Role of HPC in Advancing Computational Aeroelasticity

Guru P. Guruswamy
NASA Advanced Supercomputing Division
Ames Research Center

Abstract

On behalf of the High Performance Computing and Modernization Program (HPCMP) and NASA Advanced Supercomputing Division (NAS) a study is conducted to assess the role of supercomputers on computational aeroelasticity of aerospace vehicles. The study is mostly based on the responses to a web based questionnaire that was designed to capture the nuances of high performance computational aeroelasticity, particularly on parallel computers. A procedure is presented to assign a fidelity-complexity index to each application. Case studies based on major applications using HPCMP resources are presented.

1 Introduction

Aeroelasticity that involves strong coupling of fluids, structures and controls is an important element in designing an aircraft. Computational aeroelasticity based on low fidelity methods, such as the linear aerodynamic flow equations coupled with the modal structural equations, is well advanced. Although these low fidelity approaches are computationally less intensive, they are not adequate for the analysis of aircraft which can experience complex flow/structure interactions. For example, the B-1 aircraft experienced vortex induced aeroelastic oscillations [1]. Vertical tails of the F-18A experienced structural oscillations due to the burst of leading edge vortices [2]. Aircraft that fly in the transonic regime experience buffet associated structural oscillations [3] and also a dip in flutter speed [4]. Modern fighters such as the F/A 18E aircraft experience abrupt wing stall that can be dominated by unsteady flows

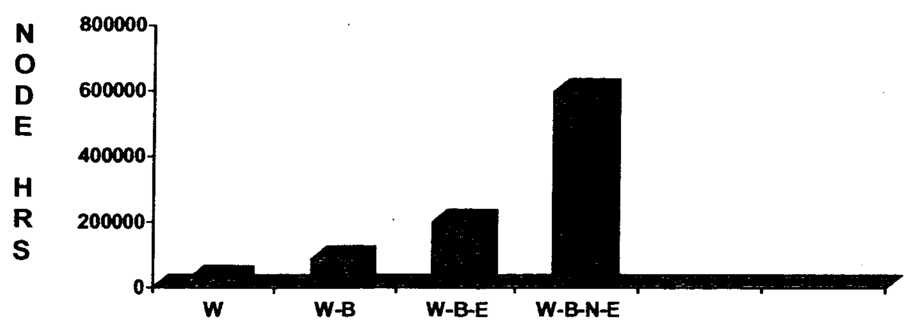

Fig. 1 Computer time in SGI Origin 2000 node hrs needed for a single design point transonic flutter boundary computation using coupled Navier-Stokes and modal equations. (W: Wing, B: Body, E: Empennage, N: Nacelle)

possibly associated with aeroelastic oscillations [5]. High fidelity equations such as the Euler/Navier-Stokes (ENS) for fluids and the finite elements (FE) for structures are needed for accurate aeroelastic computations for these complex fluid/structure interaction situations. Using these high fidelity methods, design quantities such as structural stresses can be directly computed.

Aeroelastic computations are typically orders of magnitude more expensive than steady calculations on rigid configurations because the multidiscipline aspect adds additional complexities to the physics. Figure 1 shows a typical increase in the requirement of computational time to compute a transonic flutter boundary (based on 5 modes, 5 frequencies and 20 Mach numbers, 10000 time steps per case) for increasing geometric complexities. All computer times required are presented in terms of SGI Origin 2000 single processor hours. The growth in CPU time required is exponential. Hundreds of such computations are required for a complete aircraft design.

Continuous growth in computer resources is required to advance ever challenging new technologies in the aerospace industry. To sustain the current leadership in aerospace sales, it is crucial for the US to maintain its leadership in supercomputing. In order to nourish US leadership in supercomputing, congress authorized the HPCC program when it passed the High Performance Computing Act of 1991. Created as a dynamic R&D program, it provides the sustained focus needed for developing new computer technologies and applications for the needs of a changing world. The program's major objective is to provide a foundation for the country's R&D needs for the approaching 21st century and beyond. As a leading participant in this program, the High Performance Computing Modernization Office (HPCMO) of the Department of Defense (DOD) has promoted several state-of-the-art super computer platforms. Figure 2 shows the increase in GFLOPS (billion floating point operations per second) rate for production type aeroelastic codes based on the Navier-Stokes equations coupled with modal structural equations. This effort by the DOD has advanced the development of several super computers based on parallel architectures. Use of those facilities have begun to compute

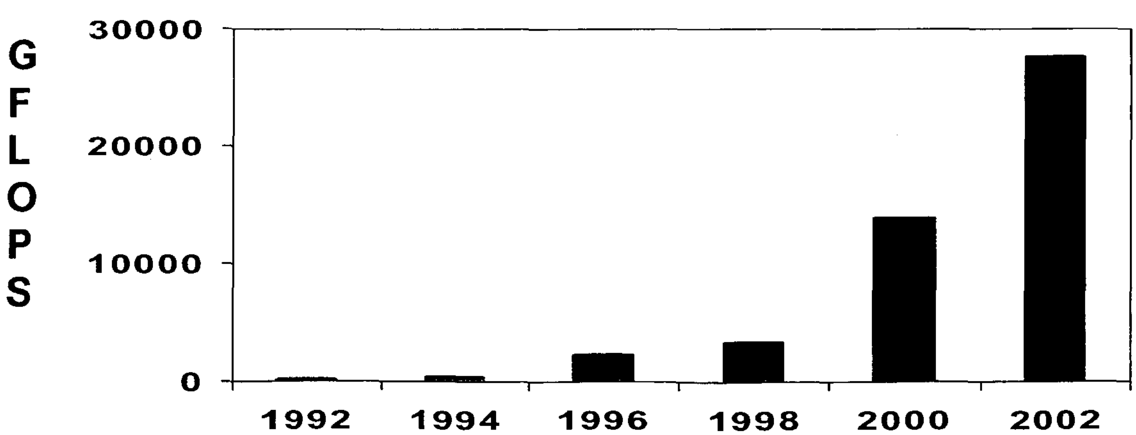

Fig. 2 Growth in supercomputing power

aeroelasticity associated with complex separated flows. Reference 6 shows vortex burst induced structural response results of the F-18/A vertical tail using a moderate grid that required about 10,000 node hrs for one set of flow parameters. However, unsteady aerodynamic and aeroelastic computations on the F-18 E/F at abrupt wing stall, which is yet to be computed, needs millions of node hrs [7].

