# Xyce Parallel Circuit Simulator

http://xyce.sandia.gov

Heidi Thornquist

Sandia National Laboratories

Electrical Systems Modeling Dept.

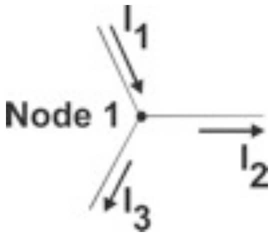
# A) Project Overview

- The project: Large-scale analog circuit simulation
- Science goals: Enable predictive simulation of large circuits in normal and hostile environments
- The team: 6 developers, differing backgrounds & levels of participation
- History: Project started 10 years ago
  - NNSA shift from test-based to simulation-based confidence
- Sponsor: ASC
- Goals: To develop a simulator, competitive with

# B) Science Lesson

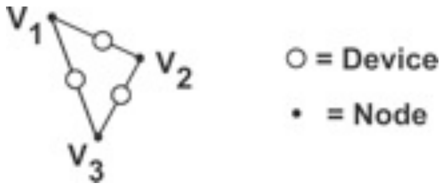
- Xyce is an analog circuit simulator (SPICE compatible)
- Models network(s) of devices coupled via Kirchoff's current and voltage laws (modified KCL)
  - Most equations are Kirchoff Current Law (KCL) equations.
  - Most solution variables are nodal voltages.
  - Most currents are obtained via

$$\sum_{i=0}^{n}(I_n) = 0$$



Kirchoff's Current Law (KCL)

$$\sum_{i=0}^{n}(\Delta V) = 0$$



Kirchoff's Voltage Law (KVL)

$$G = \frac{\delta I}{\delta V} = \text{conductance} = \text{Jacobian term}$$

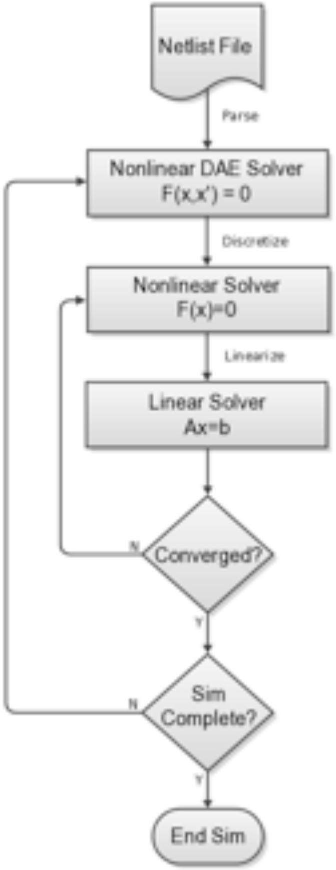
$$\text{Ohm's Law: } I = GV = V/R$$

# C) Parallel Programming Model

- Parallelism: MPI, with some OpenMP
- Language: C++
- Libraries: Trilinos, LAPACK/BLAS, Zoltan/ ParMETIS, AMD (SuiteSparse)
- Other infrastructure: None
- Platforms: Unix/Linux, Windows, OSX
- Current status: Migration to a hybrid model, that employs more threading for smaller-scale computations (device evaluation, etc.)

# D) Computational Methods

- Algorithms: implicit time integration (trap, BDF), nonlinear solver (Newton’s method), linear solver (KLU, Ksparse, preconditioned GMRES)
- FFT needed for frequency analysis methods, like harmonic balance (HB)
- Currently developing scalable, robust parallel preconditioners, parallel HB capabilities, AD for faster, error-free model development



# E) I/O Patterns and Strategy

- Input I/O and output I/O patterns:
  - Input: Single netlist file, read in on proc 0. Minimal processing on proc 0 before device lines streamed to all other processors.
  - Output: Single output file containing requested circuit information (voltages, currents, etc.)
- Approximate sizes of inputs and outputs
  - Netlist files can be large for modern ASIC designs
  - Output file size depends on amount of information requested and simulation parameters
- Checkpoint / Restart capabilities:
  - Can dump DCOP solution for restart
- Current status and future plans for I/O

# F) Visualization and Analysis

- How do you explore the data generated?
  - ASCII file output (Matlab, GNUplot, etc.)
- Do you have a visualization workflow?
  - Not currently, our customers do more visualization than us
- Current status and future plans for your viz and analysis
  - Working on visualization techniques for large ensemble calculations (UQ studies)

# G) Performance

- What tools do you use now to explore performance:
  - None
- What do you believe is your current bottleneck to better performance?
  - Parsing and better linear solvers.  Threading can provide a minor performance improvement.
- What do you believe is your current bottleneck to better scaling?  Scalable iterative linear solvers
- What features would you like to see in perf tools
  - Ease of instrumentation and different measurements
- Current status and future plans for improving

# H) Tools

- How do you debug your code?
  - Purify, gdb, valgrind, std::cout/printf
- What other tools do you use?
  - Version control (cvs), c[make/test/dash]
- Current status and future plans for improved tool integration and support
  - Tool integration on as needed basis

# I) Status and Scalability

- How does your application scale now?
  - Some parts of the code scale reasonably well, linear solvers are the scaling bottleneck.
- Where do you want to be in a year?
  - Better fine-grain parallelism, more robust solvers, higher capacity parsing capability (possibly parallel)
- What are your top 5 pains? (be specific)
  - 1 (lack of a scalable linear solver), 2 (serial bottlenecks in parser), 3 (too slow on small circuits), 4 (hidden performance issues in simulator), 5 (underlying parallelization framework)
- What did you change to achieve current scalability?
  - Separated device evaluation partitioning from linear solver partitioning

# J) Roadmap

- Where will your science take you over the next 2 years?
  - Larger ASICs to simulate in support of LEPs, possibly new compact models to correctly determine the physics. . Electrical system qualification, which incorporates numerical simulation evidence, is expected to be a growth area.
- What do you hope to learn / discover?
  - More robust, scalable solution techniques. Eradicate performance issues for small-to-large scale circuit simulation.
- What improvements will you need to make
  - Algorithms for preconditioned linear solvers, effectively integrating threading to achieve performance gains