Distribution Category:
Mathematics and
Computer Science (UC-405)

ANL-89/1

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439-4801

ANL--89/1

DE89 008323

ACTIVITIES AND OPERATIONS OF THE ADVANCED COMPUTING RESEARCH FACILITY

November 1987 - December 1988

Gail W. Pieper

Mathematics and Computer Science Division

January 1989

MASTER

DISCLAIMER

This work was supported by the Applied Mathematical Sciences subprogram of the Office of Energy Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

LEGIBILITY NOTICE

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although a small portion of this report is not reproducible, it is being made available to expedite the availability of information on the research discussed herein.

Contents

ACTIVITIES AND OPERATIONS OF THE ADVANCED COMPUTING RESEARCH FACILITY: November 1987 - December 1988

Gail W. Pieper

1 Advanced Computing Research Facility: Coming of Age

Four years ago Argonne National Laboratory established the Advanced Computing Research Facility (ACRF) to allow researchers access to advanced-computer architectures. Today the ACRF is used throughout the world for experimentation and parallel program development and has become the nation's leading center for research in high-performance computing.

In recognition of this achievement, the Mathematics and Computer Science Division held a "Coming of Age" celebration for the ACRF in September. Attending that celebration were persons who were involved in the early history of the facility, computer vendors who have worked with the ACRF staff in the selection of advanced computers, and members of our affiliates program who help faculty and students interested in using the ACRF computers.

In this 1987-1988 annual report, we highlight the activities and operations that have contributed to the success of the Advanced Computing Research Facility.

Undoubtedly our major accomplishment during the past few months was being designated as a National Science Foundation Center for Research on Parallel Computation. The Center will bring together computer scientists, mathematicians, and physical scientists from Rice University, the California Institute of Technology, Argonne, and Los Alamos National Laboratory to address the future computing needs of scientists. Central to this effort will be use of the diverse machines in the ACRF, which now range from a 2-processor Sequent used for *netlib* to a 16,384-node Connection Machine acquired as part of a joint agreement with the California Institute of Technology.

We are particularly proud of our expanding "outreach" programs. With special funding from the National Science Foundation, we held a two-week institute in parallel programming. We have also increased the number of university affiliates expressing interest in parallel computing.

What is the outlook for the future? We intend to continue to select and operate computers with innovative designs that are likely to be effective for a wide range of scientific computing tasks. Equally important, we intend to continue our research in parallel programming, devising methodologies, studying automated reasoning, developing performance measurement techniques, and devising algorithms for supercomputer applications.

2 Celebrating Success

Over 150 people celebrated the coming of age of our Advanced Computing Research Facility on September 22.

The ceremonies included addresses by Alan Schriesheim, director of Argonne National Laboratory; Frank Fradin, associate laboratory director of physical research; Don Austin, acting director, Scientific Computing Staff, DOE; Paul Huray, chairman, FCCSET Committee; Paul Messina, California Institute of Technology; Jack Dongarra, scientific director of the ACRF; and Hans Kaper, director of the Mathematics and Computer Science Division. Following the addresses, John Mucci officially dedicated the Connection Machine, and Allen Michels gave a demonstration of the Ardent Titan. The morning concluded with a tour of the ACRF and a brief social gathering.

A recurrent theme of the celebration was the success of the ACRF in promoting research in advanced scientific computing. In the remaining sections of this report we focus on these successes—the continuing expansion of the ACRF facility, the scientific impact of our research, and the educational programs we support to encourage use of the facility worldwide.

3 Expansion of the ACRF

3.1 New Machines

As part of our continuing efforts to have state-of-the-art machines available for experimentation, we temporarily acquired a Cydrome Cydra 5 supercomputer; the innovative data-flow architecture permitted researchers to experiment with computations involving fine-grained parallelism. We have also added to the ACRF two new machines.

The 16,384-node **Connection Machine** Model 2, manufactured by Thinking Machines Corporation, is our second massively parallel system. The CM-2 has 512 floating-point hardware units and a total of 128 megabytes of memory, 8 kilobytes at each node. This machine is capable of a performance in the gigaflop range.

The acquisition was arranged as part of a joint agreement with Caltech's Concurrent Computer Project.

The 4-processor **Ardent Titan** was acquired to enable scientists to explore the impact of visualization tools, which are becoming increasingly important for large-scale scientific computing. Often referred to as a graphics supercomputer, the Ardent combines high-performance graphics with a peak performance of 64 MFLOPS. Unlike other supercomputers in which the graphics capability is often simply an add-on, the Titan's color graphics subsystem is built in and has high-speed access to the memory and data bus. It can draw 600,000 three-dimensional vectors per second and display them on a 1280 x 1024-pixel color screen.

Through the use of a high-level graphics toolkit, the Titan enables researchers to visualize the results of computer simulations and to manipulate those results interactively. The Titan can also be used to create camera-quality static images and high-level animation sequences. We anticipate that users will find it valuable for such applications as molecular modeling and computational fluid dynamics.

In addition to the Connection Machine and the Ardent Titan, the ACRF features seven other multiprocessors:

- Active Memory Technology DAP 510, with 1024 processors arranged in a 32 by 32 array, with a total of 8 megabytes of memory.
- Alliant FX/8, with 8 vector processors sharing 32 megabytes of memory.
- Encore Multimax, with 20 processors sharing 64 megabytes of memory.
- Sequent b21, with 24 processors and 24 megabytes of shared memory.
- Intel iPSC/d5, a five-dimensional hypercube architecture with 32 nodes, each having a 0.5 megabyte of memory.
- Intel iPSC-VX/d4, a four-dimensional hypercube architecture with 16 nodes, each having 1.5 megabytes of memory. A special feature of this system is its vector capability, making the machine suitable for research in high-performance numerical computations.
- Sequent Balance, with two processors and 16 megabytes of shared memory. (This machine is used for *netlib*.)

