{e}_{k-1}$$

Preconditioning/Matrix Splitting

Reformulate Matrix Splitting

$$M\underline{x}_k = N\underline{x}_{k-1} + \underline{b}$$

$$M\underline{x}_k = M\underline{x}_{k-1} + (\underline{b} - (M - N)\underline{x}_{k-1})$$

$$M\underline{x}_k = M\underline{x}_{k-1} + \underline{r}_k$$

$$\underline{x}_k = \underline{x}_{k-1} + M^{-1}\underline{r}_k$$

Stationary One-step Method

$$M^{-1}A\underline{x} = M^{-1}\underline{b}$$

$$\underline{x}_k = \underline{x}_{k-1} + \alpha M^{-1}\underline{r}_k$$

Matrix splitting is equivalent to the simplest stationary one-step method applied to the system preconditioned by M^{-1}.

Model Problem

$$-(u_{xx} + u_{yy}) = f \qquad (x,y) \in [0,1] \times [0,1]$$

$$u(x,0) = u(x,1) = 0$$

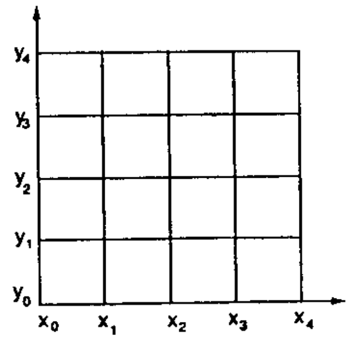
$$u(0,y) = u(1,y) = 0$$

Centered Difference Formula

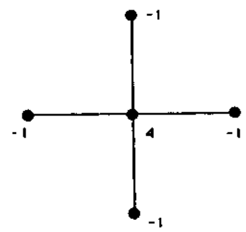
$$\frac{1}{h^2}(-u(x-h,y) + 2u(x,y) - u(x+h,y)) = -u_{xx}(x,y) + O(h^2)$$

$$\frac{1}{h^2}(-u(x,y-h) + 2u(x,y) - u(x,y+h)) = -u_{yy}(x,y) + O(h^2)$$

Mesh



Stencil



Model Problem

Matrix Problem

$$u_{ij} \cong u(x_i, y_j)$$

$$f_{ij} = f(x_i, y_j)$$

$$\begin{bmatrix} 4 & -1 & 0 & -1 & 0 & 0 & & & \\ -1 & 4 & -1 & 0 & -1 & 0 & & & \\ 0 & -1 & 4 & 0 & 0 & -1 & & & \\ -1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\ & & & -1 & 0 & 0 & 4 & -1 & 0 \\ & & & 0 & -1 & 0 & -1 & 4 & -1 \\ & & & 0 & 0 & -1 & 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} u_{11} \\ u_{12} \\ u_{13} \\ u_{21} \\ u_{22} \\ u_{23} \\ u_{31} \\ u_{32} \\ u_{33} \end{bmatrix} = h^2 \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix}$$

$$A\underline{u} = \underline{f}$$

Model Problem

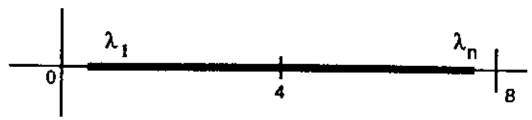
Eigenvector Decomposition ($h = \frac{1}{n+1}$)

$$A\underline{v}_{k\ell} = \lambda_{k\ell}\underline{v}_{k\ell} \quad k, \ell = 1, \cdots, n \quad (N = n^2)$$

$$\lambda_{k\ell} = (2 - 2\cos(\frac{k\pi}{n+1})) + (2 - 2\cos(\frac{\ell\pi}{n+1}))$$

$$= 4(\sin^2(\frac{k\pi}{2(n+1)}) + \sin^2(\frac{\ell\pi}{2(n+1)}))$$

$$(\underline{v}_{k\ell})_{ij} = \sin(\frac{k\pi i}{n+1})\sin(\frac{\ell\pi j}{n+1})$$



$\lambda_1 = 8\sin^2\left(\frac{\pi}{2(n+1)}\right)$ $\quad \lambda_n = 8 - \lambda$

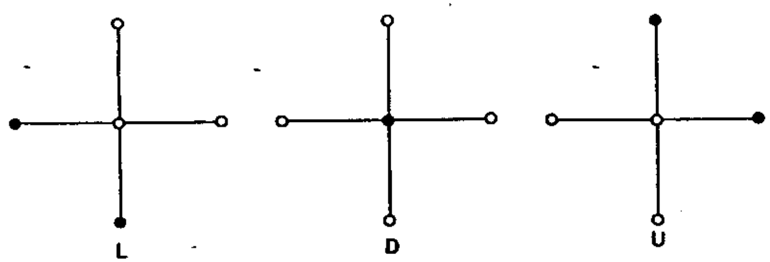
Model Problem

Write

$$A = D - L - U$$

D	Diagonal
L	Lower Triangular
U	Upper Triangular

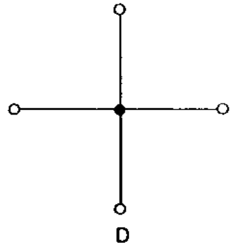
Stencil



Classical Matrix Splittings: Jacobi

Splitting

$$A = M - N$$
$$M = D$$
$$N = L + U$$

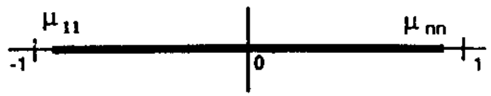


Iteration

$$D\underline{x}_k = (L+U)\underline{x}_{k-1} + \underline{b}$$
$$D\underline{e}_k = (L+U)\underline{e}_{k-1}$$
$$\underline{e}_k = D^{-1}(L+U)\underline{e}_{k-1}$$

Spectrum

$$D^{-1}(L+U)\underline{v}_{k\ell} = \mu_{k\ell}\underline{v}_{k\ell}$$
$$\mu_{k\ell} = \frac{1}{2}\left(\cos\left(\frac{k\pi}{n+1}\right) + \cos\left(\frac{\ell\pi}{n+1}\right)\right)$$



$\mu_{11} = 1 - 2\sin^2\left(\frac{\pi}{2(n+1)}\right)$ $\mu_{nn} = -\mu_{11}$

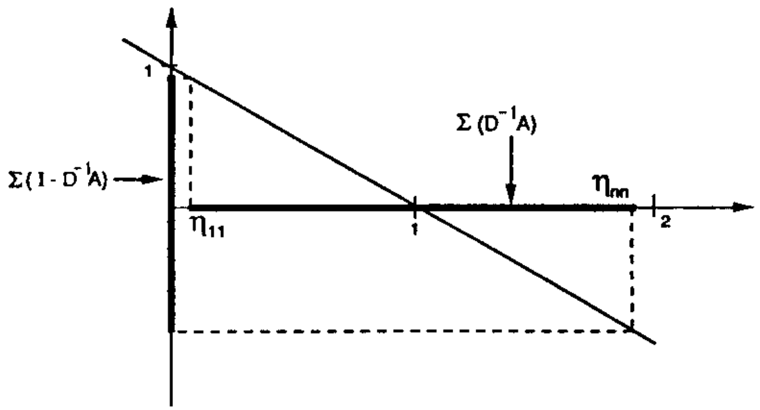
Classical Matrix Splittings: Jacobi

Stationary One-step Method

$$\underline{x}_k = \underline{x}_{k-1} + \alpha D^{-1} \underline{r}_{k-1}$$

$$\underline{e}_k = (I - \alpha D^{-1} A)\underline{e}_{k-1}$$
$$= (I - \alpha(I - D^{-1}(L+U)))\underline{e}_{k-1}$$

Optimum $\alpha = 1.0$



$$\rho = \frac{\eta_{nn} - \eta_{11}}{\eta_{nn} + \eta_{11}} = 1 - 2\sin^2\left(\frac{\pi}{2(n+1)}\right)$$

$$\varepsilon = \rho^K \Rightarrow K \cong \log\left(\frac{1}{\varepsilon}\right)\frac{1}{2}\left(\frac{\eta_{nn}}{\eta_{11}}\right) \cong \log\left(\frac{1}{\varepsilon}\right)\left(\frac{2}{\pi^2}\right)n^2$$

Classical Matrix Splittings: Jacobi

Preconditioning

$$A\underline{x} = \underline{b}$$

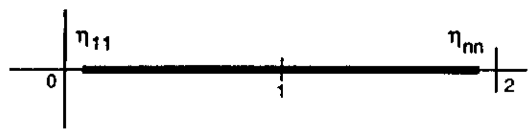
$$M = D$$

$$D^{-1}A\underline{x} = D^{-1}\underline{b}$$

$$\Sigma(D^{-1}A) = \Sigma(I - D^{-1}(L+U))$$

$$D^{-1}A\underline{v}_{k\ell} = \eta_{k\ell}\underline{v}_{k\ell}$$

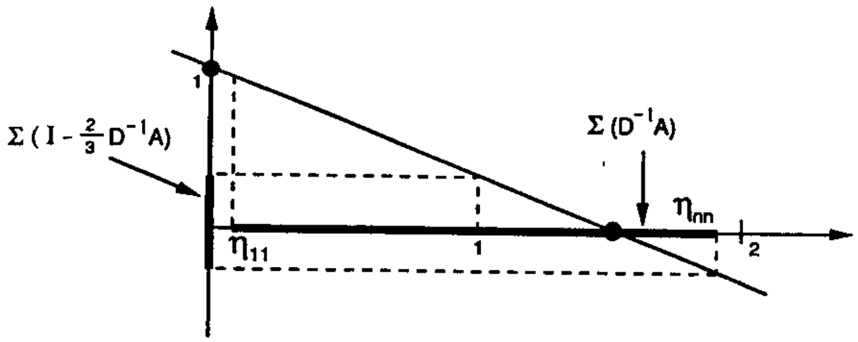
$$\eta_{k\ell} = 1 - \frac{1}{2}\left(\cos\left(\frac{k\pi}{n+1}\right) + \cos\left(\frac{\ell\pi}{n+1}\right)\right)$$



$\eta_{11} = 2\sin^2\left(\frac{\pi}{2(n+1)}\right)$ $\quad \eta_{nn} = 2 - \eta_{11}$

