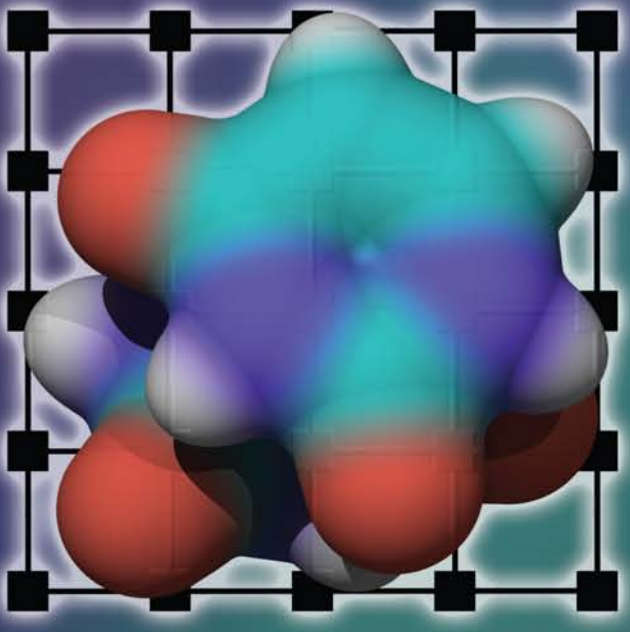
Parallel Computing *in* Quantum Chemistry



Curtis L. Janssen Ida M. B. Nielsen

CRC Press
Taylor & Francis Group

Parallel Computing
in
Quantum Chemistry

Parallel Computing *in* Quantum Chemistry

Curtis L. Janssen

Ida M. B. Nielsen

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number-13: 978-1-4200-5164-3 (Hardcover)

Library of Congress Cataloging-in-Publication Data

Janssen, Curtis L.
Parallel computing in quantum chemistry / Curtis L. Janssen and Ida M.B. Nielsen.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-4200-5164-3 (acid-free paper)
1. Quantum chemistry--Data processing. 2. Parallel programming (Computer science) 3. Parallel processing (Electronic computers) I. Nielsen, Ida M. B. II. Title.

QD462.6.D38J36 2008
541'.280285435--dc22 2007048052

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

Preface

This book is intended to serve as a reference for the design and implementation of parallel quantum chemistry programs. Development of efficient quantum chemistry software capable of utilizing large-scale parallel computers requires a grasp of many issues pertaining to both parallel computing hardware and parallel programming practices, as well as an understanding of the methods to be implemented. The text provides an in-depth view of parallel programming challenges from the perspective of a quantum chemist, including parallel computer architectures, message-passing, multi-threading, parallel program design and performance analysis, as well as parallel implementation of important electronic structure procedures and methods such as two-electron integral computation, Hartree–Fock and second-order Møller–Plesset perturbation (MP2) theory, and the local correlation method LMP2. Some topics relevant to parallel computing in quantum chemistry have not been included in this book. Thus, performance tools and debuggers are not treated, parallel I/O is only briefly discussed, and advanced electronic structure methods such as coupled-cluster theory and configuration interaction are not covered.

We will assume that the reader has a basic understanding of quantum chemistry, including Hartree–Fock theory and correlated electronic structure methods such as Møller–Plesset perturbation theory. Readers can find introductory discussions of these methods in, for example, Jensen[1] and the classic Szabo and Ostlund text.[2] A comprehensive and somewhat more advanced treatment of electronic structure theory can be found in Helgaker, Jørgensen, and Olsen.[3] No prior experience with parallel computing is required, but the reader should be familiar with computer programming and programming languages at the advanced undergraduate level. The program examples in the book are written in the C programming language, and at least a rudimentary knowledge of C will therefore be helpful. The text by Kernighan and Ritchie[4] covers all C features used in this book.

Scope and Organization of the Text

This book is divided into two parts. In Part I we will discuss parallel computer architectures as well as parallel computing concepts and terminology with a focus on good parallel program design and performance

analysis. Part II contains detailed discussions and performance analyses of parallel algorithms for a number of important and widely used quantum chemistry procedures and methods.

An outline of the contents of each chapter is given below.

Chapter 1: Introduction

Here we provide a brief history of parallel computing in quantum chemistry and discuss trends in hardware as well as trends in the methods and algorithms of quantum chemistry. The impact of these trends on future quantum chemistry programs will be considered.

Chapter 2: Parallel Computer Architectures

This chapter provides an overview of parallel computer architectures, including the traditional Flynn classification scheme and a discussion of computation nodes and the networks connecting them. We also present an overall system view of a parallel computer, describing the hierarchical nature of parallel architecture, machine reliability, and the distinction between commodity and custom computers.

Chapter 3: Communication via Message-Passing

This chapter covers message-passing, one of the primary software tools required to develop parallel quantum chemistry programs for distributed memory parallel computers. Point-to-point, collective, and one-sided varieties of message-passing are also discussed.

Chapter 4: Multi-Threading

The importance of multi-threading will continue to increase due to the emergence of multicore chips. Parallelization by means of multi-threading is discussed as well as hybrid multi-threading/message-passing approaches for utilizing large-scale parallel computers.

Chapter 5: Parallel Performance Evaluation

Design and implementation of efficient parallel algorithms requires careful analysis and evaluation of their performance. This chapter introduces idealized machine models along with measures for predicting and assessing the performance of parallel algorithms.

Chapter 6: Parallel Program Design

This chapter discusses fundamental issues involved in designing and implementing parallel programs, including the distribution of tasks and data as well as schemes for interprocessor communication.

