High-End Computing Challenges in Aerospace Design and Engineering

F. Ronald Bailey

Advanced Management Technology Inc., M.S. 258-6, NASA Ames Research Center, Moffett Field, CA 94035-1000. fbailey@mail.arc.nasa.gov

Abstract. High-End Computing (HEC) has had significant impact on aerospace design and engineering and is poised to make even more in the future. In this paper we describe four aerospace design and engineering challenges: Digital Flight, Launch Simulation, Rocket Fuel System and Digital Astronaut. The paper discusses modeling capabilities needed for each challenge and presents projections of future near and far-term HEC computing requirements. NASA's HEC Project Columbia is described and programming strategies presented that are necessary to achieve high real performance.

1 Introduction

High-End Computing (HEC) has had a major impact on design and engineering in the aerospace industry. HEC simulation is routinely used to improve understanding of complex physics phenomenon and thus lead to improved design solutions. HEC is enabling the use of CFD to significantly reduce wind-tunnel testing in vehicle design and to provide data that cannot be obtained by wind-tunnel experiments. None of these would be possible without the more than five orders-of-magnitude increase in HEC performance over the past three decades, which, in turn, has motivated the development of models of increasing fidelity and complexity. As a result, HEC applications have significantly reduced cost, lowered risk and improved performance. However, there is still huge potential for even greater benefits from expanded application of HEC in aerospace and it is timely to look at what significant advances in simulation can be initiated now. Here we explore four challenging and potentially fruitful areas for advancement.

The first challenge is Digital Flight that simulates aircraft dynamic flight and advances the application of CFD to broader areas of the flight envelope. The second challenge is space vehicle Launch Simulation in which high-fidelity modeling of the mission profile is used to improve mission planning and design as well as provide better assessments of risk. The third challenge is Rocket Fuel System Simulation that exemplifies the use of high-fidelity modeling of complex systems for development of space transportation systems. Finally, the fourth challenge is the Digital Astronaut that models the human body's response to a prolonged space environment.

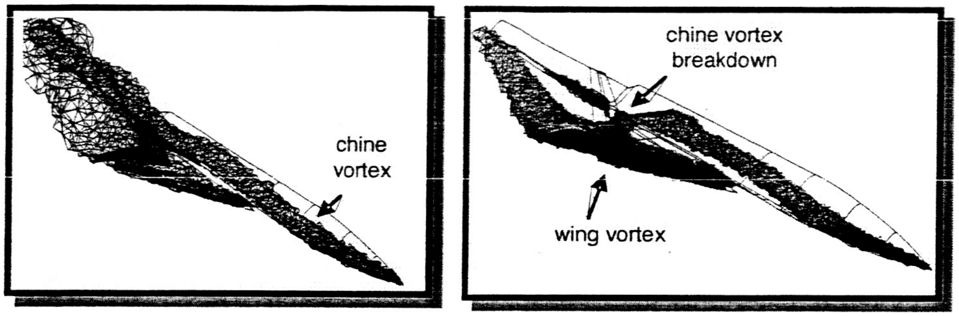
In the following we describe each challenge and give examples of recent pioneering developments in CFD modeling that provide motivation to pursue further development. We propose the solution of model problems as one means of advancing our solution capability in areas of physics modeling, algorithms and programming. These model problems are meant to be solved in the next few years and we have chosen NASA's Project Columbia as a candidate HEC platform. The problems have been sized to be solved on a 12 TeraFLOPS platform. This is 20% of Project Columbia and is considered the practical available portion of a system that is shared by all of NASA. Solution times are estimated based on CPU hours used by known codes on smaller, precursor problems and we implicitly assume that computational efficiency remains constant as we move to our model problems. Next, we estimate HEC platform performance needed to solve a complete problem with the fidelity and turnaround necessary to meet the practical demands of design and engineering. Finally, we give a short description of Columbia and discuss programming strategies necessary to achieve good performance.

2 Digital Flight

One of the most pressing needs in aerospace vehicle design is the accurate prediction of stability and control characteristics throughout the flight envelope. Accurate data is especially critical for automatic control systems. Heretofore, the prediction of stability and control parameters has depended on expensive wind tunnel and flight tests. In many cases the predictions were inadequate and vehicles exhibited unexpected stability and control problems discovered in flight test – sometimes with catastrophic consequences and at the cost of human lives. We are now on the threshold of Digital Flight: the ability to predict aerodynamic stability and control parameters over the entire flight envelope and to simulate dynamic flight behavior using CFD. The payoff is substantial. Digital Flight will lead to a much better understanding of flow characteristics and to improved designs. Design cycle time will be reduced and flight control laws improved. Wind tunnel and flight tests will be reduced leading to decreased cost. Moreover, project risk will be reduced and safety increased.