This report presents the status of the use and impact of HPCMP resources on high fidelity based aeroelastic analysis including unsteady aerodynamics.

2 History of HPCMP

The Department of Defense uses supercomputers and advanced computational methods to conduct basic research, develop and test precision weapons, and investigate new war fighting capabilities [8]. Central to this activity is a partnership among the defense laboratories, test centers and the HPCMP. The HPCMP formally started in 1993 in response to Congressional and senior DoD leadership direction. The program grew from a collection of small high performance computing departments, each with a rich history of supercomputing experience, which independently evolved within the Army, Air Force and Navy laboratories and test centers.

The HPCMP provides the supercomputer services, high-speed network communications and computational science expertise that enables defense scientists and engineers to conduct a wide-range of focused research, development, and test activities. This partnership puts advanced technology in the hands of U.S. forces more quickly, less expensively, and with greater certainty of success The HPCMP fields a unified set of supercomputing services to the DOD science, engineering, test and evaluation communities that includes some of the world's most powerful high performance computing systems, and a premier wide-area network, supporting a significant portion of the nation's top scientists and engineers with high performance computing software development and application assistance.

The HPCMP scope is bounded both in terms of the user community it serves and the technological capability that it delivers. By concentrating the majority of resources at a small number of HPC centers, the program provides computing capabilities that otherwise could not have efficiently been obtained and sustained by the individual Services or federal agencies. This sharing of resources reduces overall acquisition and sustainment costs, and fosters collaboration and cooperation across the DOD science and technology (S&T), and test and evaluation (T&E) communities.

2.1 Program Components

The program is organized into three components: HPCMP HPC Centers, Networking and Software Applications Support. Each component focuses on the most efficient means of supporting the S&T and T&E communities' requirements.

2.1.1 HPCMP HPC Centers

The HPCMP operates four large Major Shared Resource Centers (MSRCs) that enable DOD S&T and T&E communities to effectively use the full range of HPC resources. Each MSRC includes a robust complement of high-end, high performance computing and communications systems that support a wide range of projects. The Distributed Centers (DCs) provide HPC capacity and capability to a specified local and remote portion of the program's community. Modest-sized systems are deployed to DCs where there is a significant advantage to having a local HPC system, and where there is potential for advancing DOD applications using investments in HPC capabilities and resources.

2.1.2 Networking

The Defense Research and Engineering Network (DREN) is DoD's recognized research and engineering network. The DREN is a robust, high-speed network that provides connectivity between the HPCMP's geographically dispersed user sites and HPC centers. Since users and resources are scattered throughout the United States, strong interconnectivity with other major networks and high performance test beds at key exchange points are critical for optimal use of high performance computers.

2.1.3 Software Application Support

"Software Applications Support" is a new terminology that captures the evolutionary nature of the program's efforts to "Acquire and develop joint HPC application software tools, and programming environments," and "Educate and train DoD's scientists and engineers to effectively use advanced computational environments." There are two major components to software application support: Common High Performance Computing Software Support Initiative (CHSSI) and Programming Environment and Training (PET).

CHSSI provides DOD scientists and engineers efficient, scalable, portable software codes, algorithms, tools, models and simulations that run on a variety of HPC platforms. CHSSI, which is organized around 10 computational technology areas, involves several hundred scientists and engineers working in close collaboration across government, industry and academia. The PET component enables the Defense HPC user community to make the best use of the computing capacity the HPCMP provides and extends the range of DoD technical problems that can be solved on HPC systems. PET enhances the total capability and productivity of users through training, collaboration, tool development, software development support, technology tracking, technology transfer and outreach.

2.1.4 DOD Challenge Projects

Approximately 25 percent of the program's total resources are dedicated each year to a set of DOD HPC Challenge Projects. These computationally intensive, high-priority projects are selected annually through a rigorous technical and mission relevance evaluation. The Services and other federal agencies allocate the remaining resources through their unique evaluation processes.

Grand challenge projects are divided into several computational technology areas (CTA). This effort focuses in interdisciplinary computations based on Computational Structural

Mechanics (CSM) and Computational Fluid Dynamics (CFD) for aerospace configuration.

This report focuses on the area of computational aeroelasticity that involve intimately coupling CFD and CSM tools.

3. COUPLED FLUID STRUCTURE ANALYSIS

In recent years, significant advances have been made for single disciplines in both computational fluid dynamics (CFD) using finite-difference approaches [9] and computational structural dynamics (CSD) using finite-element methods [10]. For aerospace vehicles, structures are dominated by internal discontinuous members such as spars, ribs, panels, and bulkheads. The finite-element (FE) method, which is fundamentally based on discretization along physical boundaries of different structural components, has proven to be computationally efficient for solving aerospace structures problems. The external aerodynamics of aerospace vehicles is dominated by field discontinuities such as shock waves and flow separations. Finite-difference (FD) computational methods have proven to be efficient for solving such flow problems. Figure 3 illustrates a time-accurate coupled fluid-structure aeroelastic analysis process. It is step-by-step time-integration procedure. Fluid and structural solutions are independently computed and the information is passed between them at common boundaries. At every time-step the pressure data (Cp) from CFD are mapped on to structural grid points and force vector {Z} is computed. Using Z, the structural displacements are computed from CSD analysis. Then deflections are mapped onto fluid grids that move accordingly. The interface techniques depend on the type of structural modeling.

