# Using Computational Fluid Dynamics for Aerodynamics

*Antony Jameson and Massimiliano Fatica*
Stanford University

**Abstract**
In this white paper we survey the use of computational simulation for aerodynamics, focusing on applications in Aerospace and Turbomachinery. We present some representative problems to illustrate the range of complexity in fluid simulations and the associated computational requirements. We also examine the design process in current industrial practice, and the role played by computational fluid dynamics (CFD). Measured against this backdrop we assess the potential role and market for supercomputing in an environment of ubiquitous computing on the desktop. We also address some algorithmic and architectural issues, exemplified in Stanford's project to develop a new system using stream processors.

## Introduction

In a 1986 report from the National Research Council on "Current Capabilities and Future Directions in Computational Fluid Dynamics", it was stated "computational fluid dynamics is capable of simulating flow in complex geometries with simple physics or flow with simple geometries with more complex physics". This is not true anymore thanks to progress in computers and algorithm developments. 3D Euler calculations of flows for complex geometries that were "state of the art" in 1986 for both the hardware and software requirements, can now be carried out on laptops. CFD is widely accepted as a key tool for aerodynamic design. Reynolds Average Navier-Stokes (RANS) solutions are a common tool, and methodologies like LargeEddy Simulation (LES) that were once confined to simple canonical flows (isotropic turbulence in a box, channel flow), are moving to complex engineering applications. For example, the Center for Integrated Turbulence Simulations here at Stanford is using LES to simulate the reacting flow in a real combustor chamber of a jet engine.

**The complexity of fluid flows.**

The complexity of fluid flow is well illustrated in Van Dyke's Album of Fluid Motion. Many critical phenomena of fluid flow, such as shock waves and turbulence, are essentially nonlinear and the disparity of scales can be extreme. The flows of interest for industrial applications are almost invariantly turbulent. The length scale of the smallest persisting eddies in a turbulent flow can be estimated as of order of $1/Re^{3/4}$ in comparison with the macroscopic length scale. In order to resolve such scales in all three spatial dimensions, a computational grid with the order of $Re^{9/4}$ cells would be required. Considering that Reynolds numbers of interest for airplanes are in the range of 10 to 100 million, while for submarines they are in the range of $10^9$, the number of cells can easily overwhelm any foreseeable supercomputer. Moin and Kim reported that for an airplane with 50-meter-long fuselage and wings with a chord length of 5 meters, cruising at 250 m/s at an altitude of 10,000 meters, about 10 quadrillions ($10^{16}$) grid points are required to simulate the turbulence near the surface with reasonable details. They estimate that even with a sustained performance of 1 Teraflops, it would take several thousand years to simulate each second of flight time. Spalart has estimated that if computer performance continues to increase at the present rate, the Direct Numerical Simulation (DNS) for an aircraft will be feasible in 2075.

Consequently mathematical models with varying degrees of simplification have to be introduced in order to make computational simulation of flow feasible and produce viable and cost-flow equation to model the flow. Procedures for solving the full viscous equations are needed for the simulation of complex separated flows, which may occur at high angles of attack or with bluff bodies. In current industrial practice these are modeled by the Reynolds Average Navier-Stokes (RANS) equations with various turbulence models.

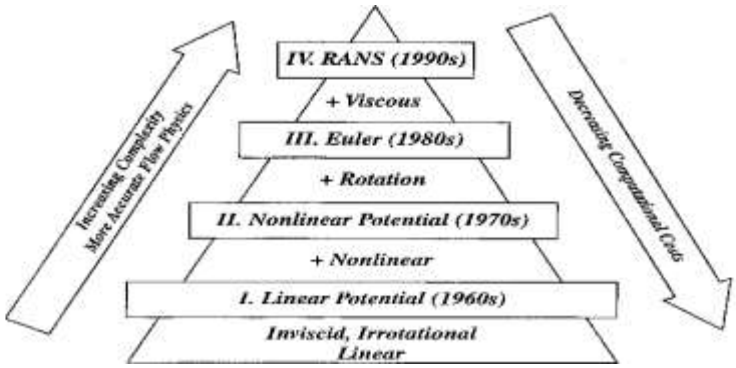


Figure 1: Hierarchy of models for industrial flow simulations

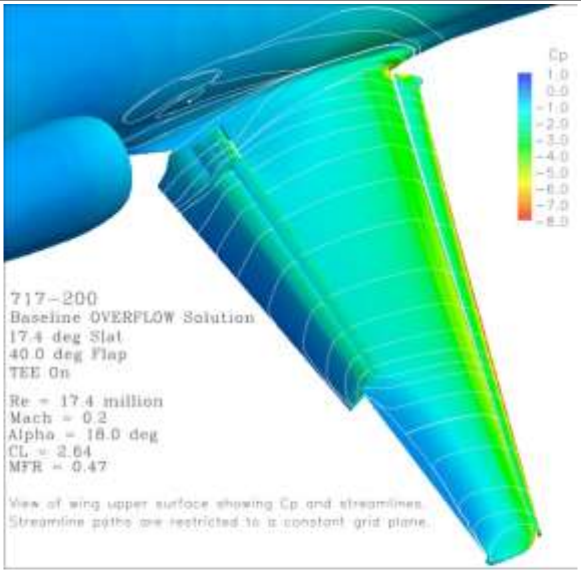
**Computational costs**

In external aerodynamics most of the flows to be simulated are steady, at least at the macroscopic scale. Computational costs vary drastically with the choice of mathematical model. Studies of the dependency of the result on mesh refinement, performed by this author and others, have demonstrated that inviscid transonic potential flow or Euler solutions for an airfoil can be accurately calculated on a mesh with 160 cells around the section, and 32 cells normal to the section. Using a new non-linear symmetric Gauss-Siedel (SGS) algorithm (Jameson and Caugley, 2001), which has demonstrated “text book” multigrid convergence (in 5 cycles), two-dimensional calculations of this kind can be completed in 0.5 seconds on a laptop computer (with a 2Ghz processor). A three dimensional simulation of the transonic flow over a swept wing on a 192x32x32 mesh (196,608 cells) takes 18 seconds on the same laptop. Moreover it is possible to carry out an automatic redesign of an airfoil to minimize its shock drag in 6.25 seconds, and to redesign the wing of a Boeing 747 in 330 seconds.

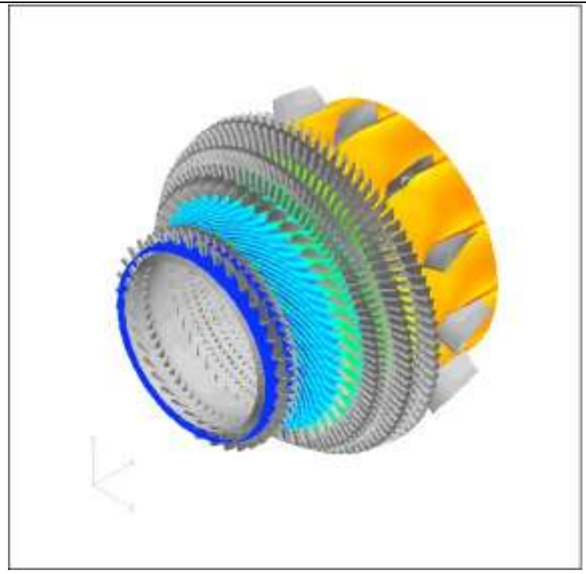
Viscous simulations at high Reynolds numbers require vastly greater resources. On the order of 32 mesh intervals are needed to resolve a turbulent boundary layer, in addition to 32 intervals between the boundary layer and the far field, leading to a total of 64 intervals. In order to prevent degradations in accuracy and convergence due to excessively large aspect ratios (in excess of 1,000) in the surface mesh cells, the chordwise resolution must also be increased to 512 intervals.Translated to three dimensions, this implies the need for meshes with 5-10 million cells (for example, 512x64x256 = 8,388,608 cells) for an adequate simulation of the flow past an isolated wing. When simulations are performed on less fine meshes with, say, 500,000 to 1 million cells,it is very hard to avoid mesh dependency in the solutions as well as sensitivity to the turbulence model. Currently Boeing uses meshes with 15-60 million cells for viscous simulations of commercial aircraft with their high lift systems deployed. Using a multigrid algorithm, 2000 or more cycles are required to reach a steady state, and it takes 1-3 days to turn around thecalculations on a 200 processor Beowulf cluster.

A further progression to large eddy simulation of complex configurations would require even greater resources. The following estimate is due to W. H. Jou of the Boeing Company. Suppose the mesh interval should then be 1/50 of the boundary layer thickness. Moreover, since the eddies are three-dimensional, the same mesh interval should be used in all three directions. Now, if the boundary layer thickness is of the order of 0.01 of the chord length, 5,000 intervals will be needed in the chordwise direction, and for a wing with an aspect ratio of 10, 50,000 intervals will be needed in the spanwise direction. Thus, of the order of 50 x 5,000 x 50,000 or 12.5 billion mesh points would be needed in the boundary layer. If the time dependent behavior of the eddies is to be fully resolved using time steps on the order of the time for a wave to pass through a mesh interval, and one allows for a total time equal to the time required for waves to travel three times the length of the chord, of the order of 15,000 time steps would be needed. A more refined estimate which allows for the varying thickness of the boundary layer, recently made by Spalart suggests an even more severe requirement. Performance beyond the teraflop ($10^{12}$ operations per second) will be needed to attempt calculations of this nature, which also have an information content far beyond what is needed for engineering analysis and design. The main current use of DNS and LES is to try to gain an improved insight into the physics of turbulent flow, which may in turn lead to improved turbulence models.

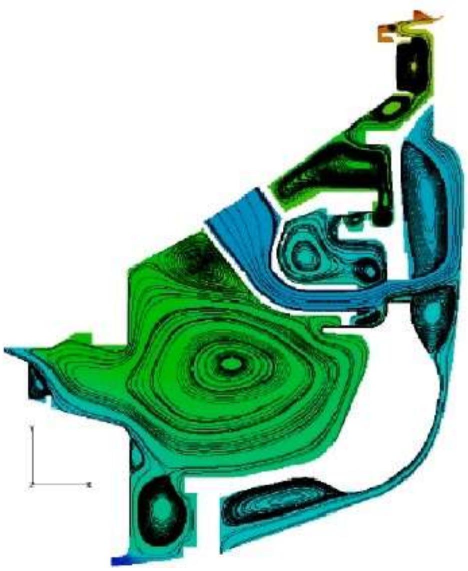
There are also important industrial applications where the flow is inherently unsteady, with a corresponding increase in the computational complexity even when using the RANS equations. One example is the simulation of a helicopter rotor in forward flight for which it would be necessary both to calculate the dynamic and aerolastic blade motions, and to track their trailing vortices. Of the order of 100 million mesh cells would be needed. Another example is the simulation of turbomachinery. A jet-engine compressor typically contains of the order of 1000 passages in about 30 interleaved rows of rotating and fixed blades. While a smaller number of stages are needed in the turbine, a complete simulation ought to treat film cooling via numerous small holes in each blade, and transitional flow. In Stanford’s ASCI Alliance center we have been calculating the unsteady flow through the complete turbine of the Pratt and Whitney 6000 engine, which has 9 blade rows. The computational mesh for this simulation, illustrated in the following table, contains 94 million mesh cells. Using a fully implicit dual time stepping scheme with a second-order accurate backward difference formula (BDF), the calculation, which is still ongoing using 512 processors of an ASCI machine, requires of the order of 3 million CPU hours. The prohibitive computational cost of simulations of this magnitude rules out their industrial use.



