Parallel algorithms at Queen's University, Ontario

Why the NAG Library for SMP and multicore?

The numerical routines in the NAG Library for SMP & multicore are available for use in various groups who have access to the High Performance Computing Virtual Laboratory (HPCVL) in Ontario, Canada. Economists, studying the stock market, can use the routines to model the global effects of capital flow. Atmospheric scientists can use them to analyze global weather patterns. Psychologists can use them to model the brain and human memory. Biotechnologists and computational chemists can use them to design complex molecules for innovative new drugs. Engineers can use them to model blood flow in artificial hearts. All these types of research require the analysis of terabytes of data and intricate mathematical calculations that benefit from well chosen parallel numerical routines.

Introduction to multicore processing

Now that multicore computers have become so pervasive, the way that numerical code is written has changed. Whilst it is possible to run multiple copies of serial code on multicore resources, this is inefficient and doesn't improve time to solution for a single simulation. It has become important to tune code and algorithms and/or select other variations of algorithms and/or look to quite different numerical approaches that are better suited to multicore usage.

Programming APIs, such as OpenMP (a portable model for developing parallel applications on platforms from desktops to supercomputers with memory shared between cores), are used to manage the code across the various cores and memory configurations of a modern computer platform.

The NAG library for SMP and multicore helps with this parallel programming by making full use of OpenMP and proving well documented access to well designed parallel numerical routines.

The broad scale of multiple core use

The issue of programming for multiple core computers extends right down to the desktop. On smaller machines the absolute performance gains, which can be achieved through the use of parallel routines, are not as large as the gains on larger clusters. However, similar relative improvements can be made. This is important because even laptops have multicore processing engines and if you don't consider parallel processing you are not making full use of these resources. In addition, modern distributed memory clusters have multicore nodes and are best used through a combination of standard parallel programming interfaces such as the Message Passing Interface (MPI), and intra-node multithreading. This is greatly facilitated when the libraries in use are already shared memory parallel, as is the case for the NAG Library.

Examples of areas of use

The types of codes and applications that benefit from parallelism in NAG routines are those that make use of some of a wide range of underlying numerical functions in the areas of: Dense and Sparse Linear Algebra; FFTs; Random Number Generators; Quadrature; Partial Differential Equations; Interpolation; Curve and Surface Fitting; Correlation and Regression Analysis; Multivariate Methods; Time Series Analysis; Financial Option Pricing and more.

Multicore drives many vendors' systems

IT vendors across the board, from the leading chip fabricators to the large 'supercomputer' vendors, now offer multicore designs as the major part of their product ranges. Multicore coding approaches are efficient for midsize Intel or AMD based computers and for top end Sun, Cray, HP or IBM machines.

Parallel routines at Queen's University

At Queen's University researchers and programmers have access to both serial and parallel versions of the NAG routines for their coding. *'The NAG Library is very good for work on multiple cores'* says Dr. Hartmut Schmider of the computational support team for HPCVL. *'This is, of course, because of the reliable parallel design of the algorithms. But it is also because of the common interface for both serial and parallel libraries. This enables users to speed up their code on many multiple core architectures with greatly reduced effort. The library design makes for a very time efficient way to gain considerable speed advantages when addressing hard problems'.*

The future and other parallel programming

The Numerical Algorithms Group continues to advance the development of the NAG Library with a particular focus on algorithms that are most appropriate for parallel work using both multicore and multiprocessor architectures. Additionally NAG have active research programs looking to next generation process technologies such as general purpose Graphic processors (GPGPU).