A Quasi-Newton Preconditioned Newton-Krylov Method for Robust and Efficient Time-Domain Integrated Circuit Simulation

Zhao Li and C.-J. Richard Shi
Department of Electrical Engineering
University of Washington
01/26/2006

Presented by Sheldon X. D. Tan
Department of Electrical Engineering
University of California at Riverside
Outline

• Motivation
• Review of previous work
• Preconditioned Newton-Krylov Method
• Conclusions
Outline

• Motivation

• Review of previous work

• Preconditioned Newton-Krylov Method

• Conclusions
Deep-Submicron VLSI Circuit Simulation

- Couplings with surrounding parasitic environments

Diagram showing the interconnections between Digital/Analog/RF Circuits, Interconnect, Digital/Analog/RF Circuits, and Substrate (Electrical / Thermal)
High Frequency Analog/RF Circuit Simulation

- Coupled circuit-EM simulation
  - Sensitive structures modeled as large-scale strongly coupled linear networks, such as inductors in a LNA modeled with the PEEC method

- Extracted parasitics due to transistor layouts
  - Complicated layouts for transistors in an analog/RF circuit
Challenges for SPICE Time-Domain Simulation

• Big Linear Small Nonlinear
  – The number of linear elements modeling parasitic effects is much larger than the number of nonlinear devices

• Parasitic couplings make circuit structures denser and strongly coupled
  – Circuit matrices become denser and hard to partition

• Per-iteration cost is dominated by LU factorization
Outline

• Motivation

• Review of previous work

• Preconditioned Newton-Krylov Method

• Conclusions
Previous Work

• Keep circuit matrices constant
  – Fixed leading coefficient numerical integration
  – Quasi-Newton methods for nonlinear iteration

• Make circuit matrices sparse (Partition)
  – Relaxation based iterative methods (Gauss-Seidel, SOR)
  – Semi-implicit methods (Phillips, ICCAD 2001)
  – Alternating-Direction-Implicit methods (Lee, ICCAD 2001)

• Use krylov subspace iterative methods
  – Conjugate gradient methods preconditioned with incomplete
    Cholesky decomposition (Chen, DAC 2001)
  – Multi-grid methods (Nassif, DAC 2000)
Our Viewpoints

• LU factorization based direct methods in SPICE are robust and accurate
  – Simulate nonlinear and linear circuits in a circuit matrix
  – Memory requirement is generally a minor factor

• To achieve efficiency, circuit matrices should be kept constant to reduce the number of LU factorizations
Matrix Change due to Numerical Integration (Trapezoid Formula)

\[ x_n \approx \frac{2}{h_n} (x_n - x_{n-1}) - x_{n-1} \quad (h_n = t_n - t_{n-1}) \]

• New LU factorization if \( h_n \) changes
Matrix Change due to Nonlinear Devices (Newton-Raphson Method)

\[ i^{k+1} = f(v^{k+1}) \approx f(v^k)'v^{k+1} + f(v^k) - f(v^k)'v^k \]

- New LU factorization if \( f(v^k)' \) changes
Outline

• Motivation
• Review of previous work
• Preconditioned Newton-Krylov Method
• Conclusions
Quasi-Newton Based Iterative Methods

• Solve $Ax = b$ with quasi-Newton methods

$$x^{(k)} = x^{(k-1)} + M^{-1} (b - Ax^{(k-1)})$$

single search direction per iteration
non-orthogonal
(slow convergence)

• How to achieve constant $M$?
  – Fixed leading coefficient numerical integration
  – Quasi-Newton method for nonlinear iteration
Krylov Subspace Based Iterative Methods

- Solve \( Ax = b \) with krylov subspace methods
  \[ M^{-1}Ax = M^{-1}b \]
  \[ x = x^{(0)} + \kappa_m(M^{-1}Ax^{(0)}) \]
  \[ \kappa_m(M^{-1}Ax^{(0)}) = \text{span}(r^{(0)}, M^{-1}Ar^{(0)}, \ldots, (M^{-1}A)^{m-1}r^{(0)}), \quad r^{(0)} = M^{-1}(b - Ax^{(0)}) \]

  \( m \)-dimensional orthogonal krylov subspace
  (better convergence)
Krylov Subspace Based Iterative Methods (cont.)

• How to achieve constant $M$ (preconditioner)?
  – Quasi-Newton like time step-size control
  – Piecewise weakly nonlinear definition of nonlinear devices

• How to construct $Ax=b$?
  – Standard implicit numerical integration (better stability)
  – Original nonlinear device models (better accuracy)
Quasi-Newton like Time Step-Size Control

- $G$ and $C$ represent conductance and capacitance/susceptance matrices, $b$ is the input vector
- $Gx + C \dot{x} = b$
- $h_n = \alpha h$
- $\left(G + \frac{2C}{\alpha h}\right)^{\circ} = \frac{2C}{\alpha h} x_{n-1} + C \dot{x}_{n-1} + b$
- Quasi-Newton Preconditioner
- $\left|\left(G + \frac{2C}{h}\right)^{-1} - I\right| = \left|\left(G + \frac{2C}{h}\right)^{-1} \left(1 - \frac{1}{\alpha}\right) \frac{2C}{h}\right| < \eta < 1$
- $\left|1 - \frac{1}{\alpha} \frac{1}{1 - z}\right| < \eta < 1$, $z = -\frac{h}{2\tau}$, $\tau = \text{eig}(G^{-1}C)$
Effective Preconditioner Region

\[ |z - 1| > 1, \quad \eta = \left| 1 - \frac{1}{\alpha} \right| < 1 \]

\[ \alpha > \frac{1}{2} \]

