A Tutorial on Numerical Stability

James Demmel
UC Berkeley
Math and EECS Depts.

Outline (1/2)

1. Basic Definitions: Forward, backward, mixed stability
2. Design space
   - How to measure errors
     - relative vs absolute, norm, componentwise, structured, deterministic vs randomized
   - How to model arithmetic
     - $(1+\delta)$, + underflow, + BlackBox, floating point, rounding, precisions

Outline (2/2)

3. Examples
   - Dot products, matmul
   - GE + variations
   - Algorithms using orthogonal tranformations
   - Symmetric eigenproblem: Bisection, D&C, MRRR
   - Fast ($O(n^\omega)$) matmul
   - Fast linear algebra, via logarithmic stability
   - Exploiting problem structure (many kinds!)

Basic Definitions (for scalar functions)

- Want $y = f(x)$, have an algorithm $\hat{y} = \hat{f}(x)$
- Forward stability: a bound on $|y - \hat{y}|$ (see metrics below)
- Backward stability: a bound on $|x - \hat{x}|$ where $\hat{y} = f(\hat{x})$
- Mixed stability: a bound on $|x - \hat{x}|$ and $|\hat{y} - \hat{\hat{y}}|$ where $\hat{\hat{y}} = f(\hat{x})$
  - Good if both small: "Almost the right answer ($\hat{y}$ instead of $\hat{\hat{y}}$) to almost the right problem ($\hat{x}$ instead of x)"
- Error metrics
  - Absolute: $|y - \hat{y}| \leq \eta$ for some $\eta \geq 0$
  - Relative: $|y - \hat{y}|/|y| \leq \epsilon$ for some $\epsilon \geq 0$
  - Mixed: $|y - \hat{y}| \leq \epsilon|y| + \eta$ (eg. used to handle underflow)
  - Bounds on ϵ and η: multiply bound on $|x - \hat{x}|$ by condition number to get a bound on $|y - \hat{y}|$

More Metrics (for vector and matrix functions) (1/2)

- Write $\hat{x} = x + \delta x$, $\hat{A} = A + \delta A$, etc
- Normwise vs componentwise: $\|\delta A\| / \|A\|$ vs $\| |\delta A| ./ |A| \|_{\max}$
  - Both kinds of (small) backward error bounds for solving $Ax = b$ (xgesvx in LAPACK), and smaller componentwise condition number: $\| |A^{-1}| \cdot |A| \|$ vs. $\|A^{-1}\| \cdot \|A\|$
  - Thm (D., Higham; Rump) Componentwise distance to singularity "close" to $1 / \| |A^{-1}| \cdot |A| \|$
  - Extends to general $E \geq 0$ instead of $|A|$; distance is NP-hard (Rohn, Poljak)

More Metrics (for vector and matrix functions) (2/2)

- Structured: If A Symmetric/Bidiagonal/Vandermonde/Totally Positive/... then so is $\hat{A}$
  - Condition numbers can be arbitrarily smaller in some cases
  - Ex: Bidiagonal SVD (xbdsqr in LAPACK) (D., Kahan)
- Randomized vs Deterministic
  - Guarantees a la Johnson-Lindenstrauss: "With probability at least $1 - \delta$ the error is at most ϵ"
  - See arxiv.org/abs/2302.11474 for a 195 page design document for RandLAPACK

How to Model Arithmetic (1/2)

- Traditional model: $\text{rnd}(a \text{ op } b) = (a \text{ op } b)(1+\delta)$, $|\delta| \le \epsilon \ll 1$
  - But new 8-bit IEEE floating point standard in progress, with $\epsilon = 1/8$ or $1/16$
  - Will (likely) support mixed precision dot products, so $\epsilon = 1/256$ or $1/2048$
  - Nvidia has tried 0 mantissa bits (all numbers are $\pm(\sqrt{2})^e$)
  - Committee meeting biweekly, lots of companies want a standard
- Traditional model + underflow:
  - $\text{rnd}(a \text{ op } b) = (a \text{ op } b)(1+\delta) + \eta$, $|\delta| \le \epsilon$, $|\eta| \le UN$
  - See (D, 1984) for extensions of classical error analysis to include underflow
- Traditional model extends to complex arithmetic, with larger ϵ

