Book Reviews
A Door to State-of-the-Art High-Performance Computing

Arturo Ortiz-Tapia • Mexican Petroleum Institute

High-Performance Computing: Paradigm and Infrastructure
Laurence T. Yang and Minyi Guo, eds.
778 pages
US$125.00
Wiley-Interscience, 2006
ISBN: 978-0-471-65471-1

In High-Performance Computing: Paradigm and Infrastructure, editors Laurence T. Yang and Minyi Guo accomplish quite a feat: bringing together numerous experts to describe the state of the art in high-performance computing (HPC) software and hardware. They gathered 41 chapters by 127 contributors (including themselves). Although they claim that the book isn’t meant to be exhaustive, most professionals will likely find something to further their research or studies, either in the book or aided by the extensive bibliography at the end of each chapter.

The book has seven parts: Programming Models, Architectural and System Support, Scheduling and Resource Management, Clusters and Grid Computing, Peer-to-Peer (P2P) Computing, Wireless and Mobile Computing, and High-Performance Applications. Each contains several chapters written in the form of a scientific paper. About 16 chapters refer to hardware (processors, memory, topology, communications between nodes, and so on), and 25 focus on software (such as algorithms and programming languages). However, because many computer algorithms are platform dependent, many chapters address both issues, so the book devotes approximately equal space to each.

The book’s main audience is applied mathematicians, computer scientists with a solid mathematical background, researchers, IT professionals, and advanced computer science students. (It doesn’t directly address “P = NP?” but does include a lot about complexity theory—both P and NP—and estimating the order of calculations.) Some of the book’s mathematical elements are accessible to anyone with elementary skills for reading mathematical formulas or set theory. Others are unreadable if you don’t have some advanced skills in, say, graph theory, topology, and reading proofs. The historical parts are interesting and give you a sense of what has worked in algorithmic and hardware problem-solving and what is still a question mark (whether or not you can expect something to become a standard). The example programs are written in several languages (F90, High-Performance Fortran, C/C++, and Java), many with OpenMP, Parallel Virtual Machine, or Message Passing Interface as their parallelizing interfaces. Enlightening diagrams illustrate many of the systems’ architectures, and in general all the computer experiments’ results have corresponding charts with straightforward interpretations.

The book tests several famous computing-intensive problems, including the n-body problem, plasma turbulence, the n-queen problem, the traveling-salesman problem, the recursive Fibonacci algorithm, and the (relatively new) Smarandache function. These tests compare different algorithms, architectures, or platforms to provide a better decision basis for improvements, modifications, or scalability.
Many of the grid and distributed-computing projects the book mentions (such as SETI@home) seem embarrassingly parallel in the sense that, in principle, no special effort is necessary to segment the task into a very large set of parallel tasks. Without any communication between nodes, however, those projects do raise important issues, such as bandwidth, task scheduling, node heterogeneity, failure management, reliability, application composition, and security. The book covers these issues, but not all at the same depth.

Yang and Guo claim that supplemental material is available at the publisher’s FTP site. However, I couldn’t find any when I wrote this review, not even by going to the parent directory or trying different word endings or combinations of underscores and dashes. The Web bibliography at the end of each chapter is a slightly different story; some of the Web pages are still available, others aren’t.

Despite the book’s heterogeneity of contributors and subjects, it offers a consistent, integrated view of the HPC state of the art. It provides advanced students with a panoramic view of HPC subareas they could study (with help from a good teacher and other textbooks related to F90, MPI, Java, microprocessor electronics, or whatever particulars they’re interested in). This book is hardly a stand-alone, though, and if I were a student, I would hesitate to buy it. For researchers or professionals who are already experts in their fields, however, it could become an invaluable compendium and is worth its price.

Arturo Ortiz-Tapia is a scientific researcher at the Mexican Petroleum Institute. Contact him at aortitz@imp.mx.

Related Links

- DS Online's Grid Computing Community (http://dsonline.computer.org/portal/site/dsonline/)

Cite this article: