

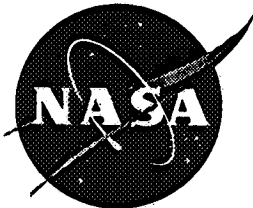
NASA Technical Memorandum 4653

NASA High Performance Computing and Communications Program

Annual Report, 1993

*Office of Aeronautics
High Performance Computing and Communications Office
Washington, D.C.*

Lee Holcomb, Director
Paul Smith, Program Manager
Paul Hunter, Program Manager



National Aeronautics and
Space Administration

Office of Management

Scientific and Technical
Information Program

1994

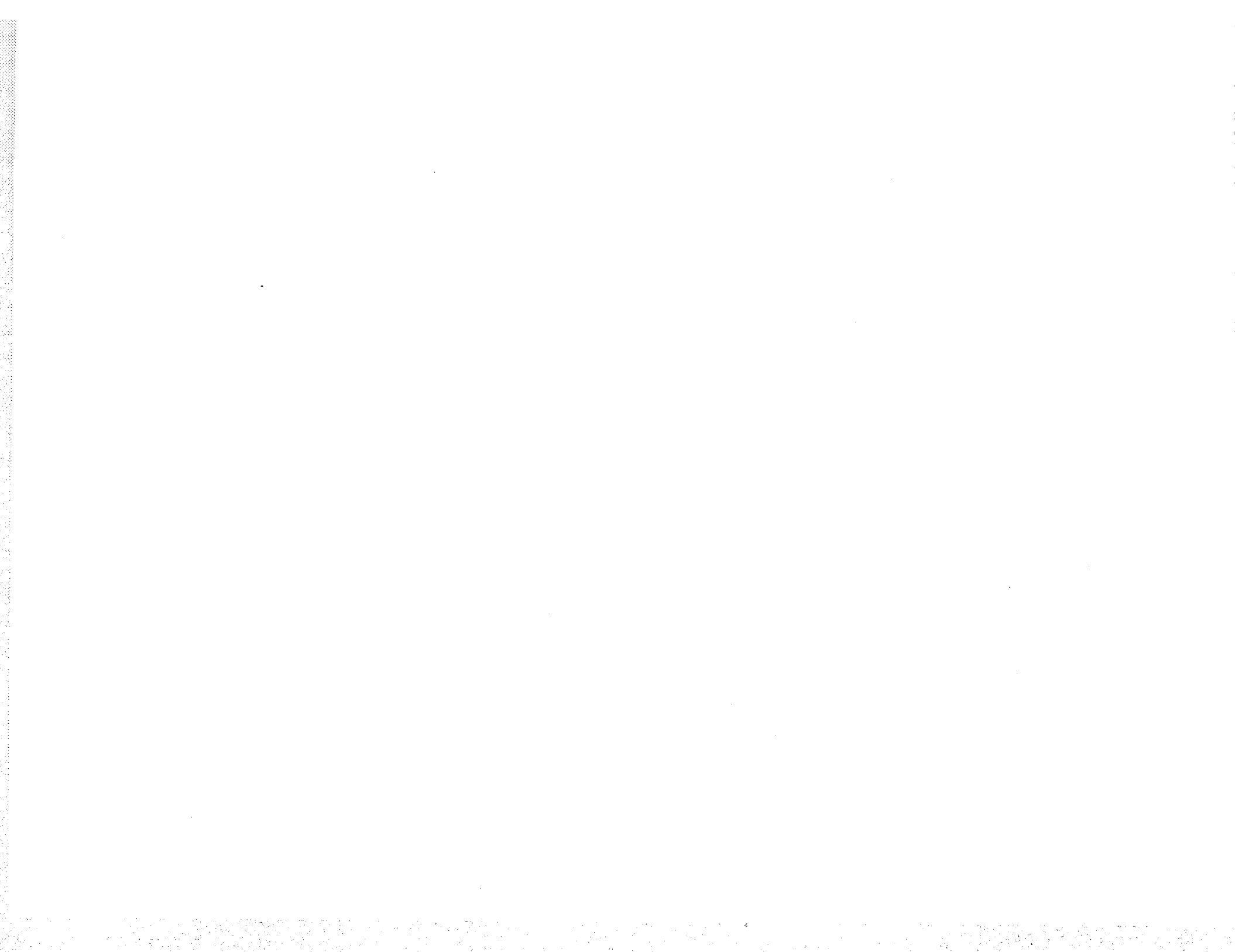


Table of Contents

List of Figures	iii
NASA Directory	v
Introduction to the High Performance Computing and Communications (HPCC) Program	vii
Introduction to the Computational Aerosciences (CAS) Project	1
Overview of CAS Testbeds	3
Numerical Aerodynamic Simulation (NAS) Facility Highly Parallel Testbeds	5
Langley CAS Testbed Activities	7
LeRC Parallel Processing Testbed	9
Overview of CAS Applications Software Research	11
Parallel Graphics Libraries for Distributed Memory Architectures	13
Controls/Computational Fluid Dynamics Cross Discipline Research	15
Multidiscipline Coupling Methods	17
Design Sensitivity Analysis on High Performance Computers	19
Structural Response of Langley Mach 2.4 HSCT Model Using CFD Generated Loads	21
Structural Matrix Generator/Assembler/Solver for a Mach 2.4 Model of the HSCT	23
ADIFOR—Automatic Differentiation for Large-Scale FORTRAN Programs	25
Parallel Numerical Algorithms for Compact Difference Schemes	27
An Implicit Multizone Navier-Stokes Solver on the Connection Machine	29
Distributed Parallel Computing for Flow Simulation	31
Parallel Computer Optimization of a High Speed Civil Transport Aerobraking Grand Challenge	33
Numerical Computation of Coupled Conduction/Turbulent-Convection Heat Transfer Using MPPs	35
Parallel Generation of 3-D Unstructured Grids	37
Overview of CAS Systems Software Research	39
Optimizing Compiler Technology for Data-Parallel Languages	41
Instrumentation, Performance Evaluation, and Visualization Tools	43
HPF-based Programming Environments for Parallel Scientific Computing	45
Simulation Studies for Architectural Scalability	47
Introduction to the Earth and Space Science (ESS) Applications Project	49
Overview of ESS Testbeds	51
Access to High Performance Scalable Testbed Systems	53
Prepared for Delivery of the Cray T3D Testbed	55
Intel Paragon Testbed	57
Evaluation of High Performance Mass Storage Systems	59
ATM Network (ATDNet) Linking GSFC and Washington, DC Area Agencies	61
	63

Overview of ESS Applications Software Research	65
Grand Challenge Science Investigations	67
ESS Science Team	69
ESS Testbed Evaluation and Collaboration with NSF	71
MIMD Numerical Techniques Library	73
Parallel Computational Technique Kernels	75
ESS/JPL Software Repository	77
Overview of ESS System Software Research	79
Advanced Data Management Technologies	81
Architecture-Independent Programming Paradigms	83
HPCC Software Exchange	85
Mentat for Science Applications on the ESS/JPL Testbed	87
Parallel Unstructured Mesh Partitioner/Refiner	89
MIMD Systolic Dataflow Tools	91
Implemented Parallel Database Rendering	93
Parallel Performance and Debugging Tools	95
Ported the Visualization System FAST to Virtual Reality	97
Introduction to Basic Research and Human Resources (BRHR) Project	99
Adaptive Mesh Refinement Algorithms on Parallel Computers	101
Advanced Methods Development	103
Applications Data Movement on MPPs	105
CESDIS University Research Program in Parallel Computing	107
Third NASA Summer School in High Performance Computational Physics	109
Langley Research Center K-12 Program	111
ESS/GSFC K-12 Program	113
ESS/JPL HPCC K-12 Education Program	115
Lewis Research Center K-12 Program	117
Explorations in Supercomputing Program	119

List of Figures

NAS Parallel Benchmarks	4
Cluster Network Study	8
Sample 3-D Image Rendered with PGL	12
Controls/CFD Cross Discipline Research	14
Illustration of Multidiscipline Coupling Methods	16
Design Sensitivity Analysis on High Performance Computers	18
Structural Response of Mach 2.4 HSCT Model	20
Structural Matrix Generator/Assembler/Solver	22
Automatic Differentiation for Large-Scale FORTRAN Programs	24
Parallel Numerical Algorithms	26
Subsonic Jet 4-Zone Computation on a CM5	28
Distributed Parallel Flow Simulation	30
Optimization of a HSCT Body Using a Parallel Computer	32
Direct Simulation Monte Carlo Scale-up Results for Mach 24 Flow	34
Numerical Computation of Coupled Conduction/Turbulent Convection Heat Transfer	36
Parallel Generation of 3-D Unstructured Grids	38
Optimizing Compiler for Data-Parallel Languages	42
Handcoded vs Compiler-Generated Code for Conjugate Gradient Iteration	46
Simulation Studies for Architectural Scalability	48
HPCC/ESS Testbeds	54
JPL/Caltech Cray T3D	56
JPL ESS MIMD Testbed	58
WolfCreek™ Tape Silo	60
ATDNet Architecture	62
ESS Grand Challenge PI Investigations	66
ESS Science Team	68
ESS Evaluation Program Components Matrix	70
Performance Evaluation of Parallel Numerical Library	72
Parallel Computational Techniques	74
JPL/ESS Software Repository Front Ends	76
Advanced Data Management Technologies	80
Virtual C Translator	82
HPCC Software Exchange	84
Applying Mentat to a Finite Element Code	86
Parallel Mesh Partitioning & Refinement	88
Integrated Network Computing Environment	90
Parallel Database Rendering	92
JPL/ESS Testbed Installed Tools	94
Ported Visualization System FAST to Virtual Reality	96
Adaptive Mesh Refinement Algorithms on Parallel Computers	100
Applications Data Movement on MPPs	104
CESDIS university Research Program in Parallel Computing	106
Third NASA Summer School in High Performance Computational Physics ..	108
Langley Research Center K-12 Program	110
ESS/GSFC K-12 Program	112
JPL/HPCC K-12 Program	

List of Figures (cont.)

Lewis Research Center K-12 Program.....116
Marshall Space Flight Center Explorations in Supercomputing Program.....118

NASA Directory

<u>Center/Institute</u>	<u>Acronym</u>
Ames Research Center..... Moffett Field, CA 94035	ARC
Center of Excellence in Space Data and Information Sciences Goddard Space Flight Center Greenbelt, MD 20771	CESDIS
Goddard Space Flight Center..... Greenbelt, MD 20771	GSFC
NASA Headquarters..... Washington, DC 20546	HQ
Institute for Computer Applications in Science and Engineering..... Langley Research Center Hampton, VA 23665	ICASE
Jet Propulsion Laboratory..... Pasadena, CA 91109	JPL
Langley Research Center..... Hampton, VA 23665	LaRC
Lewis Research Center..... Cleveland, OH 44125	LeRC
Research Institute for Advanced Computer Science..... Ames Research Center Moffett Field, CA 94035	RIACS

(THIS PAGE LEFT BLANK INTENTIONALLY)

Introduction to the High Performance Computing and Communications (HPCC) Program

Program Goal and Objectives: The National Aeronautics and Space Administration's HPCC program is part of a new Presidential initiative aimed at producing a 1000-fold increase in supercomputing speed and a 100-fold improvement in available communications capability by 1997. NASA will use these unprecedented capabilities to help maintain U.S. leadership in aeronautics and Earth and space sciences by applying them to a wide range of scientific and engineering "Grand Challenge" problems. These are fundamental problems whose solutions require significant increases in computational power and are critical to meeting national needs.

The main objective of the federal HPCC program is to extend U.S. technological leadership in high performance computing and computer communications systems. Applications of these technologies will be widely disseminated within the U.S. government, industry and academia to accelerate the pace of innovation and serve the U.S. economy, national security, education and the global environment. They also will result in greater U.S. productivity and industrial competitiveness by making HPCC technologies an integral part of the design and production process.

NASA's participation in researching such advanced tools will revolutionize the development, testing and production of advanced aerospace vehicles and reduce the time and cost associated with building them. This will be accomplished by using these technologies to develop multidisciplinary models which can simultaneously calculate changes in a vehicle's fluid dynamics, structural dynamics and controls.

This contrasts with today's limited computing resources which are forcing researchers to use simple, single-discipline models to simulate the many aspects of advanced aerospace vehicles and Earth and space phenomenon. This is more costly and time consuming than simulating entire systems at once, but it has become standard practice due to the complexity of more complete simulations and the insufficient computing power available to perform them.

As more advanced technologies are developed under the HPCC program, they will be used to solve NASA's Grand Challenge research problems. These include improving the design and simulation of advanced aerospace vehicles, allowing people at remote locations to communicate more effectively and share information, increasing scientists' abilities to model the Earth's climate and forecast global environmental trends, and improving the development of advanced spacecraft to explore the Earth and solar system.

Strategy and Approach: The HPCC program was designed as a partnership among several federal agencies and includes the participation of industry and academia. Other participating federal agencies include the Department of Energy, the National Science Foundation, the Defense Advanced Research Projects Agency, the Department of Commerce's National Oceanic Atmospheric Administration and National Institute of Standards and Technology, the Department of Education, the Environmental Protection Agency and the Department of Health and Human Services' National Institutes of Health.

Together government, industry and academia will endeavor to meet program goals and objectives through a four part strategy to (1) support solutions to important scientific and technical challenges through a vigorous R&D effort; (2) reduce the uncertainties to industry for R&D and use of these technologies through increased cooperation and continued use of the government and government funded facilities as a prototype user for early commercial HPCC products; (3) support the research network and computational infrastructure on which U.S. HPCC technologies are based; and (4) support the U.S. human resource base to meet the needs of all participants.

To implement this strategy, the HPCC program is composed of four integrated and coordinated components that represent key areas of high performance computing and communications:

- ❑ High Performance Computing Systems (HPCS)—developing technology for computers that can be scaled up to operate at a steady rate of at least 1 trillion arithmetic operations per second, i.e., one teraFLOPS. Research in this area is focusing both on increasing the level of attainable computing power and on reducing the size and cost of these systems in order to make them accessible to a broader range of applications. Although these computing testbeds will not be capable of teraFLOPS performance, they can be scaled up to that level by increasing the degree of their parallelism without changing their architecture.
- ❑ Advanced Software Technology and Algorithms (ASTA)—developing generic software and mathematical procedures (algorithms) to assist in solving Grand Challenges.
- ❑ National Research and Education Network (NREN)—creating a national network that will allow researchers and educators to combine their computing capabilities at a sustained rate of one billion arithmetic operations per second (1 gigaFLOPS).
- ❑ Basic Research and Human Resources (BRHR)—promoting long-term research in computer science and engineering and increasing the pool of trained personnel in a variety of scientific disciplines.

Under the HPCS element, NASA has established high performance computing testbeds and networking technologies from commercial sources for utilization at NASA field centers. The agency is advancing these technologies through its Grand Challenge applications by identifying what needs to be developed to satisfy future computing requirements. NASA also acts as a “friendly buyer” by procuring early, often immature, computer systems, evaluating them, and providing feedback to the vendors on ways to improve them.

The agency’s role under the ASTA element consists of leading federal efforts to develop generic algorithms and applications software for massively parallel computing systems. This includes developing a Software Exchange system to produce software in a modular fashion for sharing and reuse. In this manner, complex software systems will be developed at considerably reduced cost and risk.

NASA also leads the federal venture to develop a common standard for system software and tools. In this role, NASA has organized and directed several workshops comprised of experts from industry, universities and government to review system software and tools for high performance computing environments and identify needed developments. NASA also sponsors or participates in other symposia and technical meetings which address all aspects of high performance computing. These forums not only ensure that NASA researchers remain at the cutting edge of related technologies, they accelerate the transfer of technology from NASA research efforts.

As a participant in the National Research and Education Network, NASA pursues the development of advanced networking technologies which allow researchers and educators to carry out collaborative research and educational activities, regardless of the participants’ locations or their computational resources. During FY93, the agency continued its cooperative agreement with the Energy Department for the procurement and implementation of high speed network services, including providing access to a nationwide fiber optic network that will help meet communications needs and serve as the foundation for increasingly fast networks. Future goals of NREN include developing even faster networks that operate at 155 mbps, 622 mbps and eventually at gigabit speeds.

NASA’s contribution to the Basic Research and Human Resources component includes providing funding support for several research institutes and university block grants. NASA has successfully initiated graduate research programs in HPCC technologies at five NASA centers, funded several

post-doctoral students, established a pilot NREN access project, increased the NASA "Spacelink" education bulletin board to boost internet service and established collaborative several efforts in K-12 education.

Organization: NASA's HPCC program is organized into two projects which are unique to the agency's mission: the Computational Aerosciences (CAS) project and the Earth and Space Sciences (ESS) project. NASA's participation in NREN development is matrixed across the two projects. Each of the projects is managed by a project manager at a NASA field center, while the Basic Research and Human Resources component is managed by the HPCC program office at NASA Headquarters. Ames Research Center leads the CAS project and is supported by the Langley Research Center and the Lewis Research Center. Goddard Space Flight Center serves as the lead center for the ESS project and receives support from the Jet Propulsion Laboratory. Finally, the National Research and Education Network component, which cuts across the two projects, is managed by Ames Research Center.

Management Plan: Federal program management is provided by the White House Office of Science and Technology Policy (OSTP) through the Federal Coordinating Council on Science, Engineering and Technology (FCCSET) Committee on Physical, Mathematical and Engineering Sciences (PMES). The membership of the PMES includes senior executives of many federal agencies.

Program planning is coordinated by the PMES High Performance Computing, Communications and Information Technology (HPCCIT) Subcommittee. The HPCCIT, lead by DOE, meets regularly to coordinate agency HPCC programs through information exchanges, the common development of inter-agency programs and reviews of individual agency plans and budgets.

NASA's HPCC program is managed through the agency's Office of Aeronautics and represents an important part of the office's research and technology program. The Headquarters staff consists of the director, the HPCC program manager and the manager of the Basic Research and Human Resources component. The HPCC office is responsible for overall program management at the agency, the crosscut of NASA HPCC-related programs, coordination with other federal agencies, participation in the FCCSET, HPCCIT, its Scientific and Engineering Working Group and other relevant organizations.

Points of Contact:

Lee B. Holcomb, Director

Paul H. Smith, Program Manager

Paul Hunter, Program Manager

Office of Aeronautics
High Performance Computing and Communications Office
NASA Headquarters
(202) 358-2747

(THIS PAGE LEFT BLANK INTENTIONALLY)

Introduction to the Computational Aerosciences (CAS) Project

Goal and Objectives: The goal of the CAS Project is to develop necessary computational technology for the numerical simulation of complete aerospace vehicles for both design optimization and analysis throughout the flight envelope. The goal is supported by four specific objectives:

1. Develop multidisciplinary computational models and methods for scalable parallel computing systems
2. Accelerate the development of computing system hardware and software technologies capable of sustaining a teraFLOPS performance level on computational aerospace applications
3. Demonstrate and evaluate computational methods and computer system technologies for selected aerospace vehicle and propulsion systems models on scalable, parallel computing systems
4. Transfer computational methods and computer systems technologies to aerospace and computer industries

Strategy and Approach: This research will bring together computer science and computational physics expertise to analyze the requirements for multidisciplinary computational aerospace, evaluate extant concepts and products, and conduct the necessary research and development. The steps involved include

- 1 The development of requirements and evaluation of promising systems concepts using multidisciplinary algorithms
2. The development of techniques to validate system concepts
3. The building of application prototypes to serve as proof of concept
4. The establishment of scalable testbed systems which are connected by multimegabit per second networks

Simulation of the High Speed Civil Transport (HSCT) and High Performance Aircraft (HPA) have been chosen as CAS "Grand Challenges". Langley Research Center (LaRC) is the lead center for HSCT and Ames Research Center (ARC) is the lead center for HPA. Lewis Research Center (LeRC) has the lead on modeling propulsion systems in both HSCT and HPA. Areas of interest in systems software are related to the programming environment and include user interfaces, programming languages, performance visualization and debugging tools, and advanced result analysis capabilities.

Testbeds include a Thinking Machine CM2 and CM-5 and an Intel iPSC/860, smaller iPSC/860s at LaRC and LeRC, and the Touchstone Delta at Caltech. An Intel Paragon system was installed in February 1993 and is undergoing beta systems software development

Organization: All the activities at a particular center report through the Associate CAS Project Manager at the Center to the CAS Project Manager at Ames Research Center. The CAS Project Manager is Bill Feiereisen, who reports to the HPCC Program Manager in NASA Headquarters. In addition to this organizational reporting, matrixed reporting exists across the three areas (applications, systems software, and testbeds). Bill Feiereisen is the primary contact for the CAS Project. Manuel Salas is the focal point at LaRC and Russell Claus is the focal point at LeRC. Other points of contact are in the organizational chart found in the next section.

Management Plan:

CAS Project Manager
Bill Feiereisen

ARC Associate Manger
Bill Feiereisen

Application Leader
Terry Holst

System Software Leader
Tom Lasinski

Testbed Leader
Russell Carter

LaRC Associate Manager
Manuel Salas

Application Leader
Tom Zang

System Software Leader
Andrea Overman Salas

Testbed Leader
Geoffrey Tennille

LeRC Associate Manager
Russell Claus

Application Leader
Russell Claus

System Software Leader
Gary Cole

Testbed Leader
Jay Horowitz

Point of Contact:

Bill Feiereisen
Ames Research Center
feiereis@nas.nasa.gov

Overview of CAS Testbeds

Goal and Objectives: Testbeds are where the applications, system software, and system hardware come together for testing and evaluation. The goal of the CAS testbeds research is to enable studies and experiments in integrated application environments. The objectives of the testbed work are to provide feedback to the applications, system software, and computer system developers and to point the way to the computational resources necessary to solve the CAS Grand Challenges.

Strategy and Approach: The approach is to acquire early versions of promising computer systems and map CAS applications onto these systems via the systems software. The testbeds will be upgraded as evolving technology permits and research requirements develop. In addition, access to other systems is provided by collaborative and cooperative arrangements with facilities and researchers both within and outside of NASA. An example of this is NASA's participation in the California Concurrent Supercomputing Consortium, which operates the Intel Touchstone Delta located at Caltech. The largest of the testbeds for the CAS Project will be operated by Ames Research Center. Smaller systems will be operated by Langley and Lewis Research Centers.

Organization: Ames Research Center (ARC) has a variety of testbeds, such as a Thinking Machine Corporation's CM-5, an Intel Paragon and a Gamma, and Cray Research, Inc. C-90. Both Langley (LaRC) and Lewis (LeRC) Research Centers have the commercial version of the Intel Touchstone Gamma iPSC/860 with 32 nodes. LaRC also has an Intel Paragon with 72 compute nodes (processors). A primary testbed at LeRC is an IBM RS6000 Workstation Cluster with 32 nodes of RS/6000 model 560 workstations

Management Plan: Each center has a testbed leader. These testbed leaders form a testbed working group which coordinates use and development of the testbed systems. Further information about a testbed may be sought from the center's testbed leader.

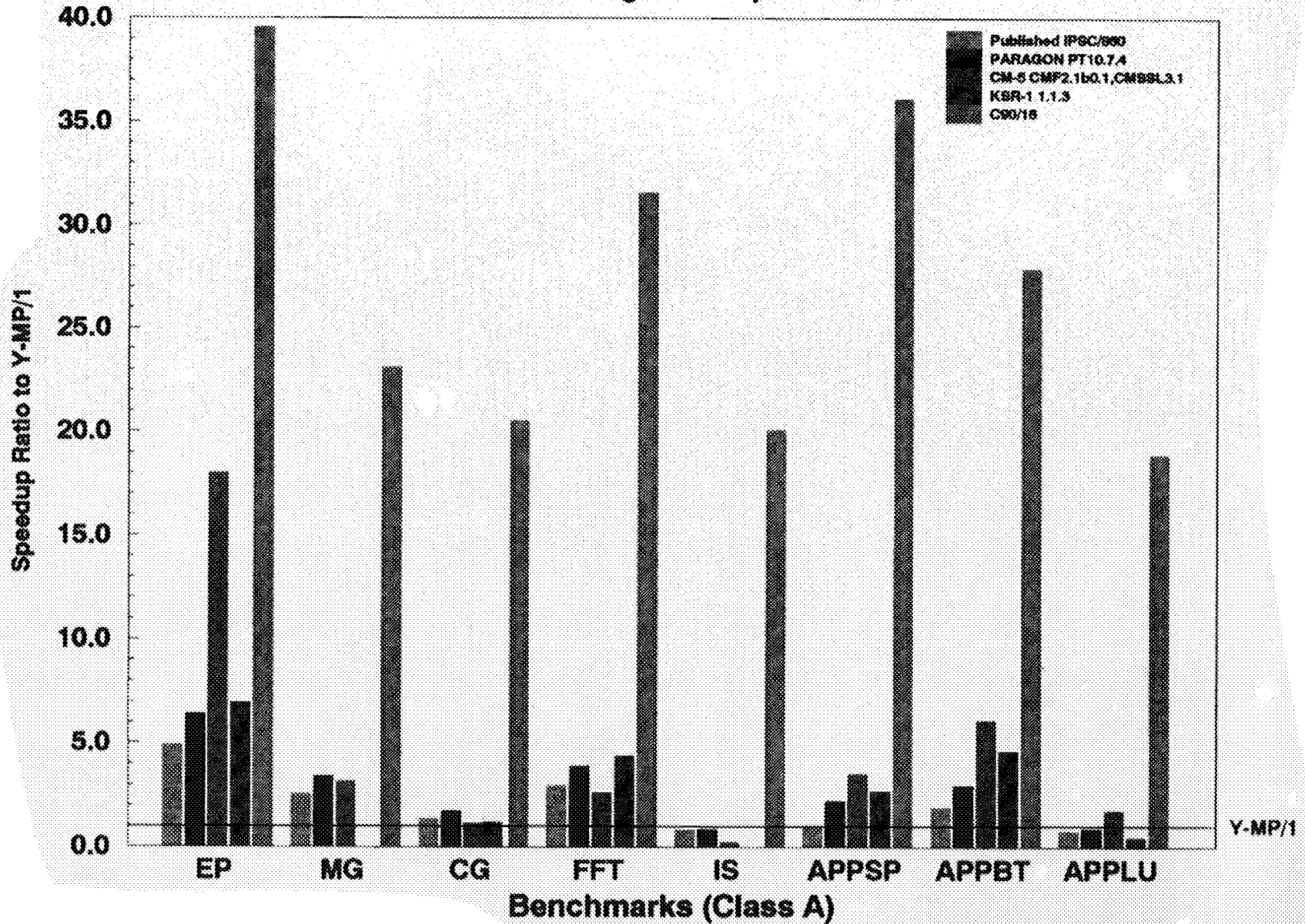
Point of Contact:

Russel Carter
Ames Research Center
rcarter@nas.nasa.gov
(415) 604-4999

Geoff Tennille
Langley Research Center
tennille@adelie.larc.nasa.gov
(804) 864-5786

Jay Horowitz
Lewis Research Center
xxjg@dali.lerc.nasa.gov
(216) 433-5194

NAS Parallel Benchmarks IPSC/860 vs. PARAGON vs. CM-5 vs. KSR-1 vs. C90/16
128 Node Configuration Updated 9/15/93



Numerical Aerodynamic Simulation (NAS) Facility Highly Parallel Testbeds

Objective: Acquire and integrate into the Numerical Aerodynamic Simulation (NAS) Support Processing Network highly parallel testbeds capable of demonstrating scalable performance on CAS workloads.

Approach: Testbeds are to be acquired through NASA procurements. The first such testbed is denoted the HPCCPT-1, scheduled to be installed in the first quarter of FY94. Integration of the testbeds is undertaken by the Parallel Systems Development Group of NAS/RND. This group has responsibility for supplying necessary system software.

Accomplishments: The NAS CM-5 and Intel iPSC/860 were successfully integrated into the NPSN. An Intel Paragon system was installed in February 1993 and is undergoing beta systems software development under a collaborative agreement between NAS and Intel SSD.

The progress of each system is monitored through a comprehensive set of availability, stability, usage, and performance metrics. A set of input and output benchmarks that mimic NAS parallel Computational Fluid Dynamics input and output requirements have been constructed and implemented on the NAS Parallel Systems.

Significance: NAS is the first center to integrate highly parallel testbeds into a production environment and to subject these systems to a large daily workload of multiple scientific users.

Status and Plans: Three highly parallel testbeds are now installed at NAS. Current plans call for the installation HPCCPT-1 in the first quarter of FY94 and a highly parallel production system capable of processing a substantial quantity of the NAS workload in the first quarter of FY95. NAS plans to pursue collaborative agreements with highly parallel systems vendors to improve system software as required to support computational science workloads.

Point of Contact:

Russell Carter
Parallel Systems Development
NAS/RND
Ames Research Center
rcarter@nas.nasa.gov
(415) 604-4999

(THIS PAGE LEFT BLANK INTENTIONALLY)

Langley CAS Testbed Activities

Objective: Acquire the most powerful and cost-effective parallel computer as a testbed for developing parallel multidisciplinary design and optimization applications of advanced aerospace vehicles. Provide for system administration, management, and maintenance of the testbed computer. Develop strategies for efficient management that drive the system toward a user-friendly production environment. Report and categorize software deficiencies to the vendor for resolution. Evaluate and procure third party software packages to enhance user productivity.

Approach: Use the Small Business Administration (SBA) 8(a) set-aside program for procuring the testbed. Leverage software evaluation with experience on parallel computers at other NASA sites, the existing Intel iPSC/860 at Langley and networked workstations. Conduct acceptance testing on all delivered hardware and software. Train users in the transition of applications to the new architecture.

Accomplishments: A 66-node Intel Paragon XP/S-5 with 38 Gbytes of disk and 16 Mbytes of memory/node was acquired in the second quarter of FY93. In the third quarter FY93 the Beta release software was upgraded to production release and the memory was increased to 32 Mbytes/node. The hardware was conditionally accepted in the fourth quarter. These activities accounted for all FY92 funds and a significant portion of FY93 funds.

High Performance FORTRAN (HPF) preprocessors were acquired to convert code to and from FORTRAN 77 along with software to emulate shared memory on a cluster of workstations. Also, software to facilitate management and administration of the XP/S-5 was developed and validation of XP/S-5 users was begun. The testbed performed in excess of 2 gigaFLOPS on 66 nodes for the LINPACK benchmark, and interprocessor communications speed was 24 Mbytes/second. A NASA Technical Memorandum on parallel software tools at Langley was completed.

Significance: Although the system software is far from robust, most codes, including the NAS parallel kernels, have demonstrated better performance on the XP/S-5 than they did

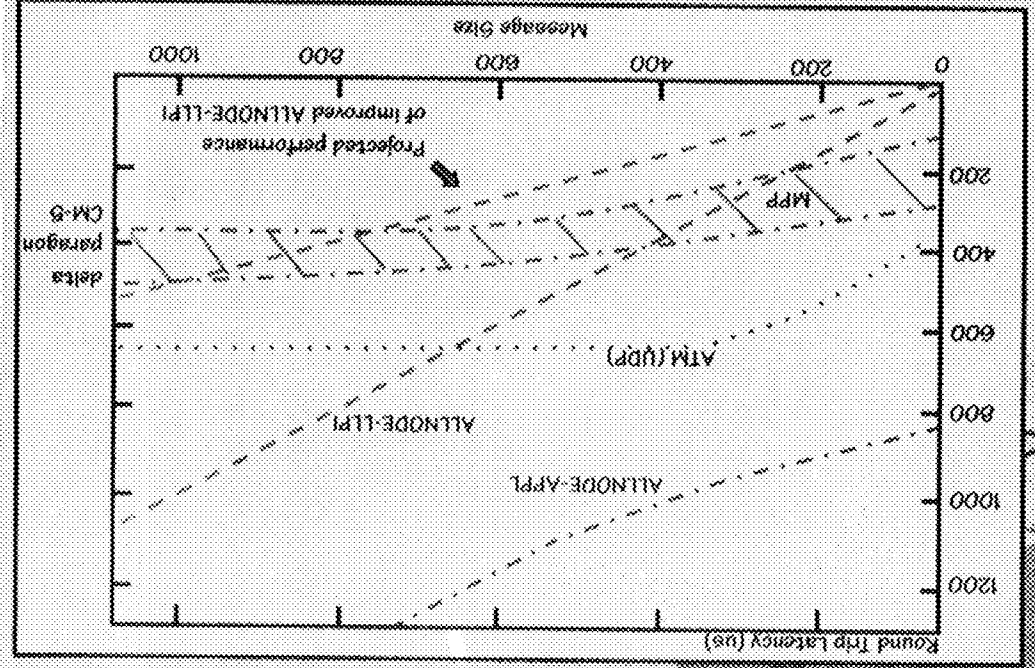
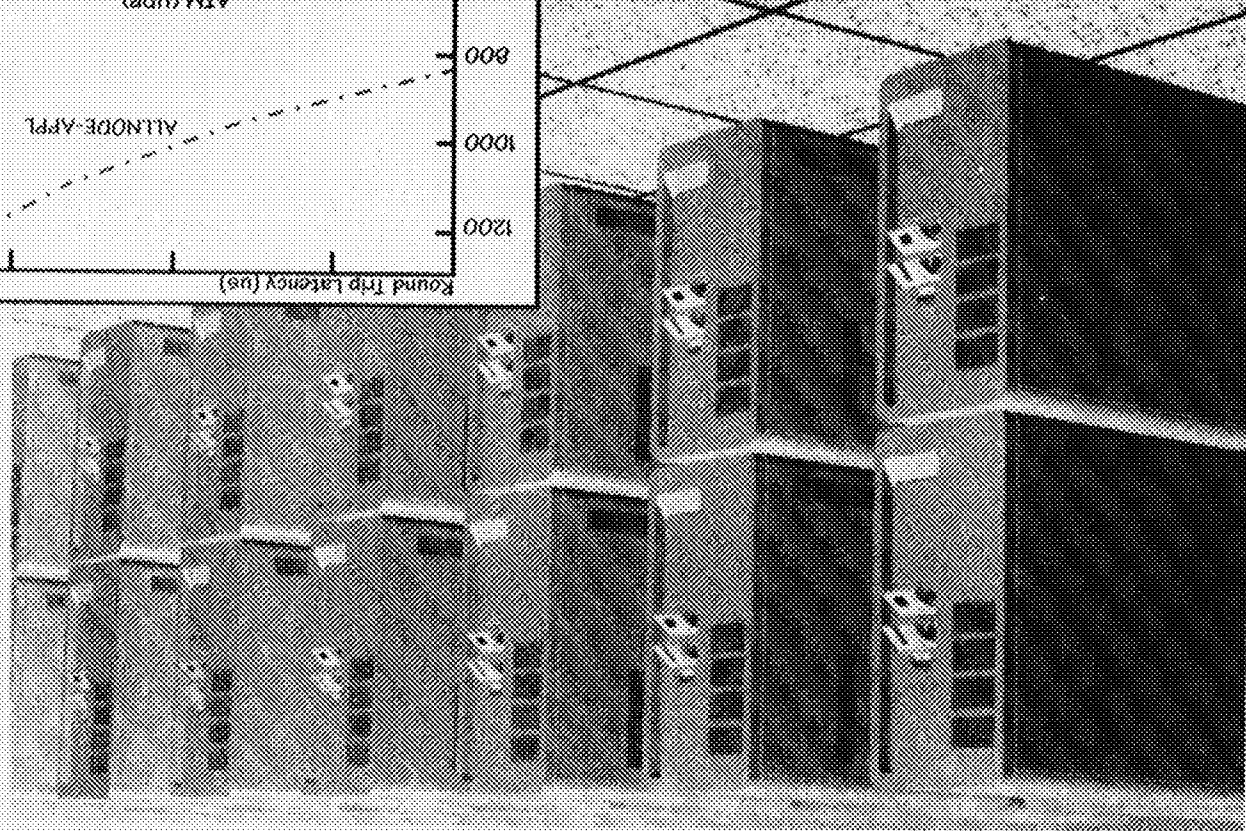
on the iPSC/860 with a comparable number of nodes. Since installation, the system software has demonstrated steady improvement in stability and performance. The architecture is reliable and scalable—if sufficient care is taken in the coding of an application.

Status/Plans: Complete acceptance of the Paragon in early FY94. With improved stability, port third-party software like PVM, Linda and Express to the XP/S-5. Evaluate graphical user interface performance analysis and debugging utilities that will be delivered in the first quarter of FY94. Conduct training for users and complete the transition from the iPSC/860. Upgrade the computational capacity for the testbed through an augmentation of the XP/S-5 or procurement of a cluster of workstations.

Point of Contact:

Geoffrey M. Tennille
NASA Langley Research Center
tennille@adelle.larc.nasa.gov
(804) 864-5786

Cluster Networking Study



LeRC Parallel Processing Testbed

Objective: To establish a testbed for early evaluation of parallel architectures responsive to the computational demands of the Lewis propulsion codes.

Approach: A localized cluster of high-end IBM workstations has been assembled and configured to provide for distributed memory, MIMD parallel processing and distributed processing applications. Internode traffic can be carried via ethernet or a low-latency crossbar switch, called ALLNODE.

Accomplishments: A highly flexible configuration of clustered IBM RISC systems has been designed, tested and applied to various industrial codes. The 32-node cluster contains 32 IBM Model 560 RISC systems, each with a minimum of 64 MBytes of memory, a 1 GByte disk, and a CPU benchmarked at 30.5 megaFLOPS (LINPAC). Some nodes will have expanded memory (4 with 128 MBytes, 2 with 512 MBytes). An IBM Model 970 with a 6 GByte disk serves as a resource manager. The cluster has an aggregate maximum of approximately 1 gigaFLOPS performance.

Each node is interconnected using either a dedicated ethernet or a low-latency crossbar switch (ALLNODE). The ALLNODE switch was tested in several applications and demonstrated significantly reduced latencies (approximately one-fourth to one-eighth) and greater bandwidth (by approximately a factor of four) than did a dedicated ethernet. Early Asynchronous Transfer Mode (ATM) hardware was tested and displayed substantial promise for non-local clustering.

Significance: The RISC Cluster will provide early evaluation of the IBM massively parallel processor environment that is intended to provide scalable teraFLOPS systems by mid-decade. In addition, the cluster is well-suited to NASA Lewis' multidisciplinary approach to aeropropulsion simulation. Different modules of the simulation (e.g., inlet, combustor, etc.) can run on different nodes of the cluster, some possibly parallelizable, others potentially requiring nodes with more memory.

Status/Plans: Networking configurations including the low-latency ALLNODE switch and ATM have been shown to substantially improve the performance of parallel applications on the cluster. Future enhancements to these network approaches may yield communication speeds approximating the current generation of massively parallel processors. The clustering concept will be further explored as an affordable and functional parallel processor.

Point of Contact:

Jay G. Horowitz
Lewis Research Center
xxjg@dali.lerc.nasa.gov
(216) 433-5194

(THIS PAGE LEFT BLANK INTENTIONALLY)

Overview of CAS Applications Software Research

Goal and Objectives: The goal of the CAS Applications Software research is to develop multidisciplinary computational models and methods for two Grand Challenge applications: the optimization of a High Speed Civil Transport (HSCT) and the optimization and analysis of High Performance Aircraft (HPA). The objectives for the HSCT Grand Challenge are to develop

1. Accurate and efficient transonic-to-supersonic cruise simulation of a transport aircraft on advanced testbeds
2. Efficient coupling of the aerodynamic, propulsion, structures and controls disciplines on advanced testbeds
3. Efficient implementation of multidisciplinary design and optimization on advanced testbeds.

The objectives for the HPA Grand Challenge are to develop

1. Efficient simulation of low-speed, maneuver flight conditions on advanced testbeds
2. Efficient coupling of the aerodynamic, propulsion, and control disciplines on advanced testbeds

Strategy and Approach: In the HSCT applications research, the disciplines of aerodynamics, structural dynamics, combustion chemistry, and controls will be integrated in a series of computational simulations about a supersonic cruise commercial aircraft. Although some unsteady computations such as transonic flutter prediction will be performed, the bulk of the computations associated with the HSCT research effort emphasize steady cruise conditions. For the HPA research the disciplines of aerodynamics, thermal ground plane effects, engine stability, and controls will be integrated in a series of computational simulations about a high performance aircraft undergoing a variety of maneuver conditions.

Organization: The HSCT research is being performed jointly by Ames Research Center (ARC), Langley Research Center (LaRC), and Lewis Research Center (LeRC). The Ames and Langley Research Centers will perform various computations associated with the airframe, and the Lewis Research Center will work on the propulsion elements. The overall lead center for the HSCT effort is LaRC.

Ames Research Center is the overall lead center for the HPA applications research. The research is being performed jointly by ARC and LeRC. ARC performs the various computations associated with the airframe, and LeRC will be in charge of the propulsion elements. Two general research areas are associated with this Grand Challenge: a powered lift application and other HPA simulations.

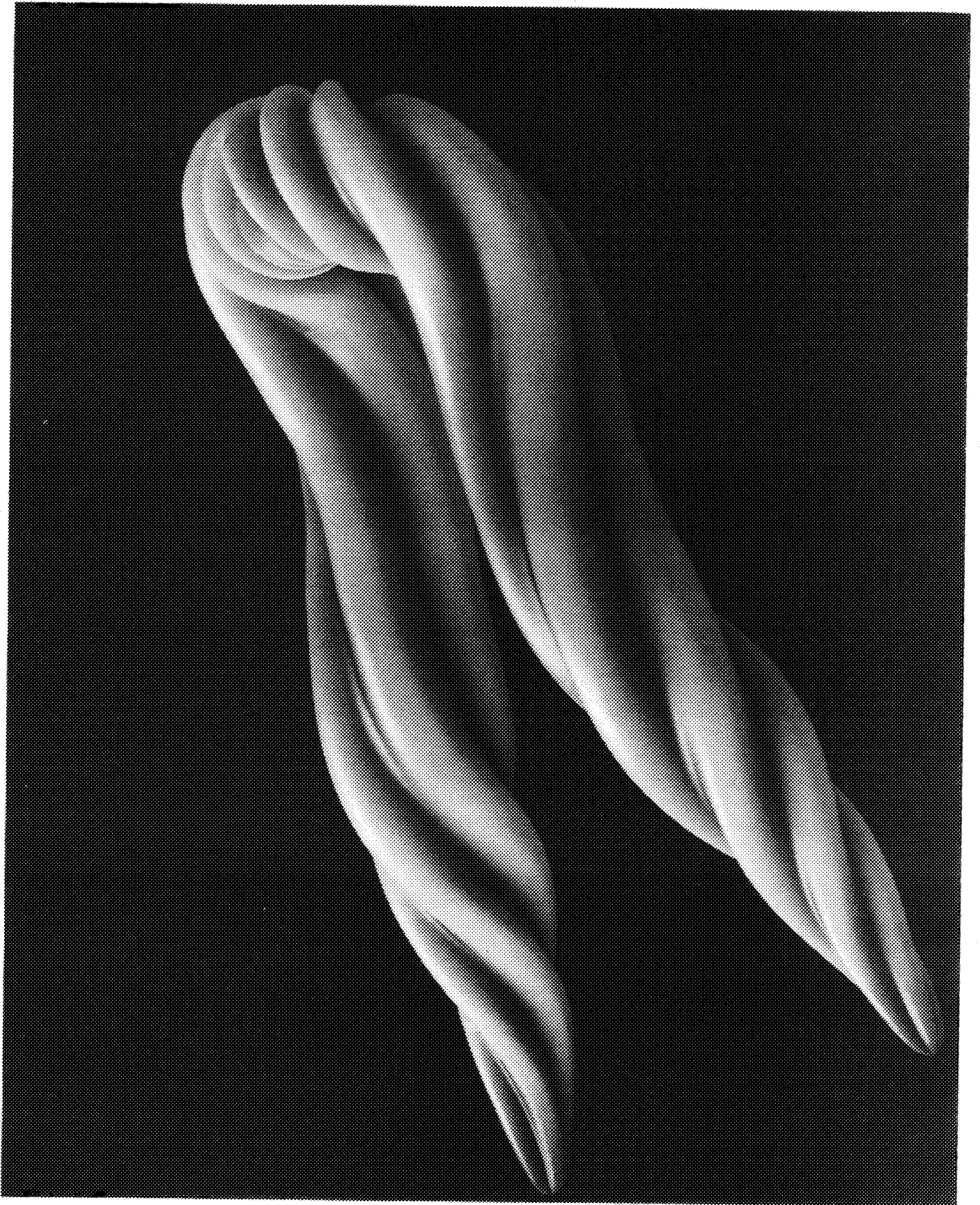
Management Plan: Three CAS application research leaders, one at each of the participating NASA Centers, report to the CAS Project Manager. These applications leaders form an applications working group which coordinates the development of CAS grand challenge applications.

