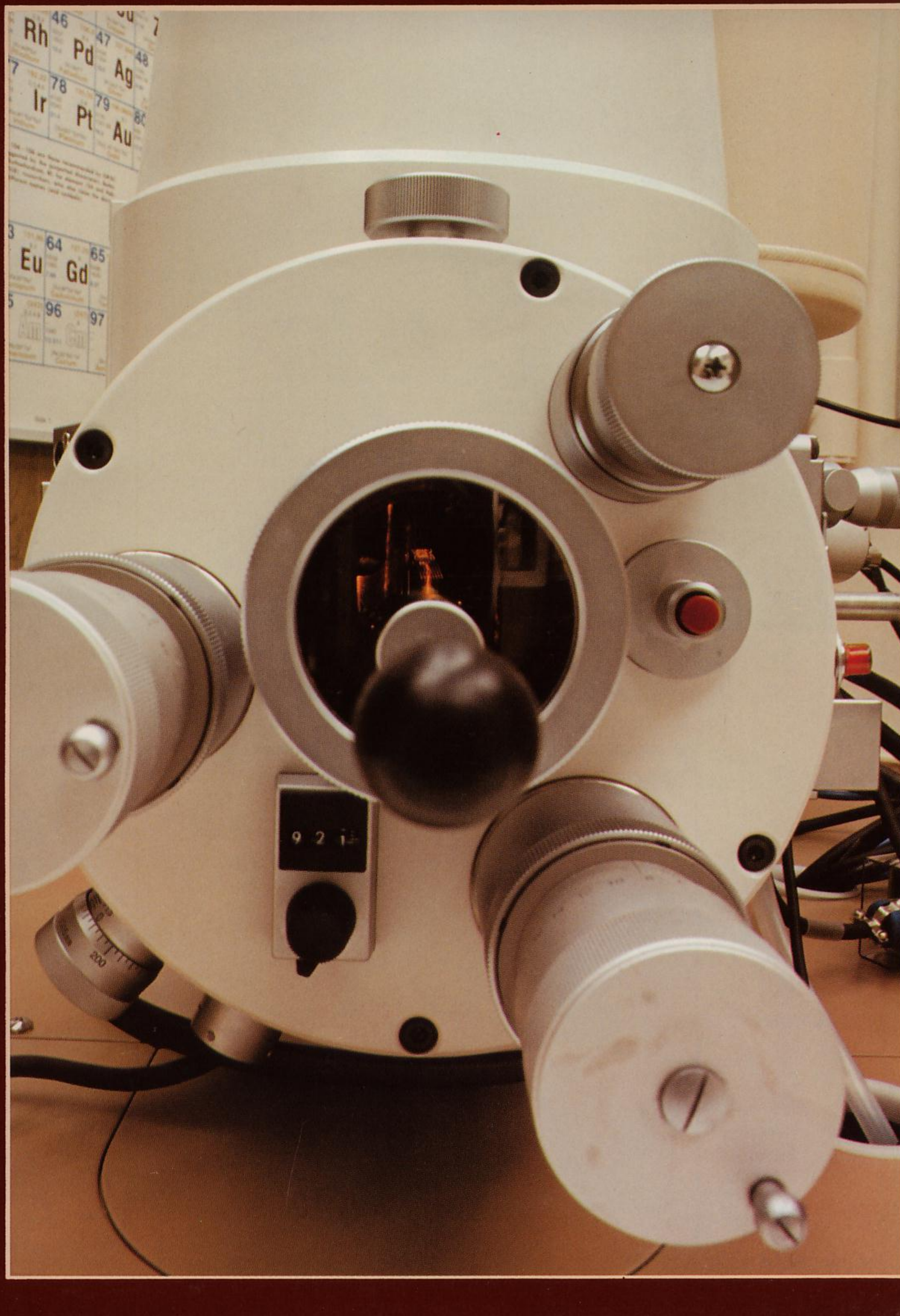
A CRAY RESEARCH, INC. PUBLICATION

CRAY CHANNELS

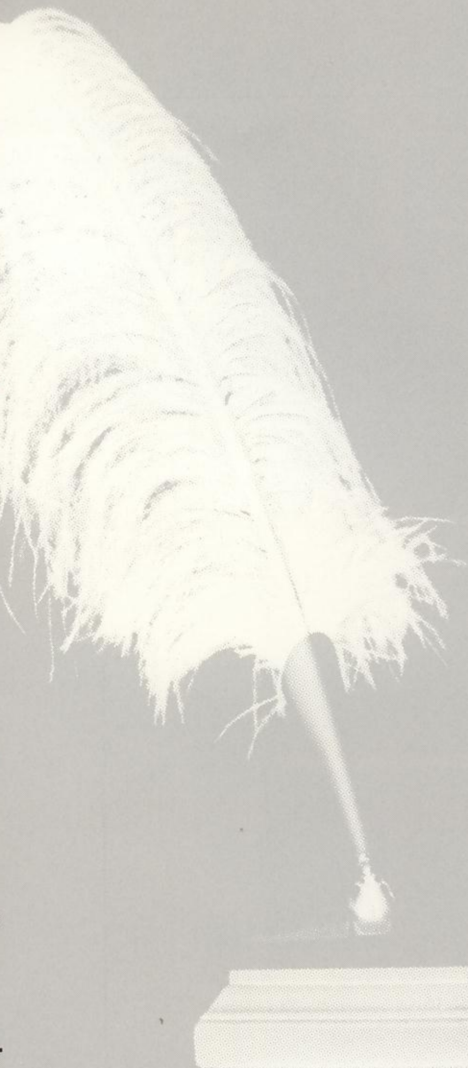
Volume 4, Number 2

FEATURE ARTICLES:

REGULAR COLUMNS:



From the editor's desk



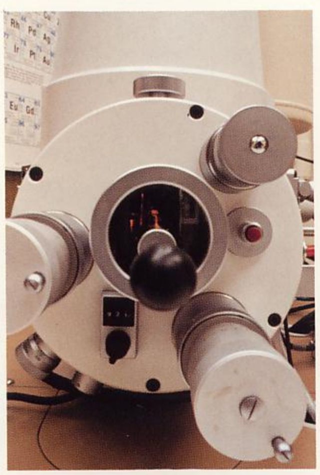
Aircraft design today as compared to that of a few decades ago is in another world. The trial-and-error research involved in creating those "fantastic flying machines" has given way to a highly exact science combining knowledge from several scientific disciplines. On the leading edge today are the computational aerodynamicists, those scientists who draw from aerodynamics, mathematics, fluid physics, and computer science to study the performance of aircraft in flight. Our first feature article describes this emerging concept and tells of the impact it could have on the aerospace and commercial airline industries.

In our second feature article, noted computer scientist John Levesque shares with us the excitement, the challenge, and the frustrations that program optimization for supercomputers has offered him over the past decade. His reminiscences may bring back many memories for you, if not a keener appreciation of the progress that has been made in just ten years.

As always, our issue includes articles on the latest news from Cray Research and from you. In future issues, we plan to bring you articles on structural analysis, seismic analysis, and more on graphics. If you have ideas for other subjects you'd like to see covered in future issues of **CRAY CHANNELS** please call us or drop us a note. We're interested in what you're thinking about!

—T.M.B.

