Sparse Rank-Revealing LU Factorization

Householder Symposium XV on Numerical Linear Algebra

Peebles, Scotland, June 2002

Michael O'Sullivan and Michael Saunders

Dept of Engineering Science
University of Auckland
Auckland, New Zealand
michael.osullivan@auckland.ac.nz

Dept of Management Sci & Eng
Stanford University
Stanford, CA 94305-4026
saunders@stanford.edu

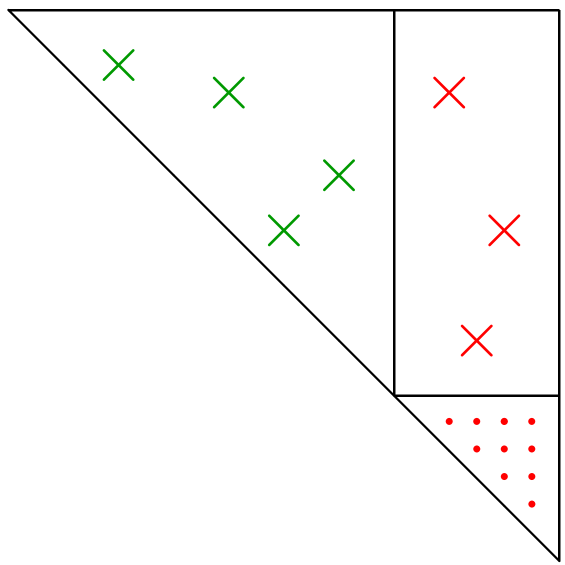
GOALS

Goals

Given a sparse matrix A ($m \times n$, usually $m \geq n$),

- Maintain sparsity
- Determine if A is ill-conditioned?
- Determine which columns to delete? (or replace)

$$P_1 A P_2 = LU, \qquad U =$$



Motivation

Dynamic Programming (Mike O'Sullivan's thesis)

- P substochastic
  $A = P - I$, at most one singularity

Optimization (MINOS and SNOPT)

- Basis Repair I
  $A = B$, square basis, perhaps ill-conditioned
- Basis Repair II
  $A = (B\ S)^T$, look for better B

LU with Threshold Pivoting

$$\texttt{Lmax} = \text{stability tolerance} = 10 \text{ or } 3.99 \text{ or } 1.99 \ldots$$
$$(\text{bound on } |L_{ij}|)$$

At each stage of Gaussian elimination:

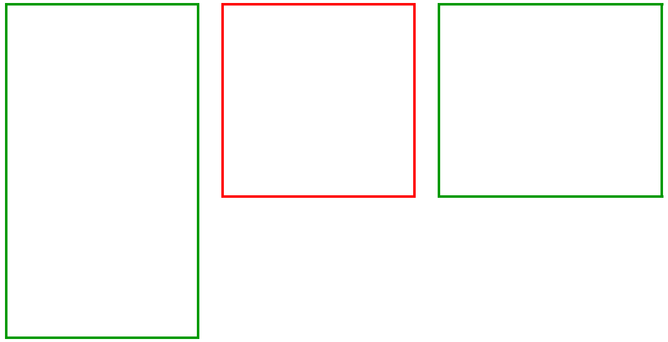
$$A \leftarrow A - lu^T$$

$$\alpha_j = \text{biggest element in col } j$$
$$\beta_i = \text{biggest element in row } i$$
$$\texttt{Amax} = \text{biggest element in } A \qquad (= \max\ \alpha_j = \max\ \beta_i)$$

RANK-REVEALING FACTORS

Rank-Revealing Factors

$$A = XDY^T =$$



Demmel et al. (1999)

X, Y full column rank, "well conditioned"

D diagonal

- SVD
- QR with column interchanges
- LU with Rook Pivoting
- LU with Complete Pivoting

Sparse Rank-Revealing Factors

- QR multifrontal — Pierce and Lewis (1997)

- TPP (Threshold Partial Pivoting) — Not RR

  $|a_{pq}| * \mathtt{Lmax} \geq \alpha_q$

- TRP (Threshold Rook Pivoting) — Gupta (2001) — WSMP, LUSOL

  $|a_{pq}| * \mathtt{Lmax} \geq \alpha_q$ and β_p

- TCP (Threshold Complete Pivoting) — LUSOL

  $|a_{pq}| * \mathtt{Lmax} \geq \mathtt{Amax}$

where Lmax (e.g. 10 or 3.99 or 1.99 . . .) bounds $|L_{ij}|$

LUSOL

LUSOL

$A = LU$ + updates, L well-conditioned

Gill, Murray, Saunders and Wright (1987)
Revised 1989–94, 2000–02

Markowitz strategy for sparse pivots
(cf. MA28, Y12M, LA05, MOPS, MA48)

