GPU Computing Gems

Jade Edition

Morgan Kaufmann's *Applications of GPU Computing* Series

Computing is quickly becoming the third pillar of scientific research, due in large part to the performance gains achieved through graphics processing units (GPUs), which have become ubiquitous in handhelds, laptops, desktops, and supercomputer clusters. Morgan Kaufmann's ***Applications of GPU Computing*** series offers training, examples, and inspiration for researchers, engineers, students, and supercomputing professionals who want to leverage the power of GPUs incorporated into their simulations or experiments. Each high-quality, peer-reviewed book is written by leading experts uniquely qualified to provide parallel computing insights and guidance.

Each *GPU Computing Gems* volume offers a snapshot of the state of parallel computing across a carefully selected subset of industry domains, giving you a window into the lead-edge research occurring across the breadth of science, and the opportunity to observe others' algorithm work that might apply to your own projects. Find out more at `http://mkp.com/gpu-computing-gems`.

Recommended Parallel Computing Titles

Programming Massively Parallel Processors
A Hands-on Approach
By David B. Kirk and Wen-mei W. Hwu
ISBN: 9780123814722

GPU Computing Gems: Emerald Edition
Editor-in-Chief: Wen-mei W. Hwu
ISBN: 9780123849885

The Art of Multiprocessor Programming
By Maurice Herlihy and Nir Shavit
ISBN: 9780123705914

An Introduction to Parallel Programming
By Peter Pacheco
ISBN: 9780123742605

Heterogeneous Computing with OpenCL
By Benedict R. Gaster, Lee Howes, David R. Kaeli, Perhaad Mistry, Dana Schaa
ISBN: 9780123877666

CUDA Application Design and Development
By Rob Farber
ISBN: 9780123884268
Coming Winter 2011

Distributed and Cloud Computing
By Kai Hwang, Geoffrey Fox, and Jack Dongarra
ISBN: 9780123858801
Coming Fall 2011

GPU Computing Gems
Jade Edition

Wen-mei W. Hwu

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Morgan Kaufmann Publishers is an imprint of Elsevier

Acquiring Editor: Todd Green
Development Editor: Robyn Day
Project Manager: Paul Gottehrer
Designer: Dennis Schaefer

Morgan Kaufmann is an imprint of Elsevier
225 Wyman Street, Waltham, MA 02451, USA

Notices

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods or professional practices, may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information or methods described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
Application submitted

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

ISBN: 978-0-12-385963-1

Contents

Editors, Reviewers, and Authors ix
Introduction xv
Wen-mei W. Hwu

SECTION 1 PARALLEL ALGORITHMS AND DATA STRUCTURES

Paulius Micikevicius (NVIDIA)

CHAPTER 1 **Large-Scale GPU Search 3**
Tim Kaldewey and Andrea Di Blas

CHAPTER 2 **Edge v. Node Parallelism for Graph Centrality Metrics 15**
Yuntao Jia, Victor Lu, Jared Hoberock, Michael Garland, and John C. Hart

CHAPTER 3 **Optimizing Parallel Prefix Operations for the Fermi Architecture 29**
Mark Harris and Michael Garland

CHAPTER 4 **Building an Efficient Hash Table on the GPU 39**
Dan A. Alcantara, Vasily Volkov, Shubhabrata Sengupta, Michael Mitzenmacher, John D. Owens, and Nina Amenta

CHAPTER 5 **Efficient CUDA Algorithms for the Maximum Network Flow Problem 55**
Jiadong Wu, Zhengyu He, and Bo Hong

CHAPTER 6 **Optimizing Memory Access Patterns for Cellular Automata on GPUs 67**
James Balasalle, Mario A. Lopez, and Matthew J. Rutherford

CHAPTER 7 **Fast Minimum Spanning Tree Computation 77**
Pawan Harish, P.J. Narayanan, Vibhav Vineet, and Suryakant Patidar

CHAPTER 8 **Comparison-Based In-Place Sorting with CUDA 89**
Hagen Peters and Ole Schulz-Hildebrandt

SECTION 2 NUMERICAL ALGORITHMS

Frank Jargstorff (NVIDIA)

CHAPTER 9 **Interval Arithmetic in CUDA** **99**
Sylvain Collange, Marc Daumas, and David Defour

CHAPTER 10 **Approximating the `erfinv` Function** **109**
Mike Giles

CHAPTER 11 **A Hybrid Method for Solving Tridiagonal Systems on the GPU** **117**
Yao Zhang, Jonathan Cohen, Andrew A. Davidson, and John D. Owens

CHAPTER 12 **Accelerating CULA Linear Algebra Routines with Hybrid GPU and Multicore Computing** **133**
John Humphrey, Daniel Price, Kyle Spagnoli, and Eric Kelmelis

CHAPTER 13 **GPU Accelerated Derivative-Free Mesh Optimization** **145**
Eric Shaffer and George Zagaris

SECTION 3 ENGINEERING SIMULATION

Peng Wang (NVIDIA)

CHAPTER 14 **Large-Scale Gas Turbine Simulations on GPU Clusters** **157**
Tobias Brandvik and Graham Pullan

CHAPTER 15 **GPU Acceleration of Rarefied Gas Dynamic Simulations** **173**
Aldo Frezzotti, Gian Pietro Ghiroldi, and Livio Gibelli

CHAPTER 16 **Application of Assembly of Finite Element Methods on Graphics Processors for Real-Time Elastodynamics** **187**
Cris Cecka, Adrian Lew, and Eric Darve

CHAPTER 17 **CUDA Implementation of Vertex-Centered, Finite Volume CFD Methods on Unstructured Grids with Flow Control Applications** **207**
X.S. Trompoukis, V.G. Asouti, I.C. Kampolis, and K.C. Giannakoglou

CHAPTER 18 **Solving Wave Equations on Unstructured Geometries** **225**
Andreas Klöckner, Timothy Warburton, and Jan S. Hesthaven

CHAPTER 19 **Fast Electromagnetic Integral Equation Solvers on Graphics Processing Units** **243**
Shaojing Li, Ruinan Chang, and Vitaliy Lomakin

SECTION 4 INTERACTIVE PHYSICS AND AI FOR GAMES AND ENGINEERING SIMULATION

Richard Tonge (NVIDIA)

CHAPTER 20 Solving Large Multibody Dynamics Problems on the GPU 269
Dan Negrut, Alessandro Tasora, Mihai Anitescu, Hammad Mazhar, Toby Heyn, and Arman Pazouki

CHAPTER 21 Implicit FEM Solver on GPU for Interactive Deformation Simulation 281
Jérémie Allard, Hadrien Courtecuisse, and François Faure

CHAPTER 22 Real-Time Adaptive GPU Multiagent Path Planning 295
Ugo Erra and Giuseppe Caggianese

SECTION 5 COMPUTATIONAL FINANCE

Thomas Bradley (NVIDIA)

CHAPTER 23 Pricing Financial Derivatives with High Performance Finite Difference Solvers on GPUs 309
Daniel Egloff

CHAPTER 24 Large-Scale Credit Risk Loss Simulation 323
Simon J. Rees and Joseph Walkenhorst

CHAPTER 25 Monte Carlo–Based Financial Market Value-at-Risk Estimation on GPUs 337
Matthew F. Dixon, Thomas Bradley, Jike Chong, and Kurt Keutzer

SECTION 6 PROGRAMMING TOOLS AND TECHNIQUES

Cliff Woolley (NVIDIA)

CHAPTER 26 Thrust: A Productivity-Oriented Library for CUDA 359
Nathan Bell and Jared Hoberock

CHAPTER 27 GPU Scripting and Code Generation with PyCUDA 373
Andreas Klöckner, Nicolas Pinto, Bryan Catanzaro, Yunsup Lee, Paul Ivanov, and Ahmed Fasih

CHAPTER 28 Jacket: GPU Powered MATLAB Acceleration 387
Torben Larsen, Gallagher Pryor, and James Malcolm

CHAPTER 29 **Accelerating Development and Execution Speed with Just-in-Time GPU Code Generation** **399**
Peter Eastman and Vijay Pande

CHAPTER 30 **GPU Application Development, Debugging, and Performance Tuning with GPU Ocelot** **409**
Andrew Kerr, Gregory Diamos, and Sudhakar Yalamanchili

