Parallel MoM Using Higher Order Basis Functions and PLAPACK Out-of-Core Solver for a Challenging Vivaldi Array

Mary C. Taylor [1], Yu Zhang [2], Tapan K. Sarkar [3], Robert A. van de Geijn [4]

[1] Syracuse University, EECS Department, 335 Link Hall, Syracuse, NY 13244-1240, USA
Email: cannella@syr.edu, Phone: 315.443.1406, Fax: 315.443.4441

[2] Email: yzhang39@syr.edu

[3] Email: tksarkar@syr.edu, Phone: 315.443.3775

[4] The University of Texas at Austin, Department of Computer Sciences, Austin, TX 78712, USA
E-mail: rvdg@cs.utexas.edu, Phone: 512.471.9720, Fax: 512.471.8885

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, the higher order polynomial function is introduced as the basis function. To extend the capability of the MoM code, this paper offers a new development using the PLAPACK out-of-core solver which breaks the RAM memory constraint. Numerical results on a typical computer platform for a challenge electromagnetic problem show that the PLAPACK out-of-core solver introduced here is a very efficient way to deal with large matrix equations. The implementation of these advancements creates a powerful tool for efficient computational solution of large and complex real world electromagnetic problems.

1. Introduction

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

Advancements in computer technology have led to improvements in the capability of computational electromagnetics (CEM). To solve large, complicated problems in reasonable time using in-core solvers, codes must be designed in a parallel way to run on a cluster. Much recent research is focused on how to implement parallel computation of MoM. MoM using RWG basis functions, where the surface of 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 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. This results in a large number of unknowns, which would require a large amount of RAM for in-core solution. Due to the expense of RAM, 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 current distributions over large surfaces with only about ten unknowns per wavelength squared.

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 makes sense to consider an out-of-core solver to tackle large, dense linear systems generated by 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 the authors' attention as a prototype of a more flexible alternative to ScaLAPACK. PLAPACK is based on message passing interface (MPI) code and designed to provide a user friendly infrastructure for building parallel dense linear algebra libraries. PLAPACK provides the following three unique features:

1) a Physically Based Matrix Distribution (PBMD) that is a step towards one that is driven by the natural distribution of an application,
2) an application interface for filling and querying matrices and vectors, which uses one-sided communication requiring only a standard compliant MPI implementation, and
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.

These features of the PLAPACK solver provide advantages for the implementation of parallel computation using MoM with higher-order basis functions presented in this paper. In this research, a parallel matrix filling scheme is designed for a PLAPACK out-of-core solver. Parallel code is developed and run on a typical computer platform. Numerical results from this platform are presented to show the capability of this parallel code.

2. Numerical Results

The Vivaldi antenna has gained interest in recent years as a compact ultrawideband (UWB) antenna array for civilian and military applications. This composite antenna structure has gained the interest of CEM practitioners because it is a challenging structure to model and simulate accurately. The Vivaldi antenna array has become a benchmark project to test the capability and speed of computational codes and high performance computing platforms. For these reasons, the results of a large Vivaldi array model is presented in this paper.

A 112-element dual-polarized Vivaldi array is shown in Figure 1. The dimensions of a single element are 76 mm x 25.6 mm × 6 mm. The dielectric constant of the substrate is 2.2. The operating frequency of this Vivaldi array is from 1 to 5 GHz, where it behaves as a traveling wave antenna. However, we analyze the structure at the highest frequency of interest as it has the largest number of unknowns. This is a composite structure, as can be seen in Figure 1, where the blue plates are metallic and the red ones are dielectric. The array is backed by an infinitely large PEC plate.

Modeling a 112-element array requires 63,952 elements which are composed of wires, plates and junctions. The number of unknowns for this array is 92,729, which requires approximately 137.6 GB of RAM in double precision arithmetic.

This Vivaldi array is simulated on a Dell PowerEdge 1855 Cluster located in the Computational Electromagnetics (CEM) Lab at Syracuse University. The cluster has ten blades, each blade having two Intel Xeon (single core) 3.6 GHz CPUs, 800 MHz FSB, 8 GB RAM and a 146 GB, 10K RPM, SCSI hard drive. Each node has a peak performance of 2 x 7.2 GFLOPS **(G**iga **FL**oating point **O**perations **P**er **S**econd**)**. The total time taken to solve the Vivaldi on this platform using 9 nodes (18 CPUs) is 17.8 hours. Table 1 summarizes the details of this large example problem. A composite structure is particularly challenging because the matrix filling time is comparable to the matrix solving time, as can be seen in Table 1.

