

# Comparison of Overset Grid and Grid Deformation Techniques Applied to 2-Dimensional NACA Airfoils

Charles M. Hoke\*

*Raytheon Missile Systems, Tucson, AZ 85706*

Robert K. Decker<sup>†</sup> and Russell M. Cummings<sup>‡</sup>

*United States Air Force Academy, USAF Academy, CO 80840*

David R. McDaniel<sup>§</sup> and Scott A. Morton<sup>¶</sup>

*Air Force SEEK Eagle Office, Eglin AFB, FL 32542*

This paper documents a comparison of overset grid and grid deformation schemes applied to flapped and non-flapped NACA airfoil configurations in order to determine the relative accuracy and computational efficiency of each method. This study is part of a larger effort to use computational fluid dynamics to perform moving control surface calculations. Three different cases, using both overset and deformed grids, are considered, including: a) lift and moment comparison of a quasi-steady, non-flapped 0012 airfoil, b) lift and moment comparison of a dynamically pitching non-flapped 0012 airfoil, and c) lift comparison of a dynamic pitching and oscillating flapped 0012 airfoil. These results are compared to experimental data from various sources. Two flow solvers of common lineage were used for the computations: Cobalt for overset and rigid mesh motion and AVUS for the deformable mesh motion. All of the methods produced nominally similar results. As expected, the rigid mesh technique required the least amount of computational resources, while the deformable mesh technique required the greatest amount of computational resources due to its serial implementation. However, in the end, it is difficult to recommend one method over another as the application of each method may be dependent on the project being solved.

## Nomenclature

$c$	Airfoil chord length
$C_l$	Lift coefficient, $C_l = l / (q c)$
$C_m$	Pitching moment coefficient, $C_m = M_z / (q c^2)$
$C_y$	Normal force coefficient, $C_y = f_y / (q c)$
$f_y$	Normal force
$l$	Lift force
$M_z$	Pitching moment
$q$	Dynamic pressure, $q = \frac{1}{2} \rho_\infty U_\infty^2$
SA	Spalart-Allmaras turbulence model
$U$	Velocity in x direction
$y$	Spacing normal to surface

\*Research Engineer, Guidance, Navigation, and Control Division, AIAA Senior Member

<sup>†</sup>Research Engineer, Department of Aeronautics, AIAA Member

<sup>‡</sup>Professor of Aeronautics, Department of Aeronautics, AIAA Associate Fellow

<sup>§</sup>Research Engineer, DoD HPCMP/CREATE Kestrel Team, University of Alabama-Birmingham, AIAA Member

<sup>¶</sup>Senior Research Engineer, DoD HPCMP/CREATE Kestrel Team, AIAA Associate Fellow

### *Subscripts*

$\infty$  Free-stream conditions

### *Symbols*

$\alpha$  Angle of attack, degrees

$\Delta t$  Global time step, seconds

$\rho$  Density

### *Superscripts*

$+$  Wall units

## I. Introduction

FOR many years, high-fidelity CFD solutions (i.e. solutions of the Navier-Stokes equations with appropriate turbulence modeling) for full aircraft configurations have been commonplace. The speed and efficiency at which high performance computing (HPC) machines are currently able to operate is astounding compared to the capabilities of ten years ago, and this trend shows no signs of slowing. As a result, the use of high-fidelity CFD to predict the stability and control characteristics of new aircraft designs is much more plausible instead of only being used to help understand complex flow behaviors and the associated instabilities discovered in wind tunnel or flight tests. While the routine use of CFD to fully populate an aerodynamic database for a new aircraft configuration is still not computationally tractable today, the simulation of particular flight conditions or configurations where complex flow phenomena are expected to exist can translate into a significant savings in wind tunnel and flight test costs. This is especially true if costly problems such as abrupt wing stall and tail buffet are discovered early in the design phase.

In future aircraft designs, the flow field surrounding the vehicle will only become more complex as thrust vectoring, active aeroelastic structures, and other related technologies are implemented for stability and control augmentation. Unmanned combat vehicles will operate in flight regimes where highly unsteady, nonlinear, and separated flow characteristics dominate since there are no man-rating requirement.<sup>1</sup>

It would be beneficial to have the ability to accurately analyze and evaluate the non-linear stability and control characteristics of an aircraft and possible associated armament earlier in the design phase than currently possible. Traditionally, a combination of linear aerodynamic theory and empirical data, scale model wind tunnel tests, and full-scale flight tests are used to assess the stability and control characteristics of a new aircraft design. The current linear modeling methods have been very successful but the accuracy is beginning to suffer as high performance designs include larger regions of vortical or separated flow. Wind tunnel techniques are expensive and time consuming but still the method of choice today for new flight vehicle designs. Still, complex models and facilities are necessary to represent the dynamic behavior associated with maneuvering vehicles and the associated control surface deflections. Additionally, there are various blockage, scaling, Reynolds number, and support interference effects to contend with. Finally, flight testing is extremely accurate and telling, but there are obvious downsides relating to cost and safety with this technique.

When applied to stability and control problems, high-fidelity CFD is definitely in a position to overcome many of the downsides of the techniques mentioned previously. Many of the opportunities and challenges associated with the application of CFD to aircraft stability and control were discussed heavily at a NASA-sponsored symposium on Computational Methods for Stability and Control (COMSAC).<sup>2,3</sup> Although it appears that an organized approach to computational stability and control is lacking, the application of computational techniques to aircraft stability seems to be of interest to a number of organizations based on current literature and various other current research endeavors with which the authors are familiar. However, the representation of moving control surfaces in high-fidelity CFD solutions is lacking, especially when dealing with unstructured meshes. This is likely due to the overhead involved with the generation of grids for various control surface deflections and the fact that adding conventional control surfaces to existing structured or unstructured grids is not a trivial undertaking, especially considering the refinements necessary to properly model the separated flow regions spawned by large surface deflections.<sup>4</sup> By no means are the authors implying that the stability problem is insignificant or trivial. Indeed, this is the topic of another paper.<sup>5</sup>