The important issues in stability and control involve separated flows. Longitudinal, directional and lateral instability are all characterized by massively separated vortical and wake flows. Resolving vortical and wake flows require very fine grids to capture the large gradients present. Because placing fine grids everywhere is not practical, adaptive grid refinement techniques are being developed that refine the grid in regions of large-gradients as the solution evolves. Figure 1 shows the application of adaptive grid refinement applied to vortical flow simulation [1]. Figure 1a shows the chine vortex extending beyond the aircraft tail which results from refinement of the initial grid. Figure 1b shows the result of the final refined grid that correctly predicts the chine vortex bursting ahead of the vertical tail.

Reynolds-averaged Navier-Stokes (RANS) models are generally sufficient for quantitative predictions of mean flow characteristics as long the flow is not dominated by massive separation. Prediction of massively separated flows is much improved by use of Detached-Eddy Simulation (DES) [2]. DES employs a Reynolds-averaged turbulence model near the wall and Large-Eddy Simulation in the separated regions. It combines the efficiency of RANS near the wall with the capability of LES to resolve geometry-dependent, unsteady, three-dimensional turbulent portions of the flow. Figure 2 compares experimental lift and pitching moment coefficients vs. angle-of-attack for the F/A-18E with RANS and two turbulence models (SA Baseline and SST Baseline) and time-averaged calculations (DES Baseline and DES Adapted) [3]. DES with adapted grid show good agreement over the entire angle-of-attack range.



(a) Unadapted grid

(b) Solution adapted grid

Fig. 1. Solution Adaptive Grid for Vortex Breakdown [1]

Now consider a Digital Flight model problem that solves unsteady RANS to simulate a six degree-of-freedom vehicle flight trajectory and includes solution adaptive grids and DES, but ignores aeroelastic effects. Quantities such as wind angles, velocity vector, forces, moments, control surface deflections, thrust, etc. are captured at points along the trajectory. These quantities populate a data base used for control law design and analysis and for input to piloted flight simulators. Data is typically captured at 50-80 Hz and the data base may contain 300,000 trajectory points.

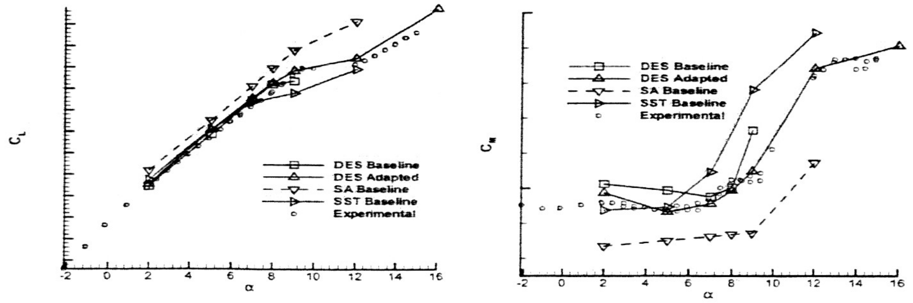


Fig. 2. Comparison of experimental and calculated C_L and C_M [3].

Experience from the Drag Prediction Workshop II [4,5] suggests that to achieve adequate quantitative results of lift, drag and moment for performance purposes at steady flight and modest angle of attack using RANS requires a good quality grid with grid adaptation and a minimum of 25 million grid points for a half configuration wind-tunnel model. For Digital Flight the number of grid cells must be doubled to cover the full configuration domain, increased in areas of solution grid adaptation and increased again for higher flight Reynolds numbers. We estimate that 60 million grid points is the minimum needed to investigate solution adequacy over a broad range of flight conditions.

To estimate computation time requirements we observe that for well known codes such as OVERFLOW-D [6, 7] about 5000 time steps [8] are required for flow to travel one body length in an RANS simulation. For a typical fighter configuration maneuvering at 0.8 Mach number the flow travels about 15 body lengths in one second. The addition of DES has been observed to decrease the time step (increase solution time) by factors of 5-10 [3]. Based on the computing times reported for OVERFLOW-D [9] on a SGI Origin 3000 and an estimated 10 times increase in computations due to DES, we estimate one second of flight requires a solution time of 12 hours on a 12 TeraFLOPS platform. At an 80 Hz sample rate it takes nine minutes to obtain a database data at one trajectory point.

