

Accelerated Integral Equation Solution Methods on Clusters and Multicore Processors

Session Name: Algorithm and techniques for parallel processing for EMI/EMC

SPECIAL SESSION

Vikram Jandhyala and Xiren Wang

Advanced Computational Engineering Lab, Department of Electrical Engineering, University of Washington Seattle WA 98195 vj@u.washington.edu 206-543-2186

ABSTRACT

Boundary element based methods are now prevalent as analysis methods for large-scale microelectronic circuits. With the advent of fast multilevel techniques, these are comparable to and often surpass PDE-based solutions. The next challenge is to ensure these algorithms are scaleable on multi-core machines and parallel clusters. This paper discusses the parallelization of multilevel low-rank based algorithms for fast boundary element solvers applied to microelectronics.

1. INTRODUCTION

This paper summarizes advances in the parallelization of integral equation methods for applications to chip-scale and chip-package signal integrity modeling. Despite the well-documented advantages of smaller matrix sizes, absence of absorbing boundary conditions, and surface-only modeling, integral equation based methods however gives rise to a dense matrix the solution of which present a time and memory bottleneck. Various fast Krylov subspace based solver approaches, that exploit the physical properties of electromagnetic interactions captured through the Green's function, are available in existing literature. A few examples are the fast-multipole method (FMM) [1-2], adaptive integral method (AIM) [3], FFT based methods [4] and QR based algorithms [5-6]. These algorithms are capable of reducing the time and memory requirements to linear complexity with respect to the number of unknowns. However, when applied to real-life circuit layouts even the fast methods fall short of acceptable efficiency targets. To mitigate the problem, circuit designers are often forced to perform simulations on partitioned circuits. Though effective in many situations, the partitioning

strategy leads to neglecting coupling effects that may lead to inaccurate simulation of circuit behavior. Consequently, parallel implementations of fast algorithms are the most promising simulation strategy for handling practical circuit layouts. In this work a parallelized implementation of low-rank tree-based algorithms is discussed. The proposed implementation leads to a chip-scale and chip-to-package transition capable electromagnetic solver that is used for signal integrity calculations and model generation for use in SPICE solvers.

2. PARALLELIZED LU DECOMPOSITION

Boundary elements give rise to a system of the form $Ax=b$ where A is a dense matrix. These are either solved by direct solvers or with iterative solvers with fast matrix-vector product techniques.

The basic idea of the parallel LU solver is to map the global matrix A (and b) to p processors. Each processor i only needs to compute and store segments of the matrix (denoted by A_i). Each processor performs an LU-decomposition on A_i . It is imperative to ensure load balancing and minimum inter-process communication. This is done by a well known approach, two-dimensional block cyclic distribution, as shown in Fig. 1. As an illustration, 4 processors are meshed into a 2x2 grid and numbered as 0 to 3. The global matrix is partitioned into 8 x 8 blocks, and this map is used to partition the matrix A onto the processor grid. Blocks that are labeled by i will be distributed to processor i , $i=0..3$.

The individual blocks are decomposed simultaneously, and subsequent local forward-backward substitutions permit sections of the system solution to be obtained. Eventually, all local solutions

are reduced or assembled in the final step towards the global solution.

0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3
0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3
0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3
0	1	0	1	0	1	0	1
2	3	2	3	2	3	2	3

Fig.1 Two-dimensional cyclic matrix distribution scheme.

3. PARALLEL MULTILEVEL LOW-RANK DECOMPOSITION ALGORITHM

The serial multilevel low-rank decomposition algorithm achieves efficient far-field sub-matrix compression through leveraging an “unrolled” oct-tree decomposition hierarchy and a merged interaction list. In this algorithm, the regions occupied by arbitrary shaped electromagnetic structures are hierarchically decomposed into small cubes and these cubes are represented by an Oct-tree data structure. The near field interactions are represented by a neighbor list at the finest level and far-field interactions are represented by interaction lists at all levels. The overall computational workload for the multilevel low-rank decomposition algorithm includes the near-field Method of Moments (MoM) matrices set up, far-field interaction decomposition—Q and R low-rank matrix set up and matrix-vector multiplication. A scalable parallel algorithm requires balanced distribution of the above workload to different processors with minimized inter-processor communication.

Near field interactions are represented by neighbor lists at the finest level and they can be grouped into a linked list data structure. Each node denotes a neighbor interaction list and contains information on source and observer regions. To minimize the inter-processor communication, each processor builds its

own copy of data structure that has a relatively trivial demand on both time and memory. As long as the workload associated with each node can be predetermined, the overall work load can be evenly distributed to each processor resulting in a scalable algorithm. By extracting the number of sources and number of observers at each node of the data structure, it is possible to find the exact load (modulo integer work at each processor) of near field MoM matrices set up for each processor: $N_{work-load} = \sum_{i=1}^h \frac{m_i \times n_i}{np}$

where h is the total number of nodes, m_i is the number of observer, n_i is the number of source and np is the number of processors. Based on the predetermined work load, each processor holds similar amount of work $N_{work-load}$. MoM matrices representing near field interactions are constructed after the work load distribution. In a manner similar to near field interactions, merged interaction lists [7] representing far field interaction across different levels can also be grouped into a link list.