## II. Previous Research

A number of researchers have integrated flight controls into CFD simulations, but a thorough treatment of the problem with appropriate validation still seems to be lacking. Most of the research has been focused on aeroservoelastic problems associated with transonic conditions where only small surface deflections are applicable. NASA's COMSAC program<sup>2,3</sup> initially extended the well-known transonic small disturbance code CAP-TSD to represent control surfaces but noted that restrictions in the code prevented proper implementation. Also, this technique would not be applicable to the wide range of flight conditions where high-fidelity CFD is required. The majority of techniques applied to Navier-Stokes or Euler solvers are applicable only to structured grids. A transpiration method was used to model the Benchmark Active Controls Technology (BACT) wing with its moving controls.<sup>6</sup> Here, the velocity components at the boundary were altered to present the control surface with the desired deflection to the flow field. One of the more common approaches is to generate multiple grids representing key surface deflections and then use an interpolation technique to compute intermediate deflections. Trans-finite interpolation is then used to deform the volume grid due to the moving surface.<sup>7,8,9</sup> Another popular technique is to represent the moving surface as a separate structured block and then use an overset capability to implement the surface and volume movement.<sup>10,11</sup> When the flow solver allows topology changes during a solution or if time-dependent surface deflections are not a requirement, a remeshing technique may also be included with an overset procedure to model the moving surface.<sup>12,13</sup> Another interesting technique in the literature includes the use of a meshless solver.<sup>14</sup> The ability to model large surface deflections is promising, but the proper representation of the moving boundaries appears to be a complex issue with meshless techniques.

In contrast to the structured grid approaches, very few control surface implementations for unstructured meshes have been accomplished. This is unfortunate considering the advantages of unstructured domains such as better representation of complex geometries, shorter and more automated generation time frames, and suitability to adaptive mesh refinement techniques.<sup>15</sup> The Arnold Engineering and Development Centers (AEDC) FD-CADRE software (Fluid Dynamics-Computational Analysis of Dynamically Responsive Environment)<sup>10</sup> applies an overset technique to unstructured grids to model store separation. The issue of modeling components that are attached and moving relative to a parent body is discussed as being complex and the focus of future work. The SIKMA (Simulation of Complex Maneuvering Aircraft) project from DLR implements an overset technique for modeling moving controls on an aeroelastic vehicle using the TAU flow solver.<sup>16</sup> A hierarchical motion-node structure is implemented to keep the overset grid assembly for the moving control connected properly as the elastic wing deforms. Inviscid results for surfaces with no gap representation are presented as part of the most recent SIKMA project update. Murayama et al model an all-moving tailplane in the simulation of a high-speed civil transport aircraft.<sup>17</sup> In their technique, the tailplane mesh is continuous with the fuselage mesh, and a mapping method is implemented to slide the surface mesh as the tailplane rotates. A spring analogy is used to deform the volume mesh.

One of the complex issues associated with the proper modeling of moving control surfaces in CFD meshes is the representation of the gaps between the moving surface and the fixed structure. Very few of the projects mentioned previously have accurately modeled these gaps.<sup>7,8,9</sup> Many of the projects have webbed the control surfaces to the fixed structure to avoid the massive shearing associated with the large deflections of encased controls such as conventional ailerons and rudders. As such, many of the projects mentioned previously only report results for small deflection magnitudes (10 deg). However, it should be noted that the results presented generally compare well with experiments when such comparisons are made.

Overset grid technologies are one way to overcome many of these issues, and Meakin et al<sup>18</sup> describes these techniques for structured grids. Unstructured grids, however, require greater computational capacity and memory requirements due to the necessary grid-point searches and interpolations. Noack et al<sup>19</sup> have developed the DiRTlib library and SUGGAR code to implement these methods for unstructured domains, and the methods utilized by Cobalt in this paper use a similar, proprietary scheme. The hole-cutting procedure is based on an octree method, described again by Noack.<sup>20</sup>

An alternative to overset techniques for representing moving components is the deformation of the mesh. A number of mesh deformation schemes have been successfully implemented to support CFD studies of aeroelastic vehicles. Samareh includes a good review of many of these techniques in a fairly recent paper.<sup>21</sup> Specific to structured grids, trans-finite interpolation is a common way to relocate (regenerate) mesh nodes.<sup>22,23</sup> Applicable to both structured and unstructured meshes, a series of linear and torsional springs may be used to model the stiffness of edges and angles, respectively, in the grid.<sup>24,25,26</sup> After the surface deformation is implemented, the volume nodes are relocated based on the solution to the coupled system.

The volume mesh may also be modeled as an elastic solid where updated node locations are determined by solving the elasticity equations.<sup>27</sup> Kholodar et al<sup>28</sup> implement a hybrid technique for unstructured viscous meshes where the prismatic boundary layer region of the mesh is relocated according to the changing normal basis vectors of the moving surface and the tetrahedral volume mesh is then updated according to the spring analogies mentioned previously. A few efficient algebraic techniques are also discussed in recent literature. Samareh<sup>21</sup> and Liu et al<sup>29</sup> both discuss promising algebraic techniques. Samareh implemented an unstructured mesh deformation technique based on quaternion algebra, and Liu et al implemented a point location technique based on a Delaunay tetrahedralization of the outer boundary and deformable surface.

### III. Case Setup

#### A. Flow Solver

The team did not have access to a flow solver that would allow both overset and deformable mesh capability for 2-D flow. However, the team was able to find a suitable compromise. The commercial code Cobalt and the Air Force Research Laboratory’s Air Vehicle Unstructured Solver (AVUS) both share the Cobalt<sub>60</sub> codebase.<sup>30</sup> Cobalt recently added an overset capability<sup>31</sup> and an updated version of AVUS that includes dynamic mesh motion terms was recently integrated into the CREATE-AV/*Kestrel* project,<sup>32</sup> enabling a coupled deformable mesh capability.<sup>33</sup>