Point of Contact:

Tom Zang
Langley Research Center
zang@tab00.larc.nasa.gov
(804) 864-2307

Russell Claus
NASA Lewis Research Center
r_claus@comsrv.lerc.nasa.gov
(216) 433-5869

Terry Holst
Ames Research Center
terry_holst@qmgate.arc.nasa.gov
(415) 604-6032



Parallel Graphics Libraries for Distributed Memory Architectures

Objective: To develop a parallel 3-D graphics library suitable for use by scientific computations which run on massively parallel computers.

Approach: Previous work developed a parallel polygon rendering algorithm suitable for MIMD message-passing architectures. This project extends that work by incorporating an improved version of the algorithm in a parallel 3-D graphics library, with significant improvements in functionality and performance. The graphics library is specifically designed to be callable from parallel application programs which employ an SPMD programming style. The resulting images may be compressed (in parallel) for live transmission to remote displays or saved to files for subsequent viewing.

Accomplishments: A parallel 3-D graphics library (PGL) has been developed which supports a standard graphics pipeline, including modeling transformations, lighting calculations, 3-D clipping, perspective viewing, interpolated shading, and z-buffered, hidden-surface elimination. The library incorporates an improved, span-based version of an earlier rendering algorithm. PGL currently runs on iPSC/860 and Paragon systems, but is designed for portability to other message-passing platforms. Sequential versions are also provided for Sun and SGI workstations. Preliminary performance results indicate that the new renderer is approximately twice as fast as the original. Parallel efficiency is doubled as well, reaching 60% on a 128-processor iPSC/860 using a uniform test scene.

Experiments also indicate that the compressed image transmission scheme used by PGL is practical. Local transmission times within Langley Research Center are on the order of one second per frame using Ethernet and 8-bit color. Cross-country transmission from NASA Ames Research Center to Langley is somewhat slower at about three seconds per frame. While not highly interactive, these rates can be expected to improve with faster networks and better network interfaces.

The page opposite shows a sample image rendered with PGL using Langley's 66-node Paragon (dataset courtesy of D. Banks, B. Singer, and R. Joslin). The image contains approximately 245 000 triangles, and depicts a vortex tube produced by a pulse injected into laminar flow over a flat plate. At 1000 x 800 resolution with 24-bit color, this image can be rendered, transmitted to a workstation, and displayed in three to five seconds.

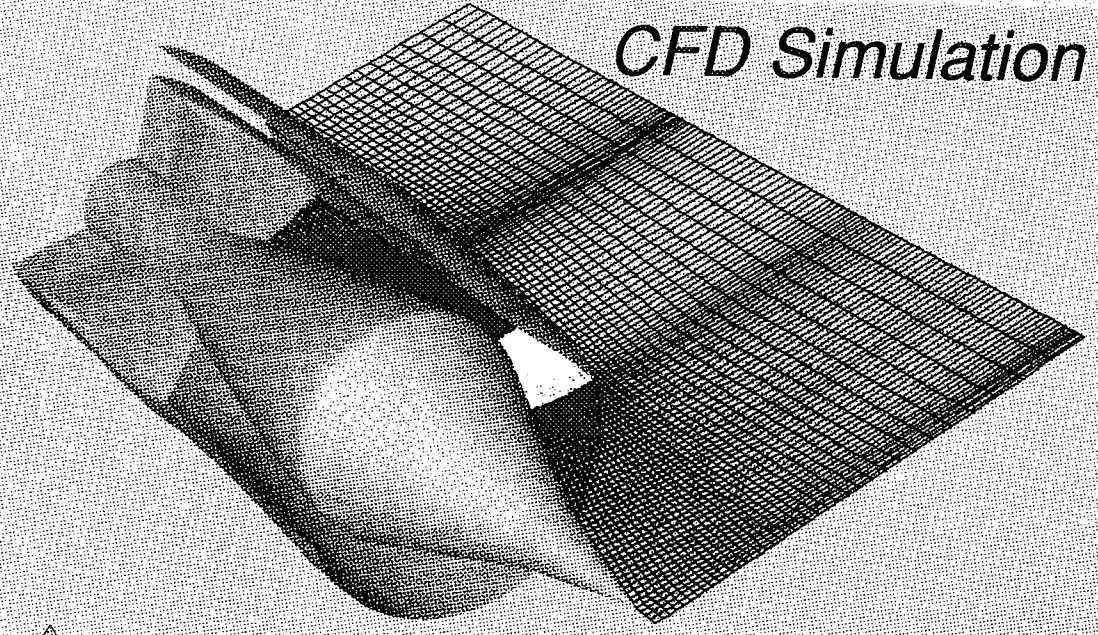
Significance: Applications that run on massively parallel computers often produce massive output datasets. Because of their size, these datasets can be cumbersome to move across the network for post-processing. By using the power of the parallel machines to perform graphics and visualization operations in place, the output data stream is reduced to a manageable size. The ability to embed graphics calls within parallel applications allows users to employ visualization techniques for debugging, execution monitoring, interactive steering, and exploration of large datasets.

Status/Plans: Detailed performance evaluation of the span-based rendering algorithm used in PGL is underway, with particular emphasis on load balancing and scalability. Of particular interest is performance as the number of processors approaches or exceeds the number of scanlines in the image. Within the next year, PGL should be integrated into one or more applications of interest to the Computational Aerosciences community. A documented, distributable version of the library will be made available as time permits. Also, a port to the CM-5 is being considered.

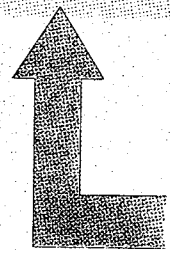
Point of Contact:

Thomas W. Crockett
ICASE
NASA Langley Research Center
804/864-2182
tom@icase.edu

**CONTROLS/CFD
CROSS DISCIPLINE RESEARCH**



**Model
Reduction
Process**



**Modeling
Parameters**

Set
Point +



-

**Control
Laws**

**Reduced
Order
Model**

Feedback

**Sensor
Models**

Measurements

Control Design

Controls/Computational Fluid Dynamics Cross Discipline Research

Objective: Create a cross-disciplinary "bridge" between control and CFD technologies by developing analytical methods and software algorithms which provide

1. An efficient interface to computational fluid dynamics (CFD) simulations
2. A capability for extracting reduced order, time accurate models from CFD results.

Approach: Develop analytical methods and software tools in the three areas below and as depicted in the figure on the opposite page:

1. Methods of extracting reduced-order models from multidimensional CFD simulations require reducing the large number of states to a "practical" size while retaining the essence of the physics
2. Software interface(s) to facilitate effective and interactive use of CFD as a numerical "experiment" for obtaining information that will significantly improve the model extraction process
3. CFD code that is parallelized to provide practical run times and has time-accurate capabilities compatible with the "controls" problem, such as: perturbation of inlet and exit boundary conditions, moving geometry, and models for bleed and bypass conditions

Accomplishments: In FY93 a grant was initiated with the University of Akron for "Interdisciplinary Research in Computational Fluid Dynamics and Control Systems" (NAG3-1450). Controls/CFD interface requirements have been defined. A distributed computing approach for implementing the Controls/CFD interface was used to demonstrate interactive execution of an unsteady, 1-D CFD simulation of an High Speed Civil Transport (HSCT) inlet. Modifications were made to enhance the PARC CFD code (2-D and 3-D), including the enforcement of uniform steps in space and time required for time-accurate operation. Numerical studies were performed to investigate and validate PARC2D capabilities using a shock tube problem and also a Mach 2.5 mixed-compression supersonic-inlet unstart transient. A typical HSCT inlet configuration was selected

in conjunction with the Boeing Company as the initial application for the Controls/CFD bridge technology.

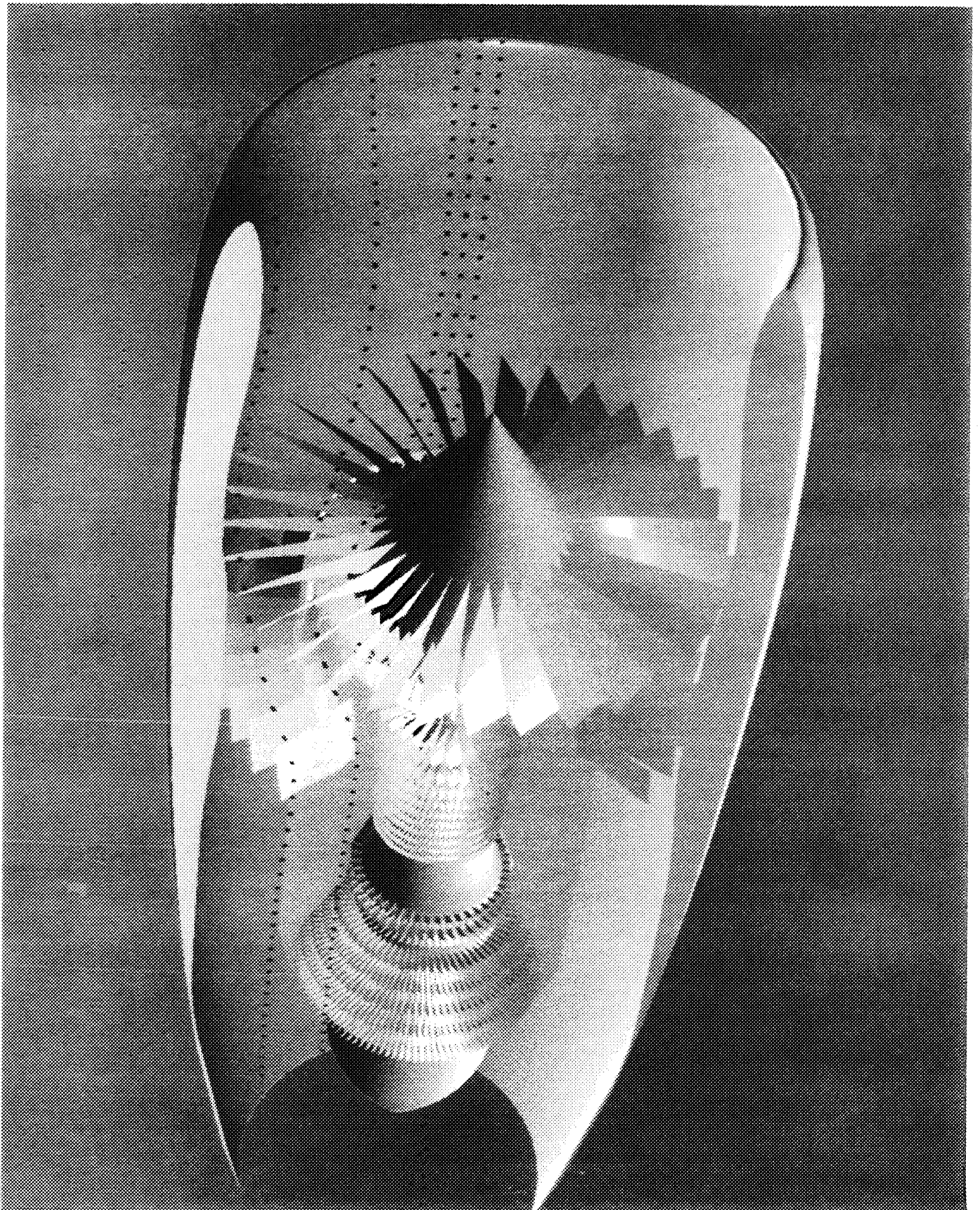
Significance: Definition of the Controls/CFD interface requirements and the ICE/APPL software interface demonstration provided the necessary groundwork for development of a Controls/CFD interface that will execute in a distributed parallel computing environment. PARC modifications and numerical studies were critical to demonstrating its suitability as a CFD code for doing the required unsteady calculations.

Status/Plans: Model reduction techniques are being studied which can be applied to a 1-D CFD simulation of the selected HSCT inlet. Consideration is being given to how these methods may be extrapolated for use with multidimensional CFD results. Requirements for the Controls/CFD interface are being translated into a knowledge base needed for use with the ICE software. An initial demonstration of the Controls/CFD interface is planned for early in FY94. Parallelization of the PARC code (both 2-D and 3-D) is underway. A blocked 3-D grid is being generated for parallel execution of the selected HSCT inlet configuration.

Point of Contact:

Gary Cole (CFD/Interface)
NASA Lewis Research Center
glc@bigfoot.lerc.nasa.gov
(216) 433-3655

Kevin Melcher (Controls)
NASA Lewis Research Center
melcher@lims01.lerc.nasa.gov
(216) 433-3743



Multidiscipline Coupling Methods

Objective: Develop the methodology and respective computer codes to computationally simulate the naturally coupled multidisciplinary interaction of various participating disciplines inherent in aerospace propulsion structural systems. Explore the feasibility of implementing this methodology on parallel processing platforms.

Point of Contact:

Christos C. Chamis
Structures Division
Lewis Research Center
(216) 433-3252

Approach: Three alternative methods will be developed and implemented in parallel processing platforms to couple the participating disciplines in the system array of mutual interactions:

1. Coupling by selective iteration
2. Coupling by specially derived matrices
3. Coupling at the fundamental formulation level

Accomplishments: The coupling coefficients (off-diagonal) terms in the system array of mutual interactions are being evaluated by optimizing each discipline separately and computing the mutual influence for the other participating disciplines. The mutual influence between disciplines are influence coefficients for linear problems and influence functions for nonlinear problems.

An initial example is a blade made from composite materials and subjected to multidiscipline loads (thermal, structural, acoustic and electromagnetic).

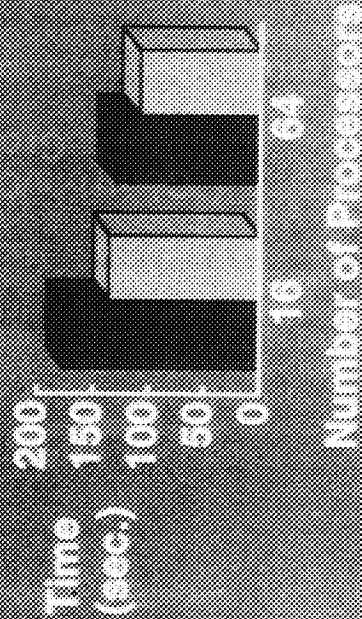
Significance: By using this approach, the mutual influence of the various disciplines is obtained without recourse to specialty computer codes. Substantial effort in problem set-up and computer solution is saved.

Status/Plans: The results for the mutual influence array are now evaluated to determine sensitivities. Initial results from the fundamental formulation and the implementation on parallel processing platforms demonstrate that both of these are doable.

Design Sensitivity Analysis on High-Performance Computers

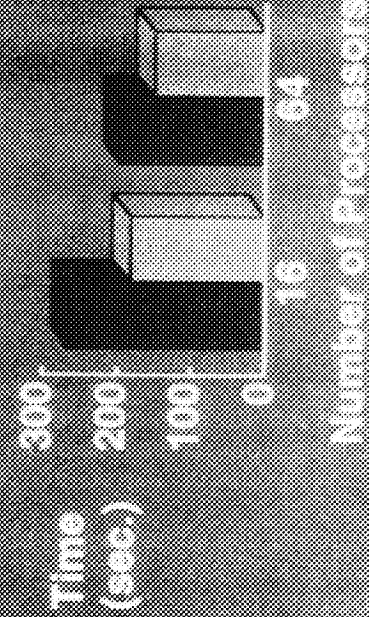
1 Design Variable
Thickness of skin,
sparse and ribs

■ Finite Difference
□ Direct Method



2 Design Variable
Thickness of skin
Thickness of spars and ribs

■ Finite Difference
□ Direct Method



Design Sensitivity Analysis on High Performance Computers

Objective: Using the Langley Mach 2.4 High Speed Civil Transport (HSCT) structural model, develop and validate an efficient algorithm for design sensitivity analysis calculations on distributed memory high performance computers.

Approach: Optimization algorithms are used to achieve a minimum weight design for the Mach 2.4 HSCT under flight conditions. A critical aspect of optimization algorithms is the calculation of design sensitivities. In this research, design constraints are assumed to be imposed on the displacement of the wing (e.g., maximum wing tip displacement). Hence, the design sensitivity sought—which must be accurately and efficiently calculated—is the derivative of the displacements with respect to the design variables, such as skin, spar and rib thickness. Calculation efficiency is especially important because in the optimization algorithm thousands of sensitivity calculations must be performed. Also, to establish baseline results, a finite element analysis of the baseline Mach 2.4 HSCT model is performed. All analyses are performance optimized and conducted on a massively parallel Intel i860 computer with 64 processor nodes.

Design sensitivity analysis can be performed by using either an approximate finite difference method or by using a direct, exact method based on solving the structural adjoint equations. This research used central differencing as the finite difference method, and the results were compared with those from the direct adjoint method. In the finite difference method all processors perform a finite element analysis in parallel, once for the design variable value (e.g., skin thickness) and once for its increment. The displacement results from the two finite element analyses are then used to obtain a numerical sensitivity derivative. In the direct adjoint method, the analysis is performed only once, regardless of the number of design variables. The derivative of the element stiffness matrix with respect to the design variable is computed analytically for each element and assembled in parallel using a new, efficient nodal assembly approach. Forward and backward substitution is then performed to determine the sensitivity for each design variable.

Accomplishments: Design sensitivity analysis was conducted using both the finite difference method and the direct adjoint method. The figure on the opposite page shows the computation times on the Intel i860 computer using each method. Timing results are shown for calculating sensitivities for one design variable, (common thickness of skin, spars and ribs), and for two design variables, (thickness of skin and common thickness ribs and spars). The results indicate that both methods are somewhat scaleable since computation time reduces for each as the number of processors increases from 16 to 64. Also, though not shown, the sensitivity values calculated by the two methods were essentially the same, thus adding confidence to the accuracy of the methods.

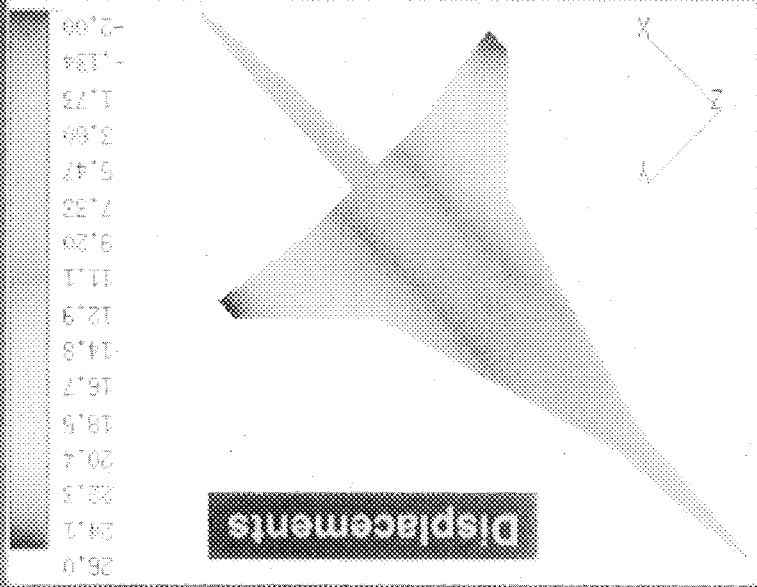
Significance: To conduct a full design optimization, the design sensitivity analysis must be performed for many design variables; hence their efficient calculation is required. The results indicate that massively parallel computers have the potential for reducing optimization computer time and that the direct method for design sensitivity calculation should be used when possible.

Status/Pans: The direct method will be applied to handling a large number of design variables. The method then will be ready for use in optimal design of structural design variables. In addition, the calculation of sensitivity derivatives with respect to geometric (e.g., airfoil shape) design variables also will be developed.

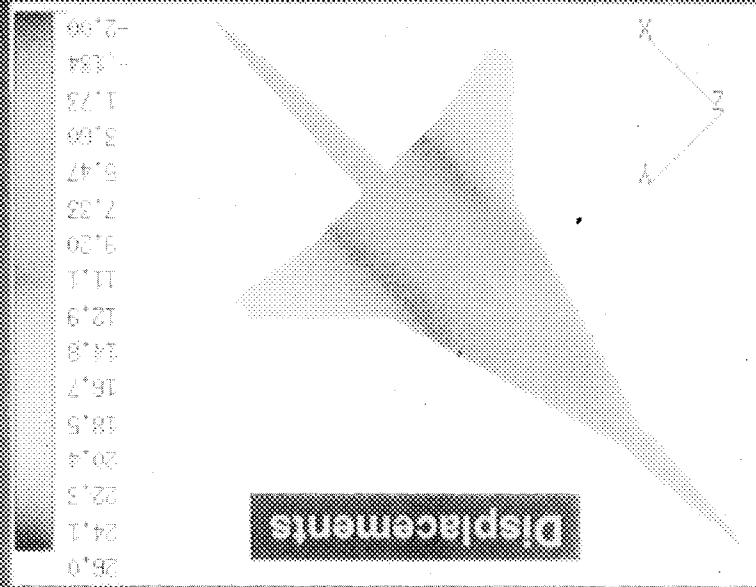
Point of Contact:

Olaf O. Storaasli
NASA Langley Research Center
Olaf@larc.nasa.gov
(804) 864-2927

Structural Response of Langley Mach 2.4 HSCT Model Demonstrated Using CFD Generated Loads



Displacements due to aerodynamic loads (Nose at panel beaming)



Displacements due to aerodynamic loads (Nose at node beaming)

Structural Response of Langley Mach 2.4 HSCT Model Using CFD Generated Loads

Objective: Develop and demonstrate methodology for applying aerodynamic loads generated by Computational Fluid Dynamics (CFD) analyses to finite element structural models of the Langley Mach 2.4 High Speed Civil Transport (HSCT).

Approach: Separate models are used for structures and aerodynamic calculations, with each model tailored to its respective discipline's computational needs. Aerodynamic pressure distributions are computed using a 3-D Navier-Stokes code, ENS3DAE. The surface pressures obtained from the code are converted to loads and transferred to the structures model using various conservative load beaming algorithms. Structural deflections are then computed for each beaming algorithm using the Computational Mechanics Testbed (COMET) and the accuracy compared.

Accomplishments: Aerodynamic and structural models of the Langley Mach 2.4 HSCT were generated. Loads at supersonic cruise were computed using the ENS3DAE code and compared with a similar CFD method in a code known as TLNS3D to ensure that accurate pressures were being predicted. A number of load transfer schemes were investigated, and finally two schemes, a nearest panel scheme and a nearest node load beaming scheme, were selected as candidates for coupling ENS3DAE and COMET. The ENS3DAE code was modified for execution on a shared memory parallel computer (the COMET code had previously been modified). Computations were performed using an eight-processor CRAY Y-MP computer. As shown in the figure on the opposite page, the wing deflections using the two methods were comparable, but the nearest panel method displayed a better load transfer capability.

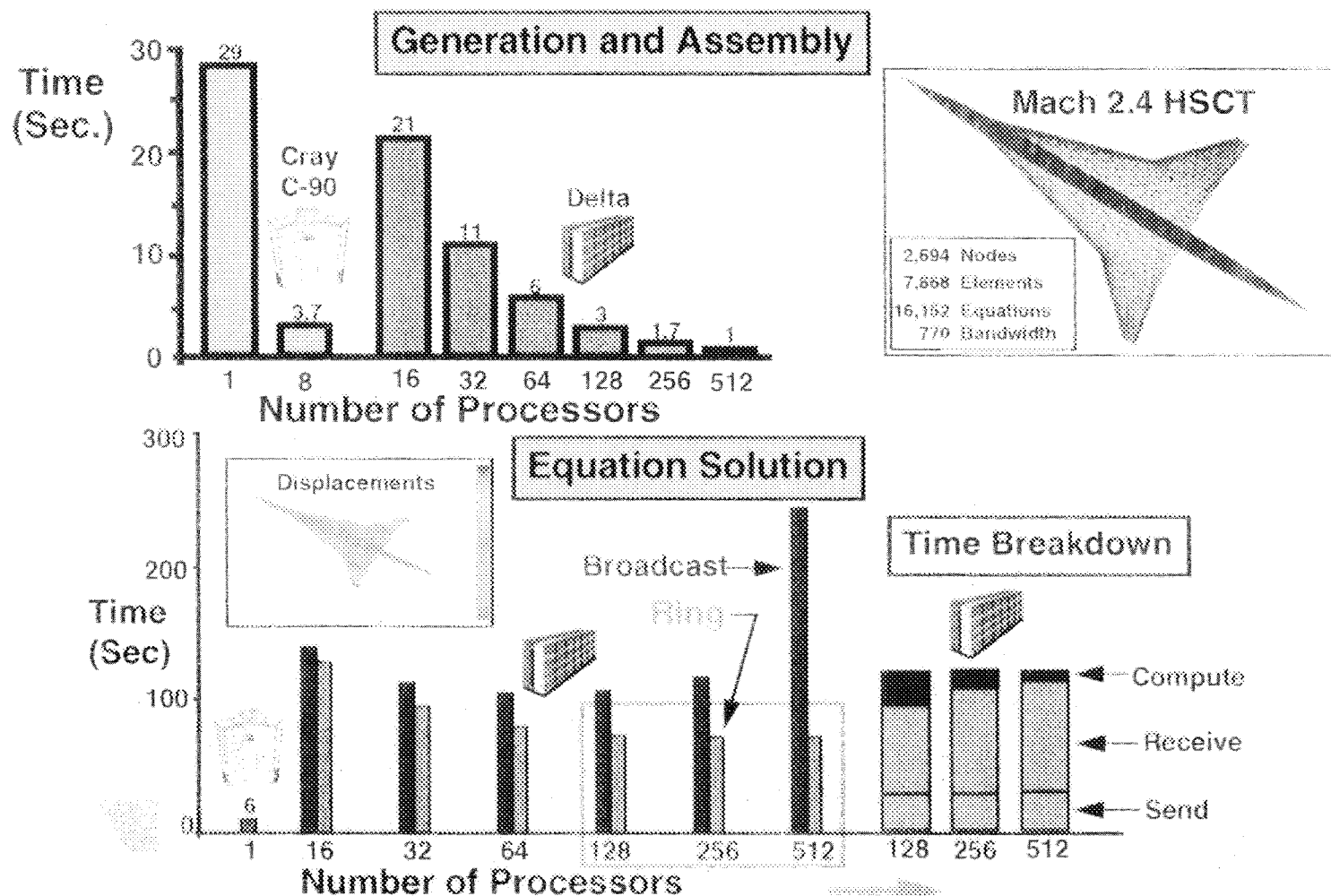
Significance: The results show the potential for a point design capability on a high performance computer using realistic loads generated by a 3-D CFD aerodynamic code and a general purpose 3-D structural finite element code. Beaming methods for transforming CFD generated pressures to structural loads have been successfully accomplished.

Status/Pans: The beaming methods will be used to couple CFD and FEM codes on massively parallel computers. Additionally, other methods of transferring loads will be investigated and compared with the nearest panel load beaming method. The integration of ENS3DAE and COMET will be extended by tightly coupling the two codes through a feedback loop in which the displacements generated by the finite element analysis are applied to the aerodynamic model and grid. The displaced grid is then used in the aero code to predict new structural loads. The process is repeated until convergence. The integration of ENS3DAE and COMET will provide a tool that allows detailed static aeroelastic analyses of realistic aircraft geometries and structural components. In addition, this integrated capability will be used to perform multidisciplinary design optimization of aerospace vehicles.

Point of Contact:

Olaf O. Storaasli
NASA Langley Research Center
Olaf@larc.nasa.gov
(804) 864-2927

Structural Matrix Generator/Assembler/Solver for High-Performance Computers Demonstrated on HSCT



Structural Matrix Generator/Assembler/Solver for a Mach 2.4 Model of the HSCT

Objective: Develop algorithms for massively parallel computers which make tractable the generation, assembly, and solution of structural matrices arising in the design of complex structural systems.

Approach: The solution of matrix equations and the generation and assembly of stiffness and mass matrices dominate finite element analysis solution time. In this generation and assembly an element stiffness matrix is associated with many nodes. Conventional finite element codes, executing on sequential computers, use an element-by-element algorithm to generate and assemble stiffness and mass matrices. This conventional procedure is parallelized by distributing element stiffness calculations among different processors. But, the results are disappointing because synchronization is required, thus increasing computation time. Synchronization of processors is required because the conventional procedure attempts to simultaneously add the stiffness matrix contributions from different processors assigned to each structural element connected to the same structural node. To overcome this problem, a parallel node-by-node stiffness and mass matrix generation and assembly algorithm was developed to distribute nodal, rather than element, calculations to different processors. The algorithm's parallel performance on a Langley Mach 2.4 High-Speed Civil Transport (HSCT) model (see figure on opposite page) was evaluated on a 512-processor Intel Delta computer and the results compared with those from a Cray C-90 computer. Once the matrices and load vector were generated, new methods to solve the matrix equations for structural displacements were developed.

Accomplishments: A parallel node-by-node element generation and assembly algorithm was developed and tested for six structural applications on the Intel Delta computer. As the figure shows, the algorithm's performance was found to be scalable (i.e., computation time reduces in direct proportion to the number of processors). This is a highly desirable result, but difficult to obtain on massively parallel computers. This scalability, achieved for both the Cray C-90 and the Delta computers, was achieved by replacing the communication-intensive element-by-element algorithm with the node-by-node algorithm to

eliminate interprocessor synchronization issues. The new equation solution algorithm also demonstrates scalability for computation time, but not for communication time as seen in the figure. Consequently, for the HSCT application execution time is nearly constant.

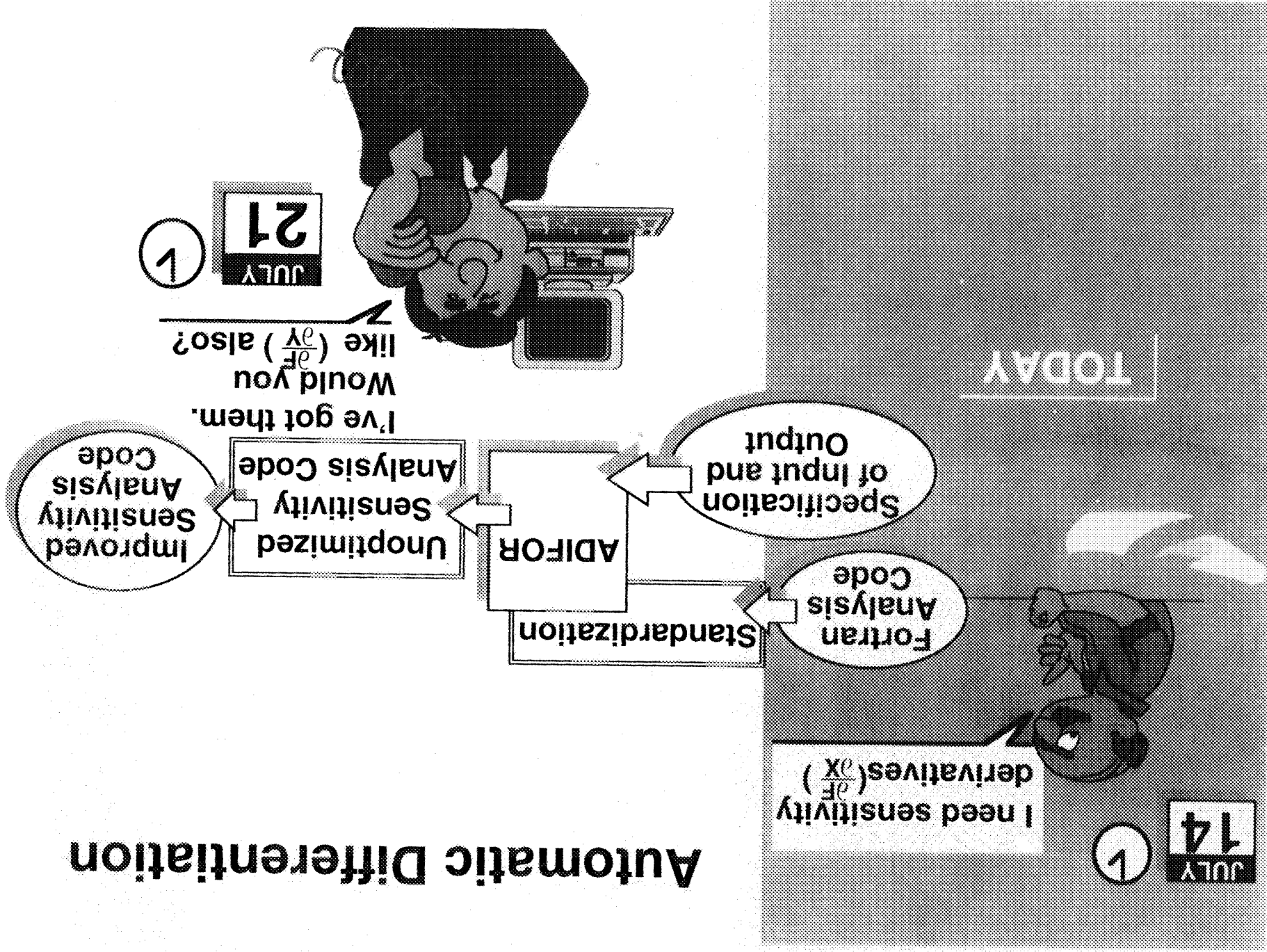
Significance: The parallel node-by-node generation and assembly algorithm is the first structural analysis method known to execute significantly faster on a massively parallel computer than on a Cray. The algorithm markedly improves computation speed for element generation and assembly as the number of processors increases. The algorithm is well-suited for applications for which the global stiffness and mass matrices are calculated repeatedly, such as structural optimization, nonlinear static and dynamic analysis, and panel flutter. The algorithm continues to perform well as the size and complexity of the structural model increases. The equation solver also is scalable when actual computation time is considered. However, due to latency and relatively slow communication time on the Delta, the scalability of the algorithm computation time is hidden. But, when the expected fast communication is available on the Intel Paragon and other scalable computers, the resulting total solution time also should be scalable.

Status/Plans: The versatility of the algorithm is being tested on HSCT models from 16 000 to 172 000 equations for newer computer architectures, such as the Intel Paragon, Thinking Machine's CM-5, and IBM's SP-1 and SP-2.

Point of Contact:

Olaf O. Storaasli
NASA Langley Research Center
Olaf@larc.nasa.gov
(804) 864-2927

Automatic Differentiation



ADIFOR—Automatic Differentiation for Large-Scale FORTRAN Programs

Objective: Develop an automated, general-purpose method for generating sensitivity derivatives of outputs from various analysis codes with respect to a set of user-selected design variables. These derivatives can be used by a designer or by a nonlinear programming code to achieve some optimal measure of product performance. Apply this new method to a variety of analysis codes used in multidisciplinary design optimization (MDO) activities at NASA. Transfer this new technology to a broad community of users.

Approach: Enhance the Automatic Differentiation of Fortran (ADIFOR) code which was developed by Argonne National Laboratory and Rice University. The initial code only demonstrated the feasibility of this new technology. ADIFOR is a preprocessor tool that augments existing analysis codes with sensitivity derivative calculations for user-specified output quantities with respect to selected input variables as is shown in the figure on the opposite page. Automatic differentiation is based on computing derivatives of all elementary operations contained in the original analysis code and applying the chain rule over and over again to compute exact, desired derivative information. The PARASCOPE compiler, developed at Rice University, is used for code parsing and to extract control flow and dependency information needed in this process. Current development efforts include improving its generality by removing present requirements for some standardization of Fortran analysis codes prior to processing by ADIFOR and to improve the computational efficiency of the resulting sensitivity calculations.

Accomplishments: In FY93 a new approach was developed for applying automatic differentiation to iterative solvers such as those used in advanced CFD codes. In addition, characteristics of certain data structures were exploited to generate a faster derivative code. ADIFOR was used to provide sensitivity derivative calculations for codes that will be used in the design and optimization of a High Speed Civil Transport. These codes include an equivalent-plate structural analysis code and the TLNS3D Navier-Stokes flow code. Sensitivity derivatives were calculated with respect to a variety of parameters, and the results were verified by divided difference

calculations. The first ADIFOR Training Workshop was held to promote the transfer of this technology throughout the government, industry and university communities.

Significance: Automatic differentiation provides a method to generate sensitivity derivatives of large codes accurately and efficiently. This is a significant improvement over competing methods since automatic differentiation does not have the problem of computational error associated with selection of proper step size, as in the divided difference approach, nor is it subject to the human error and lengthy (months) development time of the quasi-analytical approach in which the governing equations are differentiated and new codes must be written to incorporate the resulting expressions for each application. Since ADIFOR is a pre-compiler tool, it can be readily applied to a broad range of applications by both aerospace and non-aerospace industries. Efficient calculation of sensitivity derivatives is a critical technology needed for successful application of MDO methodology.

Status/Plans: Preliminary applications of ADIFOR are underway to calculate derivatives for procedures involving several interacting analysis codes such as grid generation, CFD and structural analysis codes, for solving fluid-structure interaction problems. Also in FY94, methodology will be developed to provide for automatic differentiation of explicitly parallel programs employing message-passing, data parallelism as in High-Performance Fortran and task parallelism as in Fortran-M.

Point of Contact:

Gary L. Giles
NASA Langley Research Center
gary_giles.sdyd_qm@sdmail.larc.nasa.gov
(804) 864-2811

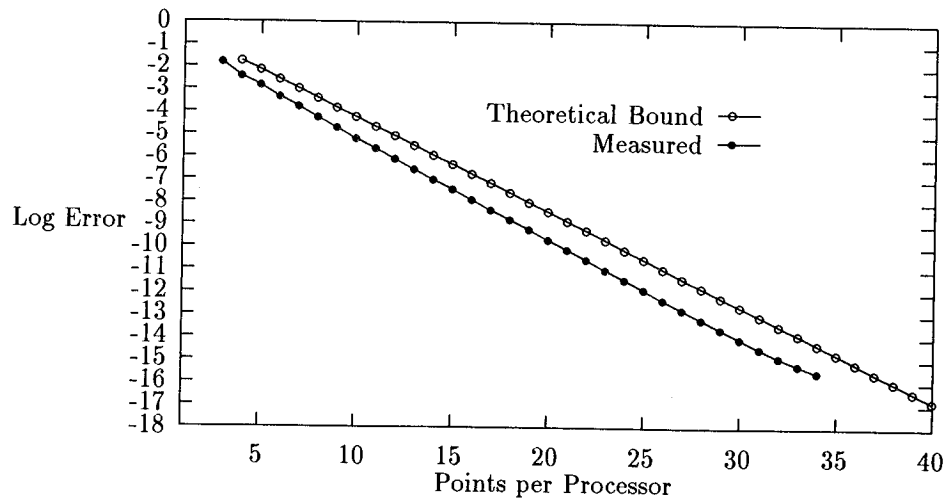


Figure 1: Measured and predicted accuracy of the PDD (Parallel Diagonal Dominant) algorithm. This figure shows the error induced by dropping low-order terms in the PDD algorithm, when applied to a diagonally dominant matrix like that occurring in the compact scheme. As can be seen, the error decreases rapidly as the number of points per processor, n/p , grows, dropping below $1.0E-8$, when one has at least 18 points on each processor. Our rigorous mathematical bound is also plotted here. Similar bounds are available for our other algorithms.

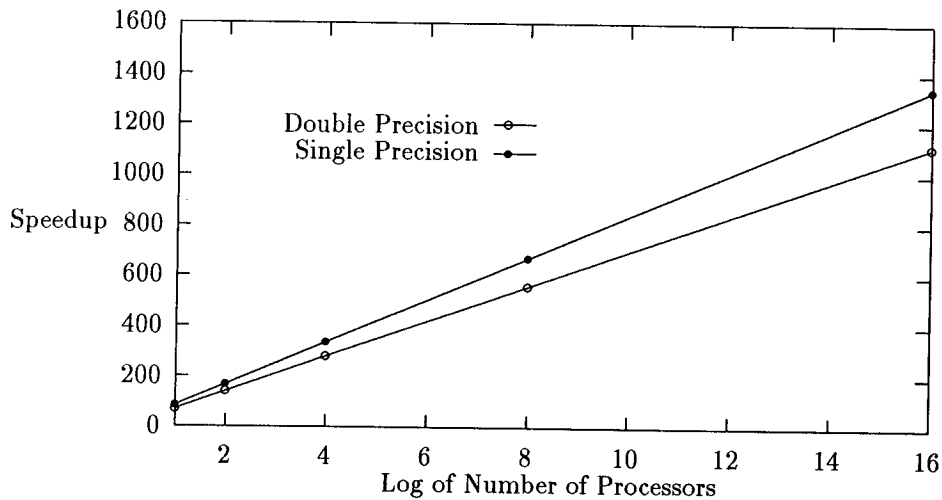


Figure 2: Speedup over the best sequential algorithm. This figure shows the speedup of the SPP (Simple Parallel Prefix) algorithm over the standard sequential Thomas algorithm. The speedup was measured on a MasPar SIMD parallel computer with 16K processing elements. The speedup is linear in the number of processors. Similar results have been obtained for the PDD algorithm.

Parallel Numerical Algorithms for Compact Difference Schemes

Objective: Design efficient parallel algorithms for high-order discretizations on distributed memory architectures, and study complex parallel algorithm tradeoffs more generally.

