SAND2011-1447P

Efficient Preconditioners for Large-Scale Parallel Circuit Simulation

SIAM Computational Science & Engineering 2011

February 28th, 2011

Heidi K. Thornquist

Sandia National Laboratories

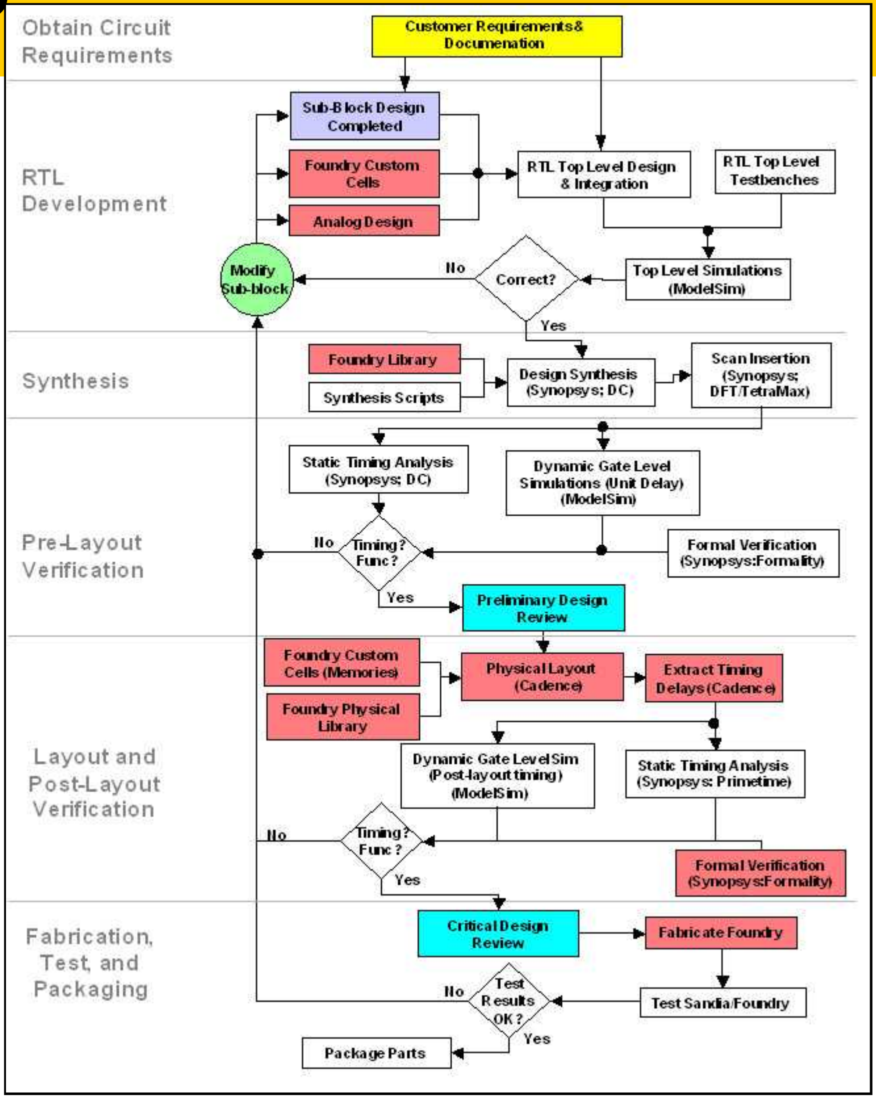
Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Outline

- Background / Motivation
- Simulation Challenges
- Efficient Preconditioning Strategies
  - Singleton Filtering
  - Load Balancing / Partitioning
  - Global Reordering
    - Block Triangular Form Structure
    - Doubly Bordered Block Diagonal Form Structure
  - Results
- Conclusions

Circuit Design Process

- Highly complex
  - Requires different tools for verifying different aspects of the circuit
- Cannot afford many circuit re-spins
  - Expense of redesign
  - Time to market
- Accurate / efficient / robust tools
  - Challenging for 45nm technology



Sandia National Laboratories

– Analog circuit simulator (SPICE compatible)