TPP (Threshold Partial Pivoting)
Search only a few sparse cols and rows — Zlatev 1981
Store α_j at top of col j — Suhl & Suhl 1990

TRP (Threshold Rook Pivoting) — New
TCP (Threshold Complete Pivoting) — New

Stability Tolerance Lmax

$$P_1AP_2 = LDU, \text{ unit diags on } L, U$$

Threshold pivoting bounds the off-diags of L and perhaps U:

$$\left.\text{TPP}\right\} \quad |L_{ij}| \leq \texttt{Lmax} \approx 10.0$$

$$\left.\begin{array}{l}\text{TRP}\\ \text{TCP}\end{array}\right\} \quad |L_{ij}|, |U_{ij}| \leq \texttt{Lmax} \approx 5.0$$

TRP, TCP are more Rank-Revealing with low Lmax:

$$\text{cond}(L),\ \text{cond}(U) < (1 + \texttt{Lmax})^n$$

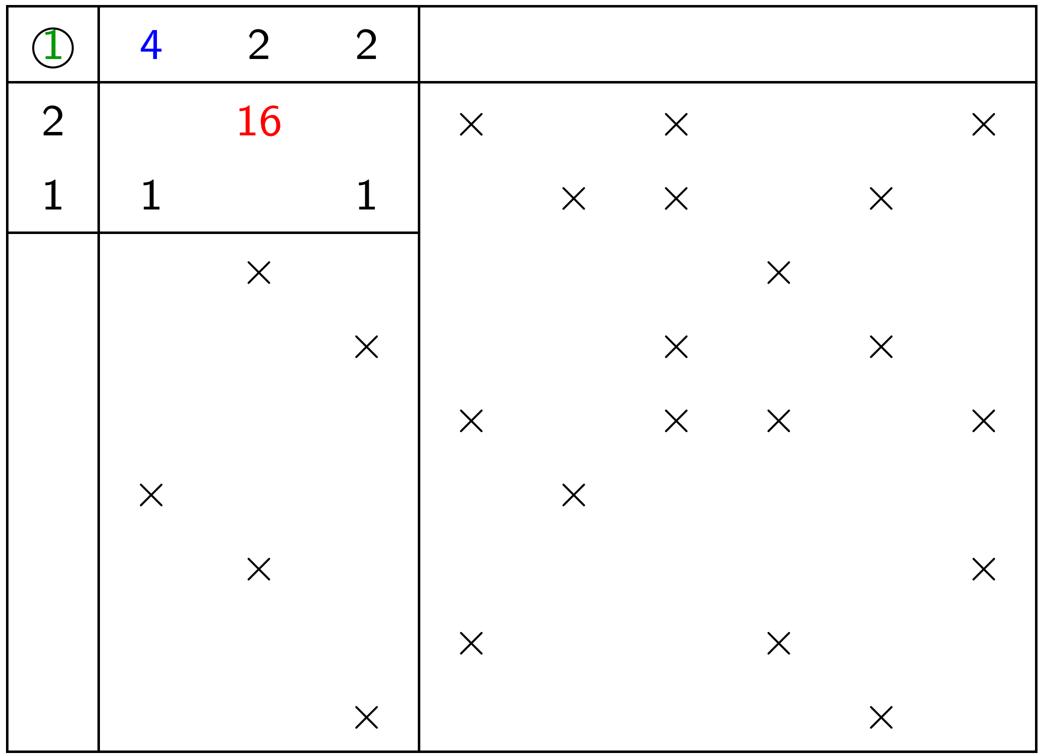
Elimination Step

Allowable pivots with Lmax $= 3.0$

TPP can pivot on ①

