Design of a Parallel Nonsymmetric Eigenroutine Toolbox

Zhaojun Bai
James Demmel
Ming Gu

Outline

- The nonsymmetric eigenproblem is harder to parallelize than the symmetric one: speed/stability tradeoff
- Survey of previous parallel algorithms
  - Reduction to Hessenberg Form
  - Hessenberg QR Iteration
  - Reduction to Tridiagonal Form (anyway!)
  - Nonsymmetric Jacobi
  - Hessenberg Divide & Conquer
- Spectral Divide & Conquer
- Matrix-Sign function
- Error analysis
- "Black box" vs. "Tool box"
- Preliminary results
- Conclusions and Future Work

Hessenberg Reduction and QR

- **Compute** $H = QAQ^T$ **where** H **is upper Hessenberg**
- **Express** $Q = \Pi_{i=1}^{n-2} P_i$ **using BLAS3:** $\Pi_{i=1}^{b} P_i = I - UTU^T$
- **Hessenberg QR implicitly computes**
  1. $f(A)$, **where** $f(A)$ **nearly singular, easy to compute** $f(A) = A - \sigma I$ **or** $f(A) = (A - \sigma_1 I)(A - \sigma_2 I)$

$$\left.\begin{array}{ll} 2. & f(A) = QR \\ 3. & A = Q^T A Q \end{array}\right\} \text{“bulge chasing”}$$

- **Degree of** $f = d > 2$ **is called “multishift”**
- **Sources of and problems with parallelism**
  1. **Chase** d**-by-**d **bulge: BLAS2 (Bai and Demmel)**
  2. **Chase bulge several columns at a time to exploit BLAS 3: too many flops (Bai and Demmel)**
  3. **Asymptotic superlinear convergence (van de Geijn)**
  4. **But slow convergence because of roundoff (DuBrulle, Watkins)**
  5. **Use of skewed layout (Henry and Van de Geijn)**

Tridiagonal reduction (Anyway!)

- Replace Hessenberg reduction + Hessenberg QR with Tridiagonal reduction + tridiagonal LR
- Lower operation counts, especially tridiagonal LR vs. Hessenberg QR
- Code from Dongarra, Geist and Romine
- Breakdown possible (Parlett, ...)
- Need to monitor stability, restart or permit band growth if needed (Howell, Wachspress ...)
- Speed vs. stability tradeoff
- High precision arithmetic helps with high probability?

Nonsymmetric Jacobi

- Solve sequence of 2 by 2 sub-eigenproblems
- Orthogonal transformations only (Eberlein, Stewart)
  - Converge to Schur form
  - Stable, but slow
- Nonorthogonal transformations (Eberlein, Paardekooper, Sameh, Shroff, Veselić)
  - Converge to diagonal form
  - Asymptotically quadratic, but possibly unstable
- Hybrid (Shroff)
  - Nonorthogonal transformation to normal case: $AA^T = A^T A$
  - Bounded condition number at each step
  - Orthogonal transformations from normal to diagonal
  - If near defective, first step slow
- Standard Jacobi (fine grain) vs. block (coarse grain)
- Speed vs. stability tradeoff

Hessenberg Divide and Conquer

- **Analogous to Cuppen's method for tridiagonals**
- **Divide**

$$H = \left[\begin{array}{c|c} H_{11} & H_{12} \\ \hline h & \\ & H_{22} \end{array}\right] = \left[\begin{array}{c|c} H_{11} & H_{12} \\ \hline 0 & \\ & H_{22} \end{array}\right] + \left[\begin{array}{c|c} 0 & 0 \\ \hline h & \\ & 0 \end{array}\right] \equiv S + E$$

- **Solve S recursively; H_{11} and H_{22} are independent**
- **Merge using Newton (Dongarra and Sidani)**
  - **Solve $g(z, \lambda) = [Hz - \lambda z, e^T z - 1] = 0$ using Newton**
  - **Use solutions for S as starting guesses**
- **Merge subproblems using homotopies (Li, Zeng, Cong)**
  - **homotopy is $H(t) = tS + (1-t)H$, $0 \le t \le 1$**
  - **Solve $g(z, \lambda, t) = [H(t)z - \lambda z, \|z\|_2^2 - 1] = 0$ for $t = 0 \nearrow 1$**
  - **Solve $\det(H(t)) = 0$ for $t = 0 \nearrow 1$**
- **Advantage: Solution often simple (90% of homotopies straight lines)**
- **Disadvantages (Jessup):**
  - **Subproblems made be harder than original**
  - **Multiple copies of single root, or truly multiple?**
  - **Failure of Newton to converge**
  - **Merging of homotopy curves (or tiny step sizes)**
- **Speed vs. stability tradeoff**