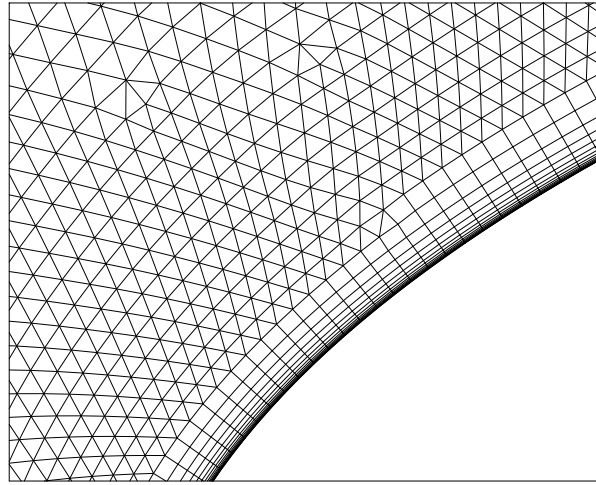
Cobalt<sub>60</sub> is a hybrid unstructured Navier-Stokes flow solver and is considered highly robust and accurate with excellent parallel performance. The code has been validated on a number of problems.<sup>30</sup> Tomaro et al converted Cobalt<sub>60</sub> from explicit to implicit, enabling CFL numbers as high as one million.<sup>34</sup> Grismer et al,<sup>35</sup> then parallelized the code, yielding a linear speedup on as many as 2800 processors. Forsythe et al,<sup>36</sup> provided a comprehensive testing and validation of the various RANS models, including the Spalart-Allmaras, Wilcox  $k-\omega$ , and Menter’s SST turbulence models. It is the well validated selection of turbulence models that makes Cobalt<sub>60</sub> a useful codebase for full aircraft computations.

As mentioned above, AVUS shares a common code base with Cobalt. Thus, it also employs a cell-centered, finite-volume scheme that is first order accurate in space. However, documentation from Cobalt Solutions, LLC suggests that changes were made to the Riemann solver and the inviscid flux scheme to improve performance and lower dissipation in the commercial Cobalt code. The AVUS solver still uses the original exact Riemann solution method of Gottlieb and Groth.<sup>37</sup> Linear gradients within each cell are determined via a least-square reconstruction technique resulting in second order spatial accuracy, although it is expected that the stencils used in the reconstruction differ between Cobalt and AVUS. Cobalt and AVUS both achieve first and second order temporal accuracy via the unconditionally stable point-implicit scheme implemented by Tomaro et al.<sup>34</sup> Recent work by Koomullil<sup>38</sup> incorporated moving mesh capabilities into AVUS allowing for mesh coordinate updates to be made at each time step via a file-in, file-out (FIFO) technique. AVUS was heavily re-factored as part of the ongoing CREATE-AV/*Kestrel* program such that the flow solver is only tasked with integrating the governing equations for the fluid domain one time step at a time. All input and output operations to include mesh coordinate updates for rigid or deforming mesh motion are accomplished externally and communicated directly in memory as part of the *Kestrel* execution paradigm.<sup>32</sup> Other small updates were made as part of the *Kestrel* program such as the addition of a solution update limiter, but none of these changes are expected to result in substantially different results compared to the original AVUS code.

#### B. Grid and Boundary Conditions

All grids used in this study were generated using Mississippi State University’s SimCenter tool.<sup>39,40</sup> The grids included quadrilateral elements in the boundary layer and triangular elements away from the airfoil surface (see Fig. 1) with an initial grid spacing such that  $y^+$  was less than one based on the chord Reynolds number. Standard external flow boundary conditions were utilized for all cases, with adiabatic no-slip applied at the airfoil surface and Riemann invariant conditions at the farfield boundaries.

The rigid motion and deformable mesh cases used a single grid. Each overset case used two grids: the quasi-steady and pitching case used a fixed background grid and a separate airfoil grid, while the pitching airfoil with the oscillating flap used a fixed background grid that included the airfoil and a separate flap grid. Attempts were made to keep the number of surface grid points equal between the rigid motion, deformable mesh, and overset grids. The overset grids were also created with a one-to-one cell spacing in the region



**Figure 1.** Enlarged view of grid near the leading edge of the airfoil. Note the use of quadrilateral elements in the boundary layer and triangular elements away from the surface. All grids were created with an initial grid spacing such that  $y^+$  was less than one based on the chord Reynolds number.

where the two grids overlapped. As the overset grids are displaced, the total number of cells in the grid will change but is expected to remain close to the initial value.

A grid convergence study was conducted for cases B, C, and D in section IV. Three grids were generated for each mesh movement type (rigid, overset, and deformation) and case. The various grids were then run through the prescribed motions and the results analyzed. There were no noticeable differences in the force and moment results due to grid cell density (a listing of the grid cell counts for case B in section IV can be found in Table 1). For the sake of consistency, all data in this report was generated on a medium grid.

**Table 1.** A listing of cell count for the grids generated for the quasi-steady case (case B) in section IV. All grids were generated with a similar surface grid spacing. However, there was no way to control how Cobalt assembled the overset grids (which helps to explain the increased cell count for the overset grids).

	Coarse	Medium	Fine
Rigid	51,715	97,245	198,540
Overset	86,157	212,937	410,987
Deformation	51,715	97,245	198,540

### C. Time Step

Cases B, C, and D in section IV were run with second order temporal accuracy and a specified global time step. Additionally, a time step sensitivity study was conducted for case C in section IV. The time step sensitivity study showed little change in the normal force and pitching moment coefficients due to a decreased time step ( $\Delta t$ ). The time steps for the results in section IV are as follows:  $2.7 \times 10^{-5}$  seconds for case B,  $1.03504 \times 10^{-5}$  seconds for case C, and  $6.0 \times 10^{-5}$  seconds for case D.

### D. Turbulence Model

Two turbulence models were investigated: Spalart-Allmaras (SA) and Wilcox's 1998  $k-\omega$ . Both turbulence models were able to capture the lift and moment curve characteristics for a static subsonic NACA 0012 airfoil at varying angles of attack. However, because this paper will be used as the building block for future

complicated 3-D flowfields, the team decided to focus on the SA model due to its applicability to 3-D external aerodynamic flow. All calculations presented in this paper used the SA model.

## IV. Results

Four distinct cases were run. The first case was a static angle of attack sweep used to verify that Cobalt and AVUS produced similar results. The second case was a low-frequency (1.8 Hz) sinusoidal pitching case that generated quasi-steady results. The third case was a high frequency ( $\approx 50$  Hz) sinusoidal pitching case. The final case was a sinusoidal pitching airfoil with an independently oscillating plain flap. Additional details about these cases are given below. While experimental data is plotted where available, the purpose of this study was not to match experimental results. Instead, the focus was on determining the relative accuracy and computational performance between the various types of mesh movement strategies. Any discrepancy between experimental and computational results may be noted with some suggestions on possible cause, but investigation of discrepancies will be left to a follow-on study.

