2021 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting (APS/URSI) | 978-1-7281-4670-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/APS/URSI47566.2021.9704640

Highly Parallelized Hybrid FDTD Solver for Simulating Electromagnetic Wave Propagation in Dense Urban Environments

Kazem Sabet#1, Anca I. Stefan#, Kamal Sarabandi$, Fikadu Dagefu%, Brian M. Sadler%

#EMAG Technologies, Ann Arbor, MI, USA $The University of Michigan, Ann Arbor, MI, USA %United States Army Research Laboratory, Adelphi, MD, USA
1ksabet@emagtech.com

***Abstract*—In this paper, we describe a hybrid parallel FDTD solver for simulation of large-scale urban propagation scenarios that utilizes both shared and distributed memory capabilities of modern processors. Simulation results are presented for an accurate real-world physical urban model (Downtown Los Angeles) at HF band (40 MHz).**

***Keywords*—finite difference methods, wireless propagation, urban areas, large-scale simulation, phase distortion, high performance computing.**

I INTRODUCTION

The ability to perform full-wave electromagnetic analysis of large-scale systems is of critical importance to the study of wave propagation and penetration in large dense urban environments at lower frequency bands. The study of lower VHF propagation in dense urban environments is motivated by the favorable propagation and penetration that can be exploited for autonomous systems [1]. The upper HF and lower VHF frequency bands present a unique set of computational challenges because propagation is characterized by penetration and diffraction, rather than multipath and fading, and therefore, ray tracing and other asymptotic techniques cannot provide adequate modeling accuracy. An accurate prediction of the spatial variation of signal amplitude and phase in the presence of inhomogeneous clutter and large numbers of wavelength-scale obstacles across a large area requires a full-wave simulation of the propagation scene. Recent publications on the HPC implementations of FDTD solvers have focused mostly on solving problems on larger numbers of processors at various frequency bands. Traditional large-scale problems of interest solved with FDTD include vehicle-installed antennas and RCS of large aircraft, and typically result in billion-cell computational domains [2]-[5]. Urban propagation scenarios easily surpass the other types of applications in overall problem size, and they also pose another major computational challenge due to their highly multi-scale nature. These challenges in computational electromagnetics (CEM) have not been sufficiently addressed. Domain decomposition, which overcomes the memory barrier, represents a priority in these types of problems. In a wave propagation problem, the regions surrounding the transmitters and receivers require a very high level of modeling fidelity, and therefore a very high mesh resolution in order to capture all the relevant physical effects and interactions. Additionally, increasing the mesh resolution has the desirable effect of minimizing the inherent numerical dispersion of the FDTD method. However, employing domain decomposition and other acceleration paradigms requires access to an HPC environment. In this work, we describe a hybrid HPC solver, where the term "hybrid" refers to the ability to make use of both shared and distributed memory capabilities of modern processors. We discuss the computational speedup achieved by this solver, as well as its ability to minimize numerical dispersion, by solving problems with enough resolution such that the degradation of the numerical phase velocity over large ranges is significantly reduced.

II HYBRID OPENMP AND MPI PARALLELIZATION

In general, FDTD code parallelization can be implemented in order to: (1) reduce the memory bottleneck and computational complexity without sacrificing accuracy, and (2) accelerate the field update equations and increase the overall computational speed of the code. The first parallelism is accomplished by employing the Message Passing Interface (MPI), which is the distributed-memory framework. The second parallelism is implemented by using the OpenMP compiler directives, which is the shared-memory framework. To achieve distributed parallelism, a large computational domain is broken up into a number of smaller sub-domains that to be distributed among multiple processors. The FDTD domain decomposition is done along the grid lines of the mesh. Each processor updates all the six components of the electric and magnetic fields inside the sub-domain assigned to it at each time step. In addition, the four tangential field components of the electric and magnetic fields on each of the six shared faces of each interior sub-domain must be matched with those of its neighboring sub-domains at each time. This requires data communication between various nodes. However, the greater the computational load of data communications among the nodes in the computing cluster, the lower the scalability of the HPC code. On non-shared faces of sub-domains, such as those present on corner sub-domains, the designated domain boundary conditions are enforced at each time step.

The proposed hybrid HPC algorithm is based on the process described in [5], and uses OpenMP at each iteration to speed up computation of electric and magnetic fields on individual compute nodes, along with an MPI implementation of the communication among the processors (nodes) that operate on each sub-domain. After initialization of field components and other needed structures, at each time step the following is performed: (1) exchange electric fields with neighbors using Open MPI; (2) update magnetic fields using OpenMP; (3) exchange updated magnetic field values with neighbors using Open MPI; (4) update electric fields using OpenMP; and (5) update fields at the complex perfectly matched layer boundaries using OpenMP.

III COMPUTATIONAL SPEEDUP ANALYSIS

The computational force behind the work described in this paper is an inhomogeneous Linux cluster with 8 workstations and a total of 204 available physical processors and a total of 3.2 TB available RAM. An 8-port 10Gb Ethernet managed network switch and 10 Gb network interface cards are utilized. In order to test the hybrid parallel FDTD implementation, we define computational speedup as the ratio t_0/t_N, where t_0 is the MPI-only simulation time (i.e., 0 threads, with no shared-memory implementation), and t_N is the hybrid OpenMP/MPI time (i.e., with both shared memory and distributed memory implementation). We also define scalability as the ratio $t_0/(N * t_N) *100$, where N represents the number of threads. The hybrid implementation, run with just 2 threads per core (N = 2), yielded a 15% to 37% improvement in computation time. Table 1 shows the results for three benchmark problems: an antenna radiation pattern calculation (8.60 million cells), a 500-m by 500-m dense urban area (149 million cells), and a 1-km by 1-km dense urban area (1.91 billion cells). The last problem represents a segment of Downtown Los Angeles, CA, with accurate models of all the individual buildings in the area as shown in Fig. 1. The results show that speedup and scalability also improve with increased problem size.