Approach: A central goal of high performance computing is to achieve high resolution solutions rapidly. One requirement for this is use of discretization methods with a high order of accuracy. An important class of high-order discretization schemes are the compact finite-difference schemes. These schemes offer spectral-like resolution and are among the most efficient schemes for simulation of advection-dominated flows. However, the communication and synchronization costs implicit in these schemes are a barrier to their use on highly parallel architectures.

The crucial issue in parallel implementation of compact schemes is solving the symmetric Toeplitz tridiagonal systems occurring. Rather than simply mapping a standard tridiagonal solver to the parallel architecture, an alternate approach based on approximate solution of the tridiagonal systems is being explored. In essence, it uses the physical properties of the difference schemes, or equivalently the strong diagonal dominance of the tridiagonal systems occurring, to drop low-order terms in the linear system solution.

Accomplishments: Three new algorithms for compact schemes have been designed and implemented. The first algorithm, the PDD algorithm, is optimized for problems of moderate size with arbitrary boundary conditions. The second algorithm, the SPP algorithm, is very efficient for vector and SIMD computing and is unique in its "vertical parallelism." The third algorithm, the convolution algorithm, is designed for the case of large problems with periodic boundary conditions. It is a good candidate for workstation clusters where communication is very costly.

Dropping the low-order terms in the linear system solution dramatically reduces communications and computation costs. For periodic problems, the number of arithmetic operations required by the PDD and by the convolution algorithm is less than for the standard sequential

algorithm for periodic problems. As a result, compact schemes can be made to execute on parallel architectures as efficiently as explicit high-order finite-difference schemes. In recent work, rigorous mathematical bounds have been obtained on the approximation errors induced by dropping low-order terms for all three algorithms.

Significance: The immediate significance of this work is that it makes compact schemes competitive with central-finite difference schemes for parallel and distributed computing. Moreover, since the same symmetric Toeplitz tridiagonal structure appears in many other scientific applications, including the alternating direction implicit method, wavelet collocation methods, spline curve fitting, and so forth, the range of applicability of these tridiagonal algorithms is wider than compact schemes. More broadly, the approach of introducing minor numerical errors in order to obtain greatly improved parallel performance, seems to be widely applicable and should be explored further in a number of contexts.

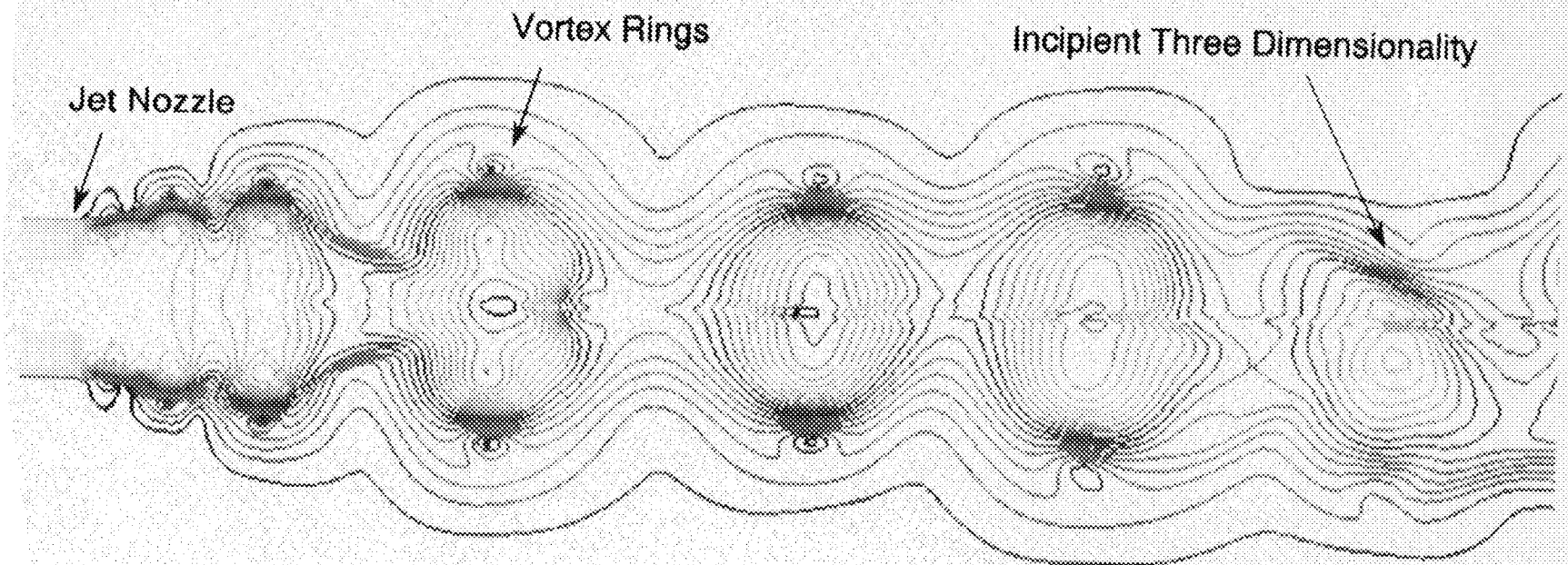
Status/Plans: The mathematical approach described greatly improves the performance of compact schemes on parallel architectures. Plans include continuing the work with the compact scheme and investigating related implicit methods for parabolic problems, where similar approaches may prove effective.

Point of Contact:

Xian-He Sun and
ICASE
NASA Langley Research Center
sun@icase.edu
(804) 864-8018

John Van Rosendale
ICASE
NASA Langley Research Center
jvr@icase.edu
(804) 864-2189

Subsonic Jet - 4 Zone Computation On CM5



- Time Accurate Navier-Stokes Computation Of Jet Flow.
- Unsteady Vortex Shedding Off Lip Of Jet Nozzle.
- Application To Jet Acoustic Source Simulation.
- Large Eddy Simulation Modeling and Assessment.

An Implicit Multizone Navier-Stokes Solver on the Connection Machine

Objective: Develop a time-implicit multi-zone Navier-Stokes code on the Connection Machine CM-2 and CM-5. Study the impact of the interzone communication on the performance of the code. Begin study of a jet exhaust problem with particular application to acoustics.

Point of Contact:

Dennis Jespersen
NASA/Ames Research Center
jespersen@nas.nasa.gov
(415) 604-6742

Approach: A finite difference code for the compressible Navier-Stokes equations with logically rectangular grids and implicit time-stepping was previously developed on the Connection Machine CM-2.

To deal with complex geometries while maintaining logically regular grids, multiple grids or zones are allowed to overlap one another in a general fashion. This overlap implies a transfer of information from one zone to another in an irregular fashion.

The irregular communication pattern potentially could be a bottleneck to code efficiency on a large parallel computer, such as the CM-2 or CM-5.

Accomplishments: Although code development is still underway, application to a subsonic jet problem for investigation of acoustic source simulation has begun. This problem entails a time-accurate computation involving unsteady vortex shedding. A sample result from this computation on the CM-5 using four zones to cover the computational domain is shown in the figure opposite.

Significance: Large regular parallel machines like the CM-5 need not suffer a severe performance penalty when faced with an irregular communication pattern in the multizone approach.

Status/Plans: Development of the code is continuing, including other solution methods for the implicit time-stepping equations. The subsonic jet computations are continuing with application to jet acoustic source simulation.

Distributed Parallel Flow Simulation

* Approach

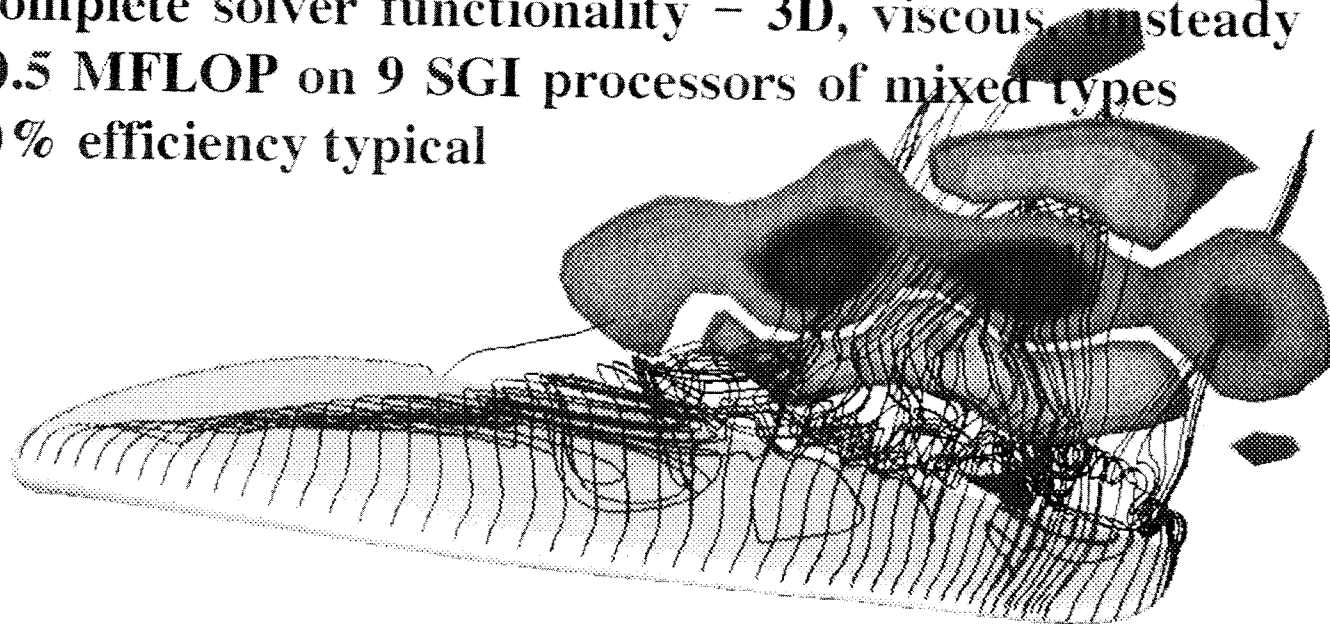
- Use coarse parallelism from multiple grids
- Distribute grids over network-connected workstations

* Implementation

- OVERFLOW-PVM based on OVERFLOW solver
- PVM communications

* Results

- Complete solver functionality - 3D, viscous, unsteady
- 50.5 MFLOP on 9 SGI processors of mixed types
- 80% efficiency typical



AV-8B Harrier Wing. $M_{f.s.}=0.5$, $Re=100,000$, $\alpha=15^\circ$

Distributed Parallel Computing for Flow Simulation

Objective: Develop a distributed parallel flow solver for use on a cluster of UNIX workstations, and demonstrate that distributed parallel computing can provide high performance computing for realistic applications.

Approach: The chimera overset grid method, in which multiple grids are used to simplify the depiction of complex geometries, has a natural parallelism. Each grid is essentially autonomous. The flow field is updated in each grid independently, and then the solutions from boundary surfaces are exchanged between grids which overlap. If each grid can be updated by a separate process, and the processes spread among the workstations of the cluster, an efficient distributed parallel flow solver should result.

The process of developing a distributed parallel flow solver has been separated into four principal tasks:

1. Select an existing flow solver to minimize coding
2. Evaluate communication packages with an emphasis on performance and ease-of-use
3. Integrate message passing with the existing flow solver
4. Test the flow solver on a problem of realistic size and evaluate performance

Accomplishments: The OVERFLOW flow solver was selected for modification due to its overset grid capabilities and wide user base. OVERFLOW has one other feature which has become extremely important for distributed computing, dynamic memory allocation. Grids in an overset grid problem can vary by an order of magnitude in size, which can result in a significant waste of memory. Using dynamic allocation, a single adaptable executable is generated which uses precisely the memory required by each grid.

The Parallel Virtual Machine (PVM) package from Oak Ridge National Laboratory is used for all inter-process communication. PVM consists of user callable libraries in C and FORTRAN, which allow the initiation and termination of processes and the passing of messages.

PVM has been integrated into OVERFLOW in a manager-worker control paradigm. The manager process starts a worker process for each grid in the system, passes out grid and solution data, and synchronizes the workers during solution. A static load-balancing algorithm has been incorporated to improve overall efficiency.

The flow about the wing of the AV-8B Harrier (shown on the opposite page) has been computed on a system of nine grids, with over a half million grid points, using from three to nine workstation processors. A maximum performance of 50.5 megaFLOPS has been achieved on the system of nine Silicon Graphics, Inc. processors at an efficiency of 80.1 percent based on single grid performance, and application performance is increasing linearly with system capability.

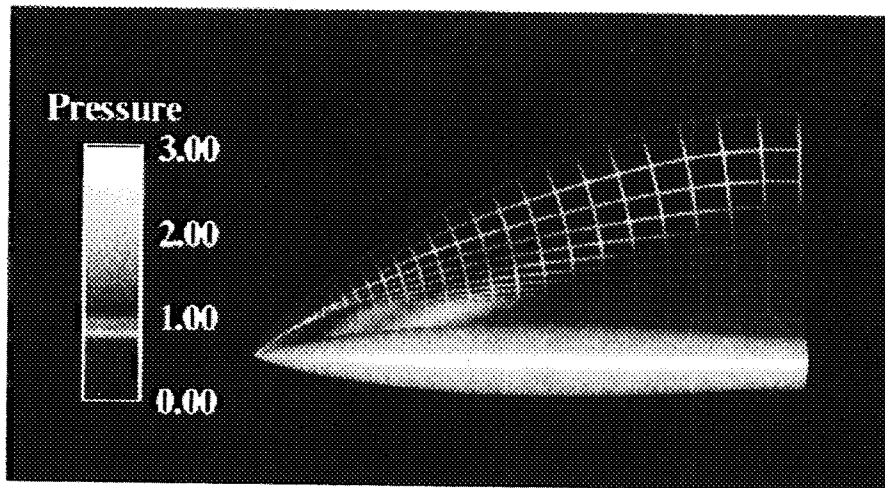
Significance: The performance-to-cost ratio of workstations, which is high, has sparked an increasing interest in the aerospace industry in using workstations for large-scale scientific and engineering computations. Until recently the size of the problems computed had been limited by the speed of a single workstation. By combining several workstations, a system of considerable capabilities can be produced.

Status/Plans: The current solver cannot use more processors than grids in the problem. Consequently, a method of domain decomposition has been identified that may provide much greater parallelism with reasonable efficiency in the distributed environment. A pilot code is being used to evaluate the efficiency of the new method. Incorporation of the domain decomposition technique into the larger distributed flow solver will hinge on the results of this work.

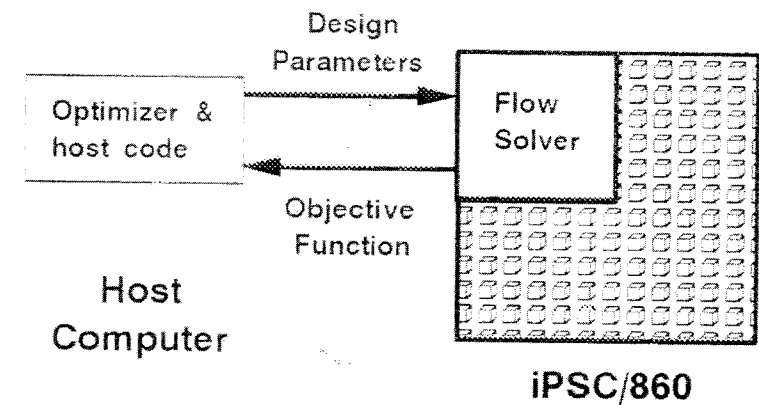
Point of Contact:

Merritt H. Smith
NASA Ames Research Center
(415) 604-4493

Optimization of a Supersonic Body using a Parallel Computer



Haack-Adams Axisymmetric Body

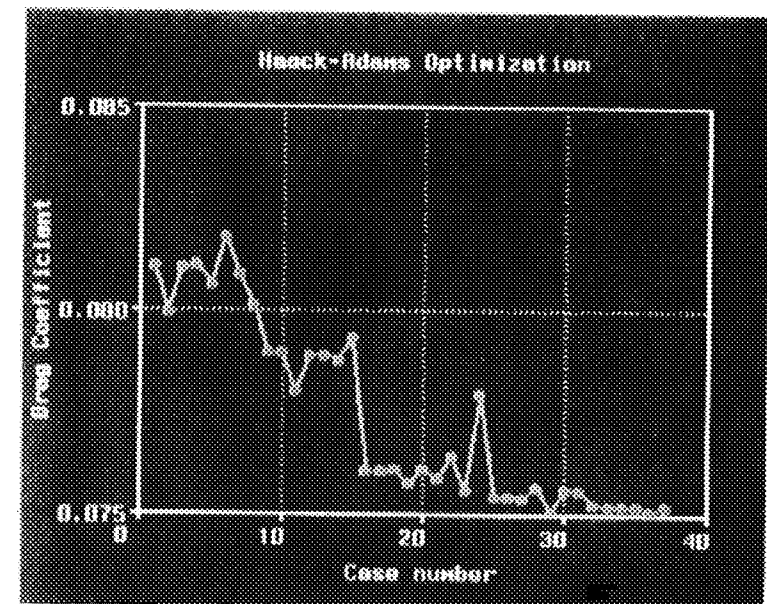


Original body:

- optimal according to supersonic slender body theory

Optimized body:

- defined by 7 terms of a Fourier series
- constrained volume, base area, and maximum thickness point
- optimized for Euler flow at Mach 2.5
- 62 x 21 x 31 grid, 39 μ sec/point/iteration on 16 processors
- about 200 solver iterations per function evaluation, 37 evaluations



Parallel Computer Optimization of a High Speed Civil Transport

Objective. The objective of this research is to perform multidisciplinary optimization of a High Speed Civil Transport (HSCT) on a parallel computer. The optimization will consider aerodynamic efficiency, structural weight, and propulsion system performance. The multidisciplinary analysis will be performed by solving the governing equations for each discipline concurrently on a parallel computer. Developing a scalable algorithm for the solution of this problem will be central to demonstrating the potential for teraFLOPS execution speed on a massively parallel computer.

Approach. The solution algorithms for each discipline will be adapted from existing, serial computer implementations to a scalable parallel computing environment. Parallelism will be pursued on all levels: fine-grained parallelism of the solution algorithm; medium-grained parallelism via domain decomposition; and coarse-grained parallelism of individual disciplines.

The disciplines will be coupled to each other directly through the boundary conditions. For example, the fluid dynamic analysis will communicate aerodynamic loads to a structural analysis. The structural analysis will return surface displacements to the fluid dynamic analysis. An optimization routine will monitor the performance of the multidisciplinary system and search the design space for an optimal configuration.

The discipline coverage and geometrical complexity of the test problem will be expanded as the solution methods mature and execution speed increases. Solving the complete HSCT optimization problem will require execution speeds of many gigaFLOPS and demonstrate scalability to teraFLOPS.

Accomplishments. An optimization capability has been added to the parallel flow solver completed earlier in this project. This required adding force and moment calculations to the parallel code, inclusion of a body shape modification capability, and integration with an optimization code. Shape modification currently supports only rigid motion of lines of grid points, but will be enhanced to include a more general capability for use on complex vehicle designs. The NPSOL optimizer is used, running on the host computer of the iPSC/860,

and calling the flow solver host subroutines for objective function evaluations. The host program then manages repeated use of the parallel processors for successive flow solutions.

Significance. Flowfield solutions and the optimization approach account for a large fraction of the compute time in multidisciplinary HSCT optimization. These capabilities form a cornerstone of the multidisciplinary design optimization goal. The code also will provide a benchmark against which improved optimization methods can be measured.

Status/Plans. The optimization process is effective on the parallel computer, but needs to be faster—and more general. Techniques for further parallelizing the optimization process will be explored. Tools to permit more general grid manipulation on the parallel computer will be developed. Propulsion and structures modules will be coupled as they become available. The work will be ported to the Intel Paragon computer, when it is ready to replace the current iPSC/860 computer.

Point of Contact:

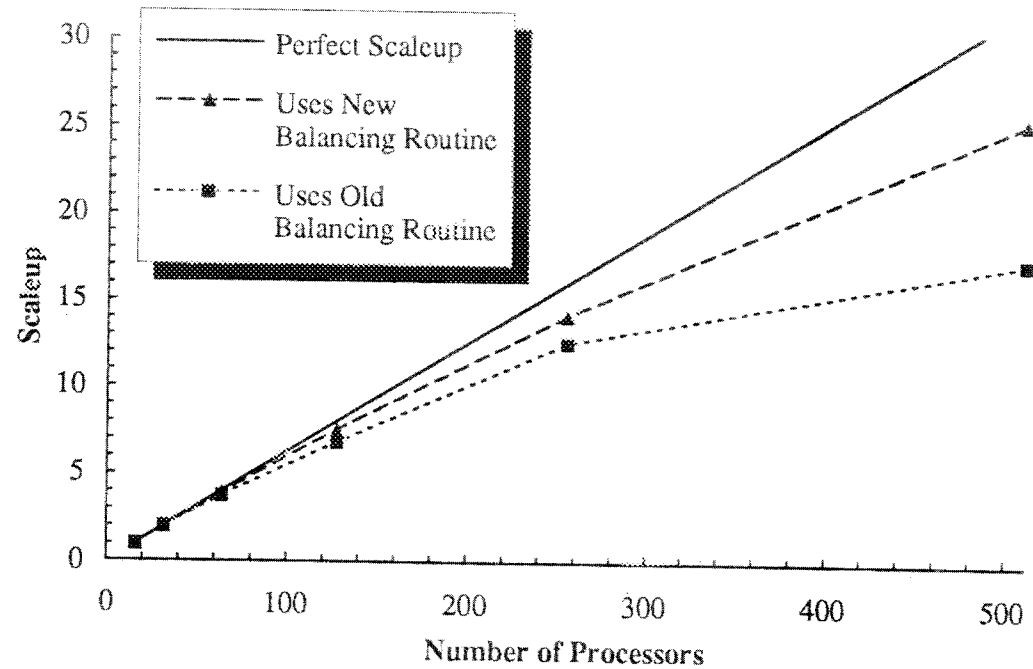
James S. Ryan
MCAT Institute
NASA Ames Research Center
ryan@nas.nasa.gov
(415) 604-4496

Aerobraking Grand Challenge



34

Scaleup Performance: Intel Delta



Sample Problem: Mach 24 flow over a blunted cone,
100 km altitude in 5-species air

Aerobraking Grand Challenge

Objective: Develop and enhance a multiprocessor code for modeling 3-D reactive flows about realistic vehicles during rarefied hypersonic aerobraking with the direct simulation Monte Carlo (DSMC) particle method. Apply this code to realistic flow scenarios to assess code performance, scale-up, and limitations.

Approach: DSMC methods represent a rarefied flow as a large collection of discrete particles which travel through space and interact via collisions. The established DSMC technique is improved by new collision-selection rules and other algorithmic changes that permit efficient implementation on massively parallel processors. Specifically, the flow field is divided into a network of small, uniform, cubic cells. These cells are grouped into indivisible blocks of fixed size. The load-balancing scheme then dynamically assigns various numbers of blocks to each processor as a means of evenly distributing the computational work. Blocks tend to align with the free stream direction to reduce diffusion and interprocessor communication.

Accomplishments: The code was modified for the Intel Gamma machine to run with one processor serving both as a coworker and as the host for the rest of the machine. This permitted ready porting to other machines, such as the Intel Delta. The figure on the opposite page shows flow field temperature profiles plotted from steady 3-D simulations of Mach 24 flow past a blunted cone during aerobraking at 100 km altitude in 5-species reactive air. The figure shows that nearly perfect scale-up was observed with up to 128 processors, although the costs of load balancing operations became excessive as more processors were used on the Intel Delta. This was remedied by restructuring the block processing from sequential to concurrent in those routines, reducing the computer time to load-balance by 93 percent, and improving the scale-up as plotted in the figure. Also, it appears that by restricting the number of cells per block from 512 to 64, calculations for a given block remain within the memory cache of the processor, reducing previous costs that resulted from frequent accessing of main memory. Physical models were developed to directly compute dynamic vehicle surface temperatures during aerobraking rather than requiring crude

isothermal estimates. Enhancements for rotational and vibrational relaxation were included in the code.

Significance: Due to limitations of the Navier-Stokes equations, continuum flow computation methods fail to accurately simulate rarefied flows associated with atmospheric entry of space vehicles or impingement of thruster exhaust upon satellite components. The DSMC methods are particularly well-suited to such scenarios, but require excessive computational resources. This multiprocessor code enables affordable simulation of flows of engineering interest and exceeds the performance of an efficient vectorized DSMC code (for the Cray-YMP) when running 64 processors or more. Good scalability permits implementation on larger machines and simulation of extremely large problems and unsteady flows.

Status/Plans: Thermochemistry models for rotational, vibrational, and chemical relaxation will be enhanced to better simulate real-gas dynamics. Load-balancing algorithms will be improved to account for and reduce interprocessor communication costs. Variable cell sizes within each block will be introduced to improve statistical accuracy when simulating flows with large density variations. Finally, a comprehensive assessment of scale-up and speedup will be completed through simulation of 2-D flows over a cylinder.

Point of Contact:

Brian L. Haas
Eloret Institute
NASA Ames Research Center
haas@corvus.arc.nasa.gov
(415) 604-1145

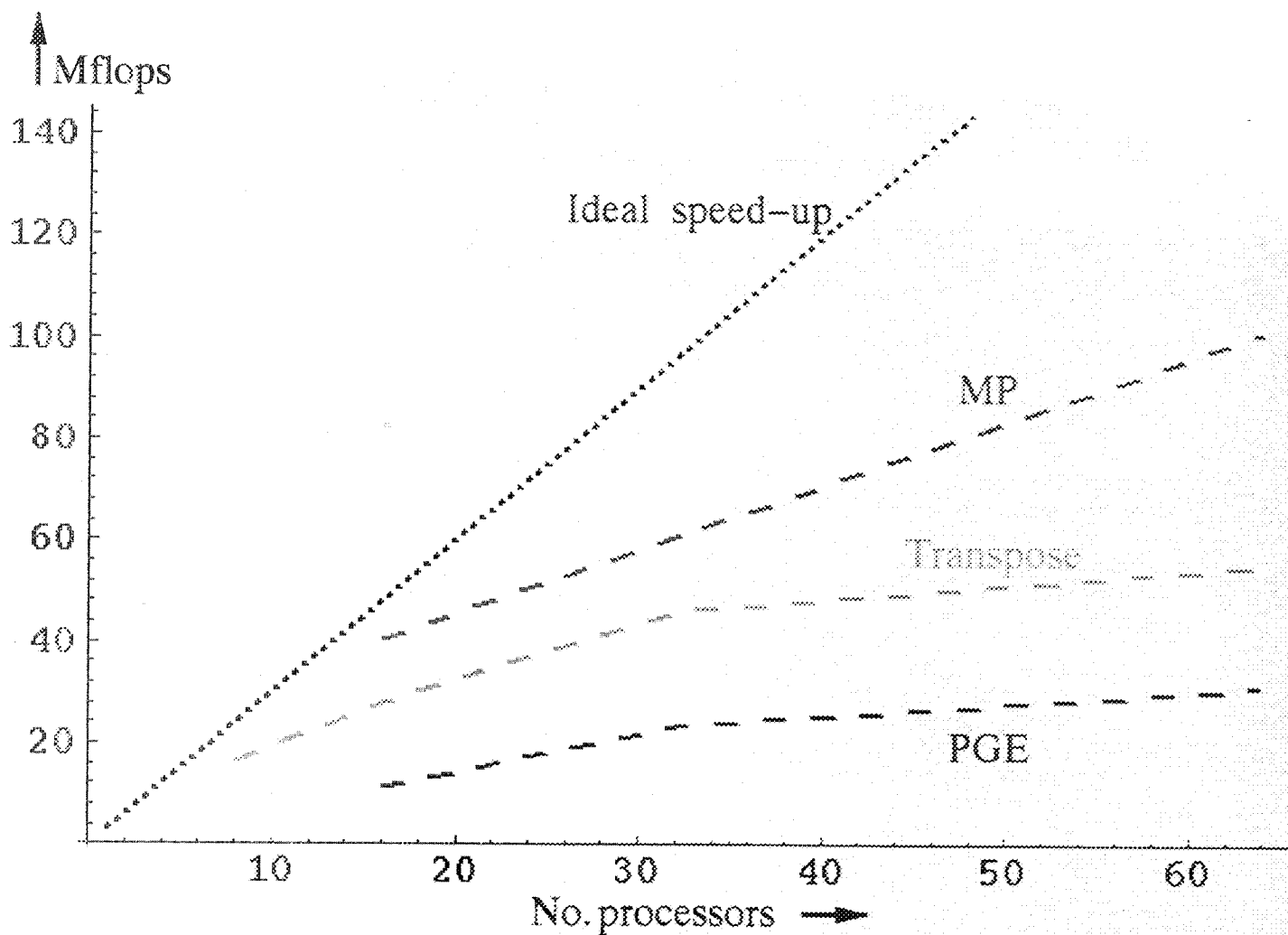


Fig. 1: Parallel performance of heat equation solver on iPSC/860

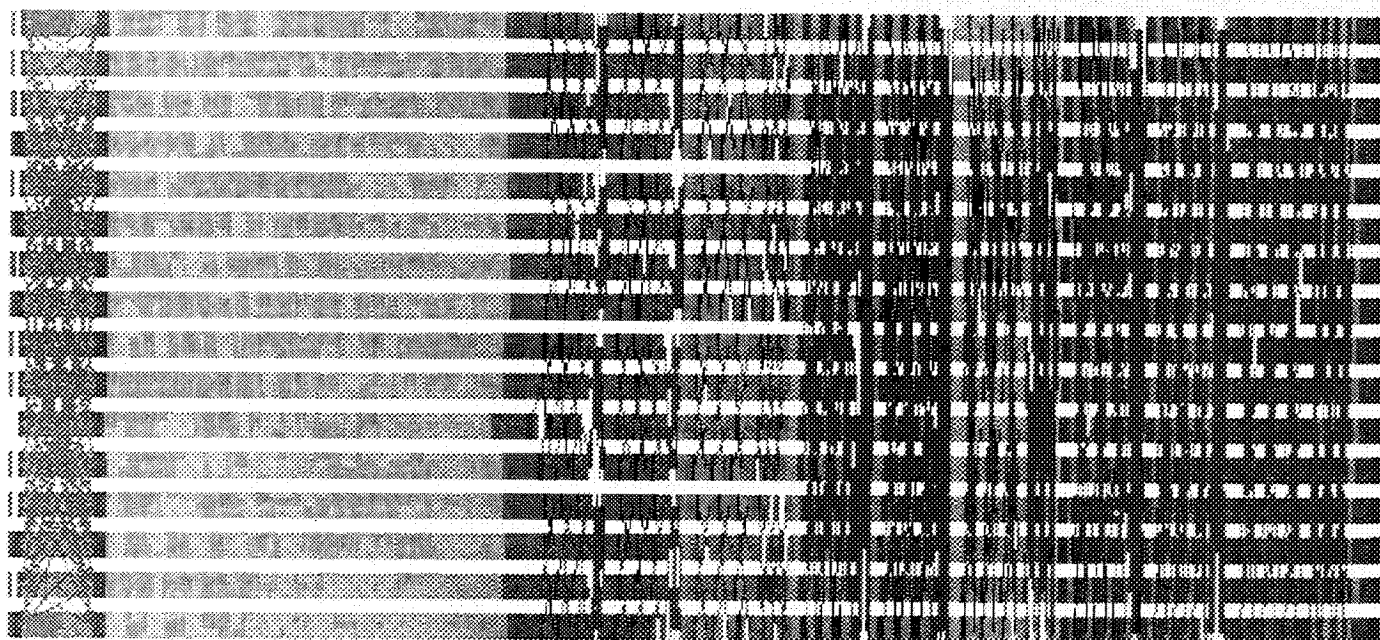


Fig. 2: Communications and idle time for MP SP benchmark problem

Numerical Computation of Coupled Conduction/Turbulent-Convection Heat Transfer Using MPPs

Objective: Develop and implement methods for computing strongly coupled conduction/turbulent-convection heat transfer for complex geometries. Optimize the implementations on massively parallel computers to improve the capability to solve large scale problems and to decrease the turn-around time of medium-sized problems.

Approach: The approach involves:

1. Developing a parallel heat equation solver P THERM3D, based on the previous sequential implementation THERM3D
2. Adapting the existing parallel flow solver POVERFLOW
3. Coupling of flow solver and heat equation solver

These implementations are on a MIMD machine, because MIMD machines are expected to be the most popular with US industries involved in aerospace manufacturing and design. It is also the most flexible parallel architecture proposed so far and is particularly well-suited for multidiscipline simulations.

Accomplishments: In FY93 three major methods (Transpose, Pipe-lined Gaussian Elimination PGE, and Multipartition MP) were studied for implementing P THERM3D, which uses an ADI technique similar to the one employed by POVERFLOW. On the current generation of MIMD computers, a method with little communication between processors fares best. MP offers the advantage of few (and still relatively small) data transfers between processors. Figure 1 on the opposite page shows a performance comparison. MP also was used to implement the NAS SP benchmark problem that mimics a flow solver like POVERFLOW.

Figure 2 on the opposite page indicates communications (dark lines) and idle time (white space) on a 16-processor machine (horizontal bars). It shows that the current implementation has very little processor idle time.

Significance: The need for multidisciplinary analysis and design of aerospace

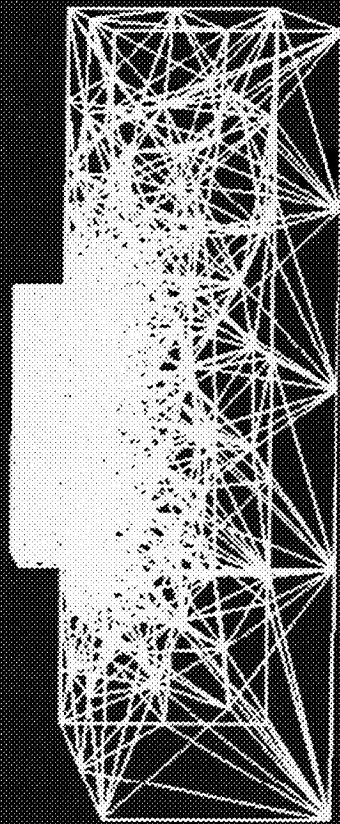
vehicles has been growing with increased system integration of modern flight vehicles and the demand for higher accuracy in unsteady simulations. The requisite computational power is becoming available and cost-effective in the form of massively parallel computers. This power cannot typically be used in a straightforward way; significant code or algorithm changes are often required when porting codes to a parallel machine. Likewise, considerable care is needed when developing new programs for parallel machines to make them efficient *and* portable across a range of computers. This project addresses parallel efficiency and portability of the multidisciplinary simulation code, as well as the quality of the underlying flow solver turbulence model and fluids/thermal interaction model.

Status/Plans: The SP benchmark program is being extended to a new version of POVERFLOW, which will be coupled to P THERM3D. The codes, which now run on the Intel iPSC/860, are being ported to the Intel Paragon. Simultaneously, an MP version of P THERM3D and POVERFLOW is being developed for use on a cluster of workstations.

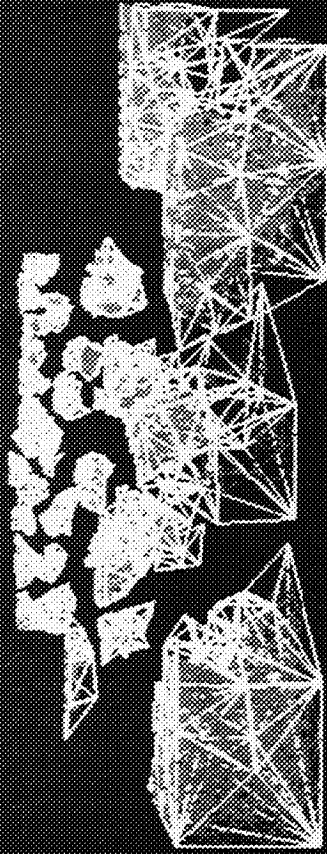
Point of Contact:

Rob F. Van der Wijngaart
MCAT Institute
NASA Ames Research Center
(415) 604-3983

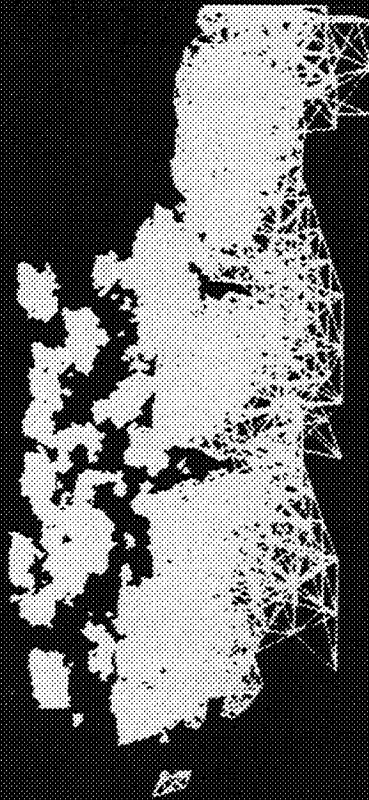
Background Mesh



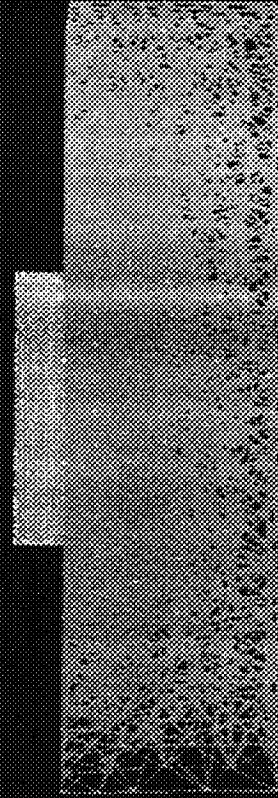
Parallel Subdomain Gridding



Parallel Corner (Interface) Gridding



Final Assembled Mesh



Parallel Grid Generation: Double Missile

NELEM=90,000
NDOFN=40

Parallel Generation of 3-D Unstructured Grids

Objective: Develop a parallel 3-D unstructured grid generator to quickly generate very large grids for CFD runs. Demonstrate scalability of the parallel 3-D unstructured grid generator, and generate grids for complete aircraft configurations, such as the B-747 and HSCT configurations.

Approach: The parallel 3-D unstructured grid generator is based on the advancing front technique that has been used successfully for a number of years on scalar and vector machines. To achieve parallelism, the space to be gridded is subdivided spatially using the background grid. Then, individual domains are gridded in parallel. Also, the regions between the subdomains are gridded in parallel in a subsequent step.

Accomplishments: A first version of the parallel 3-D unstructured grid generator was developed and run on the Intel Delta machine at Caltech. A large portion of the work was devoted to the development of new data structures because in 3-D only a fraction of the total mesh can reside in any one processor at one time. The 2-D parallel grid generator stored the complete mesh in one 'master' processor. This was possible for all but the largest (greater than 500 000 points) 2-D grids. Therefore, new data structures were required that would allow the generation of grids of arbitrary size without storing the whole grid in any one processor. The complexity of data handling was simplified by a series of rules, such as:

1. A point belongs uniquely to one (and only one) subdomain.
2. An element belongs to the lowest domain-number of its nodes.
3. A face may belong to more than one domain.
4. Any element and point is saved once, in the node where it was first generated.

This enabled the creation of grids much larger than could possibly fit into any one processor. The figures opposite show a simplified domain that demonstrates the concept on the domain surrounding two missiles. It is evident that it becomes almost impossible to show the operation of the algorithm on larger grids. The automatic

generation of suitable background grids also was investigated. Because the background grid must be subdivided to achieve parallelism, it is important to have background grids that are fine enough to achieve proper work balance. A prototype code for this purpose was developed and subsequently used.

Significance: This was the first generator for 3-D unstructured grids ported to a MIMD context. Rapid generation of these grids will enable real-time grid optimization, rapid adaptive grid regeneration within an adaptive context, and increase the ability of the user to tailor grids specifically for the application desired.

Status/Plans: Timing runs to prove the scalability of the algorithm are underway, as is the generation of grids for real problems (e.g., B-747). The parallel grid generator spends a larger than expected time gridding interfaces between domains, and this section of the code is being optimized.

Point of Contact:

Alexander Shostko and Rainald Lohner
The George Washington University
Washington, DC 20052
lohner@seas.gwu.edu
(202) 994-5945

(THIS PAGE LEFT BLANK INTENTIONALLY)

Overview of CAS Systems Software Research

Goal and Objectives: The CAS System Software research is targeting key areas of system software that are important to the development of CAS applications and need more attention than can be provided by the computer industry and others. The goal is to develop a system software suite that is efficient with respect to computer time and the applications developer's time.

Strategy and Approach: The approach of the CAS system software research activity is to target key areas more related to the end user, e.g., programming languages and environments, than the details of the hardware, such as device drivers. Areas currently under investigation include programming languages, distributed programming environments, performance analysis and visualization tools, visualization and virtual reality tools, and object oriented environments (for coupling disciplines and aircraft components). When prototype software is developed, it is used to aid in the development and execution of CAS Grand Challenge applications. Also, the software's efficiency is evaluated with respect to its efficient use of the testbed and the application developer.

In addition to prototype development, the research involves extensive evaluation of testbed vendor supplied system software and, in select cases, cooperative development or enhancement of the software. The CAS system software research does not involve developing and supporting commercial grade software, but includes developing system software technology that can become a nonproprietary standard that can be commercialized by the private sector.

Organization: The system software work is done at the NASA research centers (Ames, Langley, and Lewis), ICASE, RIACS, and by grantees. CAS Project resources are shared among ARC, LaRC, LeRC and the Earth and Space Sciences (ESS) Project.

Management Plan: Each center has a system software leader. These leaders coordinate activities within the CAS project and work with the ESS Project to coordinate all of the system software work in the NASA HPC program.