Classical Matrix Splittings: Jacobi

Dampened Jacobi $\qquad \alpha = \frac{2}{3}$



$$S(I - \frac{2}{3}D^{-1}A) = 1 - \frac{4}{3}\sin^2(\frac{\pi}{2(n+1)})$$

$$S(\text{High Frequencies}) = \frac{1}{3}$$

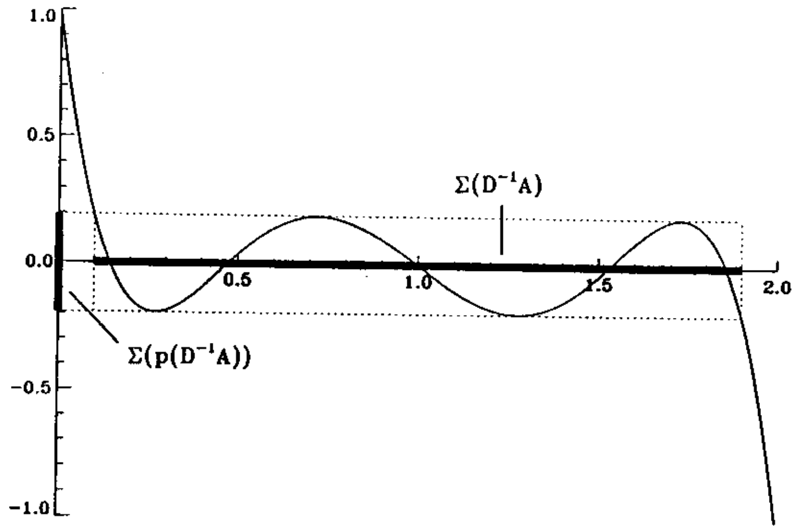
Classical Matrix Splittings: Jacobi

Chebychev Iteration (or CG)

$$\underline{x}_k = \underline{x}_{k-1} + \underline{\Delta}_{k-1}$$

$$\underline{\Delta}_k = \alpha_k \underline{r}_k + \beta_k \underline{\Delta}_{k-1}$$

$$\underline{e}_k = p_k(D^{-1}A)\underline{e}_0$$



$$\rho = \left(\frac{\sqrt{\eta_{nn}/n_{11}} - 1}{\sqrt{\eta_{nn}/\eta_{11}} + 1} \right)$$

$$\varepsilon = \rho^K \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{2}\sqrt{\frac{\eta_{nn}}{\eta_{11}}} \cong \log(\frac{1}{\varepsilon})\frac{1}{\pi}n$$

Classical Matrix Splittings: Gauss-Seidel

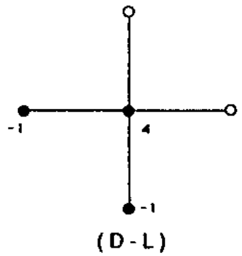
Splitting

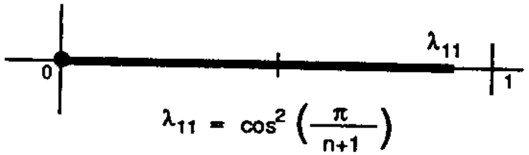
$$A = D - (L + U)$$

$$M = (D - L)$$

$$N = U$$

$$\Sigma(M^{-1}N) = \Sigma((D-L)^{-1}U)$$





For $\mu_{k\ell} \in \sum(D^{-1}(L+U))$, $\mu_{k\ell} > 0$

$$\lambda_{k\ell} = \mu_{k\ell}^2 = \frac{1}{4}(\cos(\frac{k\pi}{n+1}) + \cos(\frac{\ell\pi}{n+1}))^2$$

$$(\underline{v}_{k\ell})_{ij} = (\mu_{k\ell})^i \sin(\frac{k\pi i}{n+1})(\mu_{k\ell})^j \sin(\frac{\ell\pi j}{n+1})$$

Classical Matrix Splittings: Gauss-Seidel

Preconditioning

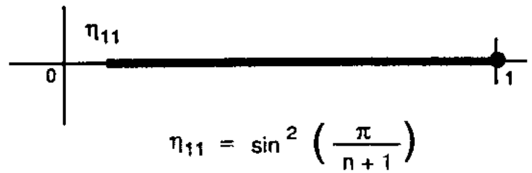
$$A\underline{x} = \underline{b}$$

$$M = (D - L)$$

$$(D-L)^{-1}A\underline{x} = (D-L)^{-1}\underline{b}$$

$$\Sigma(M^{-1}A) = \Sigma(I - (D-L)^{-1}U)$$

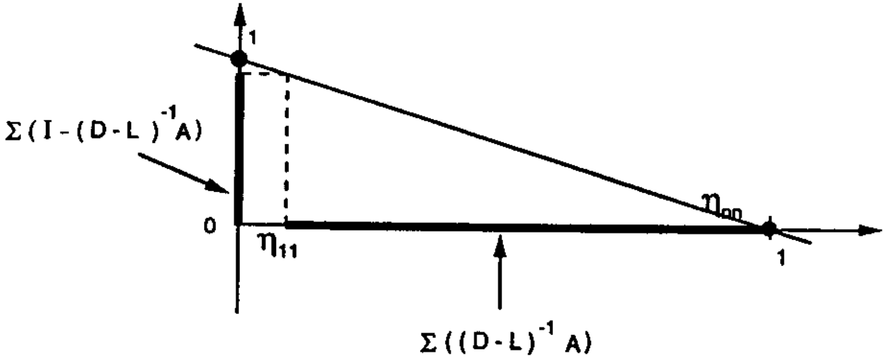
$$\eta_{k\ell} = 1 - \lambda_{k\ell} = 1 - \frac{1}{4}(\cos(\frac{k\pi}{n+1}) + \cos(\frac{\ell\pi}{n+1}))^2$$



$$\eta_{11} = 1 - (\cos(\frac{\pi}{n+1}))^2 \cong \sin^2(\frac{\pi}{n+1})$$

Classical Matrix Splittings: Gauss-Seidel

Stationary One-step Method $\quad \alpha = 1$



$$\rho = S(I - M^{-1}A) = 1 - \sin^2(\frac{\pi}{n+1})$$

$$\varepsilon = \rho^K \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{\pi^2}n^2$$

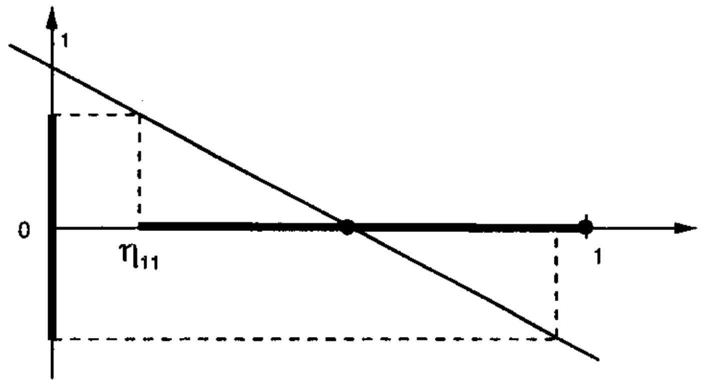
Recall

Jacobi (1-step) $\quad K \cong \log(\frac{1}{\varepsilon})\frac{2}{\pi^2}n^2$

Jacobi (Chebychev) $\quad K \cong \log(\frac{1}{\varepsilon})(\frac{1}{\pi})n$

Classical Matrix Splittings: Gauss-Seidel

Optimal $\alpha = \frac{2}{1+\eta_{11}} = \frac{2}{1+\sin^2(\frac{\pi}{n+1})}$

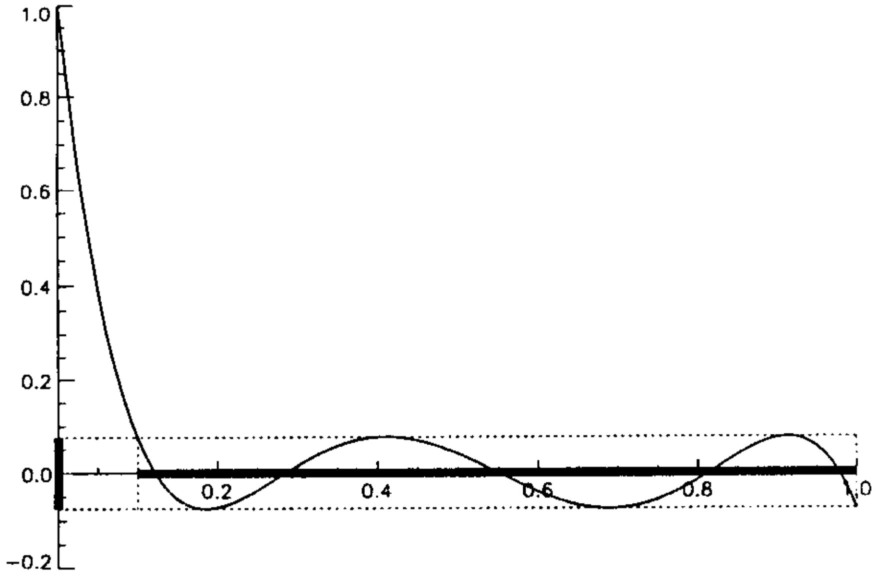


$$\rho = S(I - \alpha(D-L)^{-1}A)$$
$$= \left(\frac{1-\eta_{11}}{1+\eta_{11}}\right) = \frac{1-\sin^2(\frac{\pi}{n+1})}{1+\sin^2(\frac{\pi}{n+1})}$$

$$\varepsilon = \rho^K \Rightarrow K$$
$$= \log(\frac{1}{\varepsilon})\frac{1}{2}(\frac{1}{\eta_{11}}) = \log\frac{1}{2\pi^2}n^2$$

Classical Matrix Splittings: Gauss-Seidel

Chebychev Iteration



$$\rho = \left(\frac{\sqrt{1/\eta_{11}} - 1}{\sqrt{1/\eta_{11}} + 1} \right)$$

$$\varepsilon = \rho^K \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{2}\sqrt{\frac{1}{\eta_{11}}} \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi}n$$