– Large scale (N>1e7) “flat” circuit simulation

- solves set of coupled DAEs simultaneously

– Distributed memory parallel

– Advanced solution techniques

- Homotopy
- Multi-level Formulation
- Multi-time Partial Differential Equation (MPDE)
- Parallel Iterative Matrix Solvers / Preconditioners

– 2008 R&D100 Award

Parallel Circuit Simulation Challenges

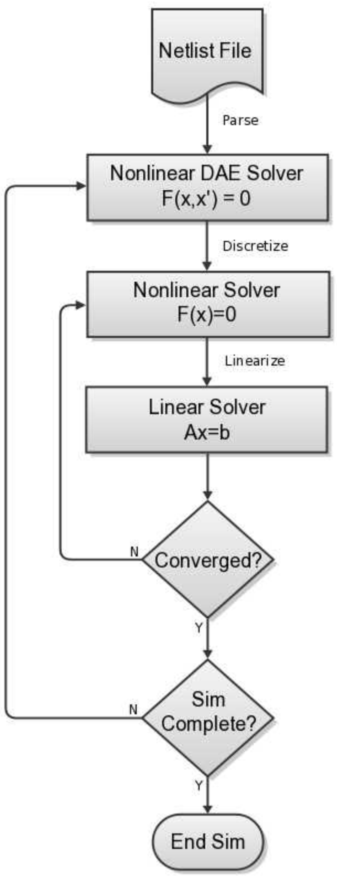
Analog simulation models network(s) of devices coupled via Kirchoff's current and voltage laws

$$f(x(t)) + \frac{dq(x(t))}{dt} = b(t)$$

- Network Connectivity
  - Hierarchical structure rather than spatial topology
  - Densely connected nodes: O(n)
- Badly Scaled DAEs
  - Compact models designed by engineers, not numerical analysts!
  - DCOP matrices are often ill-conditioned
- Non-Symmetric
  - Not elliptic and/or globally SPD
- Load Balancing / Partitioning
  - Balancing cost of loading Jacobian values unrelated to matrix partitioning for solves

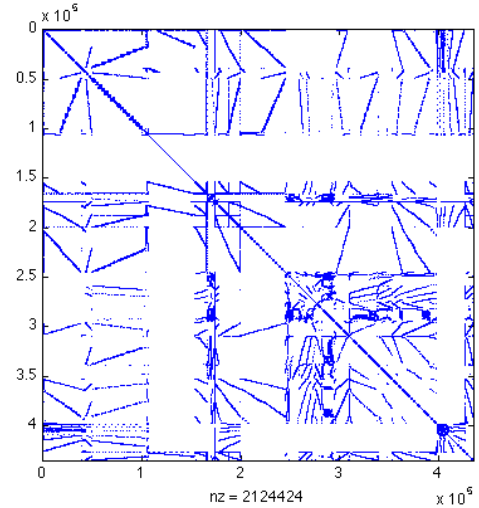
Parallel Circuit Simulation Structure
(Transient Simulation)

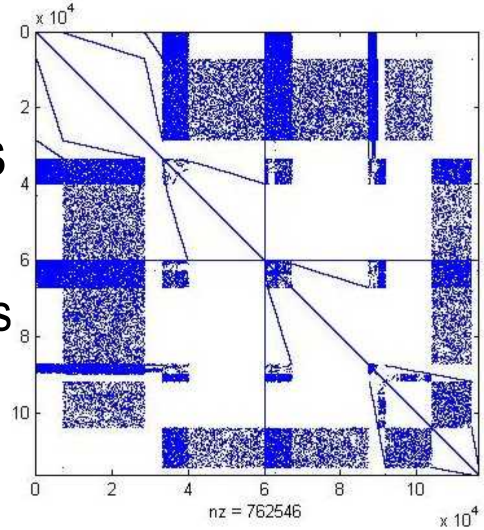
- Simulation challenges create problems for linear solver
  - Direct solvers more robust
  - Iterative solvers have potential for better scalability
- Iterative solvers have previously been declared unusable for circuit simulation
  - Black box methods **do not** work!
  - Need to address these challenges in creation of preconditioner
- Efficient large-scale simulation can leverage parallelism at many levels
  - Coarse-scale (multi-processor)
  - Fine-scale (multi-threaded)