How to Model Arithmetic (2/2)

- Traditional model + "black boxes"
  - Ex: Fused-multipy-add (FMA): $\mathrm{rnd}((a \cdot b) + c) = ((a \cdot b) + c)(1 + \delta)$, $|\delta| \leq \epsilon \ll 1$
  - Many others possible; many accelerators (eg for matmul) being built
  - Ex: What could we do with an accurate dot product?
- Floating point: $\pm m \cdot 2^e$, with a rounding rule to determine δ, η
  - Traditional model applies (some exceptions pre-IEEE 754)
  - If conventional rounding (eg to nearest) then many tricks to extend precision (examples later)
  - New 8-bit standard will also support stochastic rounding, to reduce some error bounds from $O(n\epsilon)$ to $O(\sqrt{n}\epsilon)$
    * See survey on stochastic rounding by Croci et al

Some floating point tricks for higher precision

- Two-Sum
  - Assume $|x| \leq |y|$: $head = x + y$, $tail = y - (head - x)$
  - Thm: $head + tail = x + y$ exactly
  - $head$ = leading bits, $tail$ = trailing bits
- Two-Product
  - $head = a \cdot b$, $tail = \text{fma}(a, b, -head) = a \cdot b - head$
  - Thm: $head + tail = x \cdot y$ exactly
- Long history of extensions to compute in higher precision
  - Higham, Priest, Dekker, Rump, Kahan, ...

Computing Sums $s = \sum_{i=1}^n x_i$ (1/2)

- Conventional (sequential) summation
  - $\hat{s} = x_1$, for $i = 2 : n$, $\hat{s} = \mathrm{rnd}(\hat{s} + x_i) = (\hat{s} + x_i)(1 + \delta_i)$
  - $\hat{s} = \sum_{i=1}^n [x_i \prod_{j=\max(i,2)}^n (1 + \delta_j)]$, $|\delta_j| \le \epsilon$
  - $1 - \frac{n\epsilon}{1-n\epsilon} \le \prod_{j=1}^n (1 + \delta_j)^{\pm 1} \le 1 + \frac{n\epsilon}{1-n\epsilon}$ if $n\epsilon < 1$
  - $\hat{s} = \sum_{i=1}^n x_i(1 + \bar{\delta}_i)$, $|\bar{\delta}_i| = O(n\epsilon) \Rightarrow$ backward stable
  - $|s - \hat{s}| \le \sum_{i=1}^n |x_i \bar{\delta}_i| = O(n\epsilon) \sum_{i=1}^n |x_i| \Rightarrow$ forward stable
    - Condition number for relative error $= \sum_{i=1}^n |x_i| / |\sum_{i=1}^n x_i|$
- Conventional (sequential) summation with randomized rounding
  - Round up or down with probability $\propto$ distance to other choice
  - $O(n\epsilon) \Rightarrow O(\sqrt{n}\epsilon)$ w.h.p. (Central Limit Thm) (Croci et al)
- Parallel summation with a binary tree: $O(n) \Rightarrow O(\log n)$
- Compensated summation (Kahan) : $O(n) \Rightarrow 2$ ($4n$ flops)

Computing Sums $s = \sum_{i=1}^{n} x_i$ (2/2)

- Guaranteeing a small relative error, despite cancellation
  - Obvious approach: very large ("super") accumulator
    - Time, mem cost exponential in input size (#exponent bits)
  - Faster approach:
    - Sort x_i in order of decreasing exponent (or magnitude)
    - Sum from x_1 to x_n using k extra mantissa bits
    - Thm (D, Hida; Priest): If $n \le 1 + 2^k$, relative error $\lessapprox 1.5\epsilon$
- Guaranteeing bitwise reproducibility for any summation order
  - Modern systems nondeterministic $\Rightarrow$ summation order can vary
  - Of interest for scientific, legal, political reasons ...
  - Thm (Ahrens, Nguyen, D.): Cost of reproducible summation $= 9n$ flops, $3n$ bit-wise ops, 6 word accumulator

Computing Dot Products $s = \sum_{i=1}^{n} x_i \cdot y_i$, Classical Matmul, Some Other NLA Algorithms

