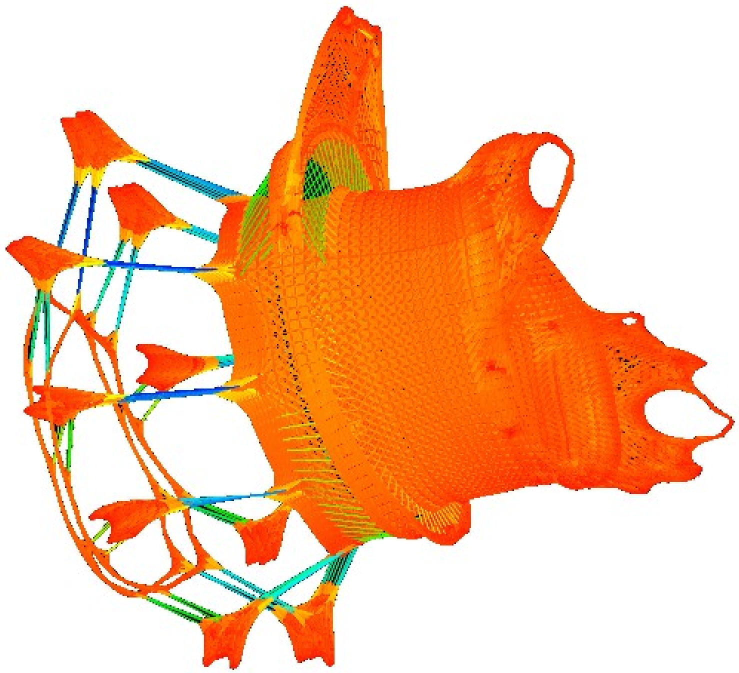
SYSC5906 - Directed Studies
(Distributed Sparse Matrices)

