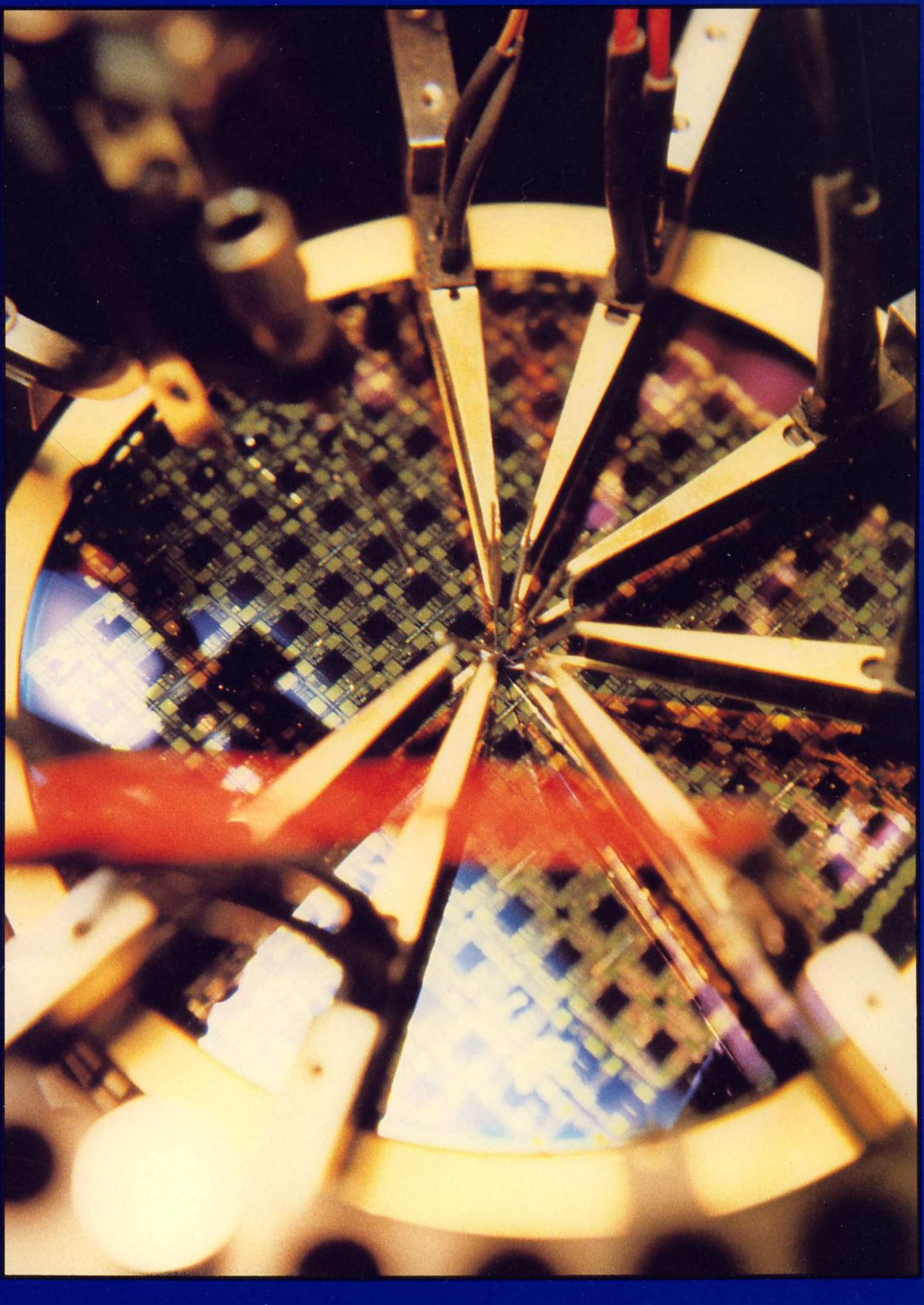
A CRAY RESEARCH, INC. PUBLICATION

CRAY CHANNELS

Volume 6, Number 4



FEATURE ARTICLES:

The impact of computational methods on aircraft design

Cosmic modeling

Image processing overview

DEPARTMENTS:

Corporate register

Applications in depth

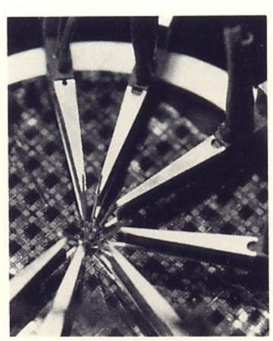
User news

IN THIS ISSUE

Aircraft designers were among the first people to exploit the power of supercomputers. Over the years, they have developed and refined computational methods for modeling increasingly complex flow phenomena. In this issue of CRAY CHANNELS, Dr. Paul Rubbert, of the Boeing Commercial Airplane Co., describes how today's computational methods are transforming the science of aircraft design. Our regular departments complement Dr. Rubbert's narrative with descriptions of a computationally designed flat-bottomed nacelle and a record breaking fluid flow calculation performed at Cray's Mendota Heights, Minnesota facility.

Along with the well-established uses of CRAY computers in the aerospace industry, CRAYs are helping scientists model a diverse range of natural and industrial phenomena. Other feature articles in this issue describe the role of CRAYs in astrophysical research and image processing. In addition, we highlight unique applications of CRAYs in modeling cell motion and chemical vapor deposition reactors.

As the number of CRAYs installed worldwide grows, so, it seems, does the imagination of the scientific community. As scientists and engineers devise more complex computational problems, the list of supercomputer uses continues to grow. It's our hope that, through CRAY CHANNELS, we can keep you abreast of the expanding range of applications for Cray computers.



On the cover is a probe station for testing semiconductors. The needle-like probes sample a device's electrical parameters for analysis by external testing equipment. Probe stations such as this one are currently being used in the development of gallium arsenide wafers at Cray's Advanced Research Projects Facility in Chippewa Falls, Wisconsin.

CRAY CHANNELS

A Cray Research, Inc. publication

Volume 6, Number 4

FEATURES

2 The impact of computational methods on aircraft design

Paul E. Rubbert, Boeing Commercial Airplane Co., Seattle, Washington

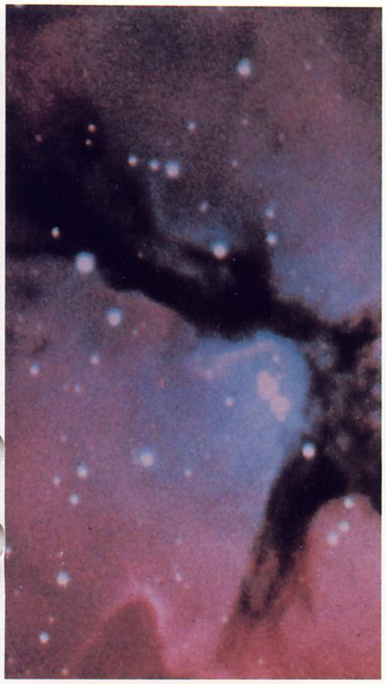
The aerodynamic design process is quite different today from what it was a few years ago — and it's obvious that further change is in store. Computational aerodynamics is coming of age.

6 Cosmic modeling: how to pack the universe into a Cray

Scientists use supercomputer simulation on a cosmic scale to enhance our understanding of the universe's evolution.

10 Image processing overview

The ultimate "image builder", image processing provides clarity and removes distortion from images to make them more useful to viewers.



6

DEPARTMENTS

14 Corporate register

Oil producers expand CRAY use, and the labs order more supercomputer power. Also, the Naval Research Laboratory comes on board.

15 Applications in depth

The Solid-state Storage Device helps set a milestone in computational fluid dynamics, and new mechanical simulation codes become available to CRAY users.

17 User news

Reproducing cell fertilization on the Cray, meeting an aircraft redesign challenge, and modeling complex fluid flow patterns and chemical reactions.

CRAY CHANNELS is a quarterly publication of Cray Research, Inc. intended for users of Cray computer systems and others interested in the company and its products. Please mail subscription requests, feature story ideas and news items to CRAY CHANNELS at Cray Research, Inc., 608 Second Avenue South, Minneapolis, MN 55402.

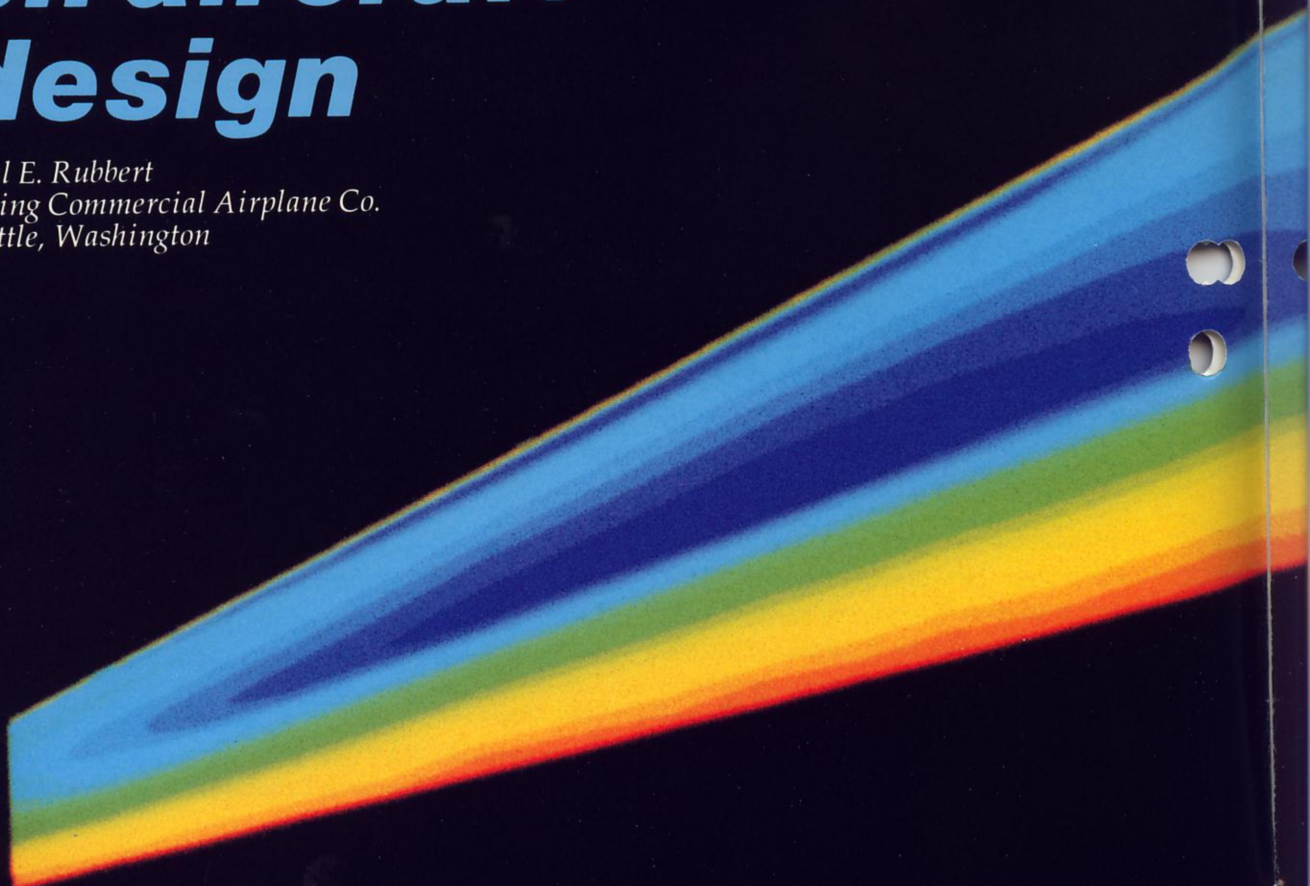
Editor
Carol Smith

Contributing Writer
Ken Jopp

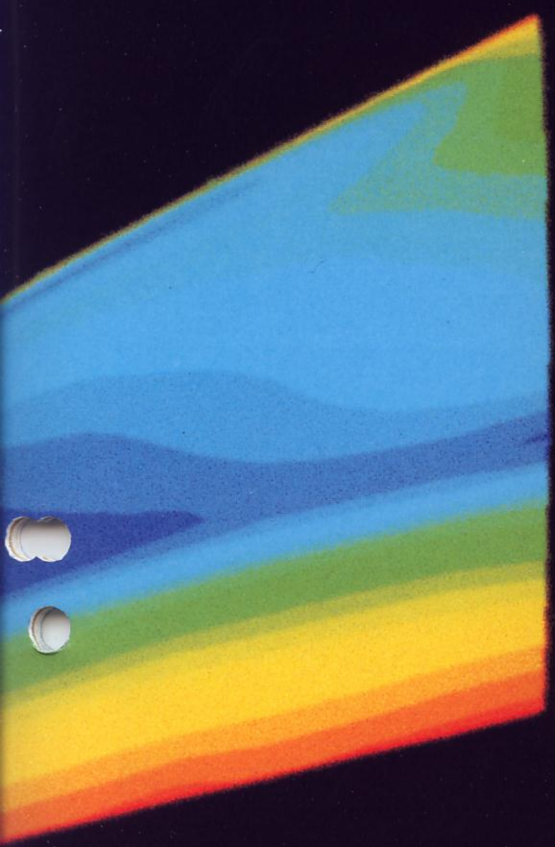
Design
Cindy Erickson

The impact of computational methods on aircraft design

Paul E. Rubbert
Boeing Commercial Airplane Co.
Seattle, Washington



Graphic display of transonic airflow over a transport aircraft's wing computed on a CRAY X-MP. Dark blue indicates highest velocities, red lowest.



