JOURNAL OF AIRCRAFT
Vol. 37, No. 3, May–June 2000

Mesh Strategies for Accurate Computation of Unsteady Spoiler and Aeroelastic Problems

Robert E. Bartels*
NASA Langley Research Center, Hampton, Virginia 23681-0001

A modification of the spring analogy scheme that uses axial linear spring stiffness with selective spring stiffening/relaxation is presented. An alternate approach to solving the geometric conservation law is taken that eliminates the need for storage of metric Jacobians at previous time steps. The method is applied to the computation of the turbulent flow about an airfoil with a two-dimensional moving spoiler surface. The aeroelastic response at low dynamic pressure of an airfoil to a single large-scale oscillation of a spoiler surface is simulated in a study of the effect of fluid domain convergence and subiterative strategies. A critical issue in the computation of aeroelastic response with a strongly nonlinear flowfield is shown to be the convergence of the fluid domain. It is confirmed that it is possible to achieve accurate solutions with a very large time step for aeroelastic problems using the fluid solver and aeroelastic integrator as discussed. Furthermore, it is shown that a local pseudo-time-based subiterative method is essential for the computation of the present cases.

Nomenclature

a_∞^*	= speed of sound
C	= spring damping matrix
c_l	= section lift coefficient
c_m	= section moment coefficient
c_p	= pressure coefficient
c^*	= chord length
e	= specific internal energy
$\hat{F}, \hat{G}, \hat{H}$	= inviscid fluxes
$\hat{F}_v, \hat{G}_v, \hat{H}_v$	= viscous terms
$g(t)$	= spring analogy boundary condition array
h	= nondimensional plunge, h^*/c^*
i, j, k	= grid point indices
J	= metric Jacobian
K	= spring stiffness matrix
k_m	= element spring stiffness
M	= grid mass matrix
m	= subiteration index
n	= time-step index
p	= nondimensional pressure
$\boldsymbol{Q}$	= conserved solution vector, ρ, ρu, ρv, ρw, e
$\boldsymbol{q}$	= solution vector, ρ, u, v, w, p
q_∞	= dynamic pressure
$\boldsymbol{r}_{ijk}$	= grid position vector, $(x, y, z)_{ijk}$
t	= nondimensional time $t^* a_\infty^*/c^*$
U, V, W	= contravariant velocities
u, v, w	= nondimensional Cartesian velocities u^*/a_∞^*, etc.
x, y, z	= nondimensional Cartesian coordinates x^*/c^*, etc.
α	= angle of attack (pitch)
δ	= grid displacement
δ_{sp}	= spoiler deflection angle
δ_η	= central difference operator (η direction)
ζ_h	= plunge damping ratio
ζ_α	= pitch damping ratio
ξ, η, ζ	= coordinates in computational space
ρ	= density

Introduction

AEROELASTICITY is the study of the interaction of inertial, elastic, and aerodynamic forces acting on a structure. Aeroservoelasticity includes the additional mutual interaction of a control system and the aeroelastic structural response. When constructing computational algorithms to model problems in these disciplines, accuracy and robustness must receive due attention. This is especially true in view of recent interest in the accurate computation of fluid–structure interaction in the presence of a strongly nonlinear flowfield. This includes examples such as the computation of limit cycle oscillation and the aeroelastic response of a structure due to large-scale control surface motion. To address these issues of robustness and accuracy, recent aeroelastic research has focused on the development of new algorithms that integrate the structural equations of motion in synchronization with the aerodynamic equations of motion. Among the methods used or promoted are the closely coupled lagged approach[1,2] to updating the structure equations and the implicit iterative coupling of both structure and fluid.[3,4] Other approaches recently evaluated include the arbitrary Lagrangian–Eulerian[5,6] and the implicit continuous-fluid Eulerian methods.[7] Recent mesh deformation algorithm development has emphasized the robustness and efficiency necessary for coupled structure–fluid time-marching computations.[8,9] However, in the discussion of fluid–structure coupling, often the convergence of the fluid domain, as a distinctly separate issue from the coupling, is not specifically addressed.

