

LA-8609-MS

Dr. 2233

R-1452

①

183
1/27/81
T.S.

**DOE Research in
Utilization of High-Performance Computers**

MASTER

University of California



LOS ALAMOS SCIENTIFIC LABORATORY

Post Office Box 1663 Los Alamos, New Mexico 87545

DOE Research in Utilization of High-Performance Computers

B. L. Buzbee
W. Jack Worlton
G. Michael*
G. Rodrigue*

DISCLAIMER

This book was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

*Lawrence Livermore Laboratory, Livermore, CA 94550.



DOE RESEARCH IN
UTILIZATION OF HIGH-PERFORMANCE COMPUTERS

by

B. L. Buzbee
W. Jack Worlton
G. Michael
G. Rodrigue

ABSTRACT

Department of Energy (DOE) and other Government research laboratories depend on high-performance computer systems to accomplish their programmatic goals. As the most powerful computer systems become available, they are acquired by these laboratories so that advances can be made in their disciplines. These advances are often the result of added sophistication to numerical models whose execution is made possible by high-performance computer systems. However, high-performance computer systems have become increasingly complex, and consequently it has become increasingly difficult to realize their potential performance. The result is a need for research on issues related to the utilization of these systems. This report gives a brief description of high-performance computers, and then addresses (1) the use of and future needs for high-performance computers within DOE, (2) the growing complexity of applications within DOE and (3) areas of high-performance computer systems warranting research.

I. INTRODUCTION

A. Definition

High-performance computers are the most powerful large-scale computers available at any given time. Here, powerful means how fast the computer can execute operations like addition, multiplication, subtraction, and division; how much storage is available; and how precisely the computer calculates (the number of digits used).

B. Value

High-performance computers are valuable scientific research tools--so valuable that it is against national policy to export them to potential adversaries of the United States. Their value emanates from their ability to increase our knowledge, their ability to free us from constraints of reality, and from their cost-effectiveness. They increase our knowledge by

- enabling us to treat complexity that is not otherwise tractable, for example, nonlinear evolutionary flows;
- enabling us to study parameters where laboratory measurement is difficult, if not impossible, for example, fusion processes;
- enabling us to explore many alternatives, thereby guiding our intuition.

One of the greatest values of computers is that their use frees us from some constraints of reality. However, other constraints come into play. For example, detonation of nuclear weapons is tightly controlled by various political considerations, but computer simulation of them is not subject to such constraints. Similarly, reactor safety experiments are subject to environmental constraints, but on the computer we can repeatedly simulate melt down, etc. In such cases, computer simulation frees us from the necessity of assembling the requisite skills and materials to physically construct the associated experiments.

Among the cost-effective benefits high-performance computers provide are

- reduced need for, and increased gains from, experimental tests;
- increased efficiency and effectiveness of new energy and weapons systems;
- improved ability to meet schedules; and,
- shortened development periods for new technology.

The value of a high-performance computer far exceeds its cost, which is small compared to the total costs of the energy and weapons systems under development and the experimental research involved in these programs. For example, the computing cost for a single weapon design is less than 1% of the total system cost [1]. Such a tool warrants research on improving its productivity and on easing its utilization.

C. Usage and Future Needs

The U.S. Department of Energy (DOE) conducts research in two major areas: advanced energy technologies and national security.

Discussed below are specific DOE programs for which high-performance computers are invaluable and for which additional computer capability is needed.

1. National Security. For most research and development, neither computation nor experimentation alone is adequate. These two methods complement each other: The computations predict how a system will perform; the experiments supply physical data and verify the computations.

The nuclear weapons program, however, is exceptionally dependent upon computer modeling because of the political, economic, and technical limitations on nuclear testing. Only by using computer models to conduct hundreds of simulations of nuclear weapon performance can weapons designers ensure that our nuclear weapons will be safe, secure, and reliable.

Reducing the need for field testing by increasing computer power also saves a lot of money. For example, a nuclear weapon designed using the CDC 6600, which has about one-fourth the speed of the 7600, required 23 field tests; whereas a similar, later weapon designed using the 7600, required only six field tests. (As a general rule, the faster the computer, the more accurate the model, because a faster, more powerful computer can handle more elaborate equations.) The elimination of 17 tests at a typical cost of \$5 million per test saved about \$85 million, and this figure does not include the accompanying savings in calendar time.

