Parallel MoM Using Higher Order Basis Function and PLAPACK In-Core and Out-of-Core Solvers for Challenging EM Simulations

Yu Zhang [(1)] , Tapan K. Sarkar* [(1)], Robert A. van de Geijn [(2)], M. Taylor[(1)]
(1) EECS , Syracuse University, Syracuse, NY, 13244-1240, USA.
(2) CS, The University of Texas at Austin, TX-78712, USA
E-mail: tksarkar@syr.edu

Abstract: Currently, the problem size that can be solved by the Method of Moments(MoM) is limited by the amount of RAM installed on the computer. To reduce number of unknowns for MoM, a higher order polynomial function is introduced in this paper as the basis function. To further extend the capability of the MoM code, this paper also describes a newly developed out-of-core solver based on PLAPACK which breaks the memory constraint while maintaining the efficiency of in-core implementation. Results on a representative computer platform for a challenge electromagnetic problem show that the out-of-core solver introduced is an efficient way to deal with a large matrix equation. The implementation of these advancements creates a powerful tool for efficient computational electromagnetic solution of large and complex real world problems.

KEYWORDS: PLAPACK, Parallel Computation, in-core, out-of-core

1 Introduction

The Method of Moments (MoM) is a numerically accurate method for simulation of electromagnetic fields for radiation and scattering applications. However, MoM analysis is typically limited to electrically small and moderately large electromagnetic structures because its inherent costs in terms of memory and CPU time increase rapidly with the electrical size of the problem.

Advancements in computer technology have led to a great increase in the capability of computational electromagnetics (CEM). To solve large, complicated problems using in-core quickly, codes must be parallelized to run on a cluster. To this end, many efforts have been made to implement parallel computation of MoM. MoM using RWG basis functions, where the surface of an object is meshed into triangular patches, has been successfully parallelized using ScaLAPACK [1] and achieves theoretical speedup in matrix filling [2]. Another parallel version of MoM using RWG basis functions [3] is available using

978-1-4244-2042-1/08/$25.00 ©2008 IEEE

PLAPACK [4]. A parallel version of NEC [5] is available where the surface of the object is approximated by wire grids. However, these widely used basis functions are sub-domain basis functions which typically require the geometrical division of the surface to be at least 1/10 wavelength. Subdomain basis functions greatly limit the application of MoM to electrical large problems. In this research, polynomials of different orders are used as a kind of larger subdomain basis function. Compared with RWG or other subdomain basis functions, this choice of higher order polynomial basis functions reduces the number of unknowns greatly. Polynomial expansions result in highly acceptable current distributions over large surfaces with only about ten unknowns per wavelength squared.

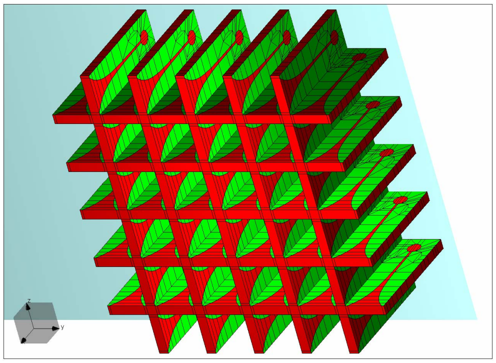
On the other side, although current computers have unprecedented memory capacity, an in-core solver is limited by the amount of RAM one can afford to purchase. Therefore, it is natural to introduce an out-of-core solver to tackle large dense linear systems generated from the method of moments (MoM). For large scale problems, prototype out-of-core implementations of some linear system solvers are provided in ScaLAPACK [6]. This library uses ScaLAPACK routines for in-core computation, and also provides an I/O layer that manages matrix input-output. As an alternative to ScaLAPACK, the Parallel Linear Algebra Package (PLAPACK [4]), containing a number of parallel linear algebra solvers, was brought to our attention as a prototype of a more flexible alternative to ScaLAPACK. PLAPACK is MPI based and designed to provide a user friendly infrastructure for building parallel dense linear algebra libraries. PLAPACK provides the following three unique features:

1) A matrix distribution that is a step towards one that is driven by the natural distribution of an application (Physically Based Matrix Distribution),
2) An application interface for filling and querying matrices and vectors, which uses one-sided communication requiring only a standard compliant MPI implementation,
3) A programming interface that allows the code to be written in a way that closely resembles the way algorithms are naturally explained, using object based (MPI-like) programming.

