JOURNAL OF AIRCRAFT
Vol. 42, No. 4, July–August 2005

Transonic Drag Prediction Using an Unstructured Multigrid Solver

Dimitri J. Mavriplis*
NASA Langley Research Center, Hampton, Virginia 23681-2199
and
David W. Levy†
Cessna Aircraft Company, Wichita, Kansas 67218

This paper summarizes the results obtained with the NSU3D unstructured multigrid solver for the First AIAA Drag Prediction Workshop held in Anaheim, California, in June 2001. The test case for the workshop consists of a wing-body configuration at transonic flow conditions. Flow analyses for a complete test matrix of lift coefficient values and Mach numbers at a constant Reynolds number are performed, thus producing a set of drag polars and drag rise curves, which are compared with experimental data. Results were obtained independently by both authors using an identical baseline grid and different refined grids. Most cases were run in parallel on commodity cluster-type machines, whereas the largest cases were run on an SGI Origin machine using 128 processors. The objective of this paper is to study the accuracy of the subject unstructured grid solver for predicting drag in the transonic cruise regime, to assess the efficiency of the method in terms of convergence, CPU time, and memory, and to determine the effects of grid resolution on this predictive ability and its computational efficiency. A good predictive ability is demonstrated over a wide range of conditions, although accuracy was found to degrade for cases at higher Mach numbers and lift values where increasing amounts of flow separation occur. The ability to rapidly compute large numbers of cases at varying flow conditions using an unstructured solver on inexpensive clusters of commodity computers is also demonstrated.

I. Introduction

COMPUTATIONAL fluid dynamics has progressed to the point where Reynolds-averaged Navier–Stokes solvers have become standard simulation tools for predicting aircraft aerodynamics. These solvers are routinely used to predict aircraft force coefficients such as lift, drag, and moments, as well as the changes in these values with design changes. To be useful to an aircraft designer, it is generally acknowledged that the computational method should be capable of predicting drag to within several counts. Although Reynolds-averaged Navier–Stokes solvers have made great strides in accuracy and affordability over the last decade, the stringent accuracy requirements of the drag prediction task have proved difficult to achieve. This difficulty is compounded by the multitude of Navier–Stokes solver formulations available, as well as by the effects on accuracy of turbulence modeling and grid resolution. Therefore, a particular Navier–Stokes solver must undergo extensive validation including the determination of adequate grid-resolution distribution, prior to being trusted as a useful predictive tool. With these issues in mind, the AIAA Applied Aerodynamics Technical Committee organized a Drag Prediction Workshop, held in Anaheim, California, in June 2001,‡ in order to assess the predictive capabilities of a number of state-of-the-art computational-fluid-dynamics methods. The chosen configuration, denoted as DLR-F4 (Ref. 1) and depicted in Fig. 1, consists of a wing-body geometry, which is representative of a modern supercritical swept wing transport aircraft. Participants included Reynolds-averaged Navier–Stokes formulations based on block-structured grids, overset grids, and unstructured grids, thus affording an opportunity to compare these methods on an equal basis in terms of accuracy and efficiency. A standard mesh was supplied for each type of methodology, with participants encouraged to produce results on additionally refined meshes, in order to assess the effects of grid resolution. A Mach-number vs lift-coefficient C_L matrix of test cases was defined, which included mandatory and optional cases. The calculations were initially run by the participants without knowledge of the experimental data, and a compilation of all workshop results including a statistical analysis of these results was performed by the committee.[2]

This paper describes the results obtained for this workshop with the unstructured mesh Navier–Stokes solver NSU3D.[3–5] This solver has been well validated and is currently in use in both a research setting and an industrial production environment. Results were obtained independently by both authors on the baseline workshop grid and on two refined grids generated independently by both authors. All required and optional cases were run using the baseline grid and one refined grid, whereas the most highly refined grid was only run on the mandatory cases. The runs were performed on three different types of parallel machines at two different locations.

II. Flow Solver Description

The NSU3D code solves the Reynolds-averaged Navier–Stokes equations on unstructured meshes of mixed element types, which can include tetrahedra, pyramids, prisms, and hexa-hedra. All elements of the grid are handled by a single unifying edge-based data-structure in the flow solver.[3]

Tetrahedral elements are employed in regions where the grid is nearly isotropic, which generally correspond to regions of inviscid flow, and prismatic cells are employed in regions close to the wall, such as in boundary-layer regions where the grid is highly stretched. Transition between prismatic and tetrahedral cell regions occurs naturally when only triangular prismatic faces are exposed to the tetrahedral region, but requires a small number of pyramidal cells (cells formed by five vertices) in cases where quadrilateral prismatic faces are exposed.

Flow variables are stored at the vertices of the mesh, and the governing equations are discretized using a central difference finite volume technique with added artificial dissipation. The matrix

---

Presented as Paper 2002-0838 at the AIAA 40th Aerospace Sciences Meeting and Exhibit, Reno, NV, 14–17 January 2002; received 11 February 2004; accepted for publication 13 February 2004.

*ICASE, Mail Stop 132C; currently Professor, Department of Mechanical Engineering, University of Wyoming, Laramie, WY 82071.

†currently Senior Engineer Specialist.

‡Data available online at http://www.aioa.org/tc/apa/dragpredworkshop/dpw.html [cited June 2001].

