Sparse Rank-Revealing LU Factorization

SIAM Conference on Optimization
Toronto, May 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

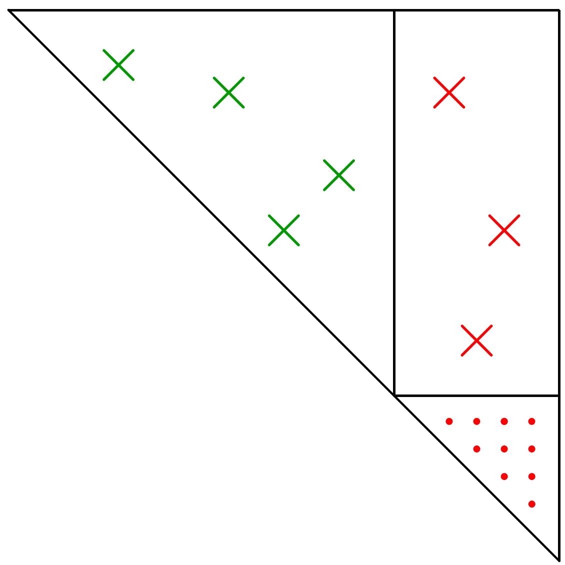
GOALS

Goals

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

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

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



Motivation

Dynamic Programming (Mike O'Sullivan's thesis)

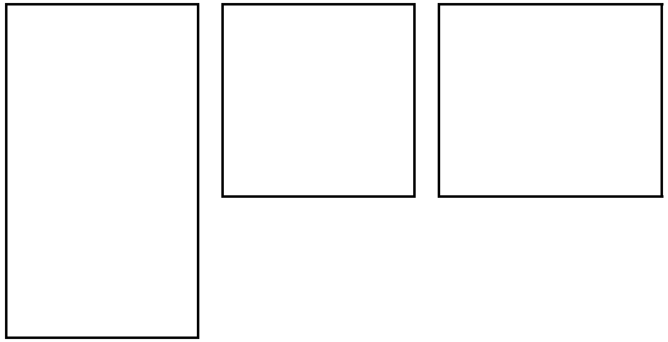
- P substochastic
  $A = P - I$, at most one singularity

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

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 (Foster 1997)
- LU with Complete Pivoting

Sparse Rank-Revealing Factors

- QR multifrontal — Pierce and Lewis (1997)
- TPP (Threshold Partial Pivoting) — Not RR
- TRP (Threshold Rook Pivoting) — Perhaps effective?
- TCP (Threshold Complete Pivoting) — This talk

PIVOTING NEEDS

$\texttt{Amax}_j$ = **biggest element in col** j

$\texttt{Amax}$ = **biggest element in** A

Partial Pivoting vs. Complete Pivoting

Dense

	Computing L and U	$O(n^3)$	
PP	Finding $\texttt{Amax}_1$	$O(n)$	
CP	Finding $\texttt{Amax}$	$O(n^3)$	Not so bad!

Sparse

	Computing L and U	$O(\texttt{nnz}(L+U))$	
TPP	Finding $\texttt{Amax}_j$	$O(\texttt{nnz}(L+U))$	
TCP	Finding $\texttt{Amax}$	$O(n^2)$?	Too much

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 $\texttt{Amax}_j$ at top of col j — Suhl & Suhl 1990

TCP (Threshold Complete Pivoting) — New

Stability Tolerance Lmax

$$P_1 A P_2 = LDU, \text{ unit diags on } L, U$$

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 $\texttt{Lmax} = 3$

TPP	①
TRP	4
TCP	16

①	4	2	2						
2		16		×		×			×
1	1		1		×	×		×	
		×					×		
			×			×		×	
				×		×	×		×
	×				×				
		×							×
				×			×		
			×					×	

(Markowitz) Just a few columns and rows change

Finding Amax from $\texttt{Amax}_j$

$\texttt{Amax}_j$	=	biggest element in column j
Amax	=	biggest element in A
jmax	=	column containing Amax