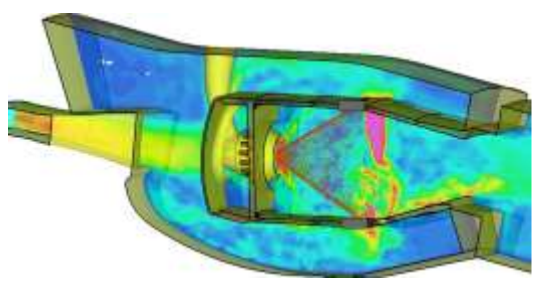

High lift configuration. 22 million cells solution using Overflow (courtesy of Boeing)



PW6000 turbine, unsteady simulation with 94 million cells using TFLO (CITS, Stanford).



Secondary system in the high pressure turbine of a PW6000 engine (CITS, Stanford)



Large Eddy Simulation in a PW6000 combustor (CITS, Stanford)

**The role of CFD in the design process**

The actual use of CFD by Aerospace companies is a consequence of the trade-off between perceived benefits and costs. While the benefits are widely recognized, computational costs can not be allowed to swamp the design process. The need for rapid turnaround, including the setup time, is also crucial.

In current industrial practice, the design process can generally be divided into three phases: conceptual design, preliminary design, and final detailed design, as illustrated in Figure 2. The conceptual design stage, typically carried out by a staff of 15-30 engineers, defines the mission in the light of anticipated market requirements, and determines a general preliminary configuration, together with first estimates of size, weight and performance. The costs of this phase are in the range of 6-12 million dollars.

In the preliminary design stage the aerodynamic shape and structural skeleton progress to the point where detailed performance estimates can be made and guaranteed to potential customers, who can then, in turn, formally sign binding contracts for the purchase of a certain number of aircraft. A staff of 100-300 engineers is generally employed for up to 2 years, at a cost of 60-120 million dollars. Initial aerodynamic performance is explored by computational simulations and through wind tunnel tests. While the costs are still fairly moderate, decisions made at this stage essentially determine both the final performance and the development costs.

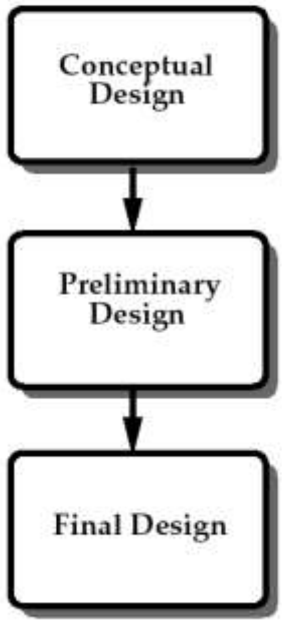



Figure 2: Phases of design

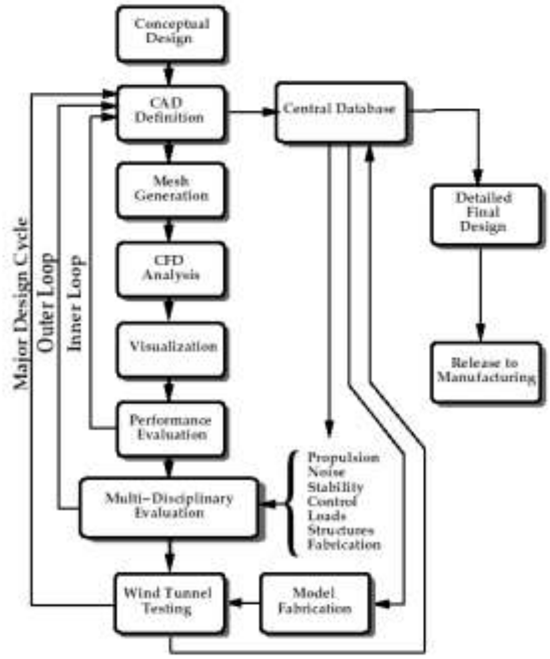



Figure 3: Overall preliminary design

In the final design stage the structure must be defined in complete detail, together with complete systems, including the flight deck, control systems (involving major software development for fly-by-wire systems), avionics, electrical and hydraulic systems, landing gear, weapon systems for military aircraft, and cabin layout for commercial aircraft. Major costs are incurred at this stage, during which it is also necessary to prepare a detailed manufacturing plan. Thousands of engineers define every part of the aircraft. Total costs are 3-10 billion dollars. Thus, the final design would normally be carried out only if sufficient orders have been received to indicate a reasonably high probability of recovering a significant fraction of the investment.