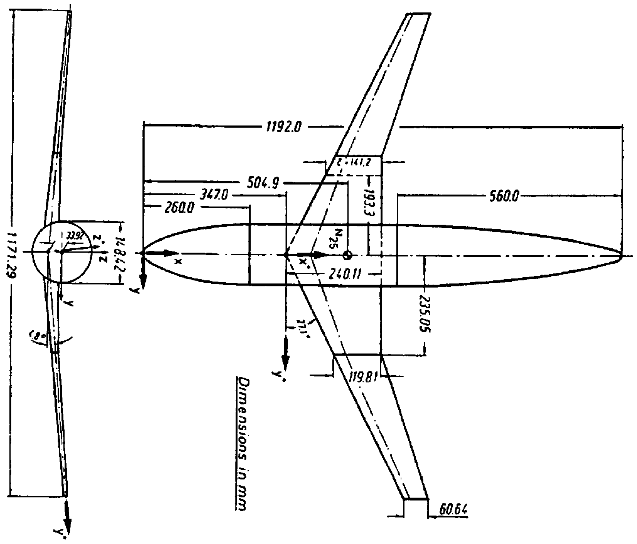

Fig. 1 Definition of geometry for wing-body test case (taken from Ref. 1).

formulation of the artificial dissipation is employed, which corresponds to an upwind scheme based on a Roe–Riemann solver. The thin-layer form of the Navier–Stokes equations is employed in all cases, and the viscous terms are discretized to second-order accuracy by finite difference approximation.[3] For multigrid calculations, a first-order discretization is employed for the convective terms on the coarse grid levels.

The basic time-stepping scheme is a three-stage explicit multistage scheme with stage coefficients optimized for high-frequency damping properties,[6] and a Courant–Friedrichs–Lewy number of 1.8. Convergence is accelerated by a local block-Jacobi preconditioner in regions of isotropic grid cells, which involves inverting a 5×5 matrix for each vertex at each stage.[7–9] In boundary-layer regions, where the grid is highly stretched, a line smoother is employed, which involves inverting a block tridiagonal along lines constructed in the unstructured mesh by grouping together edges normal to the grid stretching direction. The line smoothing technique has been shown to relieve the numerical stiffness associated with high grid anisotropy.[10]

An agglomeration multigrid algorithm[3,11] is used to further enhance convergence to steady state. In this approach, coarse levels are constructed by fusing together neighboring fine-grid control volumes to form a smaller number of larger and more complex control volumes on the coarse grid. This process is performed automatically in a preprocessing stage by a graph-based algorithm. A multigrid cycle consists of performing a time step on the fine grid of the sequence, transferring the flow solution and residuals to the coarser level, performing a time step on the coarser level, and then interpolating the corrections back from the coarse level to update the fine-grid solution. The process is applied recursively to the coarser grids of the sequence.

The single-equation turbulence model of Spalart and Allmaras[12] is utilized to account for turbulence effects. This equation is discretized and solved in a manner completely analogous to the flow equations, with the exception that the convective terms are only discretized to first-order accuracy.

The unstructured multigrid solver is parallelized by partitioning the domain using a standard graph partitioner[13] and communicating between the various grid partitions running on individual processors using either the message-passing-interface (MPI) message-passing library[14] or the OpenMP compiler directives.[§] Because OpenMP generally has been advocated for shared memory architectures, and MPI for distributed memory architectures, this dual strategy not only enables the solver to run efficiently on both types of memory architectures, but can also be used in a hybrid two-level mode, suitable for networked clusters of individual shared memory multiprocessors.[5] For the results presented in the paper, the solver was run on distributed memory PC clusters and an SGI Origin 2000, using the MPI programming model exclusively.

III. Grid Generation

The baseline grid supplied for the workshop was generated using the VGRIDns package.[15] This approach produces fully tetrahedral meshes, although it is capable of generating highly stretched semistructured tetrahedral elements near the wall in the boundary-layer region and employs moderate spanwise stretching in order to reduce the total number of points. A semispan geometry was modeled, with the far-field boundary located 50 chords away from the origin, resulting in a total of 1.65 million grid points, 9.7 million tetrahedra, and 36,000 wing-body surface points. The chordwise grid spacing at the leading edge was prescribed as 0.250 and 0.500 mm at the trailing edge, using a dimensional mean chord of 142.1 mm. The trailing edge is blunt, with a base thickness of 0.5% chord, and the baseline mesh contained five grid points across the trailing edge. The normal spacing at the wall is 0.001 mm, which is designed to produce a grid spacing corresponding to $y^+ = 1$ for a Reynolds number of 3×10^6. A stretching rate of 1.2 was prescribed for the growth of cells in the normal direction near the wall, in order to obtain a minimum of 20 points in the boundary layer.

Because the NSU3D solver is optimized to run on mixed element meshes, the fully tetrahedral baseline mesh is subsequently converted to a mixed element mesh by merging the semistructured tetrahedral layers in the boundary-layer region into prismatic elements. This is done in a preprocessing phase where triplets of tetrahedral layers are identified and merged into a single prismatic element, using information identifying these elements as belonging to the stretched viscous layer region as opposed to the isotropic inviscid tetrahedral region. The merging operation results in a total of two million created prismatic elements, while the number of tetrahedral cells is reduced to 3.6 million, and a total of 10,090 pyramidal elements are created to merge prismatic elements to tetrahedral elements in regions where quadrilateral faces from prismatic elements are adjacent to tetrahedral elements. A higher resolution mesh was generated by the second author using VGRIDns with smaller spacings in the vicinity of the wing root, tip, and trailing edge, resulting in a total of three million grid points, and 73,000 wing-body surface points. One of the features of this refined grid is the use of a total of 17 points across the wing trailing edge vs five for the baseline grid. After the merging operation, this grid contained a total of 3.7 million prisms and 6.6 million tetrahedra.