A Report

Alistair Boyle 2010

Overview



Storage

Solution

Ordering

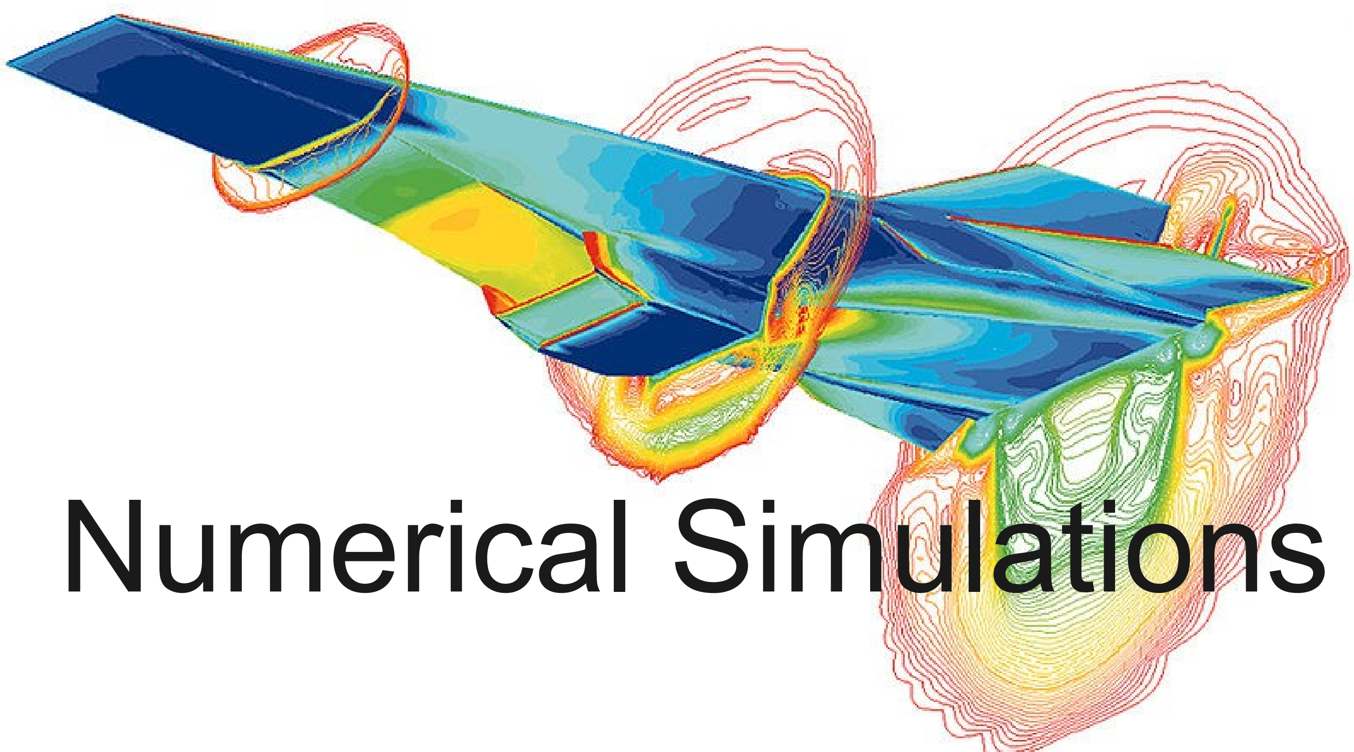
Distribution

[http://www.cise.ufl.edu/research/sparse/matrices/Rothberg/gearbox.html, 107624 nodes, 3250488 edges, UF Sparse Matrix Collection]

Title image: http://www.flickr.com/photos/8702301@N06/5006243147/

Motivation

- Linear Equations
  - Exact solutions
  - Approximations
- Eigenvalues/vectors
  - Vibration
  - Harmonics



Numerical Simulations

[http://www.dfrc.nasa.gov/Gallery/Photo/X-43A/HTML/ED97-43968-1.html]

