Journal of Computational Design and Engineering 4 (2017) 249–255

Contents lists available at ScienceDirect

Journal of Computational Design and Engineering

journal homepage: www.elsevier.com/locate/jcde

Optimization of patch antennas via multithreaded simulated annealing based design exploration

James E. Richie, Cristinel Ababei *

Dept. of Electrical and Computer Engineering, Marquette University, 1515 W. Wisconsin Ave., Milwaukee, WI 53233, USA

ARTICLE INFO

Article history:
Received 2 May 2017
Received in revised form 8 June 2017
Accepted 14 June 2017
Available online 12 July 2017

Keywords:
Microstrip patch antennas
FDTD
Design space exploration
Simulated annealing
Multithreaded parallelization

ABSTRACT

In this paper, we present a new software framework for the optimization of the design of microstrip patch antennas. The proposed simulation and optimization framework implements a simulated annealing algorithm to perform design space exploration in order to identify the optimal patch antenna design. During each iteration of the optimization loop, we employ the popular MEEP simulation tool to evaluate explored design solutions. To speed up the design space exploration, the software framework is developed to run multiple MEEP simulations concurrently. This is achieved using multithreading to implement a manager-workers execution strategy. The number of worker threads is the same as the number of cores of the computer that is utilized. Thus, the computational runtime of the proposed software framework enables effective design space exploration. Simulations demonstrate the effectiveness of the proposed software framework.

1. Introduction

Microstrip or patch antennas are becoming increasingly popular because they can be printed directly onto circuit boards. Patch antennas are low cost, have a low profile and can be easily fabricated. They are becoming very widespread within the mobile phone market. The design process is affected by many parameters that control the properties of such antennas. To identify the parameters that offer the best performance requires the exploration of the design solution space defined by such parameters. Such exploration implies multiple numerical simulations, which can take long computational runtimes.

In this paper, we present a software framework that implements an automated design space exploration (DSE) for patch antennas. The exploration is done with a multithreaded simulated annealing (SA) optimization algorithm. Each explored design solution is evaluated with the MEEP simulator. To this end, the main contributions of this paper include: (1) The SA based tool that can seek and identify the optimal patch antenna design and (2) The SA algorithm is implemented with a multithread approach, which provides a speed-up of the execution time by a factor of 7.56x when executed on an 8-core processor compared to the execution on only one thread on a single core.

The remainder of this paper is organized as follows. In the next section, we briefly review related literature. Then, we present background information and formulate the problem of optimization for patch antennas. In Section 4, we present the proposed software framework, which consists of a parallel via multithreading implementation of a simulation annealing optimization algorithm to solve the patch antenna design problem. Section 5 reports simulation results obtained on a machine running on a processor with eight cores, followed by a discussion in Section 6. Finally, we conclude and summarize our contributions in Section 7.

2. Related work

With the advent of wireless communication systems, there has been significant work done on the design of antennas. Particularly, research efforts were focused on antennas used in wireless local area network (WLAN) and Bluetooth applications. Instances of such efforts include Gondarenko and Lipson (2008), Hansen, Zheng, Perederey, and Hesselink (2011), Jayasinghe and Uduwawala (2015) and Meng and Sharma (2016) and the references therein. For example, the recent study in Jayasinghe and Uduwawala (2015) presented the design of a compact planar inverted F antenna (PIFA) for 2.4 GHz and 5 GHz bands. In order to optimize the geometry (the shorting pin position and the feed position of their patch antenna), the authors used a genetic algorithm (GA) optimization approach that employed simulations with the Ansys' high frequency structure simulator (HFSS). However, they did not report computational runtimes of their optimization

Peer review under responsibility of Society for Computational Design and Engineering.

* Corresponding author.
E-mail addresses: james.richie@marquette.edu (J.E. Richie), cristinel.ababei@marquette.edu (C. Ababei).

http://dx.doi.org/10.1016/j.jcde.2017.06.004
2288-4300/

approach, which is not publicly available. In addition, HFSS is a commercial tool, which is not free. As another recent example, the study in Meng and Sharma (2016) reported the design of a single feed dual-band miniaturized microstrip patch antenna for WLAN communications. They also use Ansys HFSS and several manual design optimization techniques including the creation of selected slots to the E-shaped inner patch and the adoption of a non-homocentric design style.

In contrast with these works, the core simulator that we use to evaluate each design solution point during separate iterations of the SA algorithm is MEEP (MEEP, 2017; Oskooi et al., 2010). MEEP is a popular open-source free implementation of the finite-difference time-domain (FDTD) method for simulating electromagnetics problems. It can be used to conduct a variety of simulations but for given specific design problems only; it does not include optimization algorithms to seek optimal solutions. Noteworthy is that its implementation is very efficient due to various code optimization techniques. Thus, it serves well the purpose of being an efficient simulator that we can call multiple times to evaluate patch antenna designs.