The continuing evolution of computational aerodynamics has been slowly but steadily changing the process of aerodynamic design. On a day-to-day basis it is not always easy to recognize the total scope of change that has taken place. But collectively, the changes have been numerous and profound.

Impact on design

From within the commercial transport organization of the Boeing Company, I have been able to observe how computational aerodynamics, or computational fluid dynamics (CFD), has changed the aerodynamic design process. The following paragraphs describe various ways in which its impact has been felt.

CFD has become an integral part of the aerodynamic design process. Today's design processes are very computationally oriented. Standard design approaches use proven CFD tools and there are continuing efforts to learn how the latest computational tools will enter into and change the design process. For an airplane development program today, the dollars invested in CFD design are a significant fraction of those invested in wind tunnel testing. The forecast is for CFD to consume an ever-increasing percentage of total design costs.

Among its effects, CFD has taught us what wind tunnels cannot do. It has opened our eyes to some of the limitations of the wind tunnel as a design tool. The wind tunnel is an excellent means for acquiring global information concerning lift, drag, moments, etc., and for revealing the presence of unanticipated flow phenomena. However, it cannot do inverse design; it does not permit a rapid, sequential design process; and it is very expensive and sometimes impossible to measure flow phenomena in fine detail.

A major use of CFD is to reveal details of fluid flow phenomena that wind tunnels cannot produce. Although CFD is used in many different ways, I have observed that it is mostly used to provide detailed understanding of the flow, such as surface pressure distributions, shock wave locations and strengths, streamline paths, boundary layer behavior, and so on. This information is used to assess the quality and characteristics of a particular flow, to learn about cause and effect relationships in design, and to point the way toward a better design.

Although the computer complements the wind tunnel and changes the character of testing, in most cases it does not replace wind tunnel testing. I find the character of wind tunnel testing to be changed in the direction of design verification with CFD providing the design and providing a priori guidance concerning the need and location for specific flow diagnostic measurements. The wind tunnel uncovers unanticipated physical phenomena that may not have been modeled in a computation. CFD, in turn, is used to aid in diagnosing and understanding certain features of the flow that are observed in the wind tunnel. In summary, wind tunnels turn out to be used for what they can do best, and CFD for what it does best. The roles are strongly complementary.

CFD, in many cases, is not used principally to reduce design costs, but rather to achieve a superior design. The economics of the commercial airplane business is such that the leverage of a superior design leads to large expenditures in the quest for the best. If CFD offers opportunities for a superior design, then it will be heavily used together with wind tunnel testing.

CFD also provides greater opportunity for innovative design. When faced with a configuration concept for which no experimental database exists, computational methods usually can produce a design that will perform well enough in its first wind tunnel entry to allow meaningful evaluation and a clear path toward incremental improvement. Without computational design, first wind tunnel results frequently turn out to be unacceptably bad and with no clear path toward improvement. Hence, CFD can be a very powerful lever in exploring new or innovative design concepts.

Specific examples

I would like to describe in some detail two specific examples where CFD has had a well-defined impact on the design of recent Boeing aircraft. I offer the following discourse to convey a feeling about the way in which computational methods have become an integral part of the design process.

Nacelle installation

CFD did what could not be accomplished in a wind tunnel — close-coupled nacelle installations. For conventional underwing-mounted turbofan engines, it is desirable for the vertical location of the nacelle to be close to the wing. This is particularly important for minimizing ground clearance problems (and hence, landing gear length) associated with the large diameter of modern turbofan engines.

The state of the art achieved through many years of wind tunnel testing concerning the positioning of a nacelle relative to a wing is shown in Figure 1. Designers found that if a nacelle was positioned so close to the wing as to appear above the dotted line in Figure 1, the drag increased to unacceptable levels. The source of this unwanted drag was not made clear by wind tunnel testing, but designers

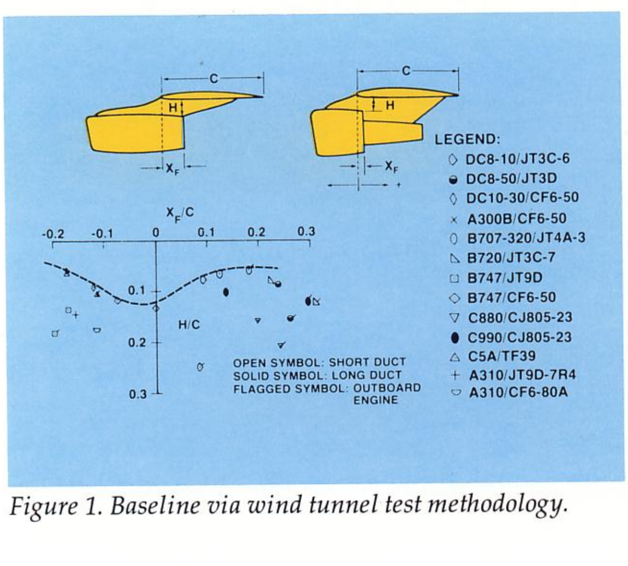


Figure 1. Baseline via wind tunnel test methodology.

coined a name for the mysterious quantity. They called it "interference drag."

A computational attack aimed at understanding and resolving this problem was initiated several years ago. The first step was to calibrate and gain confidence in a computational modeling of the wing/strut/nacelle/plume flow, which took some time and learning. But with that in hand, three different nacelle installations were analyzed and compared with experiment, and good correlation between computed and measured drag was obtained (see Figure 2). The important point is that the computation revealed the source of the interference drag, which the wind tunnel had been unable to do. It was none other than induced or vortex drag caused by a change in wing span loading due to the presence of the nacelle and strut. (One asset of computational methods is that it is usually a straightforward process to individually calculate the induced drag, wave drag, and profile drag. The usual wind tunnel test provides only the total drag).

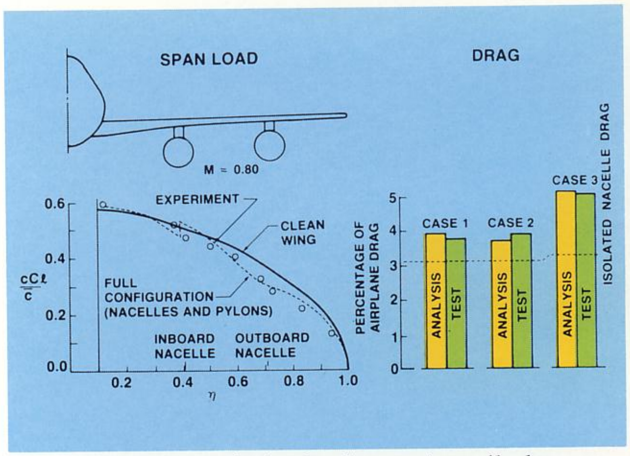


Figure 2. Analysis of wing loading and nacelle drag.

With that information in hand, the design solution became clear; namely to contour the nacelle and strut to prevent adverse impact to the spanwise load distribution of the wing. Properly contoured nacelle and strut design was done much more effectively with computational methods than with the wind tunnel because computational methods automatically produce finely detailed pressure distributions on all wing, strut, and nacelle surfaces. In the process, an added bonus was achieved by carefully redesigning the section shape of the strut to reduce local supervelocity levels that were contributing unnecessarily to profile drag and that could lead to local shock wave formation at the higher Mach numbers.

The knowledge and computational experience thus obtained subsequently were applied to the design of the Boeing 757, 767, 737-300, and KC-135R nacelle installations, enabling very close-coupled installations to be achieved without incurring a significant drag penalty. In this instance, computations allowed the achievement of a configuration goal that was never achieved by 20 years of wind tunnel testing and experimentation.

Identification of high-lift loss

In another instance, CFD was instrumental in solving a high-lift problem. A puzzling and potentially serious problem arose during flight test of a Boeing 707 re-engined with larger diameter CFM 56 engines. The problem was an unexpected loss of 10 percent in maximum lift capability that appeared during flight test but had not been predicted by wind tunnel tests. At first, there appeared to be no obvious experimentally derivable aerodynamic "fix-up" short of extensive full-scale flight experimentation that would be prohibitively expensive.

Subsequent flow visualization work showed that at wind tunnel Reynolds numbers, the maximum-lift characteristics of the wing were dominated by the outboard section characteristics. At flight Reynolds numbers, the outboard wing sections benefited from the increased Reynolds number, and the maximum-lift performance became limited by an unfavorable inboard wing boundary layer/nacelle vortex interaction. Thus, different physical mechanisms were dominating the maximum-lift characteristics at wind tunnel and flight conditions.

Hence, the puzzle was solved, but the problem was not. The traditional approach would be to embark on an expensive and time-consuming flight test fix-up program. However, with the availability of computational tools, a quite different approach became feasible. The approach was to find a way to simulate the full-scale aerodynamics, rather than the full-scale geometry, in the wind tunnel. This required the use of computational tools with design (inverse) capability.

It was a straightforward procedure with computational tools to design an alternative, nonstandard, leading-edge device that would exhibit, at wind tunnel Reynolds numbers, the same leading-edge separation characteristics that occurred on the full-scale wing at flight Reynolds number. This device was fitted to the outboard wing of the wind tunnel model. (Note: at no time was it intended to fit such an alternative leading-edge device to the full-scale wing.) In this way, the outboard wing behaved at wind tunnel Reynolds number very much like the full-scale wing did in flight; that is, nacelle vortex/wing bounday-layer interactions now determined the stall phenomena in the wind tunnel.

Having now radically adjusted the wing's stall patterns in the wind tunnel, attention could turn to possible modifications to improve the maximum-lift performance. A simple fix in the form of a nacelle-mounted vortex control device was found that delayed the stall of the inboard wing associated with the nacelle vortex phenomenon. Thus, no change to the baseline high-lift system of the full-scale airplane was required, and the baseline flight level maximum-lift performance was fully regained.

In this case, computational methods provided an effective approach to solving a problem that could not have been done solely through wind tunnel testing, and it is a very lucid example of the complementary relationship between the wind tunnel and CFD.

Conclusion: the future

The course of future development in CFD will be channeled by several factors. One is computing power, with memory size perhaps outweighing CPU speed in importance. Computing power will continue to determine what can be done in the foreseeable future.

Another factor is the rising need for a closer relationship between algorithm research and computer architecture. We are leaving an era where algorithm researchers merely accept a given machine architecture. Directions in computer architecture are clearly in the direction of multiple processing units (the CRAY X-MP now has four, with future machines expected to have more), but the computer manufacturers are only beginning to discover how algorithm developers will use them. What is needed is a joint attack on the total problem of working together on algorithms and computer architecture. I predict that within a few years, many of us presently involved in algorithm research will be more deeply involved in this entire problem.