In the development of commercial aircraft, aerodynamic design plays a leading role during the preliminary design stage, in the course of which the definition of the external aerodynamic shape is typically finalized. The aerodynamic lines of the Boeing 777 were frozen, for example, when initial orders were accepted, before the initiation of the detailed design of the structure. Figure 3 illustrates the way in which the aerodynamic design process is embedded in the overall preliminary design. The starting point is an initial CAD definition resulting from the conceptual design. The inner loop of aerodynamic analysis is contained in an outer multi-disciplinary loop, which is in turn contained in a major design cycle involving wind tunnel testing. In recent Boeing practice, three major design cycles, each requiring about 4-6 months, have been used to finalize the wing design. Improvements in CFD, might allow the elimination of a major cycle, would significantly shorten the overall design process and reduce costs.

Moreover, the improvements in the performance of the final design, which might be realized through the systematic use of CFD, could have a crucial impact. An improvement of 5 percent in lift to drag (L/D) ratio directly translates to a similar reduction in fuel consumption. With the annual fuel costs of a long-range airliner in the range of \$5-10 million, a 5 percent saving would amount to a saving of the order of \$10 million over a 25 year operational life, or \$5 billion for a fleet of 500 aircraft. In fact an improvement in L/D enables a larger. Furthermore a small performance advantage can lead to a significant shift in the share of a market estimated to be more than $1 trillion over the next decades.

In order to realize these advantages it is essential to move beyond flow simulation to a capability for aerodynamic shape optimization (a main focus of the first author research during the past decade) and ultimately multidisciplinary system optimization. Figure 4 shows the result of an automatic redesign of the wing of the Boeing 747, which indicates the potential for a 5 percent reduction in the total drag of the aircraft by a very small shape modification. It is also important to recognize that in current practice the setup times and costs of CFD simulations substantially exceed the solution times and costs. With presently available software the processes of geometry modeling and grid generation may take weeks or even months. In the preliminary design of the F22 Lockheed relied largely on wind-tunnel testing because they could build models faster than they could generate meshes. It is essential to remove this bottleneck if CFD is to be more effectively used. There have been major efforts in Europe to develop an integrated software environment for aerodynamic simulations, exemplified by the German "Megaflow" program.

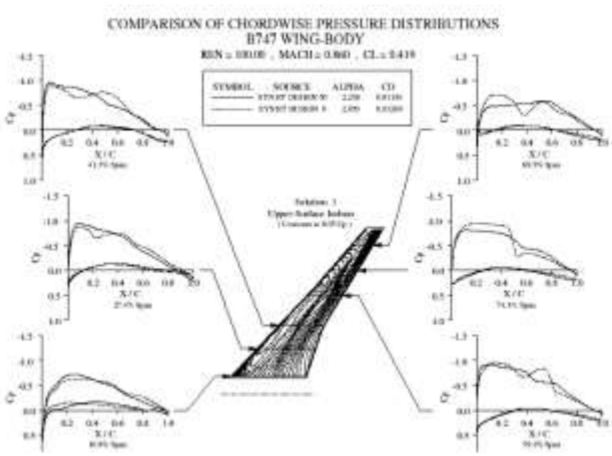



Figure 4: Redesigned Boeing 747 wing at Mach 0.86, Cp distributions

In the final-design stage it is necessary to predict the loads throughout the flight envelope. As many as 20000 design points may be considered. In current practice wind-tunnel testing is used to acquire the loads data, both because the cumulative cost of acquisition via CFD still exceeds the costs of building and testing properly instrumented models, and because a lack of confidence in the reliability of CFD simulations of extreme flight conditions.