Spectral Divide and Conquer

- **Symmetric Case: Auslander, Bischof, Huss-Lederman, Sun, Tsao, Turnbull**
- **Nonsymmetric Case: Bai, Demmel, Laub, Lin, Zmijewski**
- **Algorithm:**
  1. **Choose** $f : \mathbf{C} \rightarrow \{0, 1\}$
  2. **Compute** $f(A)$
  3. $f(A) = QR = [Q_1, Q_2] \cdot \begin{bmatrix} R_{11} & R_{12} \\ & 0 \end{bmatrix}$
  4. $A = Q^T AQ = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$
- **Works because** Q_2 **spans left null space of** $f(A)$ **= left invariant subspace of** A
- **Implementing** f **in the symmetric case:**
  - **Assume** $\lambda(A) \subseteq [0, 1]$
  - $f = g \circ g \circ \cdots \circ g$ **where** $g(x) = 3x^2 - 2x^3$
  - f **maps** $[0, .5)$ **to 0 and** $(.5, 1]$ **to 1**

Matrix Sign Function

- $\text{sign}(x) \equiv \text{sign}(\text{Re}(x))$**:**
  maps left half plane to -1 **and right half plane to** $+1$
- $f(x) = .5(\text{sign}(x) + 1)$**:**
  maps left half plane to 1 and right half plane to 0
- **(Roberts) If the Jordan form of** A **is**

$$A = (X_+, X_-) \begin{bmatrix} J_+ & \\ & J_- \end{bmatrix} (X_+, X_-)^{-1}$$

  with $\text{Re}\lambda(J_+) > 0$ **and** $\text{Re}\lambda(J_-) < 0$ **then**

$$\text{sign}(A) \equiv (X_+, X_-) \begin{bmatrix} I & \\ & -I \end{bmatrix} (X_+, X_-)^{-1}$$

  and

$$f(A) = (X_+, X_-) \begin{bmatrix} I & \\ & 0 \end{bmatrix} (X_+, X_-)^{-1} = X_+ Y_+^T$$

  so

$$f(A) = QR = [Q_+, Q_-] \cdot \begin{bmatrix} R_{11} & R_{12} \\ & 0 \end{bmatrix}$$

  with Q_- **spanning the left invariant subspace for** J_- **and** Q_+ **spanning the right invariant subspace for** J_+

- $\text{trace}(f(A))$ **= # right half plane eigenvalues (so can do bisection)**

Computing the Matrix Sign Function

- **Newton iteration (Roberts, Howland)**

$$A_{i+1} = .5.(A_i + A_i^{-1}) \longrightarrow \mathrm{sign}(A)$$

globally and asymptotically quadratically convergent

- **Equivalent to repeated squaring of Cayley Transform**

$$(A - I)/(A + I)$$

- **Acceleration via scaling (many schemes)**
(Roberts, Balzer, Higham, Byers, Kenney, Laub)