- Prior approaches apply (some require $x_i \cdot y_i = \mathit{head} + \mathit{tail}$)
- Conventional (sequential) summation for dot products
  - $\hat{s} = \sum_{i=1}^{n} x_i \cdot y_i (1 + \bar{\delta}_i)$, $|\bar{\delta}_i| = O(n\epsilon) \Rightarrow$ backward stable
  - $|s - \hat{s}| = O(n\epsilon) \sum_{i=1}^{n} |x_i \cdot y_i| \Rightarrow$ forward stable
- Conventional (sequential) summation for $C = A \cdot B$
  - $|C - \hat{C}| = O(n\epsilon)|A| \cdot |B| \Rightarrow$ forward stable
  - $\|C - \hat{C}\| = O(n^k \epsilon)\|A\| \cdot \|B\|$, k depends on norm
  - **Not** back. stable in general ($O(n^3)$ constraints on $O(n^2)$ data)
  - Unless $A \cdot A^T = I$: $\hat{C} = C + \delta C = A(B + A^T \delta C) = A(B + \delta B)$, $\|\delta B\|_2 = \|\delta C\|_2 = O(n^k \epsilon)\|B\|_2$
  - All algorithms based on orthogonal tranformations (QR, eig, SVD,...) are normwise backward stable

More on symmetric tridiagonal eigensolvers

- $T = T^T$, $n \times n$ and tridiagonal
- Bisection for eigenvalues of T (D., Dhillon, Ren)
  - Compute Inertia$(T - \sigma I)$ = #pos,zero,neg D_{ii} = #evals of T that are $> \sigma$, $= \sigma$, $< \sigma$, where $T = LDL^T$
  - Expect these counts to be monotonic in σ for correctness
  - Thm: Counts are monotonic if floating point is: $a_1 \text{ op } b_1 \geq a_2 \text{ op } b_2 \rightarrow \text{rnd}(a_1 \text{ op } b_1) \geq \text{rnd}(a_2 \text{ op } b_2)$.
- MRRR for eigenvalues and eigenvectors of T (Dhillon, Parlett)
  - Goal: $O(mn)$ flops to stably compute m pairs (λ_i, v_i): $\|Tv_i - \lambda_i v_i\| = O(\epsilon)\|T\|$ and $|v_i^T v_j| = O(\epsilon)$
  - Simple algorithm: Bisection + Inverse Iteration can fail
  - MRRR = Multiple Relatively Robust Representations meant to fix this, usually works, still some rare failures to be fixed

LU, triangular factorizations (1/4)

- Factor $P_r A P_c = A' = LU$, solve $Ax = b$ using substitution
- $A' + \delta A' = LU$, $|\delta A'| = O(n\epsilon)|L| \cdot |U|$
- $(A' + \delta A'')\hat{x} = b$, $|\delta A''| = O(n\epsilon)|L| \cdot |U|$
- Normwise backward stability depends on $\| |L| \cdot |U| \| / \|A\|$
- Instead use $Growth\_factor = |$ largest intermediate result $| / \|A\|_{max}$
- General A, Partial Pivoting (PP)
  - P_r chooses $|A'_{11}| = \max_i |A'_{i1}|$, ditto for later columns, $P_c = I$
  - $L_{ii} = 1$, $|L_{ij}| \le 1$, #comparisons $= n(n-1)/2$
  - Growth_factor $\le 2^{n-1}$, unstable but rare
  - Statistical models and experiments support growth_factor $=$ $O(n^{2/3})$ or $O(n^{1/2})$ (Trefethen, Schreiber)(Huang, Tikhomirov)

LU, triangular factorizations (2/4)

- General A, Rook Pivoting (RP)
  - P_r, P_c choose $|A'_{11}| = \max_i |A'_{i1}| = \max_i |A'_{1i}|$, ditto for later steps
  - $A' = LDU$, $L_{ii} = U_{ii} = 1$, $|L_{ij}| \le 1$, $|U_{ij}| \le 1$
  - #comparisons usually like PP, can be $\Theta(n^3)$, unlikely
  - E(#comparisons) $\le en(n-1)/2$
  - Growth_factor $\le 1.5 n^{\frac{3}{4}\ln n} \ll 2^{n-1}$
- General A, Complete Pivoting (CP)
  - P_r, P_c choose $|A'_{11}| = \max_{ij} |A'_{ij}|$, ditto for later steps
  - #comparisons $= n^3/3 + O(n^2)$
  - Growth_factor $= O(n^{\frac{2+\ln n}{4}})$
  - Was long conjectured to be n, a few counterexamples found