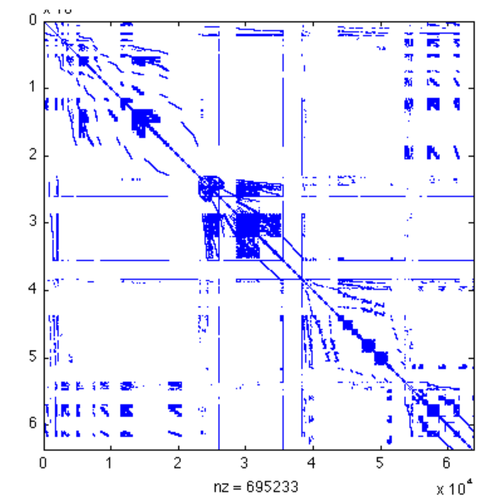


Circuit Matrix Structure

- Heterogeneous matrix structure
  - Nearly symmetric
  - Highly sparse
- Static graph
  - Enables use of expensive or serial methods
  - Reuse graph analysis
- Efficient preconditioners
  - Global reordering
  - Exact subdomain solves
  - Hybrid direct / iterative

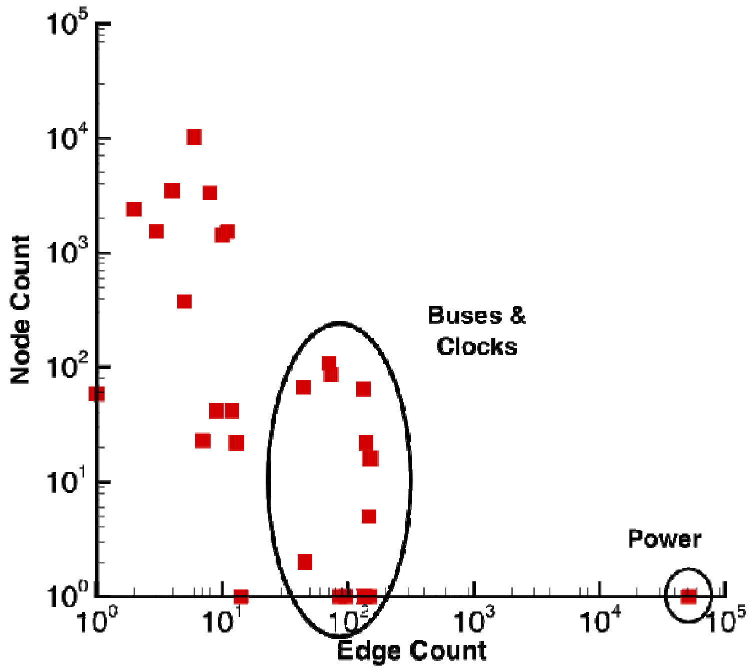






Network Connectivity

(Singleton Removal)



- Connectivity:
  - Most nodes very low connectivity -> sparse matrix
  - Power node generates very dense row (~0.9*N)
  - Bus lines and clock paths generate order of magnitude increases in bandwidth
  - Basermann et al. [2001, 2005].

Row Singleton: pre-process

$$\begin{bmatrix} & & & a_{1j} & & & \\ & & & a_{2j} & & & \\ & & & \vdots & & & \\ & & & \vdots & & & \\ 0 & \cdots & 0 & a_{ij} & 0 & \cdots & 0 \\ & & & \vdots & & & \\ & & & a_{nj} & & & \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ \vdots \\ x_j \\ \vdots \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ \vdots \\ \vdots \\ b_i \\ \vdots \\ b_n \end{bmatrix}$$