### A. Code-to-Code Validation

A relatively simple code-to-code validation study was accomplished to “prove” that Cobalt and AVUS produced nominally similar results for a pitching NACA 0012 airfoil using rigid mesh motion and the SA turbulence model (this case is identical to the case in subsection B below). The code-to-code validation results are shown in Fig. 2. The agreement between the codes in the linear region is excellent, while there is a slight difference between the codes at angles of attack near stall. Both codes seem to exhibit flow unsteadiness as noted by the oscillations in the force and moment results near stall. Based on the agreement of the codes for rigid mesh motion, the team felt comfortable assuming that Cobalt and AVUS are comparable for purposes of this study.

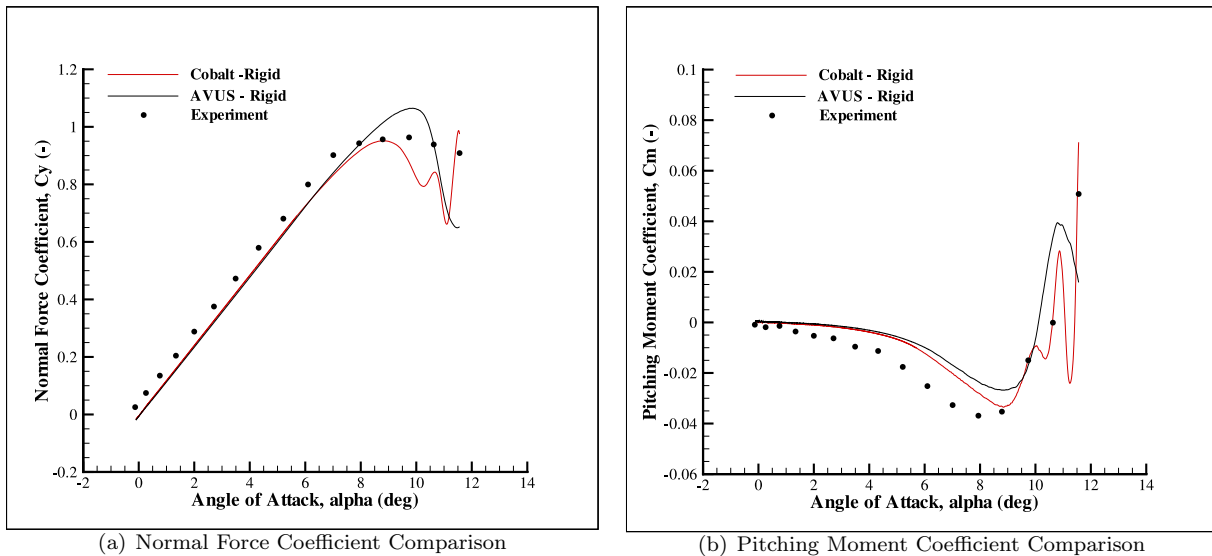
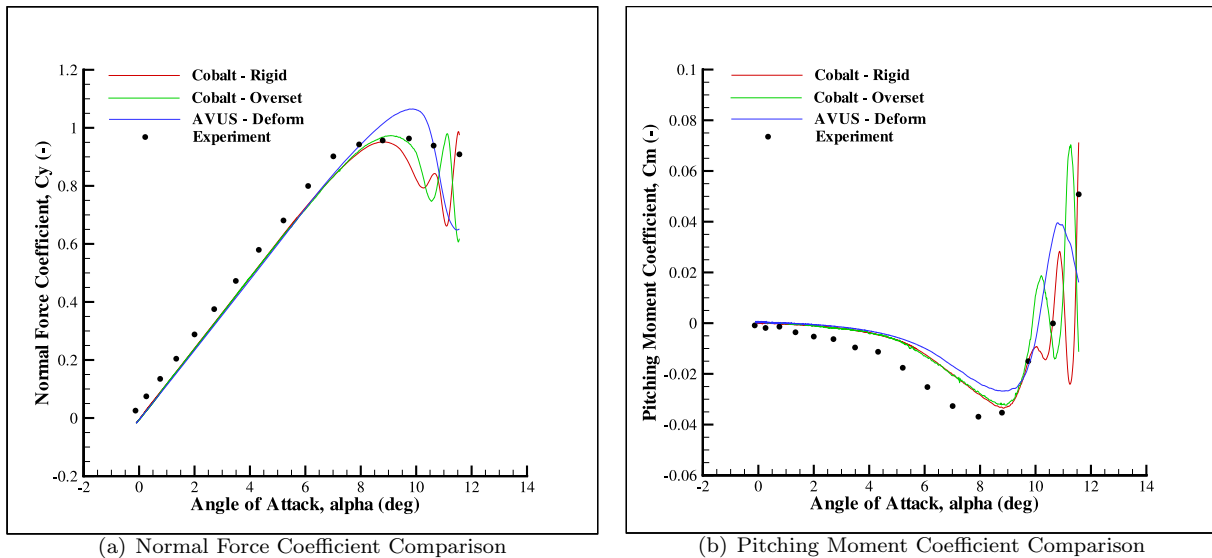


Figure 2. Results of code-to-code validation study for a NACA 0012 airfoil using the SA turbulence model. The Reynolds number was  $4.6 \times 10^6$  and the Mach number was 0.58. The pitching moment was calculated about  $0.25c$  and a positive pitching moment indicates airfoil nose down. Good agreement is seen between the solvers before stall. While there is a slight difference in the solvers near stall, the difference is small enough to assume that the two codes are comparable.

### B. Quasi-Steady NACA 0012

The first mesh movement case involved sinusoidally pitching a NACA 0012 airfoil at a relatively low frequency of 1.8 Hz (to correspond with run number 11 in Ref. 41). The case Reynolds number of  $4.6 \times 10^6$ , Mach number of 0.58, and an angle of attack of the oscillation of  $8.99^\circ \pm 9.55^\circ$  was such that the airfoil was allowed to

stall before  $\alpha$  decreased and attached flow was re-established. The solution was computed using the SA turbulence model. The results for this case are shown in Fig. 3. It should be noted that the results in Fig. 3 are limited to an  $\alpha$  range of  $5.845^\circ \pm 5.715^\circ$  because the tabulated experimental results only covered that range of  $\alpha$ . The computations included two complete pitch oscillations. However, the data in Fig. 3 is from the last complete pitch oscillation.