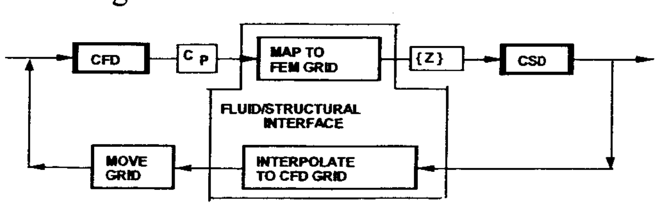

Fig. 3 Coupled fluid structural analysis

Fluids and structural domains can be modeled at various levels of complexity both in physics and geometry. For design, aerodynamic data may be used at several levels of fidelity starting from low-fidelity look-up tables to high fidelity Navier-Stokes solutions. Similarly for structures, the data can be obtained starting from low fidelity assumed shape functions and ending at detailed three-dimensional finite elements. As the fidelity of modeling increases, it becomes more difficult to handle complex geometry. Figure 4 illustrates the typical levels of modeling complexities involved both for fluids and structures. Interfacing techniques depend on the levels of fidelity in both fluids and structures. To date general purpose codes such as NASTRAN [11] can compute aeroelasticity of complex geometries using 3-D finite element structures directly coupled with linear analytical aerodynamic methods. Codes based on CFD such as HiMAP [12] can compute aeroelasticity using 3-D Navier-Stokes equations coupled with simple 3D finite element structural equations. In this effort, coupled computations using different levels of fidelity on HPCMP resources are studied

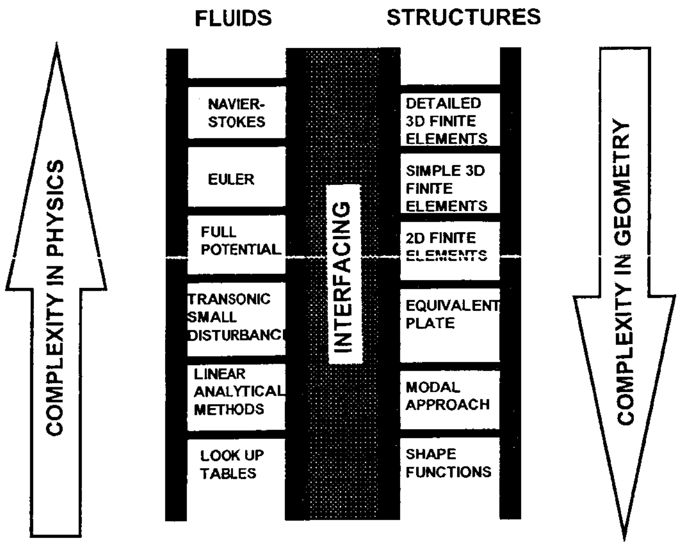

Fig. 4 Varying levels of fidelity in modeling for fluids and structures.

4. Approach for Study

1. Review all papers presented at the 2002 (Austin Texas) and 2003(Belleview, Washington) users group meeting of HPCMP in the area of fluid/structure interaction for aerospace application.
2. Interact with authors to obtain more information such as :
    a. Name of the software
    b. Flow equations used
    c. Size of the grid for fluids
    d. Structural equations used
    e. Number of elements
    f. Moving grid techniques
    g. Parallel efficiency of codes including algorithm convergence efficiency and single node processing efficiency
    h. Number cases ran
    i. Impact on design
    j. Name of the HPCMP resource used
3. Categorize the paper based on the fidelity of equations used and parallel computational efficiency
4. Provide a list of observations that may help future planning of HPCMP resources

In order to accomplish this an internet based questionnaire was generated after consulting with several lead researchers in the area of CFD and CSM. The questions were designed to capture the nuances of computational aeroelasticity and parallel computing. It is also designed to minimize the time taken to respond. Several options were given so that the responder can answer with a click of the button. A sample test performed by an experienced HPC user showed that it required about 30 minutes to complete. Appendix A is a copy of the questionnaire. The questionnaire was disseminated to all CTA members of CFD and CSM.

Appendix B lists authors that sent completed responses. This report is primarily based on information provided by responses and to some extent from those presented at HPCMP users conferences.

5 INDEXING THE APPLICATIONS

Computational expense increases with geometric complexity and fidelity of the equations solved. An attempt is made in this paper to provide a quantitative measure of expense associated with geometric complexity and computational modeling fidelity used in computational aeroelasticity. A simpler approach of assigning an index to expense is presented earlier by the author in Reference 13.

In this report a fidelity-complexity index (FCI) is assigned to each application. It is assumed that the complexity of the problem is represented by the number of intersecting surfaces of the geometry and grid size used for modeling flows and structures. It is also assumed that the complexity of the problem arising from intersecting surfaces has a strong impact on the CFD grid and no impact on FEM grids. The number of intersecting surfaces for typical aerospace configurations considered by HPCMP users is given in Fig. 5.

Fidelity of fluids modeling depends on the type of equations solved and the turbulence model used. For structures it depends on the type of element used. Level of fidelity for both fluids and structures can be measured by the number of floating point operations needed to solve equations. In this report fidelity for fluids is measured based on the number of floating point operations involved per grid point per local time step. The Euler option of the diagonal form of the Pulliam-Chaussee[14] scheme that requires about 1400 floating point operations per grid point is used as a reference number. The fidelity for structures can be measured using the number of degrees of freedom (DOF) per element. The 9-DOF triangular plate element most popularly used is used as reference.

In addition to cpu and memory requirements, need for other resources such as I/O significantly increases from steady state computations to dynamic aeroelastic computations. It is also accounted for in assessing FCI.

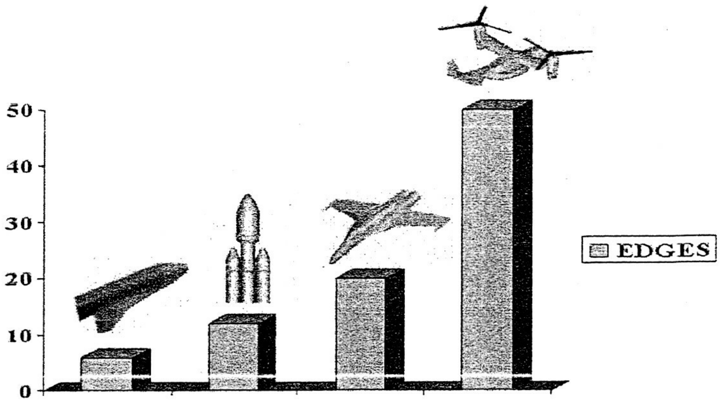

Fig. 6 Number edges for different aerospace configurations.