Sparse Storage

Triplets

(1,1) = 10

(10,100) = 2.3

(7,5103) = 6.44

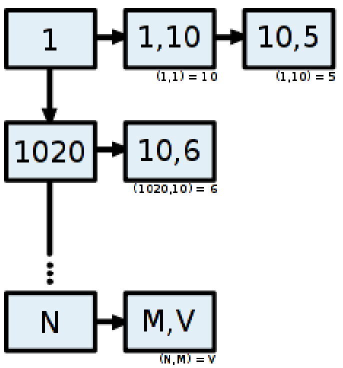
(1020,10) = 6

⋮

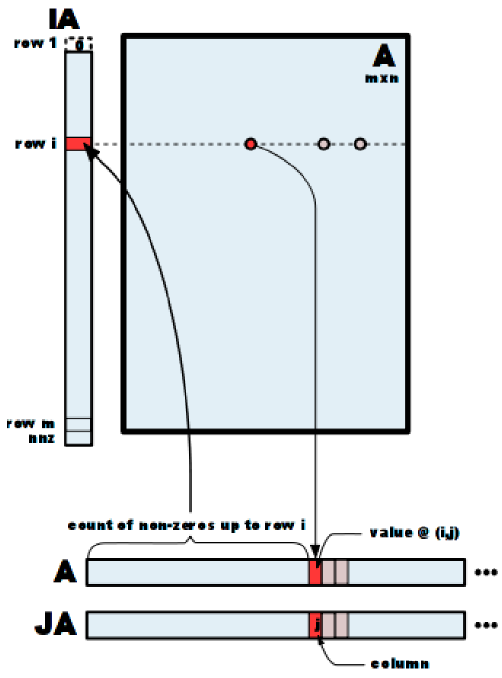
(1,10) = 5

(N,M) = V

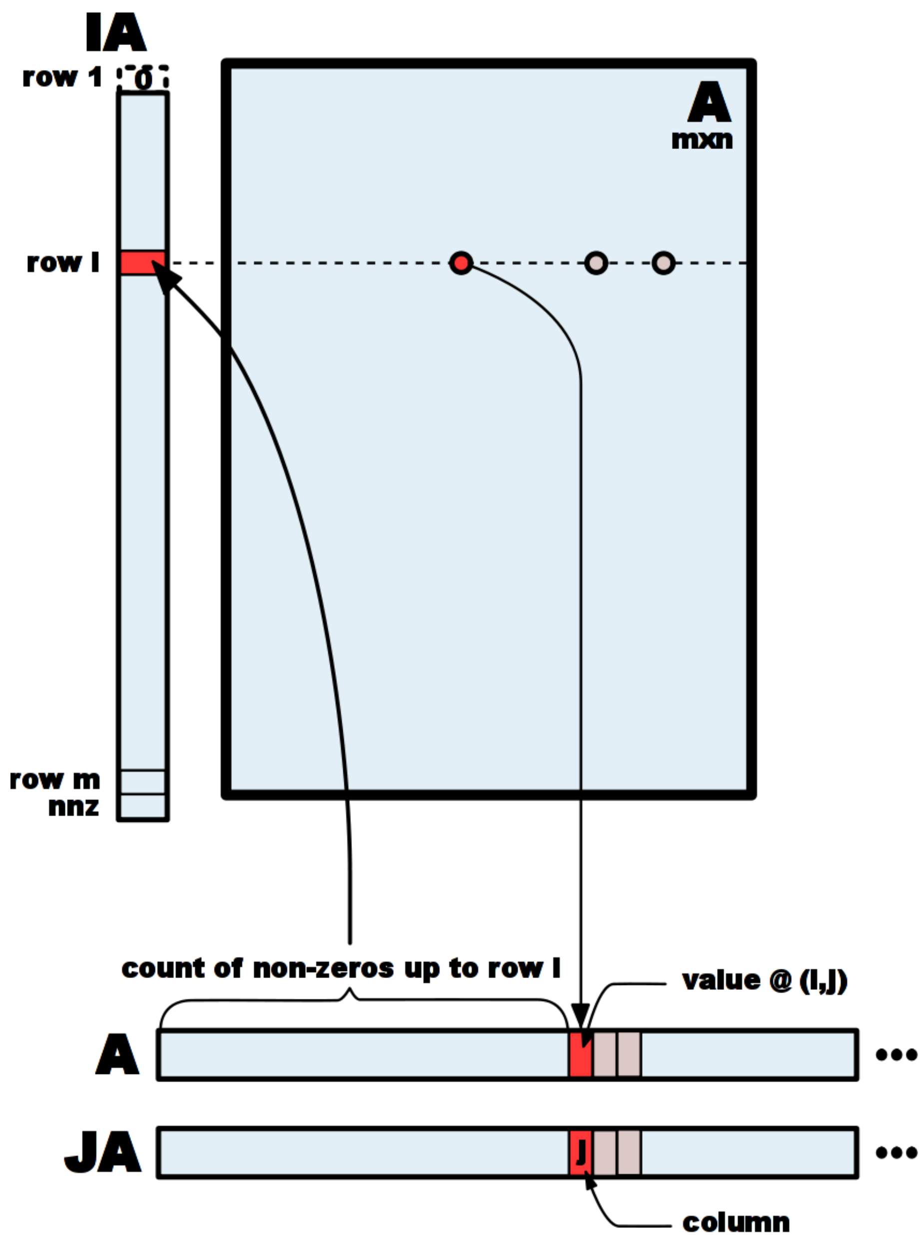
List-of-Lists



CSR



Sparse Storage
Compressed Sparse Row format



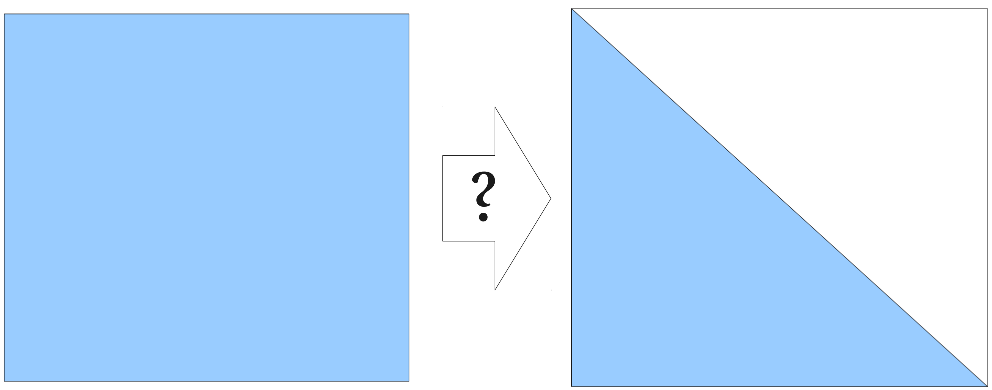
Sparse Storage File Formats

- Matrix Market
  - Triplets format, ASCII
- Harwell-Boeing
  - Compressed Sparse Column (CSC) format storage
  - Assembled, elemental, real, complex, pattern matrices
  - Support for multiple right-hand sides, guesses, solutions
- Rutherford-Boeing
  - An updated version of the Harwell-Boeing format
  - Supplementary matrix information
    - Orderings, estimates, partitions, Laplacian values, geometry, etc.