The purpose of the present paper is to revisit old approaches to mesh deformation and the integration of the structural equations of motion. In particular the spring analogy mesh scheme is revisited with modifications that enhance its value for structured grids with complex geometry. The finite-dimensional state-space predictor–corrector method[10] also merits renewed consideration in view of recent advances in computational fluid solvers.

Several modifications of the typical approach to computational fluid dynamics with a deforming mesh have been made. For unsteady problems, a self-consistent approach to the Jacobian of the coordinate transformation is to compute it by solving an independent equation called the geometric conservation law (GCL).[6,11] In Refs. 3 and 4, the GCL fluxes are retained in the Navier–Stokes equations as source terms, replacing the time rate of change of the volume. Not only does this result in self-consistency, it also eliminates the need for storage of either the discrete time derivative of the Jacobian or of the Jacobian itself at several previous time steps. A variation of that approach is used here. The metric fluxes are retained in the present scheme on both the left- and right-hand sides of the discrete approximate factorized equations in a cell-centered finite volume

Presented as Paper 99-3301 at the AIAA 14th Computational Fluid Dynamics Conference, Norkfolk, VA, 28 June–1 July 1999; received 14 September 1999; revision received 12 December 1999; accepted for publication 27 December 1999.

*Aerospace Engineer, Aeroelasticity Branch. Senior Member AIAA.

discretization of the equations. In the references cited earlier, the flow equations were solved in finite difference form using central differencing of the flux terms with explicit and implicit artificial dissipation. Several options are also available in the present code for construction of fluxes and time accuracy. Third-order upwind Roe's flux differencing and second-order backward time differencing are used here. Subiterations either include the physical time step (t-TS subiteration) only or both physical and pseudo time step based on local Courant–Friedrichs–Lewy (CFL) number for convergence acceleration (τ-TS subiteration). This affords the opportunity to assess the effect of the increased robustness of a higher-order upwind scheme with pseudo time subiterations on the simulation of unsteady aeroelastic response to a nonlinear flowfield. Finally, the present deforming mesh scheme is used with a time-accurate field equation turbulence model. Another aspect of this work is the modification of the spring analogy scheme with axial stiffness for use with a structured grid. References 7 and 12–15 give recent examples of its use in aeroelastic computations with structured and unstructured grids. There are advantages to the spring analogy, most notably its simplicity and ability to smooth grids. However, it does not ensure positivity of control volumes and in its most typical form is memory intensive. The modified spring analogy using nonlinear torsion stiffness has been shown to preclude nonpositive control volume (and, thus, the more restrictive case of grid point crossing).[9] It does not alone allow spacing control and does not include the smoothing available from the axial spring analogy. Thus, it would seem that an optimal approach in the context of the spring analogy would be a combination of several stiffness types. It will be shown that in its present form the spring analogy scheme with linear axial stiffness is efficient and with a structured grid also allows some savings in memory. It can, therefore, provide a starting point for a more elaborate deforming mesh scheme. It is not, however, without problems. Nonpositive grid volumes can still be encountered. A partial solution is presented that is useful for at least mildly complex geometry, but it must be emphasized that the present work is considered the initial step in the development of a robust deforming mesh scheme. The intention is that future developments will include the capability of grid adaptation and torsion stiffness that will allow more complex geometry and control over grid orthogonality.

The time-accurate thin-layer Navier–Stokes equations for a dynamically deforming mesh will be presented first. This will be followed by a discussion of the mesh scheme, the method of structural integration, and results.