The practical application of Digital Flight to vehicle design requires computational result production match design flow time. For example, a typical database of 300,000 trajectory points requires about six months of wind-tunnel tests (including model construction, test preparation, etc.). To match this rate our model simulation needs to calculate a trajectory point per minute and this would require a 100 TeraFLOPS platform. For practical applications the platform requirement may be larger due to uncertainties in grid resolution and modeling of separation physics. Therefore, we estimate a HEC platform in the range of 100-

500 TeraFLOPS peak performance would be needed for practical design and engineering application.

3 High-Fidelity Launch

As NASA space missions and systems become more complex associated cost and risk become growing concerns. Advances in HEC will enable the high-fidelity simulation of airframe, propulsion system and auxiliary systems throughout launch and accent. High-accuracy launch simulations will make it possible to plan and evaluate the readiness to launch with much lower uncertainty. Integrated, high-fidelity modeling will enable simulation of failures and associated vehicle response. Rapid turnaround will enable analysis of multiple, dynamic configurations even at the conceptual design level. As HEC capability advances, running statistical significant numbers of simulations will enable risk assessment inputs to better represent reality in investigations such as propulsion system failure as well as the ability to recover from failures.

Simulation of a flight mission to orbit can be divided into a launch phase and an ascent phase. The launch phase requires a software procedure with the capability to simulate lift-off in the launch pad environment including exhaust heat radiation, acoustic, debris and local weather effects. The ascent phase requires a software procedure that can simulate flight to orbit including dynamics of booster and auxiliary tank separation. Ascent simulation is essentially Digital Flight applied to multiple rigid bodies and their dynamic separation. The governing equations may be Euler or RANS at lower altitudes and Boltzman equations in the low density upper atmosphere. A high-fidelity CFD vehicle flight model capable of six degree-of-freedom (6-DOF), multiple body flight is a key component for both launch and ascent phases.

Recent simulation of the shuttle Columbia (STS-107) debris trajectories [10] is representative of current capability in complex 6 DOF modeling. OVERFLOW-D [6, 7] solving the RANS equations and CART3D [11, 12], solving the Euler equations, were used to perform unsteady, moving-body, CFD simulations of the entire shuttle/debris flow field and the aerodynamic forces and moments acting on the debris. Both are capable of 6-DOF simulation of and rigid-body relative motion among an arbitrary number of bodies. Over 40 OVERFLOW-D and 400 CART3D 6-DOF simulation were preformed in the investigation of foam debris shed from the region of the left bipod-ramp of STS-107. The analysis provided an estimate of the debris trajectory, impact velocity and foam size that was instrumental in establishing the possibility for a piece of foam debris to cause massive damage to the Shuttle Orbiter wing RCC panels and T-seals. Figure 3 shows a CART3D simulated trajectory that closely resemble the strike location observed on film.

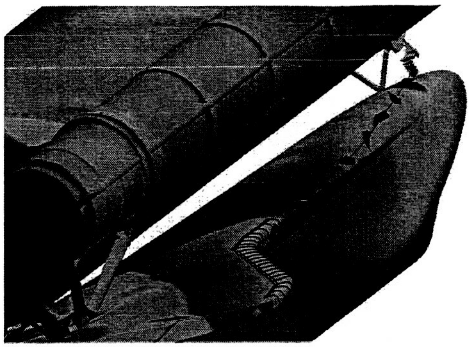


Fig. 3. CART3D 6-DOF debris trajectory [10]

Since we treated Digital Flight in the previous section, we concentrate on launch simulation. A simulation model would be designed to treat the whole launch environment until the vehicle has cleared the launch tower. The model would integrate 6 DOF multiple-body motion, debris impact, propulsion system vibration and exhaust, acoustics due to exhaust, fuel accumulation in the exhaust plume, exhaust chemistry including fuel burning, thermal stress on the vehicle structure and finally weather at the launch site. The model would be very complex and integrate data from propulsion simulation, meso-scale weather prediction and experiment. Thus it is necessary to consider developing the model in stages in which each stage adds new or increased fidelity.