**Figure 3. Results of quasi-steady simulation on a NACA 0012 airfoil using the SA. Here the Reynolds number was  $4.6 \times 10^6$ , the Mach number was 0.58, and the pitching moment was calculated about  $0.25c$ . A positive pitching moment indicates airfoil nose down. There is reasonable agreement between all methods before stall. Near stall, the deformable mesh results seem to slightly overshoot the maximum normal force and slightly undershoot the maximum pitching moment.**

The normal force coefficient (Fig. 3(a)) matches very well between the rigid mesh and the overset mesh. There is also good agreement with the deformable mesh results before stall. As with the rigid mesh data in section A, the deformable mesh results tend to slightly overshoot the stall angle of attack and maximum normal force coefficient. Given the agreement between the rigid mesh and deformable mesh results in AVUS, there is no reason to believe that the deformable mesh capability is leading to the slight overshoot. All three methods exhibit some signs of flow unsteadiness as is noted by the oscillations in the force and moment results near stall.

The pitching moment coefficient results (Fig. 3(b)) behave similarly to the normal force coefficient results. The rigid mesh and overset mesh results match each other throughout the angle of attack sweep. The deformable mesh results match reasonably well leading up to stall and undershoots the rigid mesh and overset mesh results near stall.

### C. Pitching NACA 0012

The second mesh movement case involved sinusoidally pitching a NACA 0012 airfoil at a higher frequency of 50.32 Hz (to correspond with AGARD CT case 1 in Ref. 41). The flow conditions were very similar to the quasi-steady case; the Reynolds number was  $4.8 \times 10^6$ , the Mach number was 0.6, and the simulations were completed with the SA turbulence model. The angle of attack sweep was  $2.89^\circ \pm 2.41^\circ$ . The computations included four complete pitch oscillations. However, the data in Fig. 4 represents the last complete pitch oscillation.

The normal force coefficient results (Fig. 4(a)) show that all three methods produce nominally identical answers. The general slope of the computed normal force curve is lower than that of the experiment. This is likely due to a slight under prediction of the viscous effects of the flow around the airfoil. The pitching moment coefficient results (Fig. 4(b)) show good agreement between the rigid mesh and overset mesh results. There is a noticeable difference in the deformable mesh results. Given the agreement of the normal force coefficient, it is unlikely that the discrepancy is due to something physical in the flowfield.

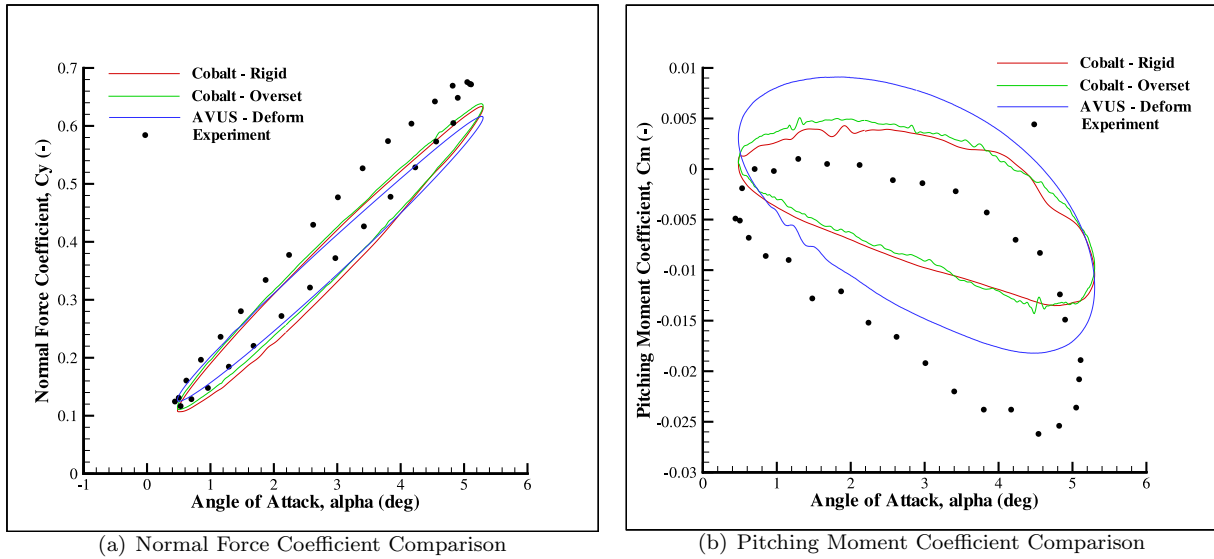


Figure 4. Results of purely pitching simulation on a NACA 0012 airfoil using the SA turbulence model. The simulation Reynolds number was  $4.8 \times 10^6$ , the Mach number was 0.6, and the pitching moment was calculated about 0.25c. The three mesh motion methods provide a similar result for normal force coefficient. There is a noticeable difference in the pitching moment coefficient. The deformable mesh difference will require additional investigation.

The pitching case also provided an opportunity to examine the state of the grids for the three mesh movement methods at two different points in time. The first point in time (Fig. 5(a) through Fig. 5(c)) shows the state of the grid at the beginning of the simulation where  $\alpha$  is  $2.89^\circ$ . The second point in time (Fig. 5(d) through Fig. 5(f)) shows the grid near the case maximum  $\alpha$  deflection of  $5.30^\circ$ .

#### D. Pitching NACA 0012 with an Independently Oscillating Flap

A more complicated dynamic simulation was run involving an airfoil and a flap that pitch at different frequencies and amplitudes. The experimental data for this case was generated at a chord based Reynolds number of  $1.63 \times 10^6$  and a Mach number of 0.4.<sup>42</sup> The airfoil was sinusoidally pitched with a frequency of 5 Hz and an amplitude of  $4.0^\circ \pm 6.0^\circ$ . The flap was sinusoidally deflected with a frequency of 10 Hz, with an amplitude of  $0.0^\circ \pm 5.6^\circ$ , and a phase shift of  $-148^\circ$ . The simulations were completed using the SA turbulence model.