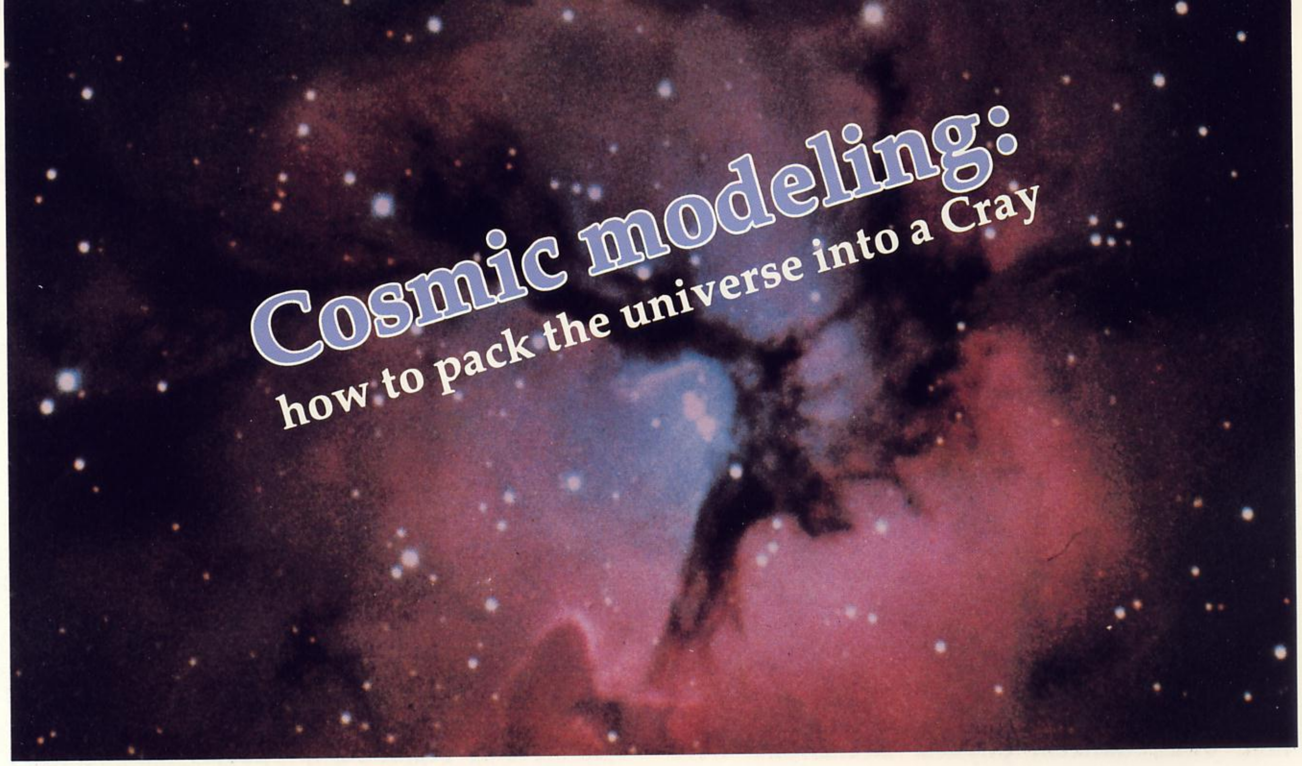
A third factor that presents a major challenge is the user interface, with spatial and surface grid generation being the major component. We must find a way to provide a reliable and easy-to-use solution to the problem of discretizing the computational space. At present, I perceive much activity directed at grid embedding, patching, matching, and overlapping. These are all useful for computing flows about more complex configurations, but they may not lead to practical use by design engineers. I think that we need a sizable breakthrough in this area in addition to continued evolutionary developments. As problems grow more complex, we may see the emergence of "CFD laboratories" that conduct analyses for the design engineer.

And we will experience an evolutionary shift in the direction of more interdisciplinary computational research. The basic skills required to develop computational procedures for solving the partial differential equations of fluid mechanics are the same as those required in other fields. I hope universities take the lead in producing graduates with Ph.D. skills in mathematics and an interest in computing, people who view the computer as an opportunity to put into practice their mathematical skills. □

Acknowledgements

The author would like to thank Ed Tinoco at the Boeing Commercial Airplane Co. for his help in preparing this paper.

This article is a condensation of a paper presented at the AIAA Atmospheric Flight Mechanics Conference, August 15-17, 1983; copyright American Institute of Aeronautics and Astronautics. Condensed and reprinted with permission.



Cosmic modeling: how to pack the universe into a Cray

Most astronomers believe that our universe began as an unimaginably violent explosion of energy. According to this Big Bang cosmology, matter was originally distributed uniformly throughout the universe. However, astronomers know that if this was once the case, it is no longer. Astronomers working today describe a universe where dense lumps of matter, called stars, are separated by vast distances nearly devoid of matter. Stars, in turn, are known to aggregate by the billions into galaxies, which further aggregate into clusters of galaxies. Finally, these clusters are themselves grouped into either long filaments or flat pancakes called superclusters, which can stretch nearly one one-hundredth of the way across the universe and enclose relatively empty, nearly-spherical voids between them. On the scale of these superclusters, the universe has a cellular "Swiss cheese" structure.

Why would matter cooling from the Big Bang, presumably randomly distributed in space, organize itself into the particular structures we observe today? To attempt to answer this question, some astronomers have turned their attention to the world of subatomic particles, and to CRAY computers to simulate particle behavior on a cosmic scale.

Simulating the large-scale structure of the universe's matter requires large computer memory and massive processing power. Typically, a researcher will feed a computer a model consisting of equations that describe how matter interacts, along with the initial positions and velocities of the particles being tracked. Because of the complexity of such models, most simulations of this type include only one type of particle. The computer evolves the model through timesteps, providing three-dimensional "snapshots" of the distribution of the particles. The researchers then scan these pictures for patterns resembling the distribution of matter in the actual universe. When a scene from the simulation is found to match the actual universe, scientists can calculate a time scale for the simulation and determine when significant events in the evolution of the universe occurred.

The problem

During the late 1970s, scientists began simulating the coalescence of the universe's baryonic, or visible, matter. This matter includes relatively large particles such as protons and neutrons. Scientists began their simulations by assuming an initially random distribution of particles that interact only gravitationally. They discovered that under the conditions modeled, baryonic matter would be expected to aggregate into the smaller clumps, such as stars and galaxies, that we see today. But the filament and pancake-shaped superclusters were not predicted by these simulations: the gravity from the universe's visible matter was insufficient to create superclusters on a time scale proportionate to that of the actual universe.

Undaunted, scientists began exploring the possibility that particles of nonbaryonic, or dark, matter might provide the needed gravity. The likeliest candidate was the neutrino, an abundant particle that is believed to have decoupled from the rest of the universe's matter only about 100 seconds after the Big Bang. (In contrast, baryonic matter is believed to have decoupled about a half-million years later.) The early decoupling of neutrinos would have given a neutrino-dominated universe a head start at supercluster formation.

Unfortunately for the theory, neutrinos were missing a vital ingredient — mass. Without mass, neutrinos would have no gravity to draw baryonic matter into clumps. Recently, however, some experimenters claim to have detected a miniscule mass, about 30 electron volts, for neutrinos at rest. These experimental results are tentative and require additional verification, but the neutrino mass reported would provide approximately the right amount of "missing gravity" needed to account for superclusters of galaxies.

Another factor favors neutrinos as the most likely "seeds" to initiate supercluster formation. Calculations show that neutrinos possessing the reported mass would gravitationally form clusters on the same scale — tens of millions of light years — as observed superclusters. Once this had occurred, such clumps would have attracted the baryonic matter. As the baryonic matter (Hydrogen gas) fell into the gravitational potential wells of the neutrinos, its particles would collide with each other, heat and emit visible radiation, thus creating the superclusters we see today.

Recently conducted three-dimensional computer simulations of an early neutrino-dominated universe have yielded some encouraging results. The simulations evolved filament and pancake shapes that correspond to superclusters in the actual universe. Joan Centrella of the University of Texas at Austin, and Adrian Melott of the University of Chicago, have been collaborating since 1982 on the simulation and graphic display of supercluster formation in a neutrino-dominant universe. They have run their simulations on a CRAY 1/S and CRAY X-MP at Lawrence Livermore National Laboratory.

The numerical model

Simulating the movement of every neutrino in the universe over the course of several billion years is a task beyond the powers of even a CRAY. In order to scale the problem down to a manageable size, Centrella and Melott carry out their simulations within a finite computational space. The space they use is a cube corresponding to a cube 300 light years on a side in the actual universe. This computational box is subdivided by a three-dimensional grid, typically of 32x32x32 or 64x64x64 cells. Groups of billions of neutrinos are represented as "clouds," with each cloud equal in size to a grid cell. During a simulation run, the clouds freely move from cell to cell and through each other, since they are modeled as collisionless matter, interacting only gravitationally. Centrella and Melott have been running simulations with anywhere from 100,000 to 1 million clouds. One such run with nearly 200 timesteps requires three hours of computer time, including disk I/O.

Centrella and Melott use a standard numerical method, the Cloud-in-Cell (CIC) method, which involves first assigning a density to the mesh. In the CIC method, the volume of a cloud overlapping a cell determines the mass of the cloud assigned to that cell. A typical cloud with a designated position and velocity will overlap eight grid cells. The more clouds that overlap a cell, the greater the cell's density, thus the greater its gravitational pull on clouds in neighboring cells. Once each cell's density is determined by the total cloud volume overlapping it, its gravitational potential is calculated based on that density. A cell's gravitational potential tells the scientists the force with which it attracts particle clouds to itself.

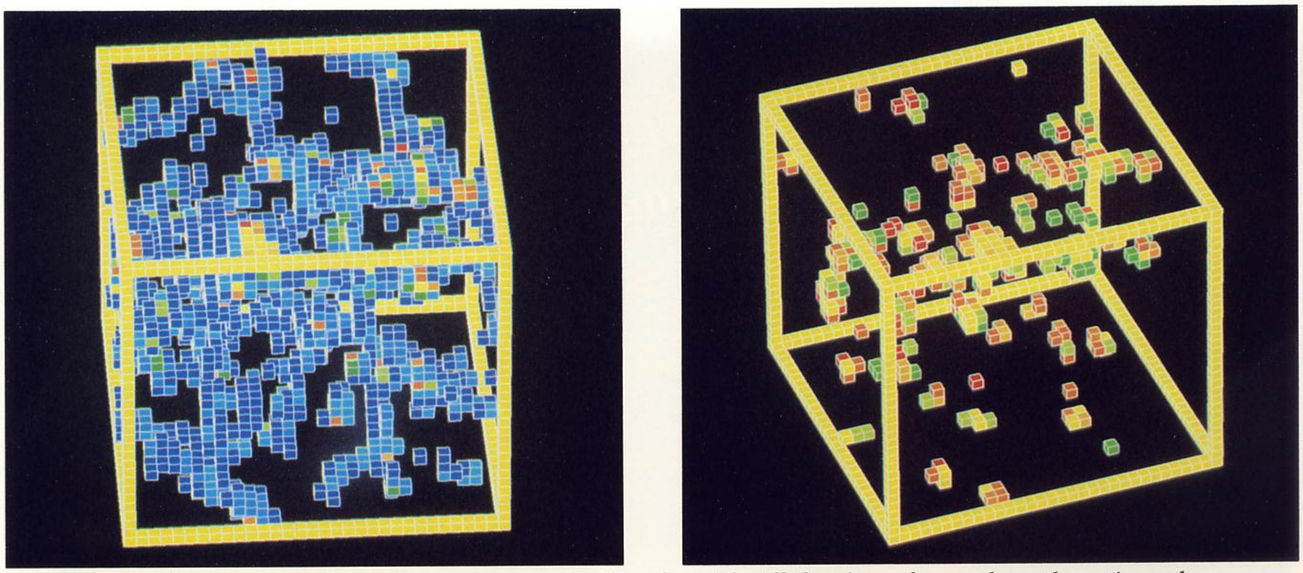
The next step is to "push the particles" using this force. The force on each particle is calculated from the forces in the cells that it overlaps, using the same volume-weighting method used to compute the density. The new density is assigned to the mesh and the cycle is repeated until the model has evolved into a state that can be compared with observations of the actual universe. For stability, timesteps must not only be small compared with the time for the fastest cloud to cross a zone, but also compared with the dynamical time for the densest accumulation of clouds. Since accumulations grow denser as the model evolves, timesteps are decreased in length as the simulation proceeds.