About the cover



Semiconductor chip ready for observation in the Scanning Electron Microscope

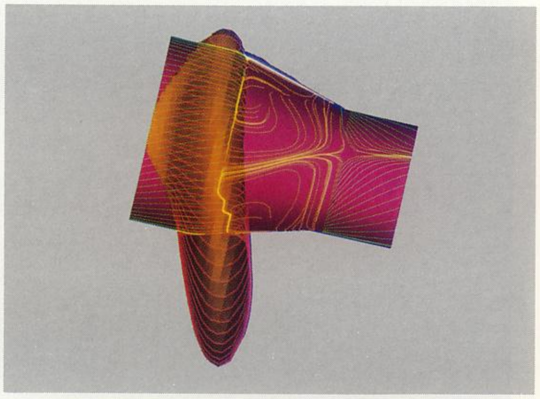
Cray Research now uses a Scanning Electron Microscope (SEM) to conduct reliability enhancement studies on various CRAY system components. The SEM, located in Chippewa Falls, is used to identify problems with integrated circuits, printed circuit boards and other mechanical parts. It emits an electron beam that moves back and forth across the sample. The sample gives off electron and X-ray signals that are detected and amplified by the microscope. A CRT attached to the SEM then produces a high resolution magnified image. The SEM has the ability to magnify an image 10 to 180,000 times. By looking at these magnified images, researchers can accurately identify and correct component failure to ensure CRAY system reliability. The photograph shows an integrated circuit in the SEM readied for observation.

CRAY CHANNELS

A Cray Research, Inc. publication Volume 4, Number 2

CONTENTS

FEATURE ARTICLES



2

REGULAR COLUMNS

CREDITS

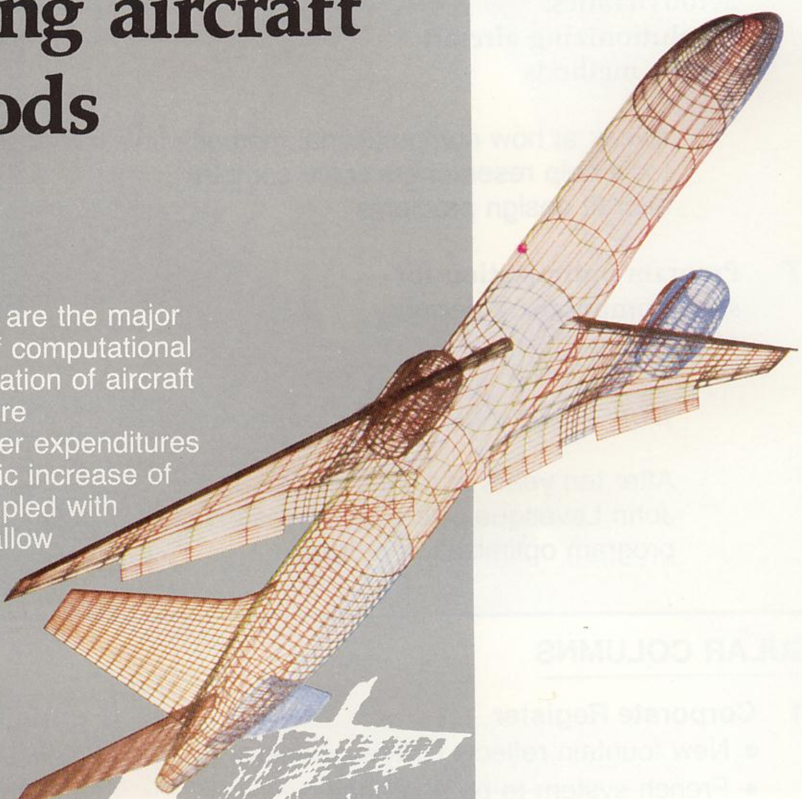
Editor	Tina M. Bonetti
Assistant editor	Carol A. Smith
Art/Design	Chad Jewett
	Kathleen Van Kirk
	Jeanne M. Jacobowski

CRAY CHANNELS is a quarterly publication of Marketing Publications, Cray Research, Inc., P.O. Box 154, Minneapolis, Minnesota 55440. CRAY CHANNELS is intended for users of Cray Research computer systems and others interested in the company and its products.

Requests for copies of CRAY CHANNELS may be addressed to the Marketing Publications Department of Cray Research at the above address. If you have a feature article or news item to submit for publication, please contact the editor of CRAY CHANNELS either by mail or by telephone at (612) 333-5889.

Computational aerodynamics:

revolutionizing aircraft design methods



Technical and economic factors are the major reasons for the advancement of computational aerodynamics. Each new generation of aircraft design requires increasingly more experimentation, leading to larger expenditures of time and money. The dramatic increase of computer speed and power coupled with declining costs for that power, allow computational aerodynamics to play a greater role in aircraft design with exciting results. This powerful design tool enables researchers to develop better designs, in less time with less risk of error.

Introduction

Computational aerodynamics is a composite of the four disciplines of mathematics, aerodynamics, computer science and fluid physics. When these fields of study are combined, scientists are able to predict the performance of an aircraft in flight. This is done by using a grid system to model a wing section or other portion of the aircraft and the airflow around it, simulating the properties of the airflow over time and taking into account the governing laws of fluid physics.

The equations expressing fluid physics theory were identified early in the 19th century. However, because of their extreme complexity they could only be solved for special cases. In the first half of the twentieth century, researchers made several attempts to mathematically analyze fluid physics as it related to aerodynamics, achieving mixed results. Less than gratifying results were obtained for analyses of flow past blunt shapes and flow moving at the Mach 1.0 level (transonic flow). Somewhat more usable predictions were derived for subsonic flow past streamlined shapes and for supersonic flows above Mach 1.5. The availability of the digital computer in the early 1950's raised hopes that these difficult equations could be solved, but that power proved inadequate.

Researchers have been struggling for the past 30 years to develop computational aerodynamics into a viable design methodology. The advent of the CRAY-1 has allowed the discipline to play an increasingly significant role in aircraft design. Cray Research computers, with their tremendous computing power and multi-million word memories, provide the aircraft designer with the power to exploit the efficiency of the computational aerodynamic design tool. That capability is becoming essential in designing fuel efficient, technologically advanced aircraft in fewer design cycles.

Comparing computational design methods with traditional methods

Computational design methods contrast sharply with conventional design methods. Under conventional methods, actual experimentation is conducted using wind tunnel tests. Portions of the aircraft, called wind tunnel models, are placed in these wind tunnels and their reactions monitored. These experiments are very expensive; a single wind tunnel model typically costs about $100,000 to construct and test. In designing a new aircraft, up to $50 million can be spent on wind tunnel testing alone. Furthermore, construction of the model is very time consuming and provides limited data.

With the use of computers many more preliminary designs can be studied in economical and timely manner, resulting in better design selection. Designers select those configurations that optimize the performance parameters and verify the computed results by testing them in the wind tunnel. Then the optimum design can logically be chosen. Costly wind tunnel testing is decreasing in importance for preliminary design, but it remains an important design verification tool. The complementary nature of computational methods and actual experimentation is illustrated in Figure 1.