Point of Contact:

Tom Woodrow
Ames Research Center
woodrow@nas.nasa.gov
(415) 604-5949

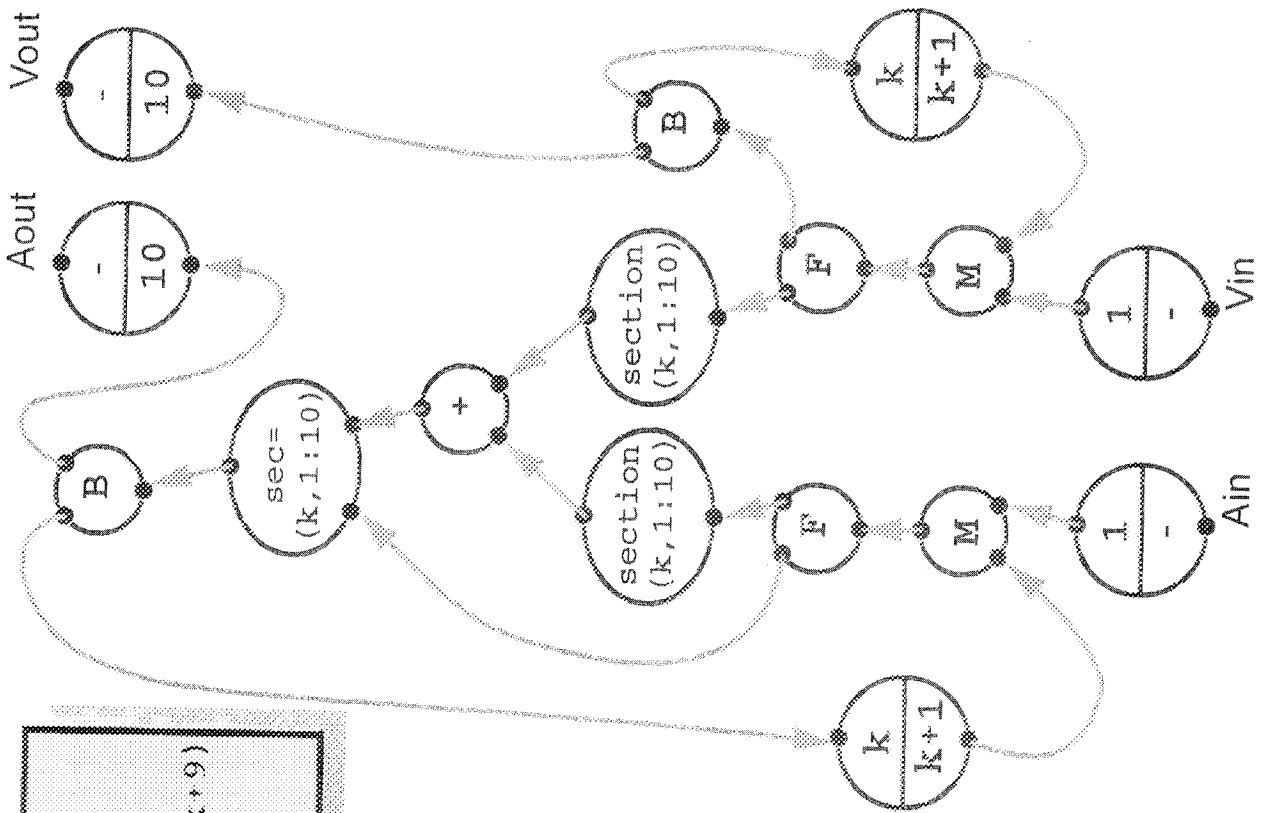
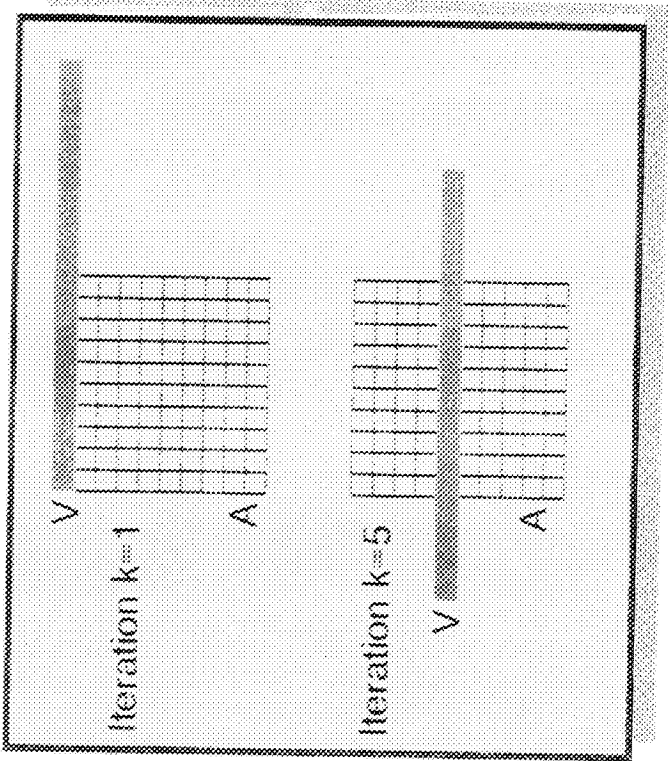
Andrea Overman Salas
Langley Research Center
overman@tab00.larc.nasa.gov
(804) 864-5790

Greg Follen
Lewis Research Center
xxgreg@convex1.lerc.nasa.gov
(216) 433-6193

```

real A(10, 10), V(20)
do k = 1, 10
  A(k, 1:10) = A(k, 1:10) + V(k:k+9)
enddo

```



Optimizing Compiler Technology for Data-Parallel Languages

Objective: Minimize communication due to misaligned operands of operations by developing optimizing compiler technology for automatically determining optimal alignments and distributions of array data in the context of High Performance Fortran (HPF) on distributed memory MIMD and SIMD parallel machines.

Approach: An HPF program is represented internally as an alignment-distribution graph (ADG) that makes explicit the features of the program necessary for determining alignments and distributions of array data. Nodes in the ADG represent program operations. Ports of nodes represent array objects, and edges connect definitions of objects to their uses. The ADG is a static representation of the data and control flow of the program. Alignments are determined for all array-valued objects (named *array variables* as well as *anonymous intermediate results*). The residual communication cost of an ADG can be determined as a function of the alignments of its ports. The alignment problem is to determine a labeling that minimizes this communication cost.

The alignment problem is separated into three subproblems: axis/stride alignment, replication labeling, and mobile offset alignment. Algorithms are developed for each of the subproblems. The axis/stride alignment is solved with a compact, dynamic programming algorithm using the discrete metric as the model of communication. Replication labeling uses network flow to choose objects to replicate. Mobile offset alignment (in which the offset alignment of an object is allowed to be an affine function of the loop induction variables of its surrounding loop nest) is solved using linear programming, with the grid metric as the model of communication.

Accomplishments: A comprehensive theory of whole-program alignment analysis has been developed. The theory handles complete programs, including loops and general control flow. It handles element-wise array operations as well as transformational array operations, such as reductions, spreads, transpositions, and shifts. The algorithms for replication labeling and determining mobile offset alignments account for replication, privatization, and dynamic realignment of arrays. An implementation of the ADG representation of programs and the alignment algorithms has been completed. The

implementation has demonstrated the feasibility of these techniques.

Significance: The current state of compiler technology requires programmers to annotate programs with alignment and distribution directives deemed beneficial by the programmer. Determining an optimal distribution of arrays is complex, often nonintuitive, and sensitive to parameters such as problem size, machine size, and machine topology. These determinations are onerous to the programmer and inhibit portability of codes. This work shifts the task from the programmer to the compiler, provides a rigorous framework for automatically determining optimal alignments, and promotes portability. The ADG representation of programs provides a uniform way of handling control flow and enables the optimization of communication in a program by making it explicit. The new algorithms for determining replication and mobile offset alignments are major advances in solving the alignment problem.

Status/Plans: A prototype compiler has been designed and several modules have been implemented to demonstrate the capabilities of the optimizations. Work continues on the implementation and integration of the remaining modules. The compiler is structured as a directives generator for HPF compilers, such as the one being developed at Syracuse University. Algorithmic research continues in distribution analysis and inter-procedural optimization.

Point of Contact:

Robert Schreiber
RIACS
NASA Ames Research Center
schreibr@riacs.edu
(415) 604-3965

Siddhartha Chatterjee
RIACS
NASA Ames Research Center
sc@riacs.edu
(415) 604-4316

Thomas Sheffler
RIACS
NASA Ames Research Center
sheffler@riacs.edu
(415) 604-4877

(THIS PAGE LEFT BLANK INTENTIONALLY)

Instrumentation, Performance Evaluation, and Visualization Tools

Objectives: The CAS instrumentation, performance evaluation, and visualization activity has four objectives:

1. Demystify the relationship between parallel programs and their performance on parallel architectures by providing tools to help detect performance bottlenecks and suggest ways to eliminate them
2. Develop scalable visual representations of execution traces, focusing primarily on limiting the amount of information gathered and displayed, and on supporting flexible trace-browsing capabilities
3. Port the Automated Instrumentation and Monitoring System (AIMS) to closely-coupled, highly parallel testbeds (e.g., TMC's CM-5)
4. Develop a version of AIMS which supports performance debugging under distributed and heterogeneous (multiprogrammed, multi-user) environments that exploit the portable message-passing interface and multitasking capabilities provided by the Parallel Virtual Machine, version 3.2 (PVM 3.2)

Approach: The approach is to provide a set of tools responsible for automatic instrumentation, monitoring, postmortem visualization, and generation of tables of statistics of program performance. These tools will involve no changes to the source program and will have an interface that is easy to use, requiring minimal user interaction. In addition, a version of AIMS on workstation clusters running PVM will be provided.

One of the most important decisions that a programmer makes in writing an efficient parallel program is about data distribution and alignment. This issue will be addressed in detail with the help of compiler support. Also, the feasibility and usefulness of data-oriented views for CFD applications developed by CAS researchers will be demonstrated.

The approach includes implementing a number of methods for limiting the volume of trace information gathered and displayed by AIMS. The results of this research will be used for developing fast and flexible trace-browsing capabilities for future AIMS releases.

Accomplishments: AIMS was released to COSMIC in August 1993. Copies of the system have been distributed to LaRC/ICASE, JPL, GSFC, ARC/RN and a number of universities for teaching and evaluation. In FY93, enhancements made to AIMS include

1. Adding source code structure information to the trace file that makes traces "self-identifying"
2. An "instrument enabling profile" that selectively turns instrumentation on or off without the need for recompiling the application software
3. An alpha version for the CM-5
4. An intrusion compensation algorithm to compensate for the overhead caused by instrumentation software

Significance: AIMS provides users with detailed information about program execution (with little overhead and a simple user interface) to enable the tuning of their parallel applications on HPCC Testbeds.

Status/Plans: Continue close collaboration with CAS application specialists to identify tool features most useful for their work. Continue development on a version of AIMS that supports intercube communications. This "intercube AIMS" is crucial to help performance evaluation of multidisciplinary and multizonal CFD codes.

Point of Contact:

Jerry C. Yan
NASA Ames Research Center
yan@nas.nasa.gov
(415) 604-4381

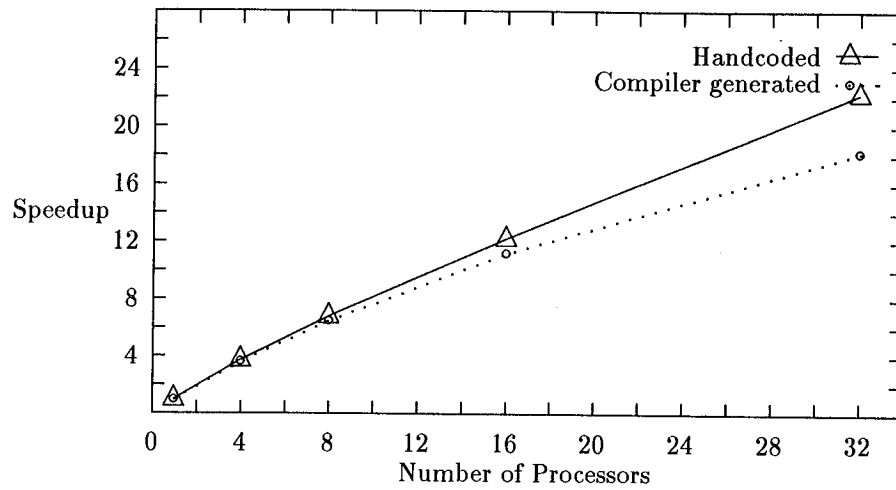


Figure 1: Speedup (relative to execution time of the same code running on one processor) of handcoded vs compiler generated code for Conjugate Gradient iteration on a 64x64x64 grid on the iPSC/860.

HPF-based Programming Environments for Parallel Scientific Computing

Objective: To design parallel programming environments that allow multiprocessor architectures to be programmed almost as easily as sequential machines so that parallelism in scientific applications can be effectively exploited.

Approach: Current parallel programming environments for multiprocessor architectures are primitive, forcing the user to deal with the complex, low-level details of interprocessor communication. Using such environments leads to programs that are so difficult to design and debug that they inhibit experimentation.

ICASE pioneered a different approach with the BLAZE and KALI languages. Using this approach, multiprocessor architectures can be effectively programmed with only minimal changes from familiar sequential programming. The basis of the approach is to add annotations to sequential programs, providing hints to the compiler that controls parallelization. Thus, the user can concentrate on the high-level aspects of the algorithm, leaving the low-level details to the compiler and the runtime system.

Accomplishments: In the last year, a consortium of researchers, including ICASE staff and consultants, designed High Performance FORTRAN (HPF), a new FORTRAN dialect for data-parallel algorithms, based on the approach pioneered at ICASE. Evaluation the effectiveness of HPF for complex NASA codes is underway.

Based on this evaluation, HPF extensions are being designed that should ease the task of expressing the parallelism in these scientific codes.

One area of focus has been block-structured grid codes, for which the fundamental issue is the distribution of the grid blocks over subsets of processors. The present HPF language provides no such capability. Extensions to HPF have been designed to give the user a finer control over the distributions of the blocks in the code. We are studying the performance of these extensions, implemented via the Vienna Compiler System (see figure on opposite page). Preliminary results suggest that these extensions will permit compilers to exploit both the intergrid and the intragrid parallelism in

multiblock codes, allowing one to effectively use architectures with very large number of processors.

Another area of research has been multidisciplinary design optimization codes, which exhibit both functional and data parallelism. To exploit the former, we have designed a tasking layer for HPF, allowing users to express both the coarse grained parallelism across disciplines and the inter-disciplinary sharing of data. At the same time, parallelism within each discipline can still be expressed via HPF, since the tasking layer is well integrated with HPF. Preliminary work on the programming environment using this tasking layer has begun, along with a runtime system which will support this approach in an heterogeneous environment.

Significance: This approach allows users to program complex applications at a high level, focusing on the factors critical to performance, while avoiding most low-level details. With the introduction of these tasking primitives, as well as other HPF extensions, the parallelism of a majority of NASA applications can be easily expressed and exploited.

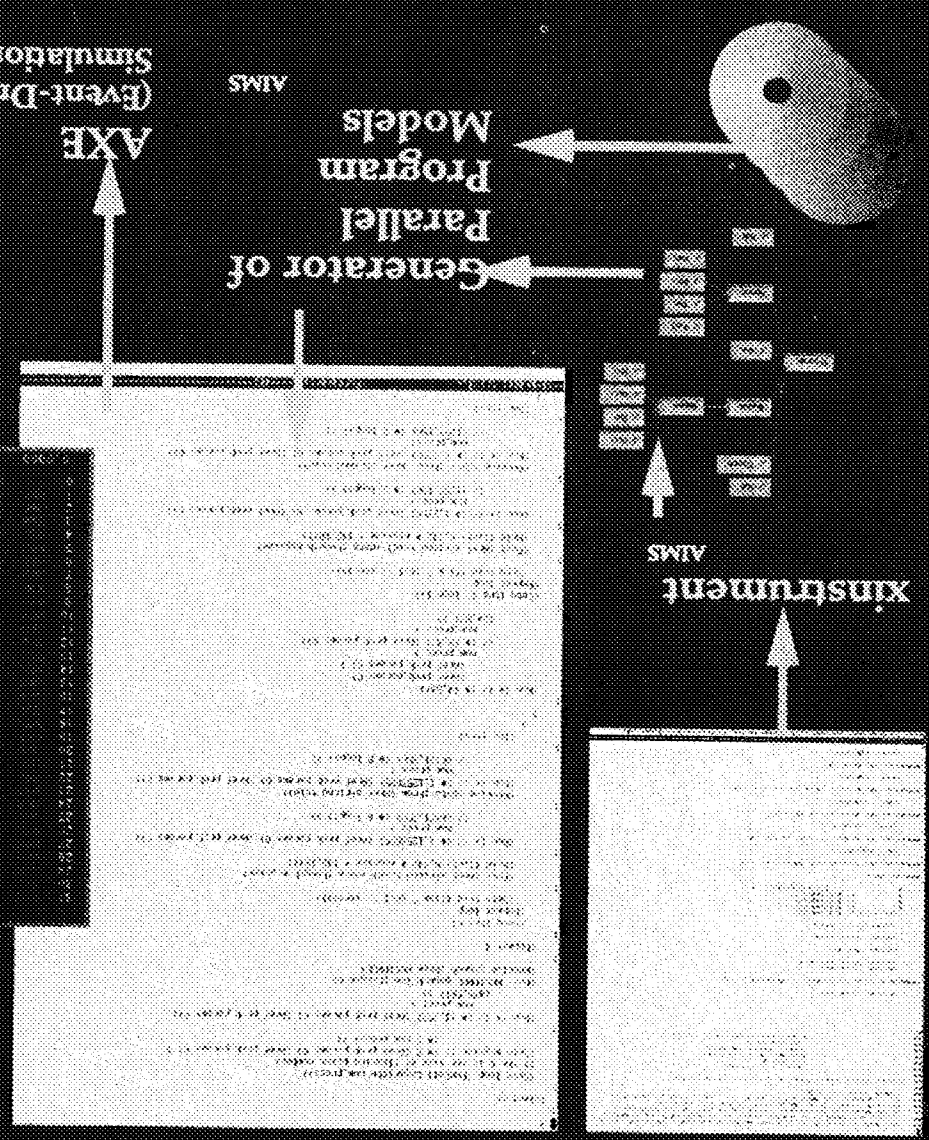
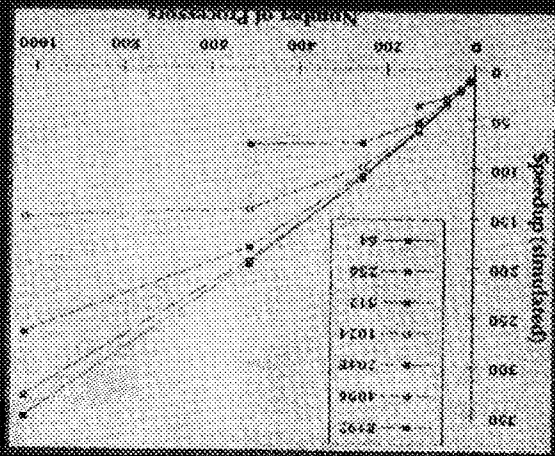
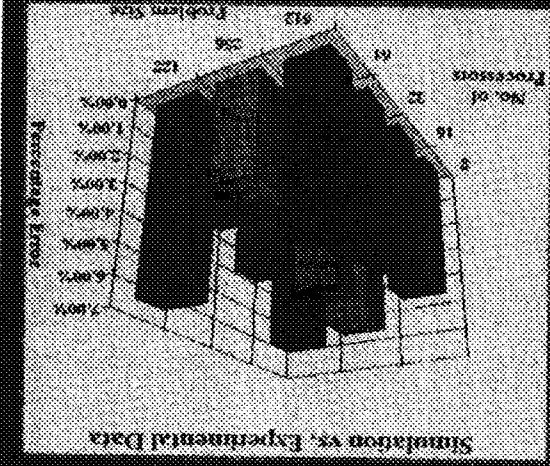
Status/Plans: We will continue evaluating the effectiveness of HPF for NASA codes and, based on our results, continue to design extensions of HPF and compiler technology needed to handle a broader range of applications, including unstructured and adaptive grid computations.

Point of Contact:

Piyush Mehrotra
ICASE
NASA Langley Research Center
pm@icase.edu
(804) 864-2188

John Van Rosendale
ICASE
NASA Langley Research Center
jvr@icase.edu
(804) 864-2189

Performance Prediction based on Simulation & Automatically Generated Program Models



Simulation Studies for Architectural Scalability

Objectives: Build tools to

1. Assist in automated modeling and rapid prototyping of parallel programs for NASA Grand Challenge applications
2. Predict performance of scaled-up applications on scaled-up systems using only the execution traces from scaled-down runs
3. Predict performance under a wide variety of architecture parameters for actual and simulated high performance computing systems

Approach: Applying Amdahl's Law and its extensions to distributed memory multiprocessors requires analysis of computational and communication complexities of applications. Instead of estimating these complexities for entire programs at once, complexity expressions for basic blocks of computation and individual communication/synchronization calls will be developed. These tools, which will combine the power of statistical regression and compiler-assisted performance estimation, will rapidly characterize programs and program segments whose complexity depends not on the content but the amount of data. Modeling and simulation using BDL/AXE will help in further narrowing down of potential scalability problems by allowing the system designer to obtain execution traces with specific problem and system sizes.

A Generator of Parallel-Program Models (GPPM), which is a capability for automating the modeling of message-passing programs, will be built to overcome the tedium and delay associated with manual model-building. By associating complexity expressions with various basic blocks and communication calls, GPPM will help the system designer study application behavior under a variety of architectural parameters, such as processor speed, message latency and message-transmission overhead.

Accomplishments: AXE has been augmented to enable generation of trace files compatible with the visualization tools of AIMS. Detailed comparisons between simulations on AXE and actual execution on the iPSC/860 are now feasible. A model of "xtrid" (a parallel implementation of a tri-diagonal matrix solver) has been "hand built", and its performance has been projected for an iPSC/860 with 1024

nodes. The models were validated against timing results on the iPSC/860 and predicted its execution time to within 7% in most cases. Components of GPPM have been built and tested for xtrid.

Significance: This simulation and modeling capability enables predicting the performance of CAS applications on various architectures with larger numbers of nodes and faster routing systems and evaluating the scalability of both machines and applications

Status/Plans: Currently, the iPSC/860 is simulated using AXE. GPPM generates BDL code that can be compiled to run on AXE. A number of different CM-5 simulators are now available. Since GPPM's internal representation is highly flexible, it also can be used to drive various architecture simulators developed by other groups. An effort will be undertaken to develop a CM-5 back-end for GPPM.

Point of Contact:

Jerry C. Yan
NASA Ames Research Center
yan@nas.nasa.gov
(415) 604-4381

(THIS PAGE LEFT BLANK INTENTIONALLY)

Introduction to the Earth and Space Science (ESS) Applications Project

Project Goal and Objectives: The goal of the Earth and Space Science Project is to accelerate the development and application of high performance computing technologies to meet the Grand Challenge needs of the U.S. Earth and space science community.

Many NASA Grand Challenges require the integration and execution of multiple advanced disciplinary models as single multidisciplinary applications. Examples of these applications include coupled oceanic atmospheric biospheric interactions, 3-D simulations of the chemically perturbed atmosphere, solid earth modeling, solar flare modeling and 3-D compressible magnetohydrodynamics. Other applications involve analyzing and modeling massive data sets collected by orbiting sensors. These Earth and space science problems have significant social and political implications in our society. The science requirements of these applications require computing performance in the teraFLOPS range.

The ESS project has three specific objectives:

1. Develop algorithms and architecture testbeds capable of fully exploiting massively parallel concepts and scalable to sustained teraFLOPS performance
2. Create a generalized software environment for massively parallel computing applications
3. Demonstrate the impact of these technologies on NASA research in Earth and space sciences

Strategy and Approach: The ESS strategy is to invest the first four years of the project (FY92-95) in formulating specifications for complete and balanced teraFLOPS computing systems to support Earth and space science applications. The next two years (FY96-97) will be spent in acquiring and augmenting an on-site (at Goddard Space Flight Center) teraFLOPS system in a stable and operational capability suitable for transition into Code S computing facilities. The ESS approach involves three principal components:

1. Selection of Grand Challenge applications and Principal Investigator Teams that require teraFLOPS computing for NASA science problems. Eight collaborative multidisciplinary Principal Investigator Teams, including physical and computational scientists, software and systems engineers, and algorithm designers were selected using the NASA Research Announcement (NRA) process. In addition, 21 Guest Computational Investigators are developing specific scalable algorithmic techniques. The investigators provide a means to rapidly evaluate and guide the maturation process for scalable massively parallel algorithms and system software, thereby reducing the technical risks assumed by subsequent ESS Grand Challenge researchers when adopting massively parallel computing technologies.
2. Provide investigators with successive generations of scalable computing systems as testbeds for the Grand Challenge applications and (a) connect the investigators with the testbeds through high speed network links (coordinated through the National Research and Education Network), (b) provide the investigators a suitable software development environment and computational techniques support.
3. Collaborate with investigator teams in conducting evaluations of the testbeds across applications and architectures and use these data as input for selecting the next generation scalable teraFLOPS testbed.

Organization: The Goddard Space Flight Center (GSFC) serves as the lead center for the ESS Project and collaborates with the Jet Propulsion Laboratory (JPL). The Office of Aeronautics and Space Technology and the Office of Space Science and Applications jointly selected the ESS Principal Investigators through a peer-reviewed NASA Research Announcement process. The

HPCC/ESS Inter-center Technical Committee, chaired by the ESS Project Manager, coordinates the Goddard/JPL roles. The ESS Applications Steering Group provides oversight and guidance to the project and is composed of representatives from NASA Headquarters (each science discipline office and the High Performance Computing Office in Code R), and representatives from GSFC and JPL. The ESS Science Team, composed of the Principal Investigators, and chaired by the ESS Project Scientist, conducts periodic workshops for the investigator teams and coordinates the computational experiments of the investigations. The ESS Evaluation Coordinator focuses selected activities of the Science Team for developing ESS computational and throughput benchmarks. A staff of in-house computational scientists develops scalable computational techniques which address the computational challenges of the ESS Investigators.

The ESS Project Manager is a member of the NASA-wide High Performance Computing Working Group, and representatives from each Center serve on the NASA-wide Technical Coordinating Committees for Applications, Testbeds, and System Software Research.

Management Plan: The ESS project is managed in accordance with the formally approved ESS Project Plan. The ESS Project Manager at GSFC and the JPL Task Leader together oversee coordinated development of Grand Challenge applications, high performance computing testbeds, and advanced system software for the benefit of the ESS investigators. Monthly, quarterly, and annual reports are provided to the High Performance Computing Office in Code R. ESS and its investigators contribute annual software submissions to the High Performance Computing Software Exchange.

Point of Contact:

Jim Fischer
Goddard Space Flight Center
fischer@nibbles.gsfc.nasa.gov
(301) 286-3465

Robert Ferraro
Jet Propulsion Laboratory
ferraro@zion.jpl.nasa.gov
(818) 354-1340

Overview of ESS Testbeds

Project Goal and Objectives: The goal of the ESS testbeds activity is to ensure that the development of high performance, scalable computer systems evolves towards sustainable teraFLOPS computational capabilities for ESS applications. The objectives are

1. Develop metrics for evaluating and comparing system completeness and balance scalable parallel systems for ESS applications and use the results to help specify systems meeting NASA requirements through competitive procurement
2. Provide ESS Grand Challenge applications performance feedback to system vendors in a form that will help them to improve subsequent generations

Strategy and Approach: Access to a wide variety of scalable high performance testbeds is required for the ESS investigators to develop portable Grand Challenge applications and testcase codes. These applications will be the source of a representative mix of parallel computational techniques and implementations. As these problems are formulated on particular parallel architectures and become useful tools for the investigators, they will be examined by project personnel to identify key computational kernel and data movement components. These key components will be recast to make them portable to other scalable systems, and they will be instrumented to report important values during execution. They will be selected to cover and link the features of the architecture that make a significant contribution to end-to-end speed of execution. In this form, the key components will be run on different scalable systems as a suite of ESS parallel benchmarks to measure the performance envelope of each system. Access to preproduction and early serial number machines enables this activity to perform a pathfinder function.

Organization: Both GSFC and JPL manage and operate ESS-owned testbeds on-site. JPL provides support to ESS investigators on the Intel Delta at Caltech. Also, GSFC has entered into a variety of arrangements with institutions that own large scalable testbeds. Some of these arrangements involve the exchange of NASA funds for machine access and user support. The ESS Evaluation Coordinator has begun to identify early parallel codes as a baseline activity preceding development of the ESS parallel benchmarks.

Management Plan: At GSFC a Deputy Project Manager for Testbeds directs the in-house testbed activities and coordinates arrangements with other institutions for testbed access. At JPL a Deputy Task Leader directs the in-house testbed activity and access to the Intel Delta. The Evaluation Coordinator reports to the ESS Project Manager.

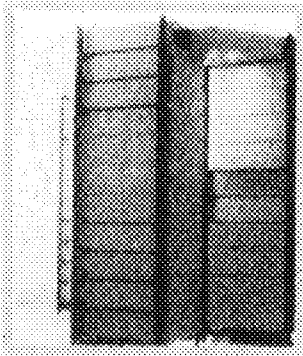
Point of Contact:

Lisa Hamet
Goddard Space Flight Center/Code 934
hamet@nibbles.gsfc.nasa.gov
(301) 286-9417

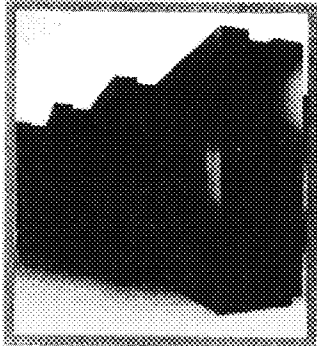
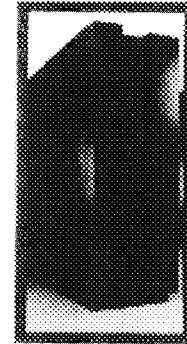
Juliana Murphy
Jet Propulsion Laboratory
julie@olympic.jpl.nasa.gov
(818) 354-7311

HPCC/ESS Testbeds

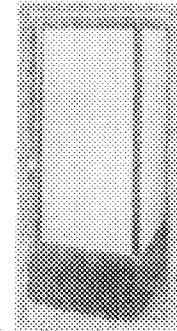
UW
KSR-1



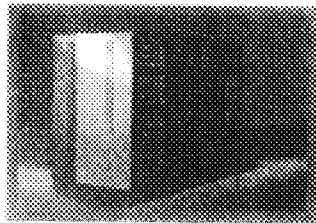
U MD
TMC CM-5



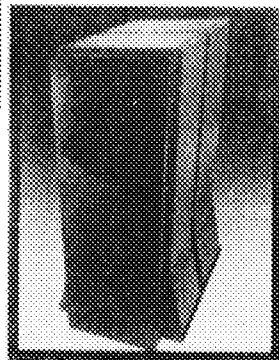
NASA/ARC
TMC CM-5



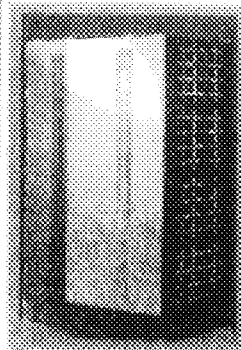
NASA/GSFC
MasPar MP-1



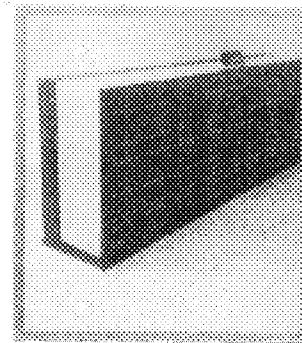
NASA/ARC
Intel Paragon



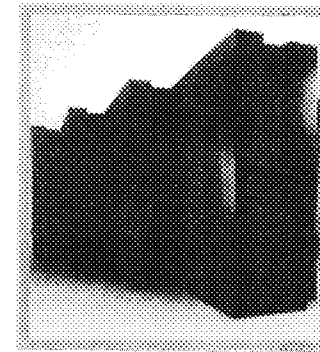
NASA/JPL
Cray T3D



NASA/JPL
Intel Paragon

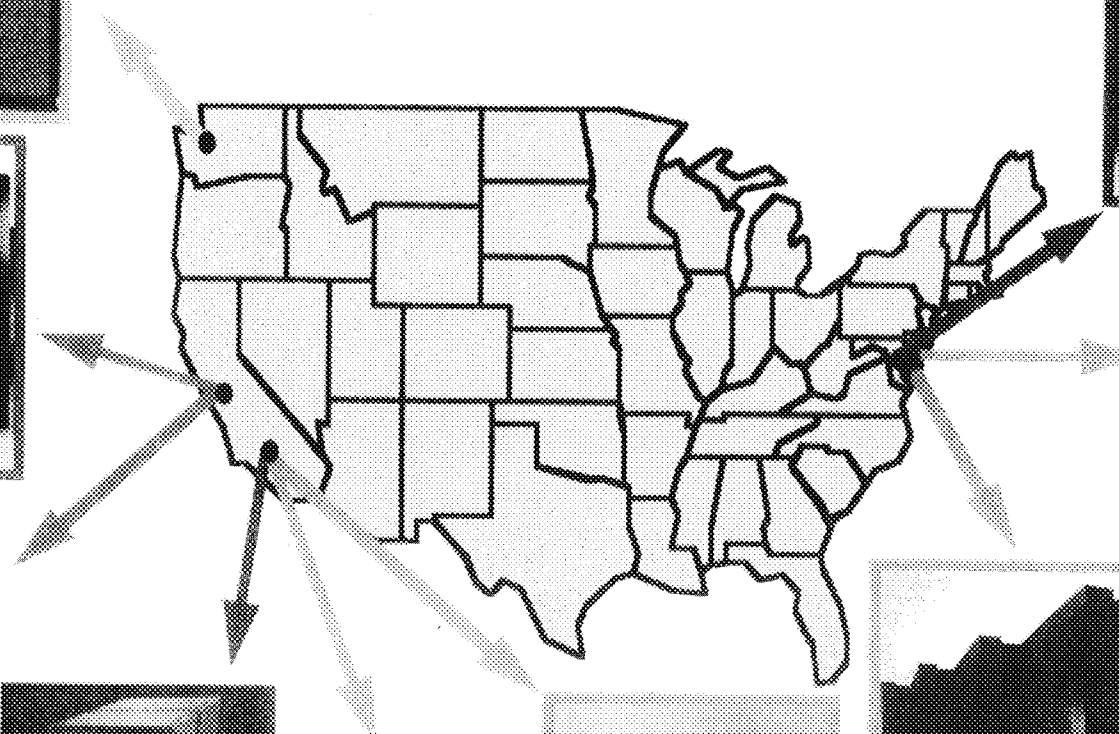


CCSF
Intel Delta



NRL
TMC CM-5

Legend	
	HPCC/ESS Owned
	HPCC/CAS Owned
	Purchased Time
	Future Access
	Investigator Owned



Access to High Performance Scalable Testbed Systems

Objective: Obtain access for ESS investigators to high performance scalable testbed systems that have the potential to scale to tera-FLOPS performance.

Approach: The ESS project provides access to GSFC and JPL scalable parallel systems and also establishes agreements to acquire additional machine time from NASA Computational Aerosciences (CAS) centers and non-NASA research labs that own large scalable systems.

This approach leverages the substantial capital investments made by other organizations and also provides investigators access to a broader variety and larger machines than NASA can afford to purchase.

As computer cycles on remote systems are obtained, ESS testbed managers work with the remote systems' administrators to establish working arrangements and to facilitate system access for ESS investigators.

Accomplishments:

- o ESS used 49,643.59 node-hours of NASA's allotted time on the Intel Touchstone Delta at Caltech.
- o At Ames Research Center ESS used 225.5 node-hours on the CAS-owned Intel Touchstone Gamma and 490 node-hours on Thinking Machines Corporation's (TMC) CM-5. Investigator teams (composed entirely of U.S. citizens) used the CM-5 allotment and are awaiting general availability of the Intel Paragon (the Gamma replacement).
- o ESS installed the Cray T3D Emulator software on the Y-MP EL at GSFC, allowing users to begin converting codes in preparation for the availability of the JPL Cray T3D.
- o ESS investigators are using the TMC CM-5 at the Naval Research Laboratory. A NASA Defense Purchase Request (NDPR) will pay for the usage.
- o Several ESS investigators are using the KSR-1 at the University of Washington that

was offered on a limited basis by Principal Investigator G. Lake.

- o GSFC is completing a study that will produce a recommendation about which machines to consider in FY94 as on-site GSFC technology refreshment testbed(s). The final report has been submitted for review.

Significance: Exposing the investigations to a wide variety of scalable systems helps to objectively rule out weak contenders with respect to ESS requirements. It also enhances the investigators' chances of success in addressing their Grand Challenge problems by increasing the likelihood of working with architectures well-matched to the problems.

The larger sizes of these shared machines allow larger problems to be run. This aids the system vendors by allowing the investigations to test closer to the maximum potential of the machines to disclose strengths and uncover weaknesses. The results of these investigations should accelerate further development of the largest systems and hasten the eventual construction of a teraFLOPS system.

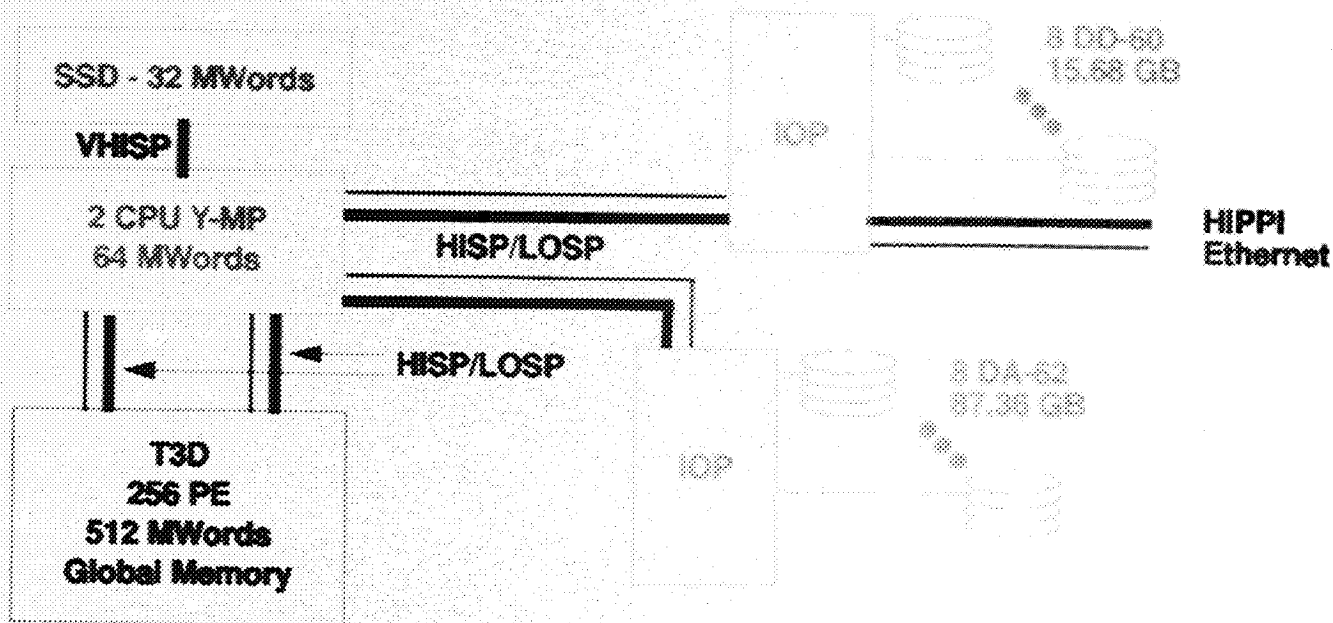
Status/Plans: In FY94, the ESS project will provide a member of the Evaluation Team for the Ames Research Center Cooperative Research Announcement (CRA) and will use the resulting awards to acquire testbeds in FY94. The ESS report will guide the selections. ESS will also complete the NDPR to NRL while working to buy development time on the University of Maryland 32-node CM-5.

Point of Contact:

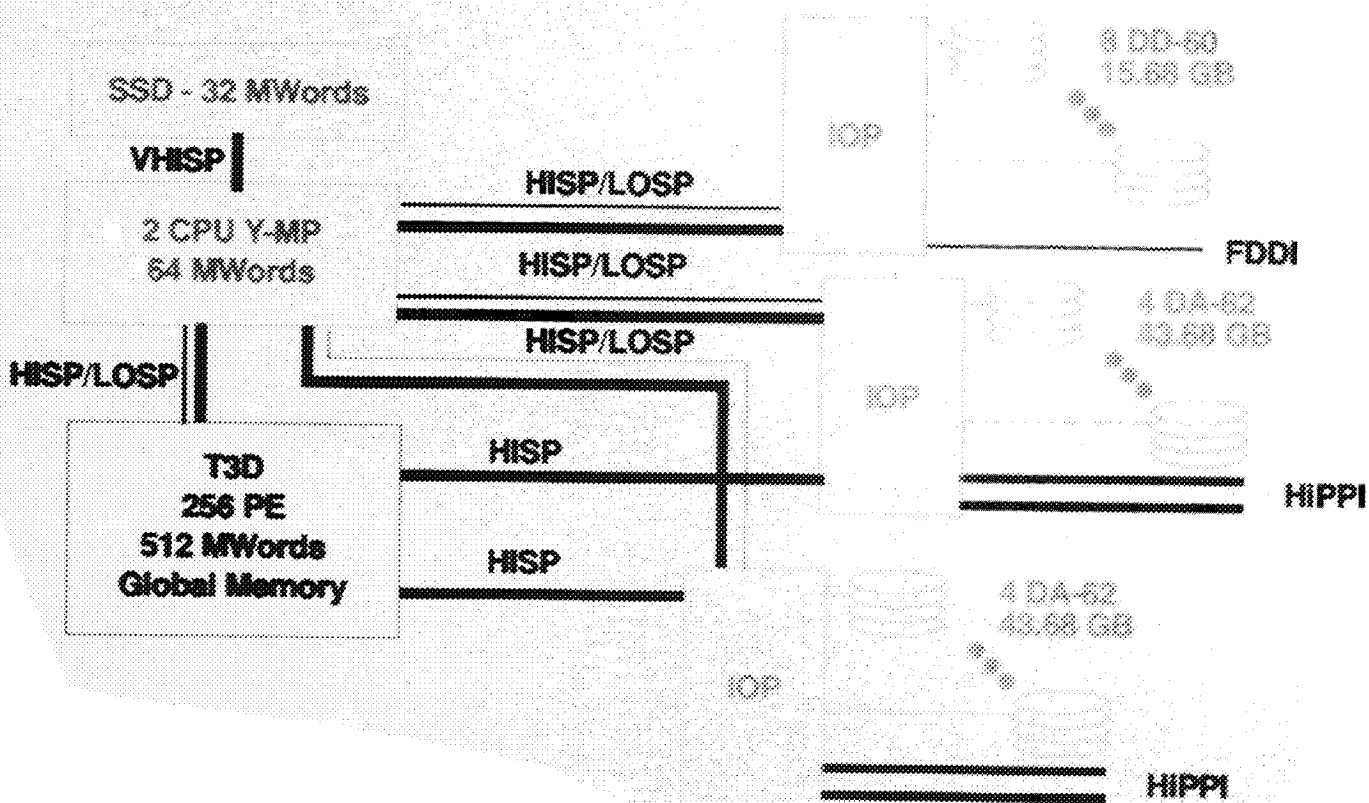
Lisa Hamet
Goddard Space Flight Center
hamet@nibbles.gsfc.nasa.gov
(301) 286-9417

JPL/Caltech Cray T3D

Phase 1 I/O Configuration



Phase 2 I/O Configuration



Prepared for Delivery of the Cray T3D Testbed

Objectives: Provide ESS investigators with access to the latest generation MIMD architectures for developing Grand Challenge applications, algorithms, and software support tools.