The ACRF machines are linked to each other and are connected to national networks such as ARPAnet and BITnet. Thus, access to the ACRF machines is readily available throughout the world.

3.2 New Staff

To handle the increasing workload, we have also expanded our ACRF staff to include two new persons:

- John Rowlan, scientific visualization researcher. John's major research interests lie in volume visualization and the modeling of three-dimensional phenomena. In addition, John maintains the graphics computers and video recording equipment of the ACRF and is coordinating the scientific visualization initiative at Argonne.
- Mark Henderson, scientific assistant. Mark helps users of the advanced computers to implement their research programs. He also assists in the selection and implementation of new software and hardware.

John and Mark join the 7 other members listed below:

Jack Dongarra, ACRF Scientific Director
Burton Garbow

Teri Huml, ACRF Secretary
David Levine
Ewing Lusk, ACRF Deputy Scientific Director
Gail Pieper
Rick Stevens, Manager, MCS Computing Facilities

3.3 New Center

In December 1988, Argonne National Laboratory—together with Rice University, the California Institute of Technology, and Los Alamos National Laboratory—was granted over four million dollars (per year for five years) to establish a Center for Research on Parallel Computation. Under the direction of Rice University professor Ken Kennedy, the Center will study and develop efficient programming methods for state-of-the-art parallel computers. According to Kennedy, the Center will focus on two main challenges. The first is to develop new algorithms, or procedures for solving problems, that can be used efficiently on parallel computers; the second challenge is to make parallel computers easy to use.

4 Scientific Impact

One of our primary interests is to devise new methods and software tools that will make parallel computing easier and more efficient without sacrificing performance. The following subsections highlight recent successes of our scientific staff in the Mathematics and Computer Science Division. Principal investigators are listed in parentheses after each heading.

4.1 LAPACK (*E. Anderson, C. Bischof, J. Demmel, J. Dongarra, J. DuCroz, A. Greenbaum, S. Hammarling, W. Kahan, and D. Sorensen*)

The National Science Foundation is supporting the development of a linear algebra library based on EISPACK and LINPACK. The package, called LAPACK, is being developed as a collaborative project with Argonne National Laboratory, the Courant Institute, and the Numerical Algorithms Group. It will involve the integration of the two sets of algorithms into a unified library, with a systematic design; incorporation of recent algorithmic improvements; and restructuring of the algorithms to make as much use as possible of the Level 3 Basic Linear Algebra Subprograms (BLAS). Use of the BLAS is the basis of our approach to achieving efficiency.

Our primary objective is to make the library highly efficient, or at least tunable to high efficiency, on each machine. Otherwise it will not improve utilization, and users will continue to write their own (not necessarily better) algorithms. Additionally, the user interface must be uniform across machines; otherwise much of the convenience of portability will be lost. Finally, the programs must be widely available.

We have begun restructuring algorithms to optimize the use of the machine's memory hierarchy. Six technical reports have been distributed as working notes for this project. We have also established contacts with potential test sites throughout the country. We expect a preliminary version of the package to be ready for test sites by early 1989.

4.2 Performance Evaluations

4.2.1 Vectorizing Compilers: A New Test Suite (*D. Callahan, J. J. Dongarra, and D. Levine*)

Researchers from Argonne National Laboratory and Tara Computer Corp. have collaborated in the design and testing of a new suite to measure the effectiveness of an automatic vectorizing compiler. An automatic vectorizing compiler takes code written in a serial language (usually, Fortran) and translates it into vector instructions.

The new test suite, comprising 100 Fortran loops, measures four broad areas of a vectorizing compiler: dependency analysis, vectorization, idiom recognition, and language completeness.

Nineteen different compilers were tested on various machines, including supercomputers, minisupercomputers, mainframes, and graphics supercomputers. The average number of loops vectorized was 55 percent, and vectorized or partially vectorized was 61 percent. The best results were 69 percent and 80 percent, respectively. A summary table of the loops is presented below.

Table 1: **Summary of Test Suite (100 loops)**

Machine	Compiler	V	F	N
Convex C Series	FC5.0	69	5	26
Cray Series	CF77 V3.0	69	3	28
Alliant FX/8	FX/Fortran V4.0	68	5	27
Unisys ISP	UFTN 4.1.2	67	13	20
Hitachi S-810/820	FORT77/HAP V20-2B	67	4	29
Amdahl VP-E Series	Fortran 77/VP V10L30	62	5	27
ETA-10	FTN 77 V1.0	62	7	31
Ardent Titan-1	Fortran V1.0	62	6	32
CDC CYBER 205	VAST-2 V2.21	62	5	33
Gould NP1	GCF 2.0	60	7	33
Intel iPSC/2-VX	VAST-2 V2.23	56	8	36
IBM 3090/VF	VS Fortran V2.4	52	4	44
CRAY X-MP	CFT V1.15	50	1	49
Cray Series	CFT77 V3.0	50	1	49
Stellar GS 1000	F77 prerelease	48	11	41
CRAY-2	CFT2 V3.1a	27	1	72
CDC CYBER 990E/995E	VFTN V2.1	25	11	64
SCS-40	CFT x13g	24	1	75

V - vectorized; P - partially vectorized; N - not vectorized

4.2.2 Testing the DAP Elementary Function Library (*W. J. Cody*)

Performance tests were conducted on the Active Memory Technology DAP 510-8 computer. For the tests, we used the ELEFUNT suite of programs from the book *Software Manual for the Elementary Functions* by Cody and Waite. Overall, the DAP elementary function library proved disappointing. All of the programs tested could be improved, some dramatically. A few have unacceptable accuracy over some of their domain and for some precisions. In some cases the boundaries of the domain are not supported; in a few cases the algorithms used are faulty. The library also suffers from not having TAN, ASIN, ACOS, ALOG10, ATAN2, and the hyperbolics.

