A98-31518

ICAS-98-2,7,5

A SYSTEMS APPROACH TO CFD CODE DEVELOPMENT

Wen-Huei Jou*
Boeing Commercial Airplane Group

Abstract

We discuss the idea of developing a CFD code using a systems integration approach. The approach considers identification of intended applications, determination of functional requirements, system architecture, selection/development of components, and trades of component technology. This is in contrast to the approach of connecting research components into an engineering code. This systems approach is illustrated by the successful transonic cruise design and analysis code, TRANAIR. We then identify Navier-Stokes analysis for separated flows as the key capability to support providing aerodynamic data base in an airplane development process. This new challenge requires further algorithm research. The necessary algorithm advancements are discussed.

Introduction

Computational Fluid Dynamics is now an important discipline in aerospace technology. A rough estimate of 4,000 people involved in CFD activities worldwide will annually spend approximately one billion U.S. dollars. These activities include algorithm research, physics modeling, code development, validation of code capabilities. This paper addresses the code development aspect, because it is the most important link in technology transfer from research to engineering applications.

We observe that many of code development activities start from a solver component which researchers developed to demonstrate the success of their algorithm technologies. By connecting with a grid generation component from another researcher, and by developing some pre-processing and post processing components, an engineering code is formed. This mode of code development had been quite successful in the early days of CFD when even an analysis capability for a simple geometry could greatly contribute to the improvement of engineering processes. The advances in CFD technology and in computers now allow us to consider much more complex geometry and much more complex fluid flows. Therefore, the engineering applications are increasingly more complex. With little understanding of how the code will be used in an engineering environment, this mode of code development often produces a code with limited capabilities and with a short shelf life.

This paper advocates a systems approach to code development. Much similar to developing an airplane or any other complex system, this approach contains gathering engineering requirements, evaluating algorithm technologies, designing code architecture, perform technology trades among the components, developing missing component technologies, and code integration. A code developed through this process will be a tightly integrated system in which components are chosen and developed through the metrics of optimizing the system performance. We illustrate this approach by the successful TRANAIR code [1,2] which was developed for cruise shape design and analysis for commercial airplane.

We then proceed to examine in which manner a Navier-Stokes code may contribute to the improvement of our engineering processes. The key ability of a Navier-Stokes code is to analyze separated flows for complex geometry. This capability will contribute to the development of airplane configuration, to providing critical airplane stability characteristics and control authority, and to providing critical loads information. We examine the current level of Navier-Stokes technology and identify the critical CFD research issues which will enable the development of such a code.

CFD for Transonic Cruise Shape Definition - A Success Story

Among all the tasks performed by aerodynamicists in an airplane development process, cruise shape definition has the highest impact on the success of the airplane program, i.e. satisfying payload and range requirements while reducing airline operating cost. Because of the extremely sensitive nature of three-dimensional transonic flows, it had also been one of the most difficult tasks using wind tunnel testing as the major design tool. Therefore, most of CFD development and application effort in the commercial airplane industry has been devoted to this task, since the break-through research of Murman and Cole[3]. Since then, CFD transonic design method has gone through several generation of development and has been exploited very successfully to improve wing performance and to reduce the design flow time. It is not an exaggeration to claim that CFD cruise shape design capability has completely revolutionized the aerodynamic design of a transport wing. The first generation of Boeing CFD wing design code which was applied to B777, was based on a pressure matching method using a full-potential

* Manager, Aerodynamic Research, Boeing Commercial Airplane Group, P.O. Box 3707, Seattle, WA 98124, USA.,

21st ICAS Congress
13-18 September 1998
Melbourne, Australia

formulation[4] coupled with a boundary layer[5]. Only simple configurations such as wing-body combination were considered. Engine integration was then performed with the assistance of TRANAIR code[1] which was capable of analyzing complex geometry at that time. This first generation CFD wing design process is described by the Figure 1. The method designs a wing surface shape that will produce a desired pressure distribution at the cruise conditions. The off design conditions are considered by compromising the cruise pressure distribution with those at off-design conditions through repeated iterations[6]. Notice that the constraints imposed by the structures and the manufacturing requirements are treated as post processing of the geometry generated by the design code. Since transonic

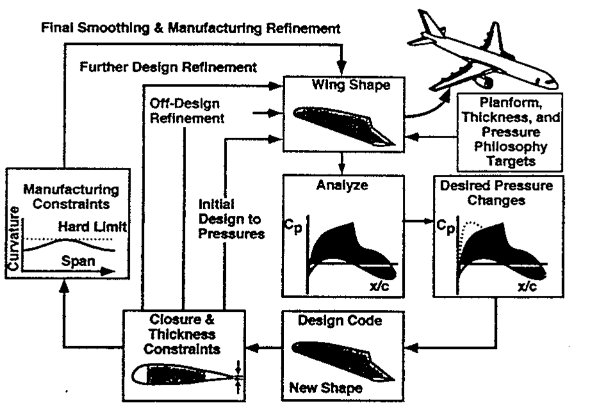

Figure 1 The First Generation CFD Wing Design Process

flows are notoriously sensitive to any change in surface geometry, this post processing often loses a good design. Many repeated design iterations were needed in order to achieve aerodynamic design objectives and at the same time produce a wing which can be manufactured in the factory. This first generation capability had already shown tremendous benefit over the cut-and-try wind tunnel testing process. In fact, a modern supercritical wing design can not be implemented on an airplane without CFD design capability. Recently, CFD design capability has reached a pinnacle with the maturity of the multiple-point design optimization method using TRANAIR code[2]. This new process is described by the Figure 2.