[The Rutherford-Boeing Sparse Matrix Collection, Duff, Grimes, Lewis, Rutherford Appleton Laboratory, 1997]

Solution Decompositions

Involves converting matrices to triangular or diagonal form through matrix transformations



Solution
LU Decomposition

- If A is square:

$$A x = b \;\Rightarrow\; x\,?$$

$$\boxed{A = L U}$$

1. decomposition
(Crout or Doolittle algorithms)

$$L y = b \;\Rightarrow\; y$$

2. forward substitution

$$U x = y \;\Rightarrow\; x$$

3. backward substitution

- L is unit lower triangular, U is upper triangular

[Gene H. Golub and Charles F. van Loan: Matrix Computations, 2nd ed., The John Hopkins University Press, 1989]

Solution
Cholesky Decomposition

- If A is square, hermitian, positive definite:

$$A\,x = b \;\Rightarrow\; x\,?$$

$$\boxed{A = L\,L^H}$$

1. decomposition
(Choleksy-Crout/Banachiewicz algorithms)

$$L\,y = b \;\Rightarrow\; y$$

2. forward substitution

$$L^H x = y \;\Rightarrow\; x$$

3. backward substitution

- Special case of LU decomposition where $U = L^H$

[Gene H. Golub and Charles F. van Loan: Matrix Computations, 2nd ed., The John Hopkins University Press, 1989]

Solution
QR Decomposition

- If A is square, nonsingular:

$$argmin \|Ax - b\|_2^2 \Rightarrow x?$$

(Least squares solution)

1. decomposition (Gram-Schmidt, Givens, Householder algorithms)

$$\boxed{A = QR} \Rightarrow Q^H A = Q^H Q R = R$$

$$\|Q^H A x - Q^H b\|_2^2 = \left\| \begin{matrix} Rx - c_1 \\ -c_2 \end{matrix} \right\|_2^2 \quad \ldots \quad c = Q^H b, Rx = c_1 \Rightarrow x$$

- Q is orthogonal/unitary, R is upper triangular

$$Q^H Q = I, Q^{-1} = Q^H$$

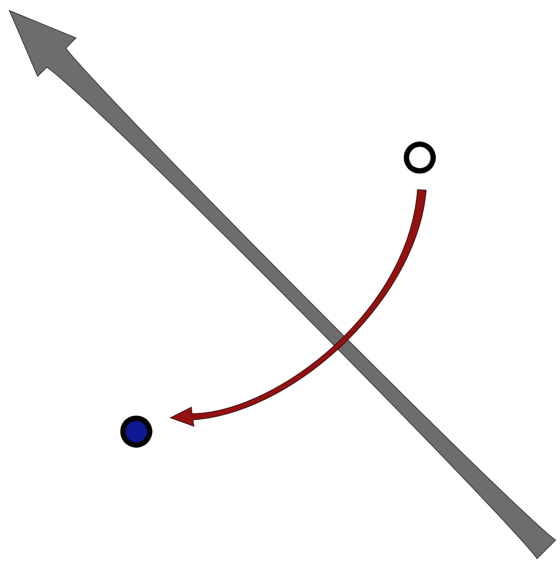
[Gene H. Golub and Charles F. van Loan: Matrix Computations, 2nd ed., The John Hopkins University Press, 1989]

Solution
Other Decompositions

Eigenproblems: Schur factorization

Singular Value Decompositions

Solution: QR factorization
Givens Rotations



- Rotation of a point about a line

Solution: QR factorization Givens Rotations

$$\begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} m \\ 0 \end{bmatrix}$$

$$m = \sqrt{x^2 + y^2}$$
$$a = x / r$$
$$b = y / r$$

Example

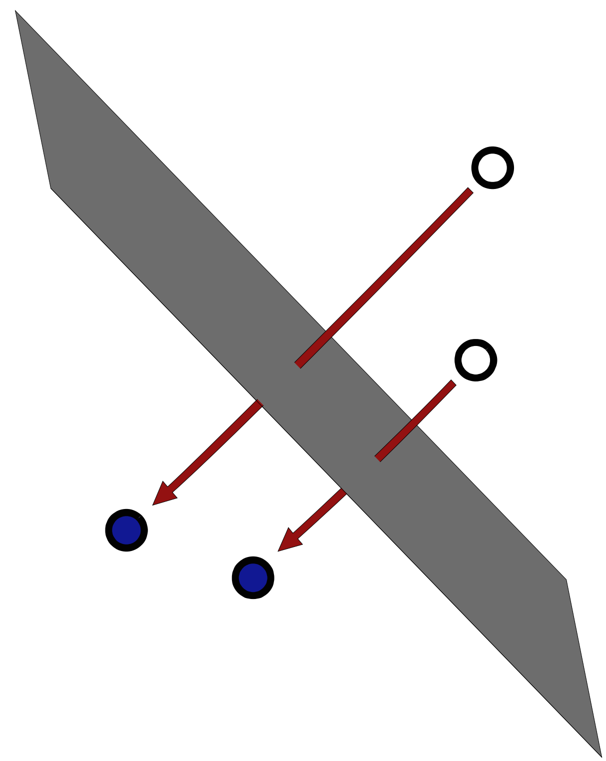
$$G_n = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & a & 0 & b & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & -b & 0 & a & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Only four entries in the transformed matrix are modified for a single Givens rotation. Result is one entry in the transformed matrix becoming zero.