$$\Rightarrow x_j = b_i / a_{ij}$$

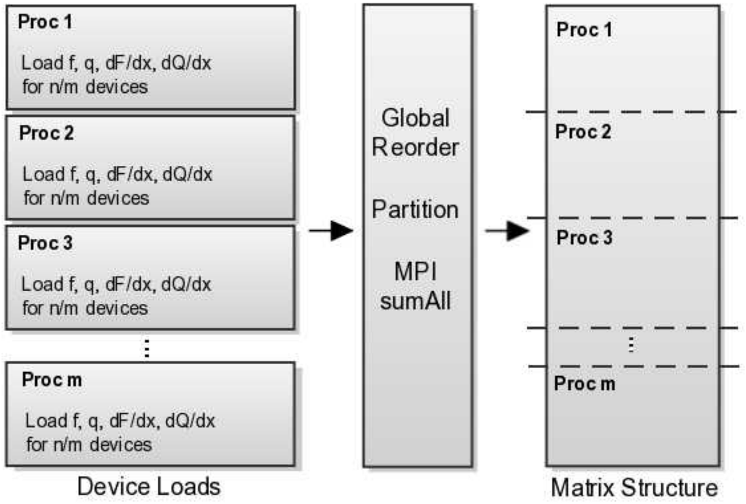
Column Singleton: post-process

$$\begin{bmatrix} & & & 0 & & & \\ & & & 0 & & & \\ & & & \vdots & & & \\ & & & \vdots & & & \\ a_{i1} & \cdots & \cdots & a_{ij} & \cdots & \cdots & a_{in} \\ & & & \vdots & & & \\ & & & 0 & & & \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ \vdots \\ x_j \\ \vdots \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ \vdots \\ \vdots \\ b_i \\ \vdots \\ b_n \end{bmatrix}$$

$$\Rightarrow x_j = \left(b_i - \sum_{k \neq j} a_{ik} x_k \right) / a_{ij}$$

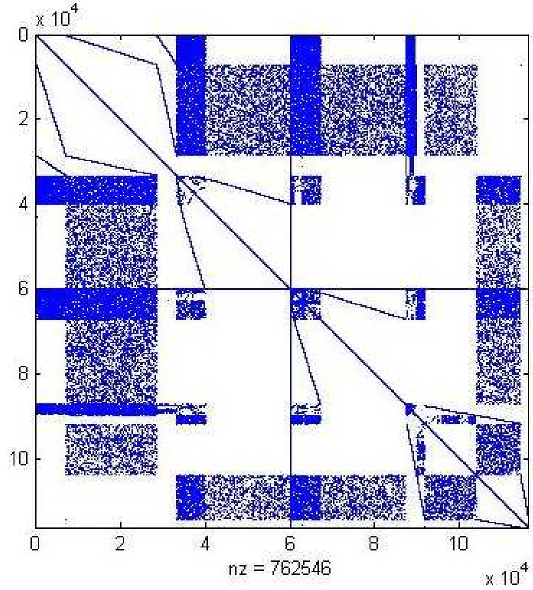
Load Balancing / Partitioning
(Device Evaluation vs. Matrix Structure)

- Balancing Jacobian loads with matrix partitioning for iterative solvers
  - Use different partitioning for Jacobian loads and solves
  - Simple distribution of devices across processors
    - Evaluation can be multi-threaded

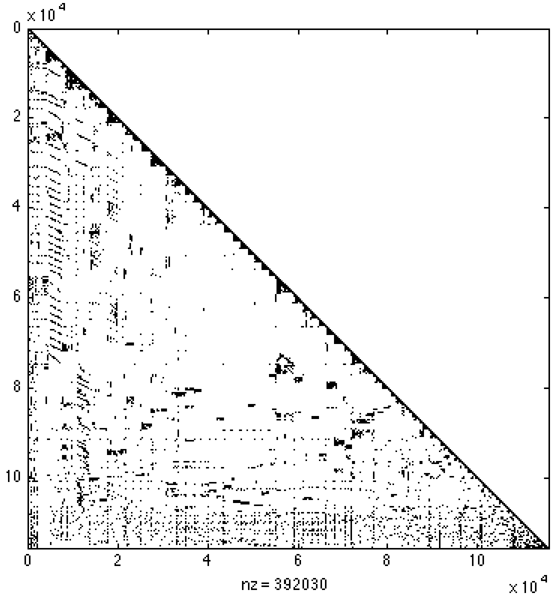


- Matrix partitioning more challenging:
  - Graph
    - Assumes symmetric structure
    - Robust software available (ParMETIS, etc.)
  - Hypergraph
    - Works on rectangular, non-symmetric matrices
    - Newer algorithms (Zoltan, etc.)
    - Expensive, but more accurately measures communication volume

Network Connectivity
(Hierarchical Structure)