There are several important distinctive advantages of the new process over the first generation process. The previous generation of CFD design code can only handle wing-body geometry. As we target higher cruise Mach number and use high by-pass ratio engines, it is increasingly difficult to achieve a good wing design using a code which can only handle wing-body combination. It is desirable to account for the engine interference when designing a wing. TRANAIR complex geometry capability now allows us to do so. It is also very desirable to remove the iterative process caused by the consideration of the off-design and manufacturing constraints in the previous process.

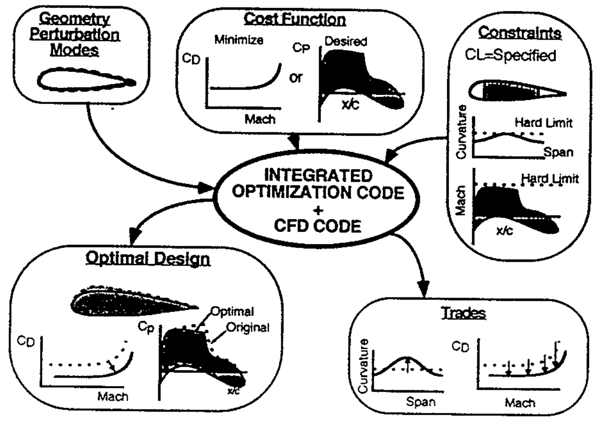

Figure 2 The Second Generation CFD Transonic Design Process

With TRANAIR optimization method, the geometry constraints and the off-design performance are treated as part of the formulation. No iterative process is required. The flow time to complete the entire task is greatly reduced, even though the wall clock time for an optimization exercise in the computer is much longer than that of one pressure matching design. Furthermore, TRANAIR code deals directly with the aerodynamic design objective function. It produces a better performing wing than that designed using an indirect indicator such as desirable pressure distribution.

Today, one of the most difficult aerodynamics tasks in airplane development in the past, has become one of the most straight forward ones because of the advances in CFD during the last three decades. We believe that we can now close a glorious chapter on the development of cruise design CFD code. Further investment in this area will yield a diminishing return.

The objective of this paper is not to discuss the success of wing design using such a code. Instead, by looking back on how the development team has accomplished this, we can learn the successful strategy which may be applied to the next challenge in CFD development. If there is a single most important reason for this success, it is that the TRANAIR code was designed and developed as a system. This is in contrast to many code development efforts which amount to connecting research code components. Since I have not been personally involved in technical decisions during the TRANAIR code development process, the following observations are offered as that from the manager overseeing the development since 1989. Although where TRANAIR ends up as a system may be quite different from where it started, the philosophy of a system approach has been an important aspect of the development process. As an example, the final TRANAIR system even includes a flutter prediction capability which was not an intended target for application. However, the team was able to modify the system to accommodate the new requirements and the new opportunity as they were identified.

The first characteristic of a system design is an understanding of the requirements for the intended applications. At cruise condition, a transonic or supersonic commercial airplane must be designed to a low drag level, i.e. low wave drag and low profile drag. Therefore, the entropy production across the shock-wave is low and the boundary layer must be prevented from large scale separation. Potential flow and boundary layer assumptions are both valid and totally adequate under these conditions. Looking back, the full-potential flow method for simple three-dimensional geometry, e.g. Reference [4] using surface fitted grid was a great algorithmic success in the late 1970s and has been successfully implemented as the enabling building block in the first generation wing design code [6]. Most of the algorithm research community, except those in the industry, has since ignored further development in potential flow algorithm and turned their attentions to Euler and Navier-Stokes algorithm in the early 1980s. It is not clear whether this is a result of the failure to understand the requirements, or a result of peer pressure to research on a more complex set of equations. Perhaps, there was no incentive, professional satisfaction or research funding, to continue working on potential flows. In any event, only the insiders of the industry had no hesitation to research potential flow models and to develop the next generation cruise design code, despite its label as a "low order model" by the research community. As we shall see that many advanced design and analysis capabilities can be more readily incorporated into the code because of the choice of the potential flow model. We are certain that similar capabilities using Navier-Stokes formulation can be developed eventually. However, we are not certain whether there is any advantage of a more complex code based on Navier-Stokes equations over a coupled viscous-inviscid, potential flow approximation in this application.