**CFD algorithms and software**

Commercial CFD software is widely available, and now amounts to an industry with annual revenues in the range of $200 million. The best known examples (CFX, Fluent and Star-CD) all had their origin in England. Commercial software, however, has yet to gain acceptance as a design tool in the Aerospace Industry, which continues to use community codes, many of them developed by government agencies such as NASA, ONERA and the DLR. Once a code has been adopted, users are very reluctant to switch to a new code because of the large investment in familiarization and validation. Accordingly software tends to have a much longer operational life than hardware. For example, FLO22, written by Jameson and Caughey in 1975, has continued to be extensively used to the present day.

The driving force in the development of CFD through the eighties was the design of shock capturing schemes which could resolve shock waves in one or two mesh cells without producing spurious oscillations. Complete success was achieved with the introduction of TVD, LED, ENO and WENO schemes. The need to treat very complex geometric configurations also poses a severe challenge. Body fitted structured meshes provide good resolution of boundary layers, but it is extremely difficult and time consuming to generate these meshes for configurations like, for example, the Space Shuttle at launch. The difficulty of mesh generation may be alleviated by the use of overset meshes as in NASA's OVERFLOW, but automatic generation of structured meshes remains out of reach. In the case of inviscid flow simulations, one response is a trend towards the use of Cartesian meshes, which are amenable to automatic generation (Boeing's TRANAIR, Lockheed's SPLITFLOW, NASA's CART3D). The other approach which is being increasingly accepted is to use unstructured meshes with tetrahedral or mixed polyhedral cells. Examples include the author's AIRPLANE code, introduced in 1986, but still in use at NASA Ames, EADS' AIRPLANE+, a derivative of AIRPLANE, the DLR's TAU code, NASA's USM3D and FUN3D, and CFD++, offered by Metacomp. The use of unstructured meshes alleviates (but not entirely eliminates) the difficulty of mesh generation and facilitates adaptive mesh refinement (AMR). It is harder to formulate accurate viscous discretizations, and also higher-accurate discretizations become very complicated. This has motivated widespread current interest in discontinuous Galerkin schemes, which offer the prospect of higher order accurate discretization with a compact stencil.

Time stepping methods have proven to offer an expedient route to the calculations of steady as well as unsteady flows. However, simple explicit schemes require many thousands of time steps to reach a steady state. This has motivated the introduction of a variety of alternative methods, including alternating direction implicit (ADI) schemes, LU implicit and symmetric Gauss- Seidel (SGS) schemes, and acceleration techniques such as Jacobian free Newton-Krylov and multigrid procedures. The choice of an algorithm cannot be made without considering the computer architecture. There is nothing inherently preventing parallelism in flow simulation algorithms, but parallelism may be lost in the formulation of implicit schemes. At the beginning of the eighties, with the expectation that future computing platforms would be parallel, the first author focused on the development of explicit schemes, using modified Runge-Kutta methods with enlarged stability regions, embedded in a multigrid procedure. This approach, which is easily adaptable to arbitrary grids, and allows complete parallelization, continues to be widely used both in the USA and Europe. However, recent results of Jameson and Caughey clearly demonstrate that Gauss- Seidel methods can be about 5 times faster, and would be preferred for calculations on single processor machines. Ultimately it seems that the best performance could be attained by an algorithm that uses the latest accessible data to update the solution at all times.

In many unsteady flow of interest, the time scales that need to be resolved are much larger than the acoustic time scale. In this situation the authors believe that the most efficient approach is to use a fast steady-state solver to perform the inner iterations of a fully implicit scheme using a backward difference formula (BDF). In Stanford's ASCI alliance center simulations of the turbine and compressor, we use a "dual time stepping" scheme of this kind, which inherits the parallelism of the solver used in the inner iterations.