The role of computers in nuclear weapons design is discussed in depth in Ref. 1. Among the conclusions drawn are:

- The design of a single nuclear weapon may require the equivalent of one year of computer time on the most powerful machines available.
- Design costs are a small fraction of total system cost.
- The judicious use of design tools, especially computer models, can have a large impact--leverage--on total weapons system cost.
- Because it is essentially impossible to obtain statistically significant test data, our confidence in the performance of nuclear devices is derived in large measure from the substantial amount of computation that supports every design.
- The lack of better computers will hamper progress in nuclear weapons technology.

2. Laser Fusion. The ultimate success of laser-initiated thermonuclear fusion as a power source depends strongly upon computer simulations of target behavior, because the associated physical processes are not entirely achievable in experiments. Current computer programs for designing targets use the fastest available computers for up to 20 hours per run, yet even more comprehensive computational studies must be done before laser fusion experimental facilities, costing between \$50 million and \$200 million, can be built.

3. Magnetic Fusion. Scientists and engineers are also using high-performance computers to study magnetic confinement systems, which may ultimately lead to the development of fusion power reactors. High-performance computers allow scientists to model "magnetic bottles" for data on plasma reactions, as well as the movement of individual particles in the magnetic

ield, thereby studying ways of managing the plasma so that fusion occurs. High-performance computers also enable sophisticated design studies of entire power plants, whereas smaller computers can model only parts of such systems.

The role of computers in magnetic fusion energy is discussed in Ref. 2. Among the conclusions drawn are:

- "---it is not yet possible to simulate the nonlinear development of high-shear internal MHD modes because of resolution requirements. Adequate resolution of such instabilities (which are believed to be the most relevant experimentally) will require an order of magnitude more in computer capability" [2, p. 50].
- "It is currently not possible to adequately simulate electromagnetic instabilities such as the drift-Alfven or the drift-tearing modes due to the faster time scales associated with electron motion and to the fact that an electromagnetic model requires much more memory. Such simulations will require a more powerful computer" [2, p. 51].

4. Reactor Safety. What would be the consequences of a nuclear reactor accident? This question is a governing issue in the design and licensing of nuclear power plants. It is impractical to perform large numbers of experiments with actual reactors. Instead, reactor behavior during hypothetical accidents, including the adequacy of emergency safety systems, is evaluated by computer simulation. These programs are at the limit of current computing capabilities; a single calculation can take up to 20 hours on a CDC 7600-class computer. For example, the nuclear regulatory commission would have preferred to have analyzed the Three Mile Island nuclear reactor accident with numerical simulation and with many different assumptions. To have done so would have tied up as much as 15% of the scientific computers in the United States [3].

II. BACKGROUND

In the 20 years of high-performance computing, the usage characteristics are well established. First, there is a limit (less than 20 hours) to the time a user will invest in a particular run. Second, speed increases are used to include more detail or complexity into numerical simulations rather than to reduce computing time. This explains DOE's history of continually procuring the biggest and fastest computers available. In view of the needs discussed in Section I.C, this trend will continue.

A. Few Vendors

There are relatively few high-performance computers in use--probably not more than 100 in the world. The total computer population by comparison is on the order of 1,000,000.

Only a few companies manufacture high-performance computers because it takes special skills to build an effective system. In fact several manufacturers have abandoned the high-speed market after unsuccessful attempts to

penetrate it. Most computer manufacturers follow the components improvement-versus-costs curve, turning out ordinary computers for the huge number of users who do not (yet) need high-speed service.

Because there are few manufacturers of high-performance systems, it is in the interest of the DOE to provide them with detailed information on its applications. This document proposes activities that will move us in that direction.

B. Complexity in Models

Growing complexity is characteristic of science and technology; an obvious example is the contrast between the Wright brothers' airplane and a supersonic jetliner. In the same manner, weapons and energy technologies are also becoming more complicated; and as system complexity grows, more technical and economic parameters must be studied.