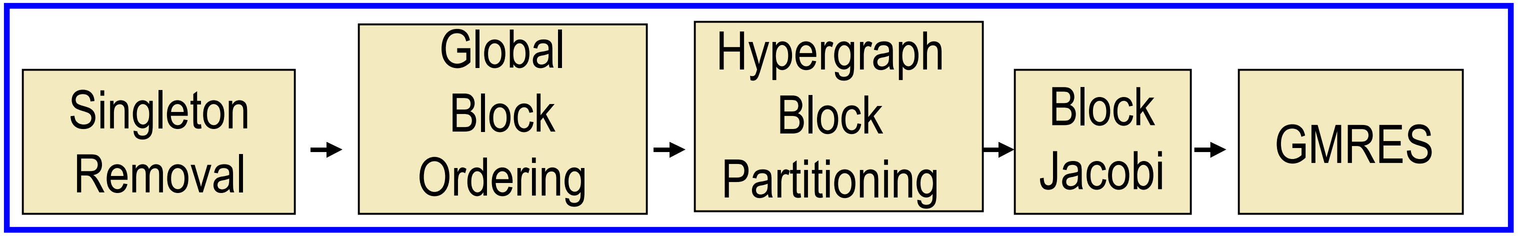
- Some circuits exhibit *unidirectionality*:
  - Common in CMOS Memory circuits
  - Not present in circuits with parasitics
  - Block Triangular Form (BTF) via Dulmage-Mendelsohn permutation



- BTF benefits both direct and preconditioned iterative methods

- Used by Tim Davis's KLU in Trilinos/Amesos (The "Clark Kent" of Direct Solvers)

BTF Preconditioned Solver Strategy

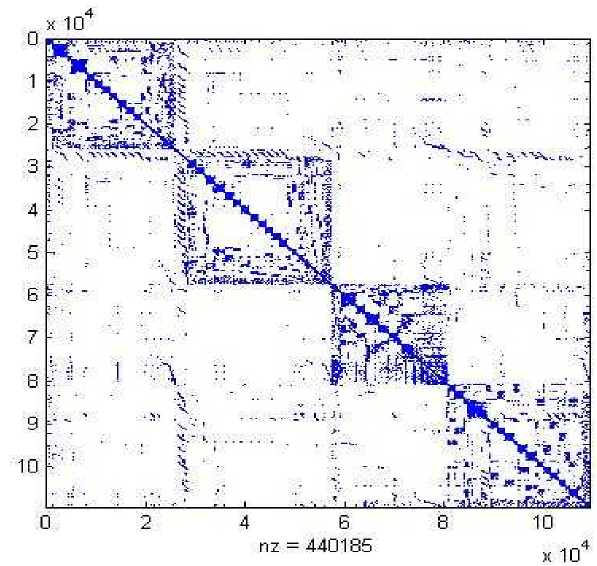


- *"A Parallel Preconditioning Strategy for Efficient Transistor-Level Circuit Simulation"*

  E.G.Boman, D.M. Day, R.J. Hoekstra, E.R. Keiter, H.K. Thornquist

- Improved on previous approaches:
  - –Using global matrix structure
  - –Block partitioning

BTF+Hypergraph
(4 procs)



BTF Preconditioned Solver Strategy
(Scaling Study)

- Xyce 680k ASIC
- Tri-Lab Linux Capacity Cluster (TLCC)