Optimizing the code

Centrella and Melott chose to run their simulations on a CRAY because of its speed and large memory. Adequate resolution of the structures that form during a run requires a minimum of 32 cells along one edge of the computational box. A more detailed study of the structures requires 64 cells per edge. "Those grid sizes translate into minimum central memory requirements of about 70,000 to 600,000 words respectively. Of course, additional memory is needed for particle positions and velocities," explained Joan Centrella. "Such memory requirements make the relatively slower DEC-20 or VAX computers, which contain large virtual memories, a better choice than faster computers, such as the Cyber 175/750, which have much smaller memories. However, these smaller computers are really very slow. In fact, our optimized code runs about 100 times faster on a CRAY with two million words of central memory than it does on a VAX."

Once memory constraints were satisfied, Centrella and Melott set out to optimize their code for the CRAY by taking advantage of the CRAY's vector capability. Particle-mesh calculations like the CIC method naturally break into two parts: the particle-pusher and the gravitational potential solver. The particle-pusher, which updates the coordinates and velocities of the particle clouds, is the most time-consuming part of the code. Therefore, to best increase the code's speed, Centrella and Melott first set out to vectorize the particle-pusher, which accounts for about 75% of all operations performed.

Three methods were used to vectorize this part of the program. These methods involved use of the Cray FORTRAN Compiler (CFT), special vector syntax and vector operations carried out in machine code. CFT automatically vectorized the basic particle-pushing operations. But the periodic boundary conditions, which introduce particles pushed out of the computational box back into the box's opposite side, required substitution of special vector syntax. In addition, the calculations of the density



Models of baryon-dominated universe using a 32x32x32 cube grid. All density values at least three times the average density are shown at left. Note the lack of pancakes or cellular structure. The right image shows density values at least 20 times the average. Note the clumps and lack of filaments. All images generated on a CRAY X-MP using UNIRAS.

on the mesh and the force on any particle required the computation of the subscripts of certain arrays. Since computed subscripts were not supported directly by the hardware vector instruction set on the CRAY, Centrella and Melott used STACKLIBE, a library of vector operations for the CRAY implemented in machine code, written by Francis McMahon at Lawrence Livermore National Laboratory. (This procedure was carried out prior to the release of CFT 1.13, which supports computed subscripts in software, and CFT 1.14, which supports the CRAY X-MP/48 hardware compressed index function.)

An example of an operation requiring the use of special vector syntax is the statement:

IF (X(I) .LT. SBOUND) X(I) = X(I) + N.

which checks to see if the x-coordinate of particle cloud (I) is below the lower bound of the computational box. If it is, the statement puts it back into the top of the box. Periodic boundary conditions such as this assure a constant total mass within the computational box. This consideration derives from the assumption that on scales larger than that of superclusters, matter is evenly distributed throughout the universe.

CFT does not recognize this statement as a vector operation, and the presence of "IF" statements inhibits vectorization of the entire DO loop in which they occur. Centrella explains, "We could place the boundary conditions in a separate unvectorized loop, allowing the basic particle-pushing operation to be vectorized. But we can do better than that by using vector merge operations for the periodic boundary checks." By substituting the vector merge operation:

X(I) = CVMGM(X(I) + N, X(I), X(I) - SBOUND)

for the previous statement, the boundary conditions are recognized as vector operations. In the latter statement the built-in function F = CVMGM(A,B,C) is defined to be F = A if C > 0, and F = B otherwise, with similar functions defined for C < 0 and C = 0.

An example of vectorization requiring the third method, which uses STACKLIBE, was the calculation of the force on a particle. The force is the volume-weighted sum of the forces in the cells the particle overlaps; each partial force is calculated from the gradients of the potential. This procedure involved computing the appropriate subscripts in the potential array and "gathering" these values of the potential for each particle. This "gather" operation was not vectorizable either automatically or by the substitution of vector syntax on the CRAY. (CFT 1.13 supports "gathering" in software and CFT 1.14 supports the X-MP/48 hardware "gather".) But it could be accomplished by calling the STACKLIBE subroutine QVTILS, which performs an operation called "transmit index list."

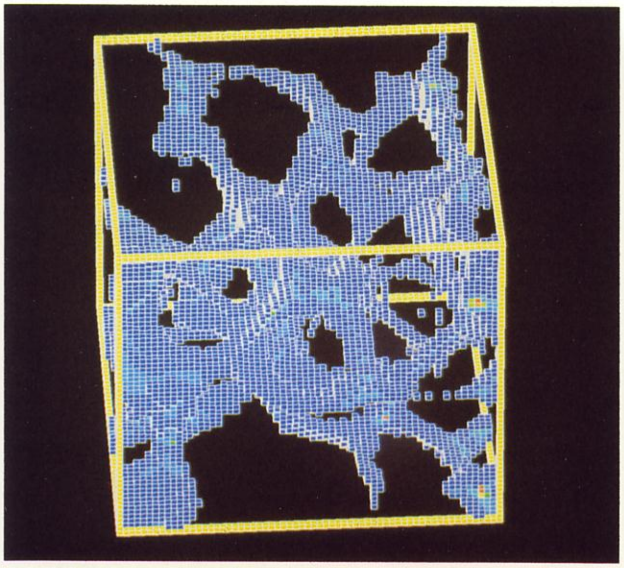
In this operation, an index list array (IL) is first filled with the subscripts of the zones in the potential array that the particle overlaps. QVTILS is then given IL, the potential array A from which the values are to be gathered, and an array B in which they are to be stored. The subroutine is then simply called in the FORTRAN program:

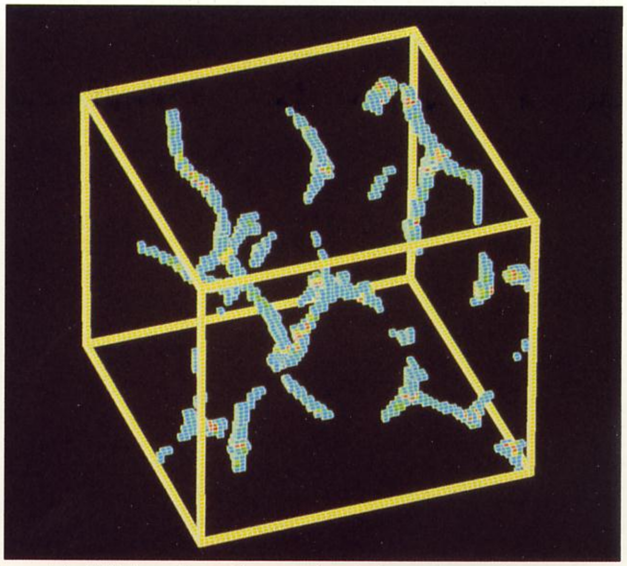
CALL QVTILS(B, IL, A, NI).

Schematically,

B(M) = A(IL(M)).

where M = 1,2, ... NI. Centrella comments, "The power of this method can be seen from the fact that we can take NI = 8NTOT; in other words we can 'gather' the contributions for each of the eight zones involved for all NTOT particles. (NTOT is the total number of particles in the box.) The force components are then calculated by taking the gradients of the potential using the values stored in arrays such as B."





Models of neutrino-dominated universe using a 64x64x64 cube grid. All density values at least three times the average are shown at left. Note the planar structures (interconnected pancakes) and large voids. The image at right shows density values at least 20 times the average. The filament shapes are the "cores" and intersections of the pancakes.

After optimizing the particle-pusher, Centrella and Mellot set out to vectorize the gravitational-potential solver. This involved solving Poisson's equation for gravitational potential on the mesh using a fast Fourier transform (FFT) method, which is computationally efficient. The FFT of a 3-D array is a triple sum over the three subscripts of the array. "We transform each of these subscripts in turn, holding the other two constant," explained Centrella. "For each of these three one-dimensional Fourier transforms we use a CRAY optimized library FFT subroutine. Since the Poisson solver transforms one mesh quantity, the density, into another, the potential, this part of the code was vectorized in a simple way."

"The result of all this optimization is that the code ran three times faster than it did in pure scalar mode," she added. "Since the particle-pusher is dominated by operations involving computed subscripts, most of this speed increase is due to the use of STACKLIBE subroutines. Optimization of the code required about one month of work, but given the speed increase achieved, it was well spent."

The results

The resulting code was run to compare neutrino-dominant and baryon-dominant models of the universe. The simulation results clearly show distinct filament- and pancake-shaped structures evolving in the neutrino-dominant model. By contrast, the baryon-dominant model developed isolated dense clumps and showed no large-scale filaments or pancakes.

At the present time, such results are far from conclusive. Using the CIC model, the question of galaxy formation remains unsolved. If is it assumed that galaxies form from the collapse of the larger pancake structures, then the relevant time scales do not concur. "No one has a consistent theory to explain all the time scales," said Centrella, "but now some people are saying we need to include hypothetical particles like the photino and the gravitino."

Cray analyst Sara Graffunder is currently working on multitasking Centrella and Melott's code to run on the four-CPU CRAY X-MP/48. Centrella explains, "We figure we can increase our speed by a factor of four. We also plan to improve our resolution by dividing our computational box with a grid that is 128 cubes on an edge. By multitasking the code, a run with better resolution may not take much longer than our current code does on the CRAY-1. With this resolution we hope to get a better idea as to the real thickness of the pancakes, which might shed some light on the process of galaxy formation."

Conclusion

Applications in engineering and manufacturing have demonstrated the value of CRAY computers for simulating devices and processes that are too expensive or dangerous to manipulate directly. Scientific researchers like Centrella and Melott have demonstrated the value of CRAY systems for simulating processes that are impossible to manipulate directly. In addition, their research uses numerical methods, and methods for vectorization and graphic display that are widely applicable to problems in industry. At least as important, however, is the promise of their research to significantly deepen our understanding of the origin of our universe and, ultimately, ourselves. □

Acknowledgement

This article includes material from Numerical Astrophysics, *Centrella, LeBlanc, Bowers ed., Jones and Bartlett Publishers, Inc., Boston, 1985.*

Image processing overview

Combine the innate limitations of the human eye with the corresponding limitations of cameras and other imaging technology, and the most information-rich imagery can be rendered unrecognizable. But translate such imagery into digital code, and computer intervention can be used to provide clarity. This, in the simplest terms, is the theory behind the development of image processing methods and applications.

Image processing techniques are extensively applied in remote sensing and medicine. Remote sensing involves gathering information at a distance via satellites or space probes. Perhaps the most familiar example of image processing applied to remote sensing is the processing of information beamed down from Landsat earth-imaging resource satellites. Landsat information is routinely used to assess crop yields, aid in urban planning and detect likely locations of natural resources, such as petroleum and mineral deposits. CRAY computers are ideal for many image processing applications. This article will concentrate on examples using Landsat data.

In general, image processing can be divided into three phases. First, an image is acquired and converted into digital code. This can be done with real life images via digital imaging sensors, or with images already stored on other media such as film or video tape. Second, the digitized image is processed through a computer to remove distortion or to enhance features of particular interest to the human observer. Finally, the processed digital representation is reconverted to analog form for viewing.

Methods used to process images are of two general types, subjective and quantitative. Subjective processing, also called image enhancement, is intended to improve the ability of a human viewer to interpret an image. Subjective processing techniques develop out of trial-and-error approaches that depend on adaptive, iterative and interactive techniques. They also often incorporate information about the physiology of human vision. The success of subjective processing techniques is based on the human viewer's ability to discern information of interest in the processed image.

Quantitative methods, by contrast, are intended to reduce imaging-system-induced distortion or to reproject an image according to a more useful geometry. An example of the latter is the mapping of remotely sensed imagery onto standard cartographic projections. Quantitative processing methods are based on predefined algorithms and their success is determined by the accuracy of the mathematical models on which they are based. Included in both quantitative and subjective methods are several techniques, with new ones being developed as necessity requires. Several techniques of each type will be described.

