— Peripherals – the quality and range of peripherals is still much better on the central computers.

As this argument continues there is a movement to let you have your cake and eat it too! Good networking to heterogeneous systems is at last within sight as *de facto* standards, such as NFS (Network File System for distributed file access), X-Windows as a windowing standard and Postscript as a standard output language for graphics, are being rapidly adopted by many suppliers. If all goes has planned —

File systems will be accessed using NFS. Applications will talk to windowing systems on workstations using X-Windows. All printers will accept Postscript format to print both text and graphics enabling users to print files wherever they want.

The progress in this area has been due to the various manufacturers coming to a consensus of opinion on the directions to follow. No longer do they wait for the standards organisations, they prefer to follow *de facto* but functional standards now.

Finally the debate is taking on larger proportions as the personal workstation manufacturers are announcing new machines of 10, 20, 30 and even 40 Mips of power. They are incorporating new RISC (Reduced Instruction Set Computers) processor chips with 10 Mips power each into multi-processor architectures. We are certainly looking at an exciting future.

Trends in Supercomputers

R. Gruber, Lausanne

(GASOV-EPFL)

The title of this report could also have been "Trends in Supercomputing". The reason is that powerful parallel vector computers can deliver a high computational performance only if the application software is adapted to their architectures. It becomes more and more evident that to solve the biggest problems, the software must be adapted to the computer architecture and *vice-versa*.

Present Situation

Supercomputer is a designation given to about 300 computers installed worldwide with a peak computing power of over 100 megaflops (million floating point operations per second). These machines have mainly been used for numerical experimentation in various scientific domains such as fluid dynamics, structural mechanics, seismic explorations, reservoir modelling, quantum mechanics, plasma physics, materials science. It is also believed that portfolio analysis and financial transaction business in banking will, in the near future, be executed on computers having many powerful processors and memories of the order of a few gigabytes.

Specific characteristics of supercomputers are high peak computing power achieved by very rapid clock periods (4.1 ns for a CRAY 2), parallel processors (65 536 in the highly parallel Connection Machine), pipeline architectures (instructions are divided into subinstructions which can all be in execution at the same time, so working like an assembly line) leading to up to two operations (1 add + 1 multiply) per clock period, large memories (more than 2 gigabytes for CRAY 2 and ETA 10) and very rapid connections to the outside world. The peak power and the memory space can be up to three orders of magnitude bigger than for workstations. The major flaws of these computers are that their operating systems and their mode of access do not yet match modern standards. In addition, to benefit most from the high computing power, it is necessary to formulate an application in such a way that at least 90% of all the computations are executed as vector operations. Unfortunately, this is not yet the case for most of the applications now running on supercomputers. This further implies that non-vectorizable organizatory work such as editing, interactive graphics and documentation, data handling and communication business should be taken care of by a user friendly, highly interactive personal workstation directly linked to the supercomputer via a high speed fibre optic network. At the moment, however, most of the supercomputers are accessible only through mainframes, thus reducing their attractiveness.

Scientist' Expectations

A scientist or engineer would like to be able to solve numerically the most realistic physical model described, for instance, by a set of partial differential equations (PDE). As an example, one would like to compute the time evolutionary solution of six coupled nonlinear PDE's in three dimensions. For a discrete approach (finite elements, finite differences or finite volumes) a discretization of 100 × 100 × 100 intervals is today considered necessary to obtain physically relevant results. To reduce the number of time steps, implicit methods should be used; hence the necessity of very efficient iterative solvers based upon the methods of multigrid or conjugent-gradient with preconditioning. To make the best use of the parallel architectures of future supercomputers, algorithms and programming techniques leading to the definition of codes with coarse parallel granularity should be adopted.

High parallelization at the subroutine level can be obtained by a decomposition of the geometrical domains into subdomains, each subdomain being assigned to one processor. To handle the connectivities between subdomains, fast networks built around high speed buses, global memories or direct connections of nearest processors are needed. An estimate for three-dimensional simulation programs shows the necessity for an effective computing power of 10 000 Mflops (then one run does not take more than one hour CPU time) and a memory space of eight gigabytes (all the

matrix elements are in memory, thus excluding input/output operations responsible for reducing turn-around time and the memory occupancy).

This expectation of what one can call a "realistic case" is the current estimate of what one will be able to compute in the next five to ten years. However, if more physics is included in the model, *i.e.* more physical quantities influence the result, the number of PDE's will increase. The solution then will show more details and, as a consequence, the number of time steps will increase. In addition, fine structures of the result will need finer spatial resolution. In the end one will require still more powerful computers. We hope that they will exist. What will they look like? Let us try to answer this question.