It is not possible to determine *a priori* the exact overall work load of far field interaction since the Q and R matrices have to be created before the exact rank of each node can be known. However, as discussed in [7], the oct-tree decomposition structure provides a predetermined rank-map (based on a merged interaction list- MIL) for far field interaction sub-matrices across all the levels. A close estimate on each node’s work load can be obtained by utilizing the predetermined rank-map. Work load of far field Q and R matrices set up for each processor is as follows:

$F_{work-load} = \frac{\sum_{i=1}^k (m_i + n_i) \times r_i}{np}$ where k is the total number of nodes, m_i is the number of observer, n_i is the number of source, r_i is predetermined rank of certain type of merged interaction list and np is the number of processors.

Given the fact that the work load of near and far field matrices set up on each processor is proportional to that of matrix-vector multiplication (matvecs), load balancing scheme discussed above also applies to that of matvecs. After the near and far field matrix set up step, each processor conducts its local matvec. After

each matvec, an MPI_Reduce operation is carried out to obtain the global matvec that collects vector data from each processor. As will be shown in the simulation section, this last operation does create an existing bottleneck in the proposed method working on a simple linear cluster. The effect is mitigated by the fact that this bottleneck appears at a higher number of processors as the problem size increases. As shown by Fig. 2, after near and far field matrices set up, each processor conducts its local matrix-vector multiplication, where \mathbf{V} is the unknown vector. After matrix-vector multiplication, an MPI_Reduce operation is carried out to obtain the global matrix-vector product $\mathbf{B}_{\text{final}}$, that collects vector data from each processor.

The proposed scheme uses the advantage of the MIL in terms of pre-determining ranks approximately. This in turn enables automated load balancing, and is sharp contrast to adaptive tree schemes and binary tree schemes that optimize ranks through iterative merges and splits. The underlying load balancing and processor communication issue becomes nearly intractable in that case. Multilevel fast multipole schemes rely on multilevel tree structures as does the QR scheme parallelized here. However, the FMM needs communication at the same level (far field interactions) and also *across* levels (grouping of incoming and outgoing plane waves). This significantly complicates the parallelization issues, compared to the presented method where the multilevel tree nature may be considered to have been “unrolled” through the MIL into a sequence of deterministic, independent matrix generation and product steps.

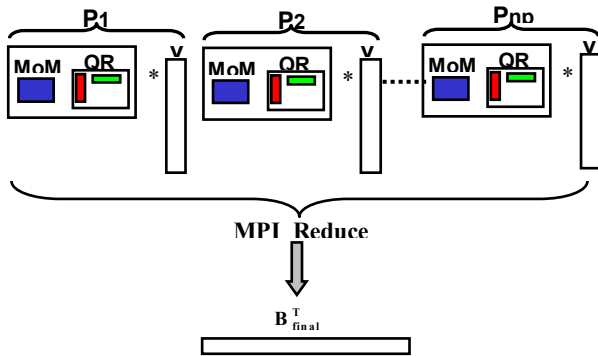


Figure 2. Matrix-vector multiplication

4. RESULTS

For the purpose of validation and efficiency testing, capacitance extraction of two test structures is carried out as a first step using the proposed method.

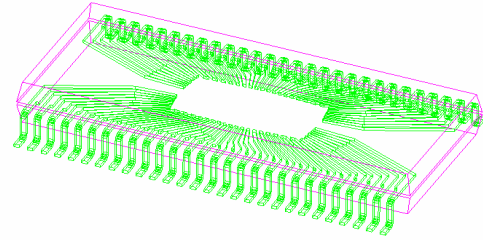


Figure 3. Packaging structure

The first test structure is a packaging structure that includes two dielectric layers and a 56-packaging-lead structure sandwiched in between the two dielectric layers. Both dielectric layers have relative permittivity of 11.9. The second test structure replicates ten copies of the 56-packaging-lead structure contained in the first test structure. The periodicity is not required or exploited in the proposed technique in any way. To test the efficiency of the load balancing schemes applied in the parallel algorithm, capacitance extraction of the first test structure was run on 1 to 16 processors. The number of unknowns on the first test structure is 7,107. Capacitance extraction of the second test structure was run on 2 to 16 processors. The number of unknowns on the second test structure is 563,216. The test performance of the proposed method and assumed ideal performance were plotted in the same figure.