When the requirements are understood, the system design and technology trades are made to construct an effective code. TRANAIR invention of a Cartesian, nested refinement data structure stems from the desire to by-pass the difficult surface conforming grid generation process for a complex geometry, and from the foresight that we need solution adaptive grid to resolve complex flow features if we are to accurately predict cruise drag and other important aerodynamic parameters. Neither automatic grid generation for a general complex geometry, nor solution adaptive capability in three dimensions has been achieved so far with a Navier-Stokes formulation. The exception is a limited class of flaps-up configurations for which the Navier-Stokes grid generation can be and has been automated. The choice of inviscid-viscous coupling model only requires grid adaptation in the inviscid flow field. For that reason, the isotropically refined Cartesian grid strategy is suitable, i.e., the directional adaptation for viscous shear layer is not required. The simplified physical model also drastically reduced the number of dependent variables to be solved. In particular, when a successful two-dimensional integral boundary layer model[7] was extended to its quasi-three-dimensional form, this simplified model reduces the flow variables to a single scalar potential in the inviscid region and a few integral boundary layer properties in the viscous layer. The small number of variables allows complex geometry to be considered even with the computing hardware available in mid-1980s. In contrast, we are still struggling with adequate grid resolution for Navier-Stokes solution even with the large computers available today. Since the flow variables are much reduced from the primary variables in the Navier-Stokes equation, a Newton iteration can be employed. With this implicit algorithm, the inviscid flow variables and the boundary layer variables are solved simultaneously without iterating between them. This strong coupling enhances the robustness of the code. In this code architecture, the components of geometry inputs, grid generation, and solution algorithm are tightly integrated into a system in order to perform the desired functions effectively.

The biggest benefit of Newton iteration method is that it allows us to develop a very flexible optimization code. Newton iteration allows computation of the sensitivity of the flow field in response to the change of surface geometry through back substitutions. Using this sensitivity matrix from the flow solution, together with a defined objective function and with various constraints, an optimization problem can be formulated independently. This separation of optimization problem from flow field sensitivity calculations creates a very flexible, easy to modify optimization code. In the initial stage of implementing optimization-based design process, practicing engineers and code developers jointly explore various possible means of applications. During that process, the code has been modified again and again in response to every new application concept. This code flexibility turns out to be essential to the successful implementation. Without this flexible code structure, it is inconceivable that a design optimization code can be implemented in an engineering environment.

What we have learned from the success is that developing a CFD code is not an after-thought of algorithm research or the by-product of a research solver. It is much similar to developing any complex system. We need to understand what and how the code will be used for. The perceived functionality of the code drives the algorithm technologies which are not only compatible but also operate in concert to maximize the system performance.

The algorithm research by community at large contributes to TRANAIR code development by providing a large gene pool of algorithm technology, e.g. GMRES, upwind schemes, octree data structures etc. Finally, we observe that success of this type of high performance code development activity is unlikely, unless the development team is immersed in an engineering environment the code is targeted for. The areas of code application provide strong focuses for the development team. We believe that the code development team in a manufacturing company must serve, as a minimum, a code architect and a code integrator. Researchers in academia, in government laboratory, and in commercial software companies may participate as the suppliers of algorithm technologies and code modules. It is very doubtful that this type of highly effective and integrated special purpose code can be developed and provided by external suppliers.

CFD-Based Aerodynamic Data Sets ?

Now that CFD codes for cruise design is well developed, where is the next challenge in CFD applications in commercial airplane design process ? Aside from the cruise shape design, is there any role CFD can play in improving the processes of defining the shape of high-lift wing and of providing aerodynamic data base ? These are the applications requiring computation of separated flows. Therefore, solution of Navier-Stokes equations in some form is required. However, despite more than a decade of research, Navier-Stokes codes have not yet been able to serve this function in an airplane development process in a substantial way. They are used as supplementary analysis tools for near cruise conditions. For separated flow analysis, they are mostlyy used in a research environment to study aerodynamics phenomena which are relevant to aerodynamic engineering. Although the research function of the codes is valuable, it is desirable to use Navier-Stokes code for quantitative data acquisition in an airplane development environment. Here, we shall evaluate Navier-Stokes algorithm technology through its potential application to providing aerodynamic data base. We hope to identify key algorithm technology elements for further development.

The first question is whether a Navier-Stokes code, if perfected, can handle the throughput required for providing aerodynamic data base. In order to answer this question, a base line needs to be established.

A complete flaps-up airplane configuration, as shown as a surface grid[8] in Figure 3, requires roughly 150 CPU hours on a departmental computer, while flaps-down configuration may require 650 CPU hours. These calculations use TLNS3D by Vatsa[9,10], a multi-block code enhancing and extending one of the best Navier-Stokes multigrid algorithms originally developed by Martinelli and Jameson[11]. The basic time stepping scheme in these codes is that from the now classic paper of Jameson et al[12].