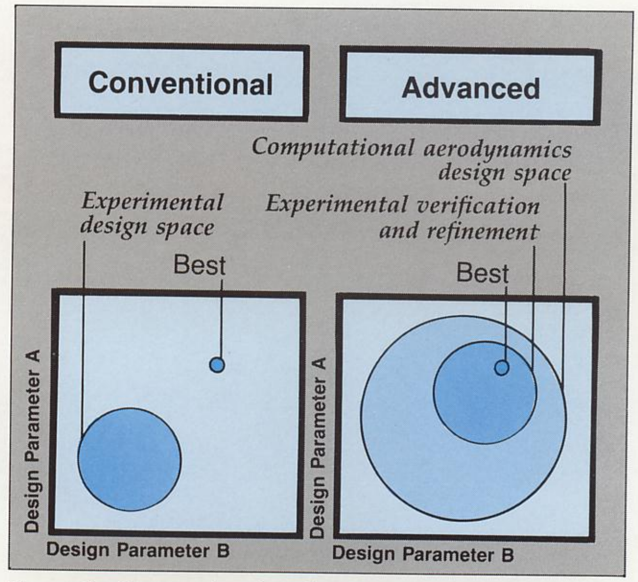
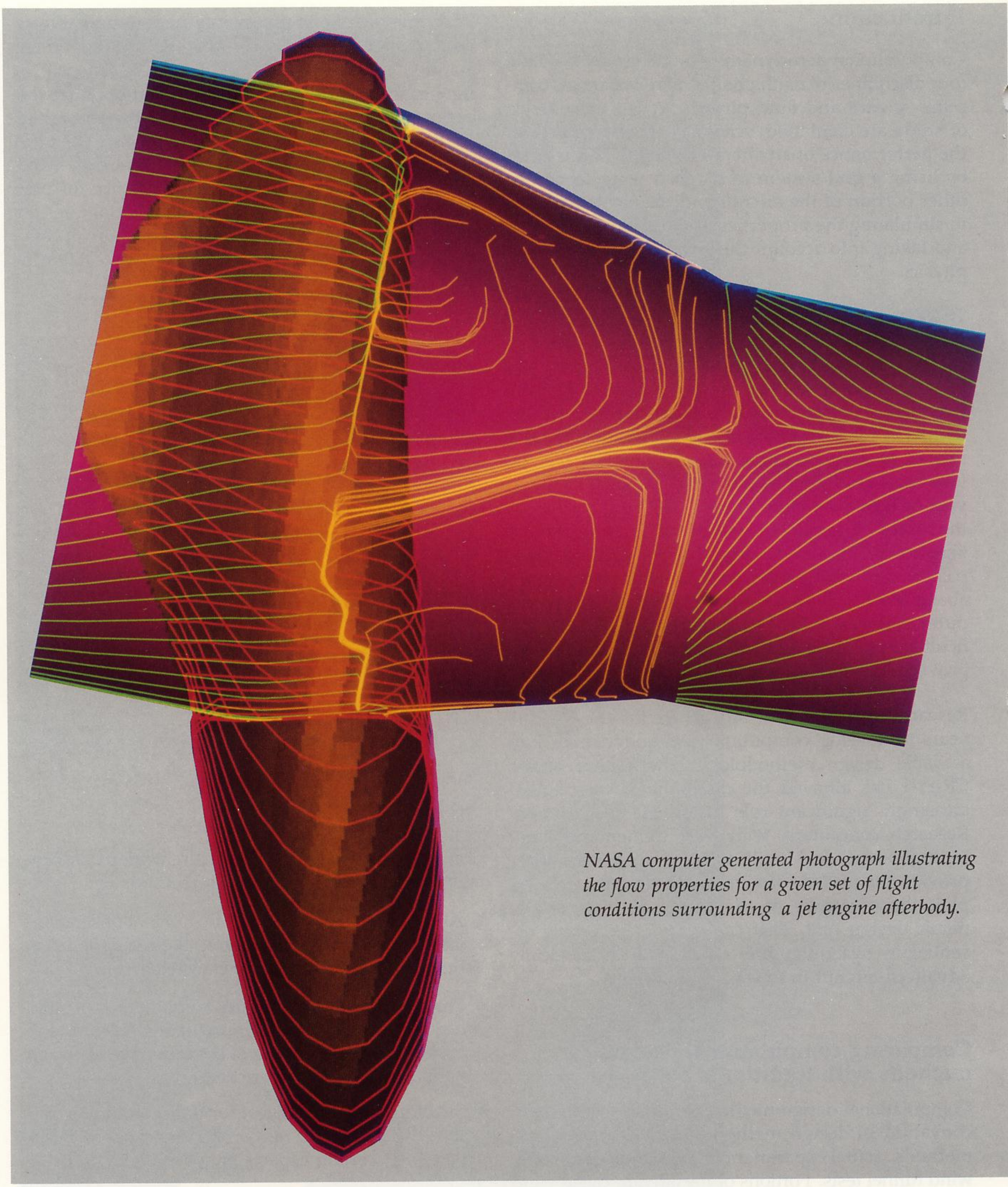


Figure 1 Conventional and computational design space comparison

Computational aerodynamics offers many advantages over conventional design methods by:

- Allowing many more combinations of design variables to be tested, increasing the likelihood that optimum designs are being selected.
- Providing a much stronger combination of fluid physics theory and experimentation, resulting in a greater understanding of the influence of design variables on aircraft performance.
- Enabling scientists to test for flow conditions outside the operating range of ground-based facilities, e.g., certain aspects of space flight.
- Providing direct estimates of free-flight conditions without the usual distortions of test results introduced by the wind tunnel, such as wall and support interference effects and Mach number limitations.
- Using the powerful mathematical theories of optimization to develop aircraft shapes that would not have been developed otherwise.



NASA computer generated photograph illustrating the flow properties for a given set of flight conditions surrounding a jet engine afterbody.

Computational design methods are only beginning to emerge from developmental stages. In comparing empirical testing with the evolving discipline, Dr. Antony Jameson, a noted computational aerodynamicist, commented, "It makes no more sense to blindly construct dozens of wind tunnel models in designing aircraft, than it does to build a dozen Verrazano bridges and wait to see which one will fall down first. Computational aerodynamics provides a logical method by which wind tunnel models are selected." Dr. Paul Rubbert of the Boeing Company explained, "Computational analysis gives us the ability to look at airflow effects over a model in great detail, while wind tunnel tests provide a macroscopic view of the airflow. The combination of the two gives us very sophisticated data."

The complexity of the problem

The peculiarities of computational aerodynamics challenge researchers daily. Among the many factors that must be considered in conducting these analyses are:

- The problem of computing airflow in an exterior domain. In order to reduce the problem to a finite number of equations, the infinite domain of the airflow must be truncated to analyze only that area necessary to obtain an accurate solution.
- The geometric complexity of the physical design of the aircraft. This is especially true of military aircraft designs.
- Sensitivity of aircraft performance to slight changes in speed or other flying conditions.
- Non-unique solutions of the flow equations for certain flight conditions.
- Inadequacy and complexity of aerodynamic algorithms. At subsonic speeds the equations are elliptic in nature, at supersonic, hyperbolic.

Among other industry experts, NASA officials agree that the greatest step in the advancement of aerodynamic technology would be the ability to solve the full Navier-Stokes equations. These powerful equations, identified in 1827, provide, for most applications, a sufficient description of the laws of physics governing fluid motion, i.e., the conservation of mass, energy and momentum. Even with the CRAY's massive power, it is still not possible to solve them in total because it is not possible to compute for very small scales of turbulence. NASA predicts that computers will not be able to completely solve these equations before 21st century.

A system of reduced equations neglecting various terms of the Navier-Stokes equations (such as viscosity and flow perturbation induced by the aircraft) is used in their stead. These equations are still extremely complex, and it is only within the past 15 years that computers have been able to tackle them in a practical way. As the complexity of the equations increases, the accuracy of the approximation increases. Likewise, as the complexity of the geometry being analyzed increases, the more useful the result becomes. The analysis of a geometry in three dimensions is preferred over one in two dimensions. Unfortunately a tremendous amount of computation is required to accomplish this. For example, the analysis of a two-dimensional geometry may require one billion floating point operations, whereas an analysis of the same geometry in three dimensions can require 50 billion floating point operations.

What can be attempted?

The computational aerodynamicist is charged with the selection of a solvable model that provides a sufficient level of useful information. Due to the limitation of computer power, there is an inverse relationship between the complexity of the mathematical equations describing flow and the complexity of the geometry that can be analyzed. When analyzing the airflow over an entire aircraft, only a simple set of linear equations can be executed. This analysis neglects a number of important factors of airflow, thus reducing its accuracy. So in many cases it may be better to approximate flow over a simple geometry like a wing section, using a complex numerical model to obtain more accurate data about the flow. Figure 2 illustrates the relationship between some well-known flow equations and the complexity of the geometries that can be analyzed.