4.3 Algorithm Design and Restructuring

A major portion of our research is devoted to the development of algorithms that exploit parallel architectures. During the past year several projects have been carried out:

- D. Sorensen has been working on two problems in algorithm restructuring. The first involves recasting the symmetric eigenvalue problem to operate in SIMD mode. Preliminary results on the Connection Machine have been encouraging. The second problem involves reorganizing the TREEQL code in terms of matrix-matrix products. The results, which will be incorporated into LAPACK, show distinct advantages, particularly on the Alliant, the CRAY-2, and the CRAY X-MP.

- C. Bischof has devised a new version of the QR factorization algorithm with column pivoting that substantially decreases the communication overhead for this method. Experiments on an iPSC hypercube show a reduction in overhead of 50 percent. He has also developed an adaptive blocking methodology that addresses the question of how to partition a given problem to achieve maximum performance. Experiments on an iPSC show this method to be optimal, independent of problem size and the number of processors employed.

- H. Kaper, M.S. Du, and M. K. Kwong have used the Encore, the Alliant, and the DAP to investigate the parallel solution of certain nonlinear reaction-diffusion equations using the shooting method and an improved Runge-Kutta method. The purpose of the study is to discover whether bifurcation of the ground state solution of a boundary value problem occurs as some of the parameters of the equations are varied. The investigation has produced an efficient code, which we are planning to further improve and use to study a much wider class of equations.

- S. Wright has been studying the application of the proximal point algorithm to linear programming problems. The operation at the core of this method is the solution of a system of linear equations whose coefficient matrix is some principal minor of a matrix of the form $A^T A$, where A is sparse and is stored either by rows or columns. For both these storage schemes, parallelizable techniques for accumulating matrix-vector products of the form $A^T Ap$ have been implemented on the Alliant. The overall algorithm is showing promise as a means of solving large sparse general linear programs, and research is continuing.

- M. Garbey and D. Levine are investigating the use of the Connection Machine to solve conservation laws (or, more generally, quasi-linear hyperbolic PDEs). The underlying method is a mixture of the use of cellular automata and the method of characteristics. The characteristics are followed at each discrete time step, with their intersection representing the occurrence of a shock.

4.4 Automated Deduction (*A. Jindal, W. McCune, and R. Overbeek*)

We have developed and implemented a parallel version of our theorem prover OTTER on the Encore. The parallelism in OTTER is achieved by distributing the work of the theorem prover among two or more processes.

During our first experiment, we parallelized OTTER using an implementation based on a scheme developed at Argonne by R. Butler and N. Karonis (two visitors to the ACRF last summer). We succeeded in obtaining speedups of up to seven (for problem solving) on the Encore using 10 processors. The timings we obtained for some of the benchmark problems are reflected in Table 2.

Table 2: **Timings for Two Benchmarks**

Problem Name	No. of Processors				
	1	2	4	8	10
SAM's lemma runtime (secs)	42.92	21.07	10.98	6.38	6.33
Lukasiewicz's runtime (hr.min)	21.35	9.51	5.15	3.50	3.20

Our experimentation showed that because of heavy traffic on the Master, a new approach was required for achieving high performance in the deduction systems.

We have already developed a new algorithm based on domain decomposition and functional decomposition techniques. In this new algorithm, we consider an environment consisting of one or more parallel pipelines. Each pipeline is a sequence of the generator, an identity-checker, a subsumer, and a distributor. The generator is responsible for generating the resolvents and passing them over to the identity-checker. The identity-checker is responsible for eliminating duplicate resolvents. The subsumer eliminates subsumed resolvents, and forwards the resulting set of resolvents to the distributor. The distributor partitions the resulting resolvents among all the generators. We are currently implementing a parallel version of OTTER using this approach on the Symmetry. Preliminary results of this experiment are encouraging. We anticipate that we will be able to demonstrate considerable speedups for problem solving using this new scheme.

5 Scientific Visualization

With the growing complexity of scientific theories and a constantly expanding volume of scientific data, tools are needed to help researchers interpret and explain experimental results. Our goal in the scientific visualization initiative is to create portable visualization tools that help scientists process large quantities of data, explore new areas of science, and investigate novel methods of scientific visualization.

5.1 Three-Dimensional Scientific Visualization (*J. Rowlan*)

We have been actively involved in four projects:

- with M. Minkoff (MCS), a three-dimensional visualization tool was developed to model the combustion process that arises in the burning of a solid fuel while taking into account the melting of the fuel. Work toward producing video movies depicting this event is under way.
- with P. Sigmund (Physics), a tool is being developed to visualize structural anomalies of materials at the superconducting transition temperature.
- with Rajiv Kalia (Materials Components and Technology), a program is being developed to simulate a model of an electron in a dense helium gas.
- a project to visualize and animate three-dimensional fluid flow.

5.2 Graphics Trace Facilities (*O. T. Brewer, J. J. Dongarra, F. Hanson, and D. C. Sorensen*)

We are exploring an approach to parallel programming that involves describing the parallel structure of the program in terms of the units of computation and execution dependencies between them. These ideas are then implemented in a Fortran program written as subroutine calls to a software package called SCHEDULE. Since the specification of data dependencies can be both tedious and error prone, we have incorporated into SCHEDULE a graphics input tool that enables a user to interactively draw the dependency graph on a workstation screen and have the workstation generate the Fortran code with the necessary calls to the SCHEDULE library. This code can then be compiled, linked with the user-supplied subroutines and the SCHEDULE library, and run on the target parallel processor. As part of the package, SCHEDULE includes a special trace facility that animates the parallel parts of the program as they execute. This facility helps researchers identify bottlenecks and detect load balancing problems. Once such anomalies are discovered, the execution dependencies can be revised to force certain processes to complete before others.

We improved the queuing system in the SCHEDULE package so that applications requiring a much larger number of processes could be executed. A demonstration program, related to the triangular solve problem, was executed using more than a million spawned processes. An additional improvement permits the iteration of dependency graphs with minimal overhead.

We are also using visualization to experiment with different memory hierarchy schemes and to observe the effects on data flow in a parallel program. The strategy is to use a set of tools called MAP (for Memory Access Pattern program) to analyze an arbitrary Fortran program, map the arrays to a graphics screen, and highlight the elements of the arrays when they are accessed. If a call to a BLAS (Basic Linear Algebra Subprograms) subroutine has been made, the trace file may contain the information about a row or column access, or both. The MAP tools also have a feature to allow statistical analysis and data collection of various cache systems.

5.3 Publications, Preprints, and Presentations

One important measure of the success of our work is the publication of papers and the presentation of our results to the scientific community. Listed below are publications, preprints, and presentations relating to advanced computing research during the past year. This list updates the lists given in previous progress reports [1,2,3].

Publications

C. Bischof, J. Demmel, J. Dongarra, J. DuCroz, A. Greenbaum, S. Hammarling, and D. Sorensen, "LAPACK Working Note No. 5: Provisional Contents," ANL-88-38 (September 1988)

O. Brewer, J. Dongarra, D. Levine, and D. Sorensen, "Graphic Tools for Developing High-Performance Algorithms," International Meeting on Parallel Computing Methods, Algorithms, and Applications, University of Verona, Italy, September 28-30, 1988 (also MCS-P28-1288)

O. Brewer, J. Dongarra, and D. Sorensen, "LAPACK Working Note No. 6: Tools to Aid in the Analysis of Memory Access Patterns for Fortran Programs," MCS-TM-120 (June 1988)

R. M. Butler and N. T. Karonis, "Exploitation of Parallelism in Prototypical Deduction Problems," *Lecture Notes in Computer Science,* Vol. 310, ed. E. Lusk and R. Overbeek, Springer-Verlag, New York, 1988, pp. 333-343

R. M. Butler, R. Loganantharaj, and R. Olson, "Notes on Prolog Program Transformations, Prolog Style, and Efficient Compilation to the Warren Abstract Machine," *Lecture Notes in Computer Science,* Vol. 310, ed. E. Lusk and R. Overbeek, Springer-Verlag, New York, 1988, pp. 323-332

W. J. Cody, "ELEFUNT Test Results under FORTRAN-PLUS on the Active Memory Technology DAP 510-8," ANL/MCS-TM-125 (September 1988)

W. R. Cowell, "DO Loop Transforming Tools," *Using Toolpack Software Tools,* ed. A. A. Pollicini, ECSC, EEC, EAEC, Brussels, 1989, pp. 161-180

W. R. Cowell, "Users' Guide to Toolpack/1 Tools for Data Dependency Analysis and Program Transformation," ANL-88-17 (September 1988)

J. Demmel and W. Kahan, "LAPACK Working Note No. 3: Computing Small Singular Values of Bidiagonal Matrices with Guaranteed High Relative Accuracy," MCS-TM-110 (February 1988)

J. J. Dongarra, J. DuCroz, I. Duff, and S. Hammarling, "A Proposal for a Set of Level 3 Basic Linear Algebra Subprograms," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp.40-44

J. J. Dongarra, J. DuCroz, S. Hammarling, and R. J. Hanson, "Algorithm 656—An Extended Set of Basic Linear Algebra Subprograms: Model Implementation and Test Programs," ACM Trans. Math. Soft. 14, no. 1 (March 1988) 18-32

J. J. Dongarra, J. DuCroz, S. Hammarling, and R. J. Hanson, "An Extended Set of FORTRAN Basic Linear Algebra Subprograms," ACM Trans. Math. Soft. 14, no. 1 (March 1988) 1-17

F. Hanson, "Computational Dynamic Programming for Stochastic Optimal Control on a Vector Multiprocessor," MCS-TM-113 (June 1988)

F. B. Hanson, "Parallel Computation for Stochastic Dynamic Programming: Row versus Column Code Orientation," *Algorithms and Applications,* Vol. 3, ed. D. H. Bailey, The Pennsylvania State University Press, 1988, pp. 117-119

E. R. Jessup and D. C. Sorensen, "A Divide and Conquer Algorithm for Computing the Singular Value Decomposition," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp. 61-66

E. L. Lusk, R. Overbeek, and L. Sterling, "Effective Utilization of OR-Parallelism: A Modest

Proposal," MCS-TM-124 (June 1988)

P. Mehrotra and J. van Rosendale, "The BLAZE Family of Languages: Programming Environments for Shared and Distributed Memory Architectures," MCS-TM-108 (June 1988)

J. Scroggs, "The Solution of a Parabolic Partial Differential Equation via Domain Decomposition: The Synthesis of Asymptotic and Numerical Analysis," MCS-TM-123 (June 1988)

Preprints

C. H. Bischof, *Incremental Condition Estimation,* MCS-P15-1088

W. R. Cowell and C. P. Thompson, *Tools to Aid in Discovering Parallelism and Localizing Arithmetic in Fortran Programs,* MCS-P6-1088

J. Demmel, J. Dongarra, J. DuCroz, A. Greenbaum, S. Hammarling, and D. Sorensen, *A Project for Developing a Linear Algebra Library for High-Performance Computers,* MCS-P37-1288