[Gene H. Golub and Charles F. van Loan: Matrix Computations, 2nd ed., The John Hopkins University Press, 1989]

[Edward Anderson, Discontinuous Plane Rotations and the Symmetric Eigenvalue Problem, 2000, LAPACK Working Note 150,http://www.netlib.org/lapack/lawnspdf/lawn150.pdf]

Solution: QR factorization
Householder Reflections



- Reflection of points across a hyper-plane

Solution: QR factorization Householder Reflections

$$A=\begin{bmatrix} a_{11} & a_{12} & \dots \\ a_{21} & a_{22} & \dots \\ a_{31} & a_{32} & \dots \end{bmatrix} \Rightarrow A'=Q\,A=\begin{bmatrix} \alpha & a_{12}' & \dots \\ 0 & a_{22} & \dots \\ 0 & a_{32} & \dots \end{bmatrix}$$

$$\alpha=\|x\|, \quad x=[a_{11}\, a_{21}\, a_{31}]^T$$

$$e_1=[1\,0\,0\dots]^T$$

$$u=x-\alpha e_1$$

$$v=u/\|u\|$$

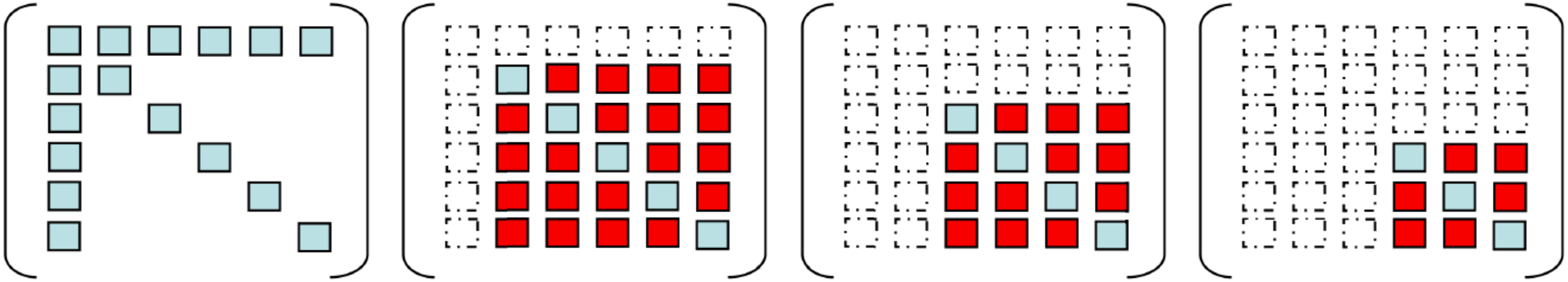
$$Q=I-2\,v\,v^T$$

Removes column at-a-time. Operates on submatrices as upper triangular matrix is produced.

[Gene H. Golub and Charles F. van Loan: Matrix Computations, 2nd ed., The John Hopkins University Press, 1989]

Fill-in

- As a matrix is decomposed, it may fill-in: previously zero entries become non-zero, making more work for the later stages.