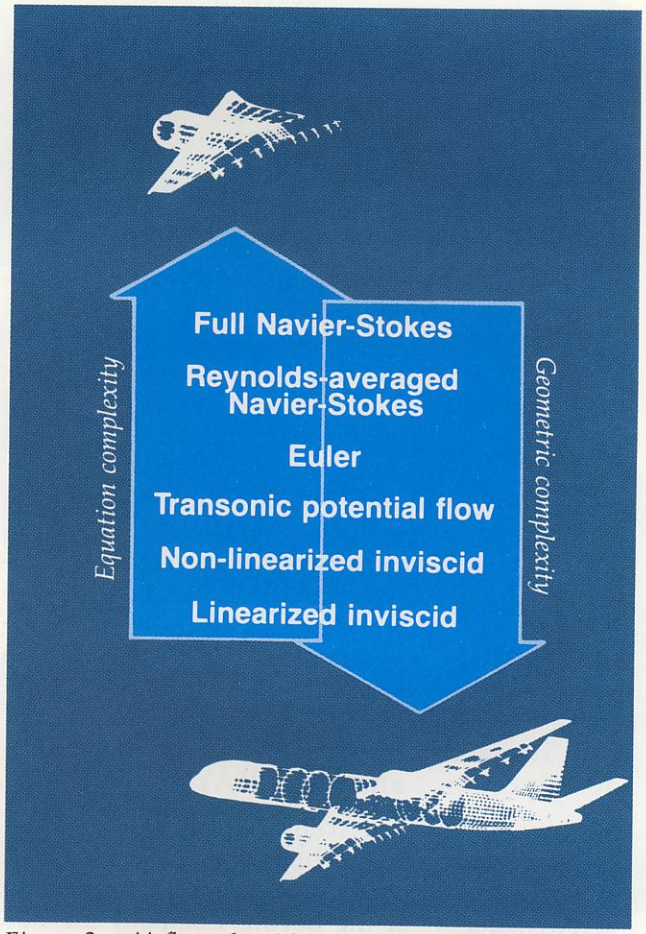


Figure 2 Airflow algorithm hierarchy

The extent of computer operations required by three typical numerical models is illustrated in Figure 3. With a mesh of 100,000 cells, 95 million floating point operations are executed per cycle. Obtaining the steady-state solution (where the values of the cell variables no longer change) requires 200-500 cycles. On the CRAY-1/S, 3-D Euler solutions are currently being computed within eight minutes on a 60,000 cell mesh. The emergence of the CRAY X-MP brings that time down to two minutes. The CRAY's architecture is ideal for executing these highly vectorized computations. A special version of FLO 57, a code

Type of algorithm	Floating point operations per cell per cycle	Number of cells	Floating point operations per cycle (in millions)	Number of cycles	Number of operations (in billions)	CRAY time (in minutes)*
Potential (3-D)	500	10,000	50	100- 200	5-10	2 - 4
Euler (2-D)	400	5,000	2	500-1000	1- 2	0.4- 0.8
Euler (3-D)	950	100,000	95	200- 500	20-50	8 -21

*Assumes 40 million floating point operations per second

Figure 3 Typical computations and timings for FLO 57

that executes the 3-D Euler equations developed by Dr. Antony Jameson for CRAY computers, is currently being run by the Boeing Company on the Boeing Computer Services Company's two-million word CRAY-1/S.

In order to construct a discrete approximation to the continuous problem, the domain is divided into small cells by introducing a computational mesh, and the flow is defined by the values of pressure, density and momentum in each cell. The calculation of the flow is then reduced to solving a set of difference equations representing the equilibrium of the flow in each cell. The evolution of the flow must be followed until the steady-state solution for a given set of conditions (Mach number, angle of attack) is achieved. The final values for each cell in the flow field, especially those attached to the airfoil, provide the data needed for analysis of the aircraft performance (lift, drag, pitching moment, etc.). Aerodynamicists then integrate data collected about the performance of the different portions of the aircraft along with empirical models and derive a prediction about the actual performance of the entire vehicle.

The impact of computational aerodynamics and the CRAY-1

Computational aerodynamicists are able to solve increasingly intricate equations for increasingly complex geometries with CRAY systems. Computational aerodynamicists at the Boeing Company are now able to approximate potential flow calculations for commercial aircraft in three dimensions. And earlier this year, for the first time, the very difficult Euler equations were computed for a three-dimensional wing and fuselage configuration on the Boeing Computer Service Company's CRAY-1/S. Their system can solve the 3-D Euler equations on meshes with up to 70,000 cells within a practical timeframe. Approximately 500,000 points must be computed for a 3-D representation of an entire aircraft. The CRAY X-MP now makes this feasible.

The impact of computational aerodynamics on the aerospace industry promises to be profound. At Boeing, computationally derived improvements as low as .6% in current 747 designs are expected to result in annual fuel savings of about $75,000 per jet. It is expected that new aircraft designs developed through computational methods and verified by wind tunnel testing will result in designs 10%-20% more efficient than aircraft designed by conventional methods. For the commercial airline industry this means many millions of dollars of fuel cost savings annually. For the aerospace industry, it means maintaining the competitive edge with technologically advanced products. CRAY computers with their massive speed and power provide the key element in the continuing evolution of this emerging design tool. □

Acknowledgements

Figure 1 was presented in the keynote address by Victor L. Peterson at a workshop held at NASA Ames Research Center, October 4-6, 1977.

Figures 2 and 3 were presented by Dr. Antony Jameson at the Science, Engineering and the CRAY-1 Conference, April 5-7, 1982.

Photograph on page 4 courtesy of Steven Deiwert, NASA Ames Research Center.

Special thanks are extended to Dr. Antony Jameson of Princeton University for his input in the preparation of this article. Additional information was provided by Dr. William Ballhaus of NASA Ames Research Center and Dr. Paul Rubbert of the Boeing Company.

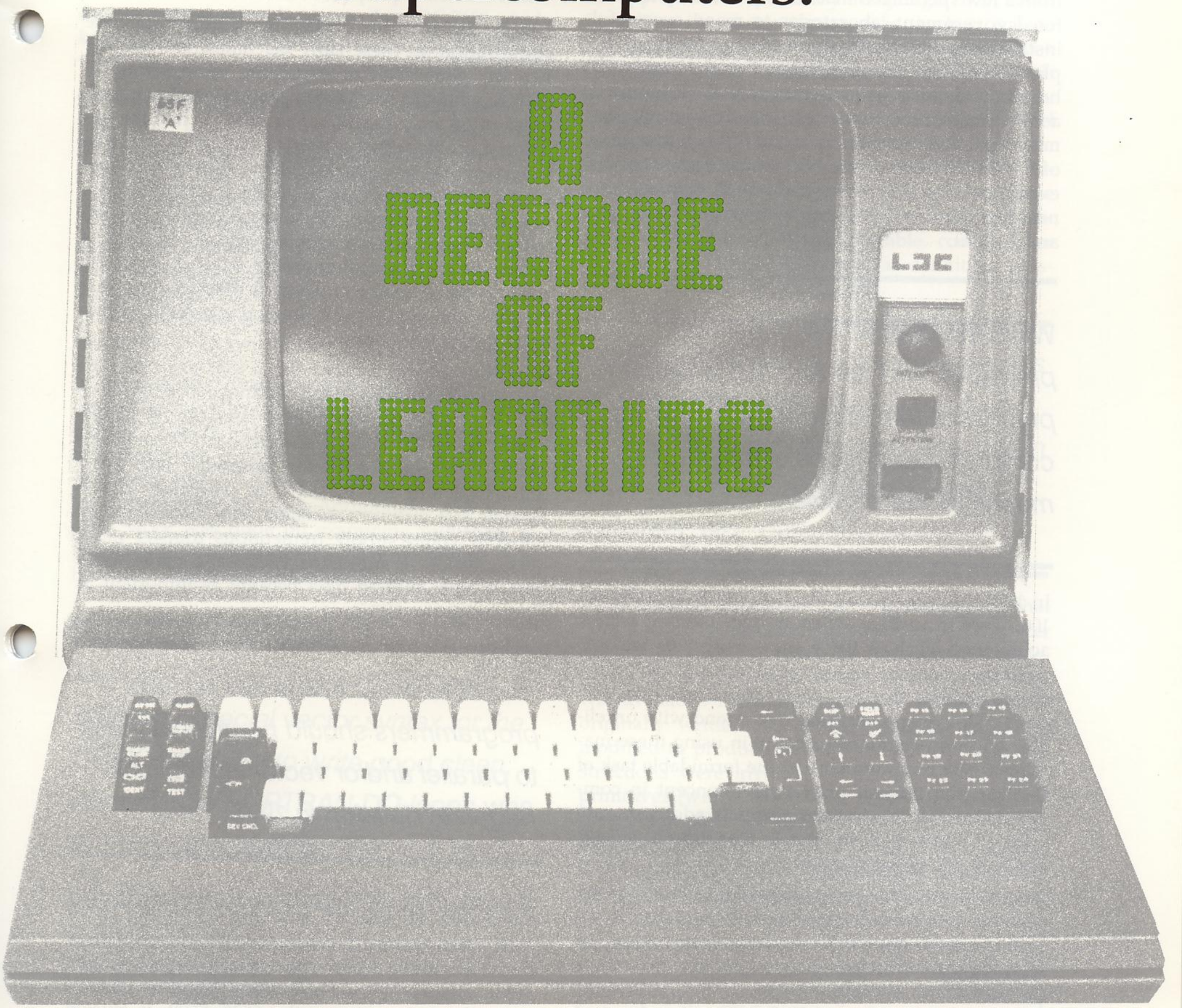
References

Chapman, Dean R., (1979), "*Computational Aerodynamics Development and Outlook,*" Dryden Lectureship in Research 79-0129.

Peterson, Victor, L. (1977), "*Computational Aerodynamics and the Numerical Aerodynamic Simulation Facility,*" NASA Conference Proceedings 2032.

Shang, J. S., Buning, P. G., Hankey, W. L., Wirth, M. C., (1980), "*Performance of a Vectorized Three-Dimensional Navier-Stokes Code on a CRAY-1 Computer,*" AIAA 79-1448R, September 1980, Vol. 18 No. 9, pp. 1073-1079.

Program Optimization For Supercomputers:



John M. Levesque, Pacific Sierra Research

John M. Levesque began working on research applications of computer technology 15 years ago at Sandia Laboratories after receiving his B.A. and M.S. in mathematics from the University of New Mexico in 1968 and 1971 respectively. Since then, he has been involved in the development and optimization of numerous application packages for organizations such as the Air Force Weapons Laboratory. In 1972 he organized a group of computer scientists for the purpose of applying new scientific techniques to a wide range of research problems. Monitoring Illiac IV code development was one of this group's early projects, giving Levesque and his colleagues a head start in gaining expertise in supercomputer use. The group was called upon to work on the first CRAY-1 installed at Los Alamos National Laboratory in 1976. John Levesque joined Pacific Sierra Research in 1979 and continues his involvement in supercomputer support through the development of specialized software products for a number of major supercomputer users.

The past decade has seen incredible growth in the number of large parallel computers in the world, from a few specialized machines found only in scattered government laboratories to many successful installations in both commercial and government applications. With this explosion of supercomputers has come a flood of programs that have required adaptation before executing efficiently on these new machines. The rapid changes and difficult problems of the last ten years have been very educational for everyone using high-speed computers. Certainly the next ten years will prove to be equally challenging and fruitful.

While the Illiac IV was a parallel processor and the Star 100 a vector processor, the difficulties in using the complete capabilities of the respective machines were similar.

In 1972, two supercomputers, the Illiac IV and Star 100, were in various stages of development and/or acceptance. Each of these supercomputers was capable of extremely high performance rates (between 50-100 million floating point operations per seconds). However, users had to contend with unreliable and non-standard software in using these machines. The programmer had the formidable task of familiarizing himself with a new concept in computing along with a new language. In addition, code had to be tested on simulators that did not always simulate the target machine.

While the Illiac IV was a parallel processor and the Star 100 a vector processor, the difficulties in using the complete capabilities of the respective machines were similar. The payoff of using the Illiac IV in parallel mode or using the Star 100 in vector mode was a decrease in execution time for the optimized kernel, where the kernels executed in one sixty-fourth to one hundredth of the former time. Unfortunately, visions of obtaining overall speed-up factors of 64-100 on entire applications soon dissolved with the realization that almost any application had to perform some amount of its operations in scalar mode. Scalar processing (one operation at a time), as opposed to either parallel or vector processing (many operations at a time), was significantly slower on these machines. Consequently, the user painfully accepted the fact that performance increases were an exponential function of the percent of code optimized. In order to achieve large increases in speed (factors of 10), more than 90% of the calculations in the program had to be done in parallel and/or vector mode.

Tedious restructuring of code was required for code optimization on these early machines. For the Star 100, the generation of very long vectors was required, while for the Illiac IV, vector lengths in multiples of 64 were desired. The Illiac IV had the additional problem that operands had to be available to each processor—the user had to worry about how long it took for processor N to get an operand from processor M's memory. The approach taken to complete an optimization for this machine was to start in heavily used subroutines and work through the entire code. A major problem at this time was that no tool was available to determine which routine used most of the time.

Input/output was another factor that surfaced with early supercomputers that continues to plague optimizers today. Because vector rates can be significantly faster than scalar speeds and I/O speed remains constant, the timing mismatch had to be addressed. On the Illiac IV, in particular, an extremely fast disk was available, that when mapped correctly could supply tremendous transfer rates. The question of supplying vector and parallel machines with enough data became as important as optimizing the CPU.

Two different philosophies on how programmers should be given access to parallel and/or vector instructions began to develop in 1975.

The CDC 7600, when operating in scalar mode, outperformed the Illiac IV and Star 100 in the 1970's with scalar speeds 5-10 times faster than either of these machines. Soon most of the large scientific applications targeted for the Star 100 or the Illiac IV were also being run on the CDC 7600. It quickly became apparent that programmers would need to optimize over 80% of the applications code in order to achieve performance rates on both of the older machines which would equal or surpass the CDC 7600. The CDC 7600 was very easy to use and the programmer typically did not have to worry much about effectively mapping onto its architecture. An interesting development which resulted from restructuring code for the Illiac IV and the Star 100 was that the optimized code frequently ran faster on the CDC 7600 as well. In fact, Lawrence Livermore's STACKLIB routines for the CDC 7600 resulted from simulation of the Star 100 on the CDC 7600.

A great deal of work was done by several CDC 7600 users to optimize code for its multiple segmented functional units. In fact, many people do not realize that the only elements lacking in the CDC 7600 which kept it from being referred to as a "vector processor" were "hardware vector instructions."

Several organizations developed "software vector instructions" in the form of libraries of subroutines. Performance factors of three to four over the initial CDC 7600 FORTRAN compiler (FTN) were possible since these optimal assembly language routines scheduled instructions to utilize the multiple-segmented functional units on the CDC 7600 at MFLOP rates of 15-20. This increase in performance from using STACKLIB disappeared as FTN got smarter and began scheduling its scalar instructions to use the multiple-segmented functional units efficiently.

In general, the extensive effort and research conducted by the scientific community to use the Star 100 and Illiac IV was not wasted. These machines introduced the concept of vector and parallel programming and indicated what the future had to hold for high-speed computing. Additionally, a lot of very good work was done on examining numerical algorithms and determining which algorithms were better for vector or parallel processing.

And since the user did not have to learn any special vector syntax for the CRAY, the ability to write good clean, vectorizable FORTRAN DO-loops was all that was required.

Two different philosophies on how programmers should be given access to parallel and/or vector instructions began to develop in 1975. It was at this time that parallelizers and vectorizers were developed to determine where parallel or vector instructions could be used in existing FORTRAN code. While the original intent of a number of these projects was to optimize an entire application code, it was soon learned that many "dusty deck" FORTRAN codes just could not be optimized well by these automatic optimizers. The real advantage of such a vectorizer was recognized later. Since transportability of user programs became very important, and since no one could agree on a standard vector syntax, automatic translation from FORTRAN to the vector syntax of the target machine was needed to assure transportability.