Consider a first stage conceptual model that treats vehicle motion, debris motion and exhaust plume effects in the launch environment, including the presence of the launch facilities. We assumed that viscous effects are not significant and that vehicle motion and exhaust blast waves can be modeled by time accurate Euler equations. Simplified plume heating and chemical reaction models are also assumed. From exhaust blast wave speed and launch tower structure details we determine time and length scales. Thus, we pick a time step based exhaust plume pressure wave velocity and a spatial resolution of, say, one foot. Utilizing CART3D and a 100 million point grid this launch simulation could be completed in about a day on a 12 TeraFLOPS platform or one sixth of Project Columbia.

For practical design and engineering application the simulation would be more complex, perhaps requiring 5-10 times more computation, need to be completed in about a day, and require a HEC platform capable of 60-120 TeraFLOPS.

4 Rocket Fuel Subsystem Simulation

The performance and reliability of rocket engines is critical to space transportation missions. A critical design challenge is rocket engine turbomachinery, which is the most expensive component in terms of development and operations and is the cause of the majority of engine failures. We currently lack a practical capability to predict transient, 3-dimensional, environments internal to turbopumps and therefore must rely on the costly test-fail-fix cycle. As a consequence we have experienced problems related to fluid dynamics in rocket engines such as the Space Shuttle Main Engine (SSME) and others. Given advanced simulation capabilities in turbomachinery we can reduce development and operations cost, reduce development time, quantify design margins and increase safety and robustness.

The goal of Rocket Fuel System Simulation is to provide a high-fidelity framework for the design and analysis of the fuel/oxidizer supply subsystem for a liquid rocket propulsion system including unsteady turbopump flow analysis. The system is basically made up of multiple pumps and feed lines and its simulation will provide the basis for determining and addressing root causes of transient flow/cavitation induced vibration that result in structural damage such as turbine blade cracks and breakage.

The first major challenge in developing a system simulation capability is to model flow through turbopumps. High-performance turbopump design is currently a semi-empirical process that experience has shown misses many important features of turbopump flows, thus CFD simulation can add greatly to improved design. Especially valuable is information such as transient flow phenomena at start-up, and non-uniform flows that impact vibration and structural integrity. Challenges to developing CFD models are significant. Rocket turbopumps have complex geometries including full and partial blades, tip leakage and an exit boundary to a diffuser. Their flows include a number of complex flow phenomena including boundary layer transition, turbulent boundary layer separation, and wakes tip vortices as well as influences of three-dimensional and Reynolds number effects. Modeling cavitation is perhaps the greatest challenge as no models have advanced to the point of producing quantitative results for engineering.

Recent advancement in turbopump simulation has been demonstrated by an unsteady computation for the SSME turbopump impeller/diffuser using the INS3D code developed at NASA Ames Research Center [13]. To resolve the unsteady interaction between the rotating and stationary parts an overset grid was used. The entire configuration including inlet guide vanes, impeller blades and diffuser blades was constructed using 34.3 million grid points in 114 zones. Instantaneous snapshots of particle traces and pressure surfaces from these computations are shown in Figure 4. In this simulation one impeller rotation requires approximately 160 hours using 128 SGI Origin 3000 CPUs (600 MHz) or the equivalent of about 20,000 CPU hours.

The next step to advancing liquid fuel subsystem simulation is modeling a multistage turbopump. We consider a model problem consisting of a six stage turbopump. We ignore cavitation, upstream and downstream manifolds and ducting, and coupled shroud and hub cavity flows. A six stage turbo-pump is estimated to require 150 million grid points and about 1 million Origin CPU hours to simulate 10 revolutions, which is sufficient time for start-up transient flow disturbances to die out. We estimate an INS3D simulation on a 12 TeraFLOPS HEC platform would be completed in about four days. This turnaround is considered reasonable for a flow analysis and for simulation development.

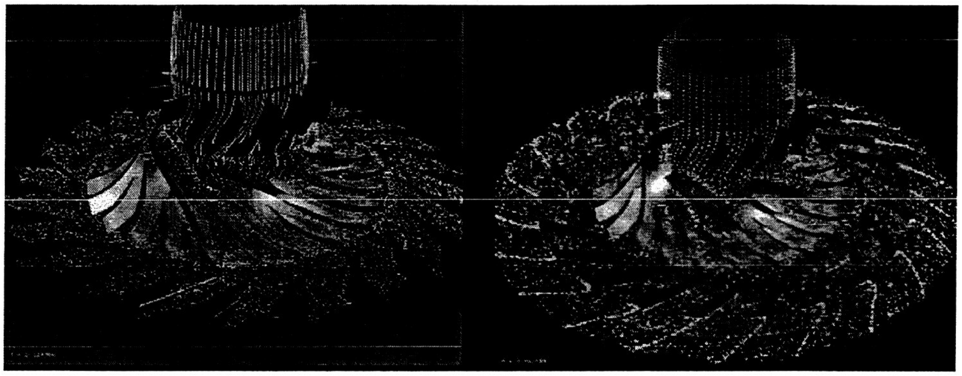


Fig. 4. Particle traces and pressure surfaces for unsteady turbopump computations (first rotation and end of third rotation) [13]

