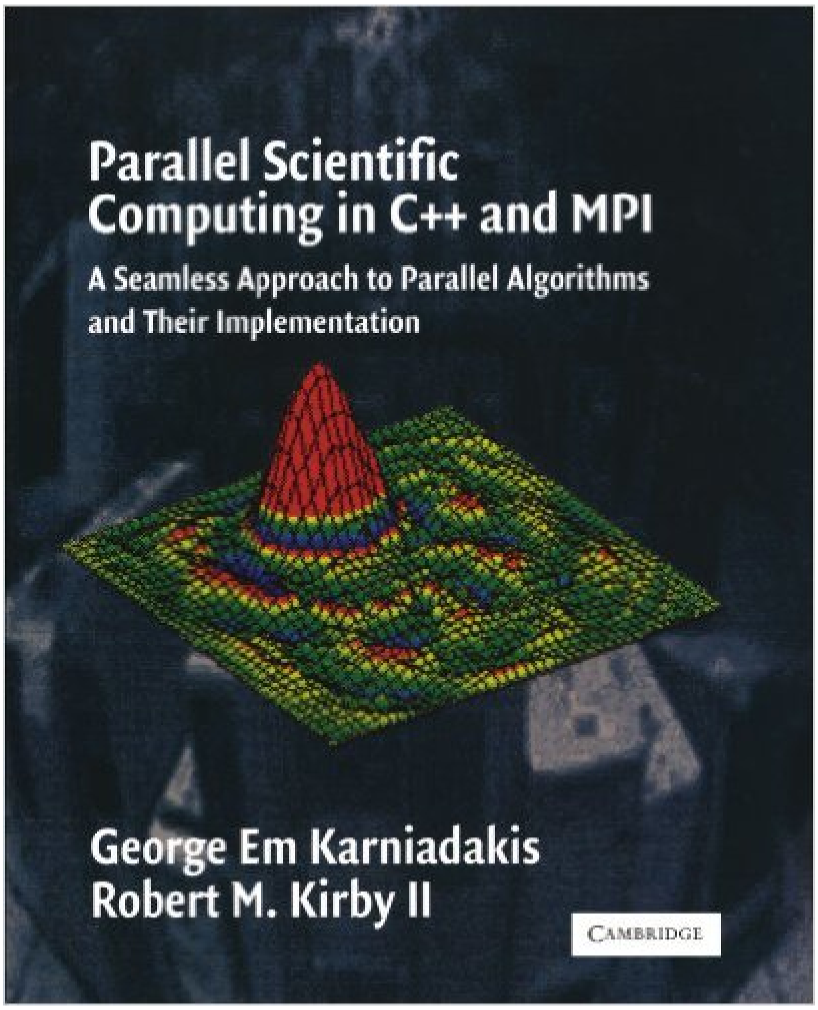
Parallel Scientific Computing In C++ And MPI: A Seamless Approach To Parallel Algorithms And Their Implementation



Synopsis

This book provides a seamless approach to numerical algorithms, modern programming techniques and parallel computing. These concepts and tools are usually taught serially across different courses and different textbooks, thus observing the connection between them. The necessity of integrating these subjects usually comes after such courses are concluded (e.g, during a first job or a thesis project) thus forcing the student to synthesize what is perceived to be three independent subfields into one in order to produce a solution. The book includes both basic and advanced topics and places equal emphasis on the discretization of partial differential equations and on solvers. Advanced topics include wavelets, high-order methods, non-symmetric systems and parallelization of sparse systems. A CD-ROM accompanies the text.

Book Information

Paperback: 630 pages

Publisher: Cambridge University Press; PAP/CDR edition (June 16, 2003)

Language: English

ISBN-10: 0521520800

ISBN-13: 978-0521520805

Product Dimensions: 8.5 x 1.3 x 9.4 inches

Shipping Weight: 2.9 pounds (View shipping rates and policies)

Average Customer Review: 3.8 out of 5 starsÂ Â See all reviewsÂ (5 customer reviews)

Best Sellers Rank: #872,633 in Books (See Top 100 in Books) #88 inÂ Books > Computers & Technology > Programming > Parallel Programming #438 inÂ Books > Computers & Technology > Computer Science > Systems Analysis & Design #476 inÂ Books > Computers & Technology > Programming > Languages & Tools > C & C++ > C++