LU, triangular factorizations (3/4)

- General A, Randomized with No Pivoting (NP)
  - Perform LU with NP on $B_r \cdot A \cdot B_c$ (Baboulin et al)
    - B_r and B_c are random butterfly matrices
    - One level $= B^n = 2^{-1/2} \begin{bmatrix} D_0 & D_1 \\ D_0 & -D_1 \end{bmatrix}$
      - $D_k(i,i)$ random in $[.95, 1.05]$, well-conditioned
    - Two level $= \begin{bmatrix} B^{n/2} & 0 \\ 0 & B^{n/2} \end{bmatrix} \cdot B^n$, etc.
    - Only use a few levels, cheap to apply or invert
    - Backward stable (and faster) in practice
  - Perform LU with NP on $V_r \cdot A \cdot V_c$ (D., Grigori, Rusciano)
    - V_r and V_c are Haar matrices
    - Thm: $E(\log(\text{Growth\_factor})) = O(\log n)$

LU, triangular factorizations (4/4)

- General A, Tournament Pivoting (TP) (Grigori et al)
  - Choose b rows from group of b columns, access data once
  - Choose subset of b rows from $2b$ rows at a time, do reduction
  - Allows LU to attain communication lower bound
  - Schur complement at each step same as PP applied to different matrix built from A, so as "stable" as PP
  - Thm: If the tournament reduction tree height $\leq H$, Growth_factor $\leq 2^{n(H+1)-1}$

Fast ($O(n^\omega)$) Matmul is Stable (D., Dumitriu, Holtz)

- Stationary Partition Algorithms for $C = A \cdot B$
  - Recursively apply formula for $k \times k$ matmul:
  - $c_{hl} = \sum_{s=1}^{t} w_{rs} P_s$ where $P_s = (\sum_{i=1}^{k^2} u_{is} x_i)(\sum_{j=1}^{k^2} v_{js} y_j)$
  - x_i (resp y_i) are entries of A (resp B) ordered columnwise
  - Includes Strassen, many others
  - $\|\hat{C} - C\| \le \mu(n)\epsilon \|A\| \|B\| + O(\epsilon^2)$
  - $\mu(n) = O(n^{\log_k(e_{max}\|U\|\|V\|\|W\|)+o(1)}) = poly(n)$
  - U, V, W are matrices of coefficients (generalizes Bini, Lotti)
  - e_{max} depends on the sparsity structures of U, V, W
- Extends to Non-stationary partition algorithms
- Extends to pre-and post-processing of A and B
- Extends to group-theoretic recursive algorithms (Cohn, Umans)

Fast Linear Algebra is Stable (1/5) (D., Dumitriu, Holtz)

- Logarithmic Stability: $\frac{\|\hat{f}(x)-f(x)\|}{\|f(x)\|} \le O(\epsilon)\kappa_f^{polylog(n)}(x) + O(\epsilon^2)$
- Getting usual error bound increases precision and complexity by $polylog(n)$
- Inverting triangular matrix recursively costs $O(n^\omega)$, log. stable

$$\begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix}^{-1} = \begin{bmatrix} T_{11}^{-1} & -T_{11}^{-1} \cdot T_{12} \cdot T_{22}^{-1} \\ 0 & T_{22}^{-1} \end{bmatrix}$$

- Ditto for recursive matrix inversion for $M^{-1} = (M^T M)^{-1} \cdot M^T$

$$H = \begin{bmatrix} A & B \\ B^T & C \end{bmatrix} = \begin{bmatrix} I & 0 \\ B^T A^{-1} & I \end{bmatrix} \cdot \begin{bmatrix} A & B \\ 0 & S \end{bmatrix}, S = C - B^T A^{-1} B$$

$$H^{-1} = \begin{bmatrix} A^{-1} + A^{-1} B S^{-1} B^T A^{-1} & -A^{-1} B S^{-1} \\ -S^{-1} B^T A^{-1} & S^{-1} \end{bmatrix}$$