The lift coefficient results are shown in Fig. 6. The overset mesh and deformable mesh results show similar behavior at higher angles of attack. The deformable mesh results show a greater change in lift coefficient as the airfoil is swept through the lower angles of attack. The deformable mesh result and overset mesh result seem to predict the same angle of attack where the lift coefficient will cross (around  $\alpha = 3.5^\circ$ ). It should be noted that it is unclear whether or not the experimental data was corrected for wind tunnel effects. It is also worth noting that the experimental apparatus was limited in its  $\alpha$  sweep range and allowed the airfoil to stay near the maximum value longer than would be calculated from a pure sine function.

#### E. Computational Timing

There was considerable interest in determining the computational effort required for the various mesh movement methods. Table 2 shows the computational timings for the three cases and associated mesh movement methods. A lower number in the table corresponds to a lower computational cost. As expected, where available for a given case, rigid mesh motion results in the lowest computational cost. However, there are limits to the types of problems that can be addressed with rigid mesh motion such as moving control surfaces. For these types of problems, overset and deformable mesh computations would be desired. From this study, overset computations appear to be less costly than deformable mesh computations. There is one caveat, however. The overset grid assembly process in Cobalt is handled in parallel while the deformable mesh



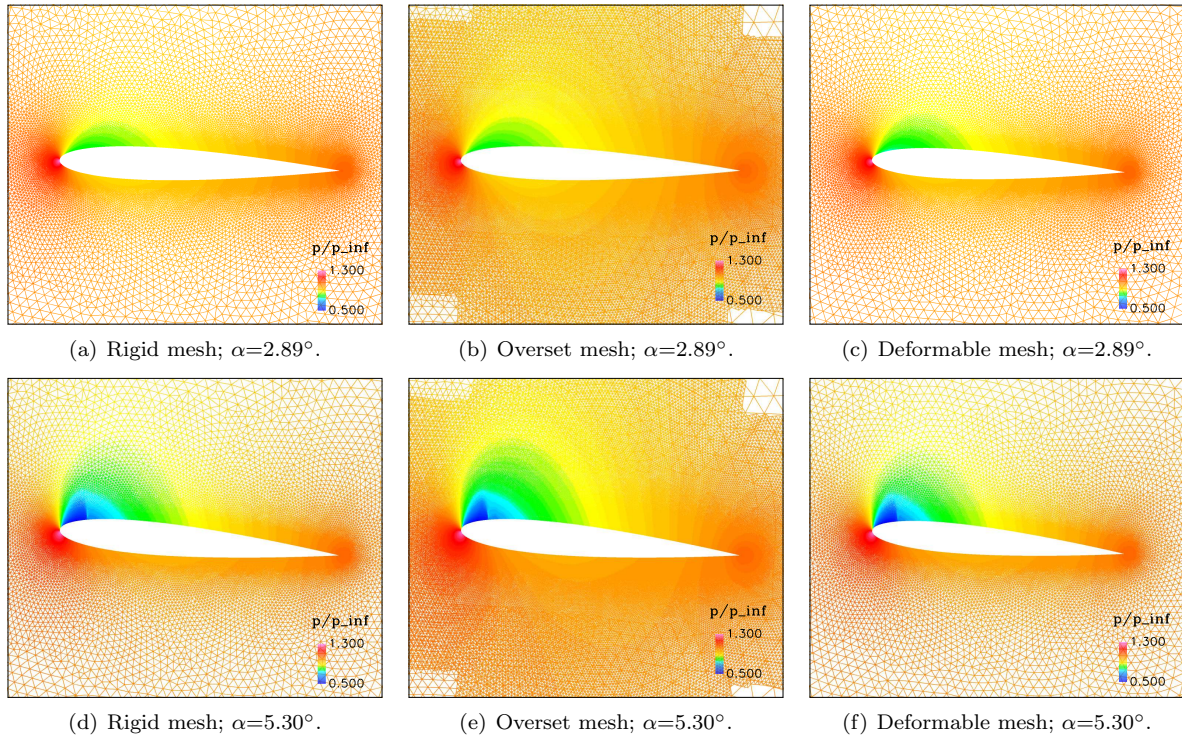


Figure 5. Comparison of grid movement schemes at two different points in time for a pitching NACA 0012 airfoil with the SA turbulence model. The simulation Reynolds number was  $4.8 \times 10^6$  and the Mach number was 0.6. Note that even the relatively low angle of attack results in transonic flow.

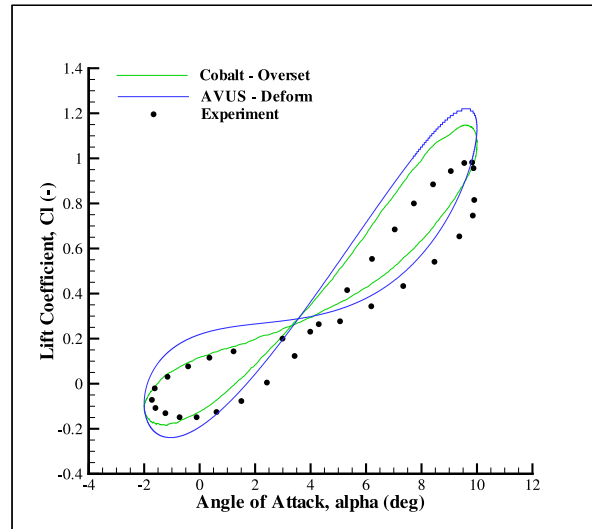


Figure 6. Results of a pitching airfoil with an independently oscillating flap. The simulations used a NACA 0012 airfoil with the SA turbulence model, a Reynolds number of  $1.63 \times 10^6$  and a Mach number of 0.4. Both methods produce similar results at higher angles of attack. At lower values of  $\alpha$ , the deformable mesh technique predicts a larger change in lift coefficient than the overset mesh technique.

process is handled serially in AVUS. There are plans to parallelize the deformable mesh process in a future release of AVUS (through the *Kestrel* project).