To study these complex alternatives, improved computer models include the following features:

- more complete and accurate representation of the laws of physics;
- more two- and three-dimensional simulations in lieu of simpler one-dimensional simulations;
- smaller spatial zones, so that the complexity of the system or device can be more accurately represented; and
- smaller time steps, so that rapidly changing physical processes can be more thoroughly tracked.

Here is a simple example from the physics of fluids. A heavily used DOE design program currently is expressed as a 100 x 100 zone model. This, incidentally, is not rated as a fine detail model. Depending on the physical properties being studied, between 25 and 200 quantities per zone are computed, and typically 1,000 cycles of the model's evolution are carried out. Thus, in the calculation space there are 100 x 100 x 1,000 zone cycles that must be done. This problem executes on the Cray-1 computer at the rate of about 15,000 zone cycles per minute, which means some 12 (fully utilized) hours of Cray-1 time are required.*

Now for this model, we have a genuine need to extend the physics to three dimensions. In its simplest manifestation, this requires 100 times the time cited above. Thus, what should be a simple extension is in truth a major hurdle, and, quite simply, it will not be done on currently available computers because users are unwilling to wait 200 hours for a result.

In addition to time demands, another fundamental limitation shows up when we work out the corresponding memory requirements. On the average, the two-

*The execution speed of this program on the CDC 7600 is about one-third of a Cray-1 or 5,000 zone cycles per minute.

dimensional model needs about 1,000,000 words for data storage. Thus, the "simplest" extension to three dimensions requires 100 million words. This necessitates use of secondary storage, probably disk. However, there is a growing disparity between arithmetic performance and the time required to transmit data from disk. For example, in the 70's the computer could typically perform about 40 arithmetic operations during the interval required to transmit one datum from disk. On the Cray-1 as many as 200 floating point operations can be performed during that interval, but few of our current simulation models require 200 operations per datum access. We must either accept the complexity of partitioning data sets across several disks in order to transmit more data per second, or develop solution techniques that reduce the number of data accesses by performing more operations per access.

Growing complexity in DOE models continues to necessitate procurement of computers with the highest levels of performance both in operations per second and in memory capacity. Providing this performance has and will continue to lead to exotic computer architectures. As we are about to see, computer architectures can have tremendous impact on algorithms and software.

C. Trends in Architecture

THE VENDORS SAY:

"Nobody, and I mean nobody, knows how to program large parallel machines."

- Seymour Cray
Business Week
12/6/76

"Many current large-scale programs cannot run efficiently on any known high performance machine..."

- David W. Hogan,
John C. Jensen, and
Merrill Cornish
Texas Instruments Inc.

USERS SAY:

"Converting problems for efficient execution on the new class of 'highly parallel' machines requires an almost complete reanalysis and rewrite of the program."

- J. E. Wirsching and
T. Kishi
Lawrence Livermore Laboratory

"Automatic vectorization is no substitute for a total rethinking of algorithms, mathematics, and programming techniques."

- M. Rowe
Langley Research Center
National Aeronautics and
Space Administration

In the early days of scientific computing, people were quite content to exploit the capabilities of whichever computer they were currently using. After all, any computer provided a tremendous speed increase over hand computation. Around 1960 the first transistorized computers "suddenly" appeared. On the average, they provided speed improvements of 6 to 10. In rapid succession speed increases totaling 100 were provided during the 60's. This lulled the user community, as well as many other people, into believing that improvements in computational speed were easy to make and likely to continue. In fact, the major portion of these speed increases was due to faster components and algorithms rather than to improved architecture. It took all of the 70's to achieve the next factor-of-10 speed increase. Further, specialized computer architecture was forcing users to tailor their programs to the architecture to even get close to maximum performance. The growth rate of computer performance, therefore, is diminishing despite increased complexity of computer architecture, and architectural complexity is influencing algorithms.

Figure 1 displays the execution bandwidth of high-performance computers over the past 25 years. The diminishing growth rate in performance is evident. In fact the data is nicely approximated by a double exponential with an asymptote of 10^9 operations per second.