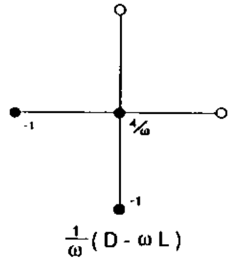
Classical Matrix Splittings: SOR

Splitting

$$A = M - N = D - L - U$$

$$M = \frac{1}{\omega}(D - \omega L)$$

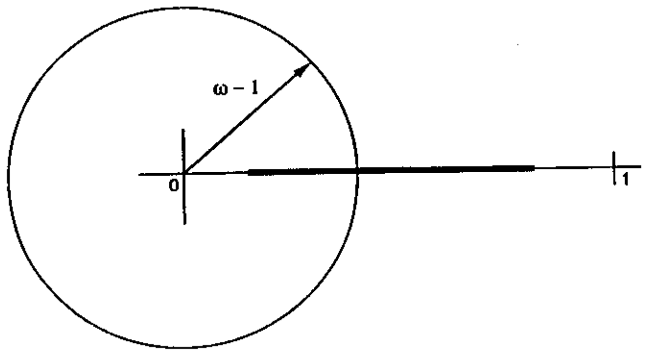
$$N = \frac{1}{\omega}((1-\omega)D + \omega U)$$



$$\Sigma(M^{-1}N) = \Sigma(\mathcal{L}_\omega)$$

$$\lambda^2 - (\omega^2\mu^2 - 2(\omega - 1)\lambda + (\omega - 1)^2 = 0$$

$$\mu \in \Sigma(D^{-1}(U + L))$$



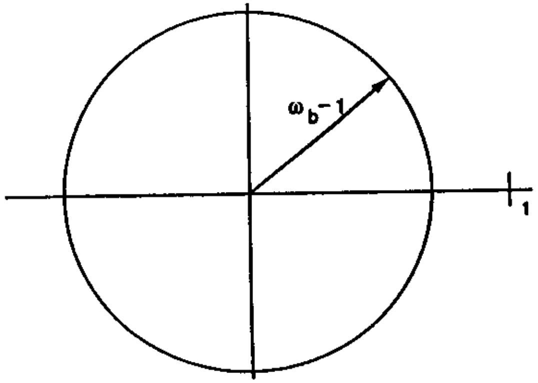
$1 \leq \omega \leq \omega_b$

Classical Matrix Splittings: SOR

Optimal ω

$$\omega_b = \frac{2}{1+\sqrt{1-\mu_{11}^2}} = \frac{2}{1+\sin(\frac{\pi}{n+1})}$$

$$\rho_b = \omega_b - 1 = \frac{1-\sin(\frac{\pi}{n+1})}{1+\sin(\frac{\pi}{n+1})}$$



$$\sum(\mathcal{L}_{\omega_b})$$

$$\varepsilon = \rho_b^K \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi}n$$

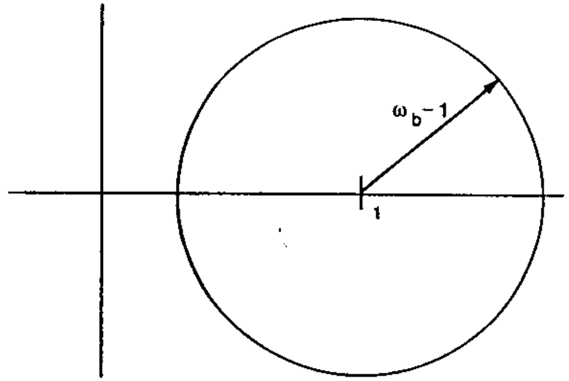
Classical Matrix Splittings: SOR

Preconditioning

$$A\underline{x} = \underline{b} \qquad M^{-1}A\underline{x} = M^{-1}\underline{b}$$

$$M = \frac{1}{\omega}(D - \omega L)$$

$$\sum(M^{-1}A)$$



- Optimal ellipse is circle centered at 1.0
- Optimal Chebychev iteration is 1-step with $\alpha = 1.0$
- No acceleration possible

Classical Matrix Splittings: SSOR

Matrix Splitting

$$(D - \omega L)\underline{x}_{i+\frac{1}{2}} = ((1-\omega)D + \omega U)\underline{x}_i + \omega \underline{b}$$

$$(D - \omega U)\underline{x}_{i+1} = ((1-\omega)D + \omega L)\underline{x}_{i+\frac{1}{2}} + \omega \underline{b}$$

$$\underline{x}_{i+1} = (D - \omega U)^{-1}((1-\omega)D + \omega L) \cdot (D - \omega L)^{-1}((1-\omega)D + \omega U)\underline{x}_i + \omega(2-\omega)(D-\omega U)^{-1}D(D-\omega L)^{-1}\underline{b}$$

Classical Matrix Splittings: SSOR

SSOR Operator

$$\mathcal{S}_\omega = (D - \omega U)^{-1}((1-\omega)D + \omega L) \cdot (D - \omega L)^{-1}((1-\omega)D + \omega U)$$

$$\omega_{\text{opt}} = \frac{2}{1 + \sqrt{2(1-\mu_1)}}$$

$$\rho = S(\mathcal{S}_{\omega_{\text{opt}}}) \leq \frac{1 - \sqrt{\frac{1}{2}(1-\mu_1)}}{1 + \sqrt{\frac{1}{2}(1-\mu_1)}} \cong 1 - \frac{\pi}{n+1}$$

$$\varepsilon = \rho^K \Rightarrow K \cong \log(\frac{1}{\varepsilon})\frac{1}{\pi} n$$

Classical Matrix Splittings: SSOR

$$((1-\omega)D+\omega L)(D-\omega L)^{-1}$$
$$= D((1-\omega)I+\omega D^{-1}L)(I-\omega D^{-1}L)^{-1}D^{-1}$$
$$= D(I-\omega D^{-1}L)^{-1}((1-\omega)I+\omega D^{-1}L)D^{-1}$$
$$= D(D-\omega L)^{-1}((1-\omega)D+\omega L)D^{-1}$$

$$\underline{x}_{i+1} = (D-\omega U)^{-1}D(D-\omega L)^{-1}$$
$$\cdot\ ((1-\omega)D+\omega L)D^{-1}((1-\omega)D+\omega U)\underline{x}_i$$
$$+\,\omega(2-\omega)(D-\omega U)^{-1}D(D-\omega L)^{-1}\underline{b}$$

$$(D-\omega L)D^{-1}(D-\omega L)\underline{x}_{i+1}$$
$$= (D-\omega L)D^{-1}(D-\omega U)\underline{x}_i$$
$$+\,\omega(2-\omega)(\underline{b}-(D-(L+U))\underline{x}_i)$$

$$M\underline{x}_{i+1} = M\underline{x}_i + \alpha\underline{r}_i$$

Classical Matrix Splitting: SSOR

Preconditioning $\quad M^{-1}A\underline{x} = M^{-1}\underline{b}$

$$M = \frac{1}{\omega(2-\omega)}(D-\omega L)D^{-1}(D-\omega U)$$

$$\left[\frac{1}{2-\omega}\right]\begin{bmatrix} \frac{\omega}{4} & -1 & \\ -1 & \frac{4}{\omega}+\frac{\omega}{2} & -1 \\ & -1 & \frac{\omega}{4} \end{bmatrix}$$

Stencil of M

- Invariant of scaling $\left(\frac{1}{(2-\omega)}\right)$
- Convergence factor depends on $\frac{\lambda_{\max}(M^{-1}A)}{\lambda_{\min}(M^{-1}A)}$

Classical Matrix Splitting: SSOR

$$\Sigma(M^{-1}A) = \Sigma(I - \mathcal{S}_{\omega_{\text{opt}}})$$

$$\eta_1 \geq \frac{2\sin(\frac{\pi}{2(n+1)})}{1+\sin(\frac{\pi}{2(n+1)})} \cong \frac{\pi}{n+1}$$

One-step ($\alpha = 2/(1+\eta_1)$)

$$\rho = \frac{1-\eta_1}{1+\eta_1}; \quad K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi}n$$

Chebychev

$$\rho = \left(\frac{\sqrt{1/\eta_1}-1}{\sqrt{1/\eta_1}+1}\right); \quad K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\sqrt{\pi}}n^{1/2}$$

Classical Matrix Splittings: Summary

Jacobi: $M = D$

1-step ($\alpha = 1$)	$K \cong \log(\frac{1}{\varepsilon})\frac{2}{\pi^2}n^2$
Chebychev	$K \cong \log(\frac{1}{\varepsilon})\frac{1}{\pi}n$

Gauss-Seidel: $M = D - L$

1-step ($\alpha = 1$)	$K \cong \log(\frac{1}{\varepsilon})\frac{1}{\pi^2}n^2$
1-step ($\alpha = 2 - O(\frac{1}{n^2})$)	$K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi^2}n^2$
Chebychev	$K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi}n$

Classical Matrix Splittings: Summary

SOR: $M = \frac{1}{\omega}(D - \omega L) \quad \omega_b \cong 2 - \frac{2\pi}{n+1}$

1-step $\quad K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi}n$

SSOR: $M = \frac{1}{\omega(2-\omega)}(D - \omega L)D^{-1}(D - \omega U)$

$(\omega_{\text{opt}} \cong (2 - \frac{2\pi}{n+1}))$

1-step $(\alpha = 1)$ $\quad K \cong \log(\frac{1}{\varepsilon})\frac{1}{\pi}n$

1-step $(\alpha = 2 - O(\frac{1}{n}))$ $\quad K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\pi}n$

Chebychev $\quad K \cong \log(\frac{1}{\varepsilon})\frac{1}{2\sqrt{\pi}}n^{1/2}$

Incomplete Factorization Preconditioning

$$M = (\Delta - \hat{L})\Delta^{-1}(\Delta - \hat{U})$$
$$\Delta = \text{diag}(\cdots \delta_i \cdots)$$

$\hat{L}$ Strictly Lower Triangular

$\hat{U}$ Strictly Upper Triangular

- Assume $\hat{U} = \hat{L}^T$
- Choose nonzero pattern in $\hat{L}, \hat{U}$
- Choose Δ, $\hat{L}$, $\hat{U}$ to match A in some sense
- Use Chebychev Iteration or Conjugate Gradient Iteration

Incomplete Factorization: IC

$$M = (\Delta - \hat{L})\Delta^{-1}(\Delta - \hat{U}) \qquad (= (m_{ij}))$$
$$A = D - L - U \qquad (= (a_{ij}))$$

Incomplete Choleski: Choose Δ, $\hat{L}$, $\hat{U}$ so that

If $a_{ij} \neq 0$ then $a_{ij} = m_{ij}$

IC(0)

$\hat{\ell}_{ij} \neq 0$ only if $\ell_{ij} \neq 0$

IC(1)

$\hat{\ell}_{ij} \neq 0$ only if there exists k $\ell_{ik}\ell_{jk} \neq 0$

Incomplete Factorization: IC

IC(0): Model Problem

$$M = (\Delta - \hat{L})\Delta^{-1}(\Delta - \hat{U})$$

$$\hat{L} = \begin{bmatrix} 0 & & & & & \\ \hat{\ell}_{2,1} & 0 & & & & \\ & \ddots & \ddots & & & \\ \hat{\ell}_{i,i-n} & \cdots & \hat{\ell}_{i,i-1} & 0 & & \\ & \ddots & & \ddots & \ddots & \\ & & \hat{\ell}_{N,N-1} & \cdots & \hat{\ell}_{N,N-1} & 0 \end{bmatrix}$$

$$\hat{U} = \begin{bmatrix} 0 & \hat{u}_{1,2} & \cdots & \hat{u}_{1,n} & & \\ & \ddots & \ddots & & \ddots & \\ & & 0 & \hat{u}_{i,i+1} & \cdots & \hat{u}_{i,i+n} \\ & & & \ddots & \ddots & \\ & & & & 0 & \hat{u}_{N-1,N} \\ & & & & & 0 \end{bmatrix}$$

$$M = \Delta - (\hat{L} + \hat{U}) + \hat{L}\Delta^{-1}\hat{U}$$

Incomplete Factorization: IC

IC(0): Stencil of M

$$M = \Delta - (\hat{L} + \hat{U}) + \hat{L}\Delta^{-1}\hat{U}$$

$$\begin{array}{ccc}
\frac{\hat{\ell}_{i,i-1}\hat{u}_{i-1,i-1+n}}{\delta_{i-1}} & -\hat{u}_{i,i+n} & \\
-\hat{\ell}_{i,i-1} & \gamma & -\hat{u}_{i,i+1} \\
 & -\hat{u}_{i,i-n} & \frac{\hat{\ell}_{i,i-n}\hat{u}_{i-n,i-n+1}}{\delta_{i-n}}
\end{array}$$

$$\gamma = \delta_i + \frac{\hat{\ell}_{i,i-1}\hat{u}_{i-1,i}}{\delta_{i-1}} + \frac{\hat{\ell}_{i,i-n}\hat{u}_{i-n,i}}{\delta_{i-n}}$$