Approach: Leverage funds from the NASA ESS project and NASA Code Y Supercomputing project to obtain early delivery of a large configuration of a new parallel architecture. Sign a cooperative agreement with the vendor to collaborate on applications development and test early version of system software and tools.

Accomplishments: Caltech, JPL and Cray Research, Inc. signed a cooperative agreement to obtain an early Cray T3D Massively Parallel Processor. The T3D will be hosted by a 2-processor Y-MP, have 256 processing elements (PEs) connected in a 3D toroidal communications network, 2 MWords (16 MBytes) of memory per PE and 103 GB of available disk storage. In the initial configuration (i.e., Phase 1 I/O) all I/O services will be provided by the Y-MP front end. Two High Speed/Low Speed I/O channel pairs will connect the T3D to I/O Controllers in the Y-MP. Two additional pairs from the Y-MP will connect to the IOPs with the attached disks. A Phase 2 I/O configuration that provides two high speed channels directly from the T3D to the IOPs with attached disks will be implemented. HiPPI. Operating System I/O services will still be controlled by the Y-MP in this configuration. PVM will be supported with the initial hardware delivery. The Cray MPP Fortran Programming Model is expected to incrementally become available beginning in the Spring of 1994.

A set of collaborative projects has been identified as the Parallel Applications Technology Program at JPL/Caltech. These projects, which will exercise the T3D hardware and software, were chosen as possible key MPP applications of interest to Cray Research, Inc. The project set includes planetary data visualization; image analysis; image rendering; electromagnetic simulation; and several dynamics modeling applications in weather, chemistry, microbiology, and plasma physics. The project set also includes tools for program development, benchmarking, and performance analysis.

Significance: The ESS Grand Challenge investigators will have access to a variety of

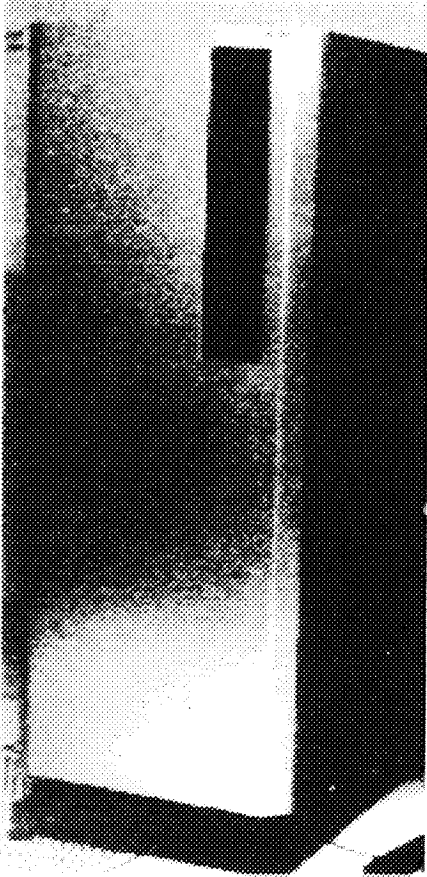
parallel architectures, including the Intel Paragons at JPL and Caltech, and the CM5 at NRL. The availability of multiple architectures is important to assure that applications development work is portable among different platforms and allow the investigators to assess the strengths and weaknesses of various parallel hardware designs.

Status/Plans: The T3D should arrive in December of 1993. ESS applications and software tools work will begin when the machine is available. Phase 2 I/O implementation will be in April 1994.

Point of Contact

Robert Ferraro
Jet Propulsion Laboratory
ferraro@zion.jpl.nasa.gov
(818)354-1340

JPL ESS MIMD Testbed

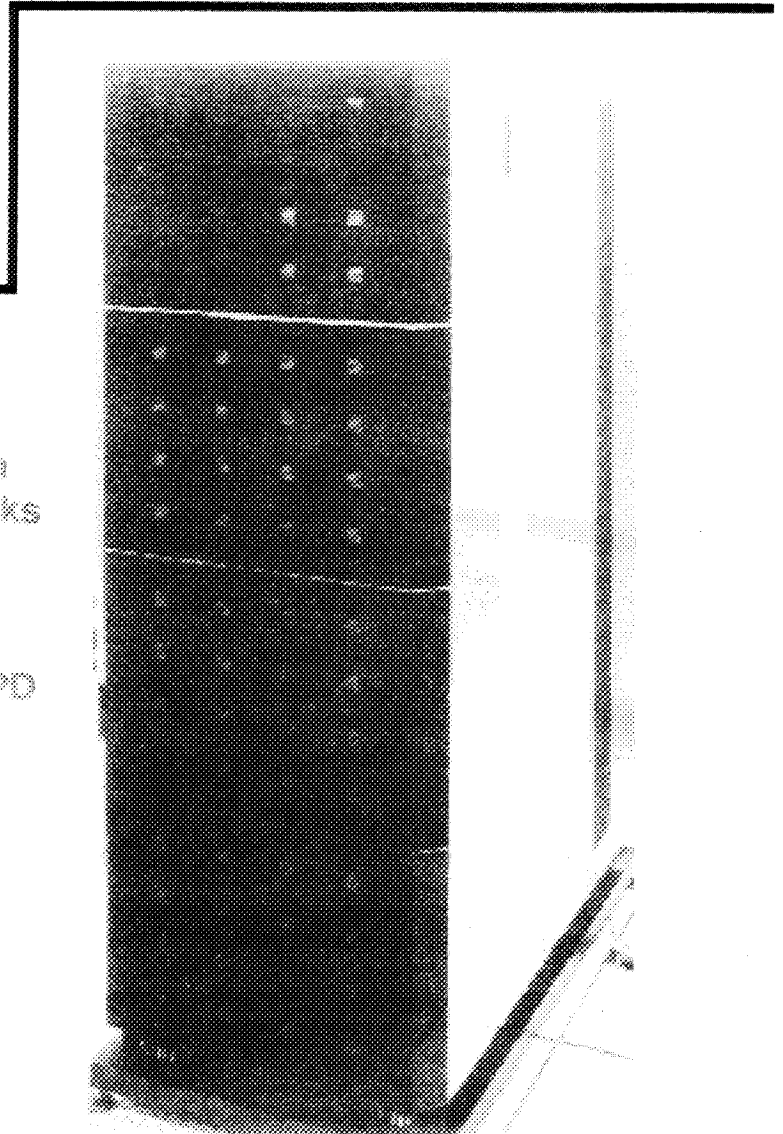


Intel iPSC/860

8 Compute Nodes with 16 MB each
1 CIO Node with 1.2 GB total disk
Hypercube communication architecture
Fortran, C, C++, BLAS, DGL, IPD
Intel NX message passing

128 MB Total Memory

320 MFlops Peak Speed



Intel Paragon XP/S Model A4

Installed in FY93

58 Compute Nodes with 32 MB each
3 RAID Controllers with 14.4 GB disks
1 User Service Node with 32 MB
1 Ethernet connection
1 4mm DAT
Fortran, C, GNU C++, BLAS, DGL, IPD
Intel NX message passing
Mesh communication architecture

To be installed in FY94

1 RAID Controller with 4.8 GB disks
1 HIPPI/FDDI connection

Final Configuration

1.9 GB total memory
4.2 GFlops Peak Speed
19.2 GB disks

Intel Paragon Testbed

Objectives: Provide ESS Investigators with access to the latest generation MIMD architectures for developing Grand Challenge applications, algorithms, and software support tools. Operate a testbed configuration that allows the development of scalable applications, access to high performance networks, and large data set configurations.

Approach: Grand Challenge applications development requires early access to hardware of sufficient size and computing power to test algorithms for performance and scalability. Software tools development also requires such access. A continuing infusion of new technology is needed to assure that software being developed is scalable to teraFLOPS performance and is relevant to the latest generation of parallel architectures. JPL operates the ESS MIMD testbed to provide this access, and upgrades the technology and configuration as warranted. The testbed also serves as a development machine for applications to be run on the CSCC Delta.

Accomplishments:

Upgraded the JPL testbed with an Intel Paragon XP/S Model A4. The Paragon is source code compatible with the iPSC/860, but uses a faster processor with a larger cache and has a 2-D mesh communication architecture and an order of magnitude increase in communications speed. The Paragon OS is based on the Mach Kernel, and supports standard UNIX services for applications. Paragon NQS is available for controlling batch jobs. The hardware configuration of the JPL/ESS Paragon is as follows:

- 56 Compute Nodes—32 MB/node
- 1 Service Node—2 MB
- 3 RAID Controllers—14.4 GB disk space
- 1 Ethernet connection
- 1 4mm DAT
- 1 HiPPI/FDDI connection (1st Qtr FY94)
- C and Fortran Compilers, GNU C++
- Software Development Environment

- Software Math Libraries
- DGL

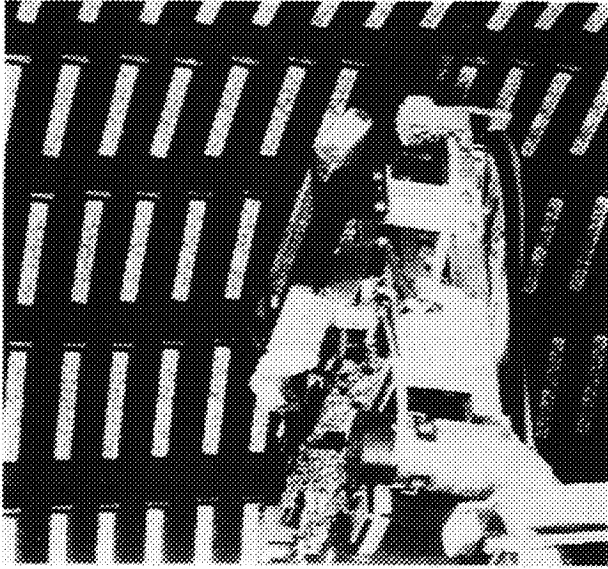
Significance: The testbed now provides users with a larger platform on which to develop and test their applications.

Status/Plans: JPL/ESS testbed users are moving onto the Paragon from the iPSC/860 by simply recompiling their applications. The iPSC/860 will be retained in an 8-node configuration while users transition to the new hardware. An additional RAID Controller with 4.8 GB disk space has been ordered, and will be installed in early FY94.

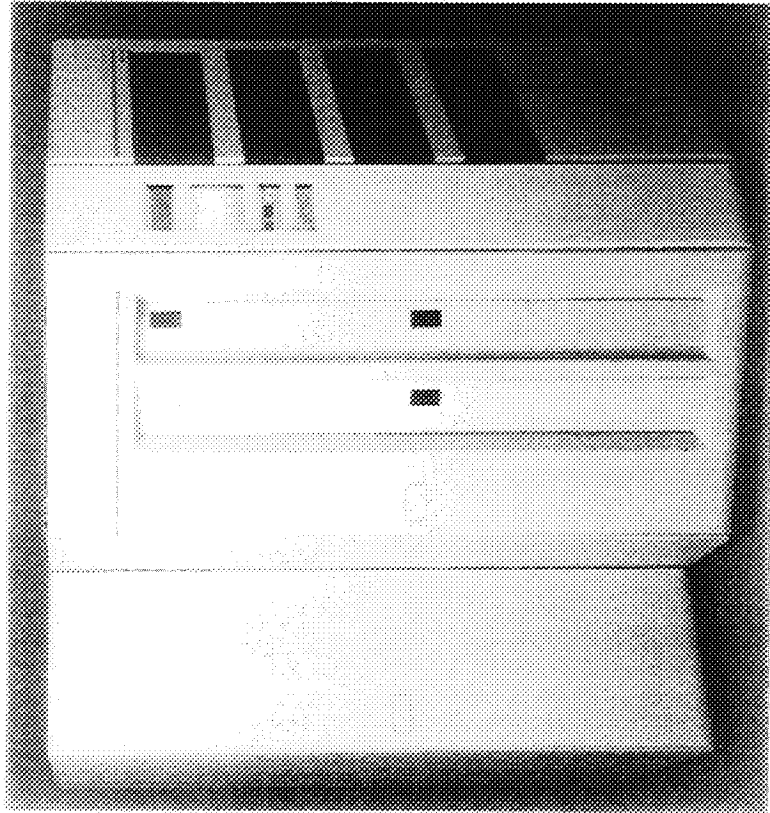
Point of Contact:

Robert Ferraro
Jet Propulsion Laboratory
ferraro@zion.jpl.nasa.gov
(818) 354-1340

WolfCreek™ Tape Silo



- 1,000 cartridges (200 GB)
- 350 cartridge exchanges per hour
- Maximum of eight transports
- Can be shipped pre-assembled for immediate installation
- Pass-thru operations to other current and future STK Libraries
- Cartridge Access Port (CAP) that houses 20-50 cartridges



Evaluation of High Performance Mass Storage Systems

Objective: Evaluate and contribute to the evolution of scalable high performance mass data storage capabilities that enable ESS Science Team investigations, such as high capacity, low latency, sustainable data rates, high reliability, and commercial availability.

Approach: Identify lessons learned and technology options at other research sites; field test and evaluate cost effective mass storage systems and upgrades; put test systems into realistic high volume use by ESS science teams; identify and evaluate limiting factors, and determine requirement shortfalls.

Accomplishments:

- ❑ Completed I/O throughput and response time performance measurements of a Storage Tek 4 400 silo; results to be presented at the NASA/GSFC Conference on Mass Storage Systems and Technologies between 19 and 21 October, 1993.
- ❑ Completed a beta test evaluation of two Storage Tek Wolfcreek™ silos, resulting in the procurement of one of those silos (see figure on the opposite page). The image at the upper right shows the internal operating mechanism of the silo and the image at the left shows the external configuration. Results of the beta test were presented to the EOS Data and Information System (EOSDIS) project government staff and the EOSDIS Hughes development staff.

Significance: Monitoring and evaluation of mass storage performance in the context of actual applications will provide feedback of correctness of performance assumptions based on field test results and will provide feedback to develop requirement shortfalls. Additional field tests of both hardware and software systems will provide alternative procurement options available for satisfying requirement shortfalls. Exploration of scalable technologies being developed will identify mass storage systems possible for future field tests.

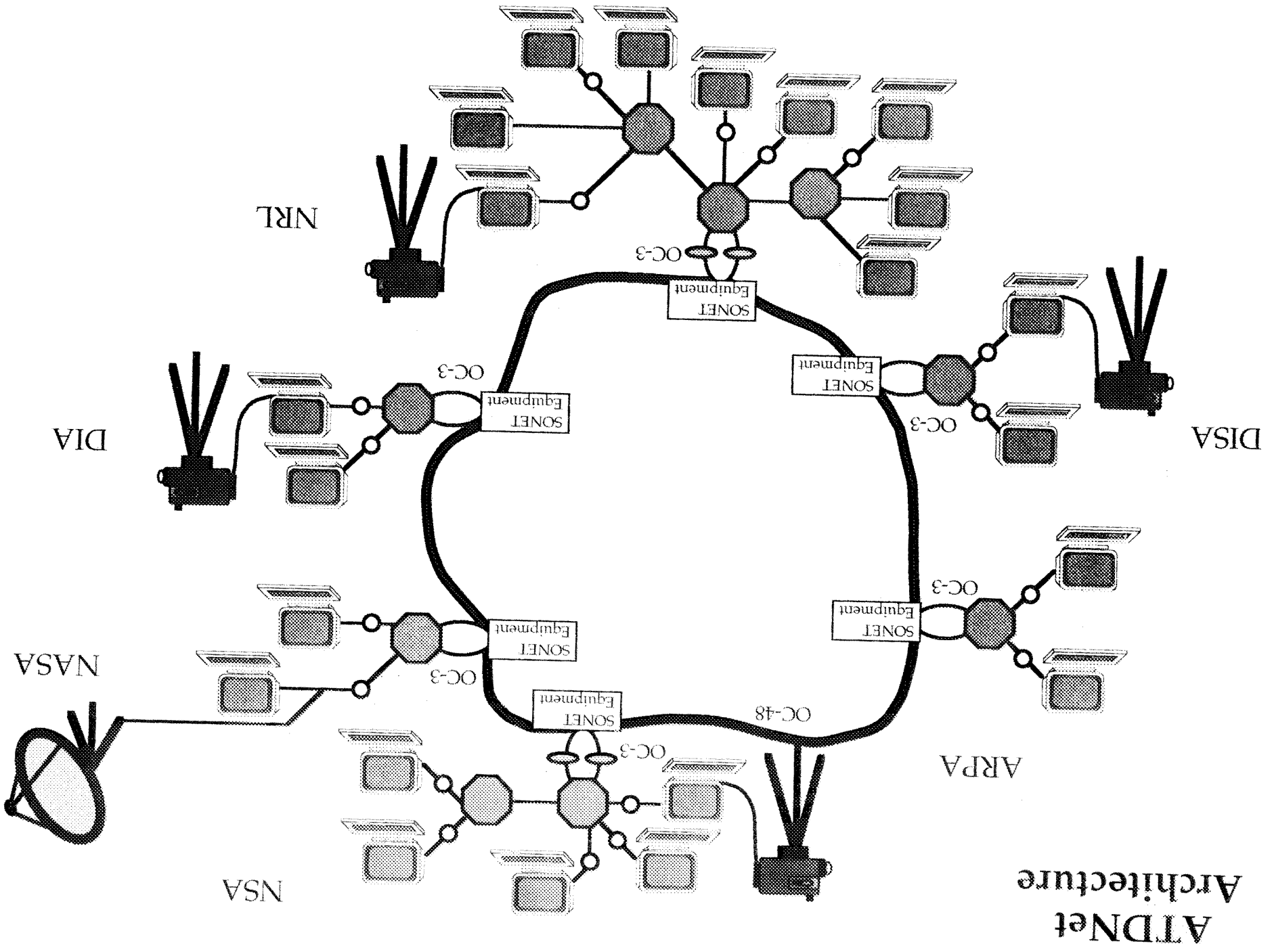
Status/Plans:

- ❑ Monitor performance of the Storage Tek Wolfcreek™ silo in the context of actual ESS Grand Challenge applications.

- ❑ Perform field tests and evaluations of additional mass storage hardware system such as the ASACA high speed optical disk drive, and expand the evaluation to include file storage management software systems such as DMF and Unitree.
- ❑ Explore scalable technologies, such as network attached storage devices; high performance optical disk systems—including holostore; high performance file storage management systems (e.g., HPSS and Unitree); and high performance parallel I/O systems and standards.

Point of Contact:

Ben Kobler
Goddard Space Flight Center
kobler@nssdca.gsfc.nasa.gov
(301) 286-3553



ATM Network (ATDNet) Linking GSFC and Washington, DC Area Agencies

Objective: Provide high performance networking and advanced network services for GSFC-based ESS testbed systems and users enabling (a) distributed processing among supercomputers and workstations; (b) synchronization of sensitive communications such as video (with audio) conferencing among desktop workstations; and (c) large volume and/or high resolution image data transfers

Approach: The ESS project cooperates with the HPCC National Research and Education Network Project and the GSFC Center Network Environment (CNE) Project on end-to-end architectures and participates in the Applications Technology Demonstration Network (ATDNet), enabling high speed access to the CM-5 at NRL. The ESS project also will establish an Asynchronous Transfer Mode (ATM) testbed at GSFC to prepare for operational deployment of new networking technologies.

Accomplishments:

- Initiated an architectural plan for a GSFC-based ATM testbed
- Developed a network design for interfacing with the initial NREN connection at GSFC
- Designed and developed a GSFC interface and node architecture for the ATDNet connection

Significance. The ATM Network is a key component for evolving to shared use of remote network resident resources by the science community. Whether ESS investigators use remote testbeds or remote investigators use ESS testbeds; whether investigators move data between distributed archives and testbeds or link various combinations of distributed testbeds; or whether high performance computing is used in some other scenario of ESS research, an evolving high performance, transparent network at GSFC is essential.

Status/Plans: Key activities in FY94 will include:

- Deploying an initial GSFC ATM testbed configuration of four workstation nodes in building 28
- Connecting to the ATDNet at 155 MBPS (pending Department of Defense funding and schedule)
- Supporting interfacing with and operational use of a NASA NREN connection at 45 MBPS

In FY95 the GSFC ATM testbed will be expanded and extended to include Cray C-98 and ESS-related personnel in a second building. Also, the interface with and operational use of a NASA NREN connection at 155 MBPS will be supported.

Point of Contact:

Pat Gary
Goddard Space Flight Center
pgary@dfnrc.gsfc.nasa.gov
(301) 286-9539

(THIS PAGE LEFT BLANK INTENTIONALLY)

Overview of ESS Applications Software Research

Goal and Objectives: The goal of the ESS applications software activities is to enable the development of NASA Grand Challenge applications on computing platforms that are evolving towards sustained teraFLOPS performance. The objectives of the applications activities are to:

1. Identify the NASA Earth and space science Grand Challenge investigations and Guest Computational investigations
2. Identify computational techniques (or "computational challenges") that are essential to the success of the Grand Challenge problems
3. Formulate embodiments of these techniques that are adapted to and perform well on highly parallel systems
4. Capture the successes in a reusable form

Strategy and Approach: The strategy of the ESS applications activities is to select NASA Grand Challenges from a vast array of candidate NASA Earth and space science problems, to select teams of aggressive scientific investigators to implement the problems on scalable testbeds, and to accelerate the progress of the investigators (and capture the results) by providing computational technique development support for solving the Computational Challenges. The approach involves using a peer reviewed NASA Research Announcement to select the Grand Challenge investigations and their investigator teams. Also, in-house teams of computational scientists have been developed at GSFC and JPL to solve the Computational Challenges.

Organization: The Office of Aeronautics and Space Technology, jointly with the Office of Space Science and Applications, selected the ESS Principal Investigators (P.I.s) through a peer reviewed NASA Research Announcement process. The ESS Science Team, comprised of these P.I.s and chaired by the ESS Project Scientist, conducts periodic workshops for the investigators and coordinates their computational experiments. The ESS Evaluation Coordinator focuses activities of the Science Team leading to development of ESS computational and throughput benchmarks. A staff of computational scientists supports the Grand Challenge investigations by developing scalable computational techniques that address the Computational Challenges.

Management Plan: At GSFC, a Deputy Project Manager for Applications directs the In-house Team of computational scientists. At JPL, a Deputy Task Leader performs the same function. ESS and its investigators contribute annual software submissions to the High Performance Computing Software Exchange.

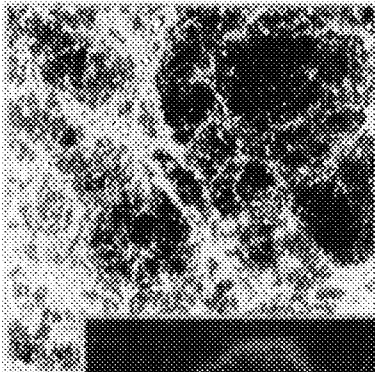
Point of Contact:

Steve Zalesak
Goddard Space Flight Center
zalesak@gondor.gsfc.nasa.gov
(301) 286-8935

Robert Ferraro
Jet Propulsion Laboratory
ferraro@zion.jpl.nasa.gov
(818)354-1340

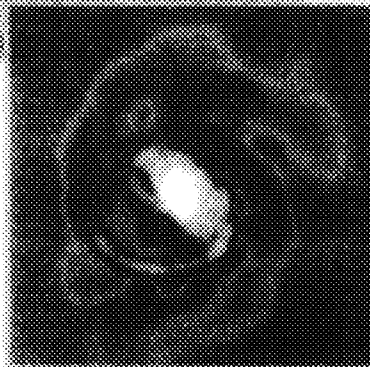
ESS Grand Challenge PI Investigations

Cosmology and Accretion Astrophysics
Wojciech Zurek
Los Alamos



**Selected through the
NASA Research
Announcement
Process**

**Convective Turbulence
and Mixing in
Astrophysics**
Robert Rosner
University of Chicago



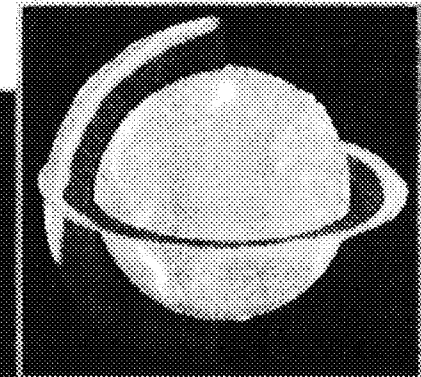
**Large Scale Structure and
Galaxy Formation**
George Lake
University of Washington

**Knowledge Discovery in
Geophysical Databases**
Richard Muntz
UCLA



**Solar Activity and
Heliospheric Dynamics**
John Gardner
NRL

Climate Models
Max Suarez
NASA Goddard



**Atmosphere/Ocean Dynamics
and Tracers Chemistry**
Roberto Mechoso
UCLA

**Four-Dimensional
Data Assimilation**
Richard Rood
NASA Goddard

Grand Challenge Science Investigations

Objective: Select NASA Grand Challenge scientific investigators who will provide a means to rapidly evaluate and guide the maturation process for scalable parallel algorithms and system software and thereby reduce the technical risks for later ESS Grand Challenge researchers when adopting similar technologies.

Approach: Issue a NASA Research Announcement (NRA) internationally requesting proposals for Grand Challenge investigations across all NASA Earth and space science. Select collaborative multidisciplinary Principal Investigator Teams that include physical and computational scientists, software and systems engineers, and algorithm designers. Also, select Guest Computational Investigators to develop specific scalable algorithmic techniques. Form the selected teams into an ESS Science Team to organize and conduct joint computational experiments.

Accomplishments: The NRA process was completed in late FY92. In FY93 awards were made to eight Principal Investigator Teams and ten Phase-1 Guest Computational Investigators. Awards to eleven Phase-2 Guest Computational Investigators will take place in FY94. The eight Principal Investigators and their research are:

John Gardner, Naval Research Laboratory, *Understanding Solar Activity and Heliospheric Dynamics*

George Lake, University of Washington, *Large Scale Structure and Galaxy Formation*

Roberto Mechoso, UCLA, *Development of an Earth System Model: Atmosphere/Ocean Dynamics and Tracers Chemistry*

Richard Muntz, UCLA, *Data Analysis and Knowledge Discovery in Geophysical Databases*

Richard Rood, NASA Goddard Space Flight Center, *High Performance Computing and Four-Dimensional Data Assimilation: The Impact on Future and Current Problems*

Robert Rosner, University of Chicago, *Convective Turbulence and Mixing in Astrophysics*

Max Suarez, NASA Goddard Space Flight Center, *Development of Algorithms for Climate Models Scalable to TeraFLOPS Performance*

Wojciech Zurek, Los Alamos National Laboratory, *Application of Scalable Hierarchical Particle Algorithms to Cosmology and Accretion Astrophysics*

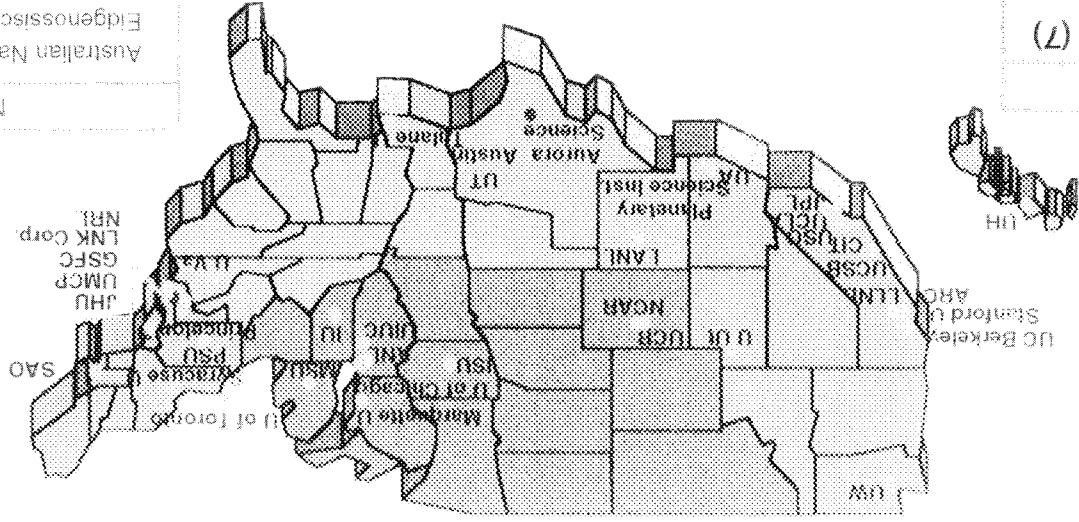
Significance: The diverse eight Principal Investigators and their teams encompass scientific and computational expertise crucial to the development of multidisciplinary approaches needed in the pursuit of NASA ESS Grand Challenges and the evolution of scalable high performance computational techniques. The collaboration developed during the NRA process among the Office of Aeronautics (Code R), the Office of Earth Science (Code Y), and the Office of Space Science (Code S) continues to strengthen and keep the Code R ESS activity highly relevant to the NASA science community.

Status/Plans: Year-2 awards will be made early in FY94. User support, training, and computational challenge development support will be provided for the investigators during the year, allowing them to complete their annual evaluations and reports at the end of FY94. The third and fourth Science Team meetings will be held in FY94

Point of Contact:

Jim Fischer
Goddard Space Flight Center
fischer@nibbles.gsfc.nasa.gov
(301) 286-3465

ESS Science Team

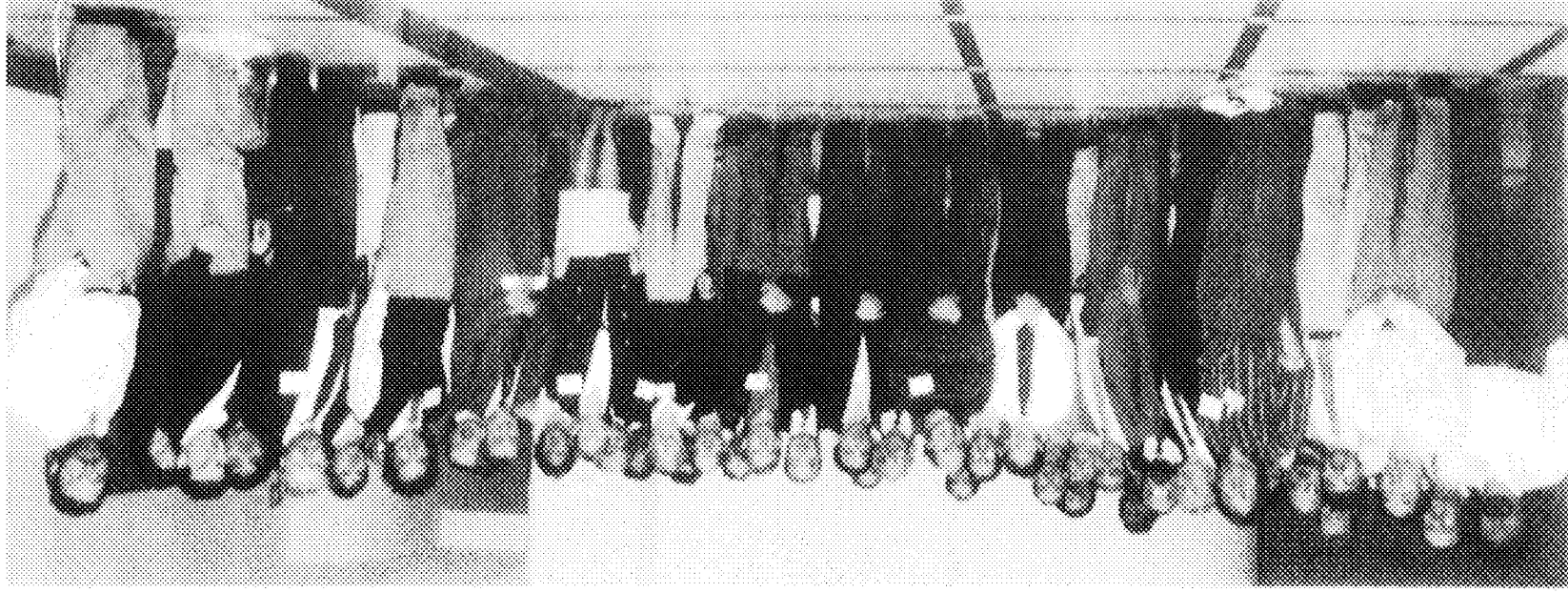


Non-US Sites

- Australian National University
- Eidgenössische Hochschule, Zurich
- Observatoire de Nice (France)
- University of Padua (Italy)

Legend

- Government Site (7)
- University (29)
- Industry (2)



ESS Science Team

Objective: Provide a forum for the ESS Grand Challenge Science investigators to meet, to carry out collaborations, and speak with one voice to the ESS Project and to the science community as a whole on topics such as their work on parallel algorithm development and parallel testbed systems and their recommendations to NASA.

Approach: The ESS Project organizes at least two annual meetings of the ESS Science Team. These meetings are co-chaired by the ESS Project Scientist and the ESS Project Manager. The ESS In-house Team and the ESS Evaluation Coordinator also actively participate. In addition to reports from the Project and discussions led by the investigators, unstructured time is provided to foster discussions among the investigators and the ESS In-house Team. The Science Team is encouraged to develop joint studies and publications of its own.

Accomplishments:

- ❑ Seventy-six people attended the first ESS Science Team meeting held at GSFC between January 27 and 29, 1993.
- ❑ Conducted the second Science Team meeting in Pittsburgh on May 3, 1993, the day before the Federal Multi-Agency Grand Challenge Workshop.
- ❑ The ESS Science Team participated in the Federal Multi-Agency Workshop on Grand Challenge Applications and Software Technology between May 4 and 7, 1993 in Pittsburgh. All eight P.I. Teams made presentations of their investigations during the Grand Challenge Applications Panel Sessions. All ESS representatives participated in the workshop working groups.
- ❑ Planned the third meeting of the Science Team to be held on November 15, 1993 in conjunction with Supercomputing '93 in Portland, Oregon.

Significance: By encouraging the Science Team to achieve its own identity and to conduct collaborative research, the ESS Project expects to generate knowledge and software that would not have been produced from more structured activity.

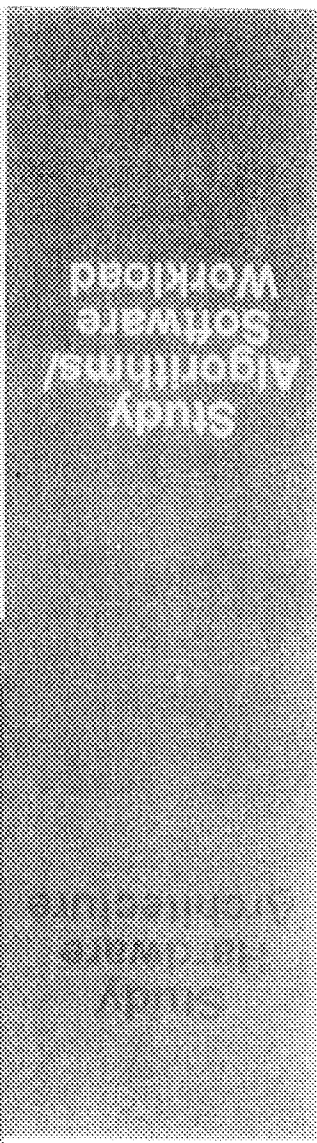
Status/Plans: The third and fourth Science Team meetings will be held in FY94. During FY94, two technical meetings will be held at Goddard with the technical staffs of each of the eight P.I. Teams to share information between each group and the ESS In-house Team. Each P.I. Team will meet separately to provide an opportunity for focused discussions that are not normally possible at the larger Science Team meetings.

Point of Contact:

Mel Goldstein
Goddard Space Flight Center
u2mlg@cdc910b2.gsfc.nasa.gov
(301) 286-7828

ESS EVALUATION PROGRAM COMPONENTS MATRIX

GOALS	TASKS	TOOLS
<p>SPC Procurements</p> <p>Scalable Architecture Structures</p> <p>Evaluation Results</p> <p>Performance</p> <p>Efficiency</p> <p>Scaling</p> <p>Sensitivities</p> <p>TeraFLOPS Potential</p> <p>Performance Prediction</p>	<p>Models</p> <p>Statistical/Analytical</p> <p>Queueing</p> <p>Petri</p> <p>Resource Characterization</p> <p>Throughput Measurements</p> <p>Execution Budget</p> <p>Starvation</p> <p>Contention</p> <p>Latency</p> <p>Overhead</p> <p>Perturbation Studies</p> <p>ESS Parallel Benchmarks</p> <p>Workload Characterization</p> <p>Models</p> <p>Statistical/Analytical</p> <p>Markov</p> <p>Parallelism</p>	<p>Testbeds</p> <p>ESS</p> <p>Univ. Research</p> <p>NSF Centers</p> <p>Metrics</p> <p>Measurements</p> <p>Paragraph (ORNL)</p> <p>Pablo (UIUC)</p> <p>AIMS (ARC)</p> <p>Languages/Compilers</p> <p>Earth and Space Science Programs</p> <p>ESS Science Team</p> <p>Investigators</p> <p>GSFC In-House Team</p> <p>JPL</p> <p>Metrics</p> <p>Measurements</p> <p>SITA (McGill)</p> <p>MICS (UCIC)</p>



ESS Testbed Evaluation and Collaboration with NSF

Objective: The ESS testbed evaluation activity has three objectives:

1. Provide metrics, measurements, and understanding of factors determining HPC system performance for ESS Grand Challenge applications, i.e., "close the loop" in HPC science for NASA HPCC ESS research
2. Develop the ESS Parallel Benchmark (EPB) suite for comparative studies of competing HPC systems
3. Disseminate findings to computational scientists, HPC hardware and software vendors, and the HPCC community

Approach: The ESS testbed evaluation activity is an innovative approach to testbed evaluation. It involves the coordinated development and implementation of a variety of metrics, measurements, and evaluation methodologies among ESS research sites as well as NSF research centers. Shown is a matrix of the components of the ESS evaluation program that relates (a) the tools to be used, (b) the tasks to be performed, and (c) the goals to be achieved in studies of the architecture, the systems, and the algorithms and software workload. Specific aspects of the approach are

- Perform evaluation studies through measurement, modeling and analysis
- Formulate, characterize, and analyze a workload set reflecting ESS requirements derived from ESS Science Team contributions
- Derive an ESS Parallel Benchmark suite exhibiting workload characteristics
- Determine an ESS Testbed execution budget for the ESS workload and derive sensitivities and scaling attributes for performance prediction
- Compare and contrast results with general findings from the Joint NSF-NASA Initiative in Evaluation (JNNIE)

Accomplishments:

- Initiated the ESS Evaluation activity

- Formulated the overall goals, approach, and methods
- Acquired measurement and analysis tools
- Assembled an initial set of ESS example codes
- Established the evaluation team with academic representation
- Represented NASA HPCC ESS in the JNNIE

Significance: The ESS evaluation activity is central to the success of the ESS project in achieving teraFLOPS capability by the end of the decade. Evaluation is critical to determining the performance, efficiency, and scalability of massively parallel processing architecture testbeds and software technology applied to the Earth and space science computational demands. The findings of this activity will be crucial to determining effectiveness of experimental computational techniques, assessing the potential prospects for useful teraFLOPS computing as applied to ESS problems, and to future procurement decisions, influencing future vendor offerings in architecture and software.

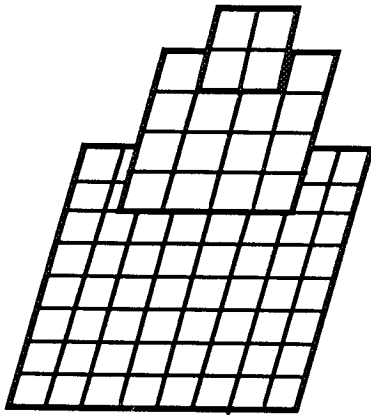
Status/Plans: In FY94, measurement and analysis tools will be fully integrated as part of an evaluation methodology including necessary ports and enhancements. An initial workload test set will be derived from example ESS codes and analyzed for architecture independent characteristics. Performance and behavior measurements on ESS testbeds will be conducted to determine behavior factors, efficiency, and bottlenecks. Initial analytical models will be derived for scaling and sensitivity studies. Evaluation results from ESS Science Team investigators will be coordinated to establish uniform metrics and reporting.

Point of Contact:

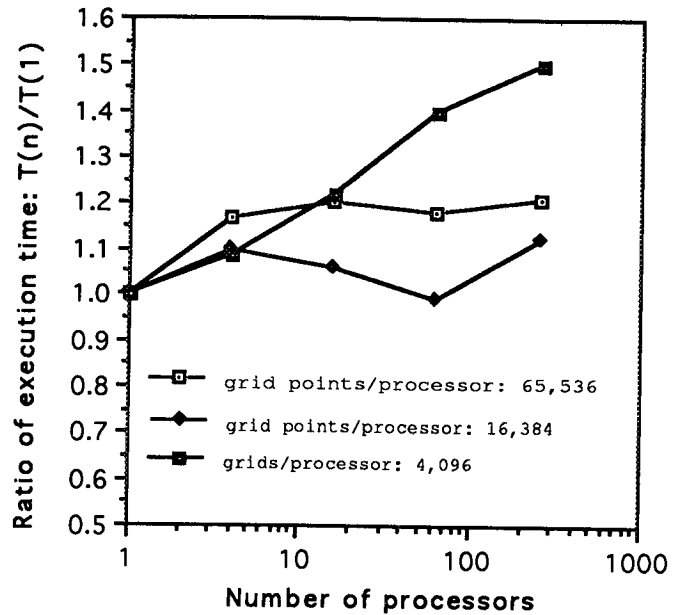
Thomas Sterling
USRA-CESDIS
Goddard Space Flight Center
tron@valor.gsfc.nasa.gov
(301) 286-2757

Performance Evaluation of Parallel Numerical Library

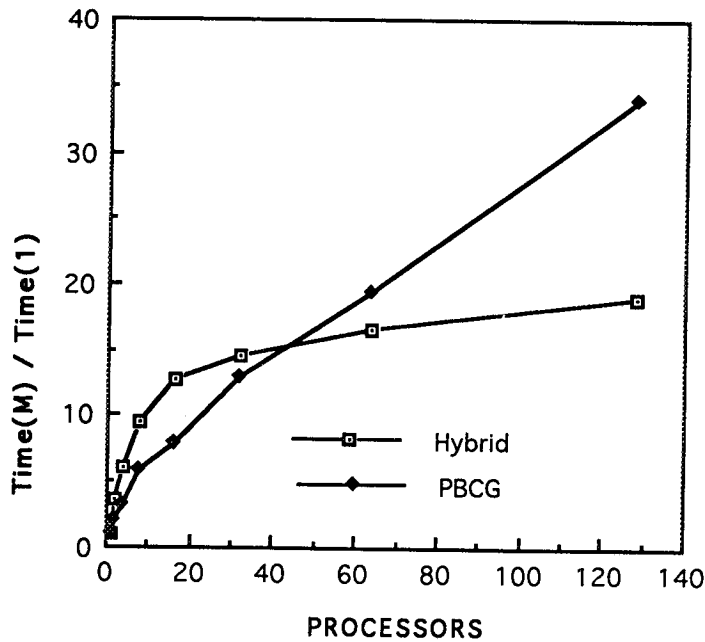
An example of grid hierarchy with finest grid size 8x8



Scaling Test on Delta Machine:
2D Multigrid Poisson Solver



Scaling of Sparse Matrix Parallel Solver



The problem size scales with the number of processors such that each processor contains fixed 1600 nodal grids. $T(M)$ is the time for solving a 1600M grid problem on M processors. $T(1)$ is the time for solving a 1600 grid problem on one processor. The pre-conditioned conjugate gradient method scales linearly as expected. The hybrid method scales as square root of M, which is superior to PBCG for large problems.