The Fidelity/Complexity Index (FCI)can be represented a

$$FCI = (Fp * Sp * Fg * Sg * Ne * SUA) \quad (1)$$

Fp = number of floating point operation per grid point per step divide by 1400
Sp = number of degrees of freedom per element divided by 9
Fg = fluids grid size divided by 100000.
Sg = structural elements divided by 1000 (only for strongly coupled cases)
Ne = number of edges divided by 10
SUA = for steady =1, unsteady =2, static aeroelastic =3, and dynamic aeroelastic=4

For example, a dynamic aeroelastic computation (SUA = 4) over a typical wing-body configuration (Ne = 10) using an Euler solver based on Pulliam-Chuassee [14] algorithm (Fp = 1400) with 1M grid points and 1000 triangular plate (Sp = 9) elements yields an FCI index of 40.

5.1 FLOW SOLVERS

Flow equations solved by CFD CTA members are either Euler, Reynolds averaged Navier-Stokes (RANS) or Direct Navier Solvers (DNS). Most popularly used RANS solvers incorporated either algebraic, one-equation, detached eddy simulation (DES) or large eddy simulation (LES) models for turbulence.

Three types of grid topologies were used. The most popular topology for the Euler equations is an unstructured grid and for the Navier-Stokes equations it is a patched structured grid. Some applications used Cartesian grids for the Euler and overset structured for Navier-Stokes equations. Some applications based on unstructured and Cartesian grids used embedded structured grids to capture turbulent flows. Figure 6 shows the distribution of use of different grid topologies based on the CFD papers presented at the 2003 HPCMP Users group conference[15]. There is a strong trend to use unstructured grids for solving Direct Navier-Stokes equations. Typical structured and unstructured CFD grids are shown in Figs. 7 and 8. Floating point operations per grid point per local time step for different flow equations are shown in Fig 8. Though the use of overset structured grid was relatively small it had unique capability to accurately solve RANS equations for configurations with large relative body movements such as store separation.[16]

CFD solvers have scaled well with improvements in the cpu speed. Parallel performance of most fluid solvers used in HPCMP, particularly MPI based, were also almost linear.

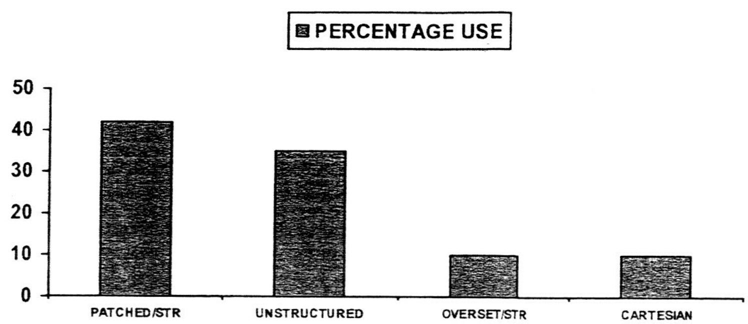

Fig. 6 Distribution of grid topology types in the latest CFD applications.

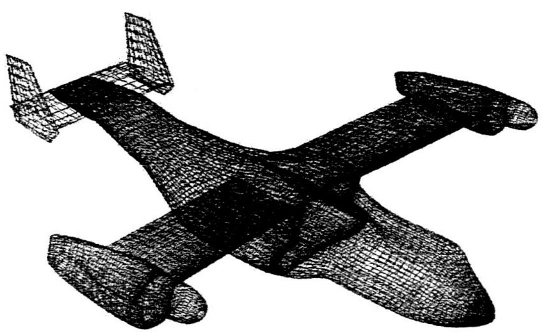

Fig. 7 Patched structured grid of size 3M for V-22 rotorcraft (from Tang, NRL).

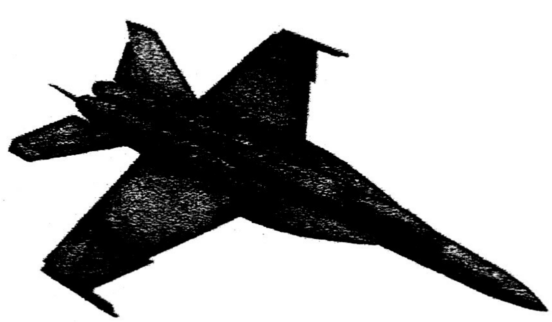

Fig 8 Unstructured grid for F18E/F aircraft (from Forsythe, Cobalt Inc)

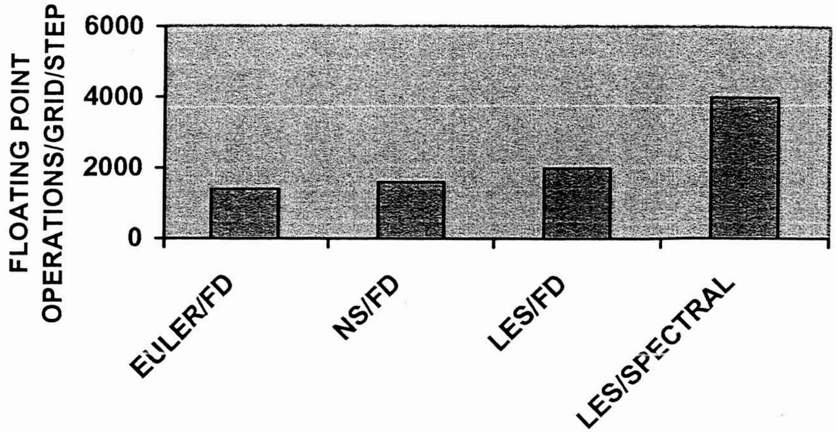

Fig 9. Computational needs for different flow equations

5.2 STRUCTURES SOLVERS

All structures codes were based on finite element method which use unstructured or irregular grids. The types of elements used in the applications were either 2D plate or 3D solid elements. The performance of structures solvers depend on the characteristics of the stiffness matrix whether it is sparse, banded or dense. It is configuration dependent unlike CFD matrices which are typically banded in nature. Therefore the performance measure of FEM codes was not as easy to measure or categorize as that for CFD. Typically it required one msec per step per degree of freedom on SGI 340 MHz Workstation.