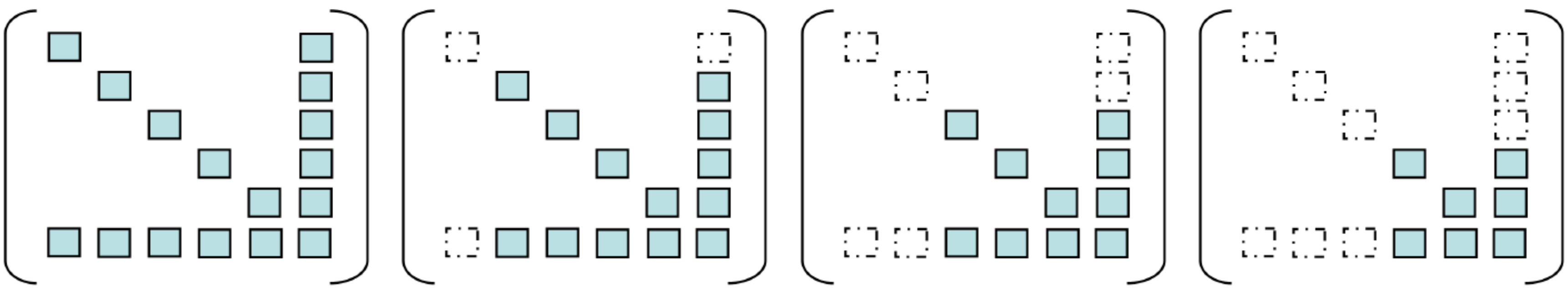


First 4 steps in Cholesky algorithm
Blue: initially non-zero,
Red: fill-in

[figure from: Minimum Degree Reordering Algorithms: A Tutorial, Stephen Ingram, sfingram@cs.ubc.ca]

Fill-in

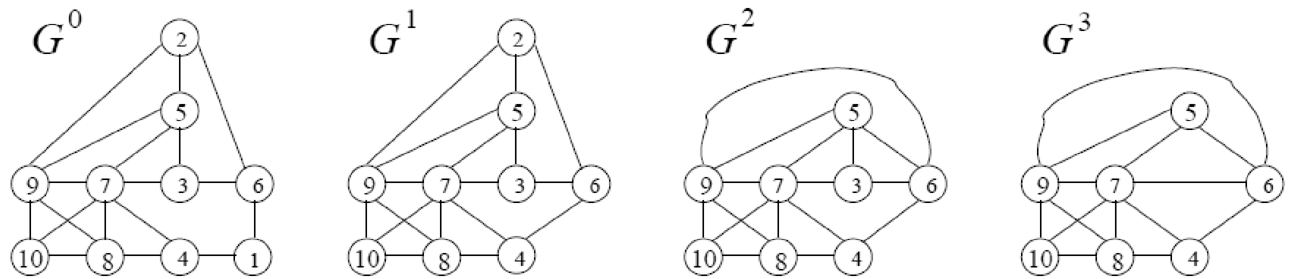
- Reordering the first row & column: massive improvement in required operations
- But... finding a “good” ordering is NP-hard



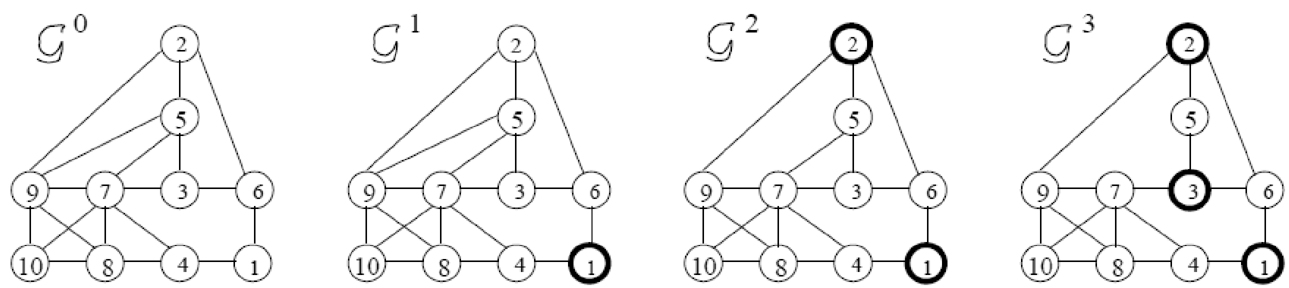
Reordered, First 4 steps in Cholesky algorithm
Blue: initially non-zero,
Red: fill-in (none)

[figure from: Minimum Degree Reordering Algorithms: A Tutorial, Stephen Ingram, sfingram@cs.ubc.ca]