An additional fine mesh was obtained by the first author through global refinement of the baseline workshop mesh. This strategy operates directly on the mixed prismatic-tetrahedral mesh and consists of subdividing each element into eight smaller self-similar elements, thus producing an 8:1 refinement of the original mesh.[16] The final mesh obtained in this manner contained a total of 13.1 million points with 16 million prismatic elements and 28.8 million tetrahedral elements and nine points across the blunt trailing edge of the wing. This approach can rapidly generate very large meshes that would otherwise be very time consuming to construct using the original mesh-generation software. One drawback of the current approach is that newly generated surface points do not lie exactly on the original surface description of the model geometry, but rather along a linear interpolation between previously existing surface coarse grid points. For a single level of refinement, this drawback is not expected to have a noticeable effect on the results. An interface for reprojecting new surface points onto the original surface geometry is currently under consideration.

The baseline grid was found to be sufficient to resolve all major flow features. The computed surface-pressure coefficient on the baseline grid for a Mach number of 0.75, Reynolds number of 3×10^6, and $C_L = 0.6$ is shown in Fig. 2, illustrating good resolution of the upper surface shock. A small region of separation is also resolved in the wing root area, as shown by the surface streamlines for the same flow conditions, in Fig. 3.

[§] Data available online at http://www.openmp.org.

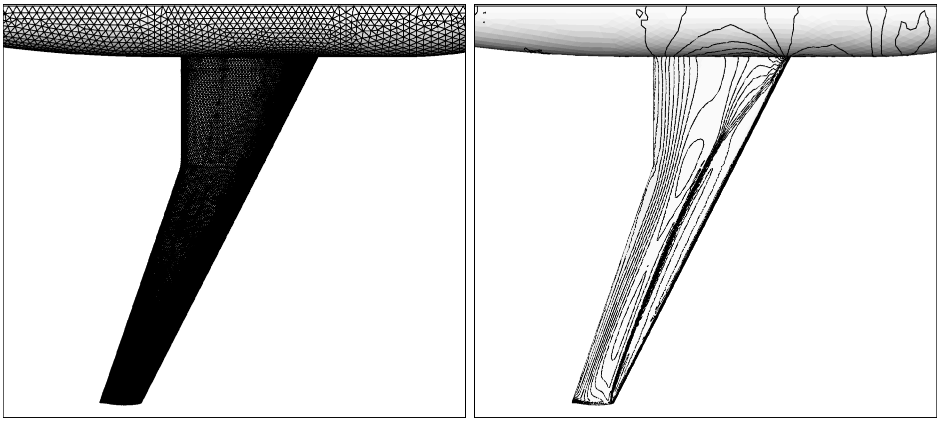

Fig. 2 Baseline grid and computed pressure contours at Mach = 0.75, $C_L = 0.6$, and $Re = 3 \times 10^6$.

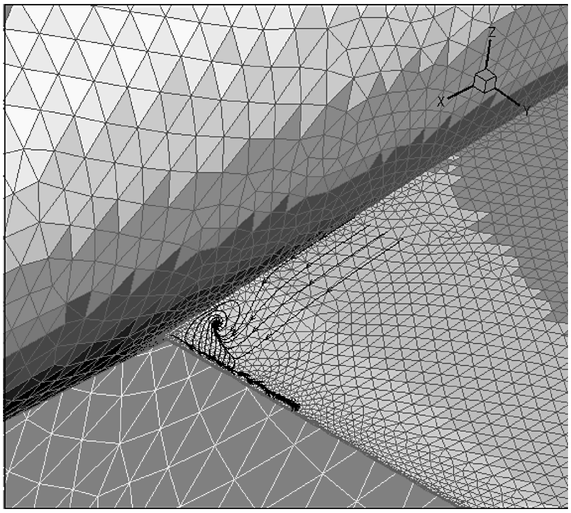

Fig. 3 Computed surface oil flow pattern in wing root area on baseline grid for Mach = 0.75, $C_L = 0.6$, and $Re = 3 \times 10^6$.

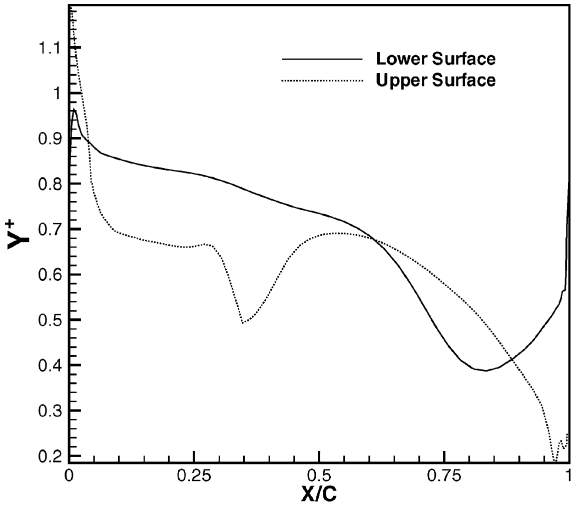

Fig. 4 Computed y^+ on wing surface at span break section on baseline grid for Mach = 0.75, $C_L = 0.6$, and $Re = 3 \times 10^6$.