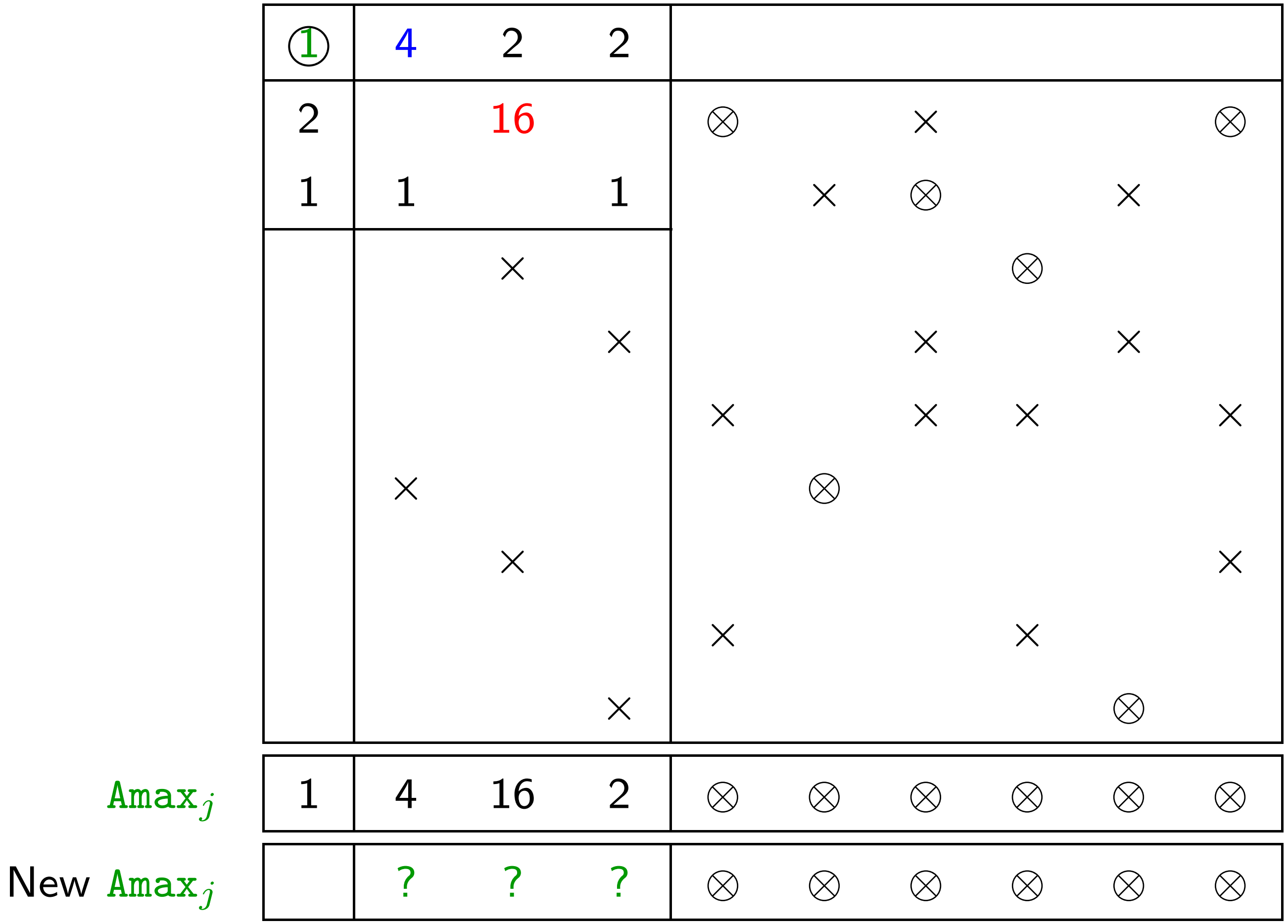
Naive Method

Find Amax by searching all $\texttt{Amax}_j$ $O(n^2)$

Theorem

Need to search all $\texttt{Amax}_j$ **only if** Amax **decreases**

Maintaining $\texttt{Amax}_j$



Easy to update modified $\texttt{Amax}_j$

Maintaining Amax

Amax is in column jmax, which may be modified:

$\mathtt{Amax}_j$	4	6	2	⊗	⊗	⊗	⊗	⊗	⊗
Amax ↗	+	9	+	⊗	⊗	⊗	⊗	⊗	⊗

$\mathtt{Amax}_j$	4	6	2	⊗	⊗	⊗	⊗	⊗	⊗
Amax ↘	+	5	+	⊗	⊗	?	⊗	⊗	⊗

Must search for Amax

If col jmax is not modified:

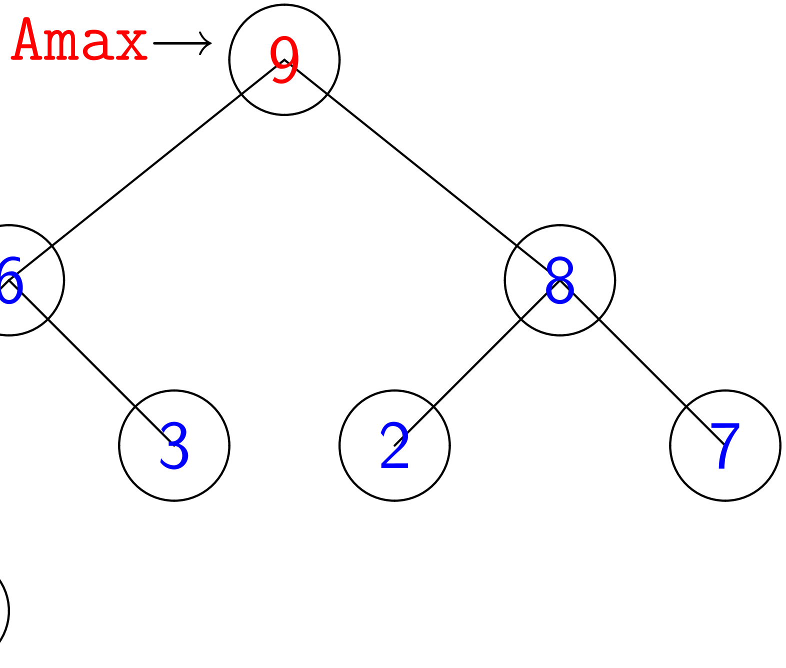
$\mathtt{Amax}_j$	4	3	2	⊗	⊗	6	⊗	⊗	⊗
Amax ↗	+	9	+	⊗	⊗	⊗	⊗	⊗	⊗