$$A_{i+1} = \alpha_i A_i + \beta_i A_i^{-1}$$

- **Newton-Schultz iteration**

$$A_{i+1} = .5A_i(3I - A_i^2)$$

no inversion; quadratically convergent if $\|A_0^2 - 1\| < 1$

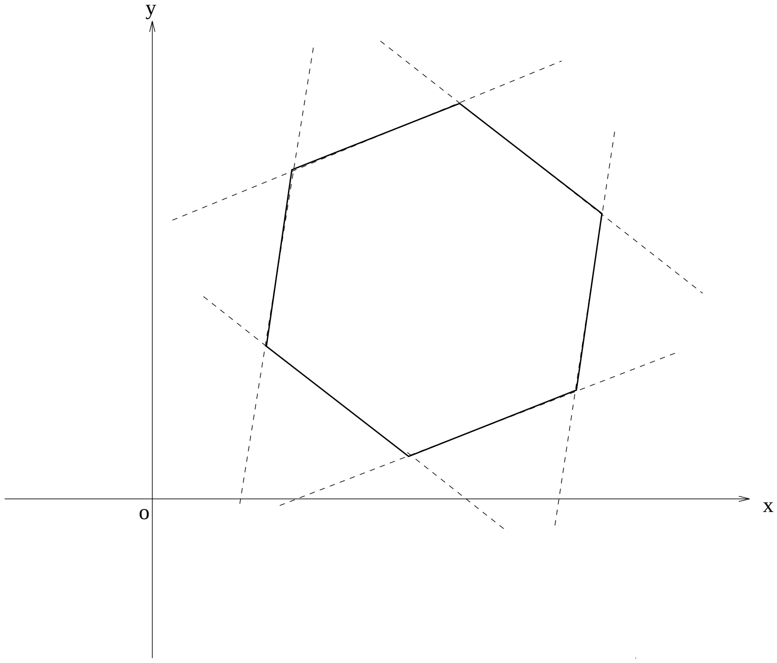
- **Higher order schemes from Padé approximation**
(Kenney and Laub)
- **Advantage: depends only on matrix inversion, matmult**

Spectrum Slicing

- **Want to apply algorithm recursively to A_{11} and A_{22} in**

$$Q^T A Q = \begin{bmatrix} A_{11} & A_{12} \\ & A_{22} \end{bmatrix}$$

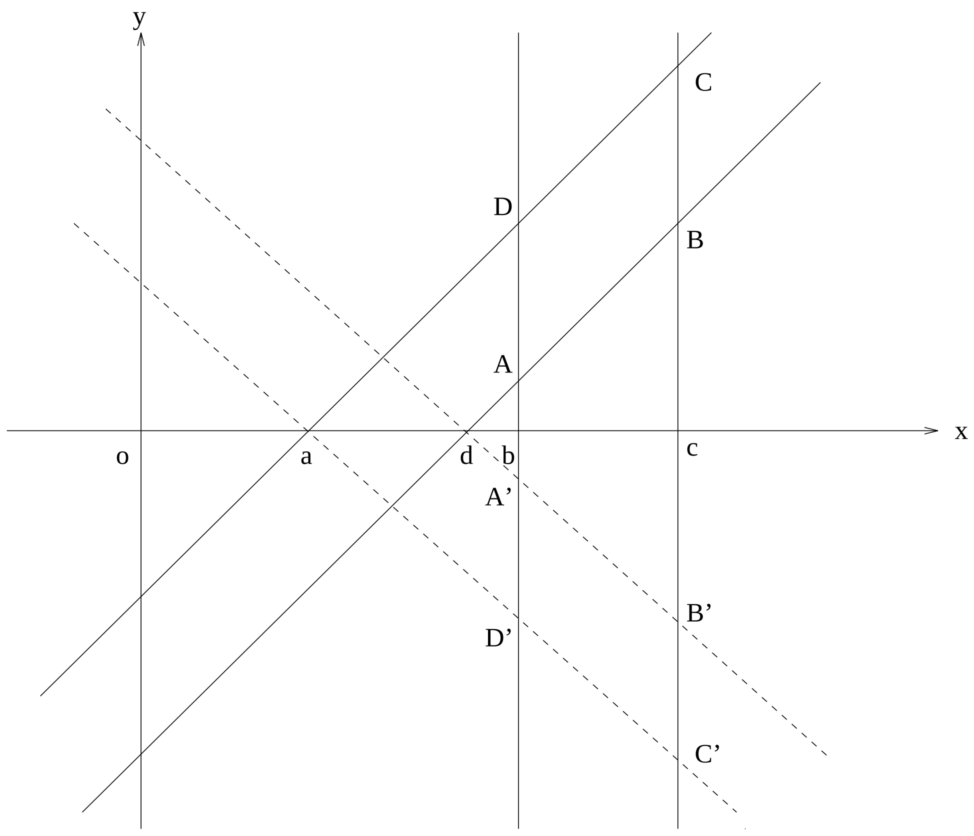
- $\mathrm{sign}(A)$ **only divides right/left halfplane eigenvalues**
- $\mathrm{sign}(e^{i\theta} A + cI)$
  - **Divide along arbitrary line in complex plane**
  - **By repeated divide and conquer, can get any convex subset of C**