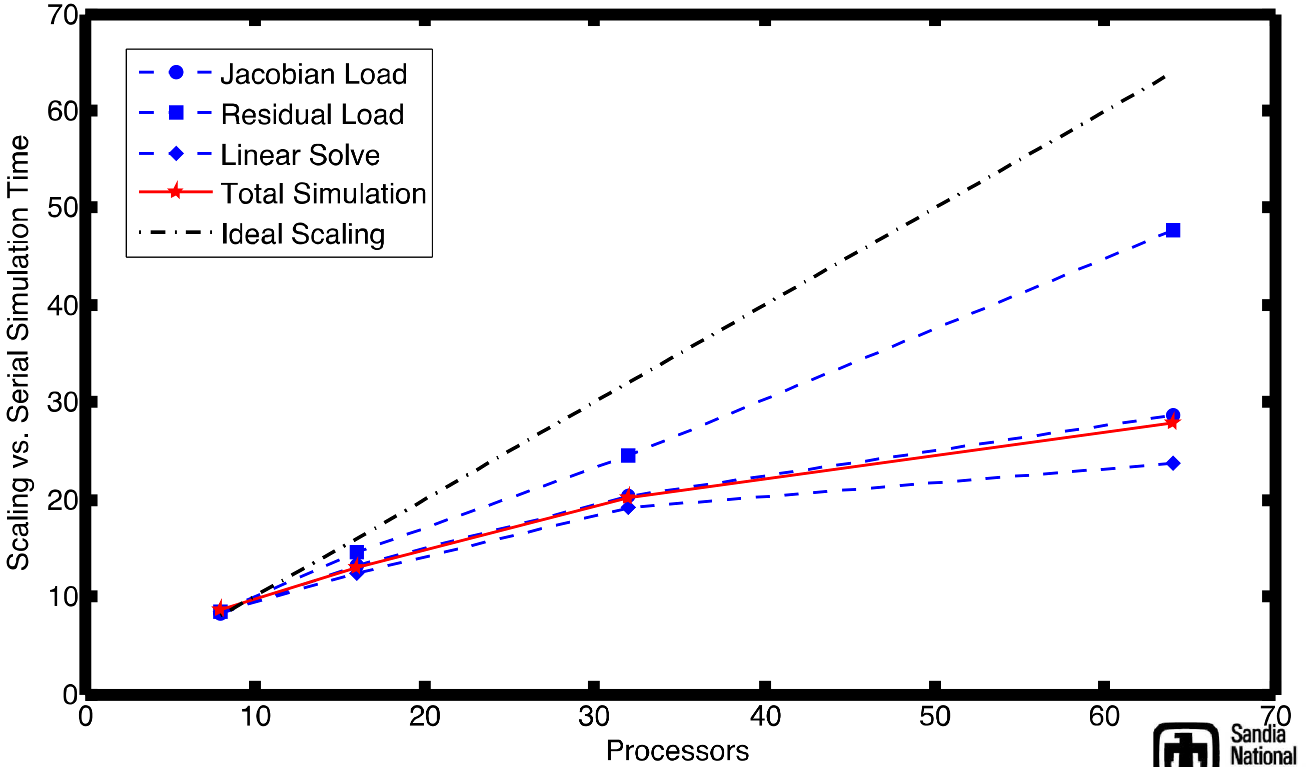
  2.2 GHz AMD four-socket, quad-core processors

  Infiniband interconnect

  32 GB DDR2 RAM,

  divided evenly across

  4 cores



Simulation Results

(Test Circuits)

Circuit	N	Capacitors	MOSFETs	Resistors	Voltage Sources	Diodes
ckt1	688838	93	222481	175	75	291761
ckt2	434749	161408	61054	276676	12	49986
ckt3	116247	52552	69085	76079	137	0
ckt4	63761	208236	11732	51947	56	0
ckt5	46850	21548	18816	0	21	0
ckt6	32632	156	13880	0	23	0
ckt7	25187	0	71097	0	264	0
ckt8	17788	14274	7454	0	15	0
ckt9	15622	7507	10173	11057	29	0
ckt10	10217	460	4243	1	23	0

Simulation Results

(16 Cores)

