

A 2-D Parallel Finite Difference Time Domain Solver Using MPI

Tucker DiNapoli

May 21, 2013

1 Introduction

This project aimed to develop a parallel implementation of a two dimensional finite difference time domain(FDTD) solver implemented in C using mpi. The primary goal of the project was the implementation of parallization using MPI, so much of the base FDTD code was taken from other open source FDTD solvers. The project thus consisted of two major parts, adapting the existing code(much of which was written in C++) and parallizing the code with MPI.

2 Scientific Background

The finite difference time domain technique is a method used to model the behavior of electric and magnetic fields. As the name suggests FDTD uses finite differences (in space and time) to determine the time evolution of the system. FDTD uses a rectangular grid composed of some number of cubes each with an area of dx^3 (or squares of area dx^2 in a 2-D problem) to represent the spacial extent of the problem. The time step must be chosen such that $\Delta t_{max} \geq \sqrt{3\Delta x^2}$

3 Numerical Analysis

4 Code

4.1 Existing Implementations

4.2 Code Adaptation

4.3 Parallization

5 Data

6 Conclusion