Fast Linear Algebra is Stable (2/5)

- Recursive (left-right) QR costs $O(n^{\omega})$, stable (not log.)
  - Do QR on left half of A (recursively)
  - Update right half of A
  - Do QR on lower right of A (recursively)
- Recursive (left-right) GEPP costs $O(n^{\omega})$, stable if $\|L^{-1}\|$ bounded
  - Ditto

Fast Linear Algebra is Stable (3/5) (Ballard et al)

- Background on eigensolvers: matrix-sign function
- Use Newton to solve $x^2 = 1$: $x_{n+1} = (x_n + x_n^{-1})/2 \to \mathrm{sign}(\Re(x_0))$
- $(I \pm (A_n + A_n^{-1})/2)/2 \to P_\pm =$ spectral projector for $\Re(\lambda) \gtrless 0$
- Do RRQR (Rank-Revealing QR):
  - $VR = G =$ Gaussian, so V Haar
  - $P_+ V^T = UR$, so $P_+ = URV$
- Update $A \leftarrow U^T A U = \begin{bmatrix} A_+ & A_{12} \\ O(\epsilon) & A_- \end{bmatrix}$, stable if really $O(\epsilon)$
- Apply to $\alpha A_\pm + \beta I$ to divide-and-conquer spectrum
- Newton = repeated squaring of Cayley Transform $(A-I)(A+I)^{-1}$

Fast Linear Algebra is Stable (4/5)

- Inverse-free repeated squaring of $A^{-1}B$ ($A_0 = A$, $B_0 = I$)

$$\begin{bmatrix} B_j \\ -A_j \end{bmatrix} = \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \cdot \begin{bmatrix} R_j \\ 0 \end{bmatrix}, \quad \begin{array}{l} A_{j+1} = Q_{12}^T \cdot A_j \\ B_{j+1} = Q_{22}^T \cdot B_j \end{array}$$

- $A_{j+1}^{-1} B_{j+1} = (A_j^{-1} B_j)^2$
- Need RRQR of $P_\circ \approx (I + (A_j^{-1} B_j))^{-1} = (A_j + B_j)^{-1} A_j$
  - $A_j = URV$ (V Haar), $\hat{R}Q = U^T (A_j + B_j)$
  - $\Rightarrow (A_j + B_j)^{-1} A_j = Q^T (\hat{R}^{-1} R) V$
- Apply to $(aA + bI)^{-1}(cA + dI)$ to split spectrum on circles
- Applies to pencils $A - \lambda B$
- All Matmul, QR; finite precision analysis w.i.p.

Fast Linear Algebra is Stable (5/5) (Banks et al)

- Shattering Approach: Add noise $A + \gamma G$, G Gaussian
  - W.h.p. separates close eigenvalues of $A = VDV^{-1}$ so V well-conditioned
- Can accurately compute matrix-sign function using Newton
  - Do binary search on 2D grid to find good split
- Cost increases/backward error decreases as γ decreases
  - Attaining $\|A - VDV^{-1}\| \leq \delta$ and $\kappa(V) \leq 32n^{2.5}/\delta$ costs $O(n^{\omega} polylog(\frac{n}{\delta}))$ arithmetic or bit operations

Exploiting Structure for Higher Accuracy (1/6)

- If my problem is structured (symmetric/sparse/diagonally dominant/Vandermonde/...) can I get a more accurate answer? Or a structured backward error?
- *Many* possibilities, will show a few
- Solving $Ax = b$ using Cholesky
  - Thm (van der Sluis): If A spd, choosing diagonal D so $\hat{A} = DAD$ has $\hat{A}_{ii} = 1 \Rightarrow \kappa(\hat{A}) \leq n \cdot \min_D \kappa(DAD)$
  - $\kappa(\hat{A})$ can be $\ll \kappa(A)$
  - Let $\hat{x}$ be computed solution: $\|D^{-1}(x - \hat{x})\| / \|D^{-1}\hat{x}\| = O(\epsilon)\kappa(\hat{A})$
  - $1/\kappa(\hat{A}) \approx$ smallest componentwise relative perturbation that makes $A + \delta A$ singular