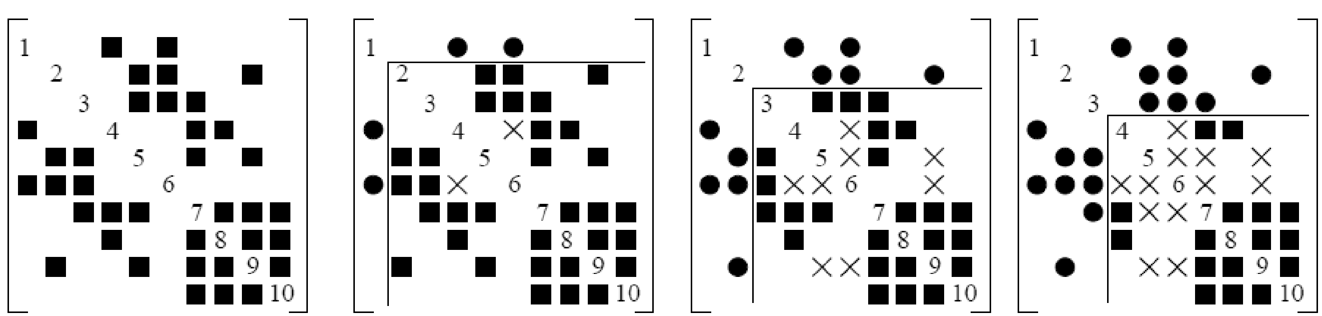
Ordering
Minimum Degree



(a) Elimination graph



(b) Quotient graph



(c) Factors and active submatrix

[An Approximate Minimum Degree Ordering Algorithm,Patrick R. Amestoy, Timothy A. Davisy, Iain S. Duz, SIAM J. Matrix Analysis & Applic., Vol 17, no 4, pp. 886-905, Dec. 1996]

Orderings
Nested Dissection

- Divide and Conquer
  - Recursively split based on mutual independence

$$A = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & a_{23} \\ 0 & a_{32} & a_{33} \end{bmatrix} \Rightarrow A' = \begin{bmatrix} a_{11} & 0 & a_{13}' \\ 0 & a_{33} & a_{23}' \\ a_{31}' & a_{32}' & a_{33}' \end{bmatrix}$$

[Direct methods for sparse linear systems, Davis, 2006, SIAM]

Ordering

- MD, MMD, AMD
- PORD

- UMFPACK

Hybrid

- METIS
- SCOTCH

Ordering
parallel ordering

ParMETIS

pt-SCOTCH

Hybrid

minimum degree

nested-dissection

[ParMETIS - http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview]

[ptSCOTCH - http://www.labri.fr/perso/pelegrin/scotch/]

Distribution
Left-Looking

- Two operations
  - Divide column by sqrt of its diagonal
  - Add a multiple of one column to another
- Column-based
  - iterates on columns to the left of the current column
- Save updates until a column is completed

[An evaluation of Left-Looking, Right-Looking and Multifrontal Approaches to Sparse Cholesky Facotrizations on Heirarchial-Memory Machines, Rothberg, Gupta, Technical Report, 1991]

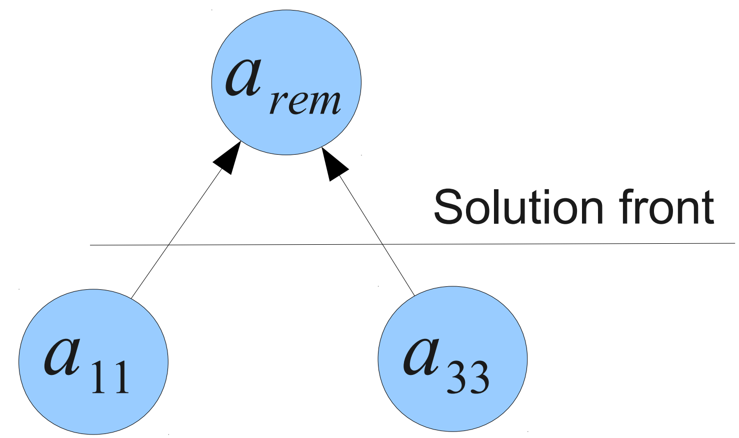
Distribution
Right-Looking

- Two operations
  - Divide column by sqrt of its diagonal
  - Add a multiple of one column to another
- “Submatrix”-based
  - Iterates on columns to the right of the current column
- Requires (inexpensive) search on destination's storage to find new non-zero locations to insert

[An evaluation of Left-Looking, Right-Looking and Multifrontal Approaches to Sparse Cholesky Facotrizations on Heirarchial-Memory Machines, Rothberg, Gupta, Technical Report, 1991]

Distribution
Multifrontal (Right-Looking)

$$A' = \begin{bmatrix} a_{11} & 0 & a_{13}' \\ 0 & a_{33} & a_{23}' \\ a_{31}' & a_{32}' & a_{33}' \end{bmatrix}$$