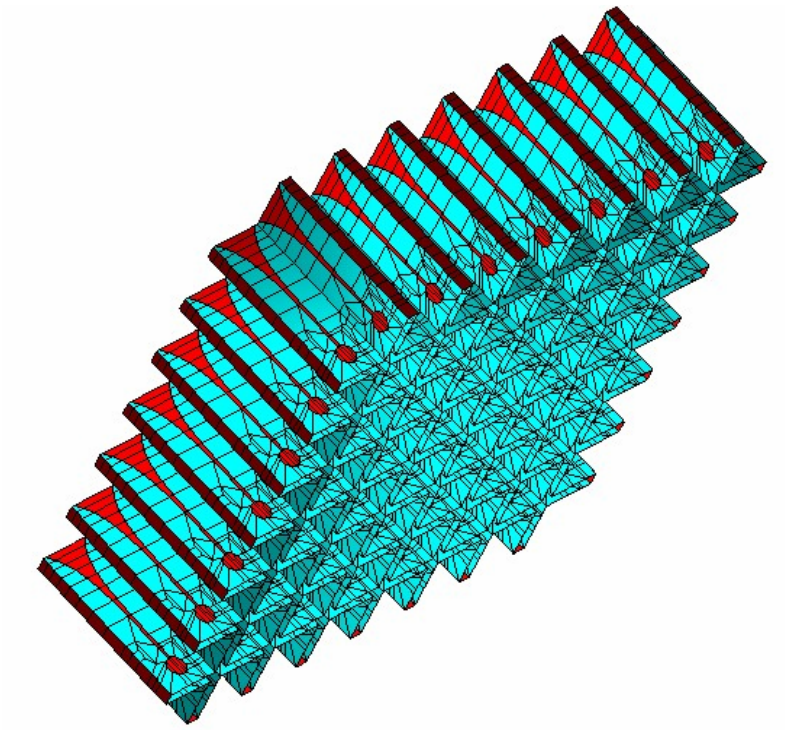
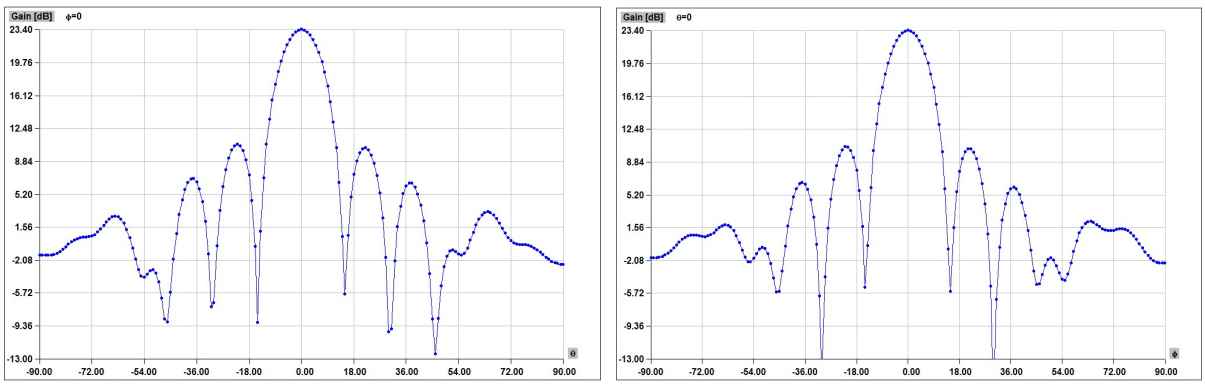


Figure 1. The 112-Element Dual-Polarized Vivaldi Array



(a) Azimuth Pattern (b) Elevation Pattern

Figure 2. Radiation Pattern of the 112-Element Vivaldi Array

Table 1. Simulation of a Large Vivaldi Array

Vivaldi 112-Element Array	
Parallel Out-of-Core Solver	
Frequency	5.0 GHz
Unknowns	N = 92,729
Memory	36.0 GB
Storage	137.6 GB
Matrix Filling	491 min.
Matrix Solving	578 min.
Total Wall Time	1069 min.

3. Conclusions

This paper presents an efficient parallel out-of-core MoM solver using the PLAPACK library. A challenging 112-element Vivaldi antenna array has been presented on a typical platform to illustrate the capability of this solver. With the methodology explained in this paper, large problems which were previously impossible to solve on a single processor, can be solved in reasonable time in a parallel mode using an out-of-core solver, which provides the advantage that RAM no longer limits the application of MoM for large real-world computational challenges.

4. References

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

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

3. T. Cwik, R. A. van de Geijn, and J. 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/~plapack/

5. A. Rubinstein, F. Rachidi, M. Rubinstein and B. 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. M. Yuan, T. K. Sarkar, B. Kolundzija, "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.