Subjective techniques

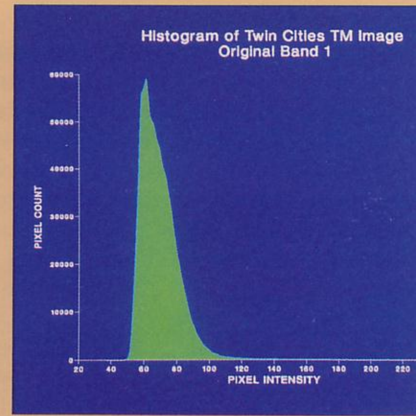
Linear contrast enhancement
To make an image useful to a human viewer, it is often necessary to manipulate its contrast, or the range of intensities from light to dark. The human eye can discern far fewer distinct shades of gray than digital imaging systems are capable of encoding. Therefore, when a digitized image is displayed, its many distinct shades of gray must be subdivided into a small number of subsets.

This process works well for many applications. However, if the digitized image has an intensity distribution such that most of the pixels fall within only a few of the subsets, then the displayed image may be of such low contrast that most of the digitized information is concealed.

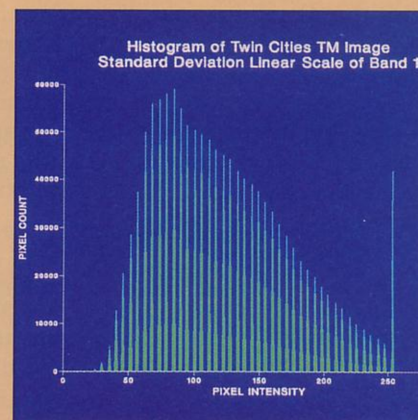
This problem can be solved by applying linear contrast enhancement, also called "contrast stretch" since it takes a range of intensities confined to a few subsets and maps it onto the full range available to the film or video display terminal. This is done by reassigning to black all of the pixels at or below the lowest value of that subset. Pixels at or above the highest value are reassigned to white. This allows the processor enough room to "stretch" the intensities within the subset over the entire range of values offered by the film or video display.

Nonlinear contrast enhancement
It is sometimes desirable to perform contrast enhancement in a nonlinear manner, as when a situation demands that a processed image be of a particular aesthetic quality. Nonlinear contrast enhancement, also called histogram equalization, is an application to which CRAY computers are particularly well suited due to the nonlinearity of the process. Figure 1 shows the effects of linear and nonlinear contrast enhancement applied to a Landsat image of Minneapolis and St. Paul, Minnesota.

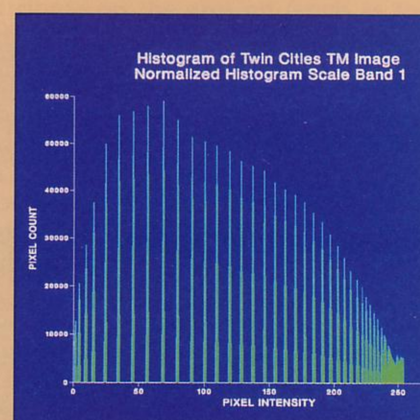
Original Landsat image from spectral band 1 and corresponding histogram showing distribution of pixels from dark to light. Note concentration of pixels at dark end of intensity scale.



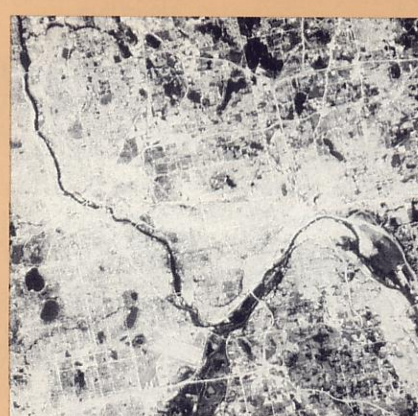
Result when same image is processed with a linear contrast enhancement, or contrast stretch, based on the standard deviation of the pixel distribution. The histogram shows the corresponding distribution of pixels.



The same image processed with a nonlinear contrast enhancement, or histogram equalization. This algorithm attempts to spread the pixel distribution over all intensity ranges in roughly equal proportion, as seen in the histogram.



A more drastic histogram equalization was applied here, which used a random number generator to fill in the discontinuities between the pixel distributions.



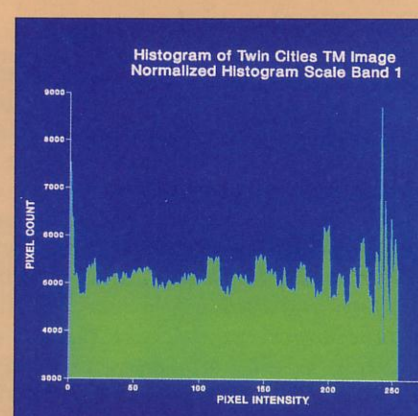


Figure 1.

Image contouring and edge detection

Another useful enhancement technique is image contouring, a manipulation used to display contours of equal intensity within a video image. This is important in radiometric analysis where considerations of albedo and shading must be taken into account. A popular method used to perform image contouring is called "bit-clipping." Using this method, an analyst "clips," or sets to zero, a certain number of the most significant bits of each digitized pixel intensity value. The effect is to break up a region of gradual black-to-white transition into several subregions having the same intensity of black-to-white gradation. But bit-clipping often renders the output image detail unrecognizable. An alternative method of image contouring that avoids this drawback is to simply designate that certain intensity values be mapped as either black or white. This method can be used to superimpose contours on an image at any given interval of digital intensity. It has the advantage of retaining most of the input data since only those pixels at the specified contour values are modified, thus minimizing image distortion.

Like image contouring, edge detection is an enhancement technique used to discern boundaries between areas having different textures. For example, edge detection applied to Landsat data can reveal roads, building boundaries and agricultural crop boundaries, information that has applications in fields such as cartography.

Edge detection typically involves scanning the digitized lines of an image with a 3x3-pixel "window". At each pass over the image, the edge detection algorithm computes the contrast gradient among the nine pixels comprising the window to produce a new value for the output image. A demonstration image processed with an edge detection algorithm is shown in Figure 2. The edge detection algorithm used for this example took the high (white) value of a contrast gradient and assigned it to the output image. Such a process can be computationally intensive when dealing with large amounts of data, since the algorithm must weigh and sum the nine window-pixel values for each pixel in the output image.

Quantitative techniques

Unlike the subjective image processing techniques described above, quantitative techniques are based on particular mathematical models. An example would be an algorithm to remove distortion introduced by the fixed shading pattern of a film scanner. Such a pattern is undesirable and can be removed by mathematically modeling it and writing a program to remove it from each scanned image. In this case, success in processing is based on the accuracy of the mathematical model, not on the evaluation of an analyst, as with subjective techniques. Other quantitative techniques include geometric transformation and multispectral classification.

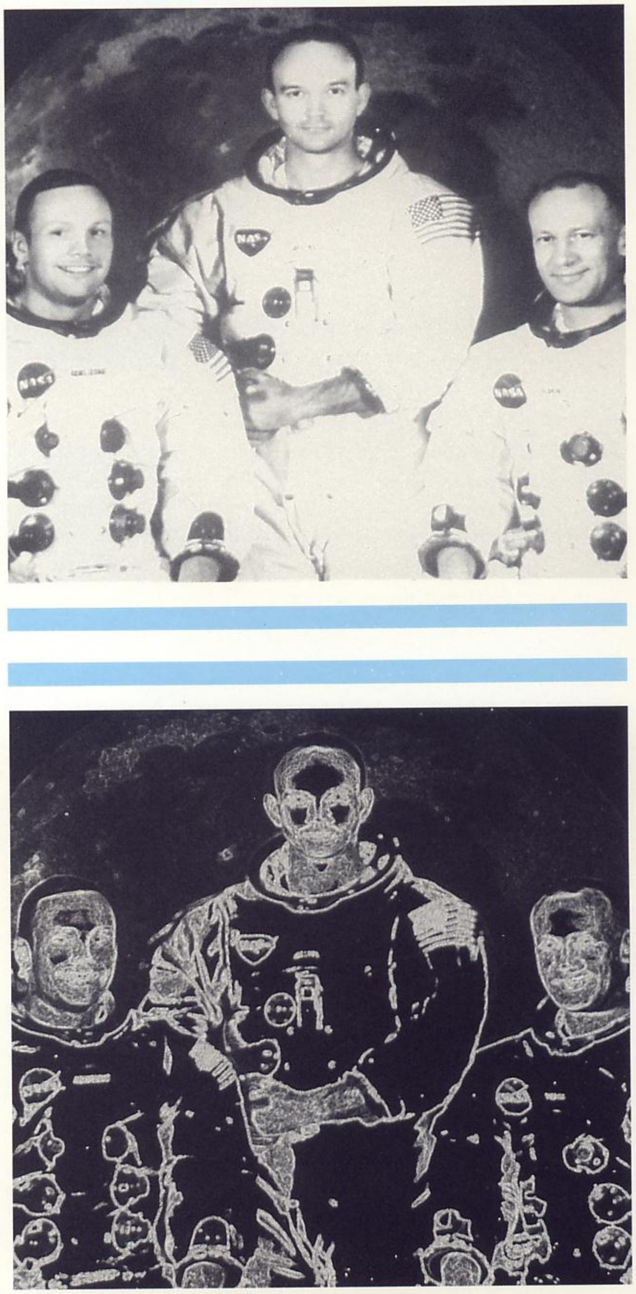


Figure 2. Original image of Apollo astronauts and the resulting image when processed with an edge detection algorithm.

Geometric transformation

Geometric transformation modifies the size and shape of an image. This technique is sometimes called "rubber sheet transformation" because the effect is similar to applying an image to a rubber sheet and stretching it into a particular geometric reference frame.

Common types of geometric transformations include standard mapping projections such as orthographic, Mercator and polar stereographic projection. In addition, there are two mapping projections that have been developed specifically for Landsat

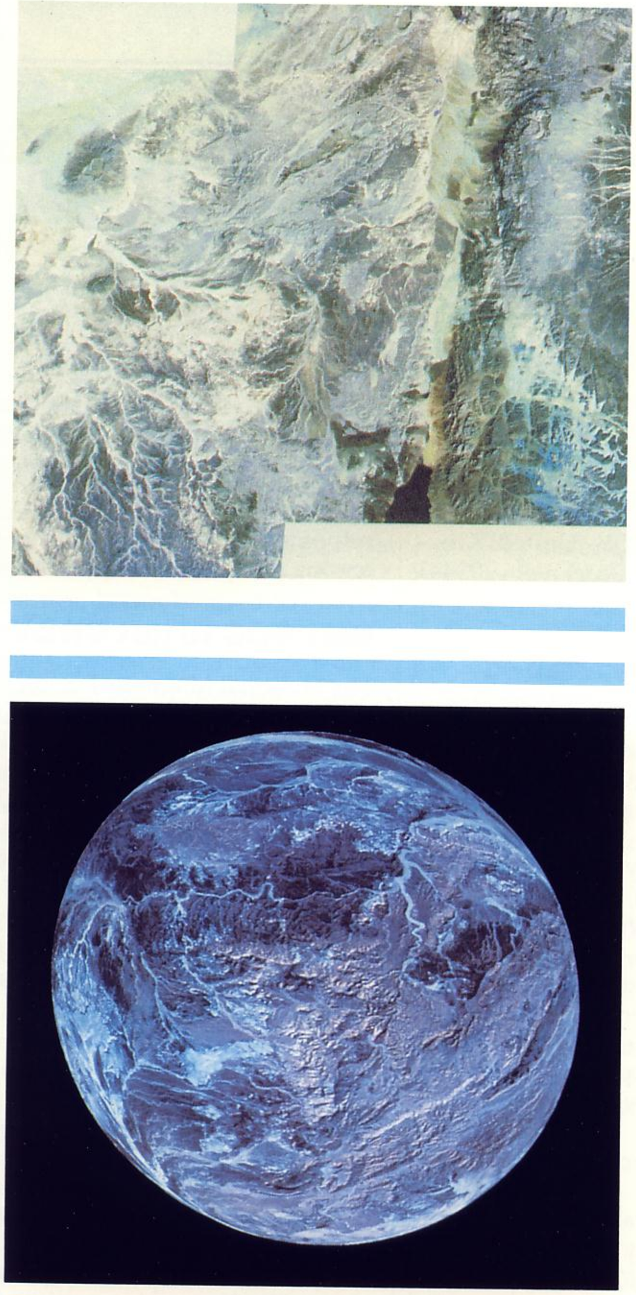


Figure 3. Landsat data from Sinai desert and the same data projected onto a sphere. © 1984 Geometric Productions, Berkeley, California.

imagery that do not correspond to any standard cartographic projections. Analysts who wish to use cartographically projected Landsat data must become familiar with the details of these projections.