Table 1. Comparison of computation times of Open MPI vs. hybrid solvers for three benchmark problems.

Problem size	**No. of Proc.**	**Time to Convergence (s)**		**% Change**	**Speedup**	**Scalability**
		MPI	Hybrid			
8.60 mil	16	121	103	-14.8	1.17	58.7%
149 mil	38	933	630	-32.5	1.48	74.0%
1.91 bil	38	29894	18595	-37.8	1.61	80.5%

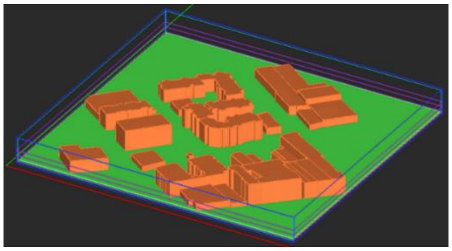

Fig. 1. A propagation model representing a surface area of 0.075 km^2 in Downtown Los Angeles. At mesh resolution 0.05 m × 0.05 m × 0.043 m, the domain size is 5,691 cells × 5,291 cells × 624 cells (approx. 18.8 billion cells).

IV A VERY LARGE-SCALE URBAN PROPAGATION PROBLEM

In this section, we present simulation results for a very large propagation problem using the hybrid MPI-OpenMP HPC FDTD solver on the 204-core, 3.2-TB HPC Linux cluster. The problem consists of the same dense urban area representing part of Downtown Los Angeles shown in Fig. 1, with hollow building models assumed to have finite wall thicknesses. The dielectric ground was modeled with $\varepsilon_r = 5$ and $\sigma = 0.005$ S/m and buildings with $\varepsilon_r = 4.0$ and $\sigma = 0.001$ S/m. We note that this problem required over 90% of the available RAM on the cluster. At a spatial resolution of 0.05 m × 0.05 m × 0.043 m, this model yielded an impressive **mesh size of 18.8 billion cells**. Fig. 2 shows the total electric field amplitude and E_z phase distribution simulated at 40 MHz.

V DISCUSSION

We described a state-of-the-art HPC FDTD solver for simulation of large-scale dense urban propagation scenes in the HF-VHF bands. The solver employs effective hybridization of MPI and OpenMP parallelization paradigms to develop a versatile domain decomposition capability that is fast and takes advantage of the multi-core architecture of modern processors. The solver is optimized for both distributed and shared memory architectures, and the user has the option to strike a balance between the number of compute nodes used for MPI domain decomposition versus those used for the OpenMP code acceleration. We believe that this choice depends on the availability of processors optimized for hyperthreading, and the impact of specific problems is currently being investigated. We presented simulation results for an urban propagation scenario with more than 18.8 billion cells on a custom-built HPC Linux cluster featuring 204 CPU nodes and 3.2 TB RAM.

REFERENCES

1 J. Choi, F. T. Dagefu, B. M. Sadler, K. Sarabandi, “Low-power low-VHF ad-hoc networking in complex environments,” IEEE Access, Vol. 5, pp. 24120–24127, 14 November 2017.

2 J.T. MacGillivray, “Trillion cell CAD-based Cartesian mesh generator for the finite-difference time-domain method on a single-processor 4-GB workstation,” IEEE Transactions on Antennas and Propagation, vol. 56, no. 8, pp. 2187-2190, Aug. 2008.

3 Y. Liu, Z. Liang, and Z. Q. Yang, “A novel FDTD approach featuring two-level parallelization in PC cluster,” Progress in Electromagnetics Research, PIERS 80, pp. 393–408, 2008.

4 A. P. Smirnov, et al., “Performance Analysis of Parallel FDTD Algorithm on IBM BlueGene Supercomputer, Progress in Electromagnetics Research, PIERS 86, pp. 253–256, 2014.

5 M.S. Su, et al., “A novel FDTD application featuring OpenMP-MPI hybrid parallelization” Proceedings of the 2004 International Conference on Parallel Processing (ICPP’04).

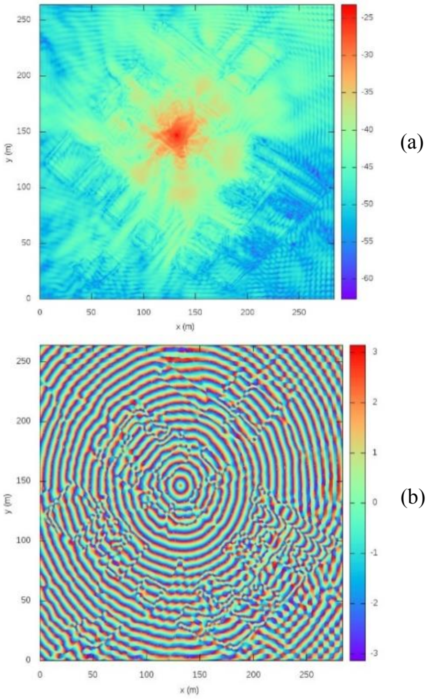

Fig. 2. Field distribution plot for the 18.8-billion-cell Downtown LA scene: (a) total electric field magnitude, and (b) Phase of electric field Ez component.