Exploiting Structure for Higher Accuracy (2/6)

- Iterative Refinement
  - Solve $Ax = b$, repeat until "convergence":
    $r = b - Ax$, solve $Ad = r$, $x = x + d$
  - (Approximate) Newton on a linear system
- Version 1: Use GEPP, compute r in double precision
  - x converges to true solution in norm if $\kappa(A)\epsilon \lessapprox 1$
- Version 2: Use GEPP, compute r in single precision (Skeel)
  - x converges to small componentwise relative backward error $\max_i |r_i|/(|A| \cdot |x| + |b|)_i$, if condition number not too large
  - Condition number $\Rightarrow \||A^{-1}| \cdot |A| \cdot |x|\|/\|x\| \leq \||A^{-1}| \cdot |A|\|$
- Version 3++: Different solvers, convergence criteria, multiple precisions (3 or even 5)

Exploiting Structure for Higher Accuracy (3/6)

- When is high relative accuracy possible in the traditional model? $\mathrm{rnd}(a \text{ op } b) = (a \text{ op } b)(1+\delta)$, $|\delta| \le \epsilon \ll 1$
- $x^2 - y^2 = (x-y)(x+y)$: possible
- $x + y + z$: impossible
- Motzkin polynomial: $z^6 + x^2y^2(x^2 + y^2 - 3z^2)$?

Exploiting Structure for Higher Accuracy (3/6)

- When is high relative accuracy possible in the traditional model? $\mathrm{rnd}(a \text{ op } b) = (a \text{ op } b)(1+\delta)$, $|\delta| \le \epsilon \ll 1$
- $x^2 - y^2 = (x-y)(x+y)$: possible
- $x + y + z$: impossible
- Motzkin polynomial: $z^6 + x^2y^2(x^2 + y^2 - 3z^2)$: possible!

$$
\begin{array}{rl}
\text{if} & |x-z| \le |x+z| \wedge |y-z| \le |y+z| \\
p = & z^4 \cdot [4((x-z)^2 + (y-z)^2 + (x-z)(y-z))] + \\
 & +z^3 \cdot [2(2(x-z)^3 + 5(y-z)(x-z)^2 + 5(y-z)^2(x-z) + \\
 & \qquad 2(y-z)^3)] + \\
 & +z^2 \cdot [(x-z)^4 + 8(y-z)(x-z)^3 + 9(y-z)^2(x-z)^2 + \\
 & \qquad 8(y-z)^3(x-z) + (y-z)^4] + \\
 & +z \cdot [2(y-z)(x-z)((x-z)^3 + 2(y-z)(x-z)^2 + \\
 & \qquad 2(y-z)^2(x-z) + (y-z)^3] + \\
 & +(y-z)^2(x-z)^2((x-z)^2 + (y-z)^2) \\
\text{else} & \ldots\ 7 \text{ more analogous cases}
\end{array}
$$

Exploiting Structure for Higher Accuracy (4/6)