A unique example of cartographic projection carried out on a CRAY is a simulated planet that was created by projecting two-dimensional Landsat data onto a sphere. This is illustrated in Figure 3. A tiling process was used to divide a Landsat image of the Sinai desert into a number of subpictures. This data was stored in the CRAY Solid-state Storage Device (SSD). The algorithm that carried out the cartographic projection calculated the location of the data to be accessed from the SSD by raytracing each point on the planet back to where the corresponding data would be on the flat tiled Landsat image. Use of the SSD enabled this process to be done at least 30 times faster than would have been possible using a traditional storage disk.

Multispectral classification

Another important quantitative image processing technique often applied to Landsat data is multispectral classification. This involves classifying materials in a scene according to their relative spectral properties. A multispectral classification of an image is performed by analyzing each pixel and determining whether the spectral response exhibited by that pixel corresponds to the spectral response of a known material. The spectral response is the relative intensity of each spectral band that the imaged material reflects. The accuracy of spectral classification algorithms can be checked against actual inventories of materials covering the area being imaged.

In a simple case of multispectral classification, a one-color image can be made by taking a weighted sum of products from the seven Landsat bands. This means that the amount a given band will contribute to the image will be determined by how heavily that band is weighted. An observer will often weight a band according to the value of its data to that observer. Once a formula of weights and functions for processing an image is determined that measure some true value or attribute on the ground (such as the chlorophyll content of a lake) it can be used repeatedly with new Landsat data to make measurements over vast areas on the ground. With a high-speed computer, such as the CRAY, these methods can be used to do a quick interactive look over the data in minutes.

Processed Landsat data provides examples of many types of image processing methods and applications. But image processing methods are also incorporated into imaging technology used in medicine, such as computerized tomography and nuclear magnetic resonance imaging. Promising applications for the future include binocular vision for robots, which would increase their versatility by giving them depth perception, and combining image processing with computer graphics for motion picture special effects. These and yet unforeseen developments will continue to expand the range of image processing applications for high-speed scientific computers. □

Acknowledgements

Special thanks to Bill Samayoa at Cray Research for his help in preparing this article.

This article includes material from **Digital Image Processing, A Systems Approach**, *by William B. Green. Copyright 1983, by Van Nostrand Reinhold Co., Inc. Reprinted by permission of the publisher.*

CORPORATE REGISTER

Bracknell Data Centre goes live

To bring supercomputer power to a wider audience in the United Kingdom, Cray U.K. initiated construction of a data center to provide facilities for software development, training support for new employees as well as offering a supercomputer service to outside users. The result was seen October 1, when the Bracknell Data Centre came to life.

The Centre houses a CRAY X-MP/12 as well as IBM and VAX front-end computers. "We knew we wanted the CRAY, IBM and VAX equipment for software development," explained Nick Edmunds, Data Centre Manager. "These are also the primary front-ends used by our typical customers."

"Selling computer time and services to scientists and customers throughout the United Kingdom will help potential Cray customers to take smaller steps in getting their applications running on the CRAY. With our support, they can build their use until they are comfortable with the CRAY," said Edmunds.

Most service bureau customers will use the CRAY remotely and lines between the Data Centre and each customer's computer center will be installed as needed. For more information on the Bracknell Data Centre, contact Nick Edmunds, Cray Research (UK) Ltd., Cray House, London Road, Bracknell, Berkshire RG12 2SY, United Kingdom, telephone Bracknell (0344) 485971.

Oil producers expand CRAY use

Three consecutive months recently saw orders for CRAY computers from oil producing companies.

In August, Cray announced the order of a CRAY X-MP/12 by Sun Company. The system is scheduled for installation in the fourth quarter of 1984 at Sun's Richardson Computer Center in Richardson, Texas. The system will be used largely by Sun Exploration and Production Company, a Sun Company unit, for oilfield reservoir simulation and seismic activities.

In September, Cray announced that EXXON Company USA also ordered a CRAY X-MP/12. The system is scheduled for installation in the first quarter of 1985 at the company's Exploration Data Processing Center in Houston, Texas. It will be used for exploration data processing.

The first order for a four-processor CRAY X-MP in the petroleum industry was announced by Cray in October. The system, a CRAY X-MP/48, was ordered by Chevron Oil Field Research Company of La Habra, California, and is scheduled for installation during the second quarter of 1985. Chevron plans to use the system for research and development in exploration data processing and petroleum engineering.

NMFECC to get CRAY

The National Magnetic Fusion Energy Computer Center (NMFECC) recently ordered a CRAY X-MP/22. The system, which will be leased, will be installed at the NMFECC in Livermore, California, in the fourth quarter of 1984, joining two CRAYs already installed. The NMFECC provides large-scale computing services to researchers in magnetic fusion and energy research at 75 laboratories and universities across the nation. The Center received its first Cray in 1978.

Los Alamos acquires sixth CRAY

Los Alamos National Laboratory (LANL) recently expanded its computing facilities by acquiring its sixth CRAY supercomputer, Cray Research announced in September. LANL's latest CRAY acquisition is a CRAY X-MP/24 with a Solid-state Storage Device that will be used for scientific and engineering research and development.

Navy orders CRAY

The Naval Research Laboratory (NRL) in Washington, D.C. ordered a CRAY X-MP/12 to be installed in the first quarter of 1985. NRL is operated by the U.S. Department of the Navy and will use the CRAY in a wide range of research projects.

APPLICATIONS IN DEPTH

Mechanical simulation codes available on CRAY

DRAM (Dynamic Response of Articulated Machinery) is a two-dimensional mechanical analysis program now available on CRAY machines. It can be used to analyze motion such as that of an aircraft landing gear, robot arm, or similar device driven by consistent input forces. It can also be used to study the structural deformation of automobiles, machine tools, or other complex mechanical systems, as well as for unusual operating conditions where input forces are erratic.

DRAM determines static equilibrium and time response (displacements, velocities, accelerations and reaction forces) of planar, multi-freedom, rigid body mechanical systems (machinery, vehicles) which perform through large displacement. DRAM requires as data only a minimal definition of the mechanical system. It proceeds from this data to develop and numerically evaluate the system equations of motion, then reports the results as graphic terminal displays or printout summaries. The input language provided is mnemonic and free-form. Extensive diagnostics warn the user of input errors.

Some current applications of DRAM are the analysis of joint loads on suspension and chassis components for finite-element analysis, simulation of nuclear reactor handling mechanism dynamics and determination of bearing forces in internal combustion engines. In one case, a computer simulation of a train crash using DRAM was admitted as evidence in a court of law.

DRAM incorporates several features to account for the behavior of realistic machinery and vehicles. These are:

- Representation of multi-degree-of-freedom (dynamic), constrained or unconstrained systems, including zero degree-of-freedom (kinematic) systems as a subset.
- A library of force elements such as springs and dampers. Included in this library are models of impact and Coulomb friction.
- A library of motion generators (ideal motors) that describe time-dependent motion of various elements.
- User-specified force effects permitting representation of unusual or nonlinear applied forces.
- Surface-to-surface (higher-pair) contact.

Persons interested in additional information about DRAM, or the three-dimensional mechanical systems simulation program ADAMS (see CRAY CHANNELS, Vol. 6, No. 2) on the CRAY should contact: Mechanical Dynamics, Inc., 3055 Plymouth Road, Ann Arbor, MI, 48105; telephone: (313) 994-3800.

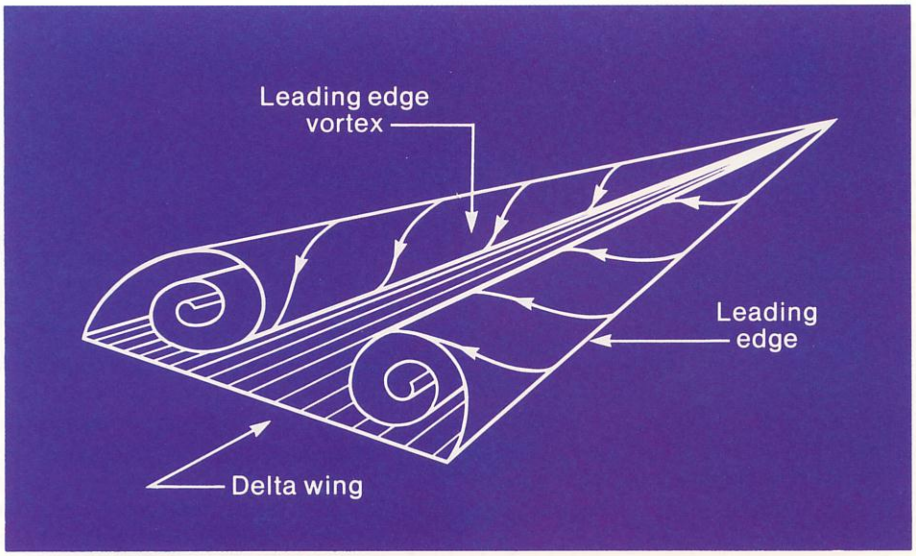
Another mechanical systems simulation program available on the CRAY is the Integrated Mechanisms Program (IMP). The IMP system is intended for the simulation of two- or three-dimensional rigid link mechanical systems having single or multiple degrees of freedom. The simulation can include revolute (pinned), prismatic (sliding), screw, spur gear, cylindric, universal, spheric (ball and socket), and planar joints in any closed-loop combinations. Linear or nonlinear springs and viscous dampers may also be included, either within joints or acting between specified points on the moving links. Mass and gravity effects can be simulated. The system can be driven either by applied forces or input motions which can be specified functions of time or system geometry.

The IMP system is capable of simulation in any of three different modes: kinematic (geometric), static (equilibrium), or dynamic (time response) mode. In any of these modes, IMP will calculate the requested positions, velocities, accelerations, static and dynamic constraint forces, natural frequencies, damping ratios, and small oscillation transfer functions (principal vibration modes) of the system simulated.

For more information on IMP on the CRAY contact: JML Research, Inc., 5713 Crabapple Lane, Madison, WI, 53711; telephone: (608) 274-2524.

SSD allows CFD breakthrough

Analysts at Cray Research recently set a milestone in computational fluid dynamics (CFD). Using a single processor of the CRAY X-MP/48 with 8 MWords of memory and a 128-MWord Cray Solid-state Storage Device (SSD), analysts Jef Dawson and Kent Misegades successfully ran a finite volume analysis on an airplane



Leading edge vortices on a delta wing.

wing using 2.5 million grid points, about 50 times the number routinely used.

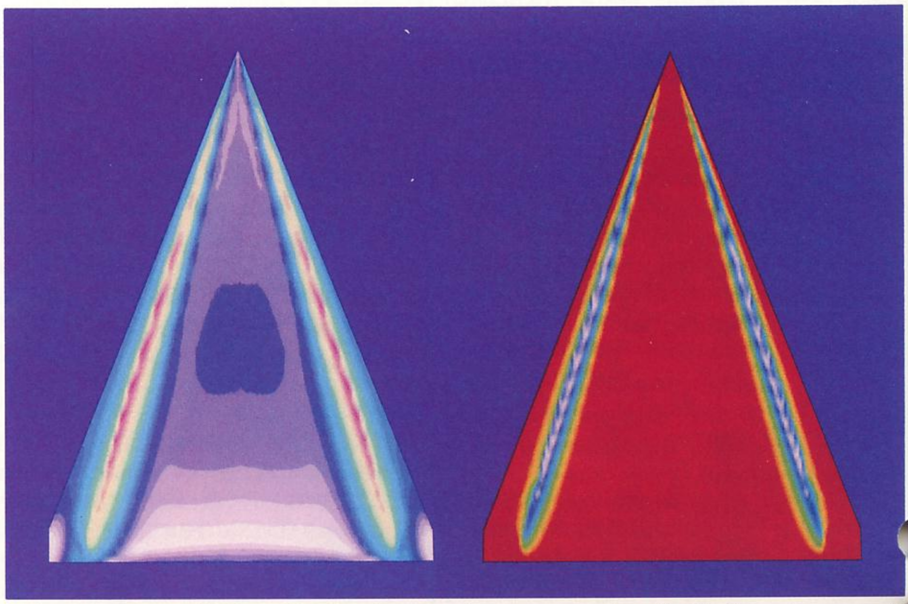
The wing analyzed was a delta wing operating at a high angle of attack (15 degrees) at transonic speed (free-stream Mach number = .7). At such angles of attack, delta wings produce leading edge vortices. These vortices cause local regions of high velocity across the wing's upper surface, resulting in lower pressure in these regions. These low pressure regions significantly contribute to the lift produced by the wing. The accurate simulation of such complicated flows demands a very fine computational grid, which requires the large data storage capacity now available with the CRAY X-MP/48 and the 128-MWord SSD.