5.3 PARALLEL METHODS.

Most of the CFD and FEM codes used MPI for parallelization. Use of Open-MP was limited to shared memory configuration such as SGI. There were some attempts to use Unix native message passing instructions which suffered portability.

It is often required either to refine the grid (h-method) or to increase the order of accuracy (p-method) to improve quality of solution. Due to faster convergence, the p-method is preferred over the h-method. However, the computational cost rapidly increases while using the p-method.

The straight forward approach to accomplish this is to utilize domain decomposition method and MPI. This requires rearrangement of zones which may be time consuming. On the other hand for shared memory systems one can combine MPI with Open_MP to avoid rezoning. Fig. 10 from Ref. 17 illustrates the advantage of combining MPI and Open_MP computations. Such a procedure is used in HPCMP applications for spectral and hp finite element discretizations [18]. The main draw back which may prevent the use of this technique is lack of portability.

Some applications use the UNIX based message passing protocol MPL that is less portable than MPI or OPEN_MP. However, MPL can perform better than MPI or OPEN_MP if optimized for a particular hardware. Based on a brief review of HPCMP applications the chart shown in Fig 11 has been generated showing relative use of different parallel protocols.

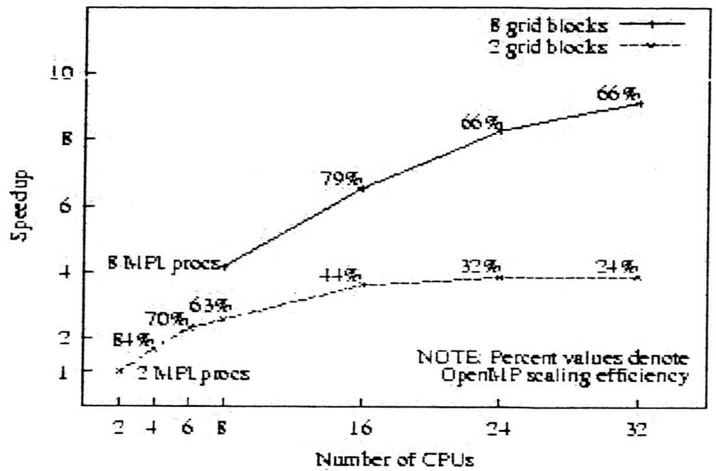

Fig. 10 Improved performance of Open_MP version of HiMAP on Sun platform (Ref 17)

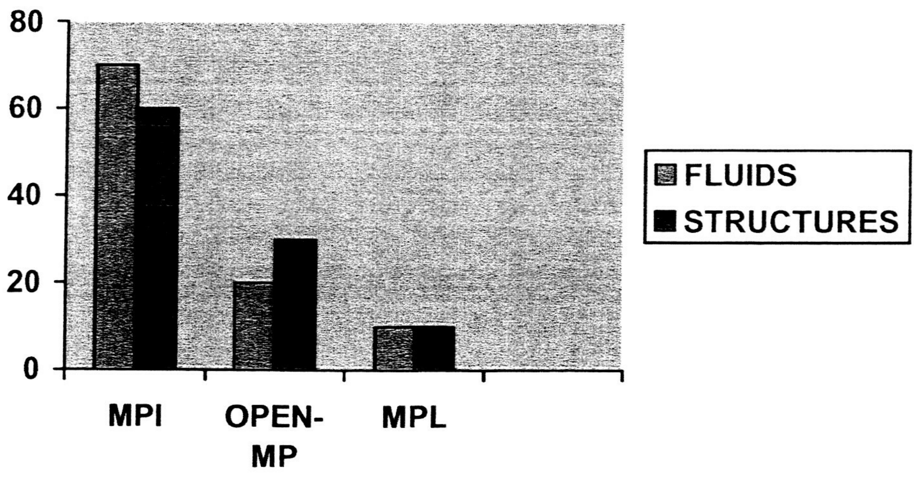

Fig. 11 Shows percentage distribution of use of MPI, Open_MP, and MPL

5.4 LOAD BALANCING

Zonal grids for both CFD and FEM are designed for accurate modeling of the configuration. Depending on the configuration the size of the grid in each zone can vary significantly. Larger variations occur when using the Navier-Stokes equations to solve viscous flows. Figure 12 shows the grid zone distribution for a full aircraft which has 35 zones and a total of 9.3M grid points. The number of grid points in the largest grid zone is 15 times that of the smallest grid zone. Simplistic grid zone mapping, which place each grid zone into a separate processor, leads to inefficient parallel performance. The inefficiency factor *e* in percent due to lack of load balancing can be expressed as

$$e = 100t / \{p*m*r\} \qquad (2)$$

where *t* the is total grid size, *m* is number of processors used, *p* is optimum grid size per processor, and *r* is the ratio of largest grid size to the smallest grid size. For the configuration shown in Fig 12 the efficiency is 6 percent where the optimum size per processor is 300K grid points on Origin 2000 system. It is assumed that each zone is assigned to one processor.