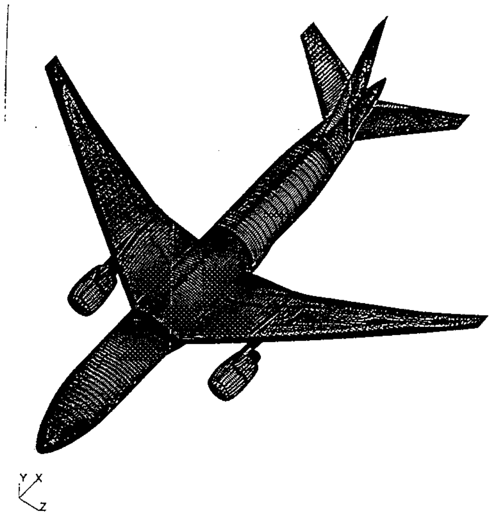

Figure 3 Surface Grid for A Flaps-up Configuration

With these numbers of CPU hours, it is not possibly at all to generate the aerodynamic data base using the computers we have today in a timely manner in airplane development environment. However, we can extrapolate the advances in computing and in algorithm improvement in the next ten years in the most conservative way to see whether this is a possible scenario. We make the following assumptions. The first assumption is that the computing hardware development will continue to provide a minimum of 5 time speed increase per CPU, including memory I/O bandwidth, every five years. In ten years, the CPU speed will increase by twenty-five fold while the cost per CPU is reduced by a minimum of a factor of two. This extrapolation is conservative considering that today's departmental computer has the same single CPU speed of a Cray Y-MP in 1990. The cost per CPU of these departmental computer, equipped with the appropriate memory for large scale CFD computations, is 250 times less expensive than the similarly scaled Cray Y-MP. The second assumption is that through algorithm research, we can gain a factor of 5 in speed of convergence, in terms of CPU hours, over TLNS3D.[8,9] Recent advances in algorithm research[13,14,15,16,17] have already reduced the CPU time for convergence by a factor of three, giving credibility to this estimate. When combining these two aspects of speed improvements, we expect to compute these cases at a single CPU speed 125 times faster than what we can do today, in ten years. The third assumption is that turbulence modeling and transition prediction research in the next 10 years will enable accurate predictions of separated flows for external aerodynamics which is a limited class of flows in fluid dynamics. Some evidence of being able to achieve that will be discussed later.

Now, an order of magnitude estimate of the upper bond of the size of the required data base is roughly 200,000 cases. Perhaps, half of the cases are flaps-up and the other half flaps-down. Therefore, in the year 2008 it will require

640,000 CPU hours to generate the entire database within one month. That computing power can be satisfied by 1,000 CPUs. If our conservative projection of the reduction in cost of the computer is valid, that is about $20 million U.S. dollar investment. This is a price well justified if we consider the comparative cost and flow time it takes to collect the same database in a wind tunnel. Short of achieving this throughput, CFD may be applied to developing wind tunnel data correction methods and to developing methods of varying the database in response to some changes in airplane parameters. In long term, the question of CFD-based data base is not the through-put, but whether the key CFD technologies can be developed in the next ten years to allow this scenario to happen. Therefore, a research manager must manage the risk and must seek a more balanced approach between CFD technology development and testing method improvement, even if we agree with the possibility of this scenario. These key CFD technologies are also required for the shape design of a high-lift wing and for providing critical aerodynamic information for configuration development.

A few examples of recent exploration of analyzing airplane at off-design conditions may provide us with a foundation to determine which are the key CFD technologies requiring further development.

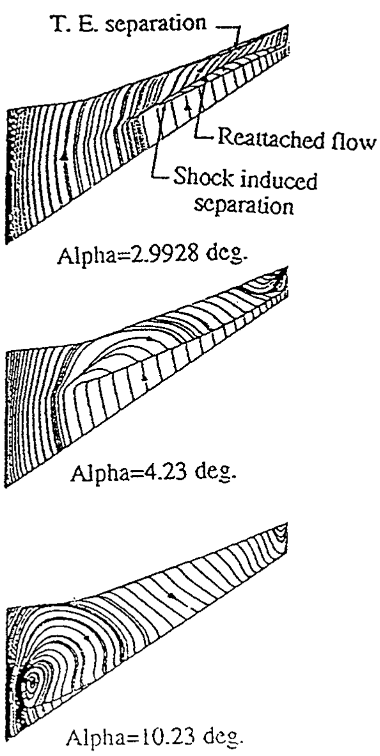

Figure 4 Near Surface Streamlines on a Wing at Varying Angle of Attack

The first example is our attempt to see whether a Reynolds-Averaged-Navier-Stokes (RANS) model can predict transonic pitch characteristics. For this purpose, we chose a simple wing-body for which we have wind tunnel data and performed Navier-Stokes calculations over a large range of angles of attack[18]. These calculations use TLNS3D code with a version of the Spalart-Allmaras turbulence model[19].

