GUEST EDITORS' INTRODUCTION

the Top 10 Algorithms

In putting together this issue of *Computing in Science & Engineering*, we knew three things: it would be difficult to list just 10 algorithms; it would be fun to assemble the authors and read their papers; and, whatever we came up with in the end, it would be controversial. We tried to assemble the 10 algorithms with the greatest influence on the development and practice of science and engineering in the 20th century. Following is our list (here, the list is in chronological order; however, the articles appear in no particular order):

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection
- Fast Multipole Method

With each of these algorithms or approaches, there is a person or group receiving credit for inventing or discovering the method. Of course, the reality is that there is generally a culmination of ideas that leads to a method. In some cases, we chose authors who had a hand in developing the algorithm, and in other cases, the author is a leading authority.

1521-9615/00/$10.00 © 2000 IEEE

Jack Dongarra
University of Tennessee and Oak Ridge National Laboratory
Francis Sullivan
IDA Center for Computing Sciences

In this issue

Monte Carlo methods are powerful tools for evaluating the properties of complex, many-body systems, as well as nondeterministic processes. Isabel Beichl and Francis Sullivan describe the Metropolis Algorithm. We are often confronted with problems that have an enormous number of dimensions or a process that involves a path with many possible branch points, each of which is governed by some fundamental probability of occurence. The solutions are not exact in a rigorous way, because we randomly sample the problem. However, it is possible to achieve nearly exact results using a relatively small number of samples compared to the problem's dimensions. Indeed, Monte Carlo methods are the only practical choice for evaluating problems of high dimensions.

John Nash describes the Simplex method for solving linear programming problems. (The use of the word *programming* here really refers to scheduling or planning—and not in the way that we tell a computer what must be done.) The Simplex method relies on noticing that the objective function's maximum must occur on a corner of the space bounded by the constraints of the "feasible region."

Large-scale problems in engineering and science often require solution of sparse linear algebra problems, such as systems of equations. The importance of iterative algorithms in linear algebra stems from the simple fact that a direct approach will require $O(N^3)$ work. The Krylov subspace iteration methods have led to a major change in how users deal with large, sparse, nonsymmetric matrix problems. In this article, Henk van der Vorst describes the state of the art in terms of

methods for this problem.

Introducing the decompositional approach to matrix computations revolutionized the field. G.W. Stewart describes the history leading up to the decompositional approach and presents a brief tour of the six central decompositions that have evolved and are in use today in many areas of scientific computation.

David Padua argues that the Fortran I compiler, with its parsing, analysis, and code-optimization techniques, qualifies as one of the top 10 "algorithms." The article describes the language, compiler, and optimization techniques that the first compiler had.

The QR Algorithm for computing eigenvalues of a matrix has transformed the approach to computing the spectrum of a matrix. Beresford Parlett takes us through the history of early eigenvalue computations and the discovery of the family of algorithms referred to as the QR Algorithm.

Sorting is a central problem in many areas of computing so it is no surprise to see an approach to solving the problem as one of the top 10. Joseph JaJa describes Quicksort as one of the best practical sorting algorithm for general inputs. In addition, its complexity analysis and its structure have been a rich source of inspiration for developing general algorithm techniques for various applications.

Daniel Rockmore describes the FFT as an algorithm "the whole family can use." The FFT is perhaps the most ubiquitous algorithm in use today to analyze and manipulate digital or discrete data. The FFT takes the operation count for discrete Fourier transform from $O(N^2)$ to $O(N \log N)$.

Some recently discovered integer relation detection algorithms have become a centerpiece of the emerging discipline of "experimental mathematics"—the use of modern computer technology as an exploratory tool in mathematical research. David Bailey describes the integer relation problem: given n real numbers $x_1, \ldots, x_n$, find the n integers $a_1, \ldots, a_n$ (if they exist) such that $a_1x_1 + \ldots + a_nx_n = 0$. Originally, the algorithm was used to find the coefficients of the minimal integer polynomial an algebraic number satisfied. However, more recently, researchers have used them to discover unknown mathematical identities, as well as to identify some constants that arise in quantum field theory in terms of mathematical constants.

The Fast Multipole Algorithm was developed originally to calculate gravitational and electrostatic potentials. The method utilizes techniques to quickly compute and combine the pair-wise approximation in $O(N)$ operations. This has led to a significant reduction in the computational complexity from $O(N^2)$ to $O(N \log N)$ to $O(N)$ in certain important cases. John Board and Klaus Schulten describe the approach and its importance in the field.

Your thoughts?

We have had fun putting together this issue, and we assume that some of you will have strong feelings about our selection. Please let us know what you think.

Jack Dongarra is a professor of computer science in the Computer Science Department at the University of Tennessee and a scientist in the mathematical science section of Oak Ridge National Lab. He received his BS in mathematics from Chicago State University, his MS in computer science from the Illinois Institute of Technology, and his PhD in applied mathematics from the University of New Mexico. Contact him at dongarra@cs.utk.edu; www.cs.utk.edu/~dongarra.

Francis Sullivan's biography appears in his article on page 69.