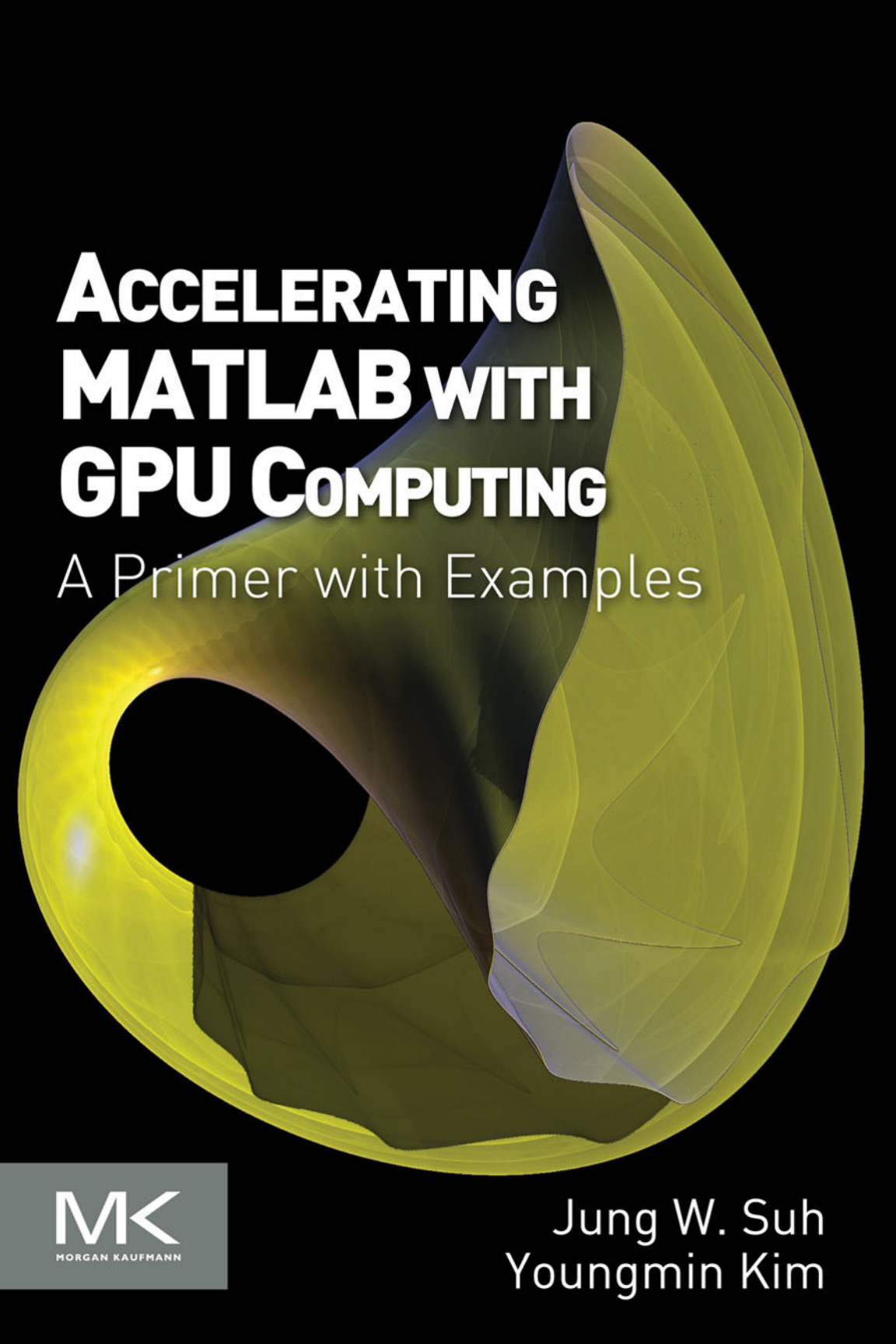
ACCELERATING
MATLAB WITH
GPU COMPUTING
A Primer with Examples
MORGAN KAUFMANN
Jung W. Suh
Youngmin Kim

Accelerating MATLAB with GPU Computing

This page intentionally left blank

Accelerating MATLAB with GPU Computing

A Primer with Examples

Jung W. Suh
Youngmin Kim

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann is an imprint of Elsevier

Acquiring Editor: Todd Green
Editorial Project Manager: Lindsay Lawrence
Project Manager: Mohana Natarajan
Designer: Matthew Limbert

Morgan Kaufmann is an imprint of Elsevier

225 Wyman Street, Waltham, MA 02451, USA

First edition 2014

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

Library of Congress Cataloging-in-Publication Data
Application Submitted

ISBN: 978-0-12-408080-5

Contents

Preface

MATLAB is a widely used simulation tool for rapid prototyping and algorithm development. Many laboratories and research institutions face growing demands to run their MATLAB codes faster for computationally heavy projects after simple simulations. Since MATLAB uses a vector/matrix representation of data, which is suitable for parallel processing, it can benefit a lot from GPU acceleration.

Target Readers and Contents

This book is aimed primarily at the graduate students and researchers in the field of engineering, science, and technology who need huge data processing without losing the many benefits of MATLAB. However, MATLAB users come from various backgrounds and do not necessarily have much programming experience. For those whose backgrounds are not from programming, GPU acceleration for MATLAB may distract their algorithm development and introduce unnecessary hassles, even when setting the environment. This book targets the readers who have some or a lot of experience on MATLAB coding but not enough depth in either C coding or the computer architecture for parallelization. So readers can focus more on their research and work by avoiding non-algorithmic hassles in using GPU and CUDA in MATLAB.

As a primer, the book will start with the basics, walking through the process of setting MATLAB for CUDA (in Windows and Mac OSX), creating `c-mex` and `m`-file profiling, then guide the users through the expert-level topics such as third-party CUDA libraries. It also provides many practical ways to modify users' MATLAB codes to better utilize the immense computational power of graphics processors.

This book guides the reader to dramatically maximize the MATLAB speed using NVIDIA's Graphics Processing Unit (GPU). NVIDIA's Compute Unified Device Architecture (CUDA) is a parallel computing architecture originally designed for computer games but is getting a reputation in the general science and technology fields for its efficient massive computation power. From this book, the reader can take advantage of the parallel processing power of GPU and abundant CUDA scientific libraries for accelerating MATLAB code with no or less effort and time, and bring readers' researches and works to a higher level.

Directions of this Book

GPU Utilization Using c-mex Versus Parallel Computing Toolbox

This book deals with Mathworks's Parallel Computing Toolbox in Chapter 5. Although Mathworks's Parallel Computing Toolbox is a useful tool for speeding