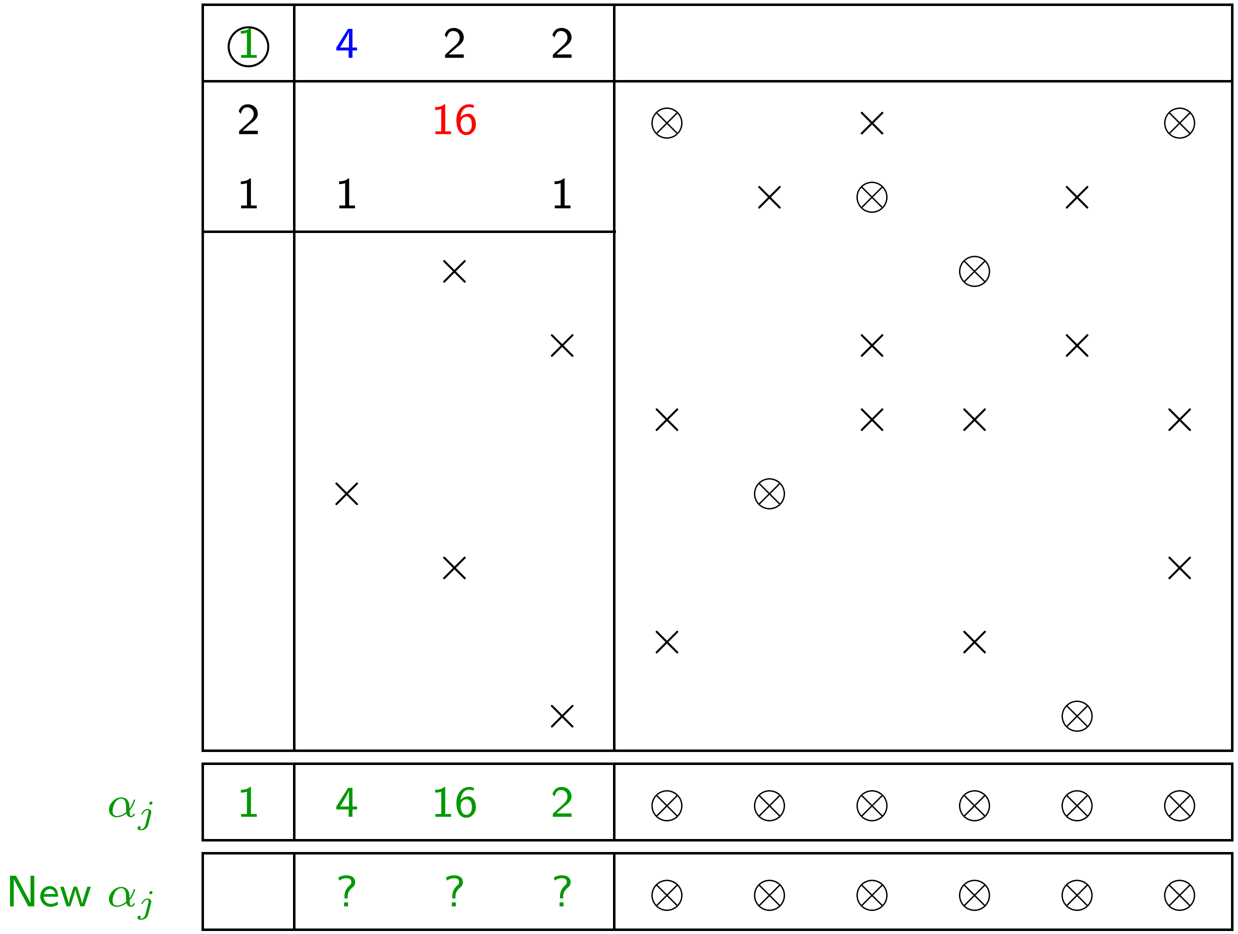
TRP can pivot on 4

TCP must pivot on 16



(Markowitz) Just a few columns and rows change

Maintaining α_j



A_{ij} stored column-wise $\Rightarrow$ easy to update modified α_j

Implementing TRP

Threshold Rook Pivoting

- Same Markowitz search strategy as TPP:
  cols of length 1, rows of length 1,
  cols of length 2, rows of length 2, . . .
  but have to search longer

- α_j are stored at top of each column

- β_i stored in separate array (new)

- A_{ij} stored column-wise $\Rightarrow$ β_i more expensive than α_j

- Could store A_{ij} row-wise $\Rightarrow$ faster but more storage

Implementing TCP

Partial Pivoting vs Complete Pivoting

Dense

	Computing L and U	$O(n^3)$	
PP	Finding α_1	$O(n)$	
CP	Finding `Amax`	$O(n^3)$	Not so bad!