MIMD Numerical Techniques Library

Objective: Develop efficient parallel algorithms and implementations for a library of multigrid, sparse matrix and multidimensional Fast Fourier Transform (FFT) techniques for solving partial differential equations (PDEs) on JPL/Caltech parallel system testbeds.

Approach: Develop scalable parallelization strategies and efficient implementations of multigrid and sparse matrix algorithms; develop both library-quality and skeleton-type codes using object-oriented design and programming techniques and C, C++ and Fortran languages. Develop code that is nearly optimized on the JPL/Caltech testbeds and easily ported to other hardware platforms. Identify numerical PDE algorithms that are interesting to the ESS science team and for which numerical library or skeleton-type codes can be used. Validate the quality and performance of the codes on ESS Grand-Challenge applications.

Accomplishments: Identified numerical PDEs which are of interest to ESS Grand Challenge investigators and for which multigrid or sparse matrix techniques can be applied; developed a 2-D parallel multigrid code that solves Poisson equations on a structured finite difference grid. Initial evaluation of the parallel multigrid Poisson solver has been performed. The interface for the sparse matrix solver was restructured. Linear system solvers were completed and a sample case designed. The 3-D FFT code with slab data distributions has been completed and tested. Communication routines for rod data distributions were completed. An illustration of the implementation strategy for this solver and its performance on Caltech's Delta machine are shown on the opposite page.

Significance: Multigrid algorithms are a class of highly efficient iterative solvers for solving numerical PDEs arising from science and engineering problems, including (magnetic) fluid dynamics and electromagnetics. Multigrid algorithms are not used widely among non-specialists because they are both mathematically demanding and difficult to implement.

This work should result in high-performance, high-quality multigrid software that can be used by Grand Challenge investigators in their applications. The sparse matrix solvers and FFT routines can be used as parallel library routines

in a variety of scientific computing applications. The sparse matrix solver is useful for computations on unstructured meshes or grids.

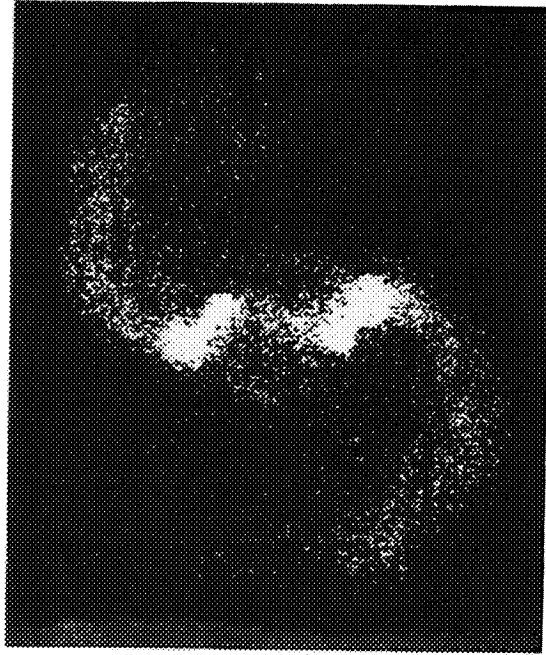
Status/Plans: Extend the parallel multigrid Poisson solver to 3-D problems. Develop multigrid solvers for other types (nonlinear, time-dependent) of PDEs. Complete the sparse matrix solver interface and add more sample applications. Complete the 3-D FFT for rod data distributions and optimize the detailed implementation on communications. Evaluate performances of the codes developed.

Point of Contact:

John Lou
Jet Propulsion Laboratory
lou@acadia.jpl.nasa.gov

Hong Ding
Jet Propulsion Laboratory
hding@redwood.jpl.nasa.gov

Parallel Computational Techniques



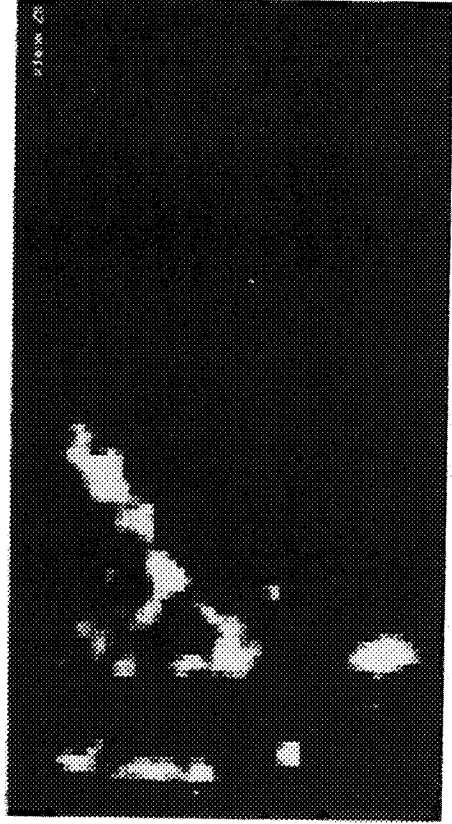
N-Body Problem



Piecewise Parabolic Method (PPM)



Flux Corrected Transport (FCT)



Particle-In-Cell (PIC)

Parallel Computational Technique Kernels

Objective: Assist the ESS Project Manager in understanding, tracking, coordinating, and assessing the parallel computational techniques work of the Grand Challenge investigators.

Approach: Develop parallel computational techniques that address the Computational Challenges of the ESS Grand Challenge investigations.

Accomplishments: Identified the computational challenges of the ESS investigations as very broad. They include tree codes, smooth particle hydrodynamics, pseudo-spectral codes, Piecewise Parabolic Method/Godunov codes, multigrid codes, 3-D Flux-Corrected Transport, elliptic solvers, finite differencing schemes, spectral codes, Monotonic Lagrangian Grid, 3-D Kalman filter, semi-lagrangian codes, grid-point codes, image analysis and classification. Four examples of the computational challenges are shown in the figure on the opposite page and described below:

- Upper Left. Gravitational N-body simulation of interacting disk galaxies performed on the 16 384 processor MasPar MP-1 using 65 536 particles. The simulation was run for 3 000 time steps, and each time step required an average of 80 seconds of computing. The speeds achieved are comparable to a single head of a Cray Y-MP. (K. Olson/USRA, J. Dorband/GSFC)
- Upper Right. Two-dimensional simulation of a Rayleigh-Taylor instability in a compressible fluid performed on the MasPar MP-1 using the Piecewise-Parabolic Method for hydrodynamics. The calculation used a grid size of 512x2048. The double precision version of the code (converted to MPL by John Dorband) runs at 350 MFLOPS. This is twice the speed obtained on a Cray Y-MP processor. A single precision version written in MasPar Fortran runs at 500 MFLOPS, which is comparable to the performance obtained with a Cray C-90 processor. (B. Fryxell/USRA)
- Lower Left. Parallel implementation of a Flux-Corrected Transport scheme performed on 64 nodes of the Intel Delta using a 68x68x64 grid. Shown is an isosurface of

density at time = 1.9 (4 000 steps) and represents about six hours of computing. The code takes about 5.5 seconds per step on 64 nodes and about 7 seconds per step on one Y-MP processor. (A Deane/USRA)

- Lower Right. PIC simulation of solar wind interacting with the Earth's magnetosphere generated on the MasPar MP-1. Performance analysis shows the MP-1 and MP-2 can achieve 34% and 110% of the performance of one C90 processor respectively. The C90 single processor implementation of this code averaged 260 MFLOPS. (P. MacNeice/HSTX)

Significance: Repeated experience has proven that taking existing serial code and trying to port it to scalable parallel systems is not only an arduous exercise, but results in poor performance. For this reason important computational kernels are being totally rewritten with the considerations of the parallel architectures in mind.

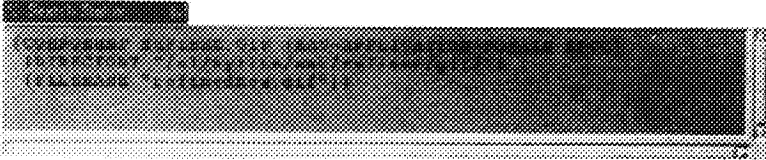
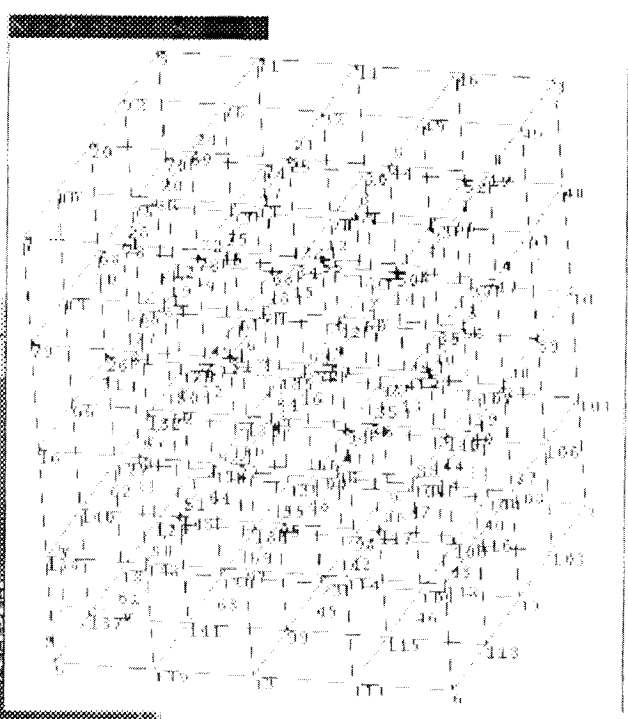
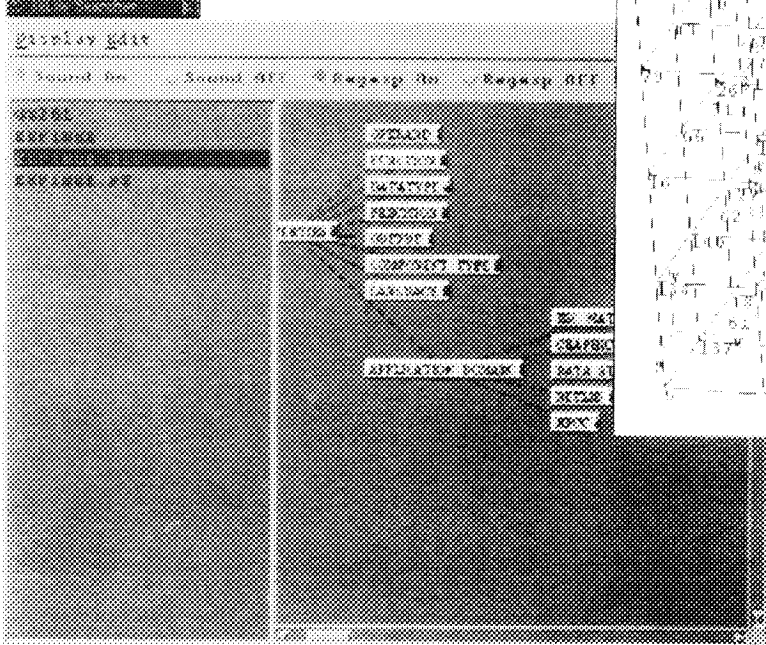
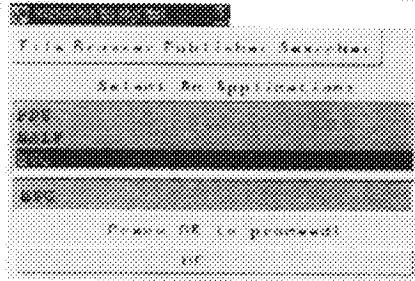
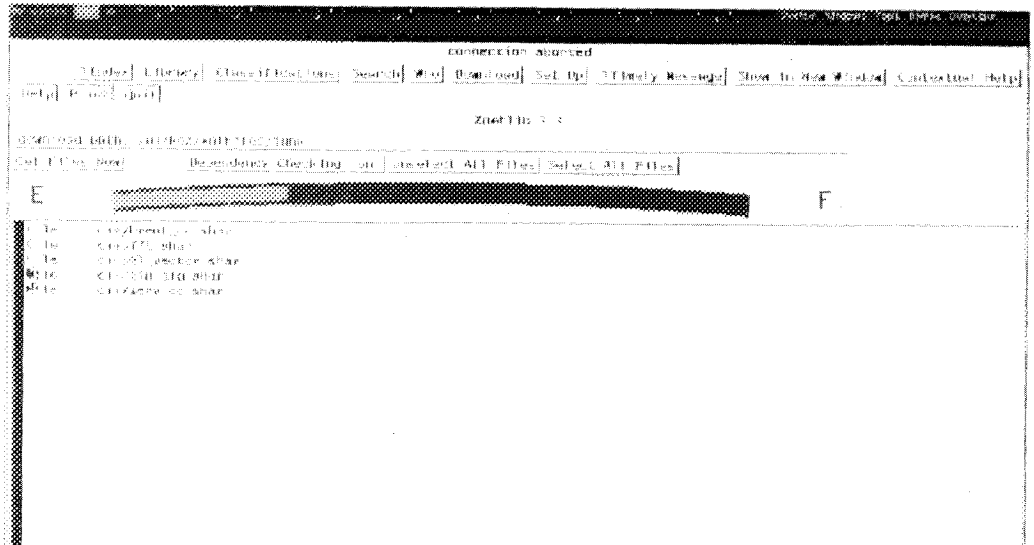
Status/Plans: In FY94 these and other kernels will be written for additional machines of interest such as the Touchstone series, the KSR-1, the CM-5, and the Cray T3D. The simple structure of the kernels will allow the code to be rewritten in an unconstrained manner, taking advantage of the varying features of the programming environments available on each of these systems. Focus of the in-house development will be adjusted based on Science Team collaborations. Selected techniques will be completed and delivered to the ESS investigators and the HPCC community.

Point of Contact:

Steve Zalesak
Goddard Space Flight Center
zalesak@gondor.gsfc.nasa.gov
(301) 286-8935

JPL/ESS Software Repository Front Ends

XNetLib



HyLiTe

ESS/JPL Software Repository

Objective: Establish a software repository at JPL for HPCC ESS project. Provide mechanisms for the user community to access and download individual routines or software packages.

Approach: Since there exist many software repository access mechanisms, existing software repository access mechanisms were surveyed. The evaluation criteria used were ease of use by a novice, visual organization, responsiveness across T1 network connection, and search and browse capabilities.

Accomplishments: Two access mechanisms were chosen and installed. The first one is NetLib (Network Library) developed at the Department of Energy's Oak Ridge National Laboratories and the second is HyLiTe (Hypermedia Library Technology) developed at the Jet Propulsion Laboratory.

NetLib was chosen because it is widely used in the scientific community and therefore is well known. It also works well and reliably through both character and X Window interfaces. HyLiTe was chosen as an experimental interface since it represents a more comprehensive way of general component classification. It is a multimedia application and as such can be used to catalog, peruse and download not only software, but speech, motion picture, still pictures, diagrams, etc. HyLiTe has a more user friendly X Windows interface wherein a user can traverse a property tree or issue queries against it.

The attached figures illustrate the graphical user interfaces of HyLiTe and NetLib. The HyLiTe interface is shown with a sample property tree opened. The list of components and one of the components itself, which happens to be a GIF format picture of a refined 3D mesh, is also shown.

Significance: A choice between NetLib and HyLiTe will provide the user community with two easy-to-use and comprehensive tools to upload and download software produced for HPCC ESS project.

Status/Plans: The ESS Software Repository is in the process of being populated with an initial set of software developed at JPL. The HyLiTe front end is not yet ready for general

public use, since its graphical drawing performance over the network is being improved.

Point of Contact:

Koz Tembekjian
Jet Propulsion Laboratory
koz@zion.jpl.nasa.gov
(818) 354-8403

(THIS PAGE LEFT BLANK INTENTIONALLY)

Overview of ESS System Software Research

Project Goal and Objectives: The goal of the ESS systems software research activities is to make future teraFLOPS computing systems significantly easier to use than early 1990s conventional vector processors for ESS Grand Challenge applications. The objectives are to:

1. Identify and remove system software weaknesses that are obstacles to NASA's eventual integration of scalable systems into production computing operations
2. Identify and develop system software components that make scalable systems easier to use than current vector processors

Strategy and Approach: Several focused, high payoff projects are being supported in important topic areas for which in-house expertise is available to provide technical direction. At GSFC, these areas are (1) the achievement of effective and efficient architecture-independent parallel programming; (2) development of data management strategies and tools for management of petabytes (10^{15}) of data; (3) implementation of advanced visualization techniques matched to teraFLOPS system requirements; and (4) development of the Federal High Performance Computing Software Exchange. At JPL, the topic areas are (1) tools for the numerical solution of partial differential equations on parallel computers—including parallel unstructured mesh generation; (2) investigation of new parallel programming paradigms applied to science applications; (3) skeleton parallel implementations of popular numerical methods; (4) investigation of automated dynamic load balancing mechanisms; and (5) systolic data flow tools for high throughput data processing applications.

Organization: Extensive collaboration with the academic and vendor communities is expected to assist the architecture-independent programming work. The visualization activity is collaborating with Sterling Software. The Software Exchange collaborates with ARPA, DOE, EPA, NIST, NOAA, NSA, and NSF, and receives interagency oversight from the Federal Coordinating Committee for Science, Engineering and Technology Working Group on Scientific and Engineering Computing. The parallel programming paradigm investigation is being performed in collaboration with the University of Virginia, and collaborations with other academic research institution are anticipated. JPL also supports basic research at the California Institute of Technology on architecture features that pertain to data movement.

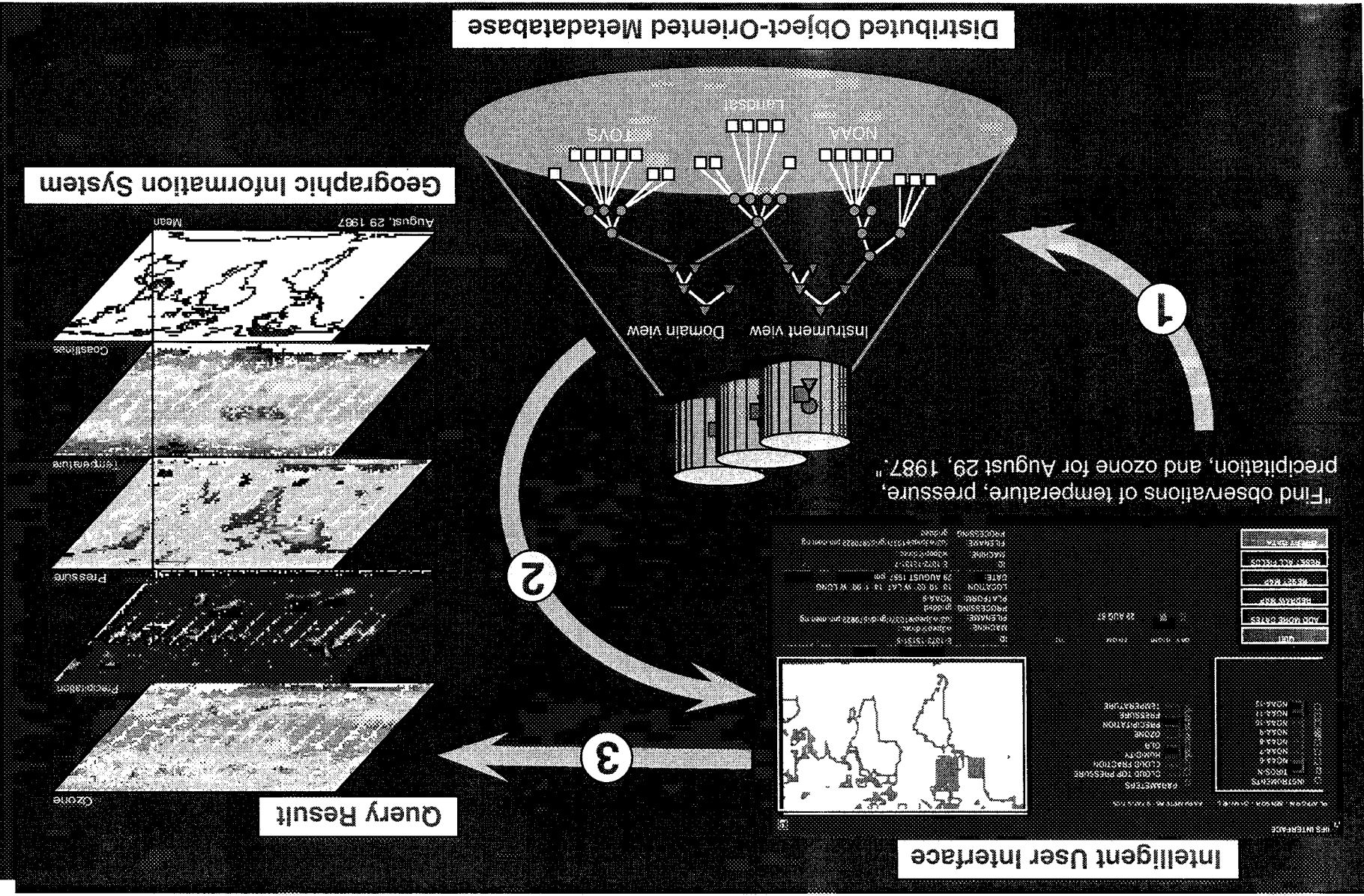
Management Plan: At GSFC a Deputy Project Manager for Systems Software Research directs sub-element activities. At JPL, a Deputy Task Leader performs the same function.

Point of Contact:

John E. Dorband
Goddard Space Flight Center
dorband@nibbles.gsfc.nasa.gov
(301) 286-9419

Robert Ferraro
Jet Propulsion Laboratory
ferraro@zion.jpl.nasa.gov
(818) 354-1340

NASA / GODDARD SPACE FLIGHT CENTER INFORMATION SCIENCE AND TECHNOLOGY OFFICE / CODE 930.1



Advanced Data Management Technologies

Objective: Develop efficient algorithms for SIMD and MIMD machines to automatically extract image content, organize and manage databases, and enable efficient methods of browsing large, complex spatial databases through faster querying methods.

Approach: The approach consists of developing (1) algorithms for automatic georegistration of spatial data sets, (2) rapid access indices for searching large, complex spatial databases, and (3) techniques for spatially organizing satellite observations and for automatic extraction of metadata from imagery.

Accomplishments:

- Completed the third draft of a "White Paper" covering the state-of-the-art in data management on high performance machines.
- Designed and implemented a probabilistic neural network on the MasPar for satellite image characterization.
- Designed and implemented a decision tree algorithm on a massively parallel machine (MasPar).
- Developed a generic genetic algorithm and applied it to the task of unsupervised clustering of remotely-sensed image data.
- Funded a study that produced a report on *The Impact of Mass Storage on Future NASA Computing Capabilities and Missions*.
- Formed a GSFC Pathfinder data team for preliminary requirements analysis to apply intelligent data management techniques.
- Served as the Technical Monitor for a Phase-1 Small Business Innovation Research (SBIR) study of the use of Gabor filters for image feature recognition.
- Wrote and submitted for publication, five papers describing results.

Significance: The White Paper is a living document on performance results of evolving technologies and will serve as documentation

of future research directions to avoid duplication of efforts. The neural network performed several orders of magnitude faster than serial machines. Decision trees and genetic algorithms are used to validate, classify and optimize data characterization (machine learning).

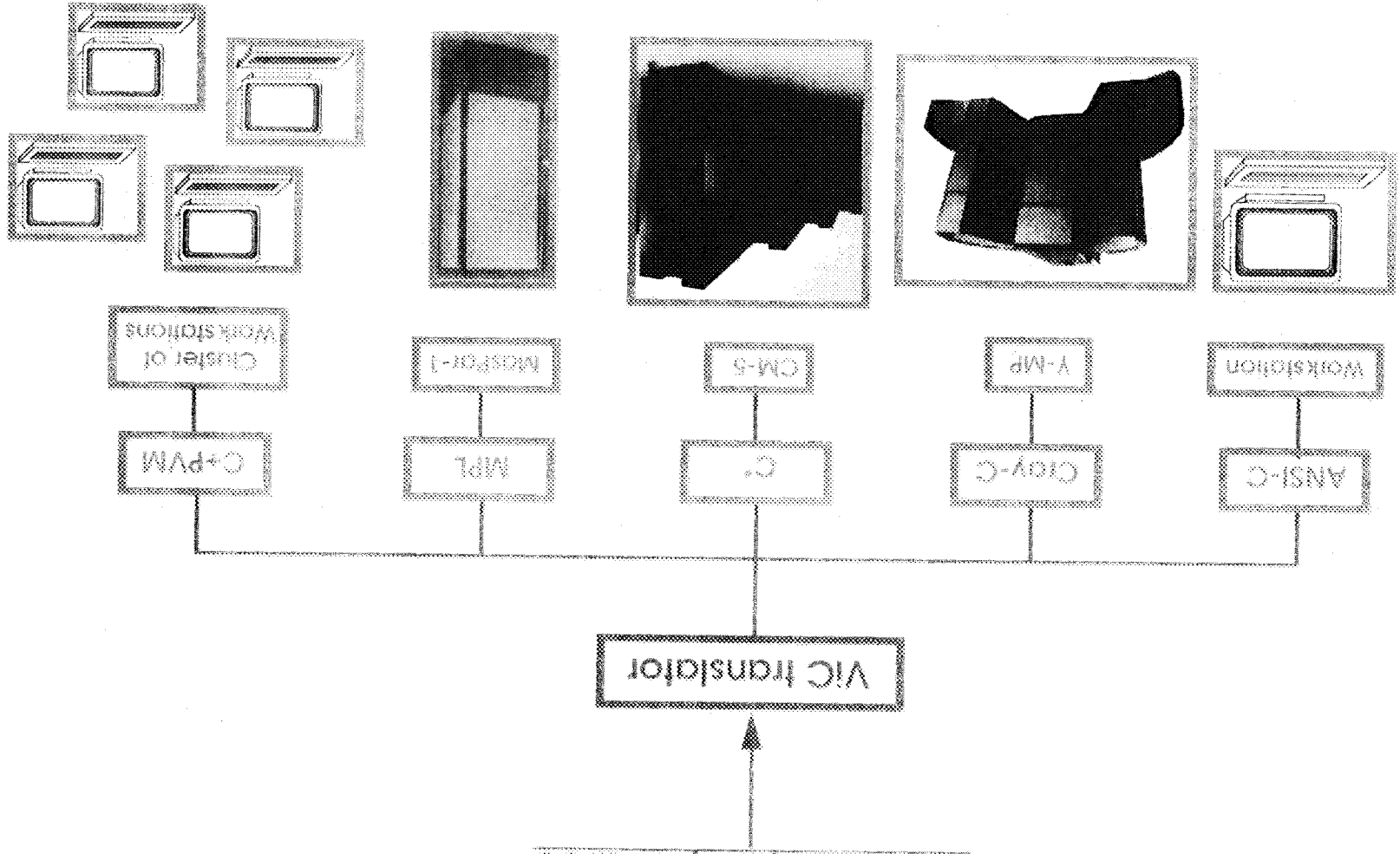
Status/Plans: Form a team with GSFC Pathfinder scientists and apply intelligent information fusion system to that domain.

Point of Contact:

William J. Campbell
Goddard Space Flight Center
campbell@nssdca.gsfc.nasa.gov
(301) 286-8785

Virtual C Translator

Virtual C (VIC) Source



Architecture-Independent Programming Paradigms

Objective: The objective of the architecture-independent programming paradigms activity is to develop styles for writing parallel applications that enhance their execution efficiency and make them portable over the NASA HPCC testbeds.

Approach: Computational Fluid Dynamics (CFD) applications naturally contain element-wise parallel operations which are supported by most MPP vendors. But these applications also contain communications, input-output, and initialization operations that are vendor specific. Until a standard portable data parallel language is generally available, the ViC translator allows the application element-wise parallel component to be vendor independent. A translator approach has been chosen. This avoids any code rearrangement; thus, the vendor's native compiler and debugger tools are still straightforward to use.

This approach is unique in its simplicity, allowing a straightforward implementation on a wide spectrum of parallel computer architectures. The macros implement the element-wise, parallel component of the application in restricted, portable subset of C*. The remainder of the application performs the support functions of communications, input-output, and initialization operations contained in a separate source text. This is a two language model, consisting of ViC, and the native parallel language. The number of auxiliary support functions is typically small, and thus written in the native language tools.

Accomplishments: A single source text of a prototype CFD application has been run and verified on a cache-based Silicon Graphics workstation, a vectorizing Cray YMP, a virtualized data-parallel TMC CM-5, an unvirtualized data-parallel MasPar MP-1, and a cluster of workstations using PVM. A library of support functions was developed for each platform. Results were reviewed and presented at Parallel CFD 93.

Significance: Portable languages for parallel computers are still on the horizon. ViC allows a class of applications to be portable today because the machine dependent part is explicitly localized and manageable.

Status/Plans: The current implementation is a macro package that uses the two standard

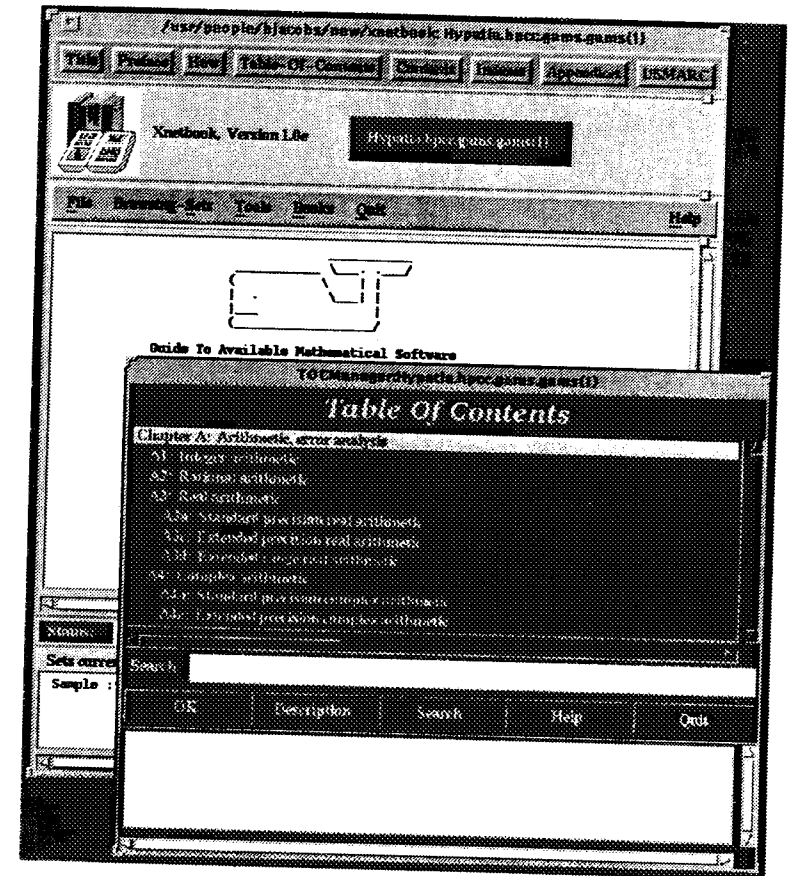
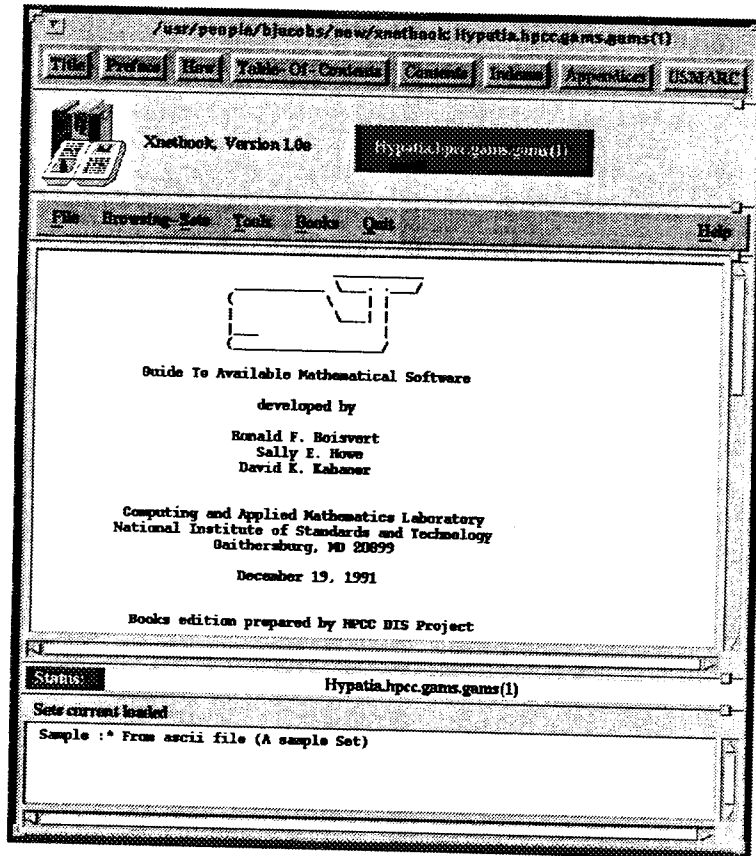
UNIX preprocessors (M4 and CPP). An effort is underway to convert the programming environment from a macro package to a full translator. This translator will allow error checking, which is not practical using the macro tools.

Point of Contact:

Clark Mobarry
Goddard Space Flight Center
mobarry@nibbles.gsfc.nasa.gov
(301) 286-2081

HPC-Software Exchange

84



HPCC Software Exchange

Objective: The NASA HPCC Software Exchange is designed to facilitate the exchange and reuse of software. Its specific objectives are to

1. Develop and demonstrate a distributed architecture and supporting technology that support software exchange
2. Implement an initial distributed HPCC Software Exchange that supports the needs of the HPCC community
3. Specify an open, non-proprietary architecture that will facilitate the emergence of a national software exchange

Approach: The Software Exchange activity will develop several architectural elements. For each element the software exchange activity will support multiple approaches, apply metrics to evaluate multiple approaches, and develop non-proprietary/open architecture specifications. This will be done in conjunction with a community-based Open Architecture Working Group. The overall architecture treats the Internet as one large "logical library" in which all the databases (e.g., repositories) and search mechanisms (e.g., directories, catalogues, etc.) appear as items on the "shelves".

Accomplishments: In FY93 the HPCC Software Exchange activity had several notable accomplishments:

- Developed a client server book (see figure on opposite page) corresponding to NIST's GAMS book and library systems.
- Developed the Karl book publisher tool.
- Implemented the HPCC Library Catalogue, Software Union Catalogue and Directory of Software Repositories.

Significance: The development of a client server book and library systems moves the system from a centralized one to a totally distributed one. These can result in a possible 10-fold increase in several areas:

1. The number of concurrent users of the HPCC Software Exchange

2. The number of books being built in the Exchange (because the development of the Karl book publisher tool greatly aids the book building process)

3. The number of software repositories and software assets managed by the HPCC Software Exchange (because the HPCC Library Catalogue, Software Union Catalogue and Directory of Software Repositories lay the framework for the scaling up of the Exchange)

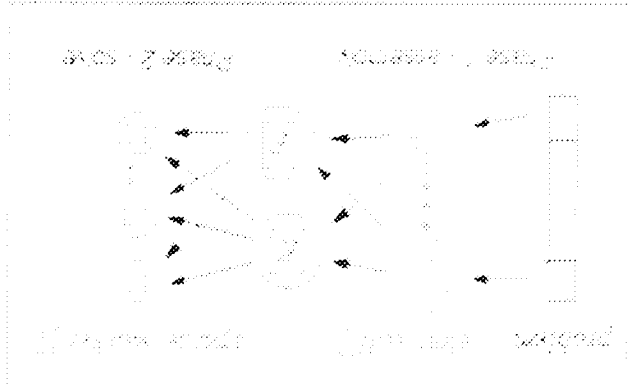
Status/Plans: In FY94 the HPCC Software Exchange Prototype System will be scaled up. The planned tasks include the incorporation of all ten Federal agencies, 20 software repositories, and 30 000 software assets and 2 000 users.

Point of Contact:

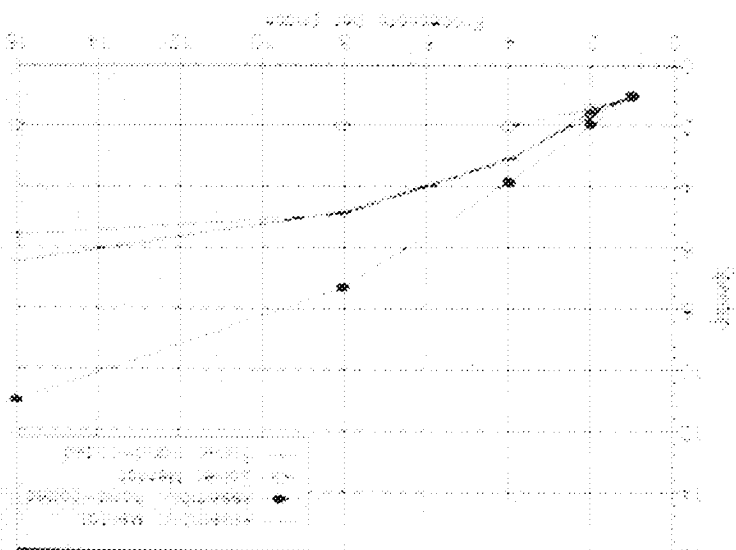
Barry Jacobs
Goddard Space Flight Center
bjacobs@nssdc.gsfc.nasa.gov
(301) 286-5661

Applying Mental to a Finite Element Code

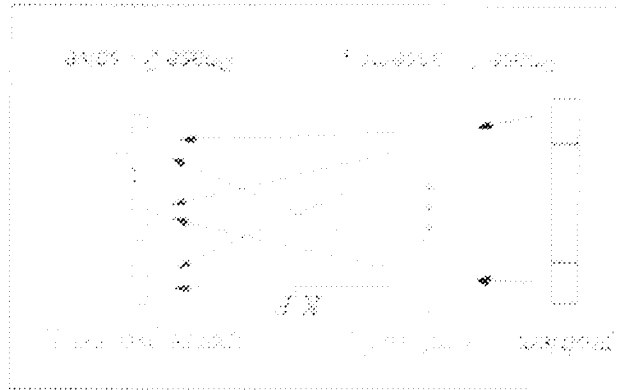
Initial Implementation of the FEM code in Mental



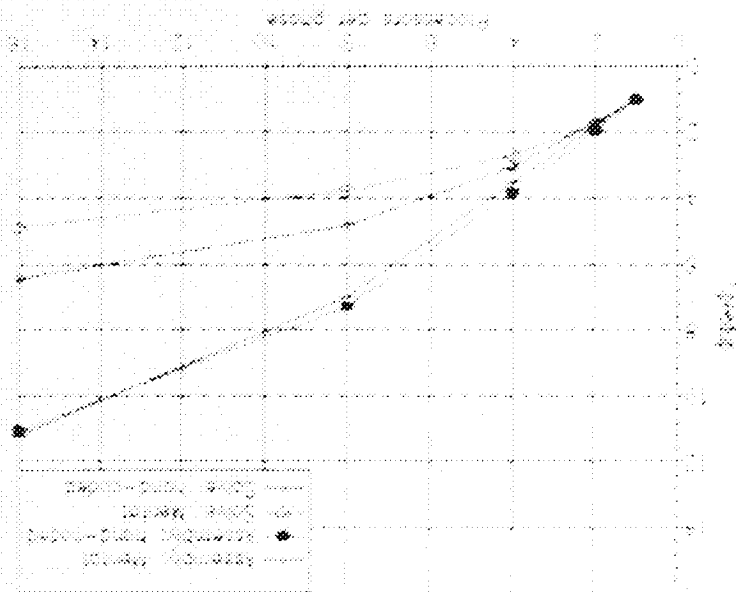
The initial implementation contained a serial bottleneck in assembling the sparse matrix K and right hand side F before beginning the sparse matrix solve.



Comparing the time spent in various stages of the FEM code in Mental.



The optimized implementation eliminated the bottleneck by breaking pieces of K and F directly to the sparse matrix workers.



Comparing the time spent in various stages of the optimized FEM code in Mental.

Optimized Implementation - Breaking the Encapsulation

Mentat for Science Applications on the ESS/JPL Testbed

Objectives: Accelerate the development of new parallel programming paradigms, and evaluate their utility for scientific programming. Assist the transition from the explicit message passing paradigm on distributed memory MIMD architectures. Evaluate the paradigms, and facilitate their revisions to make them more appropriate for scientific applications, and more usable for novel approaches to programming parallel architectures.

Approach: Many implementations of new parallel programming paradigms exist, but have been tested on only a limited set of kernel problems. A variety of such paradigms are being explored by

1. Collaborating with paradigm developers to provide an implementation on the JPL/ESS testbed
2. Porting two message passing science applications plasma PIC simulation code and (finite element electromagnetic scattering code) to the paradigm
3. Evaluating ease in porting or rewriting the applications, evaluating performance in the new paradigm against the message passing implementation, and encouraging modifications to the paradigm or its implementation to correct discovered deficiencies

Accomplishments: The University of Virginia (UVA) is providing an object-oriented parallel processing environment, Mentat, on the Intel iPSC/860. Mentat is based on C++ with extensions, and provides semi-automatic parallelization through runtime scheduling and placement of parallel objects. The UVA collaborators initially implemented the finite element code by converting it to C++, then to Mentat, using the natural object structure inherent in the finite element method. This conversion (about 8,000 lines of Fortran code) took less than 2 man-months. But the natural encapsulation introduced sequential bottlenecks that inhibited performance. A redesign changed the character of the objects to remove the bottlenecks. This resulted in scaling performance on small test cases that was within 50 percent of the message passing implementation.