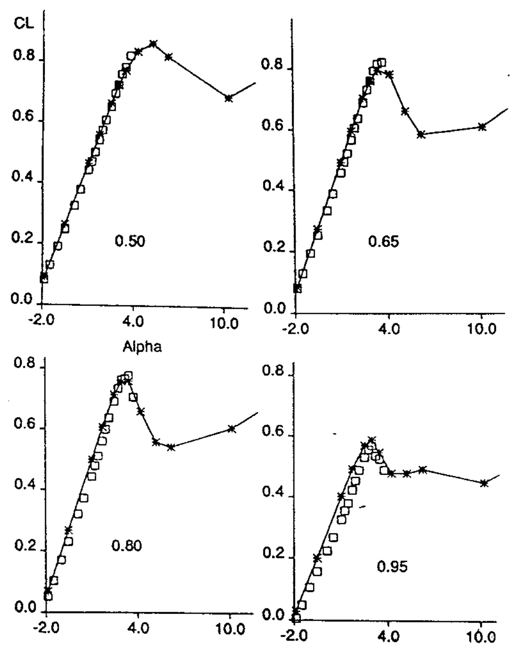

Figure 5 Sectional Lift Curve for a Wing-Body
Square Symbol : Test Data

Figure 4 shows the computed flow direction near the wing surface and Figure 5 shows the comparison of the computed sectional lift characteristics and the test data at the outboard sections of the wing. Notice that the sectional maximum lift coefficients and the corresponding angles of attack were accurately predicted, an indication that shock-induced boundary layer separation can be captured by the code and the model. Several other calculations seem to support this observation.

The other type of boundary layer separation is caused by a gradual evolution of the boundary layer under an extended adverse pressure gradient - a flow much more subtle than shock-induced separation. Although S-A model seems to predicts this type of separation reasonably well, it does predict separation at a lower angle of attack than that indicated by experimental data. We then attempted to predict the leading edge separation. The wind tunnel model is a small aspect ratio wing mounted on a short body as represented by the surface grid in Figure 6. The predicted lift curve and the test data are shown in Figure 7. Near the maximum lift, the RANS model as described did not yield a converged solution. Instead, the computed lift coefficient oscillates in a certain range as indicated by the short bars in Figure 6.

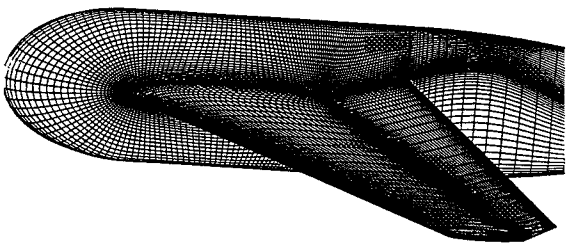

Figure 6 Surface Grid for a Low Aspect Ratio Wing

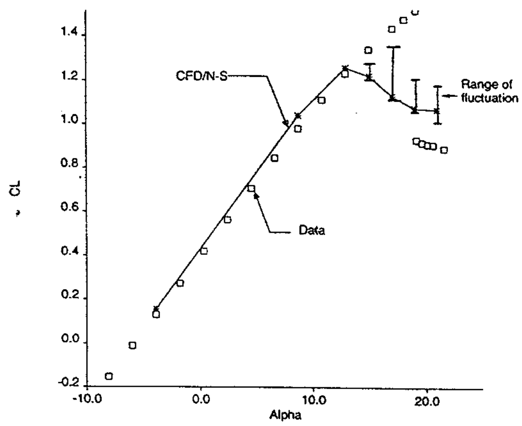

Figure 7 Lift Curve for a Low Aspect Ratio Wing

A careful examination of experimental data indicates that the lift coefficient has a hysteresis loop which covers the range of the oscillation in computed lift coefficient due to convergence difficulty. These computations were performed without a prediction of laminar to turbulent transition which often is the cause of the hysteresis. In addition, the numerical scheme contains variable time steps and multigrid convergence acceleration and is not intended to capture unsteady flows. It is possible that these highly separated flows can not be predicted with a RANS model and may require an unsteady calculation.

A muilti-block version of TLNS3D code is used successfully to explore the spoiler reversal phenomena [20]. The code captures quite accurately the boundary of the reversal phenomenon in the flow parameter range. The pressure distributions on the upper surface are shown in Figure 8. Below the threshold angle of attack for reversal, the deployment of the spoiler moves the shock-wave upstream resulting in loss of lift. On the contrary, deployment of the spoiler at an angle of attack beyond the threshold causes the shock-wave to move downstream resulting in an increase in lift. This capability of the code can be used to assist airplane designer in making configuration decisions.

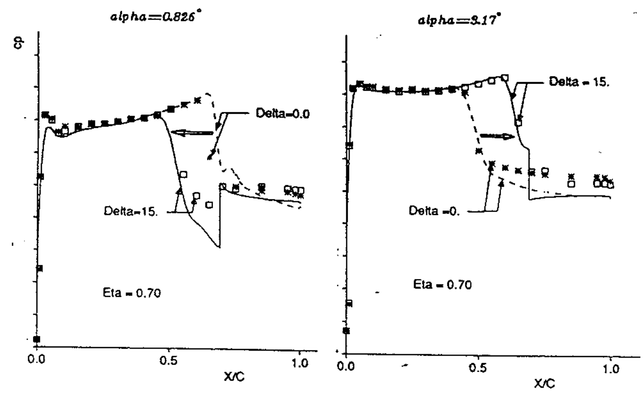

Figure 8 The Effects of Spoiler Deflection on Pressure Distribution on the Wing Upper Surface.