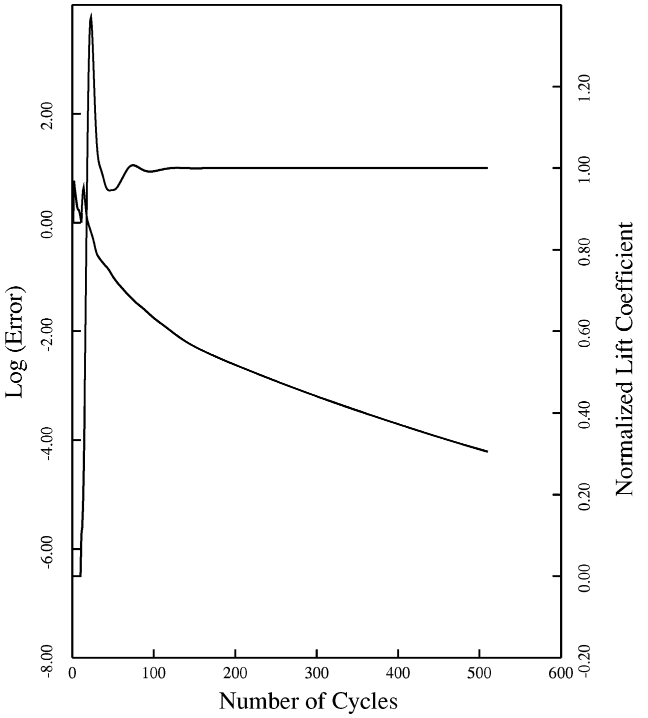

Fig. 5 Density residual and lift coefficient convergence history as a function of multigrid cycles on baseline grid for Mach = 0.75, $C_L = 0.6$, and $Re = 3 \times 10^6$.

Figure 4 depicts the computed y^+ values at the break section for the same flow conditions, indicating values well below unity over the entire lower surface and a majority of the upper surface. The convergence history for this case is shown in Fig. 5. The flow is initialized as a uniform flow at freestream conditions, and 10 single grid cycles (no multigrid) are employed to smooth the initialization prior to the initiation of the multigrid iteration procedure. A total residual reduction of approximately five orders of magnitude is achieved over 500 multigrid cycles. Convergence in the lift coefficient is obtained in as little as 200 multigrid cycles for this case, although all cases are run a minimum of 500 multigrid cycles as a conservative convergence criterion. This convergence behavior is representative of the majority of cases run, with some of the high Mach number and high C_L cases involving larger regions of separation requiring up to 800 to 1000 multigrid cycles. A flow solution on the baseline grid requires 2.8 Gbytes of memory and a total of 2.6 h of wall clock time (for 500 multigrid cycles) on a cluster of commodity components using 16 Pentium IV 1.7-GHz processors communicating through 100 Mbit Ethernet. This case was also run on 4 DEC Alpha processors, requiring 2.4 Gbytes of memory, and 8 h of wall clock time. This case was also benchmarked on 64 processors (400 MHz) of an SGI Origin 2000, requiring 3 Gbytes of memory, and 45 min of wall clock time. The memory requirements are independent of the specific hardware and are only a function of the number of partitions used in the calculations. The cases using the three-million-point grid were run on a cluster of 8 DEC Alpha processors communicating through 100-Mbit Ethernet and required approximately 8 h of wall clock time and 4.2 Gbytes of memory. The 13-million-point grid cases were run on an SGI Origin

Table 1 Grids and corresponding run times

Grid	No. points	No. Tets	No. Prisms	Memory, Gbytes	Run time, h	Hardware
1	1.65×10^6	2×10^6	3.6×10^6	2.8	2.6	16 Pentium IV 1.7 GHz
1	1.65×10^6	2×10^6	3.6×10^6	2.4	8	4 DEC Alpha 21264 (667 MHz)
1	1.65×10^6	2×10^6	3.6×10^6	3.0	45	64 SGI Origin 2000 (400 MHz)
2	3.0×10^6	3.7×10^6	6.6×10^6	4.2	8	8 DEC Alpha 21264 (667 MHz)
3	13×10^6	16×10^6	28.8×10^6	27	4	128 SGI O2000 (400 MHz)

Table 2 Definition of required and optional cases for drag prediction workshop

Case	Description
Case 1 (required) Single point	Mach = 0.75, $C_L = 0.500$
Case 2 (required) Drag polar	Mach = 0.75 $\alpha = -3, -2, -1, 0, 1, 2$ deg
Case 3 (optional) Constant C_L Mach sweep	Mach = 0.50, 0.60, 0.70, 0.75, 0.76, 0.77, 0.78, 0.80 $C_L = 0.500$
Case 4 (optional) Drag rise curves	Mach = 0.50, 0.60, 0.70, 0.75, 0.76, 0.77, 0.78, 0.80 $C_L = 0.400, 0.500, 0.600$

Table 3 Results for case 1; experimental values (Experimental data and 3-M point grid results are interpolated to specified C_l condition along drag polar.)