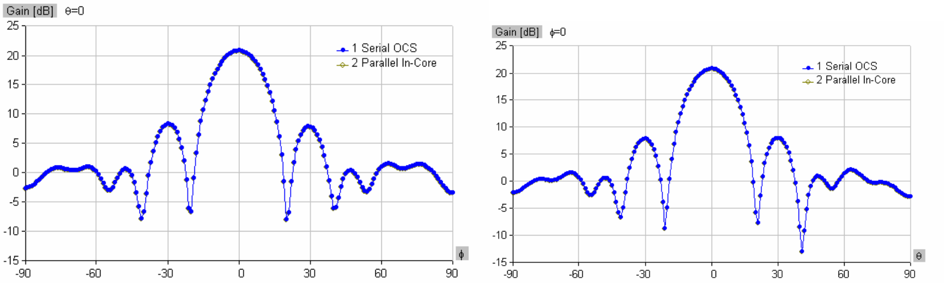
With these advantages of PLAPACK solver, we pursue the research on parallel computation for MoM with high order basis functions in this paper. Parallel matrix filling schemes are designed for a PLAPACK in-core and PLAPACK out-of-core solver respectively. Parallel codes are developed and executed on a typical computer platform. Numerical results from these platforms are presented to show the efficiency and scalability of the parallel codes.

2 Numerical Results

A 60-element Vivaldi array [7], as shown in Figure 1, is simulated is simulated on Dell PowerEdge 1855 Cluster at CEMLAB, Syracuse University [2].



(a) The structure of a 60-element dual-polarized Vivaldi array.



(b) Azimuth pattern (c) Elevation pattern

Figure 1. the 60-element Vivaldi array and its Radiation pattern

Table 1 Simulation for two Large Vivaldi Array

Vivaldi	CASE 1-1 (60-element array)	CASE 1-2 (60-element array)	CASE 1-3 (60-element array)
Parallel code	In-core	Out-of-core	Out-of-core
Frequency	5.0GHz	5.0GHz	5.0GHz
Unknowns	N_1=49,820	N_1=49,820	N_1=49,820
Memories	2.2GB/Procs	2.4G/Procs	1.7GB/Proc
Hard disk	39.7 GB	39.7 GB	36 GB
Matrix Filling	51	51	73
Matrix Solving	76	77	91
Time (Min)	$T_1=127$	$T_2=128$	$T_3=164$

The total time using 9 Nodes (with the process grid as 6×3) is listed in Table 1. The following analysis of performance is focused on the solver rather than matrix filling. Case 1-1 uses PLAPACK parallel in-core solver, Case 1-2 to Case 1-4 use parallel out-of-core solver. In Case 1-2, the in-core RAM for one slab is 2400MB (slightly larger than 2205MB/Process which is required by in-core solver), the out-of-core problem is an in-core problem with the added expense of read from and write to hard-disk. We can find that Case 1-1 and Case 1-2 require almost the same amount of time. In Case 1-3, the RAM to be used for one slab is set to be 1700 MB, the matrix is separated into two slabs because slab size is less than 2205MB. When compared to in-core of Case 1-1 or the out-of-core of Case1-2 where slab size is big enough to fit all the data using one slab, the solving time now is about 18% slower. It is worthwhile to point it out that the out-of-core solver used here is currently under developed. The parameters of the code have not been completely optimized.

3 Conclusions

This paper presents efficient parallel in-core and parallel out-of-core MoM solver with PLAPACK library. Challenging examples have been presented on a typical platform to illustrate the goals of scalability and efficiency. With the methodology explained in this paper, large problems which are difficult to solve on a single processor, can be solved in reasonable time in parallel mode, and the more important advantage is, RAM is no longer remains a factor that limit the application of MoM.

Reference

[1] http://www.netlib.org/scalapack/

[2] Yu Zhang, T. K. Sarkar, A. De, N. Yilmazer, S. Burintramart and M. Taylor, "A Cross-Platform Parallel MoM Code with ScaLAPACK Solver," IEEE AP-S International Symposium, Honolulu, Hawaii, June 11-15, 2007.

[3] Tom Cwik, Robert van de Geijn, and Jean Patterson, "The Application of Parallel Computation to Integral Equation Models of Electromagnetic Scattering," *Journal of the Optical Society of America A* , Vol. 11, No. 4, pp. 1538-1545, April 1994.

[4] http://www.cs.utexas.edu/users/plapack/

[5] Abraham Rubinstein, Farhad Rachidi, Marcos Rubinstein and Beat Reusser, "A Parallel Implementation of NEC for the Analysis of Large Structures," *IEEE Transactions on Electromagnetic Compatibility*, Vol. 45, No. 2, May 2003.

[6] E. F. D'Azevedo and J. J. Dongarra, "The Design and Implementation of the Parallel Out-Of-Core Scalapack LU, QR and Cholesky Factorization Routines," University of Tennessee, Knoxville, Jan. 1997, http://citeseer.ist.psu.edu/article/dazevedo97design.html.

[7] Mengtao Yuan, Tapan K. Sarkar, et al, "Solution of Large Complex Problems in Computational Electromagnetics using Higher-Order Basis in MoM with Out-of-Core Solvers," *IEEE Antennas Propagat. Magazine*, Vol. 48, No. 2, April 2006.