J. Dongarra, J. DuCroz, I. Duff, and S. Hammarling, *A Set of Level 3 Basic Linear Algebra Subprograms,* MCS-P1-0888

J. Dongarra, J. DuCroz, I. Duff, and S. Hammarling, *Level 3 Basic Linear Algebra Subprograms: Model Implementation and Test Programs,* MCS-P2-0888

K. W. Dritz, *Plugging the Holes in the Sieve of Eratosthenes,* MCS-P4-0888

Presentations

C. Bischof, *Incremental Condition Estimatation and an Application in a Local Pivoting Scheme,* Meeting of the American National Society, Lawrence, Kansas, October 27-29, 1988

C. Bischof, *A Parallel QR Factorization Algorithm Using Local Pivoting,* Supercomputing'88, Kissimmee, Florida, November 14-18, 1988

C. Bischof, *Incremental Condition Estimation and Applications,* Rutgers University, New Brunswick, New Jersey, Dec. 2-3, 1988

D. Callahan, J. Dongarra, and D. Levine, *Vectorizing Compilers: A Test Suite and Results,* Supercomputing'88 Conference, Kissimmee, Florida, November 16, 1988

M. Carlsson, K. Danhof, and R. Overbeek, *Practical Issues Relating to the Internal Database Predicates in an OR-Parallel Prolog: Extensions and Useful Hacks,* Fifth International Logic Programming Conference, Seattle, August 15-19, 1988

J. J. Dongarra, *A Package for Linear Algebra on High-Performance Machines,* IFIPS WG 2.5, Stanford, August 23, 1988

J. J. Dongarra, *The ACRF and High-Performance Linear Algebra Libraries,* Wilmington, Delaware, October 11, 1988

J. J. Dongarra, *Tools for Parallel Algorithm Development,* Workshop on Parallel and Distributed Processing, Bonas, France, October 4, 1988

J. J. Dongarra and D. C. Sorensen, participants in a Workshop on Vector and Parallel Processing, University of Umea, Umea, Sweden, August 29-September 2, 1988

J. J. Dongarra, D. C. Sorensen, and O. Brewer, *Tools and Methodology for Programming Parallel Processors,* Proc. IFIP Working Group 2.5 Conference, Stanford University, August 23-25, 1988

(also MCS-P29-1288)

K. W. Dritz, *Overview of Ada,* 1988 Salishan Conference on High-Speed Computing, Gleneden Beach, Oregon, March 15, 1988

K. W. Dritz, *A Concurrent Solution of Hamming's Problem (Extended) in Ada,* 1988 Salishan Conference on High-Speed Computing, Gleneden Beach, Oregon, March 15, 1988

K. W. Dritz, *A Concurrent Simulation of a Doctor's Office in Ada,* 1988 Salishan Conference on High-Speed Computing, Gleneden Beach, Oregon, March 16, 1988

M. D. Hirsch and S. Vavasis, *Exponential Lower Bounds for Finding Brouwer Fixed Points,* 28th Annual Symposium on Foundations of Computer Science, Computer Society Press, Washington, 1988

D. Levine, *Tools for Parallel Algorithm Development,* International Meeting on Parallel Computing, Verona, Italy, Sept. 28-30, 1988

D. Levine, *Argonne's Fortran Monitor and Macro Package,* University of Naples, Italy, October 3, 1988

E. Lusk, *High Performance Automated Reasoning,* University of Texas, April 1988

E. Lusk, *Parallel Programming in Logic,* University of Sidney, Australia, July 1988

E. Lusk, *Providing Research Resources for High-Performance Computing: The Advanced Computing Research Facility at Argonne National Laboratory,* Australian National University, July 1988

E. Lusk, *A Three-Day Course in Parallel Computation,* Australian National University, July 1988

E. Lusk, *Parallel Programming in Logic,* University of Wollongong, July 1988

E. Lusk, *More Oxen or a Bigger Ox: The Challenge of Parallel Computation in the 90's,* Australian National University, Australia, August 1988

E. Lusk, *Parallel Programming in Logic,* Melbourne University, August 1988

E. Lusk et al., *The Aurora OR-Parallel Prolog System,* International Conference on Fifth Generation Computer Systems, Tokyo, Japan, Nov. 28-Dec. 2, 1988

R. Overbeek, *The Implementation of a Theorem Prover That Successfully Exploits Multiprocessors,* AAAI89 Workshop on Parallel Algorithms for Machine Intelligence and Pattern Recognition, Minneapolis, August 20, 1988

R. Overbeek, *Answering Open Questions with an Automated Reasoning System,* 1988 SIAM Annual Meeting, Minneapolis, July 12, 1988

B. T. Smith, *Parallel Computing Forum (PCF) Fortran—A Report,* IFIP TC2/WG 2.5 Working Conference on Aspects of Computation on Asynchronous Parallel Processors, February 19, 1988

D. Sorensen, *Divide and Conquer Methods for the Symmetric Eigenvalue Problem and for the Singular Value Decomposition,* Numerical Linear Algebra and Parallel Computation Meeting, Oberwolfach, West Germany, February 28, 1988

D. Sorensen, *Argonne's Parallel Processing Environment,* Stanford, August 24, 1988

D. Sorensen, *Divide and Conquer Algorithms for Eigenvalue Problem on the Connection Machine,* Tromso, Norway, June 8, 1988

R. Stevens, Guest lecture and two-week course on parallel computing, Australia National University, Canberra, Australia, December 1-17, 1988

Books and Other Papers Citing Argonne's ACRF

R. G. Babb II and D. C. DiNucci, "Scientific Parallel Processing with LGDF2," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp. 307-311

J. L. Barlow and U. B. Vemulpati, *On One-Way Dissection with Singular Diagonal Blocks,* Report No. CS-88-02, The Pennsylvania State University, March 1988