Stencil of A:

$$\begin{array}{ccc}
 & -u_{i,i+n} & \\
-\ell_{i,i-1} & d_i & -u_{i,i+1} \\
 & -\ell_{i,i-n} &
\end{array}$$

Incomplete Factorization: IC

IC(0):

$$\begin{aligned}
\hat{L} &= L \\
\hat{U} &= U \\
\Delta &= \operatorname{diag}(\cdots \delta_i \cdots)
\end{aligned}$$

$$\delta_i = d_i - \frac{\ell_{i,i-1}u_{i-1,i}}{\delta_{i-1}} - \frac{\ell_{i,i-n}u_{i-n,i}}{\delta_{i-n}}$$

Model Problem:

$$\delta_i = 4 - \frac{1}{\delta_{i-1}} - \frac{1}{\delta_{i-n}}$$

$$\delta_i \to \delta = 4 - \frac{2}{\delta}$$

$$\delta_i \to 2 + \sqrt{2} \quad \text{(from above)}$$

Incomplete Factorization: IC

IC(0):

$$M = (\Delta - L)\Delta^{-1}(\Delta - U)$$
$$\Delta = \text{diag}(\cdots \delta_i \cdots)$$
$$\delta_i \to 2 + \sqrt{2}$$

SSOR:

$$M = (\frac{1}{\omega}D - L)(\frac{1}{\omega}D)^{-1}(\frac{1}{\omega}D - L)$$

- IC(0) closely resembles SSOR with

$$\omega = \frac{4}{2+\sqrt{2}} = 1.17$$
$$\frac{\lambda_{\max}(M^{-1}A)}{\lambda_{\min}(M^{-1}A)} = O(n^2)$$

- Chebychev or CG

$$K \cong \log(\frac{1}{\varepsilon})Cn$$

Incomplete Factorization: MIC

Modified Incomplete Choleski

$$M = (\Delta - \hat{L})\Delta^{-1}(\Delta - \hat{U})$$

If $a_{ij} \neq 0$ $(i \neq j)$ then $a_{ij} = m_{ij}$

Row sum of A = Row sum of M

MIC(0):

$\hat{\ell}_{ij} \neq 0$ only if $\ell_{ij} \neq 0$

MIC(1)

$\hat{\ell}_{ij} \neq 0$ only if there exists k $\ell_{ik}\ell_{jk} \neq 0$

Incomplete Factorization: MIC

MIC(0): Stencil of M

$$M = \Delta - (\hat{L} + \hat{U}) + \hat{L}\Delta^{-1}\hat{U}$$

$$\begin{array}{ccc}
\frac{\hat{\ell}_{i,i-1}\hat{u}_{i-1,i-1+n}}{\delta_{i-1}} & -\hat{u}_{i,i+n} & \\
-\hat{\ell}_{i,i-1} & \gamma & -\hat{u}_{i,i+1} \\
 & -\hat{u}_{i,i-n} & \frac{\hat{\ell}_{i,i-n}\hat{u}_{i-n,i-n+1}}{\delta_{i-n}}
\end{array}$$

$$\gamma = \delta_i + \frac{\hat{\ell}_{i,i-1}\hat{u}_{i-1,i}}{\delta_{i-1}} + \frac{\hat{\ell}_{i,i-n}\hat{u}_{i-n,i}}{\delta_{i-n}}$$

Stencil of A:

$$\begin{array}{ccc}
 & -u_{i,i+n} & \\
-\ell_{i,i-1} & d_i & -u_{i,i+1} \\
 & -\ell_{i,i-n} &
\end{array}$$

Incomplete Factorization: MIC

MIC(0):

$$\begin{aligned}
\hat{L} &= L \\
\hat{U} &= U \\
\Delta &= \operatorname{diag}(\cdots \delta_i \cdots)
\end{aligned}$$

$$\delta_i = d_i - \frac{\ell_{i,i-1}(u_{i-1,i} + u_{i-1,i-1+n})}{\delta_{i-1}} + \frac{\ell_{i,i-n}(u_{i-n,i} + u_{i-n,i-n+1})}{\delta_{i-n}}$$

Model Problem:

$$\delta_i = 4 - \frac{2}{\delta_{i-1}} - \frac{2}{\delta_{i-n}}$$

$$\delta_i \to \delta = 4 - \frac{4}{\delta}$$

$$\delta_i \to 2 \quad \text{(from above)}$$

Incomplete Factorization: MIC

MIC(0)

$$M = (\Delta - L)\Delta^{-1}(\Delta - U)$$
$$\Delta = \text{diag}(\cdots \delta_i \cdots)$$
$$\delta_i \to 2 \quad \text{(from above)}$$

SSOR:

$$M = (\frac{1}{\omega}D - L)(\frac{1}{\omega}D)^{-1}(\frac{1}{\omega}D - U)$$

- MIC(0) resembles SSOR with

$$\omega = \frac{4}{\delta_i} \to 2 \quad \text{(from below)}$$
$$\frac{\lambda_{\max}(M^{-1}A)}{\lambda_{\min}(M^{-1}A)} = O(n)$$

- Chebychev or CG

$$K \cong \log(\frac{1}{\varepsilon})Cn^{1/2}$$

Incomplete Factorization

Generalizations

- More nonzeros in $\hat{L}$, $\hat{U}$
- Block diagonal Δ
- Factor a nearby matrix $(A + \alpha D)$

Either

$$\frac{\lambda_{\max}(M^{-1}A)}{\lambda_{\min}(M^{-1}A)} = O(n)$$
$$K = \log(\frac{1}{\varepsilon})Cn^{1/2}$$

Or

$$(\Delta - L)^{-1} \quad \text{unstable}$$

Equivalent Operators

Suppose

$$M_h \underline{v}_h = \underline{f}_h \qquad Mv = f$$
$$\|\underline{v}_h - v\|_{L_2} \leq O(h)$$

$$A_h \underline{u}_h = \underline{f}_h \qquad Au = f$$
$$\|\underline{u}_h - u\|_{L_2} \leq O(h)$$

Then

$$\|\underline{v}_h - \underline{u}_h\|_{L_2} \leq (1 + \|M^{-1}A\|_{L_2})\|\underline{u}_h\|_{L_2} + O(h)$$

Thus

$$\begin{aligned}\|M_h^{-1}A_h\underline{u}_h\|_{L_2} &\leq \|M_h^{-1}A_h\underline{u}_h - \underline{u}_h\|_{L_2} + \|\underline{u}_h\|_{L_2} \\ &\leq (2 + \|M^{-1}A\|_{L_2})\|\underline{u}_h\|_{L_2}\end{aligned}$$

$$\|A_h^{-1}M_h\underline{v}_h\|_{L_2} \leq (2 + \|A^{-1}M\|_{L_2})\|\underline{v}_h\|_{L_2}$$

Equivalent Operators

Result: Let A and M be any two uniformly elliptic partial differential operators with H_2 regularity, that is,

$$\|Au\|_{L_2} \leq K_1(A)\|u\|_{H_2},$$
$$\|Mu\|_{L_2} \leq K_1(M)\|u\|_{H_2}$$

$$\|A^{-1}f\|_{H_2} \leq K_2(A)\|f\|_{L_2},$$
$$\|M^{-1}f\|_{H_2} \leq K_2(M)\|f\|_{L_2}$$

Then

$$\kappa_{L_2}(M^{-1}A) = \|M^{-1}A\|_{L_2}\|A^{-1}M\|_{L_2} < \infty$$

if and only if A^* and M^* have the same boundary conditions

Equivalent Operators

SSOR: $M = (\frac{1}{\omega}D - L)(\frac{1}{\omega}D)^{-1}(\frac{1}{\omega}D - U)$

$$\begin{array}{ccc} \frac{\omega}{4} & -1 & \\ -1 & \frac{4}{\omega} + \frac{\omega}{2} & -1 \\ & -1 & \frac{\omega}{4} \end{array}$$

Stencil of M

Consider smooth $u(x,y)$

$$\begin{array}{ccc} u(x-h,y+h) & u(x,y+h) & \\ u(x-h,y) & u(x,y) & u(x+h,y) \\ & u(x,y-h) & u(x+h,y-h) \end{array}$$

Equivalent Operators

$$\frac{1}{h^2}Au = -(u_{xx} + u_{yy}) + O(h^2)$$

$$\frac{1}{h^2}Mu = -(u_{xx} + u_{yy}) + \frac{\omega}{4}(u_{xx} - 2u_{xy} + u_{yy}) + \frac{1}{h^2}(\omega + \frac{4}{\omega} - 4)u + O(h^2)$$

Let $\omega_{\text{opt}} = 2 - Ch$

$$\frac{1}{h^2}Mu = -\frac{1}{2}(u_{xx} + 2u_{xy} + u_{yy}) + \frac{C^2}{2}u + O(h)$$

What goes wrong?