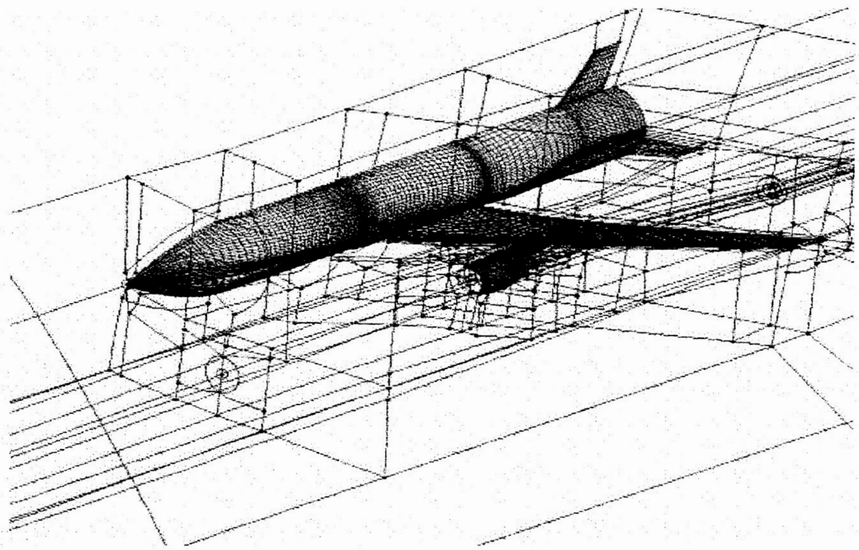

Fig. 12 Complex grid arrangement for a typical transport aircraft

When using complex configurations a load balancing scheme is needed. Procedures to load balance by splitting and coalescing zones, a simple approach and an advanced approach, are given in Ref. 19 and 20, respectively. Results of applying such a scheme to the configuration shown in Fig 12 are given in Fig 13. With the rearrangement the ratio of largest to smallest zone *r* is decreased to 1.23. Also, zones are grouped such that they can be assigned to 28 processors instead of 35. The efficiency factor *e* is increased from 6 percent to 87 percent.

Procedures needed for load balancing that can retain the characteristics of the original zonal arrangement are still under development. Patched structured grids are highly suitable for load balancing as illustrated in Ref. 20, both for MPI and Open-MP architectures. It is observed that none of the responses to the survey in the aeroelasticity area included load balancing approach. Some single discipline users stated that they used manual load balancing approach for which details were not available. Based on the data received typical efficiency factor values e computed for different grid sizes are shown in Fig. 14.

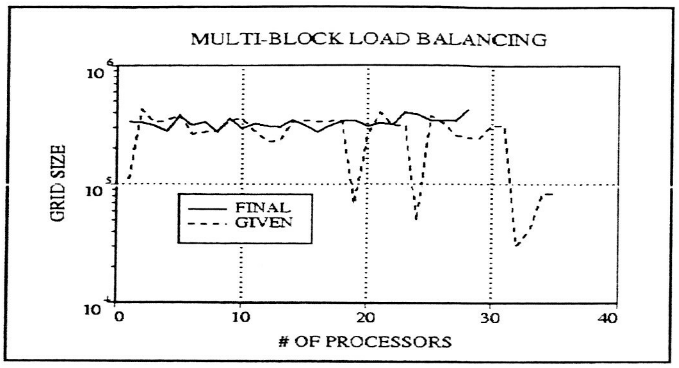

Fig. 13 Grid points per processor with and without load balancing scheme.

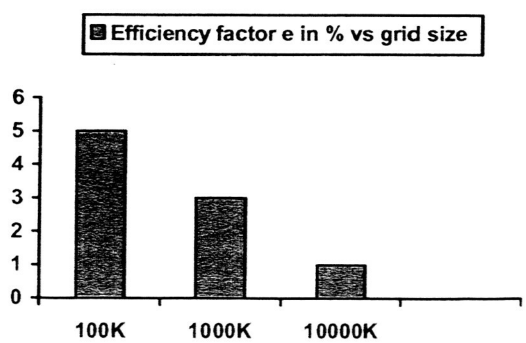

Fig 14. Decay of efficiency due to varying grids sizes

For unstructured grids automated tools such as METIS [21] were used by some CSM and CFD applications, but they did not report performance improvements. Use of automated load balancing methods such as METIS that may not account for original grid topological constraints may slow convergence, particularly for structures. It can happen if portions of zones involving of wing and body fall into same computational domain. Since the wing is flexible and the body is rigid a single stiffness matrix for both may become ill-conditioned.

5.6 HARDWARE

HPCMP provides a wide spectrum of hardware for its users. Most of the computational aeroelastic calculations were performed on a SGI, IBM or SUN computer system. Use of the IBM architecture was popular for MPI based CFD codes. Most Open_MP users utilized the SGI configuration. SUN platforms were used for both MPI and Open_MP type applications. More details about the hardware can be found on the HPCMP home page. The cpu speed of computer hardware has continuously increased and has impacted the performance of CFD and CSM solvers significantly. The cpu time in microsecond per step required by typical RANS solvers used in HPCMP applications is shown in Fig. 15 for single processors with different clock speeds. The improvement is almost linear.

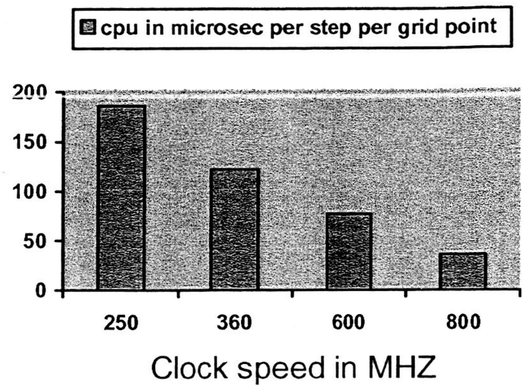

Fig. 15 Performance of RANS solver on different hardware.

6. CASE STUDIES

Based on the survey responses received from authors listed in appendix B the role of the HPCMP in advanced computational aeroelasticity was reviewed. For each case FCI was assigned based on the information given by the authors. A summary of results is shown in Fig 16. In Fig. 16 one of the advanced aeroelastic applications [22] sponsored formerly by NASA under High Performance Computing and Communication (HPCC) program is shown for comparison. Use of lower fidelity structural equations led to a lower FC index for most of HPCMP applications. However, rapid progress using unstructured grids tend to produce increasing values of FC index with further efficient utilization of advancing HPCMP resources.

