# IMPLEMENTATION OF THE FINITE-DIFFERENCE TIME-DOMAIN METHOD USING GRAPHICS PROCESSING UNITS

Approved by:

Dr. Marc Christensen

Dr. Nathan Huntoon

Professor Ira Greenberg

# IMPLEMENTATION OF THE FINITE-DIFFERENCE TIME-DOMAIN METHOD USING GRAPHICS PROCESSING UNITS

A Thesis Presented to the Graduate Faculty of the

Lyle School of Engineering

Southern Methodist University

in

Partial Fulfillment of the Requirements

for the degree of

Master of Electrical Engineering

with a

Major in Electrical Engineering

by

S. David Lively

(B.S.E.E, Southern Methodist University, 2008)

August 1, 2016

## ACKNOWLEDGMENTS

I thank my committee for their patience, insight and unfailing encouragement. Without them, this thesis would remain vaporware. Never give up, never surrender!

Lively , S. David B.S.E.E, Southern Methodist University, 2008

Implementation of the Finite-Difference Time-Domain Method Using Graphics Processing Units

Advisor: Professor Marc Christensen
Master of Electrical Engineering degree conferred August 1, 2016
Thesis completed August 1, 2016

Traditionally, optical circuit design is tested and validated using software which implement numerical modeling techniques such as Beam Propagation, Finite Element Analysis and FDTD.

While effective and accurate, FDTD simulations require significant computational power. Existing installations may distribute the computational requirements across large clusters of high-powered servers. This approach entails significant expense in terms of hardware, staffing and software support which may be prohibitive for some research facilities and private-sector engineering firms.

Application of modern programmable GPGPUs to problems in scientific visualization and computation has facilitated dramatically accelerated development cycles for a variety of industry segments including large dataset visualization, microprocessor design, aerospace and electromagnetic wave propagation in the context of optical circuit design.

The FDTD algorithm as envisioned by its creators maps well to the massively-multithreaded data-parallel nature of GPUs. This thesis explores a GPU FDTD implementation and details performance gains, limitations of the GPU approach, optimization techniques and potential future enhancements that may provide even greater benefits from this underutilized and often-overlooked tool.

## TABLE OF CONTENTS

CHAPTER

*To Audrey, Wyatt, Walter and Gwendolyn*

Chapter 1

# INTRODUCTION

FDTD is a proven algorithm, first published in (...) by (yee, et al). It is the underlying mechanism used by many commercial optics simulation packages, as well as open source software such as MIT's Meep.

Given the computationally-intensive nature of FDTD, organizations requiring simulation of large domains or complex circuits must provide significant resources. These may take the form of leased server time or utilization of an on-site high-performance cluster, amongst other options.

In this thesis, we explore an implementation of the Finite-Difference, Time-Domain (FDTD) method of electromagnetic waves simulation as implemented on graphics processing units (GPUs). Initially designed to perform image generation tasks such as those required by games, cinema and related fields, modern versions are well-suited for general computation work. GPUs are now enjoying wide adoption in fields such as machine learning and artificial intelligence, medical research, signals analysis and other areas which require rapid analysis of large datasets.

Even modern consumer-grade GPUs offer thousands or tens of thousands of processing units, while high-end CPUs offer 4-8 cores. While the two are not interchangeable (see: chapter on Device Architecture), some algorithms, such as FDTD, require little or data interdependence, no branching logic (a severe performance impediment on GPUs) and consist of short cycles of simple operations. The power of the GPU lies in performing these simple operations at large scale, with thousands of threads running in parallel.

The following sections detail FDTD. Later sections describe a CPU-based implementation (MIT's Meep simulator), and our GPU-based GoLightly simulator. We verify the GPU solution numerically, and compare performance between CPU- and GPU-based implementations. Finally, we consider future applications and enhancements.

## 1.1. FDTD Overview

### 1.1.1. Wave equation

### 1.1.2. Yee Cell

Chapter 2

# DEVICE ARCHITECTURE

## 2.1. CPU

independent cores, separate cache, dedicated ALU and registers

## 2.2. GPU

SIMD - single ALU for multiple register sets why FDTD maps well to GPUs

Chapter 3

# MEEP

## 3.1. Background

## 3.2. Modeling approach

## 3.3. Performance

## 3.4. Usability

Chapter 4

# GOLIGHTLY

## 4.1. goals

## 4.2. system architecture

### 4.2.1. Host

### 4.2.2. GPU

## 4.3. Modeling approach

## 4.4. Implementation

## 4.5. Testing methodology

### 4.5.1. Test Model

### 4.5.2. Analytical Result

### 4.5.3. Numerical Result

#### 4.5.4. Comparison

### 4.6. Additional Examples

#### 4.6.1. Coupler

#### 4.6.2. Splitter

# Chapter 5

# Conclusions

## 5.1. Meep performance

## 5.2. GoLightly performancec

## 5.3. Meep vs GoLightly

## 5.4. Results

## 5.5. Limitations

Chapter 6

# FUTURE WORK

future work...