Thin-Layer Navier–Stokes Equations

The finite volume thin-layer Navier–Stokes code that forms the basis for this new development is CFL3D version 5.0. This code has been used extensively to simulate many unsteady two- and three-dimensional single and multizonal problems.[16,17] Rigid mesh movement capability has been developed previously.[18] The code includes multigrid and grid sequencing capability and has available many choices of turbulence model. Because the form of the thin-layer Navier–Stokes equations and the solution procedures used here are not typical, the equations and details of the solution will be presented. The differential form of the equations in a general deforming coordinate system is written in a manner similar to that in Ref. 4, namely,

$$\frac{1}{J}\frac{\partial Q}{\partial t} = -R(Q) \tag{1}$$

where

$$R(Q) = \frac{\partial(\hat{F} - \hat{F}_v)}{\partial \xi} + \frac{\partial(\hat{G} - \hat{G}_v)}{\partial \eta} + \frac{\partial(\hat{H} - \hat{H}_v)}{\partial \zeta} - Q\left(\frac{\partial}{\partial \xi}\left(\frac{\xi_t}{J}\right) + \frac{\partial}{\partial \eta}\left(\frac{\eta_t}{J}\right) + \frac{\partial}{\partial \zeta}\left(\frac{\zeta_t}{J}\right)\right) \tag{2}$$

Note that the time derivative on the left-hand side of Eq. (1) is not written in strong conservation-law form and that the last bracketed item in the residual [Eq. (2)] is composed of the grid speed fluxes from the GCL. Letting

$$\hat{F} = \begin{Bmatrix} \rho U \\ \rho U u + p\xi_x \\ \rho U v + p\xi_y \\ \rho U w + p\xi_z \\ (e + p)U - p\xi_t \end{Bmatrix} \tag{3}$$

and constructing the remaining inviscid and diffusive fluxes, we obtain the complete equation set.

The residual on the right-hand side of Eq. (2) is linearized about the most recent subiteration m and the resulting equations are approximately factored. The resulting discrete equations take the form

$$\left(\left\{\left(\frac{1+\phi'}{J\Delta\tau}\right) + \left(\frac{1+\phi}{J\Delta t}\right)\right\}\tilde{M} + \delta_\xi A^* + \Omega\tilde{M}\right)\Delta q' = \left(\frac{\phi'\Delta q^{m-1}}{J\Delta\tau} + \frac{\phi\Delta q^{n-1}}{J\Delta t} - \frac{(1+\phi)(q^m - q^n)}{J\Delta t}\right)\tilde{M} - R(Q^m) \tag{4a}$$

$$\left(\left\{\left(\frac{1+\phi'}{J\Delta\tau}\right) + \left(\frac{1+\phi}{J\Delta t}\right)\right\}\tilde{M} + \delta_\eta B^* + \Omega\tilde{M}\right)\Delta q'' = \left(\left(\frac{1+\phi'}{J\Delta\tau}\right) + \left(\frac{1+\phi}{J\Delta t}\right) + \Omega\right)\tilde{M}\Delta q' \tag{4b}$$

$$\left(\left\{\left(\frac{1+\phi'}{J\Delta\tau}\right) + \left(\frac{1+\phi}{J\Delta t}\right)\right\}\tilde{M} + \delta_\zeta C^* + \Omega\tilde{M}\right)\Delta q^m = \left(\left(\frac{1+\phi'}{J\Delta\tau}\right) + \left(\frac{1+\phi}{J\Delta t}\right) + \Omega\right)\tilde{M}\Delta q'' \tag{4c}$$

for the residual just defined, and

$$\Omega = -\left(\frac{\partial}{\partial\xi}\left(\frac{\xi_t}{J}\right) + \frac{\partial}{\partial\eta}\left(\frac{\eta_t}{J}\right) + \frac{\partial}{\partial\zeta}\left(\frac{\zeta_t}{J}\right)\right) \tag{5}$$

In the preceding equations, $\tilde{M} = \partial Q/\partial q$, $A^* = \partial(\hat{F} - \hat{F}_v)/\partial q$, and so forth. The solution vector is defined by $\Delta \boldsymbol{q}^m = \boldsymbol{q}^{m+1} - \boldsymbol{q}^m$. Because the fluxes contained in Eq. (2) (last term) and (5) are similar in form to the fluxes of the flow quantities, both can be computed easily within the same subroutines. This has required the addition of 20–40 coding lines to each of the inviscid flux subroutines (right- and left-hand side terms). Furthermore, the desired first- or second-order accuracy in time is automatically achieved by making use of the metric time derivatives already being used in the spatial fluxes of conserved flow quantities. Because the volumes at several previous time steps are not stored, very little additional memory and no change in the overall code data structure is required. Using this approach does require that Eq. (5) and the metric fluxes of Eq. (2) be recomputed at each subiteration and added to Eqs. (4a–4c), resulting in a small added computational effort.