Case	C_L	α, deg	C_D	C_M
Experiment[a]	0.5000	+0.192	0.02896	−0.1301
Experiment[b]	0.5000	+0.153	0.02889	−0.1260
Experiment[c]	0.5000	+0.179	0.02793	−0.1371
Grid1 (1.6 Mpts)[d]	0.5004	−0.241	0.02921	−0.1549
Grid1 (1.6 Mpts)[e]	0.4995	−0.248	0.02899	−0.1548
Grid2 (3.0 Mpts)	0.5000	−0.417	0.02857	−0.1643
Grid3 (13 Mpts)	0.5003	−0.367	0.02815	−0.1657

[a] ONERA.
[b] NLR.
[c] DRA.
[d] Performed by first author.
[e] Performed by second author.

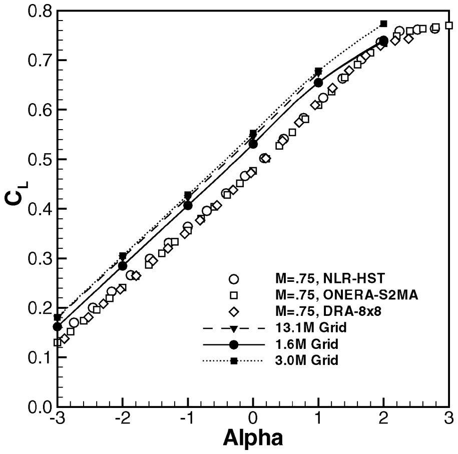

Fig. 6 Comparison of computed lift as a function of incidence for three different grids vs experimental results.

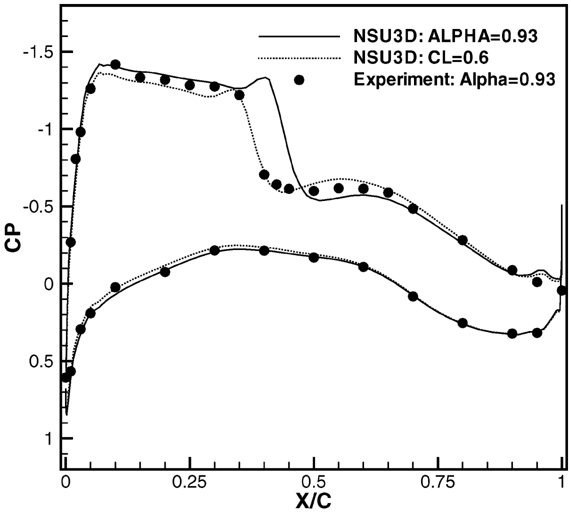

Fig. 7 Comparison of computed surface-pressure coefficients at prescribed lift and prescribed incidence vs experimental values for baseline grid at 40.9% span location.

2000, using 128 processors and required 4 h of wall clock time and 27 Gbytes of memory. A description of the three grids employed and the associated computational requirements on various hardware platforms is given in Table 1.

IV. Results

The workshop test cases comprised two required cases and two optional cases. These cases are described in Table 2. For all cases the Reynolds number is 3×10^6. The first test case is a single point at Mach = 0.75 and $C_L = 0.5$. The second test case involves the computation of the drag polar at Mach = 0.75 using incidences from −3.0 to +2.0 deg in increments of 1 deg. The optional cases 3 and 4 involve a matrix of Mach and C_L values in order to compute drag rise curves. Because an automated approach for computing fixed C_L cases has not been implemented, a complete drag polar for each Mach number was computed for cases 3 and 4. For the baseline grid, the incidence for the prescribed lift value was then interpolated from the drag polar using a cubic spline fit, and the flow was recomputed at this prescribed incidence. The final force coefficients were then interpolated from the values computed in this case to the prescribed lift values, which are very close to the last computed case. For the 3-million-point grid, the force coefficient values at the prescribed lift conditions were interpolated directly from the six integer-degree cases within each drag polar.

All cases were computed using the baseline grid (1.6 million points) and the medium grid (three million points). Only the required cases were computed using the finest grid (13 million points) because of time constraints. Table 3 depicts the results obtained for case 1 with the three different grids. The drag is seen to be computed accurately by all three grids, although there is a 10.6 count variation between the three grids. However, the incidence at which the prescribed $C_L = 0.5$ is achieved is up to 0.6 deg lower than that observed experimentally. This effect is more evident in the C_L vs incidence plot of Fig. 6, where the computed lift values are consistently higher than the experimental values. Because this discrepancy increases with the higher resolution grids, it cannot be attributed to a lack of grid resolution. The slope of the computed lift curve is about 5% higher than the experimentally determined slope and is largely unaffected by grid resolution.

Figure 7 provides a comparison of computed surface-pressure coefficients with experimental values at the experimentally prescribed C_L of 0.6 (where the effects are more dramatic than at $C_L = 0.5$) as well as at the experimentally prescribed incidence of

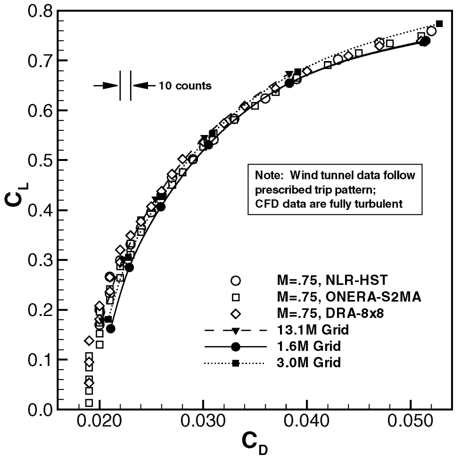

Fig. 8 Comparison of computed vs experimental drag polar for Mach = 0.75 using three different grids.