Dawson and Stephen Leicher of Dornier GmbH in West Germany performed the analysis using a code that solves the Euler equations of steady three-dimensional inviscid fluid motion. The code was written by Dr. Antony Jameson of Princeton University and modified for out-of-memory cases by Dawson and Leicher. The modified code allows the user to control the amount of data that is in memory and the amount that is out-of-memory (on hard disks or the SSD). In the past, it was not feasible to store large quantities of data out-of-memory, because I/O time was excessive. However, in this analysis, using the SSD for out-of-memory data, I/O time was less than five percent of CPU time, even though 16 times as much data was on the SSD as was in central memory. This result demonstrates that the large SSD memory, which is connected to the main memory by two 1000-Mbyte-per-second channels, is a viable alternative to extremely large main memories for large CFD problems.

"Running the code the way we did isn't difficult," said Misegades. "The SSD looks like a disk to the program since it uses the same job control statement. This means the code doesn't need any modifications to move from disk to the SSD, so our method is completely portable."

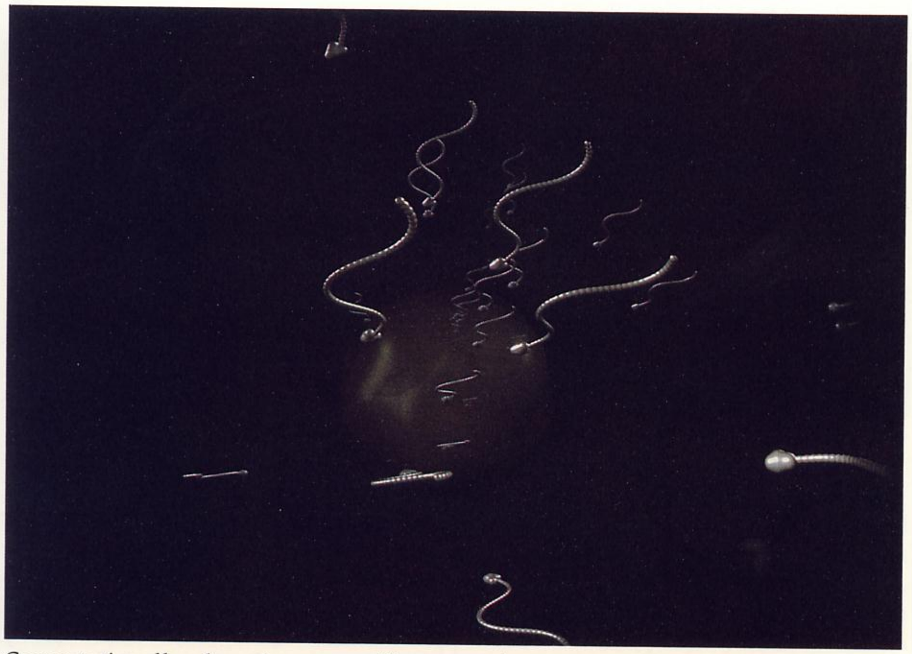
"Being able to use the 128-MWord SSD without incurring a big I/O penalty is very important," commented Dawson. "Not only did it let us use more than enough points to analyze this flow problem, but it also opens the possibility of solving the full Navier-Stokes equations for some three-dimensional geometries." The Navier-Stokes equations model fluid flow more accurately than the Euler equations by including viscous effects, but they require more grid points to solve a problem with the same numerical accuracy.

The graphics software package MOVIE.BYU was used to color-code the calculated pressure coefficient and Mach number distributes on the upper surface of the wing.



Left, pressure coefficient distribution on wing upper surface (red is lowest pressure). Right, Supersonic Mach number distribution on wing upper surface (white is highest Mach number).

USER NEWS



Computationally shaped sperm cells approach an ovum in this scene from The Final Frontier. *© Alan Barr, California Institute of Technology and Gray Lorig, Rensselaer Polytechnic Institute, 1984.*

Cell fertilization on the CRAY

In what looks like a scene from the latest animated science fiction saga, a fleet of sperm are seen swimming toward a distant ovum. Although the scene isn't from a work of fiction, it isn't quite real either. The sperm and ovum are the stars of a CRAY-generated film sequence, *The Final Frontier*. The film was made to graphically show that the evolution of the head shape of swimming sperm was determined by the laws of fluid dynamics.

The marriage of fluid dynamics and microbiology à la supercomputer was performed by Al Barr of the Computer Science Department at The California Institute of Technology and Gray Lorig of The Center for Interactive Computer Graphics at Rensselaer Polytechnic Institute. The relationship between sperm head shape and fluid dynamics was the topic of Barr's Ph.D. thesis in Applied Mathematics, and *The Final Frontier* developed out of that research.

Using equations governing fluid dynamics, Barr calculated the optimum head shape for swimming sperm. He then compared the optimum shape with the shape of actual swimming sperm heads, and found them to match. From this, Barr concluded that fluid dynamics was the principal force governing the evolution of sperm head shapes.

In creating the film, Barr used fluid dynamic principles to calculate the shape, swimming motion, and trajectory of each sperm cell. Gray Lorig developed the program to render the objects on the CRAY and to store the images on magnetic tape so that they could be photographed frame by frame on the Dicomed film recorder. The sperm and egg cells were constructed from the "blobby" modeling primitives recently created by Dr. James F. Blinn at the Jet Propulsion Laboratory. These primitives are used as mathematical building blocks that meld together, creating diverse three-dimensional forms with smooth boundaries

The 30 second film sequence shows the progression of sperm cells with the mathematically-derived head shapes swimming toward an ovum. The sequence was generated entirely on a CRAY X-MP at Cray's Mendota Heights, Minnesota facility. Each frame of the film required about five minutes of computer time to generate, at a resolution of approximately 1500 by 2000 pixels. *The Final Frontier* was one of 18 computer-generated film sequences specially prepared on Omnimax film for presentation at SIGGRAPH '84, held in July in Minneapolis, Minnesota.

New nacelles shaped computationally

Airplane designers continue to reap the benefits of computational aerodynamics. The Boeing Commercial Airplane Company recently used the resources of its computer services center, which currently houses a CRAY-1 and a CRAY-1/S, to solve an unusual design problem it faced in modifying its 737 commercial airplane. The modified version, the 737-300, owes its successful design in part to high speed computing that reduced wind tunnel testing time and gave engineers the confidence to proceed with an unorthodox design.

The 737 is a twin-engine plane that Boeing has been manufacturing since 1965. When Boeing decided to replace the plane's engines with larger, quieter ones, design engineers were faced with an unusual challenge. Because the 737's wings are relatively low to the ground, engineers had to design a nacelle (engine housing) that would accommodate the larger engines, but still provide sufficient ground clearance. "The new nacelle design had to be flattened on the bottom, and this shape had never been tried before." commented Walt Gillette, supervisor of aerodynamics for the 737-300. "Plus, for competitive reasons, we were working on a tight timetable."

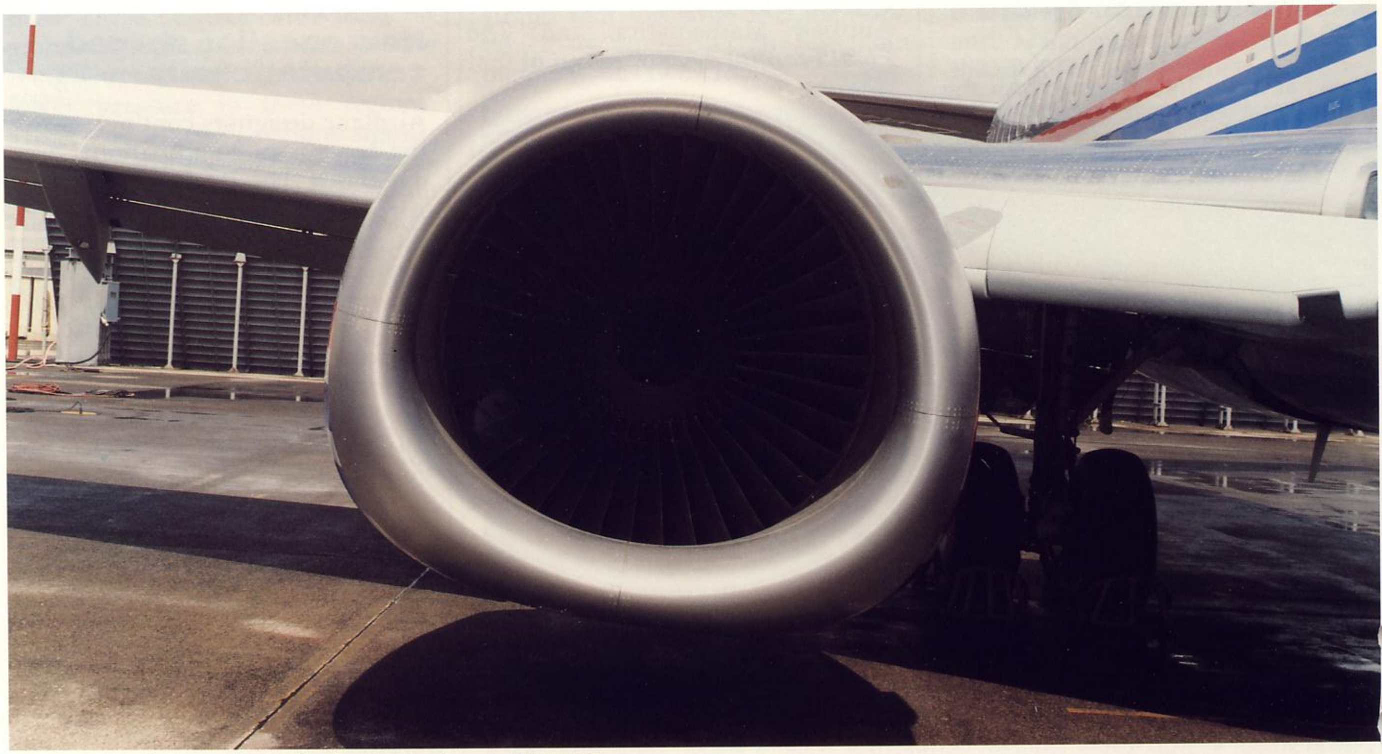
Computational methods allowed the Boeing engineers to test and modify proposed designs quickly. "Traditionally, we would develop new designs by doing some limited analytical work, then run wind tunnel tests and use the results to refine the design, then re-test, and so forth," explained Gillette. "This iterative process was very successful, but it was also very time consuming. Using the computational methods that high-speed computers like the CRAY give us access to, we can now do a more exact analysis from the beginning, and this significantly reduces the number of iterations we need to go through."

The wind tunnel simulation that was used to test the new nacelle design incorporated nearly the entire plane, including its nacelles, wings, wing flaps, fuselage and exhaust from the engine. A typical simulation involved 3000 to 4000 boundary conditions. "The pressure distributions predicted computationally were accurate enough for us to make detailed design decisions," commented Gillette. "Our computational results gave us the confidence to proceed with the unusual shape, and speeded up production considerably compared with the time wind tunnel testing alone would have taken." (Nacelle positioning relative to the wing on the 737-300 was also arrived at computationally. See this issue's related article, beginning on page 2.)