7. CONCLUSIONS AND SUGGESTIONS

1. HPCMP resources have significantly advanced computational aeroelasticity.

2. Parallel performance of most fluid solvers used in HPCMP (particularly those using MPI) was almost linear.

3. For computational aeroelasticity, flow solvers have advanced in fidelity faster than structures. Flow solvers typically use the Navier-Stokes equations. However, structures use either modal or simple 2D plate elements.

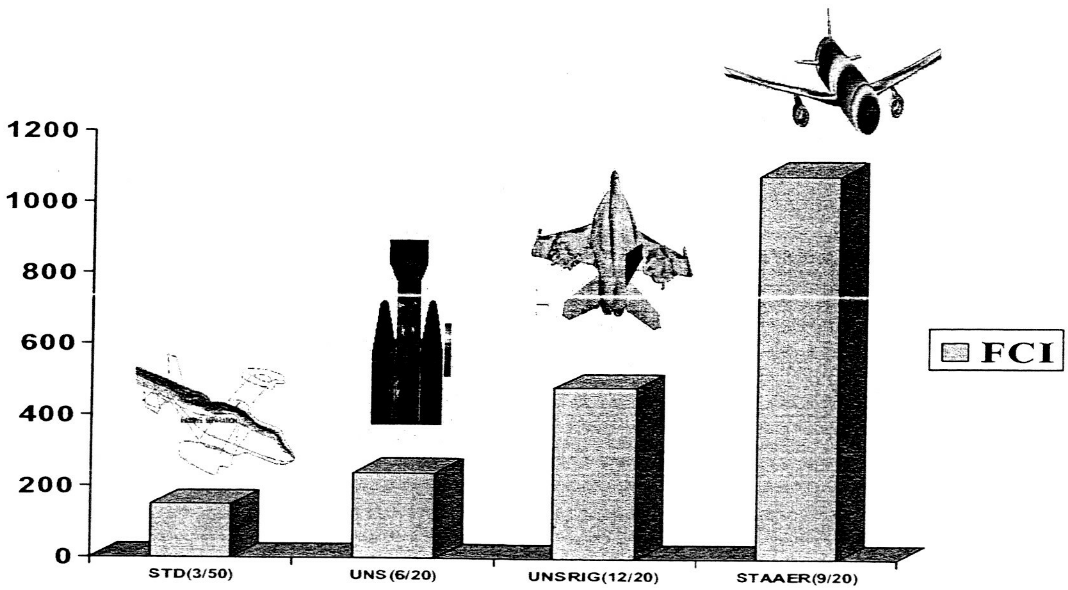

Fig. 16 Fidelity/Complexity Index (FCI) for typical aerospace applications. STD = Steady, UNS = Unsteady, UNSRIG = Unsteady flow over moving rigid configuration, STAAER = Static Aeroelasticity. (grid in M/edges)

4. Use of load balancing to significantly improve parallel computational efficiency was seldom used, particularly for fluids. Few structural applications used an automated partitioning tool. Significant increase in use of load balancing tools is needed.

5. Use of lower fidelity structural equations produced lower FCI values for most HPCMP applications. However, rapid progress is taking place in this area using unstructured grids for fluids and finite elements for structures.

6. None of the applications addressed aeroelastic optimization ostensibly due to large computational costs. Rapidly improving HPCMP resources can play a major role in making computationally intensive aeroelastic optimization practically feasible.

7. Some responses to this study reported results on local clusters. This information was not included in this report since clusters are currently not supported systems under HPCMP. However, there seems to be a rapidly growing interest for supercomputing users to move towards use of clusters. The main draw back of current clusters is that their reliability is highly dependent on local implementation and can rarely be duplicated in other locations unlike vendor supported tightly coupled MPP systems. A coordinated effort between HPCMP

and the NASA advanced supercomputing (NAS) division can make clusters a practical and reliable supercomputing resource for computationally intense aeroelasticity problems.

8. Automated web based tool for users to assess the fidelity complexity index can help to improve the computational performance and lead to more productive use of HPCMP resources

ACKNOWLEDGEMENTS

The author would like to thank Dr. Terry Holst of NASA Ames research Center for valuable consultations and Mr. Christopher Cheung a NASA Education associate for creating a user friendly web based questionnaire. The help from all the HPCMP resource users who completed survey responses is also appreciated. Partial funding support for this work was provided by the HPCMP office. The valuable guidance of Dr. Larry Davis, Deputy Director, HPCMP, led this effort to a successful conclusion. The continuous support and encouragement from Dr. Walt Brooks, Chief of the NASA Advanced Supercomputing (NAS) Division and Dr. Dochan Kwak, chief of NAS Applications Branch of Ames Research Center are both appreciated.

REFERENCES

1. Dobbs, S. K. and Miller, G. D.: Self-Induced Oscillation Wind Tunnel Test of a Variable Sweep Wing. AIAA Paper 85-0739, Apr. 1985.
2. Rizk, Y. and Guruswamy, G. P.: Numerical Investigation of Tail Buffet of F-18 Aircraft. AIAA-92-2673, AIAA 10th Applied Aerodynamics Conference, June 1992, Palo Alto, CA.
3. Eckstrom, C. V, Seidel, D. A and Stanford, M. C.: Measurements of Unsteady Pressure and Structural Response for an Supercritical wing. NASA TP 3443, Nov. 1994.
4. Farmer, M.G and Hanson, P. W.: Comparison of supercritical and Conventional Win Flutter Charcteristics. NASA TM X-72837, May 1976.
5. Grove, D.: Computational Fluid Dynamics Study of Abrupt Wing Stall on an F/A-18E. HPCMP User's Group conference 2002, June 10-14, 2002.
6. Findlay, D.: Numerical Analysis of Aircraft High Angle of Attack Unsteady Flows. AIAA 2000-1946, 6th AIAA/CEAS Aeroacoustics Conf., June 2000.
7. Jones, K., Platzer, M., Rodriguez, D. and Guruswamy, G. P.: On the effect of Area ruling on Transonic Abrupt Wing Stall. Symposium Transsonicum IV Gottingen, Sept. 2002.
8. Cray, H., Pfeiffer-Vega, Gabor, B.: The High Performance Computing Modernization program , CHIPS Magazine, Sept. 2003. (http://www.chips.navy.mil)

9. Buning, P. G., Wong, T. C., Dilley, A. D., and Pao, J. L. :Computational Fluid Dynamics Prediction of Hyper-X Stage Separation Aerodynamics. *Journal of Spacecraft and Rockets*, Vol. 38, No. 6, Nov.-Dec. 2001, pp. 820-826.