**Table 2. Computational timing results. Each value represents the total time to compute the solution multiplied by the number of processors, divided by the total number of iterations, divided by the total number of grid cells.**

	Quasi-steady	Pitching	Pitching & Flapping
Rigid	$1.370 \times 10^{-4}$	$1.783 \times 10^{-4}$	—
Overset	$5.024 \times 10^{-4}$	$4.649 \times 10^{-4}$	$3.997 \times 10^{-4}$
Deformation	$5.469 \times 10^{-4}$	$5.562 \times 10^{-4}$	$5.586 \times 10^{-4}$

## V. Conclusion

A comparison between rigid mesh motion, overset mesh movement, and deformable mesh movement was completed. While there was not a clear advantage to one method over another, none of the methods had a clear disadvantage. The only exception may be the rigid mesh technique as it cannot be applied to cases with moving control surfaces. When taken in aggregate, all three methods produced surprisingly similar answers within a reasonable amount of computational resources. The deformable mesh technique showed the highest computational cost but this was likely due to the serial implementation of the deformable mesh process and will likely decrease when the process is parallelized.

## Acknowledgments

This work was sponsored by the DoD HPC/AF SEEK Eagle Office Institute for High Performance Computing Applications to Air Armament (IHAAA), the Modeling and Simulation Research Center at the Air Force Academy, and the Air Force Office of Scientific Research. The computational resources were generously provided by the DoD HPCMP.

The first and second authors would also like to thank Dr. Bill Strang and Dr. Bob Tomaro of Cobalt Solutions, LLC for their assistance in debugging the overset grid module and associated mesh motion in Cobalt.

## References

- <sup>1</sup>M. Withrow, “Dr. Raymond Gordnier Discusses the Research Direction of Advanced Computational Methods,” *Air Force Research Laboratory Horizons*, April 2005.
- <sup>2</sup>C. Fremaux and R. Hall, “COMSAC: Computational Methods for Stability and Control; Part 1,” Tech. rep., NASA/CP-2004-213028/PT1, 2004.
- <sup>3</sup>C. Fremaux and R. Hall, “COMSAC: Computational Methods for Stability and Control; Part 2,” Tech. rep., NASA/CP-2004-213028/PT2, 2004.
- <sup>4</sup>J. Vos, A. Rizzi, D. Darracq, and E. Hirschel, “Navier-Stokes Solvers in European Aircraft Design,” *Progress in Aerospace Sciences*, Vol. 38, No. 8, 2002, pp. 601–697.
- <sup>5</sup>S. Görtz, D. McDaniel, and S. Morton, “Towards an Efficient Aircraft Stability and Control Analysis Capability using High-Fidelity CFD,” AIAA Paper 2007-1053, 45th AIAA Aerospace Sciences Meeting and Exhibit, 2007.
- <sup>6</sup>C. Stephens, A. Arena, Jr., and K. Gupta, “CFD-based Aeroservoelastic Predictions with Comparisons to Benchmark Experimental Data,” AIAA Paper 1999-0766, 37th AIAA Aerospace Sciences Meeting and Exhibit, 1999.
- <sup>7</sup>M. Allan, K. Badcock, and B. Richards, “CFD Base Simulation of Longitudinal Flight Mechanics with Control,” AIAA Paper 2005-0046, 43rd AIAA Aerospace Sciences Meeting and Exhibit, 2005.
- <sup>8</sup>A. Rampurwala and K. Badcock, “Treatment of Forced Flap Motions for Aeroelastic Simulations of an Arrow Wing,” AIAA Paper 2005-4962, 23rd AIAA Applied Aerodynamics Conference, 2005.
- <sup>9</sup>O. Boelens, B. Prananta, B. Soemarwoto, M. Allan, K. Badcock, and W. Fritz, “Towards an Aero-Servo-Elastic Simulation Capability for High-Performance Fighter Aircraft,” *NATO RTO-MP-AVT-123: Flow-Induced Unsteady Loads and the Impact on Military Applications*, No. 30, 2005.
- <sup>10</sup>G. Power, J. Calahan and D. Hensley, “A System for Moving-Body CFD Simulations on Overset Structured and Unstructured Grids,” AIAA Paper 2004-0716, 42nd AIAA Aerospace Sciences Meeting and Exhibit, 2004.
- <sup>11</sup>M. Rizk and B. Jolly, “Aerodynamic Simulation of Bodies with Moving Components using CFD Overset Grid Methods,” AIAA Paper 2006-1252, 44th AIAA Aerospace Sciences Meeting and Exhibit, 2006.