From these different philosophies, two schools emerged for vectorizing code. The first group took the approach that specifically designing a code, down to writing the vector syntax, was the best way to use the machine efficiently. The other group took the approach of writing FORTRAN DO-loops in such a way that they knew that the compiler or precompiler could recognize where vector or parallel instructions could be used and translate the code into efficient code on the target machine. The advantages and disadvantages of each approach were obvious—code written in vector syntax was definitely efficient but not transportable, code written in clean FORTRAN DO-loops was definitely transportable but not always efficient.

Part of restructuring a code so that it is vectorized consists of cleaning it up, eliminating subroutine calls from within DO-loops, eliminating unnecessary transfers, etc.

In 1976 a new supercomputer was announced that would soon become the standard of supercomputer power, the CRAY-1. When the first system was delivered to Los Alamos in June of 1976, little software was available. Programmers had to use a compiler originally written for another machine that was modified to produce code for the CRAY. Vector instructions were invoked by placing CRAY vector primitives (which looked like subroutine calls) in a program in place of FORTRAN DO-loops. Over the next couple of years the CRAY FORTRAN Compiler (CFT) appeared and became good at implicit vectorization. The scientific community's acceptance of the CRAY spread rapidly for three major reasons. Its scalar speed was faster than any other machine around, it was easy to use, and it was very reliable. The CRAY was a machine that was a factor of two faster than the CDC 7600 in scalar mode and factor of ten in vector mode. When codes were directly converted, immediate increases in processing speed were obtained with little or no vectorization. And since the user did not have to learn any special vector syntax for the CRAY, the ability to write good clean, vectorizable FORTRAN DO-loops was all that was required.

The ability to restructure codes to utilize vector machines became more and more important. Since one did not have to spend much time worrying about the stability of the machine or familiarizing oneself with a new language, more time could be spent on coming up with new techniques for rewriting scalar code to execute in vector mode. New ways of looking at recursive algorithms and good techniques for vectorizing multiple path decision processes were soon developed.

Now that nice, clean, transportable vectorized code was feasible, a very important question arose. How well would the restructured vectorizable code run on a scalar machine? Several issues had to be considered. Part of restructuring a code so that it is vectorized consists of cleaning it up, eliminating subroutine calls from within DO-loops, eliminating unnecessary transfers, etc. All of these techniques result in scalar code running faster. The problem arises when the user has to handle complex decision processes. *All* techniques for handling these decisions typically introduce some overhead. Here we define "overhead" as those operations which are performed in vector mode which are unnecessary in scalar mode. That is, operations that were needed to provide the ability to vectorize a DO-loop which were not required in the scalar loops. The answer to the original question is that sometimes the restructured code runs faster than the original on the scalar machine and sometimes it runs slower on the scalar machine depending on what restructuring techniques were used. However, the code *always* runs faster on the vector machine.

...sometimes the restructured code runs faster than the original on the scalar machine and sometimes it runs slower on the scalar machine ... However, the code always *runs faster on the vector machine.*

With respectable vector and parallel speeds complemented by enhanced scalar speeds and software maturity, the new vector architectures were accepted by the traditional supercomputer customers as well as by new commercial customers. Program optimization for these new machines was much more widespread. Users did not have to spend time worrying about the inadequacies of the machine's hardware and software, but could concentrate on cleaning up the FORTRAN and getting the most out of the machine. Vector processors spread into the commercial marketplace late in the 1970's.

Computer service bureaus obtained vector machines and offered them to the general public; no longer was use of these machines isolated to government laboratories.

Now back to the nagging problem of I/O. Here we have machines capable of performance rates 10-20 times the CDC 7600 with disks that transfer data at the same speeds as the CDC 7600 disks. This situation was the impetus for the introduction of the CRAY's I/O Subsystem and buffer memory. Given that buffer memory to main memory transfer rates are 15-20 times faster than disk to main memory, the I/O situation becomes easier to handle. However, the user who has a large scientific non-memory-contained application has to concern himself with using the I/O system efficiently or he will be I/O bound.

Multi-processing architecture, available with the CRAY X-MP, will play an increasingly prominent role in the scientific community's research over the next five years.

Just when vectorization techniques are catching up to vector machines, a relatively new concept is being implemented which once again will put a demand on the programmer. Multi-processing architecture, available with the CRAY X-MP, will play an increasingly prominent role in the scientific community's research over the next five years. The innovations in architecture and technology inherent in the system offer a new dimension of parallelism in high-speed computers. It is now possible to perform multiple vector instructions at the same time, multiple scalar instruction streams at the same time, or a scalar stream at the same time as a vector stream.

The CRAY X-MP is much different from the parallel machines such as the Illiac IV since it can be doing different operations at the same time while the Illiac IV had to be doing the same operation in each processor at the same time. The advent of this new machine will now require that new constructs be added to languages to handle multi-processors, compilers will have to be taught how to recognize multiple-processes from standard FORTRAN code, and optimizers will have to learn how to rewrite code so that they can more effectively use these new architectures.

The continuing evolution of supercomputer development presents the ongoing challenge for the program optimizer. The power of each new generation of hardware presents a new set of capabilities that must be harnessed. The lessons learned early with the optimization of code for the Illiac IV and then used in CRAY-1 code optimization provide a good backbone for the future in code optimization. □

CORPORATE REGISTER

Top view of Octal, Cray Research's new sculpture at Mendota Heights.

New fountain reflects Cray's business

While construction is still progressing on the latest addition to Cray Research's Mendota Heights facility, work has been completed on an interesting structure outside the facility's front entrance. Environmental artist Andrew Leicester designed and supervised construction of a waterwork labyrinth with a hidden code.

Named "Octal," the circular structure is composed of cement and redwood latticework. The circle is divided into sixteen segments corresponding to the form of the CRAY-1 computer. Four latticework walls spiral in towards the center of five concentrically stepped platforms. The pattern of the four walls is taken from the octal codes of the letters CRAY (103, 122, 101, 131) reduced to their binary equivalents (01000011, 0101000011, 01000001, 01011001). The binary 0s signify that the pattern is to continue moving around the perimeter of the circles. At each occurrence of a 1, the pattern jumps inward to the next interior circle. Water streams from the tops of the walls in several places and flows through the pattern to the center. Redwood seats in the interior of the structure provide a cool, quiet place to sit.

Leicester has been intrigued with the idea of the interplay between art and technology for some time, and his works reflect that interest. "Octal" symbolizes Cray Research and its business in a striking way.

French system to be upgraded

GETIA will soon be upgrading from a one-million word CRAY-1 S Series Computer System to a two-million word system. GETIA was formed as a cooperative arrangement between Electricite de France and Compagnie Internationale de Services en Informatique (CISI). Since April of 1981, Electricite de France has used the CRAY for research and development purposes, while CISI, which is the fifth largest service bureau worldwide, has marketed CRAY time to outside customers.

NASA-Lewis places order for S/2200

The National Aeronautics and Space Administration (NASA) announced recently that it has selected a CRAY computer system for its Lewis Research Center. In mid-July, NASA announced that it will install a CRAY-1 S/2200 computer system at NASA-Lewis in Cleveland, Ohio. The new system, which will be delivered during 1982, will have two million words of main memory and a one-million word I/O Subsystem. It will be used to study the performance of new jet engine designs through simulation of flight conditions. This system will be the second CRAY computer installed for use by NASA. In 1981 the NASA-Ames Research Center in Sunnyvale, California installed a CRAY-1 S/1300.