In the case of a periodic unsteady flow we believe that there are advantages in

**Aerodynamic performance prediction**

The state-of-the-art in CFD drag prediction was recently assessed by an international workshop on the subject. Figure 5 provides the 28 drag polars resulting from this drag prediction workshop (DPW). With the exception of a few out-layers, the computed polars fall within a band of about 7% the absolute level. The slopes are nearly identical. When comparing the CFD results with thetest data, we note that the CFD solutions were all run assuming fully turbulent flow, while the test data were collected with laminar runs on the wing up to transition strips on both upper and lower surfaces. To quantify the shift in drag associated with this difference, several independent calculations were performed yielding 12-13 counts higher drag levels for the fully turbulent flows. Accounting for this adjustment, the center of the CFD drag polars band coincides with themean of the test polars.

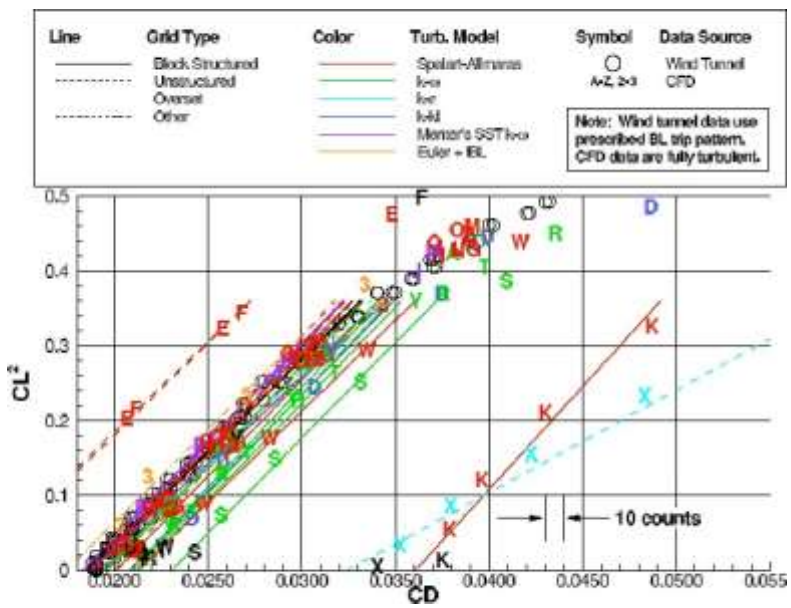



Figure 5: Results from the Drag Prediction Workshop

While this indicates that the industry as a whole is closing in on the ability to compute accurate absolute drag levels, in general, the errors are not to the level desired by aircraft design teams. However, a few of the results submitted to the DPW fall within the uncertainty band of the experimental data. Achieving this level of accuracy is dominated by the quality of the underlying grid, but also depends on the turbulence model, the level of convergence, discretization scheme, etc. It is imperative that each of these areas be studied independently of each other, otherwise "accurate" results might be obtained as a consequence of cancellation of errors. Unfortunately, an optimization based on an analysis method containing such a cancellation of errors will most likely emphasize its weakness and probably yield a new design with a false performance improvement.

**The current supercomputer scenario**

In the last decade, we have seen a departure from the "old" vector supercomputer model. Until the advent of the Earth Simulator (ES), the top supercomputers in the world were just "clusters on steroids", a collection of commercial servers or workstations interconnected by high-speed network. While these super-clusters have theoretical peak performance in the Teraflops range, sustained performance with real applications is far from the peak. Salinas, one of the 2002

Gordon Bell Awards, was able to sustain 1.16 Tflops on ASCI White (less than 10% of peak). Sustained performance in the single digit is the rule not the exception. The Earth Simulator and the Cray X1, two custom engineered systems with exceptional memory bandwidth, interconnect performance and vector-processing capabilities, are pushing the idea of real supercomputer back on the stage. A global atmospheric simulation was able to achieve 65% of the peak performance of the Earth Simulator and other CFD simulations were in the 30-50% range.

The main limitation of the clusters built from commercial servers and workstation is the limited memory bandwidth that these platforms offer. In CFD, most of the algorithms do not reuse the data, and the benefit of cache (essentially a memory bandwidth amplifier) cannot be used. The situation is even more severe for codes using unstructured meshes, often a necessity in the treatment of very complex geometries. Indirect addressing (necessary to handle the complex data structures) just kills the performances. Vector machine were much more efficient on these codes, thanks to the vector load/store instructions that could address noncontiguous memory locations.