- **Advantage: finds interesting part of spectrum only, not the rest**
- **Disadvantage: introduces complex arithmetic even if A real**

Real Spectrum Slicing

- **Avoid complex arithmetic for real A (Howland, Stickel)**
- **Compute $\mathrm{sign}((A - \sigma I)^2)$, σ real**
- **4 sign functions evaluations to deflate parallelogram $ABCD$**
- **Trapezoid, butterfly, strip also possible**



Error Analysis

- When can we expect convergence?
- Notation: $S = \text{sign}(A)$, $\rho = \|S\|$, ε = machine precision
- If $\lambda(A)$ too near pure imaginary, expect trouble
- Example: an n by n Jordan block J with $\lambda = 0$
  any shifted $J - \sigma I$ with $|\sigma| < 1$ very ill-conditioned
- Proposed remedy: Use QR decomposition + condition estimator to deflate eigenvalues $< \varepsilon^{1/2}\|A\|$
- Worst case (J) is staircase algorithm $O(n^4)$, but low probability
- Theorem: If A close to S and $\rho < \varepsilon^{-1/4}$, Newton will compute S with accuracy $\varepsilon^{1/2}\rho$.
- Theorem: Suppose $Ax_i = \lambda_i x_i$ and $y_i^* A = \lambda_i y_i^*$ with $\|x_i\| = \|y_i\| = 1$. Let $s_i = \text{sign}(\text{Re}(\lambda_i))$. Let

  $$\rho' = \max_i \frac{n}{|y_i^* x_i|} \cdot \frac{|\lambda_i + s_i| + |\lambda_i - s_i|}{|\lambda_i + s_i| - |\lambda_i - s_i|} > 1$$

  If $\rho' < \varepsilon^{-1/4}$, Newton will compute S with accuracy $\varepsilon^{1/2}\rho$.
- Disadvantage: slow down near Jordan blocks
- Advantage: Guarantee $\varepsilon^{1/2}$ stability

Inverse Free Spectral Divide and Conquerecompsition

- Based on Malyshev, Bulgakov, Godunov

repeat

$B = I$

$\begin{bmatrix} B \\ -A \end{bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \cdot \begin{bmatrix} R \\ 0 \end{bmatrix}$

