Crisis in High Performance Computing – A Workshop

11th September 1995

Lecture room G22 (also known as the Pearson Lecture Theatre)
Pearson Building
University College London
Gower Street
London WC1E 6BT

Background

State-of-the-art high performance computers are turning in what some observers consider worryingly low performance figures for many user applications. How widespread are such feelings, how justified are they and, if they prove to be justified, what implications do they hold for the future of High Performance Computing (HPC)?

Efficiency levels for *"real"* HPC applications are reported (e.g. by the NAS parallel benchmarks) ranging around 20-30% (for some 16-node systems) to 10-20% (for 1024-node massively parallel super-computers). Are low efficiencies the result of bad engineering at the application level (which can be remedied by education) or bad engineering at the architecture level (which can be remedied by <*what*>)? Maybe these efficiency levels are acceptable to users ... after all, 20% of 16 nodes (rated at 160 MFLOPS per node) is still around 500 Mflops and 10% of 1024 nodes is 16 Gflops? But they may be disappointing to those who thought they were going to be able to turn round jobs at over 100 Gflops! Are there other ways of obtaining the current levels of performance that are more cost-effective?

A further cause of concern is the dwindling number of suppliers of HPC technology that are still in the market ...

This workshop will focus on the technical and educational problems that underly this growing crisis. Political matters will not be considered ... unless they can be shown to have a direct bearing.

Participants

- potential users of HPC facilities ("what problems am I going to face ... will it be worth my while?");
- current users of HPC facilities ("what performance am I getting ... how hard has it been to achieve this ... am I getting value for the time I have invested?");
- non-users of HPC facilities ("what effect has the funding of large scale super-computers had on my ability to obtain smaller scale facilities locally - preferably on my desk?");
- architects of HPC facilities ("how can decent efficiency levels be achieved and how can application design-implementation-tune- test-and-maintain be made simple?").

Organisers

The London and South-East consortium for education and training in High-Performance Computing (SEL-HPC). SEL-HPC comprises ULCC, QMW (and the other London Parallel Application Centre colleges - UCL, Imperial College and the City University), the University of Greenwich and the University of Kent.

Timetable

09:30	Registration
09:50	Introduction to the Day
10:00	*High performance compute + interconnect is not enough* (Professor David May, University of Bristol)
10:40	*Experiences with the Cray T3D, PowerGC, . . .* (Chris Jones, British Aerospace, Warton)
11:05	*More experiences with the Cray T3D, . . .* (Ian Turton, Centre for Computational Geography, University of Leeds)
11:30	Coffee
11:50	*Experiences with the Meiko-CS2, . . .* (Chris Booth, Parallel Processing Section, DRA Malvern)
12:15	*Problems of Parallelisation - why the pain?* (Dr. Steve Johnson, University of Greenwich)
13:00	Working Lunch (provided) [Separate discussion groups]
14:20	*Language Problems and High Performance Computing* (Nick Maclaren, University of Cambridge Computer Laboratory)
14:50	*Parallel software and parallel hardware - bridging the gap* (Professor Peter Welch, University of Kent)
15:30	Work sessions and Tea [Separate discussion groups]
16:30	Plenary discussion session
16:55	Summary
17:00	Close

Registration Details

For full workshop details or to register contact:

Judith Broom
Computing Laboratory
The University
Canterbury
Kent
CT2 7NF
England

tel: +44 1227 827695
fax: +44 1227 762811
email: `J.Broom@ukc.ac.uk`

Latest information and electronic registration can be found at:

<URL:http://www.hensa.ac.uk/parallel/groups/selhpc/crisis/>

and:

<URL:ftp://unix.hensa.ac.uk/pub/parallel/groups/selhpc/crisis/>

where full details of this workshop (e.g. names of speakers and final timetable) will be updated.