10. Yang, T. Y.: *Finite Element Structural Analysis.* Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1986

11. NASTRAN User's Manual: NASA SP-222 (08), June 1986.

12. Guruswamy, G. P., Byun, C., M. Potsdam, M., and Farhangnia, M.: User's Manual for HiMAP - A Multilevel Parallel Multidisciplinary Analysis Process. NASA TM-1999-209578, Sept 1999.

13. Guruswamy, G. P and Obayashi, S: Use of High Fidelity Methods in Multidisciplinary Optimization – A Preliminary Study. AIAA paper 2002-5638, AIAA MDO meeting, Atlanta Sept. 2002.

14. Pulliam, T. H. and Chaussee, D.S: A Diagonal Form of an Implicit Approximate Factorization Algorithm. *Jl. Of Computational Physics*, Vol. 39, No. 2, 1981, pp 347-361.

15. Computational Fluid Dynamics: Proceedings of the 2003 Users Group Conference, IEEE Computer Society, Los Alamitos, CA June 2003.

16. Martel, J. D, Applied Computational Fluid Dynamics (ACFD) in Supporting of Aircraft-Store Compatibility and Weapon Integration: Application to VIV. Proceedings of the 2003 Users Group Conference, IEEE Computer Society, Los Alamitos, CA June 2003.

17. Byun, C.: Porting Experience With HiMAP On Sun HPC Environment. SUN Users meeting, Hawai, April 2002.

18. Dong, S. Lucor, D., Symeonidis, V, Xu, J., and Karniadakis, G.E.: Multilevel Parallelization Models: Application to VIV. Proceedings of the 2003 Users Group Conference, IEEE Computer Society, Los Alamitos, CA, June 2003.

19. Hatay, F., Byun, C., Guruswamy, G.P., Jespersen, D., Gee, K and Rizk, Y.: A Multi-Level Parallelization Concept for High Fidelity Multiblock Solvers. Supercomputing 97, San Jose, Nov. 1997.

20. Guruswamy, G. P.: Development and Applications of a Modular Parallel Process for Large Scale Fluid/Structures Problems. HPC Asia 2002, Bangalore, Dec. 2002.

21. Schloegel, K., Karypis, G. and Kumar, V.: Unified Algorithm for Load-balancing Adaptive Scientific Simulations. *Supercomputing 2000, Nov 2000.*

22. Potsdam, M.A. and Guruswamy, G. P.: A Parallel Multiblock Mesh Movement Scheme for Complex Aeroelastic Applications. AIAA 2001-0716, Jan. 2001.

APPENDIX A: Survey Questionnaire

PERSONAL

Name:
Address:
Phone/Fax
Email/URL

GEOMTRY

☐ Wing ☐ Wing/body ☐ Wing/Body/Control ☐ Wing/body/empennage
☐ Full Vehicle
☐ Other

DISCIPLINES

☐ Fluids ☐ Structures ☐ Controls

FLUIDS

Equations Solved

☐ Navier-Stokes ☐ Euler ☐ Full Potential ☐ Linear

Turbulence Model

☐ Algebraic ☐ One Equation ☐ LES ☐ DNS ☐ None

Grid Topology

Structured : ☐ Patched ☐ Overset ☐ Combined; Total Grid Size
Number of Blocks Smallest Block Size Largest Block Size

Unstructured: ☐ Tetrahedra ☐ Hexahedra ;
Number of Elements Number of Nodes Number of Sub-Domains

Computational Efficiency (Fluids)

Cpu Time in micro secs per grid point per step :

Number of steps for steady state convergence or unsteady cycle :

Processor Speed in MHZ with precision ☐ 32 bit ☐ 64 bit

Parallel Method

☐ MPI ☐ OpenMP

Load Balancing : ☐ Yes ; Method ☐ No

Parallel Speedup for fixed problem size (fluids)

Total Processors :

Speed Factor :

STRUCTURES

Element Type

☐ 3D FEM ☐ 2D FEM ☐ Modal

Grid

Number of nodes Number of elements Number of sub blocks

Computational Efficiency (Structures)

Cpu Time in microsec per DOF per step :

Number of steps per per static or dynamic solution :

Processor speed in MHZ Precision ☐ 32 bit ☐ 64 bit

Parallel Method

☐ MPI ☐ OpenMP

Load Balancing : ☐ Yes ; Method ☐ No

Parallel Speedup for fixed problem size (structures)

Total Processors :

Speed Factor :

CONTROLS

☐ Time Domain Feed Back ☐ Frequency Domain ☐ Empirical/Other

HPCMP Computer System

Number of Unsteady Aero computations

Number of coupled computations

Maximum number of nodes

Total Node hrs used

Computer Used

Dates of Computations : Start End

PAPERS with URL

Title:

URL:

Title:

URL:

Title:

URL:

COMMENTS

Submit

Web page : Guru Guruswamy and Chris Cheung
Consultations : Terry Holst

APPENDIX – B : List of Authors Completed Survey

Project Lead	Organization
Tai	Naval Surface WCC
Engblom	Aerospace Corporation
Chen	Aerospace Corporation
Cartwright	Kirtland Airforce Base
Snyder	Wright Patterson Airforce Base
Polsky	Naval Air Warfare Center
Kunz	Penn State Army Research Laboratory
Charles	natick
Toporkov	Navy Research Laboratory
Moin	Stanford University
Madden	Airforce Research Laboratory
Namburu	Army Materials and Structures Research Laboratory
Vahala	Old Dominion University
Karniadakis	Brown University
Forsythe	Cobalt Inc
Melville	Wright Patterson Air Force Base
Sahu	Army Research Laboratory