Customer Reviews

The authors attempted to combine introductory material in C++, numerical methods and parallel computing. That is quite a brave endevaour. They certainly break the new ground writing an introductory material for a "simulation scientist", but I believe they have achieved mixed success.On the one hand, the material they present on all subjects is really top quality, packed with 100% usefull information. Bibliography is also very good and usefull. But the organisation of the book is quite confusing. They introduce all the topics toogether - throughtout the book. Hence each chapter introduces some numerical algorithms, few new concepts in C++ and eventually MPI. I beleive a novice would experience serious difficulties following it. For example, authors introduce objects before introducing curly braces "{}" as scope delimiters and before semicolon ";" as statement delimiters.Further, very soon after introducung the very basic concepts in C++, the authors move on to BLAS. BLAS is usefull, of course, but a novice in C++ may wonder why does he needs libraries written in Fortran, if C++ is a language of the choice for numerical computations. (At least it is claimed so by the authors).Another confusing example is the one of memory memory access. In section 2.2.6 Memory Management, (pg. 41) the authors introduce basic concepts of memory management and how can loop constructs influence the efficiency of the code. Very usefull indeed, no question about it. But very soon bellow, in section 2.2.8 Exploiting the Structure of the Sparse Matrices, they come up with the claim (pg. 58): "... optimization-savy individuals, as the old saying goes, often miss the forest for the threes" :-(Hence, a novice reader might think: "Well, why do I need to worry about the memory management explained just 17 pages above?".My most serious critic of this book by far (and I hope the authors will read this) are the contents pages. The contents list only chapters and first level sub-chapters. Secind level chapters are not present!!! That makes the book very hard to use as a reference material. That is really a pitty, since there is some good material in it which is hard to find and might stay hidden. (For example, the chapter I mentioned above: 2.2.6 Memory management is NOT in the contents, so I had to browse slowly throught the book to find it and refer it here). I suggest the authors introduce: "Contents at a glance" (the present one) and a "Detailed Contents", where one could find references to all the chapters in the book. The contents is THE reason why I gave this book 3 stars instead of 4. One it lost on the confusing organisation of the book.I think the authors should have organised the book in four parts: 1 - Numerical algorithms, 2 - C++ and 3 - Parallel computing with MPI, 4 - Advanced topics. Part 1 could introduce numerical algorithms and have pointers to their implementations in Part 2 and corresponding parallel implementations in Part 3. Part 2 and 3 could have started with introductions, which a reader already familiar with those subjects, could skip. Part 4, could bring advanced topics, such as optimisation, BLAS, etc.Bottom line, it is:- brave and usefull endevaour,- full of excellent material,- organized confusingly,- and has a very poor contents.Buy it if you are simulation scientist or teacher, but prepare to struggle with its organisation and contents.

This book is great in describing some of the most important concepts and algorithms needed for the beginning numerical analyst. The book claims that it can be picked up by a complete novice and teach C++, MPI, and scientfic computing. I would say that the math goes very quickly and not quite as rigorous as necessary for the typical novice. The C++ is pretty basic but still the book leaves the reader a sense of confusion. This is largely because the book treats a large amount of the library functions as black boxes. The MPI starts very basic and gradually introduces the major concepts.My recommendation for anyone reading the book is to supplement it with a good linear algebra book (such as Demmel) and book on C++ (The C++ programming language). For further study on MPI, Using MPI would be a good supplement. That way whenever you have a concept that isn't fully described, you have a source to get it from.The book gets a high rating for going over the right content and doing so in a applied manner that gives the reader the skills to become a numerical analyst.

The book contains advanced numerical mathematics algorithms andfundamental elements of parallel computation.It will be useful for those academic instructors who believe that students should be shown the entire solution processfrom mathematical problem definition to computer implementation. It has been used as a textbook at several leading American and European universities.The authors professors Karniadakis and Kilby are innovators who demonstrate that combining education of applied mathematics with computer science is possible and extremely useful for students and their future employers.