Recent advances in Josephson junction and GaAs technology have created expectation that we may see a quantum jump in the performance of serial computers. However, estimates are that these computers will perform about 10^7 operations per second [4]. This is a factor of 10 improvement over the Cray-1, and as we have already noted, DOE needs at least a factor of 100.

Recent generations of supercomputers incorporate vector architecture. This architecture is impacting algorithms and software [4, 5]. In order for an algorithm to achieve high performance on these computers, it must fit the associated architecture [6]. For example, on the CDC STAR algorithms are sought that yield vectors whose elements can be stored contiguously and whose lengths are on the order of hundreds and preferably thousands. Determination of algorithms satisfying these conditions has yielded programs that perform two to four times faster than their CDC 7600 counterparts [7, 8, 9]. Straightforward implementation of "scalar algorithms" on the STAR results in performance that is substantially less than that of the CDC 7600. On the Cray-1 algorithms are sought that minimize data traffic between memory and the central processing unit, and that maximize parallel execution of the functional units. Determination of algorithms satisfying these conditions has yielded programs that perform 5 to 15 times faster than their CDC 7600 counterparts [10, 11, 12, 13]. Direct implementation of current algorithms on the Cray-1 yields a performance ratio of about 2.5 to 5.0 relative to the CDC 7600 [13, 14].

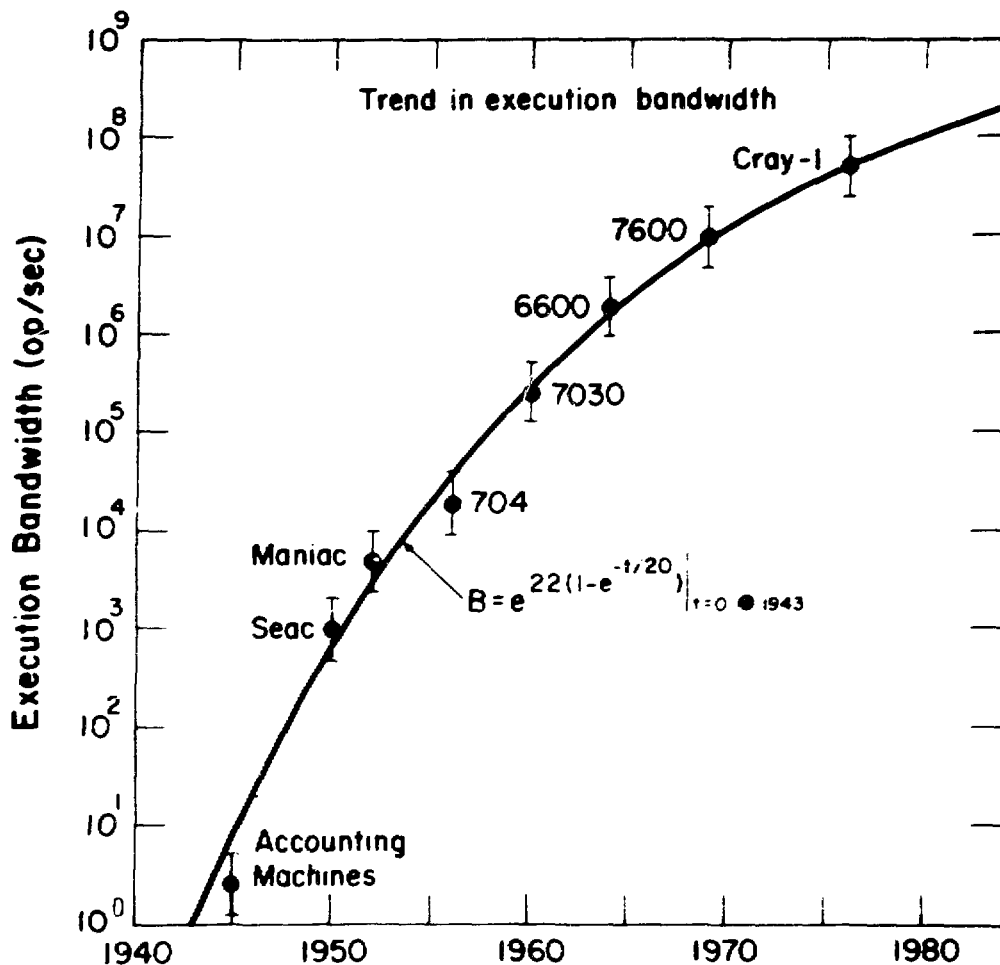


Fig. 1. Execution bandwidth of high-performance computers over the past 25 years.