D. Cann, *Porting Multiprocessor SISAL Software,* Technical Report No. CS-88-104, Computer Science Department, Colorado State University, 1988

E. De Donker and J. Kapenga, "A Parallelization of Adaptive Task Partitioning Algorithms," Parallel Computing 7 (1988)

E. De Donker and J. Kapenga, "Parallel Multivariate Numerical Integration," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp. 109-113

K. Dowers, S. Lakshmivarahan, S. K. Dhall, and J. Diaz, "Block Tridiagonal Systems on the Alliant FX/8," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, pp. 56-60

A. Esfahanian, L. Ni, and B. E. Sagan, *On Enhancing Hypercube Multiprocessors,* 1988 International Conference on Parallel Processing, St. Charles, Illinois, August 15-19, 1988

F. B. Hanson, *The Alliant FX/8,* University of Illinois at Chicago, UIC Workshop Program on Scientific Supercomputing, January 25, 1988

F. B. Hanson, *Numerical Libraries,* University of Illinois at Chicago, UIC Workshop Program on Scientific Supercomputing, February 17, 1988

F. B. Hanson, *Vectorization, Parallelization and Dynamic Programming,* Bethesda, Maryland, Mathematics Research Branch, NIDDK, National Institute of Health, March 24, 1988

F. B. Hanson and D. C. Sorensen, *The SCHEDULE Parallel Programming Package with REcycling Job Queues and Iterated Dependency Graphs,* Supercomputing '88, Kissimmee, Florida, Nov. 15, 1988

V. Henson, *Parallel Compact Symmetric FFTs,* International Conference on Vector and Supercomputing, Tromso, Norway, June 1988

B. J. Helland and R. J. Krueger, "A Parallel Algorithm for Rapid Computation of Transient Fields," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp. 270-275

J. Kapenga and E. de Doncker, "Concurrent Management of Priority Queues for Adaptive Algorithms," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp. 347-351

C-T King, W-H Chou, and L. M. Ni, *Large-Grain Pipelining on Distributed-memory Multiprocessors,* Third International Conference on Supercomputing, Boston, May 15-20, 1988

C-T King, W-H Chou, and L. M. Ni, *Pipelined Data Parallel Algorithms: Concept and Modeling,* 1988 International Conference on Supercomputing, Saint-Malo, France, July 1988

Y. Lan, L. Ni, and A-H Esfahanian, *Distributed Multi-Destination Routing in Hypercube Multiprocessors,* Third Conference on Hypercube Concurrent Computers and Applications, January 1988

Y. Lan, L. M. Ni, and A-H Esfahanian, *Relay Approach Message Routing in Hypercube Multiprocessors,* Third International Conference on Supercomputing, Boston, May 15-20, 1988

L. M. Ni, *Large-grain Pipelining on Hypercube Computers,* Proceedings of the Third Conference on Hypercube Concurrent Computers and Applications, January 1988

L. M. Ni, *Executable Assertion Development for Reliable Parallel Processing,* Proceedings of the 1988 COMPSAC, Chicago, Ill., October 1988

J. Scroggs, "Parallel Processing of a Domain Decomposition Method," *Parallel Processing for Scientific Computing,* ed. G. Rodrigue, SIAM, Philadelphia, 1988, pp. 164-168

A. Siem and N. Sabelli, *A Training Program for Scientific Supercomputing Users,* Tutorial presented at Supercomputing'88, November 14-18, 1988

R. Tuck, *An Optimally Portable SIMD Programming Language,* Report No. TR88-048, The University of North Carolina at Chapel Hill, October 1988

6 The ACRF—Reaching Out to Users

The ACRF was established to be used for advanced computing research by researchers throughout the world. To encourage use of the ACRF's specialized resources, the Mathematics and Computer Science Division conducts classes; supports an active visitors program; and sponsors a variety of seminars, workshops, and institutes.

6.1 Classes

Argonne conducted 10 classes on parallel computing during the past year. The attendees, averaging 20-25 persons per class, represented universities, industry, and various research laboratories throughout the country. The intent of the classes was to familiarize the attendees with the ACRF environment, to offer ample hands-on experience on the parallel computer systems, and to apply parallel programming to each attendee's area of research. During the classes, the attendees were taught how to write and run parallel programs in Fortran. Session topics that were addressed included parallel compilers; monitors and their implementation with macros; the SCHEDULE package; and special techniques for programming the hypercubes, the DAP, and the Connection Machine.

Frequently, as classes familiarize researchers with the machines in the ACRF, potential users propose new applications and techniques to implement on our advanced computers. Over the past year, more than 110 research proposals have been submitted and accepted.

6.2 Visitors

During the past year, we have had numerous visitors here working on projects involving computer graphics, high-performance algorithms, and software tools for advanced-architecture computers: These visitors, who include scientists from industry and academia, stay for periods ranging from two weeks to several months.