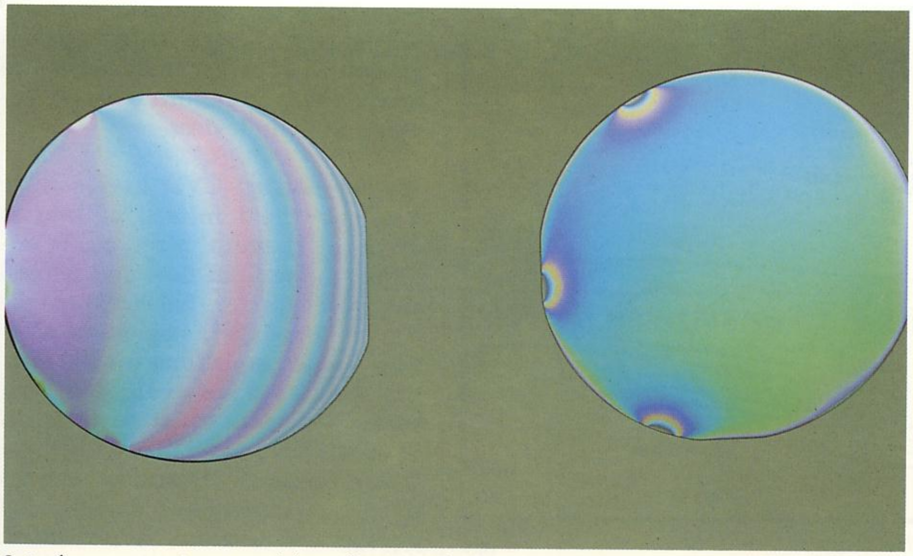
The 737-300 was certified for commercial use in November, 1984. No doubt few of the airline passengers who fly in it will appreciate that high-speed computing was instrumental in allowing this airplane to reach the market.

U of M CRAY models CVD

Chemical vapor deposition (CVD) is a process used extensively in the semiconductor manufacturing industry. CVD techniques attempt to create a thin uniform solid film on a substrate by exposing the substrate to gaseous chemical reactants. As applied to semiconductor manufacturing, it is commonly used to grow a variety of thin films such as polycrystalline silicon on silicon wafers and gallium aluminum arsenide on gallium arsenide wafers. Virtually all major electronics companies manufacturing integrated circuits for use in computers use CVD techniques during several key manufacturing steps. But despite CVD's commercial success, knowledge of its underlying physical and chemical processes is incomplete. Some researchers in the University of Minnesota's Chemical Engineering and Materials Science Department are working to understand these processes for the purpose of



Unique flat-bottomed nacelle for Boeing's new 737-300.



Interference patterns arising from radially nonuniform silicon dioxide film thickness.

improving the design and operation of CVD reaction chambers. The CRAY-1 installed at the University makes feasible the modeling of the complex fluid flow patterns and chemical reactions found in CVD reactors.

The reactors being modeled at the University are of two main types: 1) hot wall, low pressure (LPCVD) chambers and 2) cold wall chambers, which operate at atmospheric pressure. Both rely on the flow and diffusion of gases within the chamber to transport the reactant to the substrate.

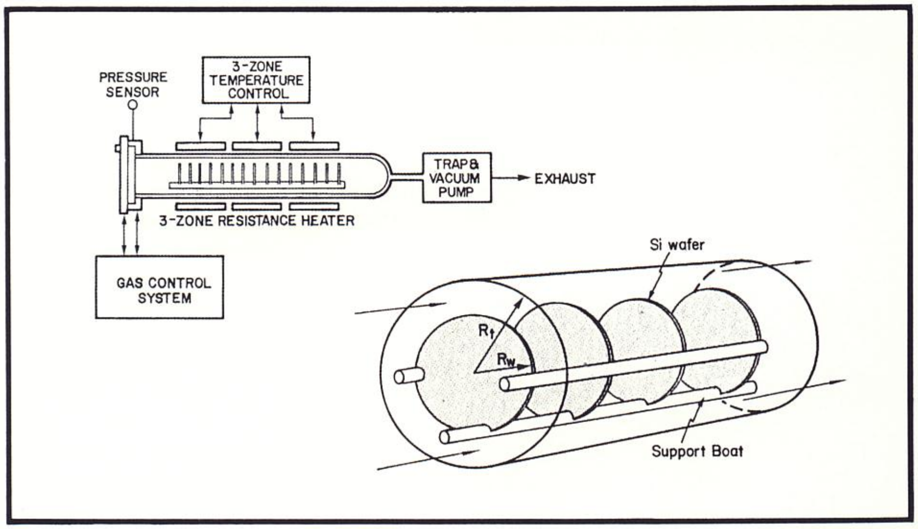
LPCVD chambers have diffusion coefficients three orders of magnitude larger than at atmosphere pressure, resulting in transport 1000 times faster than in cold wall reactors. An LPCVD chamber's hot walls also minimize convection currents that occur within a chamber because of the temperature gradient between a chamber's heat source and its walls. Convection currents are generated as warm fluid inside the chamber rises, and the resulting secondary flow interferes with the uniform distribution of the gaseous reactant over the substrate. The net result is layers of uneven thickness being deposited on the wafer, so that the final device properties will vary across the wafer, if it is even feasible to build devices. Hot wall, LPCVD reactors serve as the main CVD tool for polycrystalline silicon and dielectric films in silicon integrated circuit manufacture.

Cold wall reactors are used primarily for growing epitaxial, or single crystal, layers of silicon and compound semiconductors such as gallium arsenide. The cooled walls are needed to minimize defects in the epitaxial film. They inhibit the growth of film on the reactor's internal surfaces which might flake, depositing material on the device, which, in turn, would interfere with the growth of the epitaxial layer. Convection currents inside cold wall CVD reaction chambers are unavoidable, however, since the substrate must be heated to drive the chemical reactions that bond the reactant to the substrate. Flow patterns in a CVD reaction chamber are further complicated by the interaction of convection currents with the gas jet introducing the gaseous reactant.

Realistic modeling of CVD reactors requires accurate descriptions of fluid flow, heat and mass transport and chemical reactions. Since the chamber is three-dimensional and the film growth involves several chemical species, CVD models consist of large numbers of nonlinear partial differential equations. These are solved by finite element and orthogonal collocation techniques including Newton iteration. These procedures require solving numerous nonlinear algebraic equations — a task for which the CRAY-1 is ideally suited.

Chemical engineering professor Klavs F. Jensen and graduate students working with him are developing models for several types of CVD reactors. So far, CVD models have invoked critical assumptions that made the equations easy to solve but also meant that only qualitative trends could be predicted. Jensen's group is exploiting modern computer technology, in particular the CRAY-1, to derive accurate models for improving CVD chamber design.

One example is the commercial tubular LPCVD reactor wherein silicon wafers are placed concentrically, as phonograph records in a record rack. A two-dimensional model designed by the group for this reactor correctly predicts film thickness across each wafer and from wafer to wafer. With such a model it is possible to select operating conditions where film thickness variations are minimized. While this model can be solved on conventional computers, the CRAY-1 is essential to the ongoing modeling of complex flow and growth patterns in horizontal, cold wall atmospheric CVD chambers. Two-dimensional convection rolls in helium and nitrogen carrier gases have been computed with finite element analysis by graduate student Harry Moffat. Current efforts focus on computing the full three-dimensional roll patterns and on including realistic CVD chemistry for silicon and gallium arsenide growth. The two-dimensional analysis has predicted experimental observations and the full model



Schematic of tubular hot wall LPCVD reactor.

should further resolve disputes in the CVD literature concerning flow phenomena in horizontal CVD reactors. CRAY-aided analysis in this case provides the same insights as would very difficult experiments.

Another type of reactor, the stagnation-point flow reactor, is of particular interest to Dr. Jensen's group because of its well-defined flow configuration. In this type of reactor, the substrate is perpendicular to the flow stream carrying the reactant. The advantages of this reactor are its performance in generating a uniform film thickness and the ease with which performance can be analyzed, since the growth process for some systems may be considered one-dimensional. Jensen said this type of reactor may become increasingly attractive to the semiconductor industry as the industry moves to larger silicon wafers, since the chamber is suited to single wafer processing.

Jensen's group has developed a one-dimensional model of stagnation-point flow reactors. Currently, a two-dimensional simulation is being refined which takes into account both axial and radial flow streams. The previous, one-dimensional simulation simplified the problem by assuming a reactant substrate of infinite area, eliminating the need to take into account the substrate's edges and the chamber walls. The current simulation accounts for edge and wall effects, which can greatly influence recirculation and growth patterns in the chamber. This approach will also allow analysis of flow recirculation and resulting film nonuniformities in commonly used vertical reactors for metal-organic CVD of gallium arsenide compound semiconductors.

The current two-dimensional code is a banded matrix program with 1000 variables. "Now that the code is running, I'm trying to modify the boundary conditions to simulate reactors with different configurations," explained Carl Houtman, a student working on the program. "Because of the CRAY's vectorization capability, it can solve a 1000-variable matrix in only 2-3 CPU seconds, which is about 20 times faster than on our Cyber 74. Without the use of the CRAY, I wouldn't attempt to tackle this problem."

Stagnation-point flow modeling also forms the basis for simulation of plasma-assisted CVD. This process is growing in importance because it offers low deposition temperatures and special material properties. Low temperatures are advantageous since some dopants used in semiconductor manufacturing will diffuse when heated. Thus, low temperatures allow manufacturers to retain sharp concentrations of dopants on their devices.

Plasma-assisted CVD uses a radio frequency field instead of heat to drive the chemical reactions. In addition to the fluid flow and neutral chemical species balances of the conventional CVD stagnation-point flow model, equations for the electric field, ion concentration, and electron density must be included to accurately model plasma-assisted CVD. Moreover, solution profiles often vary rapidly over small spatial distances. Therefore, a fine mesh must be employed with a resulting very large number of nonlinear algebraic equations. Again, the speed and memory of the CRAY-1 are essential to simulate reactor performance.

"At this point I'm working on a one-dimensional code for a plasma-assisted reactor, because the physics involved is extremely complex. But a one-dimensional model is still useful for analyzing certain reactor configurations, such as the parallel plate discharge reactor," explained David Graves, a student developing the program. "A real value of modeling is what it teaches us about the physics and chemistry involved. It gets people asking the right questions."

The CVD simulations incorporate numerical techniques developed at the University of Minnesota as well as special algorithms such as Sandia National Laboratory's nonlinear differential algebraic system solver, DASSL. Access to the CRAY-1 has made much of the two- and three-dimensional modeling feasible. "Because of the CRAY's speed we'll end up paying between one-tenth and one-half of what it would cost to run these programs on the University's other computers," said Dr. Jensen. "And some of the analyses could only be carried out on the CRAY."

WORLDWIDE BUSINESS CENTERS

Domestic

Corporate Headquarters
608 Second Avenue South
Minneapolis, MN 55402
(612)333-5889

Government Relations Office
1828 L Street NW
Suite 201
Washington, DC 20036
(202)775-0155

Central Region Office
5330 Manhattan Circle, Suite F
Boulder, CO 80303
(303)499-3055
- Albuquerque
- Chicago
- Minneapolis

Eastern Region Office
11710 Beltsville Drive
Suite 500
Beltsville, MD 20705
(301) 595-5100
- Atlanta
- Boston
- Laurel
- Pittsburgh

Petroleum Region Office
5858 Westheimer, Suite 500
Houston, TX 77057
(713) 975-8998
- Dallas
- Tulsa

Western Region Office
5776 Stoneridge Mall Road
The Atrium, Suite 350
Pleasanton, CA 94566
(415) 463-2800
- Los Angeles
- Seattle

International

Cray Canada Inc.
4141 Yonge Street, Suite 302
Toronto, Ontario
Canada M2P 2A8
(416) 229-2729

Cray Research France, S.A.
Rue de Tilsitt
75017 Paris, France
01-766-01-55

Cray Research GmbH
Perhamerstrasse 31
8000 Munich 21, West Germany
89-56014-0

Cray Research Japan, Limited
W 1154, Shin Aoyama Building
1-1 Minami-Aoyama 1-chome
Minato-Ku, Tokoyo 107, Japan
03-403-3471

Cray Research (UK) Ltd.
Cray House
London Road, Bracknell
Berkshire RG12 2SY
United Kingdom
0344-485971

CRAY RESEARCH, INC.

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