The metric fluxes in the new terms are evaluated at cell face centers and are also differenced in time in a way that is consistent with the temporal differencing of conserved flow variables. This results in consistency between new metric fluxes of Eq. (5), the metric time derivatives in the physical fluxes and the integral of $J^{-1}Q_\tau$ over the control volume, and, finally, in a temporally and spatially consistent solution of the GCL. The value of the metric Jacobian at time step $n+1$ is used everywhere in Eqs. (2), (4), and (5). Equations (4) also include local CFL-based pseudo time stepping (τ-TS subiteration) for convergence. Based on the values of ϕ and ϕ', the option of first- or second-order physical and pseudo time stepping can be exercised. In the present computations the temporal discretization of Eq. (1) is accomplished via second-order backward differencing, that is, $\phi = \frac{1}{2}$. The discrete inviscid fluxes use third-order upwind-biased

Roe's flux difference splitting and a minmod flux limiter. Multigrid is used in the following computations.

CFL3D version 5.0 includes many turbulence models. The turbulence model used in the present computations is the Spalart–Allmaras field equation model, primarily because of its robustness at large-time-step size. The differential equation for eddy viscosity in the Spalart–Allmaras model is solved in nonconservation form. This simplifies the procedure when using an unsteady field equation turbulence model because no additional treatment of metrics other than consistent time differencing is required for an unsteady solution.

Mesh Scheme

The mesh scheme is a modification of the spring analogy using axial spring stiffness. The approach of Batina[12] can be written, in general, for an unsteady problem,

$$M\ddot{\delta} + C\dot{\delta} + K\delta = g(t) \tag{6}$$

In the present formulation $M = C = 0$. The element stiffness would be defined by $k_{ij} = 1/l_{ij}$ where $l_{ij} = \|\boldsymbol{r}_j - \boldsymbol{r}_i\|_2^{1/2}$ and i and j represent two adjacent nodes. For an unstructured grid, storage of stiffness values based on the current locations of each node pair is required. The spring stiffness would be updated as the mesh deforms. For a structured grid the problem is somewhat simplified. Spring stiffness in the mesh interior can be controlled by the spacing of the appropriate boundary grid points. For instance, if the nodes at (i, j, k) and $(i, j, k+1)$ are considered and if boundary spacing at boundaries $i = 1$ and $i = i_{\max}$ are used,

$$k_m = 1/l_{k+1,k} \tag{7}$$

where $m = k + 1$ designates in this case the volume edge between the k and $k + 1$ grid points, and

$$l_{k+1,k} = f(\|\boldsymbol{r}_{k+1} - \boldsymbol{r}_k\|_2)_{i=1}^{-\frac{1}{2}} + (1 - f)(\|\boldsymbol{r}_{k+1} - \boldsymbol{r}_k\|_2)_{i=i\max}^{-\frac{1}{2}}$$

$$f = \frac{i_{\max} - i}{i_{\max} - 1} \tag{8}$$

Because these stiffness values are set at the start of the computations they do not vary in time. They also require storage only for each of the six computational boundaries. Linear spring stiffness allows the possibility of grid point crossings, but because this is a restricted case of the general problem of grid lines crossing, the better solution is the addition of nonlinear torsion stiffness.[8,9] This can be added to the present spring stiffness at a future time. Finally, note that the axial stiffness approach used here results in smoothing of the mesh and also allows adaptation based on the flow solution.[8,19]