$\mathtt{Amax}_j$	4	3	2	⊗	⊗	6	⊗	⊗	⊗
Amax =	+	+	+	⊗	⊗	6	⊗	⊗	⊗

HEAPS

Storing $\texttt{Amax}_j$ in a Heap

Thanks to John Gilbert



$\texttt{Ha}(k)$	9.0	6.0	8.0	...	$\texttt{Amax}_j$
$\texttt{Hj}(k)$	2	7	1	...	j
$\texttt{Hk}(j)$	3	1	6	...	location of j in heap

Calls to Heap Functions

```
build heap from all Amax_j

for k = 1 : min(m, n)

   Choose pivot, do elimination
   Find Amax_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

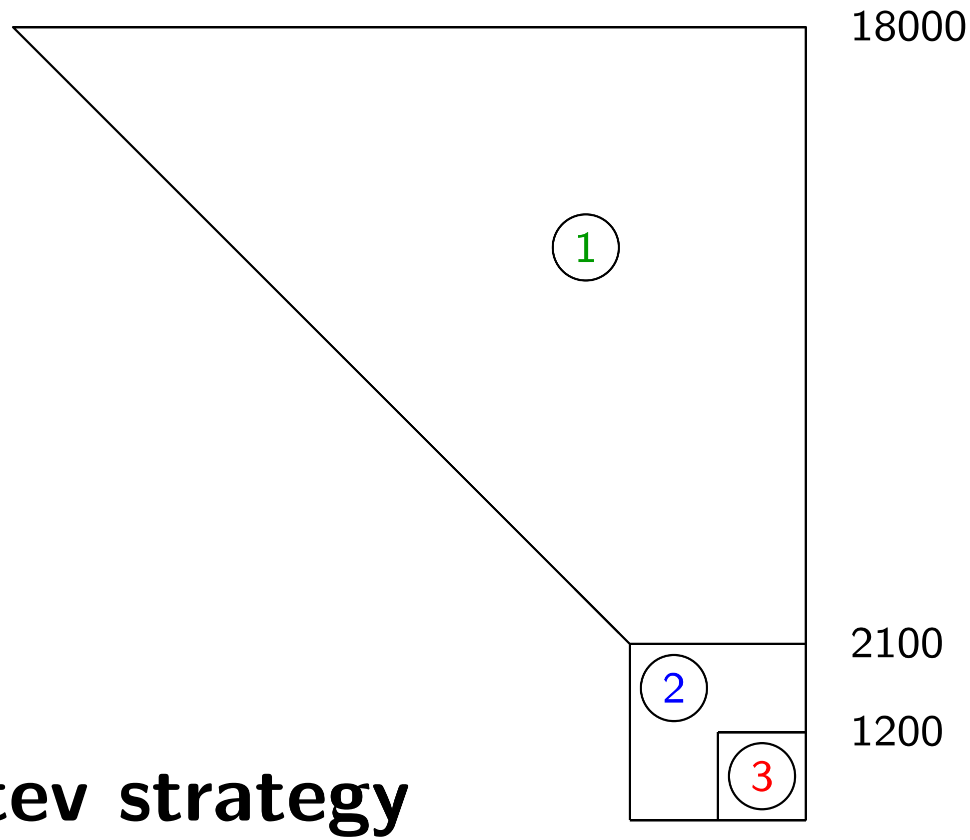
Cost of RR Feature

Problem memplus from Harwell-Boeing collection

$A = 18000 \times 18000$, 126000 nonzeros

	Lmax	nnz(L+U)	Time	(secs, 350MHz P3)
LUSOL (TCP)	100.0	128716	2	
	50.0	158260	4	
	25.0	247068	18	
	12.5	557170	76	
	10.0	485383	57	
	8.0	1182468	262	
RR	5.0	1846111	536	
RR	3.99	1835232	521	
RR	2.24	2373417	4756	
RR	1.5	7080769	27875	
SuperLU (colamd PP)	1.0	4470809	≈ 250	

Markowitz CP, 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

Profile

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%
$\texttt{Amax}_j$	For modified cols	4.7%
`change`	Heap update	0.1% (!)

CONCLUSIONS

Conclusions

- Threshold Complete Pivoting is usually Rank-Revealing with `Lmax` ≤ 4.0 (but sometimes needs 2.5)
- Heap structure allows `Amax` to be maintained cheaply
- RRLUs can be rather dense
- TPP: Work-horse, usually reliable
- TCP: Markowitz search dominates cost
- TRP: Worth investigating

HEAPS of THANKS

- John Gilbert
- Philip Gill
- The CUTE and Harwell-Boeing Teams
- Maureen Doyle
  R. Sedgewick
  Cormen, Leiserson and Rivest
- Michael Friedlander