All types of participant are welcome – see above. If you are a current HPC user and are willing to contribute by speaking in one of the 'Experiences' sessions, please email a short (one page) position statement to J.Broom@ukc.ac.uk with the word 'Experiences' in the title line.

Position statements are also welcome, but not compulsory, from all attending this workshop. They will be reproduced for all who attend and will help us define the scope of each discussion group.

Extended Abstract

Problems with efficiency?

Efficiency levels on massively parallel super-computers have been reported (e.g. in the NAS Parallel Benchmarks Results 3-95, Technical Report NAS-95-011, NASA Ames Research Center, April 1995) ranging from 50% for the "embarrassingly parallel benchmarks", through 20% for tuned "real" applications, past 10% for typical "irregular" applications and down to 3% when using a portable software environment. Low efficiencies apply not only to the larger system configurations (256 or 1024 nodes), but also to the smaller ones (e.g. 16 nodes). Seven years ago, we would be disappointed with efficiency levels below 70% for any style of application on the then state-of-the-art parallel super-computers. What has caused this regression and can it be remedied?

It seems to be proving difficult to build efficient high-performance computer systems simply by taking very fast processors and joining them together with very high bandwidth interconnect. Apart from the need to keep the computational and communication power in balance, it may also be essential to reduce communication start-up costs (in line with increasing bandwidth) and to reduce process context-switch time (in line with increasing computational power). Failure in either of these regards leads to coarse-grained parallelism, which may result in insufficient parallel slackness to allow efficient use of individual processing nodes, potentially serious cache-coherency problems for super-computing applications and unnecessarily large worst-case latency guarantees for real-time applications.

Suppliers?

A further cause of concern is the dwindling number of suppliers of HPC technology that are still in the market. Will there be a next generation of super-computers from the traditional sources? Or will HPC users have to rely on products from the commercial marketplace, in particular the PC Industry and Games/Consumer-Products Industries? If the latter, how will this change the way we approach the design of HPC facilities and applications?

Clusters?

At the other end of the spectrum, clusters of workstations are reported as offering, potentially, good value for money, but only for certain types of application (e.g. those with very high compute/communicate ratios). What are those threshold ratios and how do we tell if our application is above them? What do we do if our application does not so conform?

Software tools, languages and paradigms?

Blame is often laid at the lack of software tools to support and develop applications for high performance architectures. New standards have been introduced for parallel computing - in particular, High Performance FORTRAN (HPF) and the Message Passing Interface (MPI). Old standards stick around - e.g. the Parallel Virtual Machine (PVM).

These standards raise two problems: depressed levels of efficiency (this **may** be a temporary reflection of early implementations) and a low-level hardware-oriented programming model (HPF expects the world to be an array and processing architectures to be a regular grid, MPI allows a free-wheeling view of message-passing that is non-deterministic by default). Neither standard allows the application developer to design and implement systems in terms dictated by the application; bridging the gap between the application and these hardware-oriented tools remains a serious problem.

New pretenders, based upon solid mathematical theory and analysis, are knocking on the door - such as Bulk Synchronous Parallelism (BSP). Old pretenders, also based upon solid mathematical theory and analysis and with a decade of industrial application, lie largely unused and under-developed for large-scale HPC - such as occam. Might either of these offer some pointers to the future?

Goals

The above paragraphs raise several issues. The aim of this workshop is to exercise and debate them thoroughly, see what peoples' real experiences have been and consider in what ways HPC needs to mature in order to become viable. A major goal of the workshop is to start to try to identify standards of "good' behaviour' on software for parallel or distributed systems that will:

- enable HPC hardware architectures to operate with much greater efficiency levels;
- enable HPC applications to be developed in their own terms without regard for the underlying hardware.

Or maybe the workshop will decide that:

- HPC architectures (hardware and software) do not have fundamental problems;
- there are no lessons from the past that need re-discovery and re-application;
- everything can be sorted out by better education and tools for existing HPC standards.

Please come along and make this workshop work.