Computers incorporating arrays of processors are beginning to emerge. Like vector computers, arrays of processors perform best with algorithms that have a minimal requirement for scalar computation. In addition, arrays of processors have limited data communication abilities between processors. Thus, when attempting to use them, one must seek algorithms that are highly parallel and with communication requirements that fit the architecture. The optimum arithmetic performance of these devices is potentially much greater than that of current supercomputers.

Because of the diminishing growth rate of performance in scalar processor architectures, we expect that the dominant architectural styles for future high-performance computers will include parallel operation or specialized pipeline structures. In both cases the programming tools and algorithms available to the user are inadequate and providing them is no mean task. It took between two and three years to adapt the programs developed for scalar

computers during the 60's and early 70's to the STAR architecture. During those three years, user energies were focused on efficiency factors and little or no new physics was inserted into their programs.

Because of the Cray-1's fast scalar capability, acceptable performance can be achieved on it with much less vectorization than on the STAR. This accounts for shorter conversion times to it. However, realizing the highest level of performance on the Cray-1 is just as challenging as on the STAR.

When performance increases are easy to achieve, the quality and diversity of the computational models improve. The user's creative energy, which is limited, is allowed to focus on the physics and not the computer. Today's user communities demand that programming tools and algorithms be fabricated along with the computers and that they be easy to use as well as expressive. In particular, few users are willing to abandon the higher level languages (HOL) in order to extract the maximum performance of today's computers. Apparently, most large physics codes have reached a level of complexity that is too difficult to treat with low-level programming languages. The irony of this situation is that HOLs generally do not compile maximally efficient programs. Factors of 5 of inefficiency are quite common [15]. The methodologies needed for future large-scale scientific computations are not now in place and, in fact, are lagging behind architectural trends.

III. RESEARCH

Because DOE facilities are dependent upon high-performance computers to accomplish their programmatic goals and because DOE is the largest customer for these computers, it must take the initiative in performing research on issues relevant to successful utilization of such machines. Five areas are proposed for investigation: algorithm analysis, architecture evaluation, scientific applications software, system software, and system architecture. Each of these areas is detailed below.

A. Algorithm Analysis

There has been a steady trend toward increased parallelism in high-performance computer architectures, with current generations incorporating a vector processing capability. When executing highly vectorized codes, the performance of these computers is an order of magnitude greater than when executing purely scalar codes. However, because DOE large-scale simulation computer programs were developed for scalar computers, often as much as 50% of the total computation must be done in scalar mode (that is, nonvectorizable computations); the situation is worse when highly parallel arrays of processors are considered. Thus, DOE must develop new solution techniques that can exploit new computer architectures.

B. Architecture Evaluation

In the past, DOE has had to commit itself to new generations of computers on the basis of potential rather than realized performance. Because of the

important missions they support, DOE needs technology that can evaluate the suitability of these computers to the numerical simulation that they perform. Such technology does not now exist.

C. Scientific Application Software

When developing a computer code to simulate some physical process, a scientist's primary concerns are to determine the appropriate laws of physics to describe the phenomenon under study and to express these laws in a mathematical form amenable to solution by computers. In the past, translation of the resulting equations into actual computer code by the use of a high-level language such as Fortran was straightforward. With the advent of machines such as the STAR, Cray-1, and ILLIAC IV, it has become necessary to structure codes in a stylized manner to achieve the high execution rates possible on such computers. Scientists cannot be expected to become sufficiently expert in programming techniques to write these stylized codes personally. Consequently, large staffs of professional programmers have become necessary to support the scientists. Future machines, using advanced architectural concepts, will require even more specialized structuring if they are to be used efficiently, and it is unreasonable to expect computer professionals to rethink codes for every new machine.