However, the prediction is not accurate enough for the hinge moment prediction. In Figure 8, the pressure behind the deployed spoiler is not accurately predicted. The best effort was made to extend the dense boundary layer grid on the spoiler beyond its trailing edge in an attempt to capture the separation shear layer. But without a solution adaptive capability, this a priori grid focusing may not capture the shear layer faithfully. Although turbulence modeling can be the source of discrepancy, the numerical accuracy is also uncertain. The example illustrates that the success of a code depends on the intended use. In this example, the code is successful in the research environment to understand the cause of the reversal phenomena. But, it fails to quantitatively predict an important element of the loads database.

Because of the difficulty in grid generation for the complex geometry of a high-lift wing, we did not attempt a multi-block Navier-Stokes calculation. Instead, we attempt to use the over-set grid code OVERFLOW[21] for this application[22]. Figure 9 shows a grid configuration for a high-lift wing. Aside from the fact that the viscous grid simply follows the surface geometry, and does not capture the expected off-surface flow features such as free shear layers and vortices, months of flow time is required to generate a grid for such a complex configuration. This exercise cries out for the need for a code with automatic grid generation, perhaps using unstructured grid technology of some form.

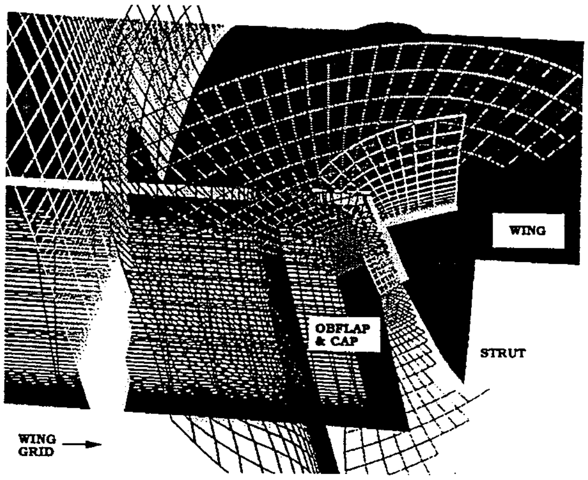

Figure 9 Overset Grid Near the Outboard Flap of a B747

As we can see in these examples, current Navier-Stokes technology provides us with a good research tool. However, they are not ready for extensive application to either high-lift shape definition or to the generation aerodynamic data base. In order to develop an effective engineering tool, further thoughts based on a system concept are required to identify the enabling algorithm technology. An effective code should accurately capture all important flow features in a separated flow, i.e. wall shear layer, free shear layer and vortices. The code should require no manual grid generation for a very complex geometry such as a high-lift wing. It should be able to compute large number of cases in a reasonable time as we have discussed. The following are the algorithm research items that we identified based on these system requirements.

Grid-adaptation as a System Integrating Technology

In a system, there is always one or a few technology elements which have the highest impact on system design and integration. For a general geometry Navier-Stokes code, grid adaptation is the algorithm technology which serves as a glue to integrate the whole system together. This technology probably will be very difficult to implement in the context of a structured grid code and will favor unstructured grid code of some form. Depending on the adaptive strategy, its implementation will affect the data structure of the code as was in the case of TRANAIR code. An effective implementation of the algorithm will reduce the precision required of an initial grid around a complex geometry. Because of the redefinition of the configuration of the surface grid during the course of computing a solution, it will require the code to directly access the surface geometry definition. Therefore, geometry description becomes part of the code. Current solution algorithm requires a "good" grid to maintain accuracy and to ensure a reasonable convergence rate. Realizing the difficulty in generating a "good grid" in the process of grid adaptation without manual intervention, this may require a system trade between the improvement of solution algorithm and the generation of a non-skewed grid. Although there may be a possibility of loosely coupling the solver and grid generation in grid adaptation, the grid generation and solution algorithm may become very integrated as demonstrated in two-dimensional cases by Hecht and Mohammadi [23]. It may even require an adjoint formulation as suggested by Giles [24]. Once an effective grid adaptation algorithm is developed, other parts of a high performance Navier-Stokes will fall in place. There is a large range of possibility and therefore it is a rich field for research.

Convergence Acceleration

Earlier, we discussed the need to improve the rate of convergence of RANS calculations. Direct solution of an implicit scheme have been attempted for Navier-Stokes calculations[30]. As of today, the demand on memory by such method is too far beyond the capacity of the computers. It is still an open question whether it can be an effective algorithm for Navier-Stokes calculations in the future. So far, multigrid method of some form is a more successful algorithm for convergence acceleration. The spectral radius of the best multigrid scheme available today is larger than 0.995. This is much larger than 0.9 or so achievable for Euler equations. Because of the highly non-isotropic nature of a high Reynolds number shear flow, grid aspect ratios of the order of 10^5 is usually required to avoid excessively large number of grid points and the resulting laborious calculations. Recent research in pre-conditioned multigrid methods[13,14,15,16,17] are making good progress in addressing the issue of high aspect ratio grid. However, the spectral radius of the improved scheme is still around 0.94. There seems to be room for improvement. Further research will be extremely valuable. Simply imagine that a factor of three improvement in CPU time will reduce the investment in computing equipment by the same factor world wide. The benefit of reducing engineering process flow time is even larger, as it impacts the down stream processes.