Sparse

	Computing L and U	$O(\mathtt{nnz}(L+U))$	
TPP	Finding all α_j	$O(\mathtt{nnz}(L+U))$	
TCP	Finding `Amax`	$O(n^2)$?	Too much

Finding Amax from α_j

α_j = biggest element in column j

Amax = biggest element in A

jmax = column containing Amax

Naive Method

Find Amax by searching all α_j $O(n^2)$ Warning from Iain

Theorem

Need to search all α_j **only if Amax decreases**

Proof

If col `jmax` is not modified:

α_j	4	3	2	$\otimes$	$\otimes$	6	$\otimes$	$\otimes$	$\otimes$
`Amax` $\nearrow$	$+$	9	$+$	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$

α_j	4	3	2	$\otimes$	$\otimes$	6	$\otimes$	$\otimes$	$\otimes$
`Amax` $=$	$+$	$+$	$+$	$\otimes$	$\otimes$	6	$\otimes$	$\otimes$	$\otimes$

If col `jmax` is modified:

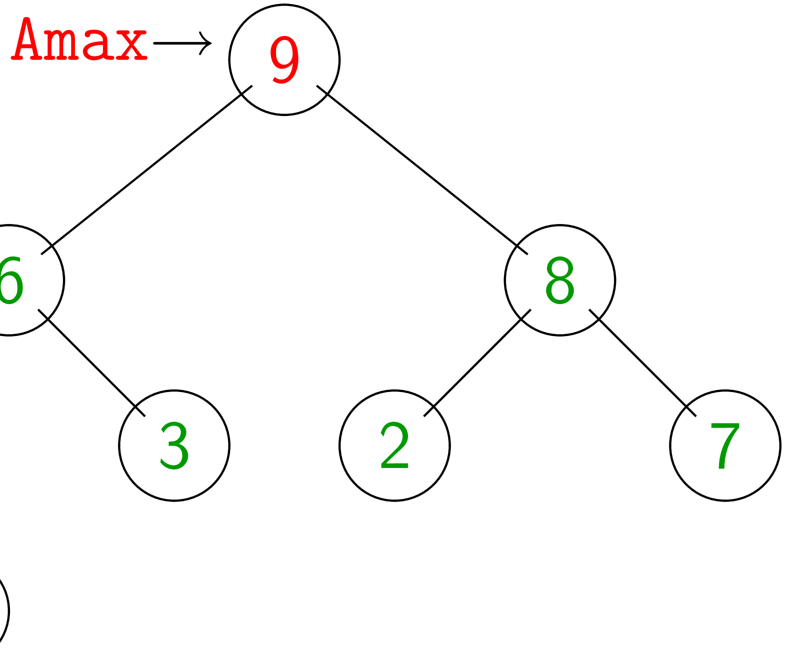
α_j	4	6	2	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$
`Amax` $\nearrow$	$+$	9	$+$	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$

α_j	4	6	2	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$	$\otimes$
`Amax` $\searrow$	$+$	5	$+$	$\otimes$	$\otimes$	?	$\otimes$	$\otimes$	$\otimes$

Must search for `Amax`

Better: Store α_j in a Heap

Thanks to John Gilbert



Ha(k)	9.0	6.0	8.0	...	α_j
Hj(k)	2	7	1	...	j
Hk(j)	3	1	6	...	location of j in heap

Calls to Heap Functions

`build` heap from all α_j

for $k = 1 : \min(m, n)$

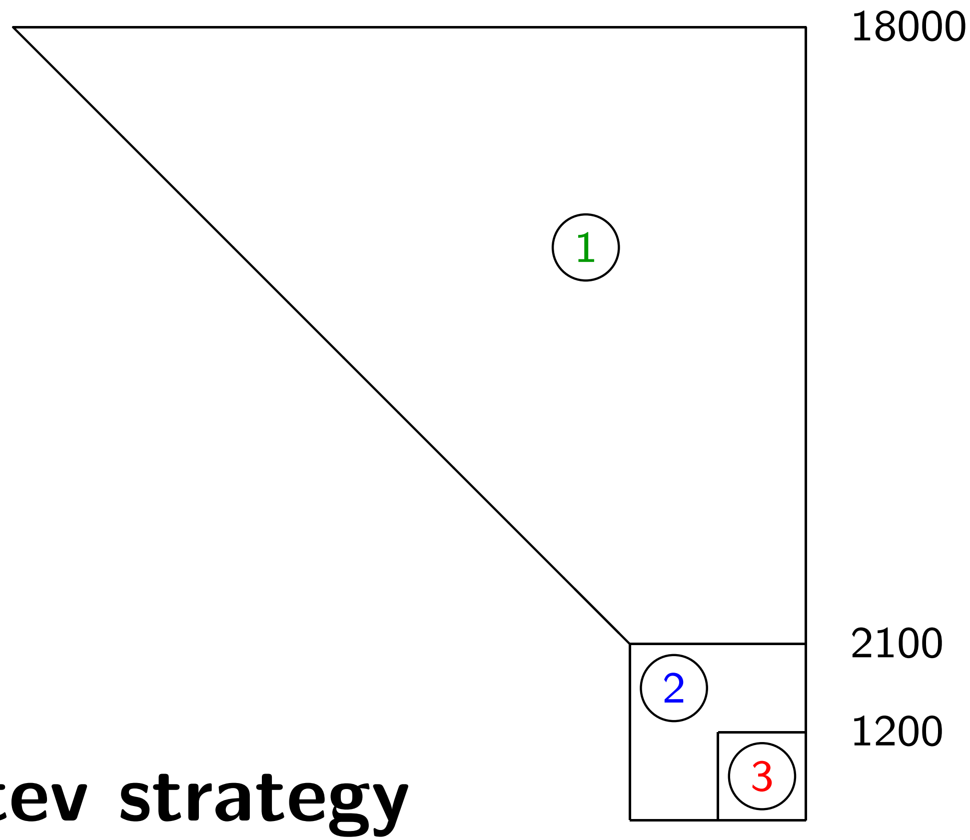
  Choose pivot, do elimination
  Find α_j for modified cols