However, if simulations are required to be done for three dimensional structures, such as in the case of numerical electromagnetic calculations in nanophotonics, the computational runtimes of such FDTD simulators can increase significantly due to large number of variable parameters. Moreover, when applying optimization schemes that require a full-field solution at each iteration of the optimization loop, such as genetic algorithms (Gondarenko & Lipson, 2008; Jayasinghe & Uduwawala, 2015), adjoint optimization methods (Hansen et al., 2011), or simulated annealing in the case of this paper, the ability to efficiently find a solution becomes limited by the computational speed of the full-field simulator. To address such computational runtime issues, one can pursue one of the following two approaches. The first approach is to focus on the FDTD simulator itself in order to speed it up via some parallelization technique. In this category, the study in Wahl, Ly-Gagnon, Debaes, Miller, and Thienpont (2013) uses CUDA programming to speed-up a 3D-FDTD solver by running it on computers equipped with Graphical Processing Units (GPUs). The authors benchmark their GPU based implementation against MEEP and reported significant speed-up for computing the absorption efficiency of a metallic nanosphere.

Note that these FDTD simulators are designed to be used for just a single simulation of a design or structure of interest at a time. They do not conduct any optimization in the sense of seeking to identify the best combination of design parameters that would provide the desired design characteristics. Such design parameters are assumed to be known and directly specified as input to these simulators. Therefore, the second approach to address computational runtime issues, applicable in the case of iterative optimizations is to parallelize the optimization algorithm itself. The simulation and optimization framework proposed in this paper falls in this category. We employ MEEP as a point-tool simulator to develop an optimization approach whose objective is to find the best solution for a patch antenna design. To make it computationally efficient, we use multithreading as a parallelization technique, which is implemented using a manager-workers strategy that allows us to run multiple MEEP simulator instances concurrently, thereby speeding up the design solution space exploration. We would like to emphasize that, aside from finding the optimal patch antenna design via the simulated annealing optimization, actually, equally important contributions of our work include: our approach is efficient because it uses the multithreading based speed-up technique, the implementation is versatile in that it can easily be changed to replaced the open-source MEEP simulator with other simulators such as Ansys's HFSS used by previous works, it will be made publicly available as it is constructed with only free tools.

3. Background on patch antennas

3.1. The MEEP model for patch antennas

The MEEP C++ source code is freely available under the GNU GPL license. Documentation is available on the MEEP Wiki pages (MEEP, 2017), including tutorials and reference material. Several examples are also part of the software package. The software can be executed using a script file in the Scheme language, or by writing C++ code that performs the simulation. In this paper, we use the C++ interface.

In MEEP, the fundamental geometric size is the *block*. One typically chooses a distance a that is one block. All quantities are then based off the block size. Each block is broken into cells using the resolution parameter. MEEP also uses *dimensionless* units, $\epsilon_o = \mu_o = c = 1$, where c is the speed of light. All distances are converted by multiplying the number of blocks by a, and frequencies are converted using $f_{meep} = fa/c$, where f_{meep} is the MEEP frequency and f is the frequency in Hz.

A typical microstrip patch antenna consists of a rectangular patch of metal over a dielectric substrate backed by a ground plane. The patch has length L and width W, as shown in Fig. 1. The dielectric height is h, as shown in Fig. 2. Roughly speaking, L is near one half-wavelength. The radiation can be modeled as the fringing of the electric field along thin slots of length W at the two edges separated by the distance L. The radiation pattern of the antenna is typically a broad beam with maximum near broadside. The pattern resembles an array of two elements (the slots) separated by L and fed in phase. The bandwidth of a rectangular patch antenna is typically quite small. Design equations for patch antennas are available and can be found in Balanis (1997) and Stutzman and Thiele (1998).

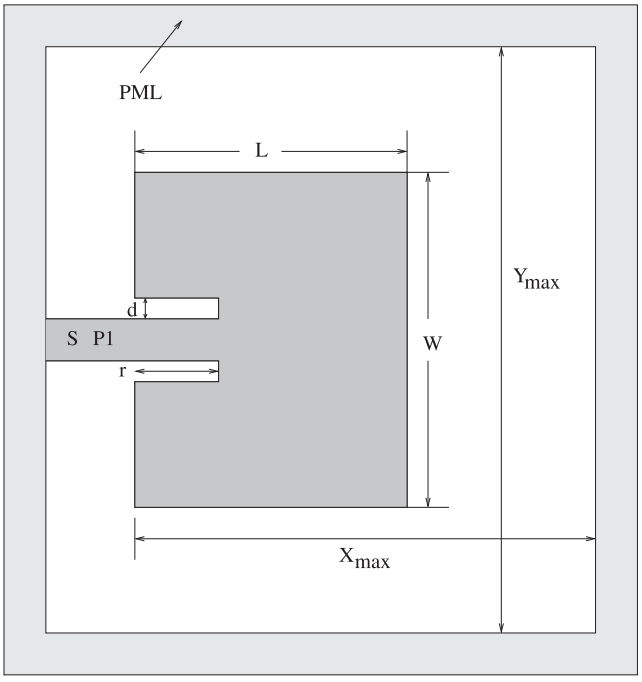

Fig. 1. Geometry and dimensions for the MEEP model of the $L \times W$ patch antenna. The source (S) is dTL from the PML edge; port 1 (P1) is dS from the source; and the patch antenna is dP1 from port 1.