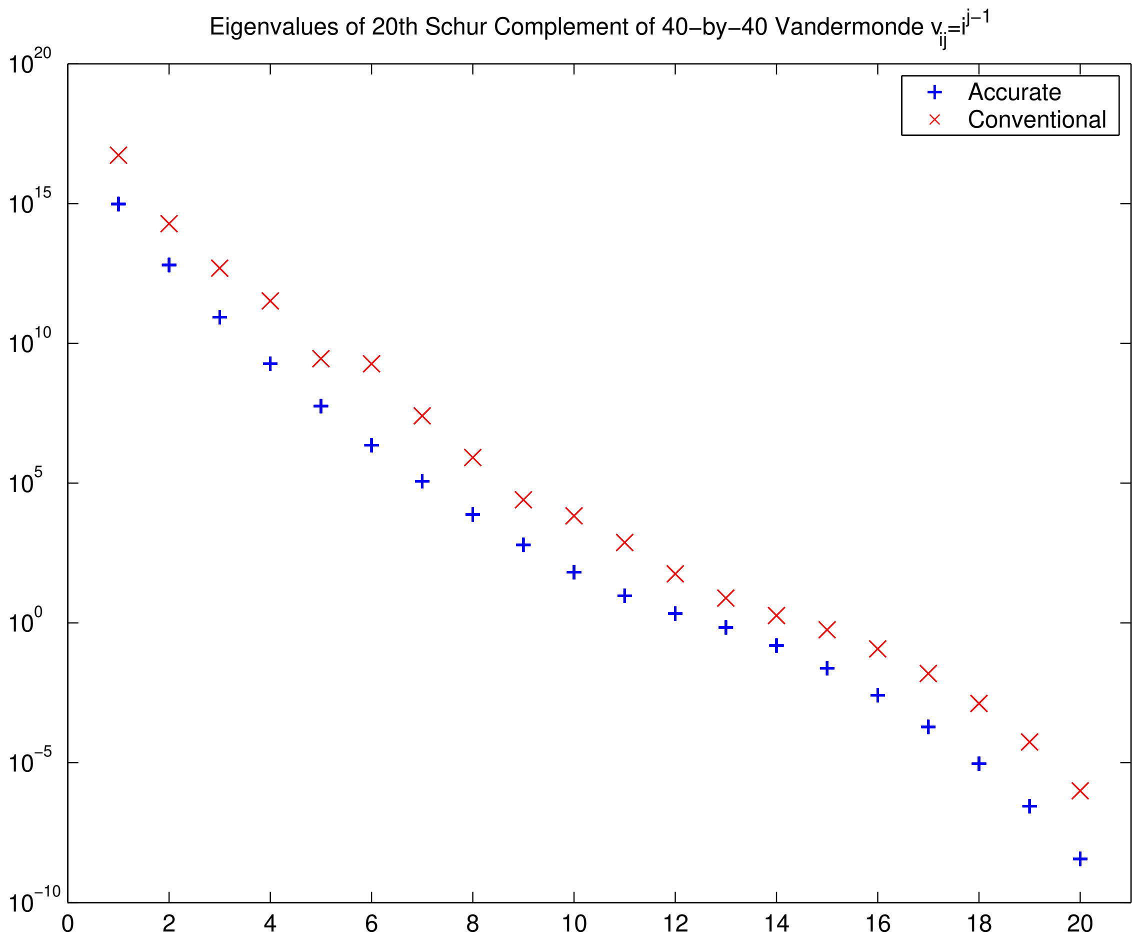
- Evaluating $p(x)$ accurately depends on its variety $V(p)$
- Def: $V(p)$ is *allowable* if it is a finite union of intersections of basic allowable sets:
  - $Z_i = x : x_i = 0$, $S_{ij} = x : x_i + x_j = 0$, $D_{ij} = x : x_i - x_j = 0$
- Thm: $V(p)$ unallowable $\Rightarrow$ p cannot be evaluated accurately on $\mathbb{R}^n$ or $\mathbb{C}^n$ (can be extended to smaller domains)
- Ex: $V(\text{Motzkin}) = \{|x| = |y| = |z|\}$
- Thm: On $\mathbb{C}^n$, $V(p)$ allowable is also sufficient for accurate evaluation ($p(x)$ factors into x_i, $x_i \pm x_j$)
- Real case: some progress toward decision procedure (D., Dumitriu, Holtz, Koev)
- Ideas extend to adding "black boxes" etc, FMA, dot-products, ...

Exploiting Structure for Higher Accuracy (5/6)

Type of matrix	$\det A$	A^{-1}	Any minor	Gauss. elim. NP	PP	CP	RRD	QR	NE	$Az=b$	SVD	EVD
Acyclic	n	n^2	n	n^2	n^2	n^2	n^2				n^3	
DSTU	n^3	n^5	n^3	n^3	n^3	n^3	n^3				n^3	
TSC	n	n^3	n	n^4	n^4	n^4	n^4				n^4	
Diagonally dominant	n^3		No	n^3		n^3	n^3				n^3	
M-matrices	n^3	n^3	No	n^3		n^3	n^3				n^3	
Cauchy (non-TN)	n^2	n^2	n^2	n^2	n^3	n^3	n^3		n^2		n^3	
Vandermonde (non-TN)	n^2		No				n^3		n^2		n^3	
Displacement rank one	n^2						n^3				n^3	
Totally nonnegative	n	n^3	n^3	n^3	n^4	n^4	n^3	n^3	0	n^2	n^3	n^3
TN^J	n	n^3	n^3	n^3	n^4	n^4	n^3	n^3	0	n^2	n^3	n^3
Toeplitz	No		No	No	No	No	No	No	No		No	No