$A = Q_{12}^H A$

$B = Q_{22}^H A$

until R **converges**

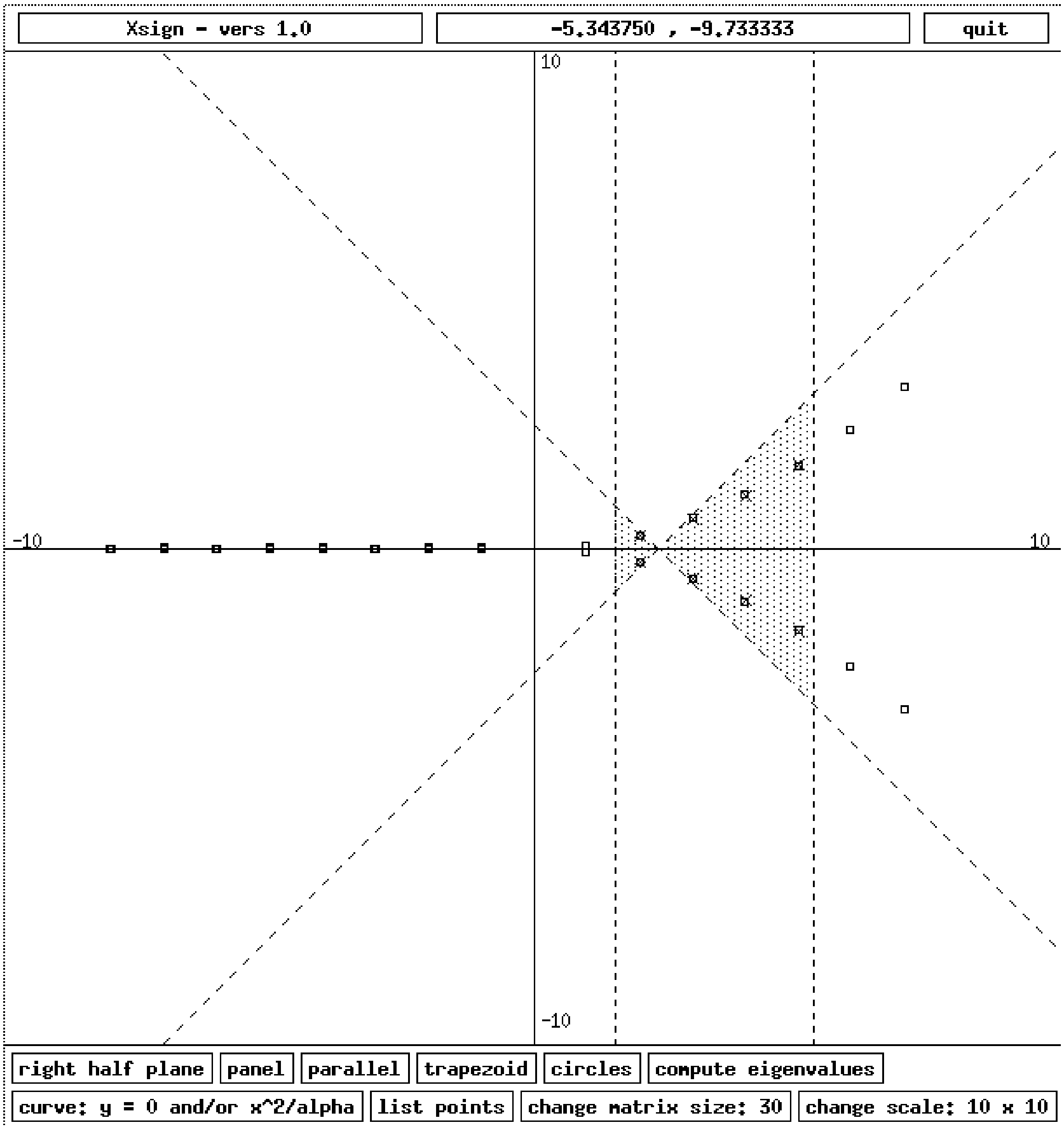
$(A+B)^{-1}A = QRP$ **(implicitly)**

$Q^H A Q = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix}$

- **Implicitly does** $A_{new}^{-1} B_{new} = (A^{-1}B)^2$
- **Divides spectrum along unit circle**
- B **can be anything, not just** I
- $(A, B) \leftarrow (\alpha A + \beta B, \gamma A + \delta B)$ **to split on any circle, line**
- **More expensive than Sign (6x), but more stable: residual in (2,1) position** $\leq O(\epsilon)/dist^2$ **where** $dist$ **= smallest perturbation making** A **"ill-posed", i.e. have eigenvalue on unit circle**