0.93 deg, at the 40.9% span location. When the experimental incidence value is matched, the computed shock location is aft of the experimental values, and the computed lift is higher than the experimental value, whereas at the prescribed lift condition the shock is further forward, and the suction peak is lower than the experimental values.

This bias in lift vs incidence was observed for a majority of the numerical solutions submitted to the workshop[2] and thus might be attributed to a model geometry effect or a wind-tunnel correction effect, although an exact cause has not been determined. When plotted as a drag polar, C_L vs C_D as shown in Fig. 8, the results compare favorably with experimental data. Although the drag polar was computed independently by both authors using the baseline grid, the results of both sets of computations were identical (as expected), and thus only one set of computations is shown for the baseline grid. The computational results on this grid compare very well with experiment in the midrange (near $C_L = 0.5$), whereas a slight overprediction of drag is observed for low lift values, which decreases as the grid is refined.

This behavior suggests an underprediction of induced drag, possibly because of inadequate grid resolution in the tip region or elsewhere. The absolute drag levels have been found to be sensitive to the degree of grid refinement at the blunt trailing edge of the wing. The drag level is reduced by four counts when going from the 1.6-million-point grid, which has five points on the trailing edge, to the three-million-point grid, which has 17 points on the trailing edge. Internal studies by the second author using structured grids have shown that up to 33 points on the blunt trailing edge are required before the drag does not decrease any further. In the current grid-generation environment and without the aid of adaptive meshing techniques, the generation of highly refined trailing-edge unstructured meshes has been found to be problematic, thus limiting our study in this area.

Figure 9 provides an estimate of the induced drag factor, determined experimentally and computationally on the three meshes.

For C_L^2 up to about 0.36, when the flow is mostly attached induced drag is underpredicted by approximately 10%, as determined by comparing the slopes of the computational and experimental curves (using a linear curve fit) in this region. Grid refinement appears to have little effect on the induced drag in this region. At the higher lift values, the three-million-point grid yields higher C_L and lower C_D values, which is attributed to a slight delay in the amount of predicted flow separation. Results for the 13-million-point grid are not shown at the highest incidence, because a fully converged solution could not be obtained at this condition. The wind-tunnel experiments used a boundary-layer trip at 15% and 25% chord on the upper and lower surfaces, while all calculations were performed in a fully turbulent mode. Examination of the generated eddy viscosity levels in the calculations reveals appreciable levels beginning between 5 to 7% chord. The exact influence of transition location on overall computed force coefficients has not been quantified and requires further study.

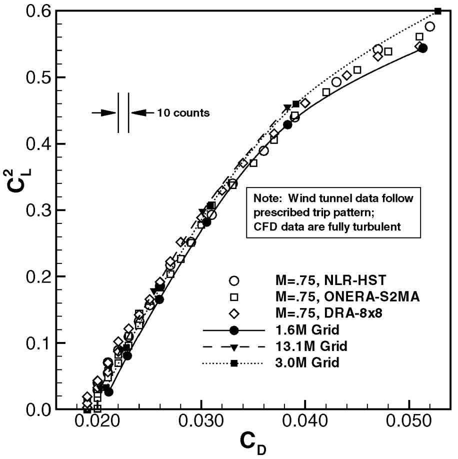

Fig. 9 Comparison of computed vs experimental induced drag factor for Mach = 0.75 using three different grids.

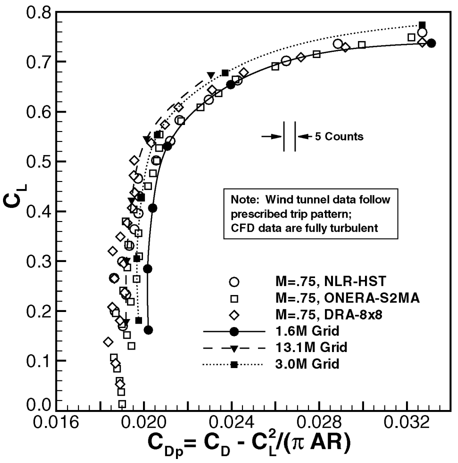

Fig. 10 Comparison of computed vs experimental idealized profile drag at Mach = 0.75 using three different grids.

Figure 10 shows the idealized profile drag,[17] which is defined by the formula:

$$C_{\mathrm{DP}} = C_D - C_L^2\Big/(\pi AR) \tag{1}$$

where AR is the aspect ratio. Plotting C_{DP} generally results in a more compact representation of the data, allowing more expanded scales. It also highlights the characteristics at higher C_L, where the drag polar becomes nonparabolic because of wave drag and separation. In the nonparabolic region, the error in drag is relatively large at a constant C_L.

The pitching moment is plotted as a function of C_L in Fig. 11 for all three grids vs experimental values. The pitching moment is substantially underpredicted with larger discrepancies observed for the refined grids. This is likely a result of the overprediction of lift as a function of incidence, as mentioned earlier and illustrated in Fig. 6. Because the computed shock location and suction peaks do not line up with experimental values, the predicted pitching moments cannot be expected to be in good agreement with experimental values.