Research is needed to develop software that allows scientists to translate their models into computer codes without the necessity of learning the intricacies of new computers or without the expense of a large staff of professional computer scientists.

D. System Software

When developing a computer program, users need a variety of software tools: compilers, debuggers, program analyzers, and input-output file managers, to name but a few. Such tools already exist, but not in a format likely to map well onto new architectures. Thus research needs to be conducted in much the same way it is in large modeling codes: there must be interaction between the computer designers and the software designers. Of particular importance is the development of compilers able to translate the users' programs into efficient object code on new architectures.

E. System Architecture

As used here, system architecture is intended to connote more than the actual high-speed computing organ. Included also are other components such as storage hierarchies, input-output devices, remote terminals, the communications network, and the operating system that ties all these things together. It would be nothing less than a disaster to find that our supercomputers were unable to receive or send data at rates suited to effective high-speed computation.

Research in networks is needed to develop data transmission protocols that match the computer capabilities in access times, storage size, and bandwidths. The management of multi-access local networks requires protocols

to tie everything together in an efficient manner. Research is also needed to develop operating systems that are able to manage high-speed concurrent processing of mixed as well as single job streams.

IV. CONCLUSIONS

DOE research and development programs gain many advantages from a relatively small investment in high-performance computers. The value of a high-performance computer far exceeds its cost, which is small compared to the total costs of the weapons and energy systems under development in the United States and experimental research involved in these programs. However, the DOE is faced with a dilemma of diminishing growth rate in computer performance, growing complexity in computer architecture, and programmatic need for higher performance.

The research proposed in this report may well be essential to realizing the requisite computer performance needed by DOE.

REFERENCES

1. T. Gold and P. Matters, Nuclear Weapons and Computers Summary, prepared for the Director of Military Applications, DOE, March 1980.
2. "Magnetic Fusion Energy and Computers," J. Killeen, Ed., DOE/ER-0033, (October 1979).
3. "Low-cost Nuclear Safety," IEEE Spectrum, 18 (March 1980).
4. T. L. Owens, "The Influence of Machine Organization on Algorithms," Complexity of Sequential and Parallel Numerical Algorithms (Academic Press, 1973).
5. R. G. Voight, "The Influence of Vector Computer Architecture on Numerical Algorithms," High Speed Computer and Algorithm Organization, (Academic Press, 1977).
6. J. E. Wirshing and T. Kishi, "Matching Machines and Problems," High Speed Computer and Algorithm Organization. (Academic Press, 1977).
7. E. Giroux, "A Large Mathematical Model Implementation on the Star-100 Computer," High Speed Computer and Algorithm Organization (Academic Press, 1977).
8. P. Dubois, A. Greenbaum, and G. Rodrigue, "Approximating the Inverse of a Matrix for Use in Iterative Algorithms on Vector Processors," Lawrence Livermore Laboratory report UCRL-80244, presented to the 1977 Fall SIAM meeting, Albuquerque, NM.
9. P. Dubois, "Equation of State Look-up on STAR," presented to the 1977 SIGNUM meeting, Albuquerque, NM.

10. T. L. Jordan, "Vector Reduction Operations on Cray-1 and Their Performance," Proceedings of the 1977 International Conference on Parallel Processing, Jean-Loup Baer, Ed., IEEE Catalog No. 77CH1253-4C.
11. K. Fong and T. Jordan, "Some Linear Algebraic Algorithms and Their Performance on Cray-1," Los Alamos Scientific Laboratory report LA-6774 (June 1977).
12. B. L. Buzbee, J. A. Howell, and G. H. Golub, "Vectorization for the Cray-1 of some methods for solving elliptic difference equations," High Speed Computer and Algorithm Organization (Academic Press 1977).
13. J. Dongarra, "LINPACK Working Note #11," to appear as a LASL report.
14. T. Keller, "Cray-1 Evaluation," Los Alamos Scientific Laboratory report LA-6456-MS (December 1976).
15. Robert Sugarman, "'Superpower' Computers," IEEE Spectrum (April 1980).