Advancing the model to include the items ignored is estimated to require a 2- to 5-fold increase in HEC performance. A typical fuel subsystem with four pumps and piping may require an addition five-fold increase. Finally, for practical design analysis purposes a two day turnaround is acceptable. Thus for a complete design simulation we estimate that a 240 - 600 TeraFLOPS HEC platform is required.

5 Digital Astronaut: Circulatory System

The Digital Astronaut is a new NASA effort aimed at an integrated modeling and database system that enables the efficient construction and utilization of a class of quantitative models of the whole human body in order to simulate the function of a normal human being during and after a space voyage. This system will include HEC enabled computer simulation models that will allow detailed study of the effects of weightless space flight as well as the effects of the altered spacecraft environment on astronauts. The Digital Astronaut will be capable of appropriate structural integration, spanning the required multiple levels of biological organization, from the whole body through the organs, tissues, and cells to the genes and proteins. In addition, the Digital Astronaut will be capable of integrating multiple coupled physiological subsystems and components of biological networks (circulatory, respiratory, musculoskeletal, etc.) into a consistent whole-body model. Finally the Digital Astronaut will provides a data integration function for integrating space and analogue related empirical data, phenomenological observations and experimental studies with theoretical principles. The resulting system will help biomedical researchers understand the human effects of space flight, to use this knowledge to improve medical care for space voyagers and to design appropriate countermeasures that reduce the biomedical risks of space flight.

Simulating the effects of space travel on the human circulatory system is a key element of the Digital Astronaut. The altered cardiac output due to deconditioning during space flight and readaption on return impacts the blood circulation in the human body. This is particularly evident in the brain where altered blood supply impacts oxygen supply to certain parts. Analysis of this condition requires the capability to simulate blood flow in arteries and capillaries. Hemodynamic modeling challenges are significant [14]. Vascular networks exhibit anatomically complex geometry and their three-dimensional reconstruction requires techniques using magnetic resonance imaging, magnetic resonance angiogram and computed tomography to obtain accurate anatomical vasculature. Blood is a non-Newtonian fluid where red blood cell aggregation at low shear rates which makes the apparent blood viscosity increase and a higher shear rates red blood cell deformation which makes apparent viscosity decrease. Blood vessels exhibit distensible wall motion due to heart pulse and relative diameter change up to 20% requiring a structural deformation model for the arterial walls. For simulation of major arteries to be computationally manageable minor arteries such as arterioles, venules and capillaries need to be truncated and an auto-regulation model is required for modeling the outflow boundaries conditions. Finally, the effects of varying gravity including those on deformable wall motion and the resulting human circulatory flow patterns must be modeled.

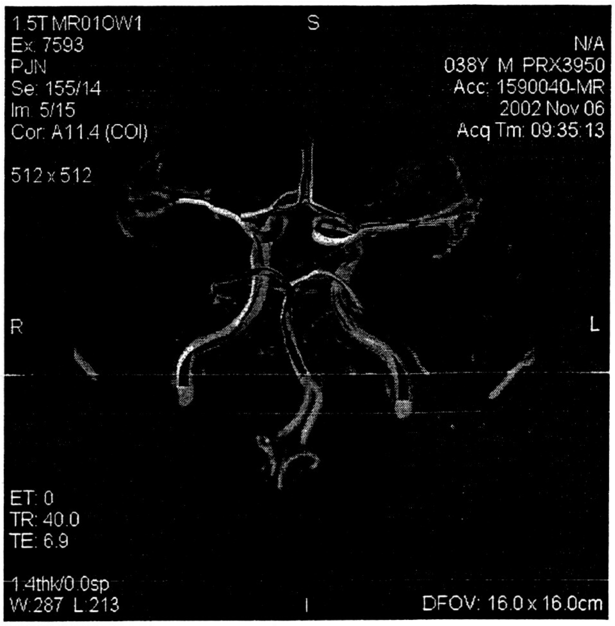


Fig. 5. Circle of Willis Simulation Superimposed on MRA [14]

A pioneering example of simulation of circulatory blood flow in the brain is illustrated in Figure 5 by the blood flow distribution through a realistic Circle of Willis configuration superimposed on an MRA image [14]. Results were obtained using an MPI-OpenMP hybrid version of the INS3D code [15] and they represent the first simulation of blood circulation using non-Newtonian flow models within deformable walls. This pioneering simulation required 3000 SGI Origin 3000 CPU (600MHz) hours for 1 million grid points.