We are facing new and challenging problems in the areas of data mining, bio-informatics, CFD and so on that require very large computational capacity. The availability of commodity hardware components like motherboards and memory chips, together with free software like Linux, the GNU compilers, and the Message-Passing Interface (in which message-passing is used to control the flow of the computation) have put massively parallel machines such as Beowulf clusters within the reach of medium-sized companies and academic departments. As parallel computing continues to merge into the mainstream of computing, it is becoming important for students and professionals to understand the application and analysis of algorithmic paradigms to both the (traditional) sequential model of computing and to various parallel models.Ã¢Â ÂœParallel Scientific Computing in C++ and MPIÃ¢Â Â•, written by George Em Karniadakis and Robert M. Kirby II, is a valiant effort to introduce the student in a unified manner to parallel scientific computing. This textbook offers the student with no previous background in computing three books in one. There is a textbook on the analysis of algorithms, a textbook on parallel programming using MPI 1.x , and finally there is also an elementary book on programming using a subset of the C++ as a better Ã¢Â ÂœCÃ¢Â Â•.Karniadakis is a Professor of Applied Mathematics at Brown University, working on simulations of turbulence in complex geometries. Kirby is an Assistant Professor of Computer Science at the University of Utah specializing in large-scale scientific computing. This textbook is largely based on the KarniadakisÃ¢Â Â™s courses at Brown University and at MIT over the past 15 years and is thus slanted towards computational fluid dynamics. It is very strong as a traditional algorithms-based textbook for an introductory course of numerical analysis at the late undergraduate or early graduate level. It examines such core topics as dense and sparse matrix computations, linear systems, finite differences, and FFT. It assumes a solid technical background including calculus, linear algebra, and differential equations.The initial chapters explain how and why parallel computing began, present an overview of parallel architectures, and introduce MPI 1.x. The authors follow this by discussing the powerful divide-and-conquer paradigm and develop the basics of each topic such as root finding and approximation with sequential and MPI-specific implementation details and a lot of useful (but not optimal) C++ code. Chapters 3, 5, and 6 are the heart of the book where approximation of functions, explicit & implicit discretization, and MPI are discussed in detail. The authors are very careful in establishing the foundation of each algorithm and a great deal of care is taken in explaining and estimating the accuracy of each numerical technique, its stability, and its convergence with benchmarks. Each chapter also includes advice on common programming pitfalls, gotchas and exercises. There are, in fact, 162 homework problems throughout the book.The authors state (on page 4): Ã¢Â ÂœOur book treats numerics, parallelism, and programming equally and simultaneouslyÃ¢Â Â¦.Ã¢Â Â•. The authors do not achieve their stated purpose of treating these topic equally in their treatment of C++ and MPI 1.x. Those looking for examples of how encapsulation, inheritance, exception handling, templates, and polymorphism can be used to control the complexity of developing, debugging, maintaining, and tuning parallel software using MPI will not find it in this book. For a more thorough discussion of MPI 1.x from a software development point of view using Ã¢Â ÂœCÃ¢Â Â•, one might consult Ã¢Â ÂœParallel Programming with MPIÃ¢Â Â•, Morgan Kaufmann Publishers, Inc., (1997). However, neither text treats MPI 2.0 features such as multithreading or C++ bindings for MPI.In spite of falling short of its ambitious goals, this is still a useful textbook for those who would like to know how to write parallel programs using MPI also wish to go beyond such cook-book texts as Ã¢Â ÂœNumerical RecipesÃ¢Â Â•, Cambridge University Press; (1986).

Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and their Implementation MPI: The Complete Reference (Vol. 2), Vol. 2 - The MPI-2 Extensions Using MPI - 2nd Edition: Portable Parallel Programming with the Message Passing Interface (Scientific and Engineering Computation) Introduction to Parallel Computing: Design and Analysis of Parallel Algorithms Using Advanced MPI: Modern Features of the Message-Passing Interface (Scientific and

Engineering Computation) Using MPI-2: Advanced Features of the Message Passing Interface (Scientific and Engineering Computation) CUDA Programming: A Developer's Guide to Parallel Computing with GPUs (Applications of Gpu Computing) Seamless: Understanding the Bible as One Complete Story (Member Book) Programming Massively Parallel Processors: A Hands-on Approach (Applications of GPU Computing Series) MCTS Self-Paced Training Kit (Exam 70-432): MicrosoftÂ® SQL ServerÂ® 2008 - Implementation and Maintenance: Microsoft SQL Server 2008--Implementation and Maintenance (Microsoft Press Training Kit) Asap Implementation at the Speed of Business: Implementation at the Speed of Business Software Fault Tolerance Techniques and Implementation (Artech House Computing Library) The Design of Innovation: Lessons from and for Competent Genetic Algorithms (Genetic Algorithms and Evolutionary Computation) Algorithms in C++ Part 5: Graph Algorithms (3rd Edition) (Pt.5) Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications (Real-Time Systems Series) Biologically Inspired Algorithms for Financial Modelling (Natural Computing Series) Strategic Computing: DARPA and the Quest for Machine Intelligence, 1983-1993 (History of Computing) Dependable Computing for Critical Applications 5 (Dependable Computing and Fault-Tolerant Systems) Wireless Computing in Medicine: From Nano to Cloud with Ethical and Legal Implications (Nature-Inspired Computing Series) Introduction to Evolutionary Computing (Natural Computing Series)