The problem of grid collapse around convex surfaces is handled by selectively increasing/decreasing stiffness based on surface curvature. Stiffness values in two coordinate planes normal to a surface are varied based on surface curvature in the coordinate projection of those planes onto the surface. The final mesh is the weighted combination of the two planar solutions. Take, for example, a $\xi\eta$-plane solid surface at $k = 1$ from which curvature information is required to construct the grid. Mathematically, the expression for interior grid point location can be written

$$\boldsymbol{r}_{ijk} = \frac{(1-\varepsilon)\left[f_1 \sum \bar{k}_{m1}\boldsymbol{r}_{m1} + f_2 \sum \bar{k}_{m2}\boldsymbol{r}_{m2}\right]}{\left[f_1 \sum \bar{k}_{m1} + f_2 \sum \bar{k}_{m2}\right]} + \varepsilon P \tag{9}$$

where

$$f_1 = e^{C_1\Lambda_\xi}, \qquad f_2 = e^{C_1\Lambda_\eta}$$

In Eq. (9) $\bar{k}_{m1} = \frac{1}{2}k_{m1}e^{-C_1\Lambda_\xi}$ and $\bar{k}_{m2} = \frac{1}{2}k_{m2}e^{-C_1\Lambda_\eta}$ for the volume edges including the endpoints $k + 1$ and $k - 1$, and $\bar{k}_{m1} = k_{m1}$, $\bar{k}_{m2} = k_{m2}$ otherwise. The index $m1$ in the summation ranges over the four nearest grid points adjacent to ijk in the $\xi\zeta$ plane (indices j, k) and index $m2$ in the $\eta\zeta$ plane (indices i, k). The constant C_1 is a gain factor that adjusts sensitivity to surface curvature. The surface curvature parameters Λ_ξ and Λ_η can be arrived at by considering, for example, the surface grids of Fig. 1 in the ξ (j index) direction. A measure that accounts, at grid i, j, k, for convex or concave curvature at the surface point i, j and $k = 1$ is

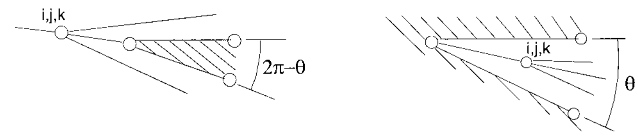

Fig. 1 Orientation of surface and interior grids.

$$\Lambda_\xi = \frac{\sin\theta}{1 - \cos\theta} \tag{10}$$

In terms of grid geometry this is

$$\Lambda_\xi = \frac{\Delta\boldsymbol{r}_i \cdot (\Delta\boldsymbol{r}_{j+} \times \Delta\boldsymbol{r}_{j-})}{|\Delta\boldsymbol{r}_i|(|\Delta\boldsymbol{r}_{j+}||\Delta\boldsymbol{r}_{j-}| - \Delta\boldsymbol{r}_{j+} \cdot \Delta\boldsymbol{r}_{j-})} \tag{11}$$

$$\Lambda_\eta = \frac{-\Delta\boldsymbol{r}_j \cdot (\Delta\boldsymbol{r}_{i+} \times \Delta\boldsymbol{r}_{i-})}{|\Delta\boldsymbol{r}_j|(|\Delta\boldsymbol{r}_{i+}||\Delta\boldsymbol{r}_{i-}| - \Delta\boldsymbol{r}_{i+} \cdot \Delta\boldsymbol{r}_{i-})} \tag{12}$$

where $\Delta\boldsymbol{r}_i = \boldsymbol{r}_{i+1} - \boldsymbol{r}_{i-1}$, $\Delta\boldsymbol{r}_j = \boldsymbol{r}_{j+1} - \boldsymbol{r}_{j-1}$, $\Delta\boldsymbol{r}_{j+} = \boldsymbol{r}_{j+1} - \boldsymbol{r}_j$, $\Delta\boldsymbol{r}_{i+} = \boldsymbol{r}_{i+1} - \boldsymbol{r}_i$, $\Delta\boldsymbol{r}_{i-} = \boldsymbol{r}_{i-1} - \boldsymbol{r}_i$, etc.