Significance: The redesigned code indicates that 0-0 approaches are viable for writing science applications, but that work remains to improve the compiler's ability to generate efficient code.

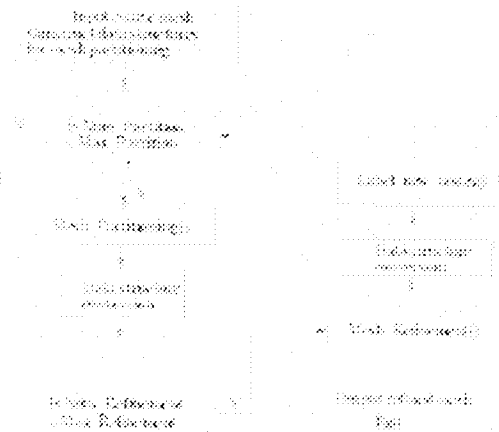
Status/Plans: Mentat is available on the JPL/ESS iPSC/860 and will be ported to the Intel Paragon and to the Cray T3D when the tools are available. The evaluation showed that encapsulation was detrimental to performance. Work is ongoing with UVA to determine whether the compiler or runtime system can automatically break the encapsulation induced bottlenecks.

Point of Contact:

Robert Ferraro
Jet Propulsion Laboratory
ferraro@zion.jpl.nasa.gov
(818) 354-1340

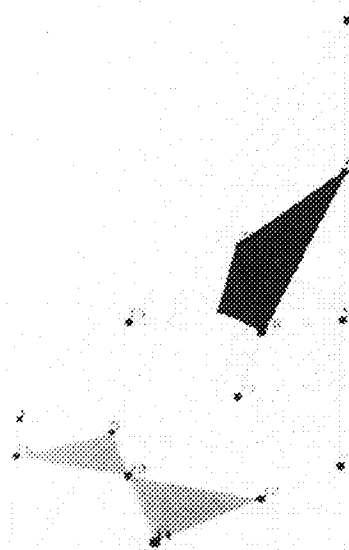
Parallel Mesh Partitioning & Refinement

Flow of Control for Mesh Partition and Refinement Code



Control Flow of the Parallel Mesh Partitioner / Refiner.

Intuitive Uniform Mesh Refinement algorithm subdivides building blocks of an object and deduces the new subobjects that they would form.



Assigning IDs to New Nodes After Mesh Refinement

Problem:

- Duplicate nodes are produced during the mesh refinement process
- Nodes for each processor have to be identified

Solution:

- The new processor for each node is determined by the processor that owns the parent node
- The new nodes are created and labeled by the processor that owns the parent node
- All processors get the list of new nodes and assign them to their own processor

Before Refinement



After Refinement



Duplicate Node Remover operates on duplicate nodes produced during independent refinement of objects by neighboring processors.

Parallel Unstructured Mesh Partitioner/Refiner

Objective: Provide a software tool for partitioning and refining 2-D and 3-D finite element and finite volume meshes on parallel computers.

Approach: A given coarse mesh describing a computational domain (e.g., for finite element/volume methods) is partitioned in parallel in N processors using a recursive partitioning algorithm and distributed among them. Using a uniform refinement algorithm, each processor refines its own portion of the mesh independently to produce a finer mesh. Elements and nodes generated at partition boundaries need to be examined and duplicate information (e.g., node IDs) needs to be resolved. Partitioning and refinement on the mesh can be performed multiple times to achieve the required resolution and load balance for the final mesh partition.

Accomplishments: A parallel mesh partitioner code and a sequential mesh refiner code have been implemented. The partition module and the refiner module have been integrated with the addition of another module that resolves duplicate information in nodes and assigns consistent global information to nodes and elements after mesh refinement. The initial evaluation of the performance of the code shows the refiner with duplicate information resolution scales reasonably well on an Intel Delta machine, while the parallel partitioner does not scale. The general control flow and the mesh refinement and duplicate resolution algorithms are illustrated in the figure opposite.

Significance: Using the parallel mesh partition and refinement code, a fine-resolution, load balanced unstructured computational mesh from a very coarse input mesh on a parallel computer can be generated. Generating large unstructured meshes directly on a parallel machine is necessary for parallel scientific computations since: 1) generating large unstructured meshes needed for a large scale computation is not practical on a workstation; 2) the transfer of a large unstructured mesh file from a sequential machine to a parallel machine is inefficient and costly, 3) the ability to generate and refine (locally and adaptively) meshes on parallel machines can be useful for implementing parallel multigrid/multilevel algorithms and adaptive algorithms.

Status/Plan: For the mesh partitioner, do further performance tests to pin down non-scalable parts of the code and improve the performance of those parts. For the mesh refiner, streamline existing code and introduce means for localized or non-uniform refinement of meshes. For the whole mesh partition/refinement code, improve interface consistency between different parts of the code and memory management so that each module of the program can be executed multiple times. Develop fully parallel mesh generator.

Point of Contact:

John Lou
Jet Propulsion Laboratory
lou@acadia.jpl.nasa.gov
(818) 354-4870

Koz Tembekjian
Jet Propulsion Laboratory
koz@yellowstone.jpl.nasa.gov
(818) 354-8403

INTEGRATED NETWORK COMPUTING ENVIRONMENT

FDDI

PARAGON

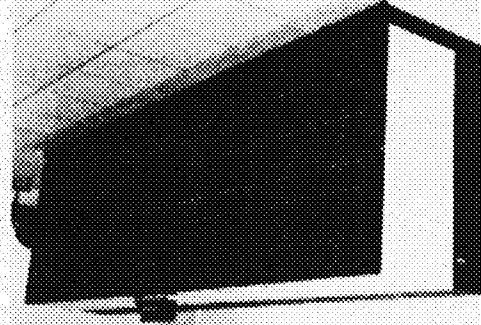
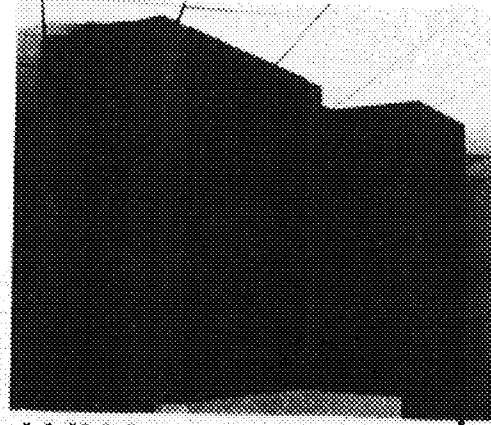
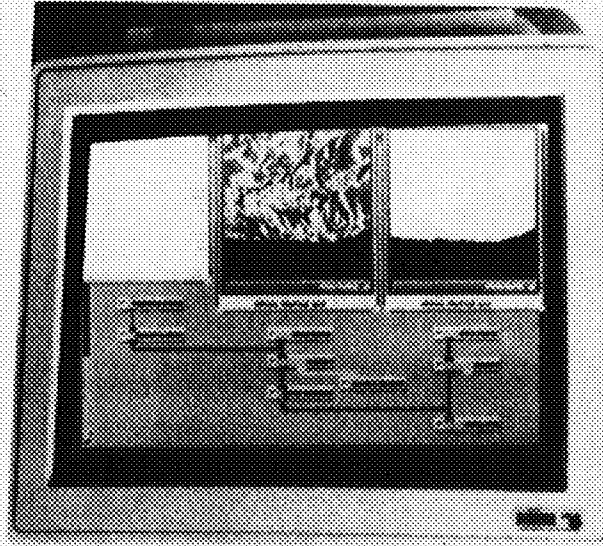
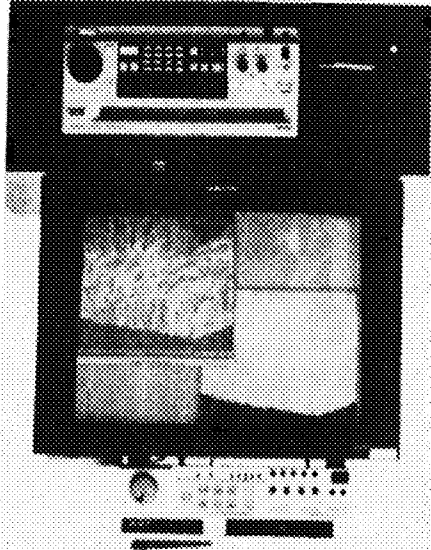
USER WORKSTATION

HIPPI

DELTA

CRAY

VIDEO



MIMD Systolic Dataflow Tools

Objective: Most parallel computing efforts have focused on the computational aspect of a specific task without providing an end-to-end solution. A computationally optimal solution of a task in a massively parallel architecture may not be an optimal solution of a global problem if the solution does not provide an efficient interface mechanism for data flow between tasks in a heterogeneous system environment. The objective of this work on MIMD systolic dataflow tools is to implement a set of tools for designing, developing, scheduling, and monitoring a large number of tasks in a heterogeneous distributed computing environment. This task emphasizes MIMD architecture-based parallel programming, network communication among heterogeneous systems, and multimedia data format and transformation.

Approach: The tool development and integration are implemented in multiple levels: user level, programmer level, and system level. For end users, the task develops an environment where a user can easily schedule, monitor, and control his/her complex tasks. For the programmers, the task develops object oriented parallel programming tools to shield a programmer from the architectural constraints. At the system level, the task integrates the state-of-the-art hardware and software systems that allow heterogeneous distributed computing, faster network access, graphical user interface, advanced visualization, and multimedia environment.

Accomplishments: Most FY93 activities were focused on system level tasks such as hardware system installation, including a Sparc 10 workstation, HiPPI, FDDI, a video archive system, and a remote video control system; and software system installation, including Khoros, PVM, and AVS. Two types of programmer tools, oimge (object oriented image class library) and ddlib (data distribution library), were developed for 2-D array manipulation and decomposition on MIMD architectures.

Significance: A heterogeneous distributed computing environment was designed and demonstrated by integrating remote HPCC sites, local workstations, and a video archive system using TCP/IP data communication protocol and the AVS visualization system. The environment enables a user to pipe computation results as computed from a remote super-

computing center to the user's workstation for post processing and visualization. A user may interactively archive the processed output in a video medium.

Status/Plans: The integrated end-to-end environment design effort was pursued with installation of faster networks to supercomputing centers, process communication procedures, and video archive facilities. The environment provides (a) HiPPI connection to JPL's supercomputing center, (b) FDDI connection to the internet, (c) a programmable laser disk recorder and player, (d) a real time scan conversion system, and (e) direct video link to JPL's audio visual center. A high level systolic dataflow design and heterogeneous distributed computing applications were implemented in the above environment using Khoros and AVS. The icon-based process flow design tools of Khoros (Cantata) have been integrated with the planetary data processing applications for interactive task process and data flow design. By the end of the next fiscal year, a full implementation will include graphical process flow design, automatic process scheduling, and interactive process and resource monitoring.

Point of Contact:

Meemong Lee
Jet Propulsion Laboratory
meemong@elroy.jpl.nasa.gov
(818) 354-2228

Better View Selection

Renderers for Host: sun10

HiResRender (X-Window, threads, render.)

Renderers (For thread: render) (parallel render) (active: 0)

Change Host [?] OK Cancel

Better View Options

It Has Renderer:

Exploration Display: sun11:10.0

Exploration Render Host: sun10

Output X dim: 400 Y dim: 400

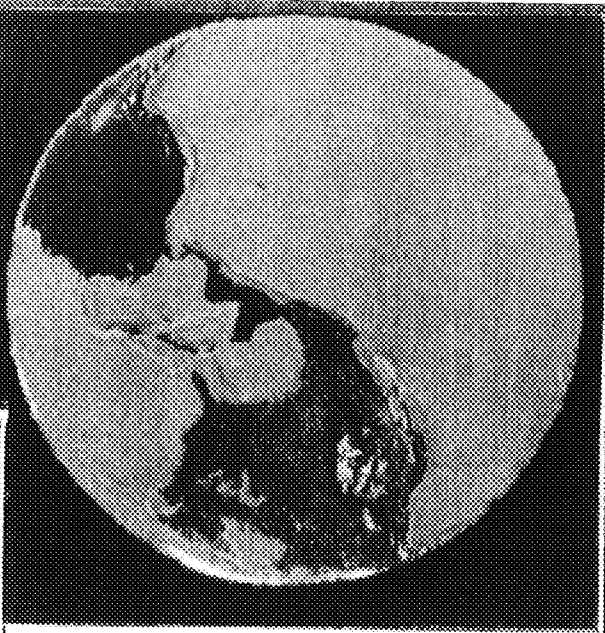
Aspect Ratio: 1 Quality: 1

Gamma Correction: 1

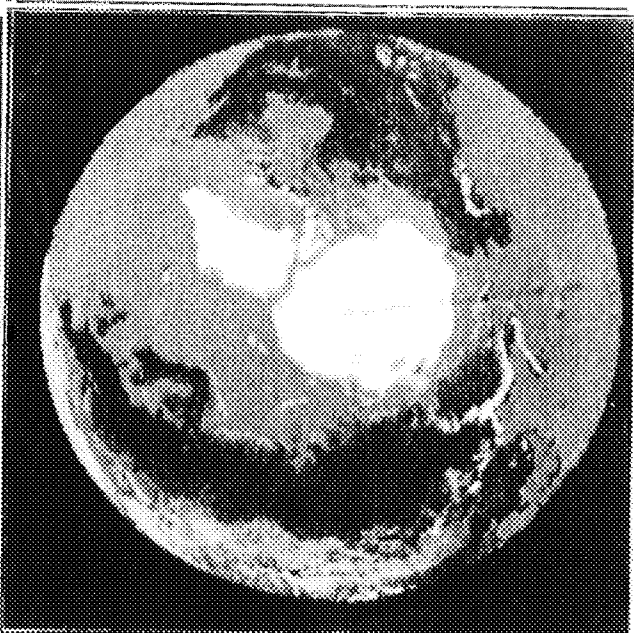
Background Color:

Rad: 138 Green: 138 Blue: 155

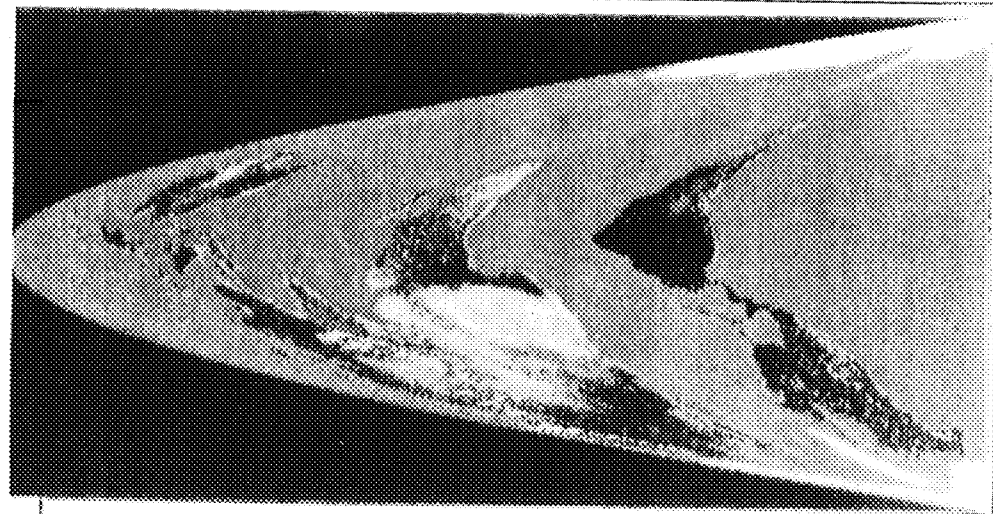
Apply Kill Select Renderer... Done



Better View - sun5



Quick View - sun18



Texture Top View

```

cndtool (CONSOLE) -/bin/esh
sun1:3 & cd -ll
sun1:4 & cd lsk
sun1:5 & import -root ~/SW.nit
  
```

Surveyor

File Edit View Explore (animate) (simulate) (?)

Quick View Quick View Options... Done

Better View Better View Options... Numerical Edit...

Look at Tag Visible Objects... Flag Info

Save to File... Rotate Speed: 10 Motion Speed: 30

Look Rotate Horizontal Flip

Home Revolve

Horizontal Flip (Degrees) 45

Implemented Parallel Database Rendering

Objective: Develop scalable, portable, massively parallel solutions for interactively computing 3-D renderings of image and terrain databases using at least 500 computing nodes on an "n core" database of five gigabytes.

Approach: The algorithmic approach is the "ray identification" method that decomposes the input database over the entire processor set. Initially an experimental tabletop renderer (i.e., the geometric model used is a flat tabletop with elevation departures from that datum) was constructed. Later, a more general geometric model, the "Whole Earth" model, that permits the elevation data to be measured from a spherical or other solid figure model was pursued.

Accomplishments: In FY93 the experimental tabletop renderer was refined and used to produce a three-minute demonstration film of Maui using a LandSat thematic mapper image and an associated digital terrain model from the USGS. The Whole Earth model was implemented and a new distributed output data structure was designed for scalability and to permit the creation of large output frames.

To make these developments more usable as a general purpose tool, the parallel renderer, which resides on the Intel Touchstone Delta, was interfaced with two separate graphics user interfaces (GUIs). The first GUI is the Surveyor software used for planning detailed flight paths and viewing direction strategies. The figure on the opposite page shows the Surveyor GUI, a sample earth dataset and two rendered images from different viewpoints. The second GUI adapted the Silicon Graphics Flight Simulator with its current position and attitude as the viewpoint to interact with the renderer. This later development incorporates the Delta's HiPPI interface and the gigabit network for the high speed transmission of the resulting renderings.

The Whole Earth model, its associated scalable output data structure algorithm, and the interfaces to Surveyor and the Flight Simulator GUIs are provisionally complete and are being debugged and tested. Quality renderings from both databases have been made. No performance data is yet available. Considerable refinement will be needed before interactive quality performance can be achieved.

Significance: Historically, quality rendering has been accomplished with slow software renderers. While hardware assisted workstation based renderers have been created with good interactive performance, they are limited in the size of the image base that can be considered at one time, and the hardware process introduces artifacts into the final images. With the introduction of scalable software renderers, interactive exploration of large databases will become possible.

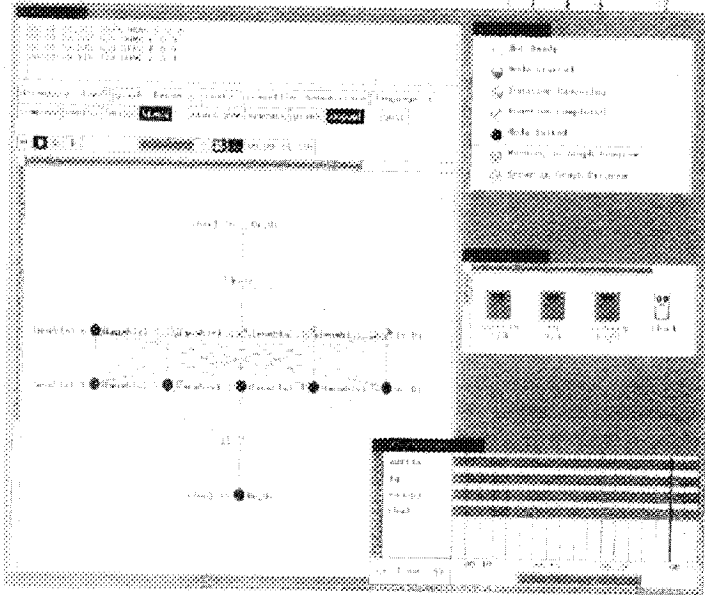
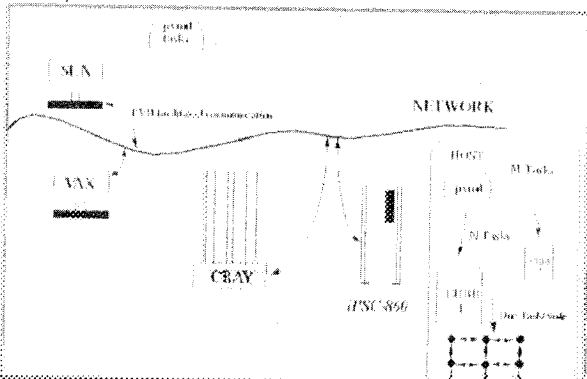
Status/Plans: Complete the debugging of the algorithm and the GUIs. Tune the Whole Earth algorithm for high performance. Introduce a new, high quality scan line raster conversion algorithm for the highest fidelity film productions.

Point of Contact:

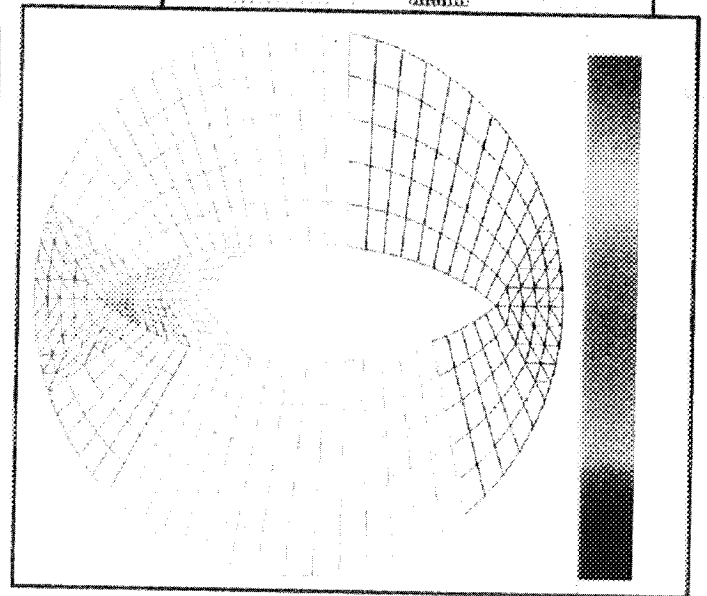
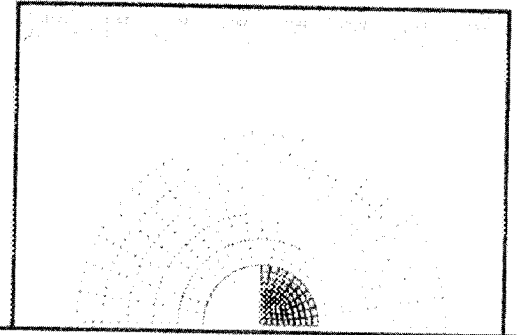
Peggy Li
Jet Propulsion Laboratory
peggy@sun11.jpl.nasa.gov
(818) 354-1341

JPL/ESS Testbed Installed Tools

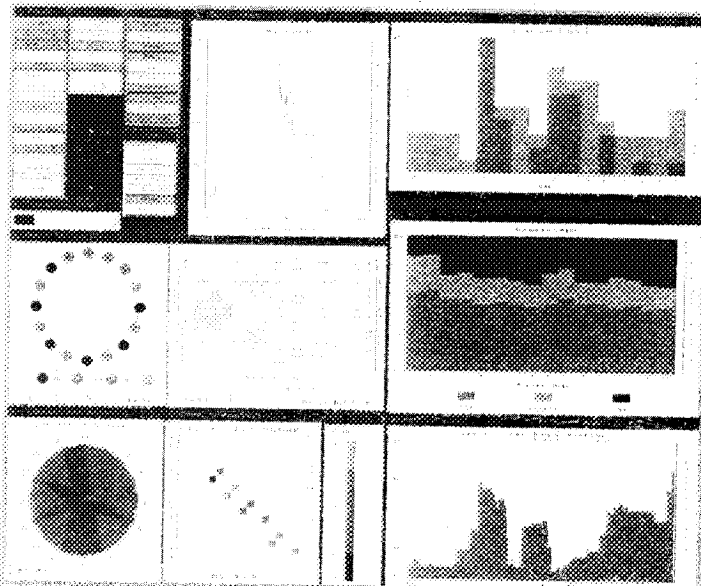
PVM/HeNCE



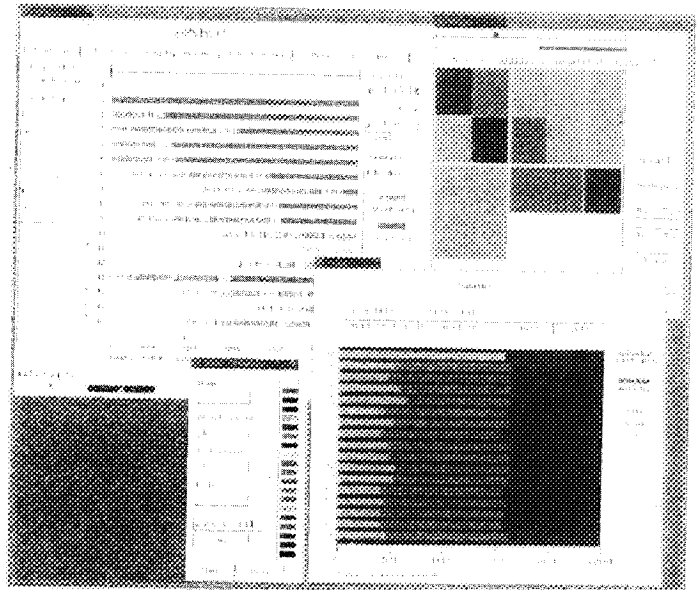
DGL



PICL/ParaGraph



P4/Upshot



Parallel Performance and Debugging Tools

Objective: Provide software tools for development, analysis and debugging of parallel programs.

Approach: The approach to accomplishing this objective is to download and install most well known parallel program development tools and write those that don't exist. Although several libraries exist, there doesn't seem to be a graphical debugger for the Intel iPSC-860 machine that JPL ESS project uses as its computing testbed.

Accomplishments: Some of the better known portable parallel program development environments are the Parallel Virtual Machine (PVM), the Portable Instrumented Communication Library (PICL), the Portable Programs for Parallel Processors (P4), and the Heterogeneous Network Computing Environment (HeNCE). Accomplishments in FY93 include

1. The libraries for portable parallel program development and performance analysis mentioned above have been downloaded and installed on the Intel machines and on the Sun network, where they can be used to develop parallel programs in heterogeneous and massively parallel processor computing environments.

2. Among performance evaluation tools, ParaGraph and Upshot are among the best known. ParaGraph is an X Windows based graphical tool that is used to analyze the trace files created by programs written using PICL tracing library calls, and Upshot is used for the same purpose, but is dependent on P4 tracing results. Both are installed on the Sun network.

3. The preliminary study was conducted to evaluate the possibility and the feasibility of writing or enhancing the Intel Parallel Debugger available on iPSC-860 machine. The study showed that an X Windows based graphical front end should be a relatively easy task. Such a front end, available in the public domain, was analyzed and it may be adaptable to IPD as well.

4. Distributed Graphics Library (DGL), a product from Silicon Graphics, Inc. adapted to run on Intel iPSC-860 nodes, was installed and used successfully in debugging the parallel

refiner work in progress within the group. Using DGL, SGI programs can be run on nodes and visualized on an SGI workstation. Sample sessions or results from each of the installed tools are illustrated on the opposite page.

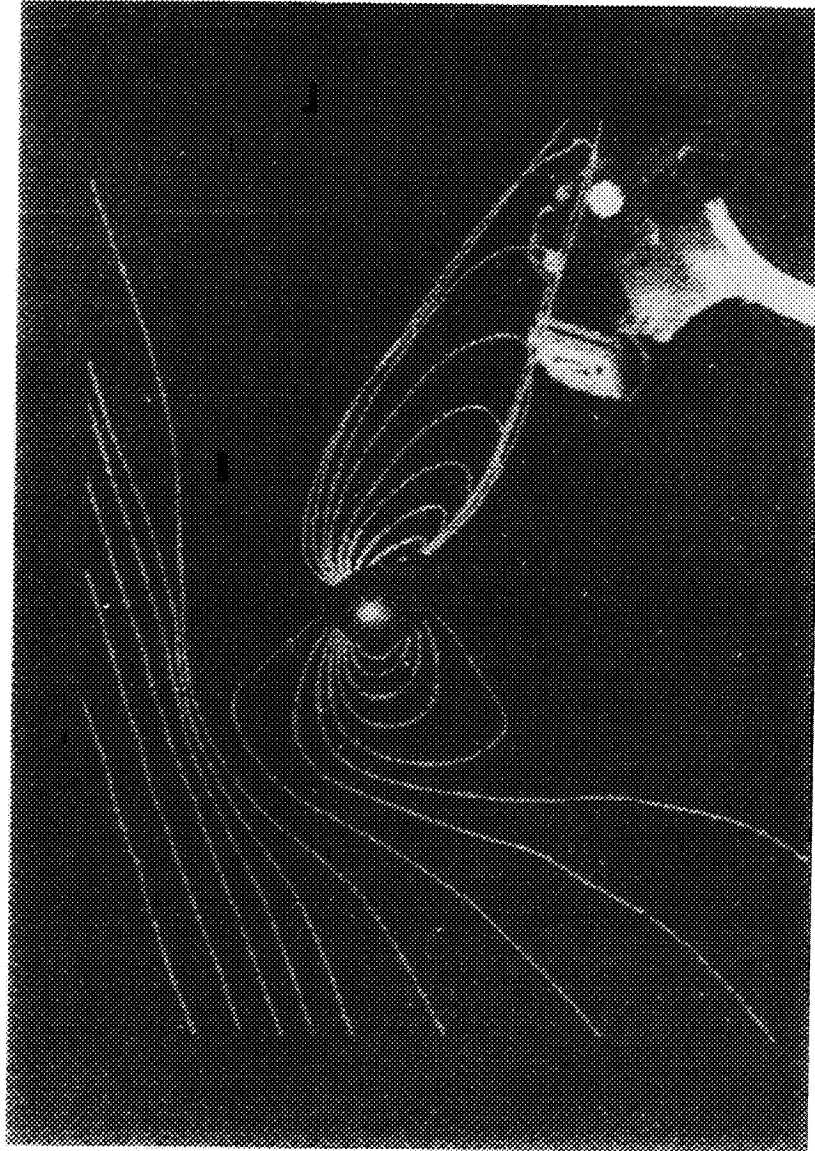
Significance: Installation of portable programming libraries and debugging and performance analysis tools provides the JPL ESS computing testbed user community with a user friendly and efficient environment for program development.

Status/Plans: More libraries and tools are being added and older versions are being updated.

Point of Contact:

Koz Tembekjian
Jet Propulsion Laboratory
koz@yellowstone.jpl.nasa.gov
(818) 354-8403

Ported the Visualization System
FAST to Virtual Reality



Ported the Visualization System FAST to Virtual Reality

Objective: Develop methods to analyze high rate/high volume data generated by ESS Grand Challenges.

Approach: The approach is straightforward and consists of two elements:

- ❑ Investigate turn-key virtual environments compatible with existing NASA science community visualization methods and software.
- ❑ Adapt the Flow Analysis Software Toolkit (FAST) developed and maintained by NAS/ARC to a virtual environment.

Accomplishments:

- ❑ Received a SGI Reality Engine Sky Writer (4-processor, 2-graphics subsystems), glove and helmet.
- ❑ Basic tracking and glove capabilities along with low resolution head mounted display have been integrated into FAST. Other capabilities added are spatial navigation, voice control, glove calibration and posture recognition, and basic object tracking and touching.
- ❑ Ford Motor Co., which adopted this approach, provided additional funds for software development.

Significance: NASA scientists increasingly sift through mountains of acquired and computationally generated data. The essence of virtual reality is dealing with the data in the same way that people deal with the actual world—through the visual cortex and motor responses, rather than through artificial interfaces. The creation of an operational virtual reality environment for rapid data searching and manipulation is required to validate the theory and transfer it to the NASA science community.

Status/Plans: All remaining modifications to FAST should be completed and the complete system delivered to GSFC in December 1993. ESS Principal Investigator Richard Rood and Co-Investigator Mark Schoeberl from the GSFC Laboratory for Atmospheres will provide initial application testing at GSFC involving exploration of atmospheric and meteorological

data. Dr. Steven Curtis from the ISTP/GGS Project at GSFC also will provide initial application testing of a "virtual magnetosphere" application. Early in FY94 a development platform will be placed at Sterling Software to support continued improvement of responsiveness and intuitive features.

Point of Contact:

Horace Mitchell
Goddard Space Flight Center
hmitchel@vlasov.gsfc.nasa.gov
(301) 286-4030

(THIS PAGE INTENTIONALLY BLANK)

Introduction to Basic Research and Human Resources (BRHR) Project

Goal and Objectives: The goal of the BRHR project is to support the long-term objectives of the HPCC Program by making substantial basic research contributions and by providing individuals fully trained to utilize HPCC technology in their professional careers. This goal is pursued through basic research support targeted at NASA's Computational Aerosciences Project and the Earth and Space Sciences Project.

Strategy and Approach: BRHR promotes long-term research in computer science and engineering, while increasing the pool of trained personnel in a variety of scientific disciplines. The BRHR project encompasses a diversity of research activities at NASA centers and US universities and spans a broad educational spectrum: kindergarten through secondary education (K-12); graduate student research opportunities; post-doctoral study programs; and basic research conducted by experience professionals. This project encourages diverse approaches and technologies, with a focus on software technology and algorithms, and leverages the ongoing NASA research base.

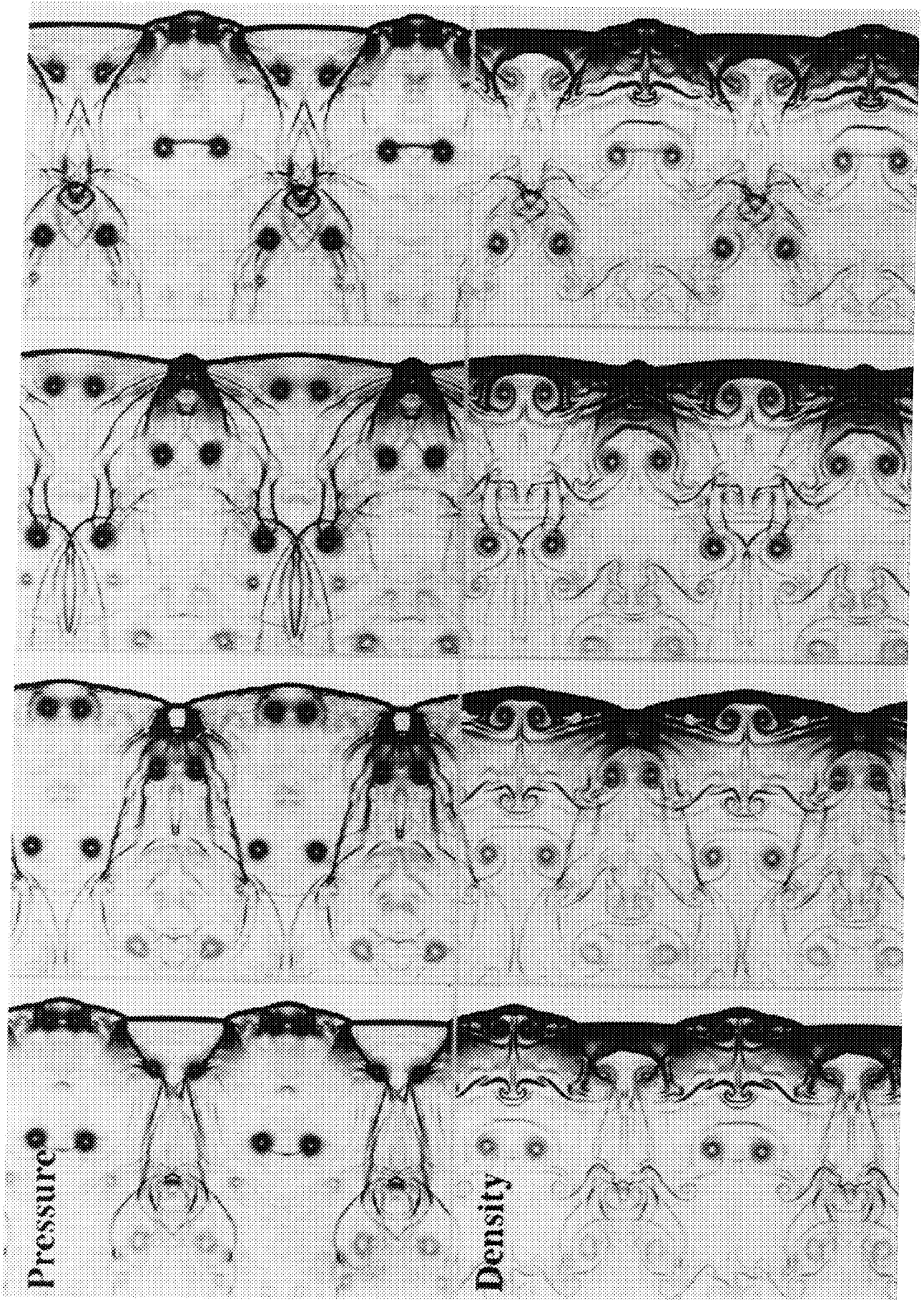
In FY93, the BRHR project continued programs for graduate and post doctoral student research and K-12 projects at all the NASA research centers involved in the HPCC Program. These efforts are being expanded as they are further integrated into the Computational Aerosciences and Earth and Space Science projects. In addition, BRHR produces fundamental research results from research funded at NASA centers, NASA supported research institutes, and universities.

Organization: NASA Headquarters serves as the lead for the BRHR project with support from the following NASA Centers: Ames Research Center (ARC); Langley Research Center (LaRC); Goddard Space Flight Center (GSFC); Marshall Space Flight Center (MSFC); Lewis Research Center (LeRC); and, the Jet Propulsion Laboratory (JPL). BRHR has research projects at the following NASA supported research institutes: the Institute for Computer Applications in Science and Engineering (ICASE) at LaRC, the Research Institute for Advanced Computer Science (RIACS) at ARC and the Center of Excellence in Space Data and Information Sciences (CESDIS) at GSFC.

Management Plan: The project is managed by the BRHR Project Manager at NASA Headquarters who coordinates developments with NASA centers, the HPCC projects, and other federal agencies and departments participating in the national HPCC program.

Point of Contact:

Paul Hunter
NASA Headquarters
Washington, DC
P_Hunter@aeromail.hq.nasa.gov
(202) 358-4618



Adaptive Mesh Refinement Algorithms on Parallel Computers

Objective: Develop portable parallel Adaptive Mesh Refinement (AMR) algorithms for the simulation of reacting and nonreacting flows.

Approach: Adaptive Mesh Refinement algorithms that dynamically match the local resolution of a computational grid to the numerical solution being sought have emerged as powerful tools for solving problems with disparate length and time scales. Several researchers have demonstrated the effectiveness of using an adaptive, block-structured hierarchical grid system for simulations of complex shock wave phenomena.

A natural parallelism can be observed by considering each grid block as a single entity for which the governing equations are solved simultaneously for each adaptation level. However, the interaction between blocks of the dynamically changing block-structure encountered during each integration step makes the parallel implementation of such algorithms a nontrivial exercise. By maintaining a global description of the changing block layout on each processor (a small memory penalty even for large problems) one can greatly reduce the communication needs of the algorithm in terms of non-flow data, and the exchange of flow data can be scheduled efficiently.

Accomplishments: The parallel implementation has been built upon a standard message passing methodology and thus can be ported across a broad class of MIMD machines. Moreover, the method of parallelization is such that the original serial code is left virtually intact, and so one is left with just a single product to support.

Although the parallel version currently lacks some of the advanced features of the serial version, it is sufficiently mature that it can be used routinely to perform very large scale simulations of detonation phenomena using either the Intel iPSC/860 or workstation clusters using various message passing libraries. Simulating a detonation flow through a straight duct resulted in over 97 percent efficiency on 32 nodes of the iPSC/860 and over 80 percent efficiency on workstation clusters; even though communication among the clusters was via Ethernet as shown on the opposite page. An example of such a simulation is also shown which depicts the transverse wave structure of

the detonation front as a sequence of four Schlieren-type snapshots. These results show that the parallel algorithm has progressed beyond being an exercise in computer science to being a powerful research tool for investigating fluid phenomena.

Significance: Large scale numerical simulations are required to further the understanding of detonation dynamics, an understanding that will be essential to the design of an Oblique Detonation Wave Engine. Sophisticated parallel mesh refinement algorithms enable simulations which otherwise would be prohibitively expensive.

Status/Plans: Further work is planned to improve the functionality of the parallel AMR algorithm. In particular, the problem of load balancing will be addressed in the near future. Also, the true portability of the parallel algorithm will be demonstrated by implementing it on other MIMD platforms (e.g., CM-5, Paragon, various workstation clusters).

Point of Contact:

James J. Quirk
ICASE
NASA Langley Research Center
jjq@icase.edu
(804) 864-2190

Ulf R. Hanebutte
ICASE
NASA Langley Research Center
ulf@icase.edu
(804) 864-8006

(THIS PAGE LEFT BLANK INTENTIONALLY)

Advanced Methods Development

Objective: Develop advanced numerical methods in support of NASA Ames Research Center (ARC) missions focusing on

1. Algorithmic capabilities not available in the past
2. The efficient use of parallel computers

Approach: A two-pronged approach is being used :

1. Algorithmic needs or bottlenecks are identified and researchers identified to address them
2. Significant new algorithms are identified, tested and modified for important ARC tasks

Accomplishments:

- ❑ A dynamic mesh adaptation scheme, 3D-TAG, has been developed by Biswas with Strawn for unstructured three dimensional grids.
- ❑ Olsson has developed a technique for constructing strictly stable methods for linear initial boundary value problems.
- ❑ Nachtigal and Freund have developed quasi-minimal residual (QMR) iterative methods for nonsymmetric systems that have been implemented and installed in the NetLib library and successfully tested on systems arising from implicit Navier-Stokes solvers

Significance: The 3D-TAG scheme has been successfully used for significant helicopter rotor simulations. Until now, no general technique for ensuring that numerical methods will have the same stability properties as the differential equations being modeled has existed. The QMR method is a stable, efficient solution procedure that does not require the storage of other methods for nonsymmetric systems.