- M is not uniformly elliptic
- M_h corresponds to mixed boundary conditions

Preconditionings: Summary

- For elliptic problems matrix splittings and incomplete factorizations at best yield

$$K = \log(\frac{1}{\varepsilon})Cn^{1/2}$$

- For elliptic problems multigrid methods yield

$$K = \log(\frac{1}{\varepsilon})C$$

- Multigrid methods may be viewed as a special form of preconditioning
- For many other applications classical preconditionings are very useful

Preconditioned Polynomial Iterative Methods

III. Conjugate Gradient-like Methods

TOM MANTEUFFEL
UNIVERSITY OF COLORADO AT DENVER

Conjugate Gradient-like Methods

A. Introduction

B. Conjugate Gradient Methods

C. Projection Methods

Conjugate Gradient Methods

General Polynomial Iteration

$$\underline{x}_k = \underline{x}_{k-1} + \sum_{j=0}^{k-1} \eta_{kj}\, \underline{r}_j$$

$$\underline{e}_k = \underline{e}_{k-1} - \sum_{j=0}^{k-1} \eta_{kj}\, \underline{r}_j$$

Recall

$$\underline{e}_k = p_k(A)\underline{e}_0$$

$$\underline{r}_k = p_k(A)\underline{r}_0$$

Krylov Space

$$K_k(\underline{r}_0, A) = sp\{\underline{r}_0, a\underline{r}_0, A^2\underline{r}_0, \cdots, A^{k-1}\underline{r}_0\}$$
$$= sp\{\underline{r}_0, \underline{r}_1, \underline{r}_2, \cdots, \underline{r}_{k-1}\}$$

Conjugate Gradient Methods

Given B Hermitian Positive Definite (HPD) Inner Product

$$\langle B\underline{x}, \underline{y}\rangle$$

Norm

$$\|\underline{x}\|_B = \langle B\underline{x}, \underline{x}\rangle^{1/2}$$

Choose η_{kj}'s to minimize

$$\|\underline{e}_k\|_B = \|\underline{e}_{k-1} - \sum_{j=0}^{k-1} \eta_{kj}\, \underline{r}_j\|_B$$

Choose η_{kj}'s such that

$$\langle B\underline{e}_k, \underline{r}_j\rangle = 0; \quad j = 0, \cdots, k-1$$

$$\underline{e}_k \perp_B K_k(\underline{r}_0, A)$$

Conjugate Gradient Methods: Step Direction

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\, \underline{p}_{k-1}$$

$$\underline{p}_{k-1} \in K_k(\underline{r}_0, A) = sp\{\underline{r}_0, \cdots, \underline{r}_{k-1}\}$$

$$\underline{e}_k = \underline{e}_{k-1} - \alpha_{k-1}\, \underline{p}_{k-1}$$

$$\underline{e}_k \perp_B K_k(\underline{r}_0, A) \quad (\supseteq K_{k-1}(\underline{r}_0, A))$$

By previous step

$$\underline{e}_{k-1} \perp_B K_{k-1}(\underline{r}_0, A)$$

Result: $\underline{p}_{k-1}$ is the unique (up to scale) vector

$$\underline{p}_{k-1} \in K_k(\underline{r}_0, A)$$

$$\underline{p}_{l\ -1} \perp_B K_{k-1}(\underline{r}_0, A)$$

Conjugate Gradient Methods: Step Length

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\, \underline{p}_{k-1}$$

$$\underline{p}_{k-1} \in K_k(\underline{r}_0, A) = sp\{\underline{r}_0, \cdots, \underline{r}_{k-1}\}$$

$$\underline{e}_k = \underline{e}_{k-1} - \alpha_{k-1}\, \underline{p}_{k-1}$$

$$\underline{e}_k \perp_B K_k(\underline{r}_0, A)$$

In particular,

$$\langle B\underline{e}_k, \underline{p}_{k-1}\rangle = \langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle - \alpha_{k-1}\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle = 0$$

$$\alpha_{k-1} = \frac{\langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

Conjugate Gradient Methods:

Generate a B-orthogonal Basis for $K_k(\underline{r}_0, A)$

$$sp\{\underline{p}_0, \underline{p}_1, \cdots, \underline{p}_{k-1}\} = sp\{\underline{r}_0, \underline{r}_1, \cdots, \underline{r}_{k-1}\}$$
$$= sp\{\underline{r}_0, A\underline{r}_0, \cdots, A^{k-1}\underline{r}_0\}$$

Gram-Schmidt Process

$$\underline{p}_0 = \underline{r}_0$$

$$\underline{p}_k = A\underline{p}_{k-1} - \sum_{j=0}^{k-1} \sigma_{kj}\, \underline{p}_j$$

$$\sigma_{kj} = \frac{\langle BA\underline{p}_{k-1}, \underline{p}_j\rangle}{\langle B\underline{p}_j, \underline{p}_j\rangle}$$

Yields:

$$\langle B\underline{p}_k, \underline{p}_j\rangle = 0; \quad j = 0, \cdots, k-1$$

Conjugate Gradient Methods: Algorithms

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\underline{p}_{k-1}; \quad \alpha_{k-1} = \frac{\langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

$$\underline{r}_k = \underline{r}_{k-1} - \alpha_{k-1} A\underline{p}_{k-1}$$

ODIR

$$\underline{p}_k = A\underline{p}_{k-1} - \sum_{j=0}^{k-1} \alpha_{kj}\, \underline{p}_j$$

OMIN

$$\underline{p}_k = \underline{r}_k - \sum_{j=0}^{k-1} \hat{\alpha}_{kj}\, \underline{p}_j$$

ORES

$$\underline{p}_k = \underline{r}_k - \sum_{j=1}^{k} \gamma_{kj}(\underline{x}_j - \underline{x}_{j-1})$$

Conjugate Gradient Methods: Computability

Need to compute

$$\alpha_{k-1} = \frac{\langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

Example: A Hermitian Positive Definite

$$B = A$$

$$\alpha_{k-1} = \frac{\langle \underline{r}_{k-1}, \underline{p}_{k-1}\rangle}{\langle A\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

More on this subject later

Conjugate Gradient Methods: Economical Computation

ODIR

$$\underline{p}_k = A\underline{p}_{k-1} - (\sigma_{k\,k-1}\,\underline{p}_{k-1} + \sigma_{k\,k-2}\,\underline{p}_{k-2})$$

OMIN

$$\underline{p}_k = \underline{r}_k - \hat{\sigma}_{k\,k-1}\,\underline{p}_{k-1}$$

ORES

$$\underline{p}_k = \underline{r}_k - \gamma_{k\,k-1}(\underline{x}_k - \underline{x}_{k-1})$$

If and only if

A is B-normal (1)

or

$$\left\{\begin{array}{ll} d(A) \le 3 & \text{ODIR} \\ d(A) \le 2 & \text{OMIN, ORES} \end{array}\right\}$$

Conjugate Gradient Methods: Economical Computation

B-adjoint

$$\langle BA\underline{x}, \underline{y}\rangle = \langle B\underline{x}, A^+ \underline{y}\rangle$$

$$A^+ = (BAB^{-1})^* = B^{-1} A^* B$$

B-normal (s)

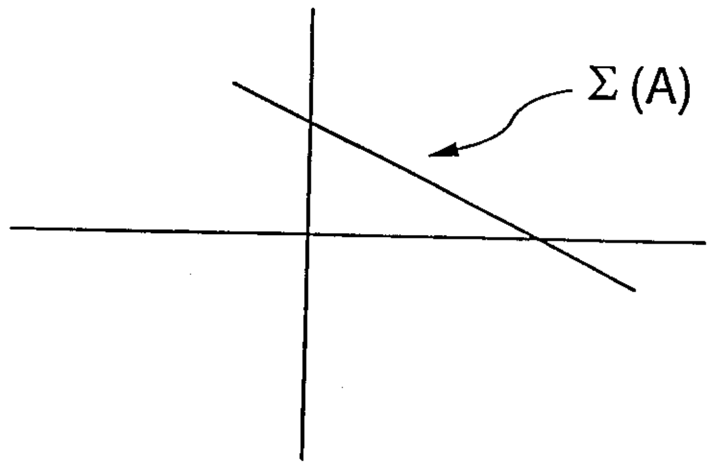
- $AA^+ = A^+A$
- A, A^+ have same complete set of B-orthogonal eigenvectors
- $A^+ = p_s(A)$ (polynomial of degree s)

Conjugate Gradient Methods: Economical Computation

A is B-normal (1)

$$A^+ = A \quad B\text{-self-adjoint}$$

$$A^+ = \alpha I + \beta A$$



Necessary Condition

Conjugate Gradient Methods: Economical Computation

ODIR

$$\underline{p}_k = A\underline{p}_{k-1} - \sum_{j=0}^{k-1} \sigma_{kj}\, \underline{p}_j$$

$$\sigma_{kj} = \frac{\langle BA\underline{p}_{k-1}, \underline{p}_j\rangle}{\langle B\underline{p}_j, \underline{p}_j\rangle}$$

Result: If A is B-normal (1) then

$$\sigma_{kj} = 0; \quad j < k-2$$

Proof: $\quad A^+ = \alpha I + \beta A$

$$\langle BA\underline{p}_{k-1}, \underline{p}_j\rangle = \langle B\underline{p}_{k-1}, A^+ \underline{p}_j\rangle$$

$$= \alpha\langle B\underline{p}_{k-1}, \underline{p}_j\rangle + \beta\langle B\underline{p}_{k-1}, A\underline{p}_j\rangle$$

$$\langle B\underline{p}_{k-1}, \underline{p}_j\rangle = 0; \quad j < k-1$$

$$\langle B\underline{p}_{k-1}, A\underline{p}_j\rangle = 0; \quad j < k-2$$

Conjugate Gradient Methods: Preconditioning

Original System	Preconditioned System
$A\underline{x} = \underline{b}$	$CA\underline{x} = C\underline{b}$

Iteration: $CG(B, C, A)$ (ODIR Algorithm)

$$\underline{r}_0 = \underline{b} - A\underline{x}_0$$

$$\underline{p}_0 = C\underline{r}_0,$$

$$\vdots$$

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\,\underline{p}_{k-1}; \quad \alpha_{k-1} = \frac{\langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

$$\underline{r}_k = \underline{r}_{k-1} - \alpha_{k-1}\, A\underline{p}_{k-1}$$

$$\underline{p}_k = CA\underline{p}_{k-1} - \sum_{j=0}^{k-1} \sigma_{kj}\,\underline{p}_j; \quad \sigma_{kj} = \frac{\langle BCA\underline{p}_{k-1}, \underline{p}_j\rangle}{\langle B\underline{p}_j, \underline{p}_j\rangle}$$

Conjugate Gradient Methods: Computability

- If $B\underline{e}_{k-1}$ is computable:

$$\alpha_{k-1} = \frac{\langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

- If $C^*B\underline{e}_{k-1}$ is computable:

$$\underline{r}_0 = \underline{b} - A\underline{x}_0$$

$$\underline{q}_0 = \underline{r}_0$$

$$\underline{p}_0 = C\underline{q}_0$$

$$\vdots$$

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\,\underline{p}_{k-1}; \quad \alpha_{k-1} = \frac{\langle C^*B\underline{e}_{k-1}, \underline{q}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$

$$\underline{r}_k = \underline{r}_{k-1} - \alpha_{k-1}A\underline{p}_{k-1}$$

$$\underline{q}_k = A\underline{q}_{k-1} - \sum_{j=0}^{k-1} \sigma_{kj}\,\underline{q}_j; \quad \sigma_{kj} = \frac{\langle BCA\underline{p}_{k-1}, \underline{p}_j\rangle}{\langle B\underline{p}_j, \underline{p}_j\rangle}$$

$$\underline{p}_k = C\underline{q}_k$$

Conjugate Gradient Methods:

Given system

$$A\underline{x} = \underline{b},$$

Preconditioning

$$CA\underline{x} = c\underline{b},$$

Hermitian Positive Definite B, then

- $CG(B, C, A)$ is computable if $C^*B\underline{e}_{k-1}$ is computable
- $CG(B, C, A)$ is economically computable (three term recursion) if and only if

$$CA \text{ is } B\text{-normal } (1)$$

or

$$d(A) \leq 3$$

Conjugate Gradient Methods: Example

Original Method of Hestenes and Steifel

$$A - HPD$$

$$C = I$$

$$B = A$$

- A is A-self-adjoint

$$\langle AA\underline{x}, \underline{y}\rangle = \langle A\underline{x}, A\underline{y}\rangle$$

- Computable

$$\alpha_k = \frac{\langle B\underline{e}_k, \underline{p}_k\rangle}{\langle B\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle A\underline{e}_k, \underline{p}_k\rangle}{\langle A\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle \underline{r}_k, \underline{p}_k\rangle}{\langle A\underline{p}_k, \underline{p}_k\rangle}$$

Conjugate Gradient Methods: Example

Preconditioned Conjugate Gradient Method

$$A\underline{x} = \underline{b}$$

$$CA\underline{x} = C\underline{b}$$

$$A, C - HPD$$

$$B = A$$

- CA is A-self-adjoint

$$\langle ACA\underline{x}, \underline{y}\rangle = \langle A\underline{x}, CA\underline{y}\rangle$$

- Computable

$$\alpha_k = \frac{\langle B\underline{e}_k, \underline{p}_k\rangle}{\langle B\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle A\underline{e}_k, \underline{p}_k\rangle}{\langle A\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle \underline{r}_k, \underline{p}_k\rangle}{\langle A\underline{p}_k, \underline{p}_k\rangle}$$

Conjugate Gradient Methods: Example

Conjugate Residual Method

$$A\underline{x} = \underline{b}$$

A, Hermitian (possibly indefinite)

$$C = I$$

$$B = A^*A = A^2$$

- A is A^2-self-adjoint

$$\langle A^2 A\underline{x}, \underline{y}\rangle = \langle A^2\underline{x}, A\underline{y}\rangle$$

- Computable

$$\alpha_k = \frac{\langle A^2\underline{e}_k, \underline{p}_k\rangle}{\langle A^2\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle \underline{r}_k, A\underline{p}_k\rangle}{\langle A\underline{p}_k, A\underline{p}_k\rangle}$$

Conjugate Gradient Methods: Example

Preconditioned Conjugate Residual

$$A\underline{x} = \underline{b} \qquad A \quad \text{Hermitian}$$

$$CA\underline{x} = C\underline{b} \qquad C \quad HPD$$

$$B = ACA$$

- CA is ACA-self-adjoint

$$\langle ACACA\underline{x}, \underline{y}\rangle = \langle ACA\underline{x}, CA\underline{y}\rangle$$

- Computable

$$\alpha_k = \frac{\langle ACA\underline{e}_k, \underline{p}_k\rangle}{\langle ACA\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle C\underline{r}_k, A\underline{p}_k\rangle}{\langle A\underline{p}_k, A\underline{p}_k\rangle}$$

Conjugate Gradient Methods: Example

Normal Equations

$$A\underline{x} = \underline{b}$$

$$A^* A\underline{x} = A^* \underline{b}$$

$$C = A^*$$

$$B = A^* A$$

- $A^* A$ is $A^* A$-self-adjoint

$$\langle A^* A\, A^* A\underline{x}, y\rangle = \langle A^* A\underline{x}, A^* Ay\rangle$$

- Computable

$$\alpha_k = \frac{\langle A^* A\underline{e}_k, \underline{p}_k\rangle}{\langle A^* A\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle \underline{r}_k, A\underline{p}_k\rangle}{\langle A\underline{p}_k, A\underline{p}_k\rangle}$$

Conjugate Gradient Methods: Example

Craig's Method

$$A\underline{x} = \underline{b}$$

$$A^* A\underline{x} = A^* \underline{b}$$

$$C = A^*$$

$$B = I$$

- $A^* A$ is I-self-adjoint

$$\langle A^* A\underline{x}, \underline{y}\rangle = \langle \underline{x}, A^* A\underline{y}\rangle$$

- Computable $\quad C^* B\underline{e}_k = A\underline{e}_k = \underline{r}_k$

$$\alpha_k = \frac{\langle B\underline{e}_k, \underline{p}_k\rangle}{\langle B\underline{p}_k, \underline{p}_k\rangle} = \frac{\langle \underline{r}_k, \underline{q}_k\rangle}{\langle \underline{p}_k, \underline{p}_k\rangle}$$

Conjugate Gradient Methods: Example

Normal Equations of Preconditioned System

$$A\underline{x} = \underline{b}$$

$$M^{-1}A\underline{x} = M^{-1}\underline{b}$$

$$(M^{-1}A)^* M^{-1}A\underline{x} = (M^{-1}A)^* M^{-1}\underline{b}$$

$$C = A^* M^{-*} M^{-1}$$

- PCGNS

$$B = A^* M^{-*} M^{-1} A$$

- PCGNE (Craig's Method)

$$B = I$$

Conjugate Gradient Methods: Example

Preconditioning the Normal Equations

$$A\underline{x} = \underline{b}$$

$$A^* A\underline{x} = A^*\underline{b}$$

$$(M^*M)^{-1} (A^*A)\underline{x} = (M^*M)^{-1} A^*\underline{b}$$

$$C = M^{-1} M^* A^*$$

- PCGNR

$$B = A^* A$$

- PCGNM

$$B = M^* M$$

Conjugate Gradient Methods: Basic Patterns

Pattern	Name	B	CA	Orthodir: Restrictions	Orthodir: α	Orthomin: Restrictions	Orthomin: $\hat{\alpha}$
P1	GCGHS	CA	CA	CA hpd	$\frac{\langle s,p\rangle}{\langle CAp,p\rangle}$	CA hpd	$\frac{\langle s,s\rangle}{\langle CAp,p\rangle}$
P2	GCR	$(CA)^*(CA)$	CA	CA herm	$\frac{\langle s,CAp\rangle}{\langle CAp,CAp\rangle}$	CA hpd	$\frac{\langle s,CAs\rangle}{\langle CAp,CAp\rangle}$
P3	GPCG	EA	DEA	EA hpd D herm	$\frac{\langle Er,p\rangle}{\langle EAp,p\rangle}$	EA hpd D hpd	$\frac{\langle Er,s\rangle}{\langle EAp,p\rangle}$
P4	GPCR	A^*EA	CA	E hpd EAC herm	$\frac{\langle Er,Ap\rangle}{\langle EAp,Ap\rangle}$	E hpd EAC hpd	$\frac{\langle Er,As\rangle}{\langle EAp,Ap\rangle}$
P5	GCGE	I	A^*DA	D herm	$\frac{\langle r,Dq\rangle}{\langle p,p\rangle}$	D hpd	$\frac{\langle r,Dr\rangle}{\langle p,p\rangle}$
P6	GCGIB	B	$B^{-1}A^*DA$	B hpd D herm	$\frac{\langle r,Dq\rangle}{\langle Ap,Dq\rangle}$	B hpd D hpd	$\frac{\langle r,Dr\rangle}{\langle DAp,q\rangle}$
P7	GCGCB	B	A^*DA	B hpd BD herm $AB=BA$	$\frac{\langle Br,Dq\rangle}{\langle BAp,Dq\rangle}$	B hpd BD hpd $AB=BA$	$\frac{\langle Br,Dq\rangle}{\langle BAp,Dq\rangle}$

Conjugate Gradient Methods:

Method:

- Refers to a specific choice of B, C, A
- Computable if $C^*B\underline{e}$ is computable
- Iterates are uniquely determined (given $\underline{x}_0$)
- Economically computable if CA B-normal (1)

Pattern:

- Relationship among B, C, and A that yields CA B-normal (1)

Algorithm:

- Sequence of arithmetic steps used to implement a method

Conjugate Gradient Methods: OMIN Algorithm

$$\vdots$$

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\underline{p}_{k-1}$$
$$\underline{r}_k = \underline{r}_{k-1} - \alpha_{k-1}A\underline{p}_{k-1}$$
$$\underline{s}_k = C\underline{r}_k$$
$$\underline{p}_k = \underline{s}_k - \sum_{j=0}^{k-1} \hat{\sigma}_{kj}\underline{p}_k$$

$$\vdots$$

B-orthogonal Basis

$$sp\{\underline{p}_0, \underline{p}_1, \cdots, \underline{p}_k\} = sp\{\underline{s}_0, (CA)\underline{s}_0, \cdots, (CA)^k\underline{s}_0\}$$
$$\supseteq sp\{\underline{s}_0, \underline{s}_1, \cdots, \underline{s}_k\}$$

If $\underline{s}_k = \underline{s}_{k-1}$ (i.e., $\alpha_{k-1} = 0$)

$$\underline{s}_k \in sp\{\underline{s}_0, \underline{s}_1, \cdots, \underline{s}_{k-1}\}$$

Conjugate Gradient Methods: OMIN Algorithm

- CA B-normal (1)

$$\vdots$$

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_{k-1}\underline{p}_{k-1}$$
$$\alpha_{k-1} = \frac{\langle B\underline{e}_{k-1}, \underline{p}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle} = \frac{\langle B\underline{e}_{k-1}, \underline{s}_{k-1}\rangle}{\langle B\underline{p}_{k-1}, \underline{p}_{k-1}\rangle}$$
$$r_k = \underline{r}_{k-1} - \alpha_{k-1}A\underline{p}_{k-1}$$
$$\underline{s}_k = C\underline{r}_k$$
$$p_k = \underline{s}_k + \beta_{k-1}\underline{p}_{k-1}$$
$$\beta_{k-1} = \frac{\langle B\underline{e}_k, \underline{s}_k\rangle}{\langle B\underline{e}_{k-1}, \underline{s}_{k-1}\rangle}$$

OMIN Algorithm

- Requires less computation than ODIR
- Has better numerical properties
- Has smaller applicability

Conjugate Gradient Methods: OMIN Algorithm

Recall

$$\alpha_k = \frac{\langle B\underline{e}_k, \underline{p}_k \rangle}{\langle B\underline{p}_k, \underline{p}_k \rangle}$$

$$\underline{p}_k = \underline{s}_k - \beta_{k-1}\underline{p}_{k-1}$$

Thus

$$\alpha_k = \frac{\langle B\underline{e}_k, \underline{s}_k \rangle}{\langle B\underline{p}_k, \underline{p}_k \rangle} - \beta_{k-1} \frac{\langle B\underline{e}_k, \underline{p}_{k-1} \rangle}{\langle B\underline{p}_k, \underline{p}_k \rangle}$$

Finally

$$\langle B\underline{e}_k, \underline{s}_k \rangle = \langle B\underline{e}_k, CA\underline{e}_k \rangle = \langle \underline{e}_k, BCA\underline{e}_k \rangle$$