- <sup>12</sup>G. Fillola, M. Le Pape, and M. Montagnac, "Numerical Simulations Around Wing Control Surfaces," *24th International Congress of the Aeronautical Sciences*, 2004.
- <sup>13</sup>S. Obayashi and G. Guruswamy, "Navier-Stokes Computations for Oscillating Control Surfaces," AIAA Paper 92-4431, AIAA Atmospheric Flight Mechanics Conference, 1992.
- <sup>14</sup>M. Lesoinne and V. Kaila, "Meshless Aeroelastic Simulations of Aircraft with Large Control Surface Deflections," AIAA Paper 2005-1089, 43rd AIAA Aerospace Sciences Meeting and Exhibit, 2005.
- <sup>15</sup>J. Shang, "Three Decades of Accomplishments in Computational Fluid Dynamics," *Progress in Aerospace Sciences*, Vol. 40, No. 3, 2004, pp. 173–197.
- <sup>16</sup>A. Schütte, G. Einarsson, A. Raichle, B. Schoning, W. Monnich, J. Neumann, J. Arnold, T. Alrutz, J. Heinecke, T. Forkert, and H. Schumann, "Prediction of the Unsteady Behavior of Maneuvering Aircraft by CFD, Aerodynamic, Flight-Mechanic, and Aeroelastic Coupling," *NATO RTO-MP-AVT-123: Flow-Induced Unsteady Loads and the Impact on Military Applications*, No. 11, 2005.
- <sup>17</sup>M. Murayama, F. Togashi, K. Nakahashi, K. Matsushima, and T. Kato, "Simulation of Aircraft Response to Control Surface Deflection using Unstructured Dynamic Grids," *Journal of Aircraft*, Vol. 42, No. 2, 2005, pp. 340–346.
- <sup>18</sup>R. Meakin, "On Adaptive Refinement and Overset Structured Grids," AIAA Paper 1997-1858, 13th AIAA Computational Fluid Dynamics Conference, 1997.
- <sup>19</sup>R. Noack, "DiRTlib: A Library to add Overset Capability to Flow Solvers," *6th Symposium on Overset Composite Grid and Solution Technology*, 2002.
- <sup>20</sup>R. Noack and T. Kadanthot, "An Octree Based Overset Grid Hole Cutting Method," *8th International Conference on Numerical Grid Generation in Computational Field Simulations*, 2002, pp. 783–792.
- <sup>21</sup>J. Samareh, "Application of Quaternions for Mesh Deformation," Tech. rep., NASA/TM-2002-211646, 2002.
- <sup>22</sup>A. Gaitonde and S. Fiddes, "Three Dimensional Moving Mesh Method for the Calculation of Unsteady Transonic Flows," *Aeronautical Journal*, Vol. 99, No. 984, 1995.
- <sup>23</sup>S. Morton, R. Melville, and M. Visbal, "Accuracy and Coupling Issues of Aeroelastic Navier-Stokes Solutions on Deforming Meshes," *Journal of Aircraft*, Vol. 35, No. 5, 1998, pp. 798–805.
- <sup>24</sup>J. Batina, "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aerodynamic Analysis," *AIAA Journal*, Vol. 29, No. 3, 1991, pp. 327–333.
- <sup>25</sup>C. Farhat, C. Degand, B. Koobus, and M. Lesoinne, "Torsional Springs for Two-Dimensional Dynamic Unstructured Fluid Meshes," *Computational Methods in Applied Mechanical Engineering*, Vol. 163, 1998, pp. 231–245.
- <sup>26</sup>C. Burg, "A Robust Unstructured Grid Movement Strategy using Three-Dimensional Torsional Springs," AIAA Paper 2004-2529, 34th AIAA Fluid Dynamics Conference and Exhibit, 2004.
- <sup>27</sup>K. Stein, T. Tezduyar, and R. Benney, "Mesh Moving Techniques for Fluid-Structure Interactions with Large Displacements," *Transactions of the ASME*, Vol. 70, 2003.
- <sup>28</sup>D. Kholodar, S. Morton, and R. Cummings, "Deformation of Unstructured Viscous Grids," AIAA Paper 2005-0926, 43rd AIAA Aerospace Sciences Meeting and Exhibit, 2005.
- <sup>29</sup>X. Liu, N. Qin, and X. Hao, "Fast Dynamic Grid Deformation Based on Delaunay Graph Mapping," *Journal of Computational Physics*, No. 211, 2006, pp. 405–423.
- <sup>30</sup>W. Strang, R. Tomaro, and M. Grismer, "The Defining Methods of Cobalt<sub>60</sub>: A Parallel, Implicit, Unstructured Euler/Navier-Stokes Flow Solver," AIAA Paper 1999-0786, 37th AIAA Aerospace Sciences Meeting and Exhibit, 1999.
- <sup>31</sup>S. Morton, R. Tomaro, and R. Noack, "An Overset Unstructured Grid Methodology Applied to a C-130 with a Cargo Pallet and Extraction Parachute," AIAA Paper 2006-0461, 44th AIAA Aerospace Sciences Meeting and Exhibit, 2006.
- <sup>32</sup>S. Morton, D. McDaniel, D. Sears, B. Tillman, and T. Tuckey, "Kestrel – A Fixed Wing Virtual Aircraft Product of the CREATE Program," AIAA Paper 2009-0338, 47th AIAA Aerospace Sciences Meeting and Exhibit, 2009.
- <sup>33</sup>D. McDaniel and S. Morton, "Efficient Mesh Deformation for Computational Stability and Control Analyses on Unstructured Viscous Meshes," AIAA Paper 2009-1363, 47th AIAA Aerospace Sciences Meeting and Exhibit, 2009.
- <sup>34</sup>R. Tomaro, W. Strang, and L. Sankar, "An Implicit Algorithm for Solving Time Dependent Flows on Unstructured Grids," AIAA Paper 1997-0333, 35th AIAA Aerospace Sciences Meeting and Exhibit, 1997.
- <sup>35</sup>M. Grismer, W. Strang, R. Tomaro, and F. Witzemman, "Cobalt: A Parallel, Implicit, Unstructured Euler/Navier-Stokes Solver," *Adv. Eng. Software*, Vol. 29, No. 3-6, 1998, pp. 365–373.
- <sup>36</sup>J. Forsythe, W. Strang, and K. Hoffmann, "Validation of Several Reynolds-Averaged Turbulence Models in a 3-D Unstructured Grid Code," AIAA Paper 2000-2552, Fluids 2000 Conference and Exhibit, 2000.
- <sup>37</sup>J. Gottlieb and C. Groth, "Assessment of Riemann Solvers for Unsteady One-Dimensional Inviscid Flows of Perfect Gases," *Journal of Computational Physics*, Vol. 78, 1988, pp. 437–458.
- <sup>38</sup>R. Koomullil, "Validation of Dynamic Grid Capability in AVUS," Summer Faculty Fellowship Project Summary, University of Alabama-Birmingham, 2005.
- <sup>39</sup>"SimCenter Computational Simulation and Design Center, Mississippi State University High Performance Computing Collaboratory," <http://www.hpc.msstate.edu>.
- <sup>40</sup>D. Marcum and N. Weatherill, "Unstructured Grid Generation using Iterative Point Insertion and Local Reconnection," *AIAA Journal*, Vol. 33, No. 9, 1995, pp. 1619–1625.
- <sup>41</sup>R. Landon, "NACA 0012: Oscillatory and Transient Pitching," *AGARD-R-702: Compendium of Unsteady Aerodynamic Measurements*, NATO Research and Technology Organization, 1982.
- <sup>42</sup>A. Krzysiak and J. Narkiewicz, "Aerodynamic Loads on Airfoil with Trailing-Edge Flap Pitching with Different Frequencies," *Journal of Aircraft*, Vol. 43, No. 2, 2006, pp. 407–418.