CHAPTER 31 **Abstraction for AoS and SoA Layout in C++** **429**
Robert Strzodka

CHAPTER 32 **Processing Device Arrays with C++ Metaprogramming** **443**
Jonathan M. Cohen

CHAPTER 33 **GPU Metaprogramming: A Case Study in Biologically Inspired Machine Vision** **457**
Nicolas Pinto and David D. Cox

CHAPTER 34 **A Hybridization Methodology for High-Performance Linear Algebra Software for GPUs** **473**
Emmanuel Agullo, Cédric Augonnet, Jack Dongarra, Hatem Ltaief, Raymond Namyst, Samuel Thibault, and Stanimire Tomov

CHAPTER 35 **Dynamic Load Balancing Using Work-Stealing** **485**
Daniel Cederman and Philippas Tsigas

CHAPTER 36 **Applying Software-Managed Caching and CPU/GPU Task Scheduling for Accelerating Dynamic Workloads** **501**
Mark Silberstein, Assaf Schuster, and John D. Owens

Index **519**

Editors, Reviewers, and Authors

Editor-In-Chief

Wen-Mei W. Hwu, University of Illinois at Urbana-Champaign

Managing Editor

Andrew Schuh, University of Illinois at Urbana-Champaign

NVIDIA Editor

Nadeem Mohammad, NVIDIA Corporation

Area Editors

Thomas Bradley, NVIDIA Corporation (Section 5)

Frank Jargstorff, NVIDIA Corporation (Section 2)

Paulius Micikevicius, NVIDIA Corporation (Section 1)

Richard Tonge, NVIDIA Corporation (Section 4)

Peng Wang, NVIDIA Corporation (Section 3)

Cliff Wooley, NVIDIA Corporation (Section 6)

Reviewers

John Ashley, NVIDIA Corporation

Nathan Bell, NVIDIA Corporation

Avi Bleiweiss, NVIDIA Corporation

Jonathan Cohen, NVIDIA Corporation

Andrew Corrigan, Naval Research Lab

Mike Giles, Oxford University

Dilip Sequeira, NVIDIA Corporation

David Sullins, NVIDIA Corporation

Stanimire Tomov, University of Tennessee, Knoxville

Cliff Wooley, NVIDIA Corporation

Gernot Ziegler, NVIDIA Corporation

Authors

Emmanuel Agullo, INRIA (Chapter 34)
Dan A. Alcantara, UC Davis (Chapter 4)
Jérémie Allard, INRIA Lille & Lille University (Chapter 21)
Nina Amenta, UC Davis (Chapter 4)
Mihai Anitescu, Argonne National Laboratory (Chapter 20)
V.G. Asouti, National Technical University of Athens (Chapter 17)
Cédric Augonnet, INRIA (Chapter 34)
James Balasalle, University of Denver (Chapter 6)
Nathan Bell, NVIDIA Corporation (Chapter 26)
Thomas Bradley, NVIDIA Corporation (Chapter 25)
Tobias Brandvik, University of Cambridge (Chapter 14)
Giuseppe Caggianese, Università degli Studi della Basilicata (Chapter 22)
Bryan Catanzaro, UC Berkeley (Chapter 27)
Cris Cecka, Stanford University (Chapter 16)
Daniel Cederman, Sweden Chalmers University of Technology (Chapter 35)
Ruinan Chang, UC San Diego (Chapter 19)
Jike Chong, Parasians LLC & UC Berkeley (Chapter 25)
Jonathan Cohen, NVIDIA Corporation (Chapters 11, 32)
Sylvain Collange, École normale supérieure de Lyon (Chapter 9)
Hadrien Courtecuisse, INRIA Lille & Lille University (Chapter 21)
David D. Cox, Harvard University (Chapter 33)
Eric Darve, Stanford University (Chapter 16)
Marc Daumas, Université de Perpignan (Chapter 9)
Andrew A. Davidson, UC Davis (Chapter 11)
David Defour, Université de Perpignan (Chapter 9)
Gregory Diamos, Georgia Institute of Technology (Chapter 30)
Andrea Di Blas, Oracle Corporation (Chapter 1)
Matthew F. Dixon, UC Davis (Chapter 25)
Jack Dongarra, University of Tennessee, Knoxville (Chapter 34)
Peter Eastman, Stanford University (Chapter 29)
Daniel Egloff, QuantAlea GmbH (Chapter 23)