OMIN will not stall if BCA is definite:

$$\langle \underline{x}, BCA\underline{x} \rangle > 0 \quad \text{for all } \underline{x}$$

Conjugate Gradient Methods: Error Bounds

B-condition of A

$$\kappa_B(A) = \|A\|_B \, \|A^{-1}\|_B$$

If CA is B-self-adjoint

$$\frac{\|\underline{e}_k\|_B}{\|\underline{e}_0\|_B} \leq 2 \left(\frac{\sqrt{\kappa_B(CA)} - 1}{\sqrt{\kappa_B(CA)} + 1} \right)^k$$

Warning: Not all norms are the same

Conjugate Gradient-like Methods

Projection Methods

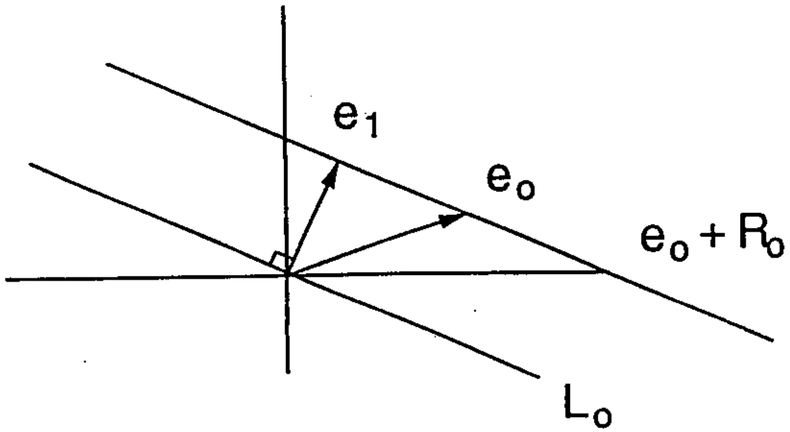
$$A\underline{x} = \underline{b}$$

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i$$

$$\underline{e}_{i+1} = \underline{e}_i - \alpha_i \underline{p}_i$$

$$\underline{p}_i \in \mathbf{R}_i \qquad dim(\mathbf{R}_i) = r_i$$

$$\underline{e}_{i+1} \perp \mathbf{L}_i \qquad dim(\mathbf{L}_i) = l_i$$



PROJECTION METHODS

R_i Matrix whose columns span $\mathbf{R}_i$

L_i Matrix whose columns span $\mathbf{L}_i$

$$\alpha_i \underline{p}_i = R_i \underline{w}_i$$

$$\underline{e}_{i+1} = \underline{e}_i - \alpha_i \underline{p}_i \perp \mathbf{L}_i$$

$$L_i^* \underline{e}_i = L_i^* R_i \underline{w}_i$$

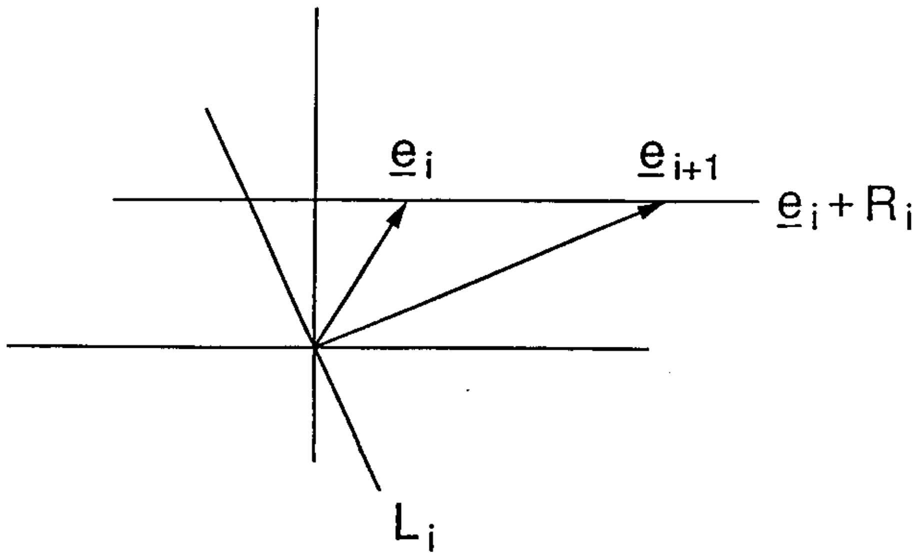
$$\alpha_i \underline{p}_i = R_i (L_i^* R_i)^{-1} L_i^* \underline{e}_i$$

Exists uniquely iff $L_i^* R_i$ invertible

PROJECTION METHODS

Breakdown

- $\alpha_i \underline{p}_i$ does not exist
- $\alpha_i \underline{p}_i$ is not unique



$(L_i^* R_i)$ Singular

PROJECTION METHODS

Write

$$L_i = B_i^* R_i$$

$$B_i^* = L_i (R_i^* R_i)^{-1} R_i^*$$

Then

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i$$

$$\underline{e}_{i+1} = \underline{e}_i - \alpha_i \underline{p}_i$$

Where

$$\alpha_i \underline{p}_i = R_i (R_i^* B_i R_i)^{-1} R_i^* B_i \underline{e}_i$$

Thus

$$\underline{e}_{i+1} = (I - R_i (R_i^* B_i R_i)^{-1} R_i^* B_i) \underline{e}_i$$

Projection Methods

Finite Termination

If: $dim(R_i) = i + 1$

Then: iteration will converge in at most N steps

(In absence of breakdown)

BALANCED PROJECTION METHODS

$$\mathbf{L}_i = B^* \mathbf{R}_i$$

Bounded Iterates

$$B_S = \frac{1}{2}(B + B^*) \qquad B_N = \frac{1}{2}(B - B^*)$$

- B Definite

$$\|\underline{e}_k\|_{B_s} \leq (1+\delta)^k \|\underline{e}_0\|_{B_s}$$

$$\delta = 0(\|B_N\|)$$

- B HPD

$$\|\underline{e}_k\|_B \leq \|\underline{e}_{k-1}\|_B$$

BALANCED PROJECTION METHODS

Convergence

If:

- B HPD
- $\underline{r}_{i-1} \in \mathbf{R}_i$

Then: there exists $\varepsilon > 0$

$$\|\underline{e}_k\|_B \leq (1-\varepsilon)^k \ \|\underline{e}_0\|_B$$

PROJECTION METHODS

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i$$

$$\underline{e}_{i+1} = \underline{e}_i - \alpha_i \underline{p}_i$$

$$\underline{p}_i \in \mathbf{R}_i$$

$$\underline{e}_{i+1} \perp B_i^* \mathbf{R}_i$$

Polynomial Methods (Krylov Spaces)

$$\mathbf{K}_i(r_0, A) = \{\underline{r}_0, A\,\underline{r}_0, \ldots, A^{i-1}\underline{r}_0\}$$

$$\mathbf{R}_i \subseteq \mathbf{K}_i(r_0, A)$$

$$\underline{e}_{i+1} = p_{i+1}(A)\underline{e}_0$$

$$\underline{e}_{i+1} \perp B_i^* \mathbf{R}_i$$

SEMI-KRYLOV PROJECTION METHODS

$R_i \subseteq \mathbf{K}_i(r_0, A)$

$\underline{e}_{i+1} \perp \mathbf{L}_i \subseteq B^* \mathbf{K}_i(r_0, A)$

Krylov Projection Methods

$\mathbf{R}_i = \mathbf{K}_i(r_0, A)$

$\underline{e}_{i+1} \perp B^* \mathbf{R}_i$

KRYLOV PROJECTION METHODS

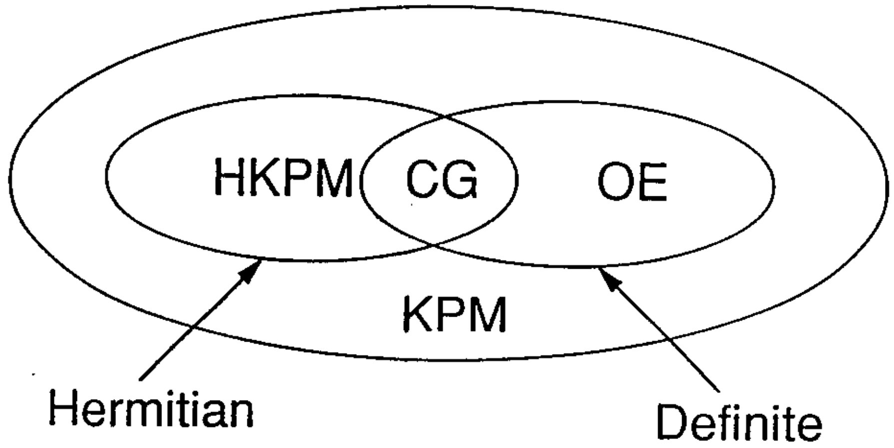
$$\mathbf{R}_i = \mathbf{K}_i(r_0, A)$$

$$\mathbf{L}_i = B^* \mathbf{K}_i(r_0, A)$$

B HPD - Conjugate Gradient Methods

B Definite - Orthogonal Error Methods

B Hermitian - Hermitian Krylov Projection Methods



KRYLOV PROJECTION METHODS

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i$$

$$\underline{p}_i \in \mathbf{K}_i(r_0, A)$$

$$\underline{e}_{i+1} = \underline{e}_i - \alpha_i \underline{p}_i$$

$$\underline{e}_{i+1} \perp B^* \mathbf{K}_i(r_0, A)$$

Step Direction

$$p_i \in \mathbf{K}_i(r_0, A) \ , \ \ p_i \perp_B \mathbf{K}_{i-1}(r_0, A)$$

Step Length

$$\alpha_i = \frac{<B\,\underline{e}_i, \underline{p}_i>}{<B\,\underline{p}_i, \underline{p}_i>}$$

KPM - ALGORITHMS *(Young/Jea)*

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i \qquad \alpha_i = \frac{<B\,\underline{e}_i, \underline{p}_i>}{<B\,\underline{p}_i, \underline{p}_i>}$$

$$\underline{r}_{i+1} = \underline{r}_i - \alpha_i A\,\underline{p}_i$$

ODIR

$$\underline{p}_{i+1} = A\,\underline{p}_i - \sum_{j=0}^{i} \sigma_{ij} \underline{p}_j$$

OMIN

$$\underline{p}_{i+1} = \underline{r}_{i+1} - \sum_{j=0}^{i} \hat{\sigma}_{ij} \underline{p}_j$$

ORES

$$\underline{p}_{i+1} = \underline{r}_{i+1} - \sum_{j=0}^{i} \gamma_{ij} \underline{x}_j$$