Circulatory blood flow simulation is in its infancy with many challenges remaining in geometry and physics modeling. A challenging next step would be simulating circulatory blood flow in the large vessels of the heart. We estimate 10 million grid point heart simulation with INS3D can be completed in less then a day on a 12 TeraFLOPS platform.

For analysis and counter measure development, three days to simulate the brain-heart circulatory system is a reasonable target and we estimate a HEC platform capable of 100 - 200 TeraFLOPS would be required for a 100 million grid simulation.

6 Project Columbia

Project Columbia has been initiated by NASA to provide an unparalleled HEC capability to solve large-scale computational aerospace science and engineering problems. The system, with a peak performance in excess of 60 TeraFLOPS, is being installed in the NASA Advanced Supercomputing (NAS) facility at Ames Research Center and is scheduled for full operation by the end of 2004. The system is configured as a cluster of 20 SGI Altix 3700 computers each with 512 processors and 1024 GB of memory. The total system has 10,240 Intel Madison processors, 20TB of memory and utilizes the Linux operating system. Two communication fabrics connect the Altix systems. An Infiniband switch fabric provides low latency MPI communication and a 10 Gigabyte Ethernet switch fabric provides user access and I/O communications. Each SGI Altix uses the NumaLink communication scheme to implement a non-uniform memory access architecture resulting in a single system image. Processors share a single address space and each processor is provided with low latency access to global memory. Columbia can be configured into a capability portion and a capacity portion. Four Altix systems will be linked via advanced NumaLink to allow MPI to use global shared memory constructs to significantly reduce inter processor communication latency. This 2048 processor subsystem will provide a powerful 12 TeraFLOPS "capability" platform for pioneering more finely-gained applications. The remaining 16 Altix systems then provide a powerful 48 TeraFLOPS "capacity" platform for the bulk of NASA's large-scale science and engineering applications.

7 Programming Strategies for Columbia

The turnaround time and peak performance estimated for pioneering and design and engineering applications are summarized in Table 1. We are estimating peak performance required, but the actual application performance obtained on Columbia will depend to a large degree on problem characteristics, such as number of grid points and zones, algorithm used and the programming approach. Our goal is to achieve no less than about 80 percent linear scalability and 20% processor efficiency to be consistent with our experience on the 1000 processor SGI Origin 3000 systems at NAS. In some cases, such at Digital Fight, there is obvious coarse grained parallelism and good scalability potential since several trajectories can be run in parallel to populate a stability and control data base. Some cases, like simulating the human circulatory system, good scalability for large processor count may not be so obvious and may require innovative programming strategies to achieve.

Table 1. Turnaround Time and HEC Performance Estimates for Aero Challenge Pioneer and Design & Engineering Applications

Aero HEC Challenge	Pioneer Problem		Design & Engineering Problem	
	Solution Time	Platform Peak Performance (TeraFLOPS)	Solution Time	Platfrom Peak Performance (TeraFLOPS)
Digital Flight	9 minutes	12	1 minute	100-500
Mission Launch	1 day	12	1 day	60-120
Rocket Fuel System	4 days	12	2 days	240-600
Digital Astronaut Circulatory System	1 day	12	3 days	100-200

Columbia's architecture provides flexibility in programming approaches for increased scalability. It supports both MPI and OpenMP programming. Parallel programming among the Altix systems is implemented by MPI while programming within each Altix can be implemented by MPI or Open MP. Columbia is also well suited for hybrid programming [16]. For example, within each Altix system MPI is used to implement coarse grained parallelism among Altix systems and their nodes while shared-memory OpenMP is used to implement parallelism within each node. Hybrid programming has the advantage of increasing the number of processors available to work on the problem while requiring loop-level programming to span only a small number of processors. Note, that connecting four Altix systems by NumaLink to form a capability system will allow MPI to use global shared memory constructs to dramatically reduce latency across the 2048 processors.