Ugo Erra, Università degli Studi della Basilicata (Chapter 22)
Ahmed Fasih, Ohio State University (Chapter 27)
François Faure, INRIA Rhóne-Alpes & Grenoble University (Chapter 21)
Aldo Frezzotti, Politecnico di Milano (Chapter 15)
Michael Garland, NVIDIA Corporation (Chapters 2 and 3)
Gian Pietro Ghiroldi, Politecnico di Milano (Chapter 15)
K.C. Giannakoglou, National Technical University of Athens (Chapter 17)
Livio Gibelli, Politecnico di Milano (Chapter 15)
Mike Giles, Oxford University (Chapter 10)
Pawan Harish, International Institute of Information Technology, Hyderabad (Chapter 7)
Mark Harris, NVIDIA Corporation (Chapter 3)
John C. Hart, University of Illinois at Urbana-Champaign (Chapter 2)
Zhengyu He, Georgia Institute of Technology (Chapter 5)
Jan S. Hesthaven, Brown University (Chapter 18)
Toby Heyn, Univeristy of Wisconsin, Madison (Chapter 20)
Jared Hoberock, NVIDIA Corporation (Chapters 2 and 26)
Bo Hong, Georgia Institute of Technology (Chapter 5)
John Humphrey, EM Photonics, Inc. (Chapter 12)
Paul Ivanov, UC Berkeley (Chapter 27)
Yuntao Jia, University of Illinois at Urbana-Champaign (Chapter 2)
Tim Kaldewey, IBM (Chapter 1)
I.C. Kampolis, National Technical University of Athens (Chapter 17)
Eric Kelmelis, EM Photonics, Inc. (Chapter 12)
Andrew Kerr, Georgia Institute of Technology (Chapter 30)
Kurt Keutzer, UC Berkeley (Chapter 25)
Andreas Klöckner, New York University (Chapters 18 and 27)
Torben Larsen, Aalborg University (Chapter 28)
Yunsup Lee, UC Berkeley (Chapter 27)
Adrian Lew, Stanford University (Chapter 16)
Shaojing Li, UC San Diego (Chapter 19)
Vitaliy Lomakin, UC San Diego (Chapter 19)
Mario A. Lopez, University of Denver (Chapter 6)

Hatem Ltaief, University of Tennessee, Knoxville (Chapter 34)
Victor Lu, University of Illinois at Urbana-Champaign (Chapter 2)
James Malcolm, AccelerEyes (Chapter 28)
Hammad Mazhar, University of Wisconsin, Madison (Chapter 20)
Michael Mitzenmacher, Harvard University (Chapter 4)
Raymond Namyst, INRIA (Chapter 34)
P.J. Narayanan, International Institute of Information Technology, Hyderabad (Chapter 7)
Dan Negrut, University of Wisconsin, Madison (Chapter 20)
John D. Owens, UC Davis (Chapters 4, 11, 36)
Vijay Pande, Stanford University (Chapter 29)
Suryakant Patidar, NVIDIA Corporation (Chapter 7)
Arman Pazouki, Univeristy of Wisconsin, Madison (Chapter 20)
Hagen Peters, Christian-Albrechts-University of Kiel (Chapter 8)
Nicolas Pinto, Massachusetts Institute of Technology (Chapters 27 and 33)
Daniel Price, EM Photonics, Inc. (Chapter 12)
Gallagher Pryor, AccelerEyes (Chapter 28)
Graham Pullan, University of Cambridge (Chapter 14)
Simon J. Rees, Barclays Capital (Chapter 24)
Matthew J. Rutherford, University of Denver (Chapter 6)
Ole Schulz-Hildebrandt, Christian-Albrechts-University of Kiel (Chapter 8)
Assaf Schuster, Technion (Chapter 36)
Shubhabrata Sengupta, UC Davis (Chapter 4)
Eric Shaffer, University of Illinois at Urbana-Champaign (Chapter 13)
Mark Silberstein, Technion (Chapter 36)
Kyle Spagnoli, EM Photonics, Inc. (Chapter 12)
Robert Strzodka, Max Planck Institut Informatik (Chapter 31)
Alessandro Tasora, University of Parma (Chapter 20)
Samuel Thibault, INRIA (Chapter 34)
Stanimire Tomov, University of Tennessee, Knoxville (Chapter 34)
X.S. Trompoukis, National Technical University of Athens (Chapter 17)
Philippas Tsigas, Sweden Chalmers University of Technology (Chapter 35)
Vibhav Vineet, International Institute of Information Technology, Hyderabad (Chapter 7)