In practice, the values of Λ_ξ and Λ_η are limited in upper and lower bound, currently $\Lambda_\xi = \min(C_1\Lambda_\xi, 0)$ or $\Lambda_\xi = \max(C_1\Lambda_\xi, -12)$, and similarly for Λ_η. An update of $f_1(\Lambda_\xi)$ and $f_2(\Lambda_\eta)$ are required once per time step for each surface or grid edge plane. This leaves only the computation of Eq. (9) to solve for the interior of the mesh. This is a linear equation typically requiring 1–2 iterations of a Gauss–Seidel procedure for moderate motions of the grid. Finally, the function P defines an outward normal to the surface. Setting $\varepsilon = \exp(-C_2 k)$ ensures that at least several grid points near the surface remain orthogonal to the surface. The complete mesh solution per time step involves an update using initialization of the mesh via transfinite interpolation and a second step involving one to two iterations of Eq. (9) to arrive at the final mesh.

Aeroelastic Method

The-time marching simulation of the aeroelastic responses is obtained using the finite-dimensional state-space predictor–corrector method, described in Refs. 10 and 20, that solves the decoupled modal structural equations of motion. Several versions of the finite-dimensional state-space variable method are presented in Refs. 10 and 20. The predictor step marches the structure using the solution of the modal equations at step n to get the surface deflection at time step $n + 1$. This solution is based on a second-order (trapezoidal) extrapolation of the generalized aerodynamic forces at n and $n - 1$. This provides the surface shape for a recomputation of the fluid mesh and the fluid-domain solution at $n + 1$. After a solution of the fluid domain involving multiple subiterations, the corrector step then solves the modal equations at the time step $n + 1$ using the averaged generalized forces at n and $n + 1$.

Results

The results of this paper are aimed at assessing what is required to compute unsteady two-dimensional spoiler and coupled spoiler aeroelastic problems. Specifically, the issues addressed are the geometric modeling of spoilers and the convergence and time-step behavior of the present fluid and aeroelastic solvers. In the following cases the airfoil/spoiler geometry is based on the benchmark active controls technology (BACT) NACA 0012 test model.[21] The spoiler is modeled here as a ramp and backward-facing step. A two-dimensional fluid mesh is shown in Fig. 2. The backward step behind the spoiler trailing edge has a slight slope modeled by spacing three grid points between the spoiler trailing edge and the wing surface. This clearly does not model the cavity beneath the spoiler, nor the gap between the spoiler and flap leading edge. As the spoiler moves, the spacing over the backward step is constrained via arc length to remain more or less true to the original grid spacing over the wing. Spoiler motion is typically accomplished by shearing wing surface

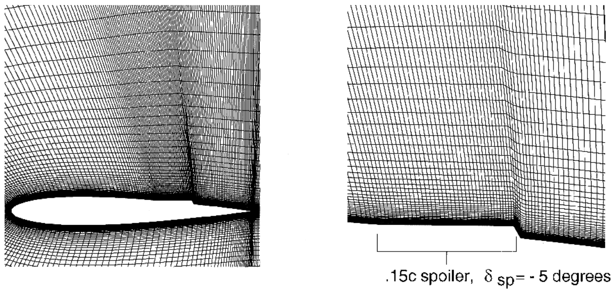

Fig. 2 Two-dimensional mesh for oscillating spoiler study.

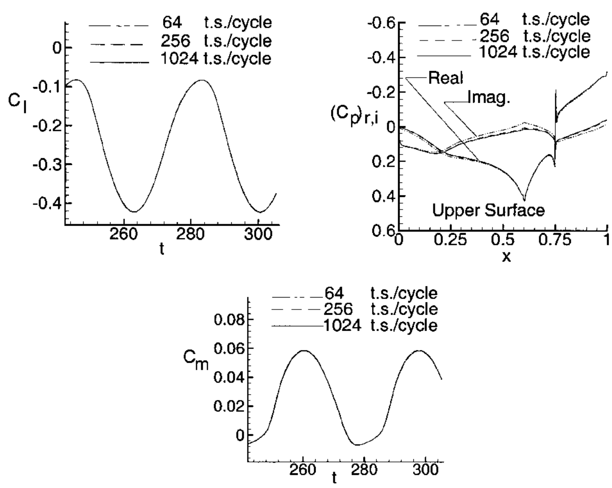

Fig. 3 Time step study for two-dimensional NACA 0012 airfoil with oscillating spoiler: M_∞ = 0.77, α = 0 deg, and $\alpha_{sp} = -5 - 4.5 \sin(2\, M_\infty\, kt)$ deg.

grids or, at best, by using a two-coordinate modal control surface motion, either of which results in surface warping. The present approach embodies true rigid-body control surface rotation.