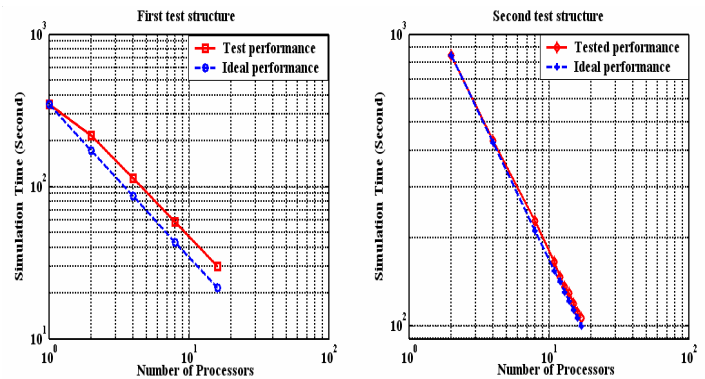


Figure 4: Simulation time versus Number of Processors

As shown by Fig. 4, the proposed method has linear scalability in both test cases owing to the balanced

workload distribution. As will be shown in Table 1, this linear scaling does have one hidden bottleneck in the matvec cost. Next, the second test structure was further discretized to 1,062,560 unknowns and run on 16 processors and it took 256 seconds for the simulation with one right hand side.

A *full-wave simulation* using the fast solution to the electric field integral equation parallelized as described was run on coupled spiral inductors. A problem with approximately 50,000 unknowns required approximately 220 sec on 18 processors for the DC to 5 GHz range. The next set of examples is related to an open IBM benchmark presented at a recent conference session (IEEE EPEP 2006) [7]. The entire benchmark consisted of seven multiple layers and twenty traces that were modeling with approximately 125,000 triangles (initial mesh) and 950,000 triangles (final adaptively refined mesh). The timing result as a function of number of processors (Dell Poweredge cluster, Gigabit Ethernet, 2 GBytes per processor, 2.4 GHz processors) is shown in Table 1. As seen there is a saturation in solution time with number of processors *for a fixed problem size*. This bottleneck is related to the relative cost of `mpi_reduce` which grows as the product of problem size and number of processors. As the problem size grows, the relative effect of this bottleneck is seen at larger problem sizes and hence this is an issue only if a large number of processors are used for small problem sizes.

5. CONCLUSIONS

This paper presents a parallel multilevel low-rank decomposition algorithm for parasitic extraction of electrical large structures applicable to multicore machines and clusters. Linear scalability is achieved through load balancing schemes on near and far field matrices construction. It is expected that the proposed scheme will enable large-scale distributed simulation due to the fact that both fast multilevel simulation technology and parallel processing are simultaneously exploited. A minor bottleneck in cost scaling when using an excessive number of processors relative to the problem size being solved is discussed.

We would like to acknowledge research funding from Intel Corporation and the National Science Foundation, and an equipment SUR grant from IBM, and we would like to IBM for benchmark information

with particular thanks to collaboration with Dr. Albert Ruehli.

6. REFERENCES

- [1] R. Coifman, V. Rokhlin and S. Wandzura, "The fast multipole method for the wave equation: a pedestrian prescription", IEEE Trans. Antennas Propagat. Mag., vol. 35, pp. 7-12, June 1993.
- [2] L.J. Jiang and W.C. Chew, "Modified fast inhomogeneous plane wave algorithm from low frequency to microwave frequency," IEEE Antennas and Propag. Soc. Int. Symp., vol. 2, pp. 22-27, June 2003.
- [3] E. Bleszynski, M. Bleszynski and T. Jaroszewicz, "AIM: Adaptive integral Method for Solving large-scale electromagnetic scattering and radiation problems", Radio Science, vol. 31, pp. 1225-1251, Sept-Oct 1996.
- [4] Z. Zhu, B. Song and J. White, "Algorithms in FastImp: a fast and wide-band impedance extraction program for complicated 3-D geometries", IEEE Trans. CAD of Integrated Circuits and Systems, vol. 24, pp. 981 - 998, July 2005.
- [5] S.Kapur and D.Long, "IES³: a fast integral equation solver for efficient 3-dimensional extraction", International conference on Computer Aided Design, pp. 448-455, Nov. 1997.
- [6] D.Gope and V.Jandhyala, "Oct-Tree based multilevel low-rank decomposition algorithm for rapid 3D parasitic extraction", IEEE Trans. CAD of Integrated Circuits and Systems, vol-23, pp. 1575 - 1580, Nov. 2004.
- [7] V. Jandhyala, C. Yang, S. Chakraborty, I. Chowdhury, J. Pingenot, and D. Williams, "A framework and simulator for parallel fast integral equation simulation of microelectronic structures," presented at special session on parallel electromagnetic simulation (A. Deutsch and G. Katopis, organizers), IEEE EPEP Conference, Scottsdale, AZ, Oct 2006.

Table 1: Setup time and total time for full parasitic extraction for the IBM 7-metal layer benchmark

# processors	Setup time (in sec)	Total time (in sec)
2	195.35	236
5	80.05	140
9	45.77	107.66
11	37.51	98.03
13	31.44	103.9
15	27.7	95.79
16	25.9	95.82