Accelerated, Parallelized Integral Equation Techniques for Packaged Microelectronics

Vikram Jandhyala
Founder and CEO, Physware Inc., Bellevue WA
And
Associate Professor
Director, Advanced Computational Engineering Lab
Dept of Electrical Engineering
University of Washington
Seattle
Acknowledgements

ACE Lab: Dr. Indranil Chowdhury (Cadence), Dr. Yong Wang (Synopsys), Dr. Chuanyi Yang (Cadence), Ani Siripuram, Arun Sathanur, Ritochit Chakraborty, Ying Li, Mosin Mondal

Physware, Inc.: Dr. Dipanjan Gope, Dr. Swagato Chakraborty, Dr. Feng Ling, Dr. Xiren Wang, Dr. Junho Cha, Dr. Robert Chao, James Pingenot, Devan Williams

Sponsors and Collaborators: DARPA, NSF, SRC, INTEL, IBM, NVIDIA, SPAWAR, AFRL, TGIF (UW), WRF, RRF (UW), BOEING, LLNL, DOD (SBIR)
Challenges

- **Complexity**
  - Mixed-signal, broadband, material effects, multiscale crosstalk mechanisms

- **Scaling and Scale**
  - Large system sizes, Smaller features, increased crosstalk and proximity, 3D SoC and SIP, material effects, chip-scale integration

- **Variability**
  - Manufacturing and process variability, yield prediction and control

- **Design**
  - Rapid parametric solution, fast incremental modeling, design cycle acceleration
Approach

- Rapid EM Solvers
  - Fast $O(N \log N)$ integral equation solvers
- EM-SPICE coupling
  - EM and SPICE co-simulation and coupled solution
- Parallelization
  - Fast parallelized tree algorithms for multicore / cluster configurations
- Geometry and Mesh
  - Application specific mesh optimization, macromodeling, and geometry processing
- Variability and statistical modeling
  - Accelerated Monte-Carlo techniques for multiobjective and multi-parameter variability and optimization
Computational cost of the MoM

A graph showing the computational cost over the number of unknowns.
The curse of complexity

- An explicit inversion scheme for an MoM matrix has a cost of approximately $\frac{2}{3} N^3$ (e.g. LU decomposition) and as can be seen becomes very expensive for large problems.

- A problem with a million unknowns would require more than 300 years to solve with LU decomposition!!

- If algorithms could be devised that scale as $N$ or $N\log N$, even with very large constants (5000 or 10000), the time savings are dramatic. The same problem would require only a few minutes to solve!
Memory

- Storing the MoM matrix: Huge bottleneck!
Multilevel Tree-Based N-Body Methods

- Two analogies
  - FFTs in space
  - Trans-Atlantic landline topology
Multilevel QR Based Compression Scheme

\[ m \times n \]

\[ Z_{N \times N} \]

\[ A_{m \times n} \]

n Sources

\[ A_{m \times n} = Q_{m \times r} \]

m Observers

\[ Q_k = (A_k - \sum_{i=1}^{k-1} R_{ik} Q_i) / R_{kk} \]

\[ R_{ik} = Q_i^T A_k \]

k = 1...r

r < (m,n)

Memory and solve time per RHS reduced by

\[ \frac{m \times n}{(m + n) \times r} \]
Parallelization

- Availability of Shared Memory Multicore CPUs growing
- Chip companies claim 100 cores in 5-7 years is a reality
- Low-cost clusters with distributed memory also growing
- Need true parallelized simulation methods
- Amdahl’s Law: Your parallelization is limited by the percentage of serial code
Parallel Architectures

- Hybrid Memory
  - Clusters of SMPs
Multicore Paradigms- Here to Stay

- Frequency Scaling Slowing Down – Stopped!
- More Cores with Shared Memory
  - Most users have multicore, clusters less common
- No Free Lunch for S/W: time to parallelize and parallelize correctly
  - No memory overhead; Thread safety; Amdahl’s Law!
- Significantly more challenging than distributed / MPI simulation
- Discussed in embedded tutorial in EPEP 07
Parallelization: Load Distribution-Near Field

Link list of neighbor lists

Near field interaction workload distribution

\[ \frac{N_{mom}}{N_p} : \text{processor workload} \]

Load Balancing

\[ N_{mom} : \text{number of Mom entries} \]

\[ N_p : \text{number of processors} \]
Load Distribution-Far Field

Far field interaction workload distribution

Predetermined Rank Map  Load Balancing
Dynamic Load Balancing for Sparse Approximate Inverse

Not done

Done

T₀

T₁

T₂
Matrix-vector multiplication and data collection

MPI Reduce

Data Collection

Data Operation
Setup time vs no. of processors

Y-axis: (setup time for 1 processor/setup time of N processors)
Setup time scales linearly with the number of processors
Solve time vs no. of processors: Effect of Amdahl’s Law
Scaling with Serial Bottlenecks Removed

- Speedup for matvec

- For larger problems, approaches closer to the ideal
Challenges for fast convergence

- Convergence is the bottleneck for a true $O(N)$ solution in cases of:
  - Electrically small packages – low frequency simulation in a broadband application
  - Nonuniform mesh density – realistic package layouts, vias
  - Thin and long/wide metal – metal layers in a package

- Practically any realistic microelectronic simulation has these features

- Charge and Current in the EFIE leads to a low-frequency problem related to separation into curl-free and divergence-free solutions
Automatic loop detection

- Local loops around internal vertices
- Global loops around holes, handles and junctions
Automatic global loop detection