## REFERENCES


[1] *On the singularity structure of fully developed turbulence* (Amsterdam, 1983), North-Holland.

[2] Anselmet, F., Gagne, Y., Hopfinger, E. J., and Antonia, R. A. High-order velocity structure functions in turbulent shear flows. *Journal of Fluid Mechanics 140*, -1 (1984), 63–89.

[3] Batchelor, G. *The theory of homogeneous turbulence*. Cambridge University Press, Cambridge, 1971.

[4] Champagne, F. H. The fine-scale structure of the turbulent velocity field. *Journal of Fluid Mechanics 86*, 01 (1978), 67–108.

[5] Ditlevsen, P. Cascades of energy and helicity in the goy shell model of turbulence. *Phys. Fluids 9*, 5 (1997), 1482–1484.

[6] Ditlevsen, P., and Mogensen, I. Cascades and statistical equilibrium in shell models of turbulence. *Phys. Rev. E 53*, 5 (1996), 4785–4793.

[7] Frisch, U. *Turbulence*. Cambridge University Press, Cambridge, 1999.

[8] Gibson, C. Turbulence in the ocean, atmosphere, galaxy, and universe. *Appl. Mech. Rev. 49*, 5 (1996), 299–315.

[9] Grant, H. L., Stewart, R. W., and Moilliet, A. Turbulence spectra from a tidal channel. *Journal of Fluid Mechanics Digital Archive 12*, 02 (1962), 241–268.

[10] Hinze, J. *Turbulence; an introduction to its mechanism and theory*. McGraw-Hill, New York, 1959.

[11] Kolmogorov, A. The local structure of turbulence in incompressible viscous fluid for very large reynolds numbers. *Reprinted Proc. R. Soc. Lond. (1991) 434*, 1890 (1941), 9–13.

[12] Lundgren, T. Inertial range scaling law. *Journal of Turbulence 6*, 22 (2005).

[13] McComb, W. *The Physics of Fluid Turbulence*. Clarendon Press, Oxford, 1991.

[14] Melander, M. V. Analysis of a symmetry leading to an inertial range similarity theory for isotropic turbulence, 2007. http://www.citebase.org/abstract?id=oai:arXiv.org:physics/0702073.

[15] Melander, M. V., and Fabijonas, B. R. Self-similar enstrophy divergence in a shell model of isotropic turbulence. *J. Fluid Mech. 463* (2002), 241–258.

[16] Melander, M. V., and Fabijonas, B. R. Intermittency via self-similarity – an analytic example, 2005. http://www.citebase.org/abstract?id=oai:arXiv.org:physics/0512198.

[17] Moser, R. D., Kim, J., and Mansour, N. N. Direct numerical simulation of turbulent channel flow up to $\mathrm{re}_\tau = 590$. *Physics of Fluids 11*, 4 (1999), 943–945.

[18] Noullez, A., Wallace, G., Lempert, W., Miles, R. B., and Frisch, U. Transverse velocity increments in turbulent flow using the relief technique. *Journal of Fluid Mechanics 339*, -1 (1997), 287–307.

[19] Pisarenko, D., Biferale, L., Courvoisier, D., Frisch, U., and Vergassola, M. Further results on multifractality in shell models. *Phys. Fluids A 5* (1993), 2533–2537.

[20] Rosenblatt, M., and Atta, C. V., Eds. *Statistical self-similarity and inertial subrange turbulence* (Berlin, 1993), vol. 12, Springer.

[21] She, Z., and Leveque, E. Universal scaling laws in fully developed turbulence. *Phys. Rev. Lett. 72*, 3 (1994), 336–339.

[22] Sneddon, I. *Fourier Transforms*, 1st ed. McGraw Hill, New York, 1951.

[23] Taylor, G. Statistical theory of turbulence. *Proc. Roy. Soc. A 151* (1935), 421–478.

[24] Vincent, A., and Meneguzzi, M. The spatial structure and statistical properties of homogeneous turbulence. *Journal of Fluid Mechanics 225*, -1 (1991), 1–20.

[25] Yakhot, V., Thangam, S., Gatski, T., Orszag, S., and Speziale, C. Development of turbulence models for shear flows by a double expansion technque. Tech. Rep. 91-65, NASA ICASE, 1991.

[26] Yamada, M., and Ohkitani, K. Lyapunov spectrum of a chaotic model of three-dimensional turbulence. *J. Phys. Soc. Jap. 56*, 12 (1987), 4210–4213.