Faculty and Senior Visitors

Ralph Butler, Research Associate
University of North Florida
Multiprocessing tools, theorem proving systems, and Prolog implementations

Moody Chu, Faculty Research Leave at Argonne
North Carolina State University
Iterative processes for solving linear systems, matrix factorization, and eigenvalue problems using the homotopy methods

Patricia Eberlein, Senior Visiting Scientist
State University of New York
Mastering new techniques and software for exploiting advanced-architecture computers

Floyd Hanson, Faculty Research Leave at Argonne
University of Illinois, Chicago
Stochastic dynamic programming

Rolf Hempel, Resident Associate
GMD, West Germany
Creation of a Fortran-callable version of portable message-passing mechanisms

Robert McFadden, Resident Associate
Northern Illinois University
Application of automated reasoning methods to the study of semigroups

Lionel Ni, Faculty Research Leave at Argonne
Michigan State University
Partitioning and load balancing for parallel processing, automated reasoning in VLSI design, implementation of parallel homotopy methods for finding the roots of a system of polynomial equations

T. Saitoh
Institute for Supercomputing Research, Tokyo
Study of CFD applications on the Connection Machine and the Titan

Douglas Salane, Scientist
Consultant
New techniques and software for exploiting advanced-architecture computers

Will Winsborough, Resident Associate
Joint Appointment with University of Chicago
Parallel implementation of Parlog

Stephen Wright, Research Associate
North Carolina State University

Numerical solution of nonlinear programming problems

Students and Other Visitors

James Crammond, Research Associate
Imperial College
Implementation of a committed-choice dialect for loosely-coupled multiprocessors

Ian Foster, Research Associate
Imperial College
Implementation of a committed-choice dialect for loosely-coupled multiprocessors

Andrea Hudson, Resident Student Associate
University of Kentucky
Parallel algorithms in linear algebra on massively parallel computers

Anita Jindal, Resident Student Associate
University of Illinois at Chicago
Producing the next generation of LMA/ITP based on multiprocessors and Prolog technology

Nicholas Karonis, Scientific Assistant
Syracuse University
Computational logic, helping users of a large SIMD multiprocessor

James Kohl, Laboratory Aide
Purdue University
Performance evaluation studies, development of a C++ based version of the Argonne parallel programming toolkit

Peter McDonough, SERS Student Program
Mt. Union College
Development of software to solve systems of PDEs on advanced computers

Robert Olson, Scientific Assistant
University of Illinois at Urbana
Implementation of automated deduction systems in C and Prolog

Nathan Pfluger, Student Research Participation Program
New College
Prototyping a database to hold information pertinent to molecular biology, study of similarities between distinct RNA sequences

Morgan Price, Student Research Participation Program
University of Chicago
Prototyping a database to hold information pertinent to molecular biology and aligning RNA sequences

Steve Tuecke, SERS Student Program
St. Olaf College
C++ implementation of the multiprocessing tools

6.3 Seminars

During 1988, MCS and the Computing and Telecommunications Division jointly sponsored a series of seminars on high-performance computing. Below are listed the names of our speakers, their affiliations, and the title of their talk.

Stefan Fangmeier, *The Setting Up and Management of an Advanced Computer Animation Facility for Scientific Research Applications,* NCSA Scientific Visualization Program - University of Illinois, February 18, 1988

J. C. Browne, *Architectural and Language Independent Parallel Programming,* University of Texas, April 5, 1988

Robert Schnabel, *DINO—A New Language for Numerical Computation on Distributed Memory Multiprocessors,* University of Colorado, May 19, 1988

Iain Duff, *The Solution of Large-Scale Least-Squares Problems on Supercomputers,* Harwell Laboratory, May 20, 1988

James Demmel, *Topics in Numerical Linear Algebra,* Courant Institute, July 28, 1988

Oliver A. McBryan, *Promise vs. Performance for Massively Parallel Computers,* University of Colorado, August 19, 1988

Peter Deuflhard, *Concepts of an Adaptive Hierarchical Finite Element Code,* Center for Information Technology - West Germany, September 15, 1988

Nicholas Higham, *Fast Polar Decomposition of an Arbitrary Matrix,* Cornell University, September 29, 1988

Alan Huang, *Optical Digital Computers,* AT&T Bell Laboratories, October 6. 1988

Jon A. Solworth, *PARSEQ: A Parallel Programming Language with Sequential Semantics,* University of Illinois at Chicago, October 20, 1988

Craig Upson, *Co-Processing Environments for Interactive Visualization,* Stellar Computer Inc., October 27, 1988

Henk A. van der Vorst, *High-Performance Preconditioning and Numerical Aspects of Parallel Computing,* Delft University of Technology, The Netherlands, October 28, 1988

Gerard Meurant, *Practical Use of the Conjugate Method on Vector and Parallel Supercomputers,* Centre d'Etudes de Limeil, November 1, 1988

Willi Schönauer, *Why I Like Vector Computers,* University of Karlsruhe, West Germany, December 9, 1988

John Gilbert, *Sparse Matrix Computation on the Connection Machine,* Xerox Palo Alto Research Center, December 14, 1988

6.4 University Affiliates Program

We have established an affiliates program for colleges and universities interested in using the ACRF multiprocessing systems and in supporting our research. Among our new members we welcome the following:

- Linda Morris, Cornell Theory Center, Cornell University
- George Miminis, Dept. of Computer Science, Memorial University of Newfoundland, Canada
- Richard Enbody, Dept. of Computer Science, Michigan State University
- R. J. Plemmons, Mathematics Dept., North Carolina State University
- Jorge Nocedal, Electrical Engineering and Computer Science Dept., Northwestern University
- J. Sechrest, Computer Science Dept., Oregon State University
- Floyd Hanson, Dept. of Mathematics, Statistics, and Computer Science, University of Illinois - Chicago
- Ahmed Sameh, Center for Supercomputer Research and Development, University of Illinois - Urbana-Champaign
- Julio Diaz, Mathematics and Computer Science Dept., University of Tulsa
- Naveed Sherwani, Dept. of Computer Science, Western Michigan University

Several of these affiliates have used the ACRF machines for classes in advanced computing. In particular, we note the following:

- North Carolina State University - "Parallel Algorithms and Supercomputers" - R. White
- University of Illinois, Chicago - "Numerical Methods for Supercomputers" - F. Hanson
- University of Tulsa - "Parallel Programming" - J. Diaz

In addition, the University of Illinois at Chicago sponsored a Workshop Program on Scientific Supercomputing, in which approximately twenty-four graduate students spent time optimizing applications for parallelization and vectorization on advanced computers. Two visits were made to the ACRF during the 1987-1988 academic year.