Figure 12 depicts the drag rise curves obtained for cases 3 and 4 on the baseline grid and the first refined grid (three million points). Drag values are obtained at four different constant C_L values for a range of Mach numbers. Drag values are predicted reasonably well except at the highest lift and Mach-number conditions. There appears to be no improvement in this area with increased grid resolution, which suggests issues such as transition and turbulence modeling may account for these discrepancies. However, because the two grids

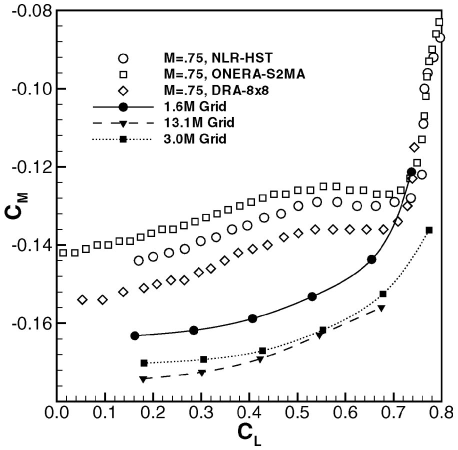

Fig. 11 Comparison of computed vs experimental pitching moment for Mach = 0.75 using three different grids.

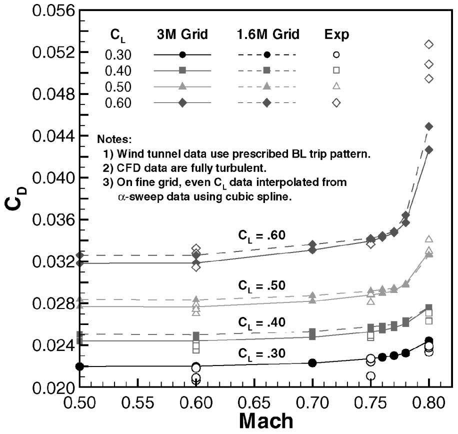

Fig. 12 Comparison of computed vs experimental drag rise curves for three different C_L values on two different grids.

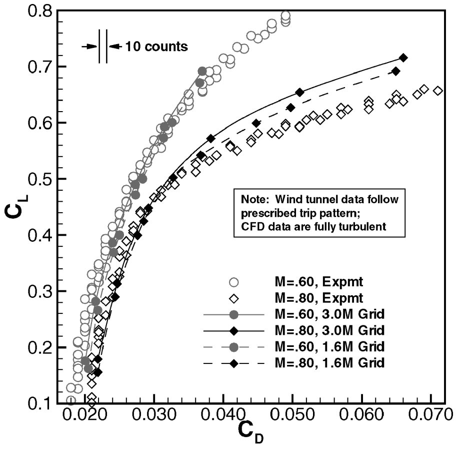

Fig. 13 Comparison of computed vs experimental drag polars for Mach = 0.6 and Mach = 0.8 on two different grids.

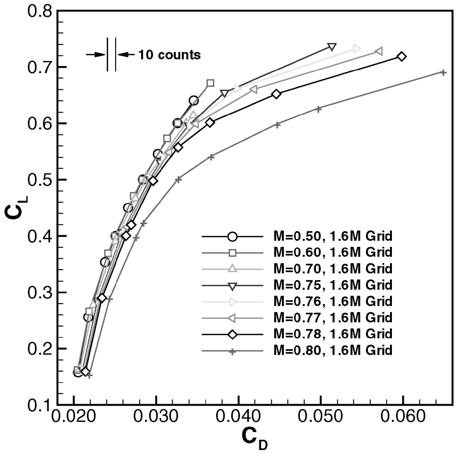

Fig. 14 Depiction of all 72 individual cases run on baseline grid plotted in drag polar format.

have comparable resolution in various areas of the domain grid-resolution issues still cannot be ruled out at this stage.

The results obtained for cases 3 and 4 can also be plotted at constant Mach number, as shown in the drag polar plots of Fig. 13. The plots show similar trends, with the drag being slightly overpredicted at low lift values on the coarser grid and with the refined grid achieving better agreement in these regions. For the higher Mach numbers, the drag is substantially underpredicted at the higher lift values. These discrepancies at the higher Mach numbers and lift conditions point to an underprediction of the extent of the separated regions of flow in the numerical simulations. The comparison of idealized profile drag in Fig. 10 also suggests that the drag caused by flow separation is not predicted accurately at the higher lift conditions. However, the character of the curves also suggest that the error might be caused as well by the C_L offset (shown in Fig. 6). Additional information concerning the regions of flow separation found in the wind tunnel would be needed to more accurately quantify the nature of the error.

The preceding results indicate that the current unstructured mesh Navier–Stokes solver achieves a reasonably good predictive ability for the force coefficients on the baseline grid over the majority of the flow conditions considered. The overall agreement, particularly at the low lift values, is improved with added grid resolution, whereas the more extreme flow conditions that incur larger amounts of separation are more difficult to predict accurately. On the other hand, the observed bias between computation and experiment in the lift vs incidence values has an adverse affect on the prediction of pitching moment. Although the source of this bias is not fully understood, it was observed for a majority of independent numerical simulations undertaken as part of the subject workshop[2] and can likely be attributed to geometrical differences or wind-tunnel corrections.