Exploiting Structure for Higher Accuracy (6/6)

- Eigenvalues of the 20th Schur Complement of the 40-by-40 Vandermonde matrix $V_{ij} = i^{j-1}$, computed both using a Conventional algorithm (x) and and Accurate algorithm (+)



References (1/5)

- N. Higham, “Accuracy and Stability of Numerical Algorithms”, 2nd ed., 2002
- D., “The componentwise distance to the nearest singular matrix,” SIMAX, 1992
- S. Rump, “Ill-conditioned matrices are componentwise near to singularity,” SIAM Review 1999
- S. Poljak, J. Rohn, “Checking robust singularity is NP Hard,” Math. Controls Signals Systems, 1993
- D., W. Kahan, “Accurate Singular Values of Bidiagonal Matrices,” SISC, 1990
- R. Murray et al, “Randomized Numerical Linear Algebra: A Perspective on the Field With an Eye to Software,” arxiv:2302.11474

References (2/5)

- D., "Underflow and the Reliability of Numerical Software," SISC, 1984
- M. Croci et al, "Stochastic Rounding: implementation, error analysis and applications," Royal Society Open Science, 2022
- D. Priest, "Algorithms for Arbitrary Precision Floating Point Arithmetic," 10th IEEE Symp. Comp. Arith., 1991
- D. Priest, UC Berkeley PhD Thesis, 1992
- T. Dekker, "A floating-point technique for extending the available precision," Num. Math., 1971
- S. Rump, "Ultimately Fast Accurate Summation," SISC, 2009
- W. Kahan, "Further Remarks on Reducing Truncation Errors," CACM, 1965

References (3/5)

- D., Y. Hida, “Accurate and Efficient Floating Point Summation,” SISC, 2003
- P. Ahrens et al, “Efficient Reproducible Floating Point Summation,” ACM TOMS, 2020
- D., I. Dhillon, H. Ren, “On the correctness of some bisection-like parallel eigenvalue algorithms in floating-point arithmetic,” ETNA 1995
- I. Dhillon, B. Parlett, “Orthogonal eigenvectors and Relative Gaps,” SIMAX 2004
- L. Trefethen, R. Schreiber, “Average-case stability of Gaussian Elimination,” SIMAX 1990
- H. Huang, K. Tikhomirov, “Average-case analysis of the Gaussian Elimination with Partial Pivoting,” arXiv:2206.01726

References (4/5)

- M. Baboulin et al, "Accelerating linear system solutions using randomization techniques," ACM TOMS, 2013
- D., L. Grigori, A. Rusciano, "An improved analysis and unified perspective on deterministic and randomized low rank matrix approximation," arXiv:1910.00223 (to appear in SIMAX)
- L. Grigori, D., H. Xiang, "CALU: A communication optimal LU factorization algorithm, " SIMAX 2011
- D., I. Dumitriu, O. Holtz, R. Kleinberg, "Fast Matrix Multiplication is Stable," Num. Math., 2007
- D., I. Dumitriu, O. Holtz, "Fast Linear Algebra is Stable," Num. Math., 2007
- G. Ballard, D., I. Dumitriu, "Minimizing communication for eigenproblems and the SVD," arXiv:1011.3077, 2010

References (5/5)

- J. Banks, J. Garza-Vargas, A. Kulkarni, N. Srivastava, "Pseudospectral shattering, the sign function, and diagonalization in nearly matrix multiplication time," FOCM, 2022
- D., "On floating point errors in Cholesky," LAPACK Working Note #14, 1989
- E. Carson, N. Higham, S. Pranesh, "Three-precision GMRES-based Iterative Refinement for Least Squares," SISC 2020
- N. Higham, T. Mary, "Mixed precision algorithms in numerical linear algebra," Acta Numerica 2022
- R. Skeel, "Scaling for numerical stability in Gaussian Elimination," JACM, 1979
- D., I. Dumitriu, O. Holtz, P. Koev, "Accurate and Efficient Expression Evaluation and Linear Algebra,", Acta Numerica, 2008