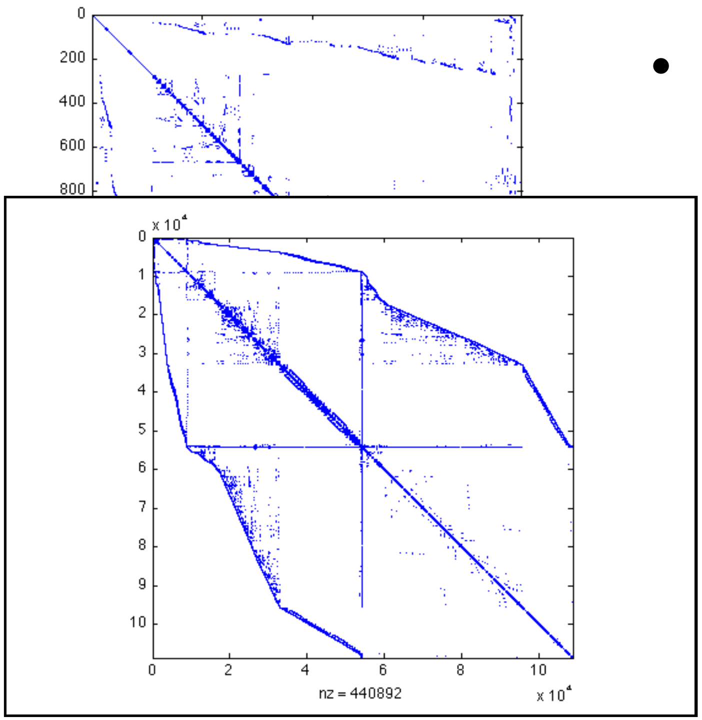
Circuit	Task	KLU (serial)	SuperLU Dist	ParMETIS + ILU	BTF + Hypergraph	Speedup (KLU/BTF)
ckt1	Setup	2396	F3	207	199	**12.0x**
	Load	2063	F3	194	180	**11.4x**
	Solve	1674	F3	3573	310	**5.4x**
	Total	**6308**	**F3**	**4001**	**717**	**8.8x**
ckt3	Setup	131	29	F2	29	**4.5x**
	Load	741	181	F2	175	**4.2x**
	Solve	6699	1271	F2	84	**79.8x**
	Total	**7983**	**1470**	**F2**	**306**	**26.1x**
ckt4	Setup	552	32	F2	F1	**-**
	Load	153	21	F2	F1	**-**
	Solve	106	133	F2	F1	**-**
	Total	**840**	**192**	**F2**	**F1**	**-**

F1 = BTF large irreducible block
F2 = Newton convergence failure

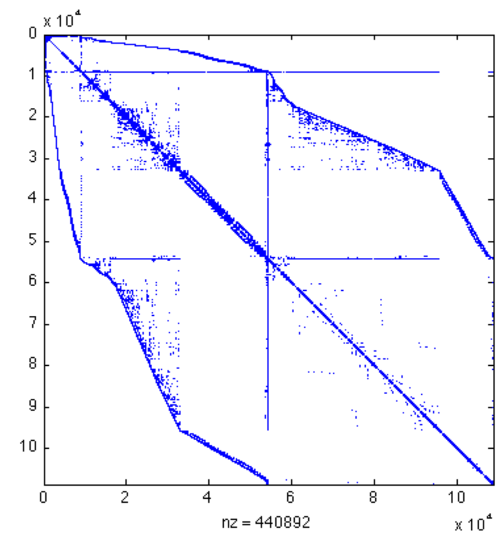
F3 = Out of memory

Network Connectivity

(Parasitics)



- Other circuits **do not** exhibit *unidirectionality*:
  - Common in circuits with modern MOSFETs
  - Common in post-layout circuits
    - circuits with parasitics
    - important for design verification
    - often **much** larger than original circuit



- Dulmage-Mendelsohn permutation results in large irreducible block

Other Linear Solver Strategies

(for circuit simulation)

- The SPICE industry standard is Markowitz ordering
  - BTF structure is known, but KLU is not fully adopted
- Preconditioned iterative methods presented before
  - C. W. Bomhof and H.A. van der Vorst [NLAA, 2000]
    - Requires doubly bordered block diagonal matrix partition
  - A. Basermann, U. Jaekel, and K. Hachiya [SIAM LA 2003 proc.]
    - Requires ParMETIS to give good initial ordering
  - H. Peng and C.K. Cheng [DATE 2009 proc.]
    - Domain decomposition approach, requires knowledge of device boundaries
  - …

Doubly Bordered Block Diagonal Matrix Partition

- *"A Parallel Linear System Solver for Circuit Simulation Problems"*

  [C. W. Bomhoff and H.A. van der Vorst]