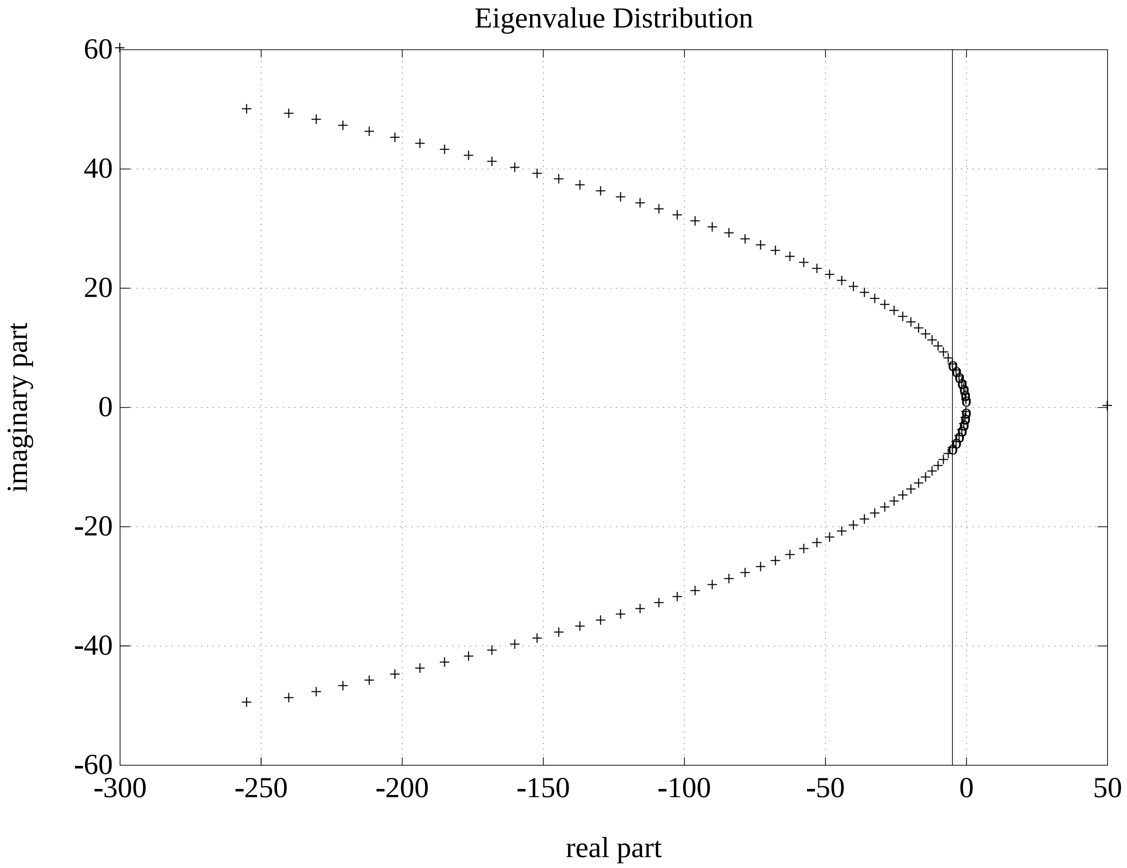
“Black boxes” versus “Tool boxes”

- “Black box” for average users
  - Traditional
  - Puts reliability before speed
  - Hard to find only part of spectrum
  - Hard to incorporate user knowledge
- “Template” or “toolbox” for expert users
  - Requires user intervention
  - Supply building blocks from which user tunes his implementation
  - High performance possible, at cost of reliability for general problems
  - Interface of choice for HPCC applications?
- Unavoidable tradeoff between speed and stability
- Contents of toolbox (for dense problem)
  - Hessenberg reduction and QR
  - Matrix sign-function
  - “Staircase” deflation, condition estimation
  - Iterative refinement with Jacobi-Schur
  - ...
  - X-window interface to choose (Dongarra)


Xsign - vers 1.0
-5.343750 , -9.733333
quit
10
-10
10
-10
right half plane
panel
parallel
trapezoid
circles
compute eigenvalues
curve: y = 0 and/or x^2/alpha
list points
change matrix size: 30
change scale: 10 x 10

Preliminary Results

- **Example 1**
  - **100 eigenvalues on parabola $x = -y^2/10$, cond = 10^6**
  - **Compute sign function of $A + 5I$**
  - **Get 14 rightmost eigenvalues to 11 digits out of 16**
  - **Cost is $31n^3$ flops, versus $22n^3$ for full QR**



- **Example 2**
  - **Random, normal matrices on Cray Y-MP**
  - **Eigenvalues good to 9 digits**

matrix order	Half plane Speedup QR/Sign	Strip Speedup QR/Sign	Parallelogram Speedup QR/sign
50	1.38	1.44	0.98
100	1.65	1.22	1.38
200	1.40	1.46	1.25
300	1.51	1.41	1.33
400	1.60	1.33	1.38

Conclusions and Future Work

- Parallelism and stability tradeoff
- Tool boxes, not black boxes
- Future work
  - Continue building basic routines (Scalapack)
  - Make tools easy to assemble (X-windows)
  - Refine error analysis
  - Test on realistic problems (geometric design)
  - Extend to generalized eigenproblem (Malyshev)
- To get a paper:

  anonymous ftp to toe.berkeley.edu
  cd pub/tech-reports/cs/csd-92-703
  get csd-92-703.ps.Z