CORPORATE REGISTER

New COS Version, NOS and NOS/BE service released

A new version of the CRAY-1 Operating System (COS) is now available. COS Version 1.11 executes on CRAY-1 Computer Systems having I/O Subsystems or Data General MCUs. Significant features in this release include:

- On-line tape support with read/write capability of unlabeled and IBM/ANSI labeled tapes
- Station slot feature
- NSC HYPERchannel adapter multipoint communications via a Central Processing Unit channel
- Initial support of I/O Subsystem Buffer Memory-resident datasets
- New CRAY channel disk driver with improved error logging and increased reliability
- Interactive text editor (TEDI)
- Symbolic debugger (SID)
- OPTION and SUBMIT control statements
- Speeded-up system generation

Also announced as a service to Cray Research customers is a new version of the supplemental software for the CDC NOS and NOS/BE stations enabling these stations to be logically linked to COS 1.11 executing on a CRAY-1. Significant features included with this improved service are:

- Implementation of station commands and displays
- Station slot fields in the dataset header for customer use
- Constant monitoring of data transfer rates possible via operator display
- Multiple input spooling streams
- Reduced use of host computer resources

Customers may order using the same process as for all other standard Cray products.

Cray Research sponsors internal technical symposium

Cray Research, Inc. held its first Technical Symposium during the month of June. A total of forty employees representing every technical division within the company were invited to participate in the conference, which was held in northern Wisconsin. The dual purposes of the symposium were to allow participants to exchange information on technical projects under development within the company and to create a forum for the sharing of ideas, research, and experiences.

The conference activities spanned three days of technical presentations and workshops. Special presentations were made by several guest speakers, including John Rollwagen, Chairman of Cray Research and Hans Bruijnes, Deputy Director of the Magnetic Fusion Energy Computing Center at Lawrence Livermore Laboratories. Among the topics of technical discussion and presenters were: "The State of VLSI Circuit Simulation and Supercomputers," John May; "Multi-processing—A Compiler View," David Whitney; and "R & D in Algorithm Applications," Chris Hsiung.

Two-million word system to go to France

During the third quarter of 1982, Cray Research will install a two-million word CRAY-1 S/2300 in France. This system, which will be purchased, will be used by an agency of the French government for scientific research.

The CRAY S/2300 to be sent will be the second CRAY-1 system installed in France. An export license for this order has been approved by the U.S. Department of Commerce. Cray Research expects to ship two additional systems to France later this year.

Digital Productions places Cray's first order for X-MP

Digital Productions, Inc. (DPI), has ordered a CRAY X-MP/22 to be installed in the fourth quarter of 1983. DPI specializes in creating high-quality, high resolution film images for the entertainment, industrial and scientific communities, in a process called Digital Scene Simulation. The company has developed software programs that allow the CRAY to generate highly realistic images and special effects on film, such as the computer generated animation in the movie TRON. Digital Productions' founders and principals John Whitney, Jr. and Gary Demos received screen credits for their roles in shaping the use of Digital Scene Simulation in TRON.

Said Whitney, Jr., president of Digital Productions, "In order to achieve the next level of realism beyond TRON in computer generated special effects for feature films, the Digital Scene Simulation process becomes a computationally intensive

Cray Research computer center at the Mendota Heights facility.

application of computer graphics that demands the power of a CRAY supercomputer. DPI's own graphics software has been developed specifically for the CRAY. Utilizing its new tools, we will produce simulated scenes that suspend the film viewing audience's ability to tell the difference between live action intercut with photographically realistic computer simulation."

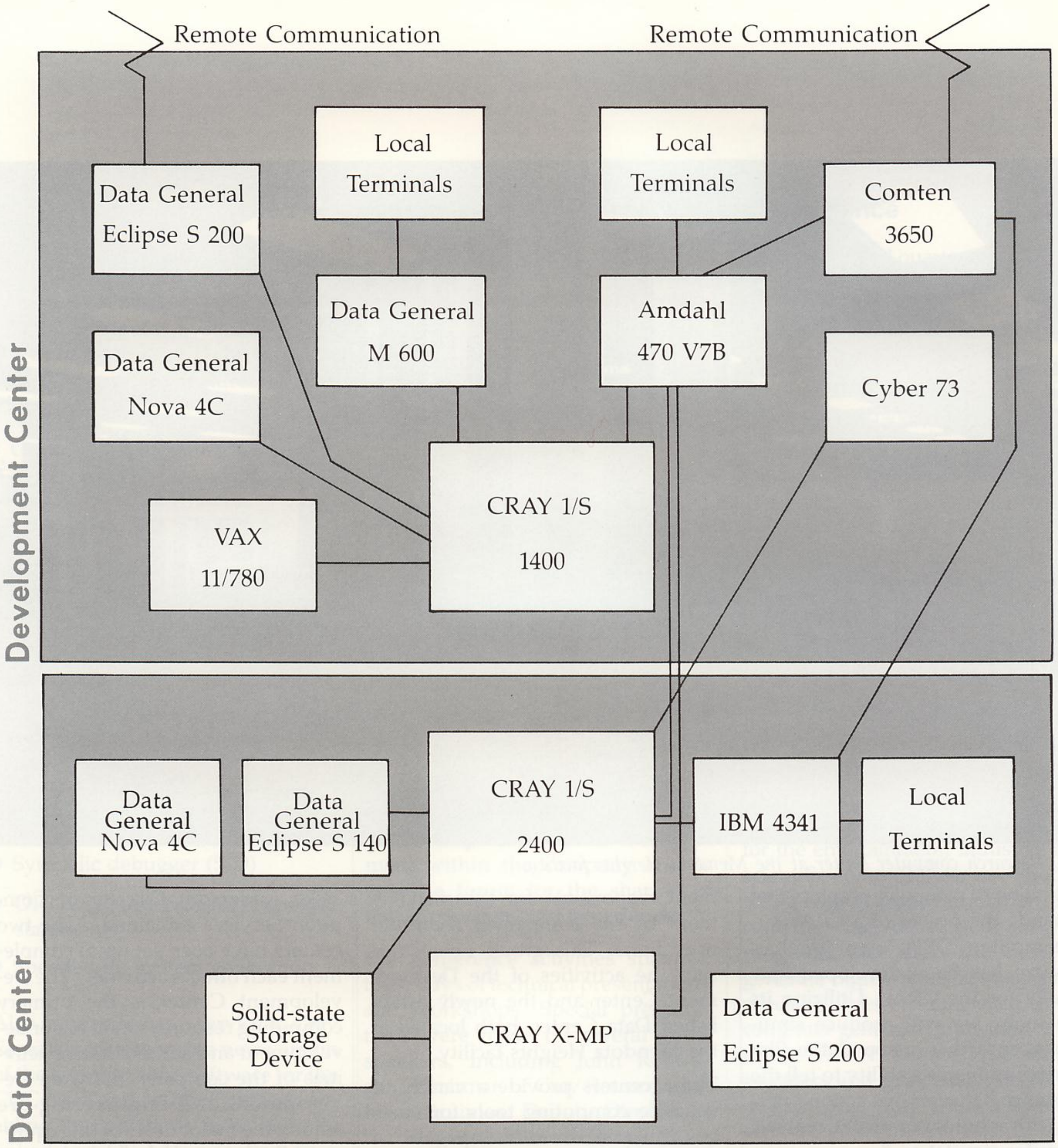
Cray Research expands computer service operations

In response to growing computing needs, Cray Research has established a Computer Services department responsible for the management of the company's computer operations. This group now oversees the activities of the Development Center and the newly established Data Center, both located at the Mendota Heights facility.

These centers provide a variety of valuable computing tools for users involved in the development of CRAY computers and execution of customer benchmarks. Computers housed there will include two CRAY-1/S computers, a CRAY X-MP, three I/O Processors, one Solid-state Storage Device (SSD), six front-end processors from a number of vendors, and over 150 terminals and peripheral units.

Mike Anderson, Director of Computer Services explained, "The two centers have been set up to complement each other's activities. The Development Center is the primary computing resource for all major development and administrative activities of the company. Software Development and Engineering are among the major users of this equipment."