Vasily Volkov, UC Berkeley (Chapter 4)

Joseph Walkenhorst, Barclays Capital (Chapter 24)

Timothy Warburton, Rice University (Chapter 18)

Jiadong Wu, Georgia Institute of Technology (Chapter 5)

Sudhakar Yalamanchili, Georgia Institute of Technology (Chapter 30)

George Zagaris, Illinois Rocstar LLC (Chapter 13)

Yao Zhang, UC Davis (Chapter 11)

Introduction

Wen-mei W. Hwu

STATE OF GPU COMPUTING

We are in the golden age of the GPU computing. Since the introduction of CUDA in 2007, more than 100 million computers with CUDA capable GPUs have been shipped to end users. GPU computing application developers can now expect their application to have a mass market. With the introduction of OpenCL in 2010, researchers can now expect to develop GPU applications that can run on hardware from multiple vendors. Furthermore, from my own experience in teaching CUDA and OpenCL programming, C programmers can begin to write basic programs after only attending one lecture and reading one textbook chapter. With such a low barrier of entry, researchers all over the world have been engaged in developing new algorithms and applications to take advantage of the extreme floating point execution throughput of these GPUs.

Today, there is a large community of GPU computing practitioners. Many of them have reported 10 to 100 times speedup of their applications with GPU computing. To put this into perspective, with the historical 2× performance growth every two years, these researchers are experiencing the equivalent of time travel of 8 to 12 years. That is, they are getting today the performance they would have to wait for 8 to 12 years if they went for the "free-ride" advancement of performance in microprocessors. Interestingly, such a free ride is no longer available. Furthermore, once they developed their application in CUDA, they will likely see continued performance growth of 2× for every two years from this day forward.

After discussing with numerous researchers, I reached the conclusion that many of them are solving similar algorithm problems in their programming efforts. Although they are working on diverse applications, they often end up developing similar algorithmic strategies. The idea of *GPU Computing Gems* is to provide a convenient means for application developers in diverse application areas to benefit from each other's experience. When we issued the call for proposals for the first *GPU Computing Gems*, we received more than 280 submissions, an overwhelming response. In the end, we accepted 86 final chapters. Many high-quality proposals were not accepted because of concerns that they may not be accessible to a large audience. With so many chapters, we were forced to divide these gems into two volumes. In the first volume (Emerald Edition), we published 50 Gems from 10 diverse application areas. In this volume, we have collected 36 Gem articles written by researchers in 5 diverse application areas as well as programming tools and techniques. Each gem is first edited by an area editor who is a GPU computing expert in that area. This is followed by my own editing of these articles.

For applications, each Gems article reports a successful application experience in GPU computing. These articles describe the techniques or "secret sauce" that contributed to the success. The authors highlight the potential applicability of their techniques to other application areas. In our editorial

process, we have emphasized the accessibility of these gems to researchers in other areas. For programming tools and techniques, the authors emphasize how an application developer can best benefit from their work.

I would like to thank several people who have worked tirelessly on this project. Nadeem Mohammad at NVIDIA and Andrew Schuh at UIUC have done so much heavy lifting for this project. Without them, it would have been impossible for me to coordinate so many authors and area editors. My area editors, whose names are in front of each section of this volume, have volunteered their valuable time and energy to improve the quality of the gems. They worked closely with the authors to make sure that the gems indeed meet high technical standard while remain accessible by a wide audience. I would like to thank all the authors who have shared their innovative work with the GPU computing community. All authors have worked hard to respond to our requests for improvements. It has been a true privilege to work with all these great people.

Online Resources

Visit `http://mkp.com/gpu-computing-gems` and click the Online Resources tab to connect to `http://gpucomputing.net`, the vibrant official community site for GPU computing, where you can download source code examples for most chapters and join discussions with other readers and GPU developers. You'll also find links to additional material including chapter walk-through videos and full-color versions of many figures from the book.