Multi-Level Parallelism (MLP) is another hybrid programming approach that takes advantage of the Altix single system image design. MLP is an extension of the Cray shared memory programming model of the 1980-90s [17]. It was developed for NUMA architectures that permit shared memory access to global data such as the SGI Origin series and SGI Altix. MLP replaces MPI with UNIX/LINUX forked processes and has a total of three routines in the library. Because communication is through shared memory loads/stores for all tasks, latencies are on the order of hundreds of nanoseconds rather than several microseconds. Lower communication latency improves the performance of coarse parallelism. Fast dynamic load balancing is supported by the dynamic creation of virtual nodes and the shared memory interface. Thus, MLP provides high levels of scaling efficiency for large processor counts.

Hybrid programming is well suited to CFD codes where the grid domain is portioned into zones. During iteration each zone's solution is updated independently and then the global solution is updated by exchange of zonal boundary data. MPI/MLP tasks are initiated to perform the zonal iteration on groups of zones assigned to each node in a manner that attempts to balance the workload among tasks. As each zone is processed the task takes advantage of loop-level parallelism using OpenMP directives. MLP scalability experiments with highly optimized OVERFLOW [18] code conducted on a 1024 SGI Origin 3000 platform showed scalability to nearly 1000 processors.

All HEC platform processors suffer from the fact that processor performance is improving at a much greater rate than that of memory. Large cache memories have been introduced to bridge this gap and Columbia's processors have six MB of cache (nine MB on the capability system). To get the most benefit from cache the programming strategy needs to map the problem so that cached operands flow uninterrupted during the execution of fine-grained loops.

Figure 6 illustrates the benefits of hybrid programming and cache optimization by showing the history of runtime improvement in INS3D turbopump simulations. The initial simulation required 42 days using 32 CPUs (250 MHz) on a SGI Origin 2000 using MPI and was reduced to just over a day using 480 CPUs (400 MHz) on a SGI Origin 3000 using MLP with cache use optimized. A new CFD flow solver has recently been developed that takes advantage of the hybrid programming [19]. Experiments performed on a SGI Origin 3000 resulted in a speedup of 514 on 640 CPUs for an OpenMP implementation vs. 392 on 640 CPUs for an MPI implementation.

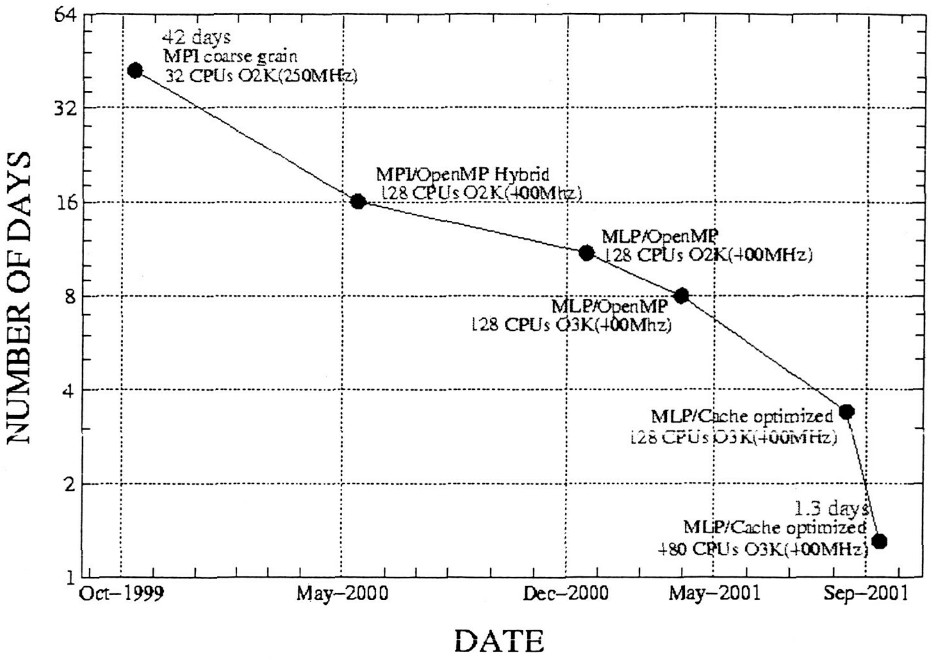


Fig. 6. Runtime Improvement for INS3D Turbopump (34 million grid points, 114 zones) on SGI Origin 3000.