"The Data Center," Anderson explained, "provides a production environment for customer benchmarks and testing in addition to software testing. Some SSD and general software development activities are also conducted on the Data Center equipment.



Cray Research computer center

"Overall, the network is designed to offer dedicated and batch time for both local and remote users," Anderson explained. He concluded: "The heavy level of systems use under a variety of conditions, ensures a comprehensive checkout of software prior to release to the field."

While the extensiveness of the computer center makes it a very powerful resource, it also can present a challenge in keeping all systems operational. Recognizing this fact, Computer Services has made the smooth operation of this network a key objective. Mike Anderson emphasized, "We recognize that the center is a computing resource for many diverse purposes, and we are committed to servicing users' needs."

APPLICATIONS IN DEPTH

New release of Applications Software Library

On July 1, Cray Research released a new version of its Cray Applications Software Library. With this release, the library has been very significantly reorganized and expanded. The reorganization makes it easier for users to exercise selectivity in ordering software. Furthermore, due to the reorganization, current library users may purchase additional software without risking duplication of software in most cases.

One copy of all user documentation now accompanies each software library section free of charge. In the past, documentation was available optionally for an extra charge.

Although there have been several administrative changes, the big news is really all the additional software that has been added to the library with this latest release. Below is a summary of the new or significantly revised programs in the library.

- FITPACK, a package of routines for curve and surface fitting using splines under tension
- MINPACK, a math package for nonlinear systems used at Argonne Labs
- A collection of more than 30 digital filter programs used widely in electronics, collected and standardized by IEEE
- A linear programming package provided by a Cray Research site analyst
- A scalar PASCAL compiler from Los Alamos National Laboratory
- TRAC PD2, a nuclear engineering code that solves problems similarly to TRAC-P1A but features improvements in the heat transfer model and in the thermodynamics treatment
- A newer and faster version of SPICE (2G2.5)
- Several more FLO programs for aerodynamics, including FLO 27M, FLO 30M, FLO 42, FLO 52, FLO 54, and a new version of FLO 57
- The quantum chemistry programs MULTAN 76 and HONDO 5
- Two hydrology programs, HEC-1 and HEC-2, from the U.S. Army Corps of Engineers
- SETS, a fault tree analysis program for nuclear analysis
- The nuclear engineering code DOT3.5
- A CALCOMP-compatible plotter package, GLDPLOT, which enables Calcomp-like graphics processing to be performed on the CRAY-1 and enables the use of the Gould 5000 electrostatic printer/plotter as a graphics output device
- Several new utilities programs, including a character conversion package and a series of utilities developed in the U.K.
- DMSP, a data management support processor that runs on an IBM/370 Series (or compatible) processor under the VM/370 Operating System as a stand-alone virtual machine, and controls the flow of data and control messages between the 370 and the CRAY-1

For more information on the new library release, please contact David Darling, Applications Department, Cray Research, Inc., 1440 Northland Drive, Mendota Heights, MN 55120.

APPLICATIONS IN DEPTH

Symposium proceedings available

On April 5-7 of this year, Cray Research sponsored a symposium entitled "Science, Engineering and the CRAY-1." The meeting gave specialists from a variety of applications areas an opportunity to compare notes and share experiences. During the symposium's three days, more than thirty speakers made informative presentations dealing with the automotive industry, seismic data processing, computer graphics, reservoir simulation, aerodynamics, electronics, chemical engineering, and large-scale processing in general.

Now, the abstracts and papers provided by the speakers have been assembled and bound into a symposium proceedings. The proceedings can be ordered directly from Nancy Williams, Applications Department, Cray Research, Inc., 1440 Northland Drive, Mendota Heights, MN 55120.

Cray Research produces new applications software directory

The Applications Department of Cray Research recently made available a new directory of CRAY applications software. This directory is a guide to the applications software and software tools currently available from a variety of sources for use on the CRAY computer systems. It is intended as a replacement for the "Scientific Applications Package Handbook" for those customers specifically interested in CRAY-available software. Ten applications sections are represented in the new publication: mechanical engineering, nuclear engineering, chemical engineering/geophysics, electronics, aerodynamics, pure science, mathematics and statistics, mathematical programming, graphics, and software tools. Software is further divided into four classes depending on whether it is distributed by Cray Research or by a third party and by whether it is fully supported or not. To obtain a copy of the directory, contact Nancy Williams, Applications Department, Cray Research, Inc., 1440 Northland Drive, Mendota Heights, MN 55120.

DI-3000 converted for CRAY systems

DI-3000, a device- and machine-independent graphics system conforming to the standards established for the SIGGRAPH Core system, has been converted to run on the CRAY computer systems. DI-3000 is suitable for intertasking and distributed processing environments. It supports a full range of graphics applications including CAD/CAM, business graphics, architectural design, and mapping.

Applications written using DI-3000 can drive most types of displays (storage tubes, color raster displays, vector refresh devices, plotters, and film recorders) without recompilation. Up to six devices can be run concurrently. DI-3000 will take advantage of the high performance features of advanced graphics hardware (e.g., hardware fill). Alternatively, advanced graphics methods are simulated in software for the less sophisticated devices.

DI-3000 is a proprietary product of Precision Visuals, Inc. For more information, contact Precision Visuals at 250 Arapahoe Avenue, Boulder, CO 80302, telephone (303) 449-0806.

CPS-1 converted for use on CRAY

Graphics users will be pleased to learn that the graphics package CPS-1 (Contour Plotting System) has been converted for use on the CRAY computer systems. CPS-1 produces sophisticated graphics displays of three-dimensional surfaces calculated from gridded or randomly distributed data points, with an emphasis on contour plots. Gridded surface and control point manipulation procedures are also available.

CPS-1 allows users great flexibility in designing the data manipulation processes and graphics output formats to best meet their needs. Automatic partitioning of data enables the interpolation of large datasets to very fine grip points. Graphic outputs exceeding the dimensions of the plotting system are automatically segmented for reconstruction. CPS-1 has its own data management system, complete with file maintenance capabilities.

Several optional modules are also available to further extend CPS-1's power. CPS-1 is fully supported and distributed by the Radian Corporation. For more information, call or write Radian at 8501 Mo-Pac Blvd., Austin, TX 78766, telephone (512) 454-4797.

CORPORATE ADDRESSES

Regional Sales Offices

Central Regional Sales Office
5330 Manhattan Circle, Suite F
Boulder, CO 80303
Tel: (303) 499-3055

Eastern Regional Sales Office
11120 New Hampshire Avenue
Silver Spring, MD 20904
Tel: (301) 681-9514

Petroleum Regional Sales Office
5858 Westheimer
Houston, TX 77057
Tel: (713) 975-8998

Western Regional Sales Office
Sunset Office Plaza
1874 Holmes Street
Livermore, CA 94550
Tel: (415) 447-0201

Sales Offices

Albuquerque, NM

Atlanta, GA

Austin, TX

Boston, MA

Chicago, IL

Dallas, TX

Laurel, MD

Los Angeles, CA

Mountain View, CA

Pittsburgh, PA

Seattle, WA

International Subsidiaries

Cray Research France, S.A.
3, Avenue de Madrid
Neuilly Sur Seine 92200
France
Tel: 01-747-9440

Cray Research GmbH
Wartburgplatz 7
8000 Munich 40
West Germany
Tel: 089-363076
TLX: 8415213211

Cray Research Japan, Limited
New Aoyama Bldg, West 1661
1-1 Minami-Aoyama 1-chome
Minato-Ku, Tokyo 107
Japan
Tel: 03-403-3471
TLX: T8128893

Cray Research (UK) Limited
Seymour House, The Courtyard
Denmark Street, Wokingham
Berkshire, England
Tel: 0734-791180
TLX: 848841

Corporate Headquarters
Cray Research, Inc.
608 Second Avenue South
Minneapolis, Minnesota 55402
Tel: (612) 333-5889