The results presented in this paper involve a large number of individual steady-state cases. For example, on the baseline grid, a total of 72 individual cases were computed, as shown in Fig. 14, to enable the construction of Figs. 8, 12, and 13. The majority of these cases were run from freestream initial conditions for 500 multigrid cycles, whereas several cases particularly in the high-Mach-number and high-lift regions were run 800 to 1000 cycles to obtain fully converged results. The baseline cases (500 multigrid cycles) required approximately 2.6 h of wall clock time on a cluster of 16 commodity PC processors. This enabled the entire set of 72 cases to be completed within a period of one week. This exercise illustrates the possibility of performing a large number of parameter runs, as is typically required in a design exercise, with a state-of-the-art unstructured solver on relatively inexpensive parallel hardware.

V. Conclusions

A state-of-the-art unstructured multigrid Navier–Stokes solver has demonstrated good drag predictive ability for a wing-body configuration in the transonic regime. Acceptable accuracy has been achieved on relatively coarse meshes of the order of several million grid points, whereas improved accuracy has been demonstrated with increased grid resolution. Grid resolution remains an important issue, and considerable expertise is required in specifying the distribution of grid resolution in order to achieve a good predictive ability without resorting to extremely large mesh sizes. These issues can be resolved to some degree by the use of automatic grid adaptation procedures, which are planned for future work. The predictive ability of the numerical scheme was found to degrade for flow conditions involving larger amounts of flow separation. Slight convergence degradation was observed on two of the grids for the cases involving increased flow separation, whereas a fully converged result could not be obtained on the finest grid (13 million points) for the highest lift case at a Mach number of 0.75. The current results utilized the Spalart–Allmaras turbulence model exclusively, and the effect of other turbulence models in this regime deserves additional consideration. The rapid convergence of the multigrid scheme coupled with the parallel implementation on commodity networked computer clusters has been shown to produce a useful design tool with quick turnaround time.

References

[1] Redeker, G., "DLR-F4 Wing Body Configuration," AGARD Rept. AR-303, Vol. II, Neuilly sur Seine, France, Aug. 1994.

[2] Hemsch, M., "Statistical Analysis of CFD Solutions from the Drag Prediction Workshop," AIAA Paper 2002-0842, Jan. 2002.

[3] Mavriplis, D. J., and Venkatakrishnan, V., "A Unified Multigrid Solver for the Navier–Stokes Equations on Mixed Element Meshes," *International Journal for Computational Fluid Dynamics*, Vol. 8, 1997, pp. 247–263.

[4] Mavriplis, D. J., and Pirzadeh, S., "Large-Scale Parallel Unstructured Mesh Computations for 3D High-Lift Analysis," *Journal of Aircraft*, Vol. 36, No. 6, 1999, pp. 987–998.

[5] Mavriplis, D. J., "Parallel Performance Investigations of an Unstructured Mesh Navier–Stokes Solver," ICASE, Rept. 2000-13, NASA CR 2000-210088, March 2000.

[6] van Leer, B., Lee, W. T., Roe, P. L., Powell, K. G., and Tai, C. H., "Design of Optimally-Smoothing Schemes for the Euler Equations," *Communications in Applied Numerical Mathematics*, Vol. 8, 1992, pp. 761–769.

[7] Riemslagh, K., and Dick, E., "A Multigrid Method for Steady Euler Equations on Unstructured Adaptive Grids," *6th Copper Mountain Conference on Multigrid Methods*, NASA Conf. Publ. 3224, edited by N. D. Melson, T. A. Manteuffel, and S. F. McCormick, 1993, pp. 527–542.

[8] Morano, E., and Dervieux, A., "Looking for O(N) Navier–Stokes Solutions on Nonstructured Meshes," *6th Copper Mountain Conference on Multigrid Methods*, NASA Conf. Publ. 3224, edited by N. D. Melson, T. A. Manteuffel, and S. F. McCormick, 1993, pp. 449–464.

[9] Pierce, N., and Giles, M., "Preconditioning on Stretched Meshes," AIAA Paper 96-0889, Jan. 1996.

[10] Mavriplis, D. J., "Directional Agglomeration Multigrid Techniques for High-Reynolds Number Viscous Flows," *AIAA Journal*, Vol. 37, No. 10, 1999, pp. 1222–1230.

[11] Lallemand, M., Steve, H., and Dervieux, A., "Unstructured Multigridding by Volume Agglomeration: Current Status," *Computers and Fluids*, Vol. 21, No. 3, 1992, pp. 397–433.

[12] Spalart, P. R., and Allmaras, S. R., "A One-Equation Turbulence Model for Aerodynamic Flows," *La Recherche Aérospatiale*, Vol. 1, 1994, pp. 5–21.

[13] Karypis, G., and Kumar, V., "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs," Univ. of Minnesota, TR 95-035, Minneapolis, MN, March 1995.

[14] Gropp, W., Lusk, E., and Skjellum, A., *Using MPI: Portable Parallel Programming with the Message Passing Interface*, MIT Press, Cambridge, MA, 1994.

[15] Pirzadeh, S., "Three-Dimensional Unstructured Viscous Grids by the Advancing-Layers Method," *AIAA Journal*, Vol. 34, No. 1, 1996, pp. 43–49.

[16] Mavriplis, D. J., "Adaptive Meshing Techniques for Viscous Flow Calculations on Mixed-Element Unstructured Meshes," *International Journal for Numerical Methods in Fluids*, Vol. 34, No. 2, 2000, pp. 93–111.

[17] Tinoco, E. N., "An Assessment of CFD Prediction of Drag and Other Longitudinal Characteristics," AIAA Paper 2001-1002, Jan. 2001.