Loop around a hole

Loop around a handle
The complete three stage preconditioner has converged to the correct LU solution. The rest are far from the correct solution for the given residual of $1 \times 10^{-5}$. In general: error in solution $\leq$ residual $\times$ condition number of the matrix.
## DDR Designs for Hard Disk Storage

<table>
<thead>
<tr>
<th>PhysPack</th>
<th>[Design Details not shown]</th>
<th>1 GB</th>
<th>2min 30sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEM</td>
<td></td>
<td>4.5 GB</td>
<td>10 min</td>
</tr>
<tr>
<td>PhysPack</td>
<td></td>
<td>5 GB</td>
<td>1 hour</td>
</tr>
<tr>
<td>FEM</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FEM could not solve in 16 GB

---

**Graphs:**

1. **Return Loss of RX Nets**
   - PHYware, RXP
   - PHYware, RXM
   - FEM, RXM
   - FEM, RXP

2. **Insertion Loss of RX Nets**
   - PHYware, RXP
   - PHYware, RXM
   - FEM, RXM
   - FEM, RXP
Impedance on Package-Board

Red: Simulation
Blue: Measurement

Time: 10 minutes Per Frequency Point
Memory: 5.7GB
Geometry Challenges

- Mesh Refinement and Geometry “Cleanup” need to be accomplished simultaneously
Package Modeling

Complete Package Structure
Automatic Port Generation

Bondwire Side

Solder Ball Side
Speed and Memory Profile

Minutes per frequency solve: 30 seconds

Minutes for 40 freq points: 20 minutes

Memory: 750 MB

Processor: 8 core Intel Xeon @2.66GHz
4x4 Steerable Array Antenna with Phase Shifters

S Parameters

- S11 (Measured)
- S11 (Simulated)
Radiation Pattern Plot

- E Plane (Measured)
- E Plane (PhysPack)
What is Next?

- Moving from Verification and Modeling to Design
  - Rapid Design Iteration
  - What-If Simulation
  - Modeling Manufacturing Variability

- Seamless Integration
  - Coupling to Layout-Level and Schematic-Level Simulators
  - Generation of Broadband Time-Domain Models
  - Back-Annotation to Layout Tools
Variability Modeling

Manufacturing Variability
Fast Parametrics and Optimization
Statistical and Yield Models
Integration with SPICE

Approach: Adaptive Response Surfaces
PDF generation

PDF of L
- Actual distribution of L
- The Gaussian distribution with same μ and σ

PDF of R at 10 GHz
- Actual distribution of R
- The Gaussian distribution with same μ and σ

PDF of Q at 10 GHz
- Actual distribution of Q
- The Gaussian distribution with same μ and σ

Influence of conductivity and trace width variances on the PDF of Q of the inductor at 10 GHz
- Type 1 Variation
- Type 2 Variation
Yield Modeling

![Distribution of Optimal Inductors](image)

- All Inductors
- Optimal Inductors

Inductance in nH

Quality factor

Distribution of Optimal inductors (L within 5% of mean and Q > 9)
PDF of CMRR/Differential $S_{11}$

PDF of the CMRR

PDF of the differential return loss

**CMRR** : Note the mean of the CMRR shifts to a lower value as process variations increase.

Differential Return loss
Sample yield diagram for differential LNA

- Criterion: Differential S11 <= -15 dB and Diff Gain >= 16 dB

<table>
<thead>
<tr>
<th>Circuit Performance</th>
<th>Type 1 Variation</th>
<th>Type 2 Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diff Gain &gt; 16 dB</td>
<td>99.7%</td>
<td>90.1%</td>
</tr>
<tr>
<td>DS_{11} &lt; -15 dB</td>
<td>99.9%</td>
<td>95.2%</td>
</tr>
<tr>
<td>Overall</td>
<td>99.6%</td>
<td>85.4%</td>
</tr>
</tbody>
</table>
Differential return loss for the package as a function of Build-up and Core Voiding extensions

Package Optimization:
UWEE ACE Lab/Physware/Intel
Summary

- **Rapid Advances in Integral Equation-Based Simulation for Packaged Microelectronics**
  - Goes beyond state of the art finite element and finite difference based competing approaches
  - Enables Verification and Modeling at Unprecedented Scale and Speed

- **Significant Challenges Remain!**
  - Technological: Design, Synthesis, Variability, Integration, Multiphysics
  - EDA Community: Preponderance of protected / proprietary and even incompatible formats impedes integration
    - “Compete on core engines and technology, not on file formats!”

---

**Note:**
- The content is a summary of a slide presentation titled "Rapid Advances in Integral Equation-Based Simulation for Packaged Microelectronics." It highlights the advancements in simulation techniques and the challenges they face, emphasizing the need for core engines and technology over file formats.

---

**Technical Details:**
- **Integral Equations:** Integral equations are equations in which an unknown function appears under an integral sign. They are used in various fields of science and engineering, particularly in the study of electromagnetic, fluid, and structural problems.
- **Finite Element Method (FEM):** A numerical technique for finding approximate solutions to various problems, including those involving partial differential equations, by decomposing complex systems into simpler pieces called finite elements.
- **Finite Difference Method (FDM):** A method for obtaining numerical solutions to differential equations by approximating derivatives with differences, suitable for both linear and nonlinear problems.

---

**Integration Challenges:**
- The EDA community faces significant challenges with the preponderance of protected and proprietary formats, which can lead to integration difficulties.
- The recommendation to “Compete on core engines and technology, not on file formats” underscores the need for open standards and interoperable solutions to enhance efficiency and innovation in the EDA field.