6.5 Technology Transfer

For several years Argonne National Laboratory has actively encouraged interactions between research divisions and industry. This past year, the ACRF was fortunate to have two scientists join our staff as part of the Laboratory Technology Exchange Program.

Kirk Jordan spent several months at Argonne on leave from Exxon Research and Engineering (ER&E) Company. His objective was to carry out performance testing on machines in the ACRF. The focus of these efforts was an elastic wave code which Jordan modified extensively to run in parallel form on the Alliant. His work was aided considerably by the use of two Argonne-developed tools: SCHEDULE and Toolpack. Jordan summarized his experience with the hope that ER&E would take greater advantage of and provide support for the ACRF. "The benefit to ER&E and to Exxon Corporation," he said, "is the opportunity to conduct extended tests on very different architecture machines without having to invest in the equipment." He added, "In turn, giving feedback on . . . tools helps those in MCS to better meet the needs of end-users."

Tim Lindholm of Quintus Computer Systems, Inc., worked at Argonne for a four-month period from July through October 1988. Quintus is one of the world's foremost implementors of commercial Prolog systems. Lindholm's objective in collaborating with ACRF staff was to develop a high-quality, high-performance commercial implementation of an OR-parallel Prolog system on a shared-memory multiprocessor. The engine he designed was extremely satisfactory. According to Lindholm, "A parallel system employing the engine would be likely to outperform sequential systems." The new system in fact surpassed the 15-35 percent improvement hoped for at the start of the project. "It is clear that the work at Argonne advanced the state of the art substantially," Lindholm said.

6.6 Summer Institute

On September 6-16, 1988, we hosted a **Summer Institute in Parallel Programming** to familiarize university researchers with new techniques in high-performance computing. Twenty-five graduate students and postdoctoral researchers were selected out of 80 applicants to participate in the institute, which was sponsored in part by the National Science Foundation and in part by the Department of Energy.

Invited speakers included Arvind (MIT), G. Demos (DemoGraFX), J. Fisher (Multiflow Corporation), K. Kennedy (Rice University), S. Nelson (Cray Research), C. Moler (Ardent), C. Seitz (Caltech), B. Smith (Tera Computer), G. Steele (Thinking Machines), and B. Wulf (NSF). Each invited speaker conducted a three-hour session on advanced computing. In addition, Argonne research staff conducted lectures and "hands-on" sessions, in which the attendees used the ACRF machines.

The participants unanimously agreed that the institute gave them a deeper understanding of the state of the art in parallel computing and provided useful information about software tools available for parallel program development.

A report of the workshop is documented in ANL-88-45.

6.7 Bulletins and Brochures

For our ACRF users, we initiated a periodic bulletin called *Adventures in Parallelism.* The first issue was mailed out in July, the second in December. Our objective is to let users know of new projects, equipment, and research results in parallel computing associated with our ACRF. We would also like to use *Adventures in Parallelism* as a forum for discussing issues in parallelism.

For visitors to Argonne, we revised our ACRF brochure. This brochure is handed out to those visiting the laboratory or those who have just heard mention of the Advanced Computing Research Facility. The brochure contains a capsule summary (in layman's terms) of the activities and operations of the ACRF.

References

1. T. Mihaly and G. W. Pieper 1986. *Activities and Operations of the Advanced Computing Research Facility, January 1985 - July 1986*, ANL-86-34 (August).

2. G. W. Pieper 1986. *Activities and Operations of the Advanced Computing Research Facility, July 1986 - October 1986*, ANL-86-55 (November 1986)

3. G. W. Pieper 1987. *Activities and Operations of the Advanced Computing Research Facility, October 1986 - October 1987*, ANL-87-50 (November 1987)

Distribution for ANL-89/1

Internal:

C. Adams
J. M. Beumer (50)
T. Fields
F. Y. Fradin
T. M. Huml (100)
H. G. Kaper
A. B. Krisciunas
G. W. Pieper
D. P. Weber

ANL Patent Department
ANL Contract File
ANL Libraries
TIS Files (3)

External:

DOE-OSTI, for distribution per UC-405 (66)
Manager, Chicago Operations Office, DOE
Mathematics and Computer Science Division Review Committee:

J. L. Bona, Pennsylvania State University
T. L. Brown, University of Illinois, Urbana
P. Concus, Lawrence Berkeley Laboratory
S. Gerhart, Microelectronics and Computer Technology Corp., Austin, Texas
H. B. Keller, California Institute of Technology
J. A. Nohel, University of Wisconsin, Madison
M. J. O'Donnell, University of Chicago

B. Alper, Active Memory Technology, Irvine, CA
D. Austin, Department of Energy - Energy Research, Washington
J. Diaz, University of Tulsa
R. Enbody, Michigan State University
G. Fielland, Sequent Computer Systems, Inc., Beaverton, OR
G. Fox, California Institute of Technology
F. Hanson, University of Illinois at Chicago
R. Kuhn, Alliant Computer Systems, Littleton, MA
P. Messina, California Institute of Technology
G. Michael, Lawrence Livermore National Laboratory
G. Miminis, Memorial University of Newfoundland, Canada
C. Moler, Ardent Computer Corp., Sunnyvale, CA
L. Morris, Cornell University
I. Nassi, Encore Computer Corp., Marlborough, MA

J. Nocedal, Northwestern University
R. Plemmons, North Carolina State University
A. Sameh, CSRD, University of Illinois at Urbana
D. Scott, Intel Development, Beaverton, OR
J. Sechrest, Oregon State University
N. Sherwani, Western Michigan University
G. Steele, Thinking Machines, Cambridge, MA