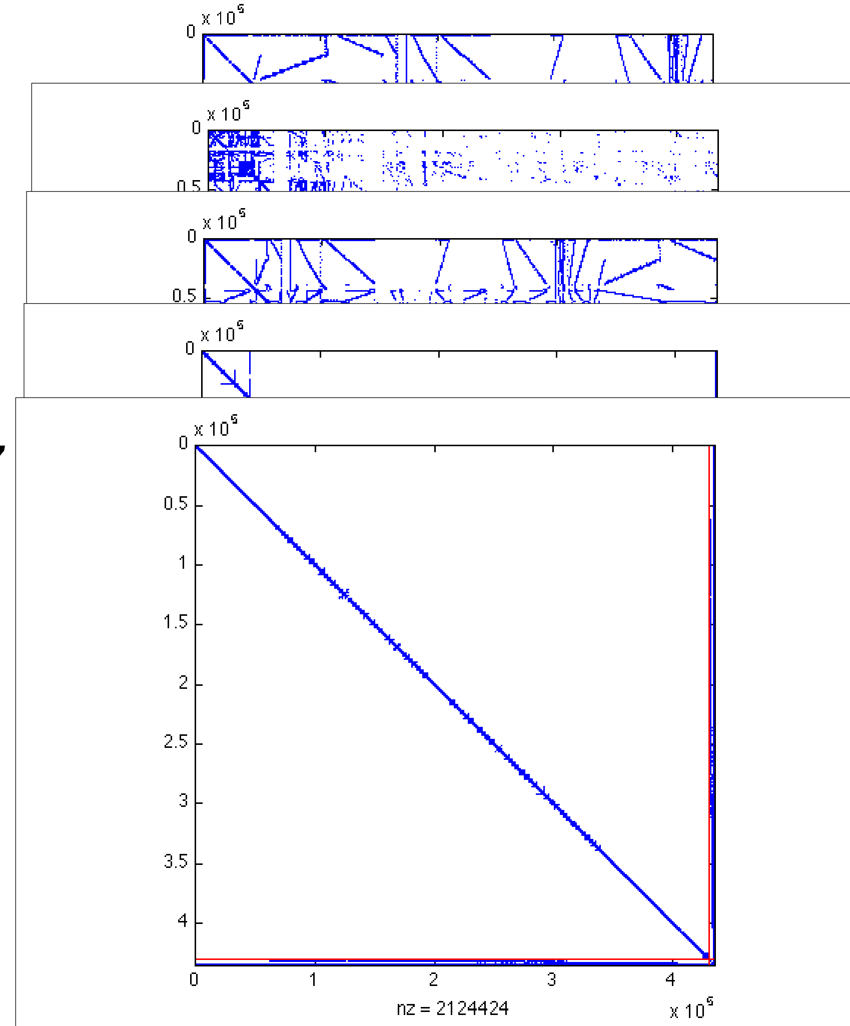
$$\begin{bmatrix} A_{1,1} & 0\cdots & 0 & A_{1,m} \\ 0 & \ddots & \vdots & \\ \vdots & & 0 & A_{2,m} \\ 0\cdots & 0 & A_{m-1,m-1} & \vdots \\ A_{m,1} & A_{m,2} & \square & A_{m,m} \end{bmatrix}$$

- For circuits, size($A_{m,m}$) < size(A) / *20*
- Hybrid iterative / direct method
  - Initial fill reducing ordering (global)
  - Direct solves on diagonal (delayed pivoting)
  - Preconditioned iterative Schur complement solve
- Use elimination tree of $A+A^T$ to determine the partition
  - METIS

Circuit Matrix Examples

(Master tile circuit)

- Sandia circuit design
  - master tile in a 2D array
  - 549144 devices
- dim(A) = *434749*
- BTF: irreducible block size *334767*
- Elimination tree height: *908*
- Direct / iterative breakdown:
  - *Direct : 429974 rows*
  - *Iterative : 4775 rows*



Conclusions

- Iterative linear solvers can enable scalable circuit simulation
  - Dependent upon choosing correct preconditioning strategy
- BTF preconditioning strategy has been successful
  - Great for CMOS memory circuits (ckt3) and Xyce 680k ASIC (ckt1)
- But it is still not a silver bullet …
  - Circuits with parasitics are more challenging (ckt4)
- Hybrid direct / iterative techniques are promising
  - Can help to more efficiently precondition circuits with large irreducible blocks
- Robust integration of iterative linear solvers into circuit simulation
  - Graph analysis based linear solver strategies

Acknowledgements

- **Sandia** researchers:
  - Eric Keiter
  - David Day
  - Erik Boman
  - Mike Heroux
  - Robert Hoekstra

Xyce™ PARALLEL ELECTRONIC SIMULATOR **Development team**

Questions?