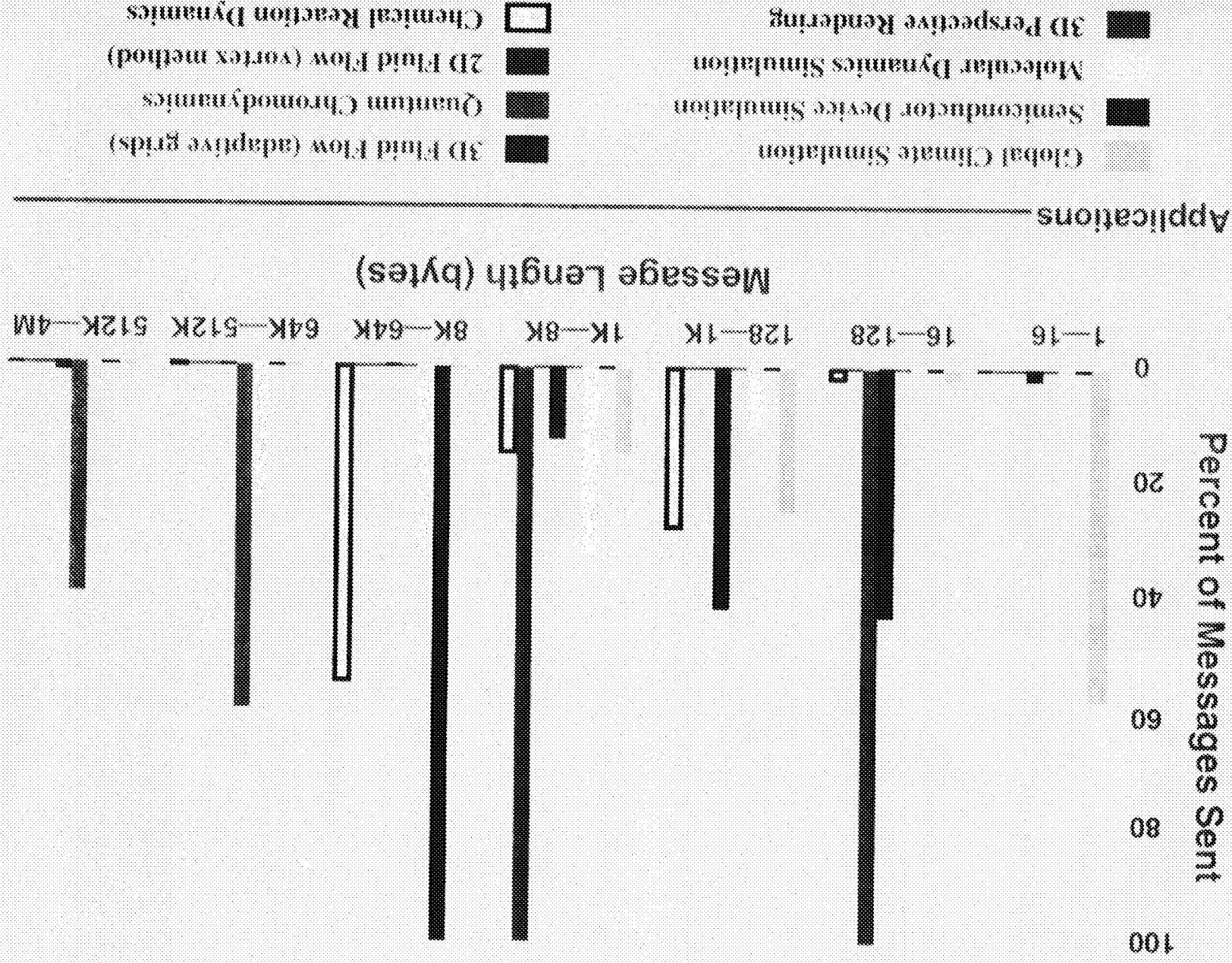
Status/Plans: A new version of 3D-TAG is being developed for efficient execution on the CM-5. Olsson's theory for the initial boundary value problem is being extended to nonlinear problems, and parallel implementation of these methods is under way.

Although the QMR project has been completed, a new class of nonlinear iterative methods useful in optimization and control has been identified, and research is underway to apply these methods to problems in computational aeronautics.

Point of Contact:

Joseph Oliger
RIACS
NASA Ames Research Center
oliger@riacs.edu
(415) 604-4992

HPC Applications Differ Widely in Size of Messages Sent



Applications Data Movement on MPPs

Objectives: Gain an understanding of architectural features of distributed memory MPP hardware and software that will support effectively the communications needs of Grand Challenge applications.

Approach: Study the communications and I/O behavior of large-scale engineering and scientific applications programs used on massively parallel computer systems, including the Intel Delta, and nCUBE 512 and 1024 node systems. Measure the size and frequency of messages of the applications and other data movement quantities.

Accomplishments: Eight applications were studied. The floating-point, memory, I/O, and communications requirements of these highly parallel scientific applications were quantified. For three of the applications, analytical models were developed for the effect of varying problem size and degrees of parallelism. A paper describing these results was submitted to a highly competitive conference, the International Symposium on Computer Architecture, held in May 1993. The paper was one of 33 papers accepted from 206 submitted: further, the conference program committee designated it a "high-impact" paper.

Significance: The knowledge gained through this research project will help guide the design of future massively parallel computer systems. For example, the message sizes used by different applications varied widely. The implication of the message size data is clear: communications networks must be able to perform well with either very short or very long messages as well as a mixture of both. Another insight gained is that the applications studied performed relatively few broadcasts and barrier synchronizations. As a result, special hardware support for these operations is not justified (based on existing data). However, hardware support for global broadcast would be useful for initial loading of programs into memory.

Status/Plans: This line of investigation will continue with a study of parameterized communications routines designed to be efficient on a variety of distributed memory MIMD architectures. The routines will be installed on the Intel Delta system at Caltech and other systems, used with applications programs, and their performance will be measured. In addition,

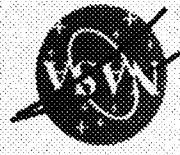
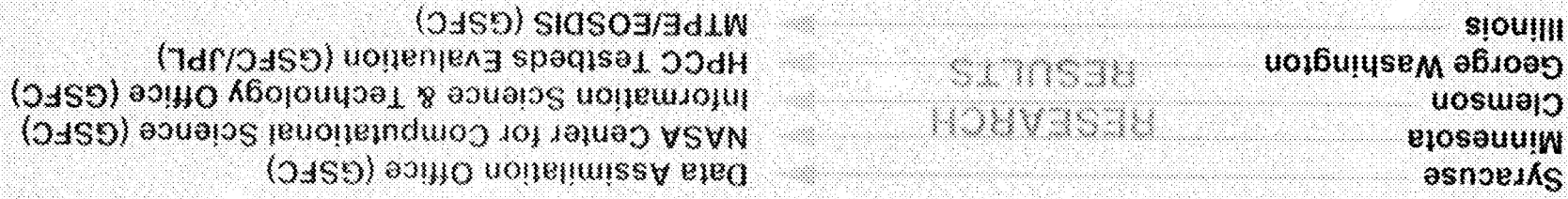
the insights gained through this research will be used to guide some of the activities of the Scalable I/O initiative that Caltech is undertaking in collaboration with a number of other investigators.

Point of Contact:

Paul Messina
California Institute of Technology
messina@ccsf.caltech.edu
(818) 395-3907

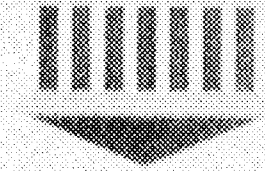
CESDIS University Research Program in Parallel Computing

Topic: Parallel Input/Output Systems

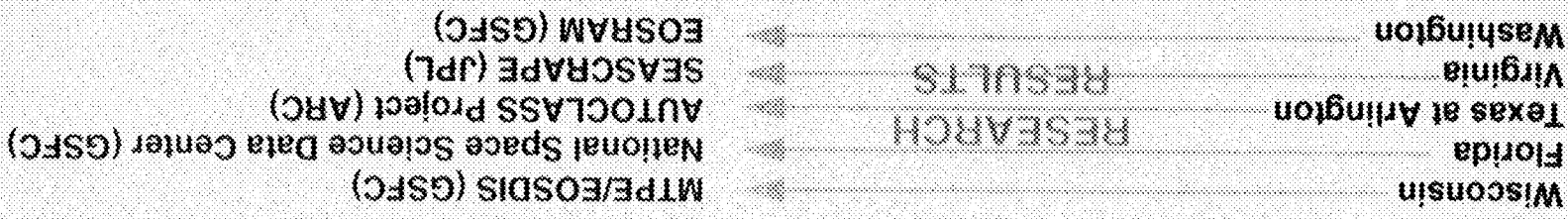


NASA Beneficiaries

University Teams



Topic: Parallel Database & Data Management



CESDIS University Research Program in Parallel Computing

Objective: Initiate a basic research program at selected U.S. universities focusing on two key areas of major importance to the NASA High Performance Computing and Communications Program: (1) parallel input/output systems and (2) parallel database and data management systems.

Point of Contact:

Terrence W. Pratt
Goddard Space Flight Center/Code 930
pratt@cesdis1.gsfc.nasa.gov
(301) 286-4108

Approach: Issue a Request for Proposal (RFP) to the computer science community. Use pre-proposals to peer review and select a subset from whom to request full proposals. Peer review and select winners. Attend to NASA collaborations with the selected university groups so that benefits to NASA from the research are as immediate as possible.

Accomplishments:

- Issued the RFP in December 1992. In January 105 pre-proposals were received. Peer review selected 30 groups to submit full proposals. Twenty-nine proposals were received in March. Peer review selected the top twelve. Ten awards were announced in May to the following universities: Syracuse, Illinois, George Washington, Clemson, Minnesota, Wisconsin, Florida, Virginia, Washington, and Texas (at Arlington).
- Awards are for three years at \$50,000 per year.
- Proposals were excellent, despite modest award size.

Significance: The research offers potential improvements in the input/output and data management systems of massively parallel computer systems used for Earth and space science applications.

Status/Plans: Close attention will be paid to nurturing the collaborations between university and NASA groups. An anonymous FTP site has been established. NASA source codes and data sets are being collected; they will be made available over the Internet to the research groups as examples of codes requiring parallel input/output and parallel database facilities.

Third NASA Summer School in High Performance Computational Physics

Summer School Goal

Train the next generation of physicists in massively parallel techniques and algorithm development to support the High Performance Computing and Communications Program (HPCC).

Instruction

Lectures were provided by six members of the ESS Inhouse Team:

- S. Zalesak/GSFC
- C. Moberly/GSFC
- A. Deane/USRA
- B. Fryxell/USRA
- P. MacNeice/HSTX
- K. Olson/USRA

The summer school took place in 3 weeks of intensive lectures and lab sessions.

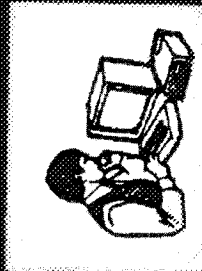
Techniques & Methods

- Finite Volume Methods
- Shock Capturing Methods
- Generalized Finite Volume and Finite Element Methods
- Spectral and Spectral-Element Methods
- Particle Methods



Students

Sixteen Doctoral Candidates selected through an USRA solicitation.

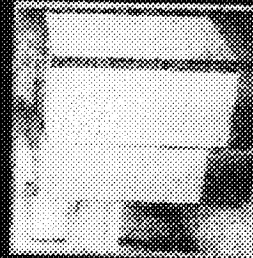
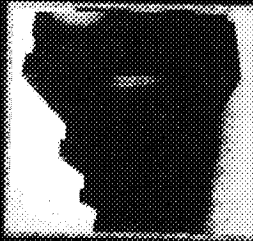


Annual HPCC/ESS activity to train members of ESS Investigation Teams.

Massively Parallel Testbeds

Connection Machine-5

MasPar MP-1



Lab sessions in code development given by the vendors:

- MasPar Computer Corporation
- Thinking Machines Corporation

Third NASA Summer School in High Performance Computational Physics

Objective: Train the next generation of physicists in massively parallel techniques and algorithm development to support the goals of the HPCC Program.

Approach: The NASA Summer School for High Performance Computational Physics, which is jointly sponsored by the ESS Project and the GSFC Space Data and Computing Division, is an intensive three-week program of lectures and lab sessions at the Goddard Space Flight Center (GSFC). Sixteen students, who must be working on PhDs in a physical science and must be interested in using massively parallel computer architectures, are selected by a national solicitation and housed at the University of Maryland. During the session the students have access to scalable computer systems. Vendors of the selected systems (MasPar Computer Corporation and Thinking Machines Corporation in FY93) provide lectures and hands-on workshops on code development for their products.

Accomplishments: Members of the GSFC/ESS In-house Team were the primary instructors for the FY93 session (S. Zalesak and C. Mobarry of GSFC, A. Deane, B. Fryxell, and K. Olson of USRA, and P. MacNeice of HSTX). Their lectures focused on advanced techniques in computational science, with special emphasis on computational fluid dynamics and on algorithms for scalable parallel computer architectures. The summer school achieved its objective as indicated below:

- Students became functional in the art of "thinking parallel"
- They became knowledgeable in advanced computational techniques
- They became captivated by the power of emerging scalable parallel systems, and most students requested continued access to GSFC's parallel testbed computers

Significance: The three sessions conducted to date have contributed significantly to the knowledge of ESS Project personnel concerning the formal training requirements for intelligent-but-novice users of scalable parallel systems. Building on lessons learned, the in-

structors and vendors continue developing an updated, suitable curriculum to meet these needs. Book publication of the lecture series is being considered.

Status/Plans: The ESS Project and GSFC plan to continue the school in FY94 and are considering expanding it to provide training for members of the P. I. investigation teams.

Point of Contact:

Dan Spicer
Goddard Space Flight Center
spicer@vlasov.gsfc.nasa.gov
(301) 286-7334

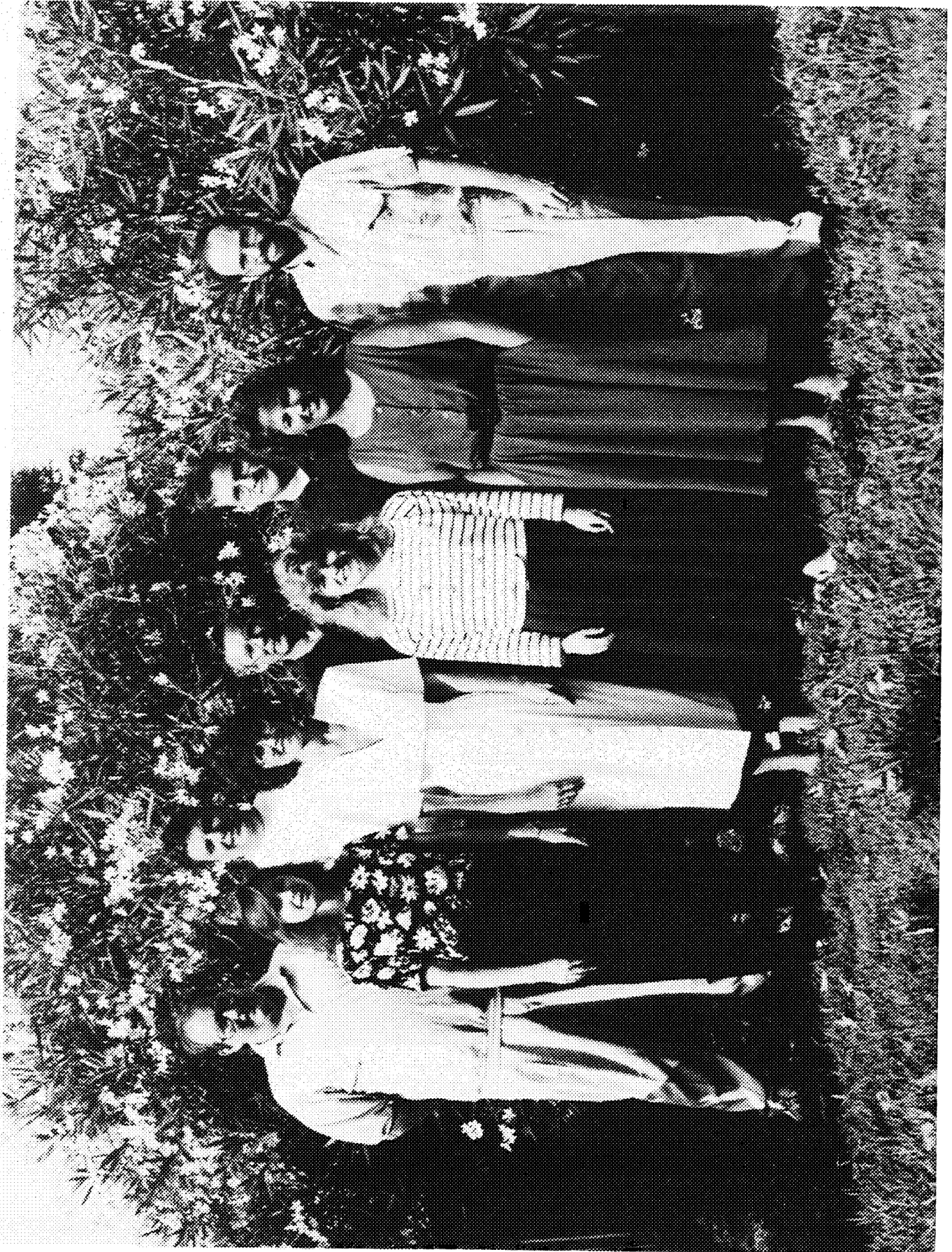


Photo: Pasadena Press
© 1965 Pasadena Press

NASA

Langley Research Center K-12 Program

Objective: To enhance the science, mathematics and engineering K-12 curricula and enable K-12 teachers and students to access the Internet by

1. Maximizing the available resources by using volunteer researchers to train the teachers to use workstations, software, and networking
2. Reaching as many local schools, and therefore students, as the resources allow

Approach: In its first year, the program was designed to train either mathematics or science teachers from local school systems in the use of workstations for science and engineering. Five teachers were selected from the school systems in the cities of Hampton, Newport News, Gloucester, and Williamsburg, Virginia, and the magnet school New Horizons. The teachers and NASA researchers conducted a series of meetings to prepare a summer training program tailored to the needs and objectives of the teachers and to identify the best computing and communication hardware to accomplish their objectives in the classroom. The meetings quickly revealed that the needs, objectives, and backgrounds of the teachers—and their support at the schools—varied considerably. Therefore, the summer training program was wide in scope to provide a large base of opportunity for all the teachers. The training lasted four weeks and included the fundamentals of using a scientific workstation; the UNIX operating system; scientific languages, such as FORTRAN, C, and MATHEMATICA; software designed for the classroom (e.g., Interactive Physics); and software, such as LabView, that connects a workstation to measuring probes. In addition, the teachers were trained to use the Internet, maintain computer security, and observe computer ethics.

Accomplishments: Comments from the teachers and the researchers who provided the training indicated a strong sense of accomplishment. The teachers developed new curriculum material for their classrooms. From resources available in FY93, each teacher was provided a workstation and telecommunication capability. Also, 1.5 classrooms were completely outfitted. The remaining 3.5 classrooms will be outfitted using FY94 funding. The five teachers are using the Internet to communicate

with each other and with NASA researchers, and will serve as mentors in their schools for other teachers.

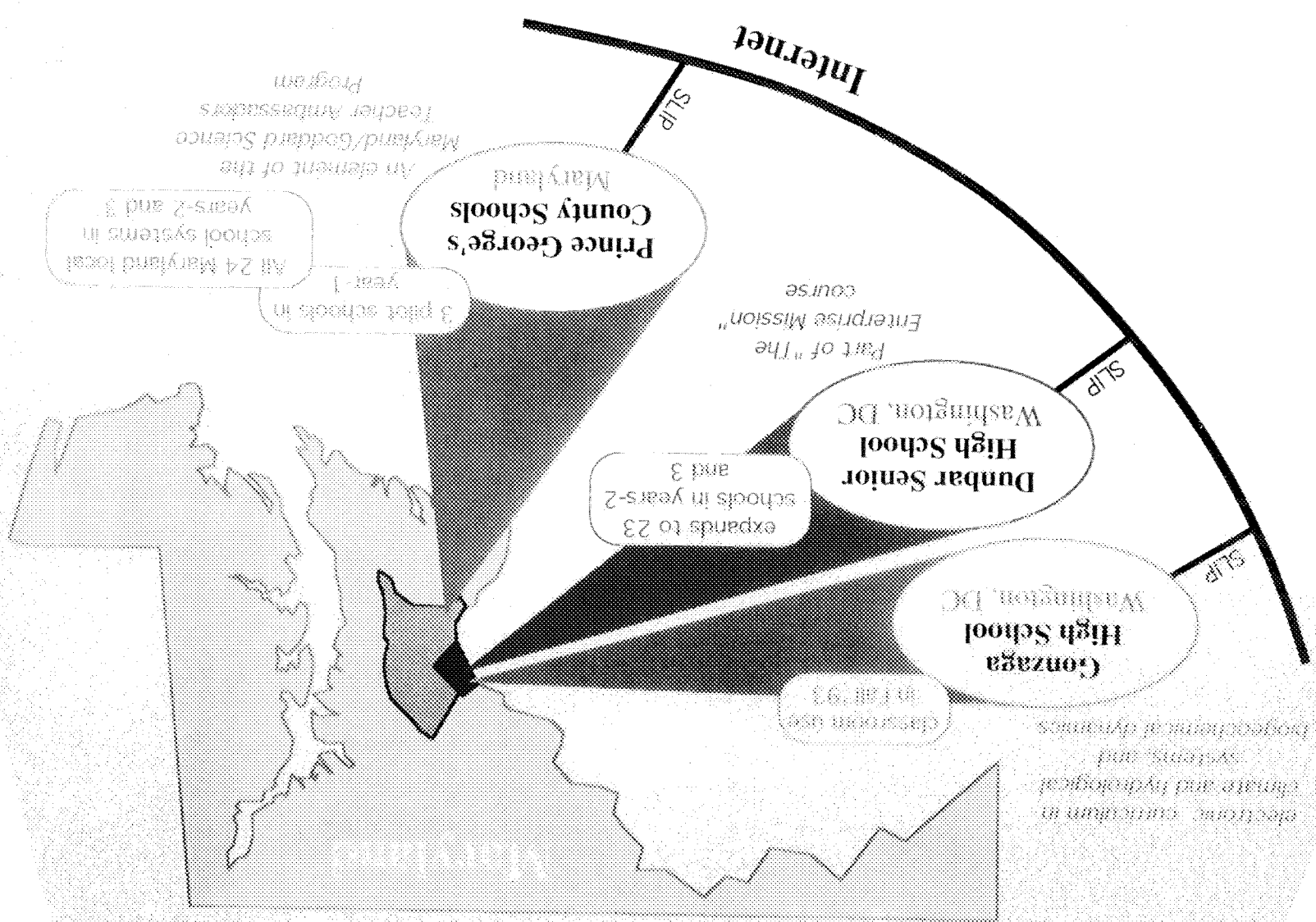
Significance: The new curricula should provide students with better skills in mathematics, science, and engineering. Lessons learned from this group of teachers will be used in future training programs to improve the program.

Status/Plans: Funding from FY94 will be used to complete outfitting the remaining 3.5 classrooms, and train as many as nine more teachers. With the aid of the current teachers, the new teachers will be selected in the second quarter of FY94. The first five teachers will be monitored throughout the year to determine how they use their new capabilities. Their problems and successes will be assessed and the results used to enhance future training programs.

Point of Contact:

Gary P. Warren
NASA Langley Research Center
g.p.warren@larc.nasa.gov
(804) 864-2162

Established ESS/GSFC K-12 Program



ESS/GSFC K-12 Program

Objective: The objective of the ESS/GSFC K-12 program is to facilitate rapid exchange of educational ideas and materials among K-12 teachers through use of the revolution in electronic communications afforded by the Internet.

Approach: The approach involves providing means for teachers to learn about and use the Internet as a teaching resource. It leverages the existing High Performance Computing Software Exchange as the means to provide the Internet accessible on-line repositories by

- Giving teachers access to the Internet
- Providing on-line repositories accessible through the Internet that permit teachers to deposit and withdraw resources (e.g., lesson materials)
- Stimulating development of educational resources that they can be electronically exchanged

Accomplishments: In April a formal proposal containing four components was submitted to NASA Headquarters. Each component will span three years. Headquarters selected three components for award in FY93. Two Space Act awards were approved: one to Prince Georges County, Maryland public schools, and the other to Dunbar High School, a public school in Washington, DC. The award to Prince Georges County public schools is the first phase of the Maryland/Goddard Science Teacher Ambassadors Program. The award to Dunbar High School is now part of the school's space-oriented, interdisciplinary credit course known as "The Enterprise Mission."

A purchase order award was made to Gonzaga High School (a private school in DC) in July. Its purpose is to produce a complete interdisciplinary curriculum focusing on the two areas of highest science priority in the U.S. Global Change Research Program: climate and hydrological systems, and biogeochemical dynamics.

Significance: Teachers and students are empowered, and because the program is scalable (i.e., it produces an activity or product that can be inexpensively replicated, a large number of students eventually can benefit) it deals gracefully with obsolescence of equipment and software. And it provides

qualitative improvement—"We couldn't do this before." The program has low overhead costs, (e.g., for user support) and it provides benefits to students studying fields other than science (all grade levels and subjects). Finally, it enables direct transfer of NASA HPCC technology to the public schools, including a focus on an inner-city public school.

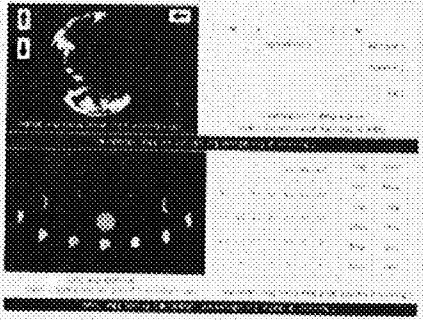
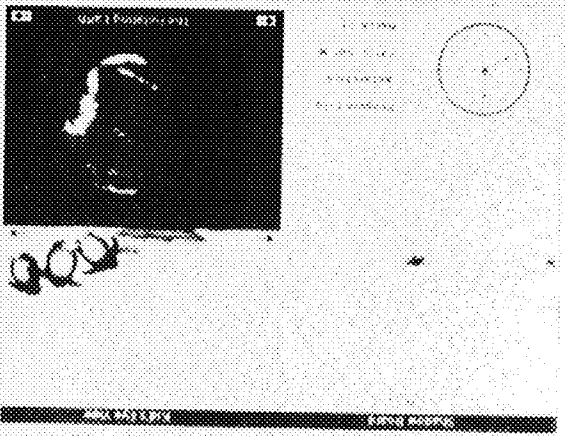
Status/Plans: In Prince Georges County public schools in year-1, this proposal establishes, monitors, and evaluates three pilot sites in the Maryland school system. In the second and third years, it will expand to each of the 24 local school systems in the state of Maryland. Plans for succeeding years at Dunbar High School include expanding this activity to the 23 schools (Cluster 4) supervised by the Dunbar principal. Beginning in early FY94 at Gonzaga High School, curriculum developed this summer will be used by students in collaboration with Charles Goodrich at the University of Maryland Advanced Visualization Laboratory.

Point of Contact:

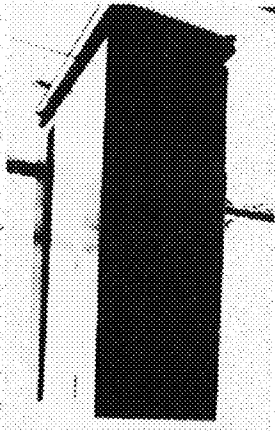
Jim Fischer
Goddard Space Flight Center/Code 934
fischer@nibbles.gsfc.nasa.gov
(301) 286-3465

JPL HPCC K-12

Project SEED Software Development
Daytime Astronomy Simulation

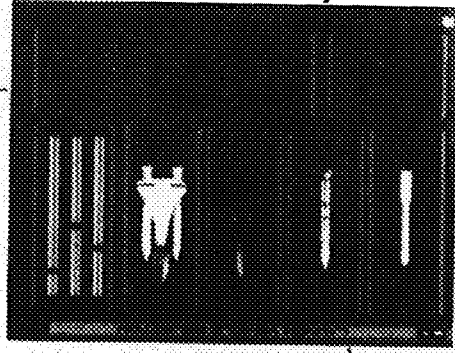
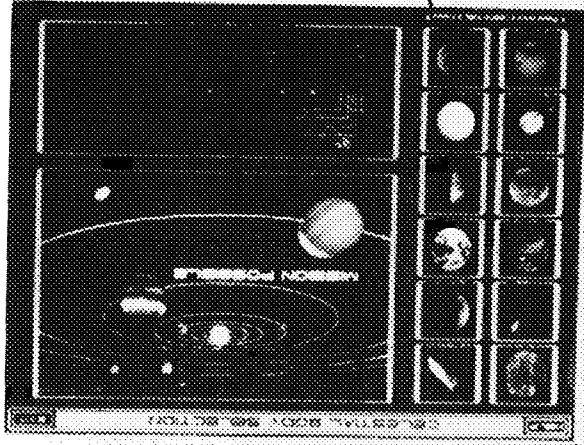


ESS Tested



Communication and Data Manipulation Utilities
in support of
Windows on Global Change

JPL Teaching Resource Center
"Mission Possible" Simulation



ESS/JPL HPCC K-12 Education Program

Objective. The ESSJPL HPCC K-12 Education Program has five objectives:

1. Enhance JPL's ability to provide teachers and students with tools for learning about planetary exploration and earth science
2. Develop repositories and catalogs of curriculum materials
3. Enhance elementary science education through development of both curriculum-specific and generic software
4. Develop communication utilities for transferring data (both imagery and tabular) and for facilitating interaction among educational institutions (e.g., scientist-to-teacher, classroom-to-classroom)
5. Develop high performance imagery data manipulation tools for datasets to be sent to classrooms.

Approach. The K-12 program is currently focusing on three projects with coordination and exchange among each. First is the JPL Public Education Office (PEO), Teaching Resource Center (TRC) *Mission Possible* Project for developing a planetary exploration simulation system and resource materials. The TRC also is developing an on-line repository for curriculum materials. The second project is a software development effort to complement the existing hands-on, kit-based Science for Early Education Development program (Project SEED). The third is the development of communications tools, a database of on-line imagery and tabular data, and data manipulations utilities. These utilities are designed to support the Windows on Global Change (WoGC) Project and the other two K-12 projects. These utilities use the JPL HPCC Earth and Space Science (ESS) Paragon Testbed and dedicated HPCC K-12 storage devices.

Accomplishments. During FY93 each of the JPL K-12 projects completed a conceptual design of the software and tools to be developed. In addition, the TRC completed the framework for the *Mission Possible* planetary simulation and began implementation of the simulation modules. The TRC also began the curriculum repository. Project SEED, with the advice of teacher consultants, developed soft-

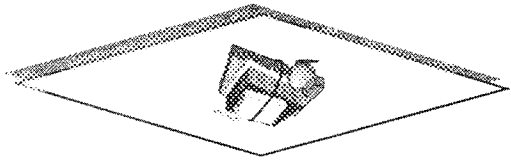
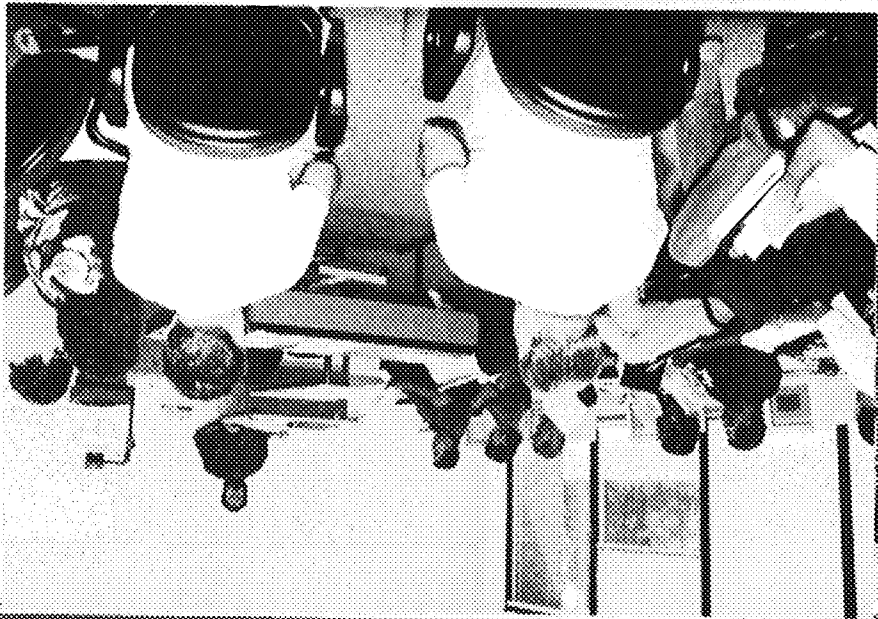
ware to aid in the instruction of the 5th Grade Daytime Astronomy unit. This software will be piloted in early FY94. The communications support for WoGC initiated the development of an imagery and tabular data database as well as setting up mechanisms for data transfers to and from the ESS Paragon Testbed system.

Significance. NASA and JPL have much to offer in outreach to the educational community from the early educational years through post graduate studies. Fundamental to the educational goal is the ability to develop and refine problem solving and analytic skills. The NASA missions provide an exciting and technologically advanced (e.g., high performance computing) setting for learning about our ourselves, our planet, and the universe.

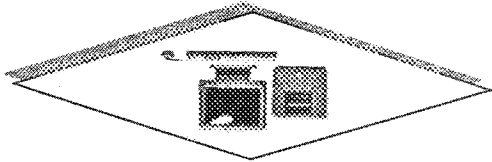
Status/Plans. Each of the JPL HPCC K-12 projects anticipates piloting software and imagery communications tools with students and teachers early in FY94. A teachers' in-service workshop is being planned for early FY94. Each of the projects will continue the consultation with teachers, software development, and the piloting and training cycle that was initiated during FY93.

Point of Contact:

Jean Patterson
Jet Propulsion Laboratory
jep@yosemite.jpl.nasa.gov
(818) 354-8332



K-12
Education+HPCC



Lewis Research Center K-12 Program

Objective: Introduce HPC technology into the K-12 school systems for the benefit of mathematics, science and engineering studies.

Approach: The LeRC K-12 program has leveraged the experience of other similar programs. Early involvement of teachers has helped guide the program. Also, existing LeRC training facilities have been leveraged. As a result, funds available have been spent on the schools.

Accomplishments: A librarian and twelve Cleveland-area educators in physics, mathematics, biology, and chemistry attended a two-week training session at LeRC. The topics included computing, networking, programming languages, hardware, science and technology at LeRC and security. The teachers had hands-on training with several computing systems (e.g., Macintosh, workstations). Two teachers were hired for the summer to address networking and assist in the training.

Four schools in the Cleveland area are involved: Garrett Morgan School of Science, Cleveland East Tech, Barberton, and Fairview. Teacher resource centers have been established at each school, including the capability to connect with the Internet.

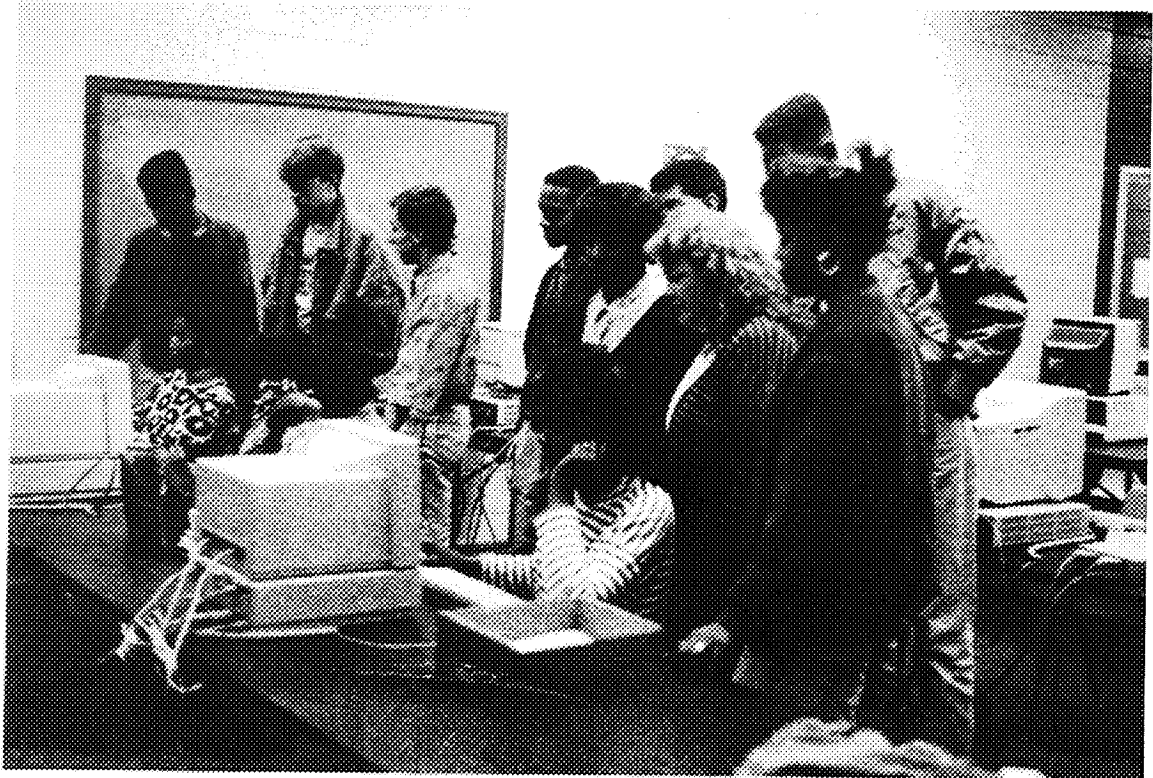
Active participation by LeRC researchers is evident in their "Adopt-a-Class." Researchers attended a training session to aid them in working with a teacher to develop an aeronautic project involving computing which a class could accomplish in FY94. Projected projects include visualizing physics laboratory data; constructing a wind tunnel; modeling blood flow through an artery; investigating the design and fabrication of Shuttle and solar cells; and, developing a flood model of the Cuyahoga River.

Significance: The LeRC K-12 program, although in its infancy, demonstrates that with the cooperation of school officials and teachers, and minimal NASA resources and expertise, science, technical and mathematics curricula at all K-12 grade levels can be significantly enhanced by the introduction of appropriately designed and applied computing curriculum components.

Status/Plans: Funding from FY94 will be used to conduct another training session in the summer. Four or five additional schools will be added to the program. An additional computer will be provided to each of the pilot program schools. Two local teachers will be sent to *Supercomputing 93*. Researchers will continue participating in "Adopt-a-Class".

Point of Contact:

Gary P. Warren
NASA Langley Research Center
g.p.warren@larc.nasa.gov
(804) 864-2162



Explorations in Supercomputing Program

Objective: The objectives of the Explorations in Supercomputing (EiS) program are to train teachers in Alabama, Arkansas, Mississippi and Tennessee in High Performance Computing and Communications (HPCC) methods and to encourage the implementation of HPCC in secondary mathematics and science curricula. EiS-trained teachers become "Master Teachers" who are capable of training teachers in their local districts.

Approach: The EiS approach allows extensive contact with the teachers during the training period. Two teachers from each school are invited to participate in an in-depth, two-week workshop during the first summer. During the following academic year, the teachers host a regional workshop and instruct other teachers in the basic methods of computational science, as well as remaining in close contact with the program directors. After a second summer of training, the participating teachers can function as Master Teachers for their region beginning the second academic year.

These Master teachers are expected to teach a class on computational science and to have their students do computational science projects. They also train area teachers in the fundamental concepts and methods of computational science during three days of in-service during the academic year. The students in the classes learn the methods of high performance computing, apply these methods to actual science or mathematics problems and display their projects in state or regional competitions. The teachers trained during the in-service workshops gain access to the Internet and discover how this access can enhance their classes.

Accomplishments: During August 1993 teachers attended a two-week training workshop in Huntsville, Alabama. These teachers returned to their classrooms prepared to expose students to technology. In the first year of operation, the schools and teachers were not prepared to create complete new courses. They were able, however, to implement much of what they learned in their existing courses.

Significance: Science and engineering are accomplished not just by experiment and

theory, but also through simulation and computation. This is not a "fad" which will soon be replaced by something new; this represents a fundamental change in how science will be done in the future. Unfortunately, teachers receive little, if any, training in how to do computation in science and mathematics. At the same time, it is imperative that high school graduates be prepared for careers in science and engineering. Part of their preparation must include simulation and computational solutions.

In addition to the educational benefit from including computation in education, there are additional benefits. Students and teachers find high performance computing, with its access to the Internet, its mathematical modeling and its visualization of results, to be both challenging and invigorating. All students can accomplish computational science projects if given the appropriate guidance by the teacher, and data indicate that that students who do accomplish their goals and projects have a higher level of self esteem and confidence in their ability to continue in science and mathematics courses.

Status/Plans: The teachers will return during the 1994 summer for additional training. In the 1994-95 academic year these schools will provide in-service training for area schools. By establishing a computing infrastructure in these schools and by training teachers to be leaders in their states, other schools will become involved in computational science. This has been demonstrated in Alabama in the past few years by the growth in the numbers of teachers and students working on computing projects and learning computational methodology.

Points of Contact:

Jim Pruitt
Marshall Space Flight Center
jim.pruitt@msfc.nasa.gov
205-544-0213

John Ziebarth
Univ. of Alabama in Huntsville
ziebarth@cs.uah.edu
205-895-6093

(THIS PAGE LEFT BLANK INTENTIONALLY)

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY (leave blank)		2. REPORT DATE November 1994	3. REPORT TYPE AND DATES COVERED Technical Memorandum	
4. TITLE AND SUBTITLE NASA High Performance Computing and Communications Program: Annual Report, 1993			5. FUNDING NUMBERS	
6. AUTHOR(S) Lee Holcomb, Paul Smith and Paul Hunter				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA TM-4653	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category-82			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) <div style="border: 1px solid black; padding: 10px;"> <p>The National Aeronautics and Space Administration's HPCC program is part of a new Presidential initiative aimed at producing a 1000-fold increase in supercomputing speed and a 100-fold improvement in available communications capability by 1997. As more advanced technologies are developed under the HPCC program, they will be used to solve NASA's 'Grand Challenge' problems, which include improving the design and simulation of advanced aerospace vehicles, allowing people at remote locations to communicate more effectively and share information, increasing scientist's abilities to model the Earth's climate and forecast global environmental trends, and improving the development of advanced spacecraft. NASA's HPCC program is organized into three projects which are unique to the agency's mission: the Computational Aerosciences (CAS) project, the Earth and Space Sciences (ESS) project, and the Remote Exploration and Experimentation (REE) project. An additional project, the Basic Research and Human Resources (BRHR) project, exists to promote long term research in computer science and engineering and to increase the pool of trained personnel in a variety of scientific disciplines. This document presents an overview of the objectives and organization of these projects, as well as summaries of early accomplishments and the significance, status, and plans for individual research and development programs within each project. Areas of emphasis include benchmarking, testbeds, software and simulation methods.</p> </div>				
14. SUBJECT TERMS NASA programs, research and development, parallel programming, software engineering, computerized simulation, computational fluid dynamics, aerospace engineering, space exploration, earth sciences			15. NUMBER OF PAGES 119	16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclass	18. SECURITY CLASSIFICATION OF THIS PAGE Unclass	19. SECURITY CLASSIFICATION OF ABSTRACT Unclass	20. LIMITATION OF ABSTRACT Unlimited	

Available from NASA Center for AeroSpace Information
800 Elkridge Landing Road
Linthicum Heights, MD 21090-2934
(301) 621-0390

PRECEDING PAGE BLANK NOT FILMED