8 Conclusion

Recent CFD development have demonstrated the feasibility of embarking on new HEC challenges in aerospace design and engineering that will decrease costs, improve performance and reduce risk. NASA's Columbia Project will provide a 60 TeraFLOPS HEC platform with the capability needed to foster the development of new models to meet these challenges. While a lot of work lies ahead in developing accurate physics models and algorithms and programming strategies to achieve adequate real, sustained performance the payoff will be truly significant.

References

1. S. Z. Pirzadeh: *A Solution-Adaptive Unstructured Grid Method by Grid Subdivision and Local Remeshing.* AIAA Journal of Aircraft, 37(5):818-824, 2000.

2. P. R. Spalart, W-H Jou, M. Strelets and S. R. Allmaras: *Comments on the Feasibility of LES for Wings, and on RAHS/LES Approach*. Advances in DNS/LES, 1st AFOSR Int. Conf. on DNS/LES, Greyden Press, Columbus OH, August 1997.
3. J. R. Forsythe and S. H. Woodson: *Unsteady CFD Calculations of Abrupt Wing Stall Using Detached-Eddy Simulation*. AIAA Paper 2003-0594, January 2003, Reno, Nevada.
4. K. R. Laflin, O. Brodersen, M. Rakowitz, J. C. Vassberg, E. N. Tinoco, R. A. Wahls, J. H. Morrison, and J. Godard: *Summary of Data from the Second AIAA CFD Drag Prediction Workshop*. AIAA Paper 2004-0555, Jan. 2004. Reno, NV.
5. M. J. Hemsch and J. H. Morrison: *Statistical Analysis of CFD Solutions from 2nd Drag Prediction Workshop*. AIAA Paper 2004-0566, Jan. 2004. Reno, NV.
6. R. Meakin: *Adaptive Spatial Partitioning and Refinement for Overset Structured Grids*. Computational Methods Applied Mechanical Engineering, 189:1077-1117, 2000.
7. R. Meakin: *Automatic Off-Body Grid Generation for Domains of Arbitrary Size*. AIAA Paper 2001-2536. June 2001. Anaheim, CA.
8. R. Meakin. Private Communication.
9. T. A. Nygaard and R. L. Meakin: *An Aerodynamic Analysis of a Spinning Missile with Dithering Canards*. AIAA Paper 2002-2799. June 2002. St. Louis, MO.
10. R. J. Gomez, D. Vicker, S. E. Rogers, M. J. Aftosmis, W. M. Chan, R. Meakin, and S. Murman: *STS-107 Investigation Ascent CFD Support*. AIAA Paper 2004-2226, June 2004. Portland, OR.
11. M. J. Aftosmis, M. J. Berger and J. E. Melton: *Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry*. AIAA Journal 36(6):952-960. June 1998.
12. M. J. Aftosmis, M. J. Berger and G. Adomavicius: *A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries*. AIAA Paper 2000-0808, Jan. 2000. Reno, NV.
13. C. Kiris, W. Chan, and D. Kwak: *A Three-Dimensional Parallel Time-Accurate Turbopump Simulation Procedure Using Overset Grid Systems*. Computational Fluid Dynamics 2002, pp. 673-684. Springer-Verlag, Berlin. 2002
14. C. S. Kim, C. Kiris, D. Kwak, and T. David: *Numerical Models of Human Circulatory System under Altered Gravity: Brain Circulation*. AIAA Paper 2004-1092, Jan 2004. Reno, Nevada.
15. C. Kiris and D. Kwak: *Aspects of Unsteady Incompressible Flow Simulations*. Computers & Fluids, 31: 627-638, 2002.

16. L. Oliker, X. Li, P. Husbands, and R. Biswas: *Effects of Ordering Strategies and Programming Paradigms on Sparse Matrix Computations.* SIAM Review 44(3):373-393, 2002.
17. J. R. Taft: *Achieving 60 GFLOP/s on the Production Code OVERFLOW-MLP.* Parallel Computing, 27(4):521-536. 2001.
18. D. C. Jespersen, T. H. Pulliam, and P. G. Buning: *Recent Enhancements to OVERFLOW.* AIAA Paper 97-0644, Jan 1997. Reno, NV.
19. M. J Berger, M. J. Aftosmis, D. D. Marshall and S. M. Murman: *Performance of a New CFD Solver using a Hybrid Programming Paradigm.* To be published in Journal of Parallel and Distributed Computing.