While in the old Cray days, it was the norm to sustain 50% of peak performance, now we are happy when we get 10%! Nevertheless the clusters have an important role in these days. They are an economic way of providing fast turn-around time, more memory and decent levels of performances. Commercial vendors are also starting to pay attention to the memory bandwidth.

**Visions for the future**

The high expense of custom-built machines like the ES or the Cray X1 is a fundamental limitation of massively parallel vector supercomputers. The Blue Planet Virtual Vector Architecture and machines based on the "system-on-a-chip" idea, such the Blue Gene/L CPU, are two examples of alternative approaches being explored to obtain great performance at reduced cost.

The Imagine chip, developed by Stanford University's Computer Systems Laboratory, has shown the promise of a streaming processor for signal and image processing in recent years. A new, high-performance architecture might be constructed using stream processors as its foundational components. Under the direction of Professors Dally and Hanrahan, the Stanford Streaming Supercomputer (now known as Merrimac) project aims to achieve top-tier performance by integrating stream processors, a high-performance interconnection network that efficiently supplies good global bandwidth, and a novel programming paradigm to fully take advantage of this advanced architecture. Final hardware should be scalable from a 2 Tflops workstation to a 2 Pflops machine-room size computer with up to 16K processors. Although the hardware and language group work closely together to define the hardware and language standards, the applications development team works closely with them as well. Using the streaming paradigm, the authors have converted two CFD programs, StreamFLO (a multigrid finite volume Euler solver) and StreamFEM (a finite-element code), to Merrimac (coded by Tim Barth at NASA Ames). The results of the first tests are promising.

Improvements in computer hardware may also be seen from a different perspective. During the last 15 years, the size and cost of a computer adequate for the bulk of engineering simulations have progressively decreased while the sustained performance obtained by the best supercomputers has continuously increased. In 2001, a Sony Vaio 505 laptop computer weighed 4 pounds and cost $3,000, however it had the same processing power and memory as a Convex desktop computer from 1986, which weighed over 1,000 pounds and cost $600,000. If current trends continue over the next 15 years, a wristwatch-sized device selling for roughly $15 will have the computing power to do RANS simulations of the flow over a wing. Although data transmission needs

and visualization may rule out computers in wristwatches, but we may still expect a future where every engineer has access to the desktop computing capacity necessary to do aerodynamic and interdisciplinary research and design. While it is unrealistic to expect businesses to invest $20-$100 million on a supercomputer, they may augment this with clusters to enable computations needing a big throughput, such as aerodynamic tools. In light of this, it seems that government action, maybe in the form of acquisitions for government laboratories, may be required to maintain a sustainable market for supercomputers. In this light, it becomes clear that adopting a scalable architecture, where the same hardware and software components are shared by machines of varying sizes (from desktops to departmental servers to the greatest supercomputers), offers the best chance of achieving long-term economic viability.

**References:**

Album of Fluent Motion, by M. Van Dyke, Parabolic Press, 1982.

"A viewpoint on computational methods for aerodynamic analysis and design," by A. Jameson, appeared in Advances in Aerospace Sciences, volume 37, pages 197-243, 2001.

15th AIAA Computational Fluid Dynamics Conference, June 11-14, 2001, Anaheim, CA. A. Jameson and D.A. Caughey, "How Many Steps are Needed to Answer the Euler Equations of Steady Compressible Flow: In Pursuit of a Quick Solution Algorithm," AIAA 2001-2673.

Science Magazine, January 1997, volume 276 issue 1; pages 62-68; P. Moin and K. Kim, "Tackling turbulence with supercomputers."

AIAA paper 2004-1090, "STREAMFLO: an Euler solution for streaming architectures," will be presented by M. Fatica, A. Jameson, and J.J. Alonso at the 42nd Aerospace Sciences Meeting and Exposition Conference in Reno, Nevada, January 2004.