Some codes, e.g. Frink[25], use "wall functions" to by-pass the difficulty of tight grid spacing in the near-wall laminar sub-layer. While there is good evidence that a logarithmic velocity profile does exist even for a separated flow in three-dimensions, the assumption cannot be invoked when the code is required to simulate laminar to turbulent transition.

Turbulence Modeling and DES - a System Trade

Frustration with turbulence modeling leads to many researchers to question whether the concept of RANS is altogether adequate for separated flows. This frustration also leads to the desire to seek for a new direction away from traditional modeling. Promotion of Large Eddy

Simulation (LES) for engineering applications is the result of it. We made an estimate of the computing requirements of an LES for a transport airplane wing and concluded that comparing to a RANS calculation, a LES calculation is 10^6 time more laborious. This estimate is based on the fact that the large eddies in a wall shear layer are three-dimensional in nature with a length scale of the thickness of the wall shear layer. We will be forced to refine the computational grid in an isotropic manner in order to capture the dynamics of these large eddies. As a result, the grid configuration will be very large. In addition, the time scale for establishing a circulation on the wing is very large in comparison to the time scale of the eddy dynamics. Since a time accurate scheme is required for a LES, very large number of time steps will be required for computation of forces and moments. Given that we are struggling with the constraint of computer capability for RANS calculations today, we did not see the possibility in the farthest horizon. We communicated this estimate to the research community[26]. Recently, we formally published our refined estimates[27]. In addition to making the estimate, we suggest a new approach of Detached Eddy Simulation (DES). The method attempt to capture the success of turbulence modeling for a wall shear layer in attached flows and some of the separated flows. LES for the wall shear layer may not be necessary. On the other hand, some of our RANS calculations seems to suggest that they may not be adequate when vortex shedding occurs in a highly separated flow. Here, the "vortex shedding" implies a reduced frequency based on body dimension of the order of one. Steady state calculation, if possible, is always the preferred one from the computational efficiency point of view. However, for a highly separated flow, it may be too much to expect advances in turbulence modeling to provide an accurate RANS steady state calculation. Therefore, a trade between the difficulty of turbulence modeling and a higher computational effort may be required to provide the capability of computing highly separated flows. The idea of DES is to retain turbulence modeling where it is powerful near the wall, and compute large scale vortex shedding by unsteady flow calculations if necessary. The blending of turbulence modeling and the "detached eddy simulation" is smooth without any artificial zonal cutoffs and patching of these two approaches. If this approach can be successful, only unsteady component of the reduced frequency of unity need to be captured. This is a much more feasible idea than LES. However, we want to emphasize that steady state calculations with turbulence modeling is the main approach for most of our applications. Only in the event that these large scale separations are critical in our database or in our knowledge base, will these DES be performed. External aerodynamics is a limited subset of all viscous flows. For example, heat transfer is not a critical issue for subsonic transport, and we do not want to deal with complex scalar transport issues. Therefore, we expect that turbulence modeling and limited DES will cover most of the engineering applications in providing aerodynamic data base and knowledge base.

In order to implement DES, certain requirements for unsteady numerical algorithm are required. Considering unsteady phenomena of reduced frequency one, a CFL number based on the inviscid grid spacing is required to be of the order of one. Since viscous grid spacing normal to the surface is usually of the order of 10^5 smaller than the inviscid grid spacing, the CFL number of a DES scheme must be of the order of 10^5. There are algorithm research efforts[28,29] which lead to this capability.

Conclusions

With the recent success of transonic design and optimization, we re-affirm the high value of CFD to the development process of a commercial airplane. The next challenge to the CFD community is to provide an analysis capability for configuring an airplane, and for generating aerodynamic database needed for external loads and flight control. These applications require analysis of separated flows and therefore a Navier-Stokes code. With current algorithm technology, there are limited flow conditions which can be analyzed in research environment to gain knowledge. Although these analysis are very valuable, it is desirable to develop a code which can be used in real time in an airplane development program. The requirements for such a code is beyond the current level of CFD algorithm technology. The present paper advocates a systems approach through which further algorithm requirements can be identified and communicated to the research community.

Acknowledgments

The author is indebted to Dr. Neng-Jong Yu who provided many figures in this paper, some of them his unpublished work. Dr. Hoa Cao, and Mr. Ed Tinoco also provided some of the figures. Many of the ideas in this paper come from research within the CFD Group in Aerodynamic Research at Boeing Commercial Airplane. In particular, many informal discussions with Drs. Forrester Johnson, Philippe Spalart, Neng-Jong Yu, Venkat Venkatakrishnan, Steven Allmaras, and John Bussoletti over many years shapes the ideas in this paper.

References:

1.Melvin, R. G., Johnson, F. T., Young, D. P., Foutch, D. W., Bussoletti, J. E., and Bieterman, M. B., "Using a Full Potential Solver for Propulsion System Exhaust Simulation", Journal of Propulsion and Power, 9, (1993), p 412-421

2.Huffman, R.P., Melvin, R.G., Young, D.P., Johnson, F.T., Bussoletti, J.E., Bieterman, M.B., Hilmes, C.L., "Practical Design and Optimization in Computational Fluid Dynamics", AIAA Paper 93-3111, July 1993.