• Effective preconditioner region covers all left-half z-plane poles
Piecewise Weakly Nonlinear Definition

- Nonlinear functions are partitioned into a few PWNL regions
- Fixed chords within PWNL regions
  - Constant circuit matrices
- Chords change for new PWNL regions
Convergence of Quasi-Newton Nonlinear Iteration

- Quasi-Newton nonlinear iteration within a PWNL region

\[ \varepsilon_{i+1} \approx \varepsilon_i \left(1 - \frac{f'(x_i)}{g} - \varepsilon_i^2 \frac{f'''(x_i)}{2g} \right) \]

\[ \left| 1 - \frac{f'(x_i)}{g} \right| < \delta, \quad 0 < \delta < 1 \]

- Guideline for generating PWNL regions

\[ \frac{f'_\text{max}}{1 + \delta} < g < \frac{f'_\text{min}}{1 - \delta} \]
Generalized MOSFET PWNL Definition

- PWC regions of $g_{ds}$ and $g_{m}$ are equivalent to PWNL regions of $I_{ds}$
  
  $$g_n = g_{\text{max}}$$
  
  $$g_{i-1} = (1 - \delta)g_i, \quad i = n, \ldots, 2$$

- Generalized MOSFET PWNL definition used to construct quasi-Newton preconditioner $M$

- Original MOSFET models for $Ax = b$
Low Rank Update

• L and U matrices of $A_{\text{new}}$ derived from those of $A_{\text{old}}$ if

$$A_{\text{new}} = A_{\text{old}} + cr^T$$

$A_{\text{new}}$ and $A_{\text{old}}$ are $n \times n$ matrices, $c$ and $r$ are $n \times m$ vectors

$m << n$

• Efficient if the number of nonlinear devices switching PWNL regions is small
Approximate Preconditioner

• The preconditioning process is to solve $My=c$
  – LU factorization of $M$
  – ILU factorization of $M$
  – Krylov subspace methods to solve $My=c$, such as FGMRES

• ILU preconditioner
  – Reduce the preconditioning cost of LU preconditioner

• Hybrid preconditioner
  – Solve $y^0$ with ILU preconditioner
  – Solve $y$ with the FGMRES method starting from $y^0$
  – More preconditioning efforts for less FGMRES iterations
In complete LU Preconditioner

\[ A = LU = \begin{bmatrix} 1_{11} & \cdot & \cdot & 0 \\ \cdot & \cdot & 1_{ii} & \cdot \\ \cdot & \cdot & \cdot & 1_{jj} \\ \cdot & \cdot & \cdot & \cdot & 1_{nn} \end{bmatrix} \begin{bmatrix} 1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & 1 \end{bmatrix} \]

- ILU preconditioner derived by removing small elements in already factorized L and U matrices
  \[-|l_{ij}| < c \cdot \max(|l_{*i}|), \quad |u_{ij}| < c \cdot \max(|u_{i*}|), \quad c=0.001\]

- ILU preconditioner updated by low rank update during nonlinear iteration
General Nonlinear Circuits – Normalized #LU

- #Tran LU is reduced to below 1/5
- #Tran Iter is kept almost the same as that of SPICE3
General Nonlinear Circuits - #FGMRES Iteration

- #FGMRES Iter per FGMRES call is generally below 5 with $\varepsilon = 1e-10$
Power/Ground Example

IR drop and L*dl/dt effects

Output

Input
Run Time vs. #Elements

- SPICE3
- FGMRES w LU Preconditioner
- FGMRES w ILU Preconditioner

20X
Histogram of Number of L/U Matrix Elements in ILU Preconditioner

- #L/U matrix elements is reduced to $1/4 \sim 1/15$ of that in LU preconditioner ($\sim 3e6$)
Run Time vs. ILU Coefficient $c$

- $c = 0.001$ is optimal for tested power/ground examples
In most nonlinear iterations, the number of switching MOSFETs is small – low-rank update is efficient.
• Less #FGMRES Iter per FGMRES call with hybrid preconditioner, however more FGMRES calls
### Power/Ground Simulation Results

<table>
<thead>
<tr>
<th>#Elems</th>
<th>SPICE3</th>
<th>FGMRES (LU)</th>
<th>FGMRES (ILU)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Tran Iter</td>
<td>#Tran Iter</td>
<td>#Tran LU</td>
</tr>
<tr>
<td>4002</td>
<td>4023</td>
<td>4106</td>
<td>54</td>
</tr>
<tr>
<td>34802</td>
<td>4006</td>
<td>4087</td>
<td>55</td>
</tr>
<tr>
<td>61602</td>
<td>4377</td>
<td>4253</td>
<td>53</td>
</tr>
</tbody>
</table>

- #Tran LU is reduced to about 1/80
- #Tran Iter is kept almost the same as that of SPICE3
Outline

- Motivation
- Review of previous work
- Preconditioned Newton-Krylov Method
- Conclusions
Conclusions

• Quasi-Newton Preconditioned Newton-Krylov Method
  – Quasi-Newton like time step-size control for preconditioner construction
  – PWNL definition of nonlinear devices for quasi-Newton preconditioner
  – Low-rank update for fast L/U matrix update
  – Incomplete LU preconditioner

Orders of magnitude speedup for circuits where the number of linear parasitic elements dominates the number of nonlinear devices
Thank you!