Form a tree and solve independent portions.
Updates to the matrix occur at the "front".
Updates are kept on a stack – typically +25% space.
Can have multiple independent fronts in parallel.
Can take advantage of "super-nodes"

[Direct methods for sparse linear systems, Davis, 2006, SIAM]

Solving in Serial

- CHOLMOD
- MA57
- MA41
- MA42
- MA67
- MA48
- Oblio

- SPARSE
- SPARSPAK
- SPOOLES
- SuperLLT
- SuperLU
- UMFPACK

Legend

- symmetric positive definite
- symmetric
- non-symmetric

[Direct Solvers for Sparse Matrices, X.Li March 2010]

Solving in Parallel shared memory

- BCSLIB-EXT
- Cholesky
- DMF
- MA41
- MA49
- PanelLLT
- PARASPAR
- PARADISO
- SPOOLES
- SuiteSparseQR
- SuperLU_MT
- TAUCS
- WSMP

Legend

- symmetric positive definite
- symmetric
- non-symmetric

[Direct Solvers for Sparse Matrices, X.Li March 2010]

Solving in Parallel distributed

- DMF
- DSCPACK
- MUMPS
- PaStiX
- PSPASES
- SPOOLES
- SuperLU_DIST
- S+
- WSMP

Legend

symmetric positive definite

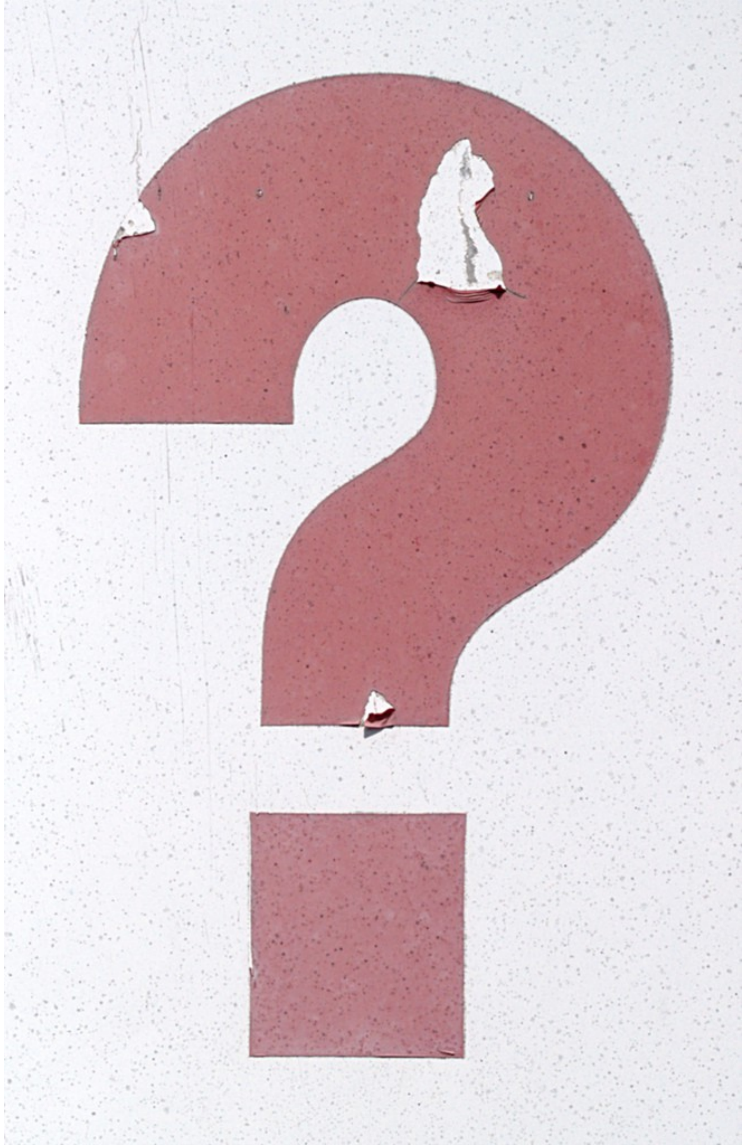
symmetric

non-symmetric

[Direct Solvers for Sparse Matrices, X.Li March 2010]

[A fully asynchronous multifrontal solver using distributed dynamic scheduling, PR Amestoy, IS Duff, JY L'Excellent, J, SIAM Journal on Matrix, 2002]

Questions?



[http://www.flickr.com/photos/takomabibelot/4164289232/]