3. Murman, E. M. And Cole, J. D., "Calculation of Plane Steady Transonic Flows", AIAA Journal, 1971, Vol. 9, pp.114-121.

4. Jameson, A. And Caughey, D. A., "Finite Volume Method for Transonic Potential Flow Calculations," Proceeding, AIAA 3rd Computational Fluid Dynamics Conference, 1997.

5. McLean, J. D. and Matoi, T. K., "Shock/Boundary-Layer Interaction Model for Three-Dimensional Transonic Flow Calculations", Proceedings, IUTAM Symposium, Turbulent Shear-Layer/Shock-Wave Interactions, J. Delery, ed., September, 1985

6. Goldhammer, M. I. And Steinle, F. W., "Design and Validation of Advanced Transonic Wing Using CFD and Very High Reynolds Number Wind Tunnel Testing", ICAS Proceeding, 1990, pp. 1028-1042.

7. Drela, M. And Giles, M. B., "ISES: A Two-Dimensional Viscous Aerodynamic Design and Analysis Code", AIAA-86-0424, January, 1986

8. Yu, N.J., Su, T.Y., and Wilkinson, W.M., "Multiblock Grid Generation Process for Complex Configuration Analysis Using Navier-Stokes Solvers", AIAA-96-1995, 1996.

9. Vatsa, V. N., and Wedan, B. W., "Development of an Efficient Multi-grid Code for 3-D Navier-Stokes Equations" AIAA-89-1791, 1989.

10. Vatsa, V.N., Sanetrik, M. D., and Parlette, E.B., "Development of a Flexible and Efficient Multigrid-based Multi-block Flow Solver", AIAA 93-0677, 1993.

11. Martinelli, L. and Jameson, "Validation of a Multgrid method for the Reynolds Averaged equations," AIAA-88-0414, 1988.

12. Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes," AIAA-81-1259, June 1981.

13. Pierce, N. and Giles, M., "Preconditioning Compressible Flow calculations on Stretched Meshes", AIAA-96-0889, 1996.

14. Pierce, N., Giles M., Jameson, A, and Martinelli, L., "Accelerating Three-Dimensional Navier-stokes Calculations", AIAA-97-1953, 1997.

15. Allmaras, S., "Algebraic Smoothing Analysis of Multi-grid Method for the 2-D Compressible Navier-Stokes Equations," AIAA-97-1954, 1997.

16. Mavriplis, D. J., "Directional Agglomeration Multigrid Techniques for High Reynolds Number Viscous Flow Solvers," AIAA-98-0612, 1998.

17. Venkatakrishnan, V., "Improved Multigrid Performance of Navier-Stokes Solvers", AIAA-98-2967, 1998.

18. Yu, Neng-Jong, Su, T. Y. And Jou, W. H., "Exploration of Navier-Stokes Applications for Off-design Analysis" Boeing Coordination Sheet, AERO-B-B153-C95-008, (unpublished) 1995.

19. Spalart, P. R., and Allmaras, S. R., "A One-equation Turbulence Model for Aerodynamic Flows", La Recherche Aerospatiale, No.1, 1994, pp 5-21.

20. Wilkinson, W. M., Lines, T. R. And Yu, N. J., " Navier-Stokes Calculations for Massively Separated Flows," AIAA-96-2383, 1996.

21. Buning, P.G., et al, "OVERFLOW Users' Manual, Version 1.6ap" NASA Ames Research Center, Moffett Field, Cal., 1994.

22. Cao, H. V. and Su, T. Y., "Navier-Stokes Analyses of a 747 High-Lift Configuration", AIAA-98-2623, 19988

23. Hecht, F. And Mohammadi, B., "Mesh Adaption by Metric Control for Multi-scale Phenomena and Turbulence", AIAA-97-0859, 1997.

24. Giles, M., "On Adjoint Equations for Aero-Analysis and Optimal Grid Adaptation in CFD", Computing the Future II - A symposium in honor of Earll Murman on the occasion of his 55th birthday, ed D. Caughey, Everett, Washington, June 1997.

25. Frink, N. T., "Assessment of an Unstructured-Grid Method for predicting 3-D Turbulent Viscous Flows", AIAA-96-0292, 1996.

26. Jou, W.-H., Boeing Memorandum AERO-B113B-L92-018, September 1992. To Dr. Joseph Shang.

27. Spalart, P. R., Jou, W-H., Strelets, M., and Allmaras, S.R., "Comments on the Feasibility of LES for Wings, and on a Hybrid RANS/LES Approach", Advances in DNS/LES - Proceeding of the First AFOSR International Conference on DNS/LES, Ruston, Louisiana, USA, August, 1997.

28. Alonzo, J. J. And Jameson, A., "Fully-implicit Time-Marching Aeroelastic Solutions," AIAA-94-0056, 1994.

29. Venkatakrishnan, V. And Mavriplis, "Implicit Method for the Computation of Unsteady Flows on Unstructured Grids," J. Comp. Phys., 123 (1996), pp.380-397.

30. Venkatakrishnan, V., "Newton Solution of Inviscid and Viscous Problems," AIAA-88-0413, 1988.