Future Supercomputers

The performance of today's supercomputers can be increased in different ways. They are:

(a) Reduction of the cycle time

Within 4.1 ns up to two floating point operations can be computed per processor in a CRAY 2. During the same time an electrical and a light signal travel 1.2 m in a cable. This means that the physical size of the computer starts to impose limits on the cycle times. Reductions of the cycle times have been realized by replacing silicon by a GaAs technology (CRAY 3 ready in 1989) or by cooling the electronic components to liquid nitrogen temperature (ETA 10 already available). For the near future, it is believed that the discovery of high temperature superconductors will lead to high speed circuits based upon the Josephson effect. These circuits will have cycle times far below 1 ns. One can expect that by the end of the century, circuits in the 100 ps (10^{-10} s) range will be brought to the market. Thus in the next 30 years one can expect that the cycle time will be reduced by another factor of 100.

(b) Multipipelining and functional units

Before the advent of the CDC 6600 in the sixties, operations were performed strictly sequentially. At a given time, only one operation or one instruction interpretation step was in execution. The CDC 6600 introduced stacking. Not only does the interpretation of one instruction work together with an executable step, but it is also possible to activate all the processing units at once. This way of making better use of the available hardware was the first step towards pipelining. The step was further advanced on the CRAY 1. In this machine, the fundamental operations such as add or multiply are sliced into suboperations which can all be active when working on a vector instruction. The load/store unit, the branching processor, the scalar unit and the instruction interpreter can be simultaneously active as well.

A new type of computer (TRACE 7 of Multiflow) has recently appeared on the market and another has been announced (CHoPP). In these computers one tries to make maximal use of the pipelining concept by combining a set of instructions into one super instruction. The format of these long super instructions offers a multiple compute and store and branch pipelined sequence every cycle. In order to avoid memory conflicts the LOAD operation is executed in broadcast mode. In the TRACE computer the most probable path is followed in the case of a branch. The compiler provides for compensation code to be able to undo the wrong path and to take the right one afterwards.

(c) Parallelization

Parallelization is another way of increasing computing power. Different types of parallel architectures are proposed. In the synchronous approach all the processors perform the same instruction at a given time, whereas in the asynchronous one, each processor executes its own program. The biggest problem in these parallel computers is the intercommunication between processors. This can be done by direct local connections as in hypercubes (where each processor has its own local memory) or by a global memory as in the CRAY's. In the ETA 10 architecture each processor has its own local memory but communicates with the others through a global memory.

Parallelization can be obtained by splitting a program into a number of independent tasks, where each one is to be executed at the same time by a different processor (this mode is called multitasking) or by decomposing a *DO loop* into small tasks (called microtasking). The first case must be programmed by the user, whereas the latter is handled by the compiler. One could imagine that in future architectures, the tasks will be given to different types of processor, each one optimal for the task. Vector code would automatically be sent to the special vector processor, scalar code to the special scalar part, graphics to graphics machines *etc.* This would guarantee a better use of the computer hardware.

(d) Software improvement

One of the major causes of inefficiency in supercomputers is that the software is not optimized for the specific architecture of the machine. Users usually would like to continue using an old code developed on conventional computers. These sequential codes are often not vectorizable since logical paths are followed step by step where one step consists of one single operation. This has been done to minimize the number of floating point operations. Such an organization is opposed to the concept of high level parallelization and medium level vectorization. In addition, a compiler often cannot improve the efficiency of such codes. Substantial parts of a program remain unoptimized.

The compilers (FORTRAN, PASCAL, C) available at the moment are not always at the level the user would like them to be. At best the innermost loops can be optimized. Many memory transfers are necessary and, as a consequence, the maximum speed is relatively low. More sophisticated compilers and better designed languages will hopefully improve this situation.

It is believed that in the near future the so called "Cancer Codes" which grew in a more or less uncontrolled way to monsters of several hundred thousand lines of code will be rewritten to benefit from parallel vector architectures and from huge central memories.

Outlook

One can see that there are opportunities to improve the computational power of present supercomputers by many orders of magnitude. The ETA 10 computer will soon outperform a CRAY 1 by a factor of 30 and by 1989, CRAY 3 and NEC SX3 computers will be 100 times faster. These machines incorporate more and more parallelism and offer memories of up to 8 Gigabytes (CRAY 3).

The announcement by Seymour Cray of 64-processor machines for the early nineties shows that the classical supercomputer manufacturers' trend is towards high parallelism. They will in the future benefit from the developments now under way on highly parallel architectures. With these interesting experimental machines one can study problems of organization of the data flow either through a global memory or by an interprocessor communication system. However, a pragmatic user of supercomputers is still reluctant to invest too much of his time to adapt a program to an experimental architecture.