Chapter 7: Two-Electron Integral Evaluation

An important, basic step performed in most quantum chemistry programs is the computation of the two-electron integrals. Schemes for parallel computation of these integrals and detailed performance models incorporating load imbalance are discussed.

Chapter 8: The Hartree–Fock Method

The Hartree–Fock method is central to quantum chemistry, and an efficient Hartree–Fock program is an essential part of a quantum chemistry program package. We outline the Hartree–Fock procedure and present and analyze both replicated data and distributed data Fock matrix formation algorithms.

Chapter 9: Second-Order Møller–Plesset Perturbation Theory

Second-order Møller–Plesset (MP2) perturbation theory is a widely used quantum chemical method for incorporating electron correlation. This chapter considers parallel computation of MP2 energies, comparing the performance achievable with simple and more sophisticated parallelization strategies.

Chapter 10: Local Møller–Plesset Perturbation Theory

Local correlation methods represent an important new class of correlated electronic structure methods that aim at computing molecular properties with the same accuracy as their conventional counterparts but at a significantly lower computational cost. We discuss the challenges of parallelizing local correlation methods in the context of local second-order Møller–Plesset perturbation theory, illustrating a parallel implementation and presenting benchmarks as well.

Appendix A: A Brief Introduction to MPI

The Message-Passing Interface (MPI) is the primary mechanism used for explicit message-passing in scientific computing applications. This appendix briefly discusses some of the most commonly used MPI routines.

Appendix B: Pthreads

Pthreads is a standard for creating and managing multiple threads. We give a brief introduction to multi-threaded programming with Pthreads, including an example Pthreads program.

Appendix C: OpenMP

OpenMP is a set of compiler extensions to facilitate development of multi-threaded programs. We describe these compiler extensions, using example source code illustrating parallel programming with OpenMP.

References

1. Jensen, F. *Introduction to Computational Chemistry*. Chichester, UK: John Wiley & Sons, 1999.
2. Szabo, A., and N. S. Ostlund. *Modern Quantum Chemistry*, 1st revised edition. New York: McGraw-Hill, 1989.
3. Helgaker, T., P. Jørgensen, and J. Olsen. *Molecular Electronic-Structure Theory*. Chichester, UK: John Wiley & Sons, 2000.
4. Kernighan, B. W., and D. M. Ritchie. *The C Programming Language*, 2nd edition. Englewood Cliffs, NJ: Prentice Hall, 1988.

Acknowledgments

We are indebted to many people who provided support and advice in the preparation of this book. Lance Wobus of Taylor & Francis encouraged us to undertake the writing of the book, and Jennifer Smith expertly guided us through the final stages of manuscript preparation. We thank Craig Smith and Kurt Olsen at Sandia National Laboratories for facilitating the project, and we appreciate the assistance of Gregg Andreski, who prepared most of the artwork. Jan Linderberg kindly provided historical references, and Shawn Brown, Joe Kenny, and Matt Leininger made insightful comments on various drafts of the manuscript. Ron Minnich, Helgi Adalsteinsson, Jim Schutt, Ed Valeev, Daniel Crawford, Theresa Windus, and David Bernholdt provided stimulating discussions, feedback, and technical data. Finally, CLJ would like to thank Mike Colvin for introducing him to parallel computing.

Funding Statement

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Authors

Curtis L. Janssen is a Distinguished Member of the Technical Staff at Sandia National Laboratories and holds a Ph.D. in theoretical chemistry from the University of California at Berkeley. He has an extensive publication record in the areas of quantum chemistry methods development and high-performance computing, and he is the lead developer of the MPQC program suite.

Curtis L. Janssen
Sandia National Laboratories
7011 East Ave.
Livermore, CA 94550
cljanss@sandia.gov

Ida M. B. Nielsen is a scientist at Sandia National Laboratories and holds a Ph.D. in theoretical chemistry from Stanford University. She has published numerous articles on the development and application of quantum chemical methods and high-performance computing and is one of the core developers of the MPQC program suite.

Ida M. B. Nielsen
Sandia National Laboratories
7011 East Ave.
Livermore, CA 94550
ibniels@sandia.gov

Disclaimer of Liability

Trademarks

Part I

Parallel Computing Concepts and Terminology

1

Introduction

The need for parallel software for scientific computing is ever increasing. Supercomputers are not only being built with more processors, but parallel computers are also no longer limited to large machines owned and managed by high-performance computing centers; parallel desktop computers are increasingly widespread, and even laptop computers have multiple processors. Development of scientific computing software must adapt to these conditions as parallel computation becomes the norm rather than the exception. In the field of quantum chemistry, additional factors contribute to the need for parallelism. Quantum chemistry has become an indispensable tool for investigating chemical phenomena, and quantum chemical methods are employed widely in research across many chemical disciplines; this widespread use of quantum chemistry reinforces the importance of rapid turnaround computations, which can be addressed by parallel computing. Additionally, for quantum chemistry to continue to be an integral part of chemical research, quantum chemical methods must be applicable to the chemical systems of interest, including larger molecules, and parallelism can play an important role in extending the range of these methods. Parallel implementations can broaden the scope of conventional quantum chemical methods, whose computational cost scales as a high-degree polynomial in the molecular size, and enable the treatment of very large molecular systems with linear-scaling or reduced-scaling methods.

In the following, we will first give a brief historical sketch and current perspective of parallel computing in quantum chemistry. We will then discuss trends in hardware development for single-processor and parallel computers as well as trends in parallel software development, including the parallel programming challenges following the emergence of new quantum chemical methods and the changes in hardware.