KPM - DEFINITE B

Economical Computation *(Faber/Manteuffel, Joubert/Young)*

$$\underline{p}_{i+1} = A\underline{p}_i - (\sigma_{ii}\underline{p}_i + \sigma_{ii-1}\underline{p}_{i-1})$$

or

$$\underline{p}_{i+1} = \underline{r}_i - \hat{\sigma}_{ii}\underline{p}_i$$

or

$$\underline{p}_{i+1} = \underline{r}_i - \gamma_{ii}(\underline{x}_i - \underline{x}_{i-1})$$

If and only if

A is B-normal (1)

KPM - DEFINITE B

Economical Computation

$$<BA\,\underline{x},\underline{y}> = <B\,\underline{x},A^{+}\underline{y}>$$

$$A^{+} = (BAB^{-1})^{*}$$

B normal(s)

- $AA^{+} = A^{+}A$
- A, A^{+} same complete set of B-orthogonal eigenvectors
- $A^{+} = p_S(A)$ polynomial of degree s

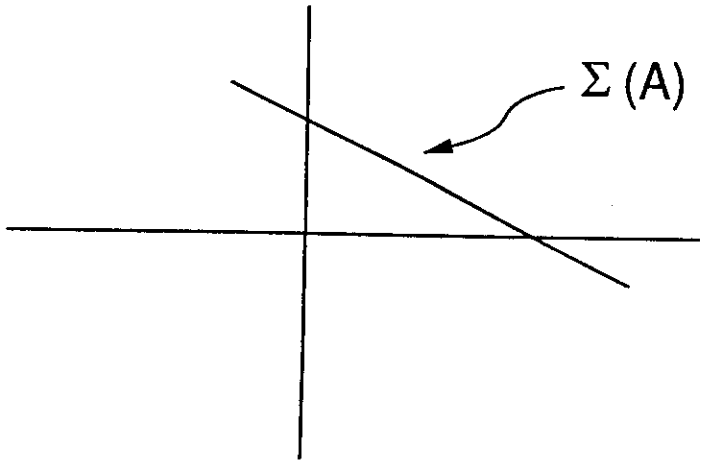
KPM - DEFINITE B

Economical Computation

B normal (1)

$$A^+ = A$$

$$A^+ = \alpha I + \beta A$$



KPM - DEFINITE B EXAMPLES

Orthogonal Error Method *(Concus, Golub, Widlund: CGW)*

$$A\underline{x} = \underline{b}$$

$$A_S = \frac{1}{2}(A + A^*) \qquad A_N = \frac{1}{2}(A + A^*)$$

$$C = A_S^{-1}$$

$$B = A$$

- CA is A-normal (1)

$$A^+ = 2I - A$$

- Computable

$$\alpha_i = \frac{\langle B\underline{e}_i, \underline{p}_i\rangle}{\langle B\underline{p}_i, \underline{p}_i\rangle} = \frac{\langle A\underline{e}_i, \underline{p}_i\rangle}{\langle A\underline{p}_i, \underline{p}_i\rangle} = \frac{\langle \underline{r}_i, \underline{p}_i\rangle}{\langle A\underline{p}_i, \underline{p}_i\rangle}$$

KPM - DEFINITE B EXAMPLES

Conjugate Gradient Method

$$A\underline{x} = \underline{b}$$

$$C = A_S^{-1} A^* A_S^{-1}$$

$$CA = A_S^{-1} A^* A_S^{-1} A$$

$$B = A_S$$

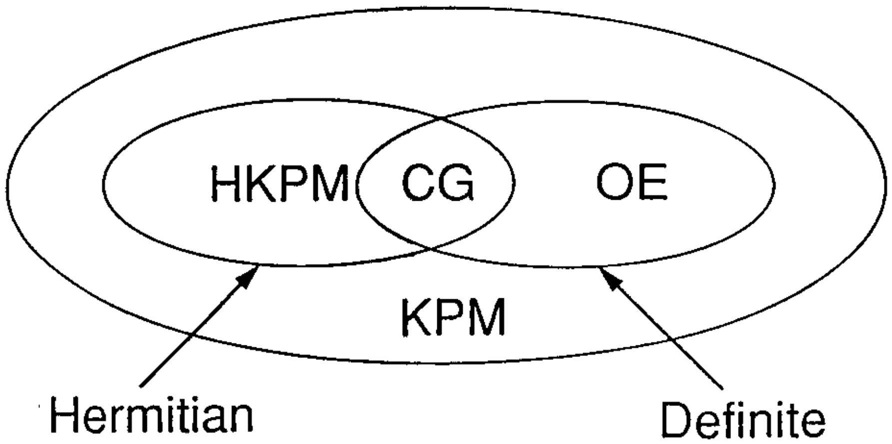
- CA is A-self-adjoint

Result *(Hageman, Luk, Young)*:

For A real even steps of CGW same as this method

KPM - DEFINITE B

- Finite Termination
- No Breakdown
- Bounded Iterates
- Economical Recursion
- Small Applicability



KPM - INDEFINITE B

Biconjugate Gradient Method

$$\hat{A}\underline{x}=\underline{b}\ ,\quad \hat{A}^*\underline{y}=\underline{d}$$

$$\begin{pmatrix}\hat{A} & 0\\ 0 & \hat{A}^*\end{pmatrix}\begin{pmatrix}\underline{x}\\ \underline{y}\end{pmatrix}=\begin{pmatrix}\underline{b}\\ \underline{d}\end{pmatrix}$$

$$A=\begin{pmatrix}\hat{A} & 0\\ 0 & \hat{A}^*\end{pmatrix}\ ,\quad B=\begin{pmatrix}0 & \hat{A}^*\\ \hat{A} & 0\end{pmatrix}$$

$$BA=A^*B\ ,\ A^+=A$$

A is B self-adjoint

KPM - BICONJUGATE GRADIENT METHOD

Lose Boundedness

$$\alpha_i=\frac{<B\,\underline{e}_i,\underline{p}_i>}{<B\,\underline{p}_i,\underline{p}_i>}$$

Risk Breakdown

ODIR

$$\underline{p}_{i+1}=A\,\underline{p}_i-(\sigma_{ii}\underline{p}_i+\sigma_{ii-1}\underline{p}_{i-1})$$

$$\sigma_{ii}=\frac{<BA\,\underline{p}_i,\underline{p}_i>}{<B\,\underline{p}_i,\underline{p}_i>}$$

OMIN

$$\underline{p}_{i+1}=\underline{r}_{i+1}-\hat{\sigma}_{ii}\underline{p}_i$$

$$\sigma_{ii}=\frac{<B\,\underline{r}_{i+1},\underline{p}_i>}{<B\,\underline{p}_i,\underline{p}_i>}$$

KPM - BICONJUGATE GRADIENT METHOD

Breakdown *(Joubert)*

- Depends upon $\underline{e}_0$
- Occurs on set of measure zero

Near Breakdown

- Good definition?
- Measure of Probability?

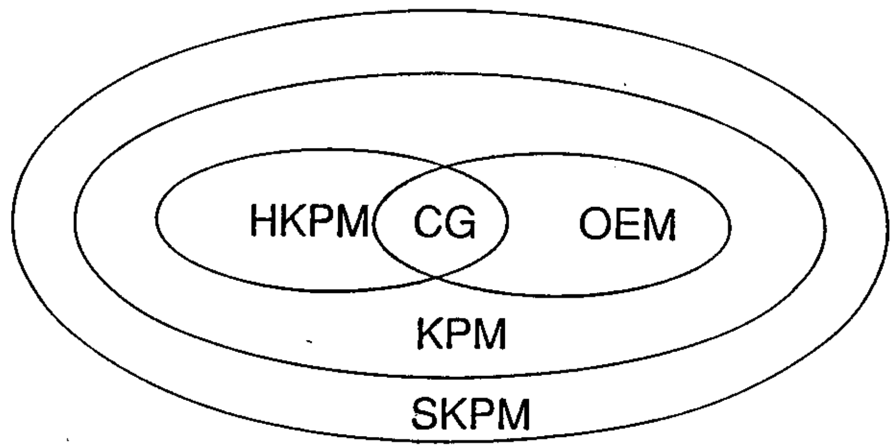
Semi-Krylov Projection Methods

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i$$

$$\underline{p}_i \in \mathbf{R}_i \subseteq \mathbf{K}_i(r_0, A)$$

$$\underline{e}_{i+1} \perp \mathbf{L}_i \subseteq B^* \mathbf{K}_i(r_0, A)$$

Lose finite termination



SKPM - Truncated Methods
(Vinsome, Elman, Young/Jea)

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i \underline{p}_i \qquad \alpha_i = \frac{\langle B\underline{e}_i, \underline{p}_i \rangle}{\langle B\underline{p}_i, \underline{p}_i \rangle}$$

$$\underline{r}_{i+1} = \underline{r}_i - \alpha_i A \underline{p}_i$$

ODIR(S)

$$\underline{p}_{i+1} = A\underline{p}_i - \sum_{j=i-s}^{i} \sigma_{ij} \underline{p}_j$$

OMIN(S)

$$\underline{p}_{i+1} = \underline{r}_{i+1} - \sum_{j=i-s}^{i} \hat{\sigma}_{ij} \underline{p}_j$$

ORES(S)

$$\underline{p}_{i+1} = \underline{r}_{i+1} - \sum_{j=i-s}^{i} \gamma_{ij} \underline{x}_j$$

For example:

$$B = A^* A$$

SKPM - TRUNCATED METHODS

ODIR(S), OMIN(S) Are Balanced Methods

$$\underline{p}_i \in \mathbf{R}_i = \{\underline{p}_{i-s}, \ldots, \underline{p}_i\} \subseteq \mathbf{K}_i(r_0 A)$$

$$\underline{e}_{i+1} \perp_B \mathbf{R}_i$$

ORES(S) Not A Balanced Method

$$\underline{p}_j \in \mathbf{R}_i \subseteq \mathbf{K}_i(r_0, A)$$

$$\underline{e}_{i+1} \perp_B \mathbf{L}_i \neq \mathbf{R}_i$$

SKPM - Restarted Methods

$$\underline{x}_{i+1} = \underline{x}_i + \underline{p}_i$$

$$\underline{p}_i \in \mathbf{R}_i$$

$$\underline{e}_{i+1} \perp_B \mathbf{R}_i \qquad (\text{e.g., } B = A^* A)$$

For $i = 1, \cdots, s$

$$R_i = K_i(r_0, A)$$

For $i = s+1, \cdots, 2s$

$$R_{s+j} = K_j(\underline{r}_s, A) \subseteq K_i(r_0, A)$$

For example: Restarted GMRES
(Saad, Schultz)

OPEN QUESTIONS

HKPM - Indefinite B

Breakdown

- Definition of near breakdown
- Probability
- Fix-ups

SKPM

Relative Merits of various methods

- Convergence Criteria
- Convergence Rates
- Domain of Applicability

PROJECTION METHODS

- Structure
- Onion of Hypotheses
    - Attractive Features
    - Smaller Applicability
- Goal: Explore the consequenses of relaxing each assumption