  `delete` entry for pivot column
  for $l = 2$: lenpivrow
    `change` entry for each modified column
  end
end

Remarkably little work

NUMERICAL RESULTS

Markowitz, then Dense CP



Modified Zlatev strategy

1 Markowitz1 until 30% dense: Search at least 5 cols, 4 rows

2 Markowitz2 until 50% dense: Search at least 5 cols, 0 rows

3 Dense Complete Pivoting

Problem memplus from Harwell-Boeing collection

$A = 18000 \times 18000$, 126000 nonzeros, Scaled

		Lmax	nnz(L+U)	Time
TRP		100.0	142000	5
		10.0	141000	5
	RR	3.99	142000	5
	RR	2.50	146000	6
	RR	1.99	166000	7
	RR	1.58	172000	7
	RR	1.26	174000	8
TCP		100.0	140000	2
		10.0	579000	30
	RR	3.99	2460000	475
	RR	2.5	2890000	6610
	RR	1.5	7080000	27875
PP (SuperLU, colamd)		1.0	4470000	≈ 250

Problem BRATU2D from CUTE optimization collection

$A = 4900 \times 4900$, 24000 nonzeros

Permuted triangle

≈ 64 singularities

$$
P_1 A P_2 = \begin{pmatrix}
-1 & & -1 & & 4 & -1 & \ldots & & -1 & \ldots \\
& -1 & -1 & & & & \ldots & 4 & -1 & \\
& & 4 & & -1 & & \ldots & -1 & & \\
& & & -1 & -1 & & \ldots & -1 & 4 & \\
& & & & \ddots & & & & & \\
& & & & & \ddots & & & &
\end{pmatrix}
$$

Marching pattern from PDE

TRP and TCP must have `Lmax` < 4.0

TRP Profile

CUTE Problem BRATU2D

$A = 4900 \times 4900$, 24000 nonzeros

TRP, `Lmax`$= 1.26$, LU $=$ 206000 nonzeros

Update β_i	for modified rows	57.7%
Markowitz	Find stable pivot	31.4%
Elimination	The algebra	4.0%
Dense CP	228×228	2.0%
Update α_j	for modified cols	1.6%

TCP Profile

Harwell-Boeing Problem memplus

$A = 18000 \times 18000$, 126000 nonzeros

TCP, `Lmax` $=$ 10.0, LU $=$ 578000 nonzeros

Markowitz	Find stable pivot	65.0%
Dense CP	600×600	18.5%
Elimination	The algebra	7.4%
Update α_j	for modified cols	4.7%
`change`	Heap update	0.1% (!)

CONCLUSIONS

Conclusions

- TPP: Work-horse, usually reliable
- RRLUs can be rather dense. Scaling essential
- TRP: Markowitz search costs more
- TCP: Markowitz search costs MUCH more
  but Heap allows `Amax` to be maintained cheaply
- TRP, TCP are usually Rank-Revealing with `Lmax` ≤ 4.0
  (but sometimes needs 2.5)
- SNOPT optimization code on CUTE test problems:
  TRP reveals same rank as TCP almost always
  (much more cheaply)

HEAPS of THANKS

Iain Duff	John Gilbert
R. Sedgewick	Cormen, Leiserson and Rivest
Les Foster	
Philip Gill	The CUTE and Harwell-Boeing Teams
Michael Friedlander	