The unsteady Navier–Stokes solutions in Fig. 3 investigate time step convergence, which is important before launching into more costly three-dimensional oscillating spoiler computations. They are especially challenging because they involve the growth and collapse of a large region of recirculating flow. They require significantly more subiterations (50–60 at 64 time steps per cycle). This computation also required tuning of the τ-TS CFL for extremely large physical time steps to reach reasonable accuracy at each time step. Nevertheless, the convergence at each time step was carefully monitored. The L_2 norm was maintained at approximately 10^{-7} or better at each time step. This required 14 subiterations per time step at 256 time steps per cycle and 4 subiterations at 1024 time steps per cycle. As can be seen in Fig. 3, there is no discernable difference in the lift and moment coefficients in going from 1024 to 64 time steps per cycle of spoiler oscillation. However, the out-of-phase part of the transform of the unsteady pressure coefficient shows some reduction in accuracy for the larger time steps. Although all of the computations appear to be sufficiently accurate for engineering purposes, it is clear that for high accuracy, 260–300 time steps/cycle or more are required. This is true, for example, for simulation of response to control system input at a condition near flutter onset, which is typically sensitive to the phase angle of the aerodynamic forcing.

The simulation of an aeroelastic transient induced by a single oscillation of a two-dimensional spoiler for an airfoil subject to pitch and plunge is presented in Figs. 4–6 and Tables 1 and 2. This case provides a nominal test of the performance of the complete fluid-mesh-structure solution technique. The aeroelastic response is due to a single large-scale asymmetric oscillation of the spoiler. The spoiler was oscillated at approximately the free vibration frequency of the plunge mode. The dynamic pressure (q_∞ = 20 psf) is low enough to give a moderate aeroelastic response for this size of spoiler motion. After an aeroelastic response using the mesh shown

Table 1 Computed damping ratios for two aeroelastic modes[a]

Δt	Time steps (per plunge cycle)	No. of fluid domain subiterations	ζ_h	ζ_α
0.0092	8192	2	0.02781	0.05229
0.0366	2048	6	0.02781	0.05230
0.1465	512	12	0.02779	0.05228
0.5859	128	32	0.02761	0.05218

[a]Same model and condition as Fig. 4.

Table 2 Effect of number of subiterations on damping ratio[a]

Δt	Time steps (per plung cycle)	No. of fluid domain subiterations	ζ_h
0.5859	128	32	0.02761
0.5859	128	20	0.02750
0.5859	128	8	0.02648

[a]Same model and condition as Fig. 4.

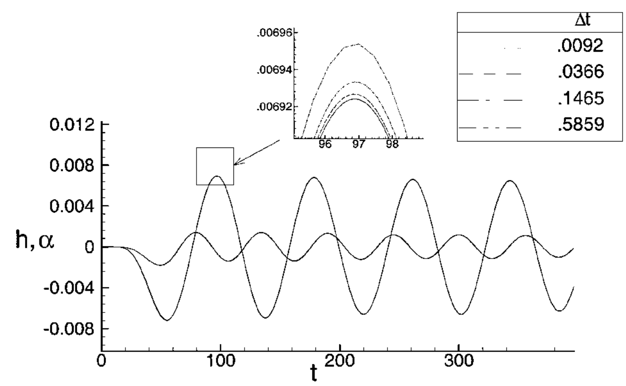

Fig. 4 Computed aeroelastic response due to a single spoiler oscillation (9.5-deg excursion), two-dimensional BACT model: M_∞ = 0.77, α = 0 deg, q_∞ = 20 psf, and U_∞ = 373 fps.

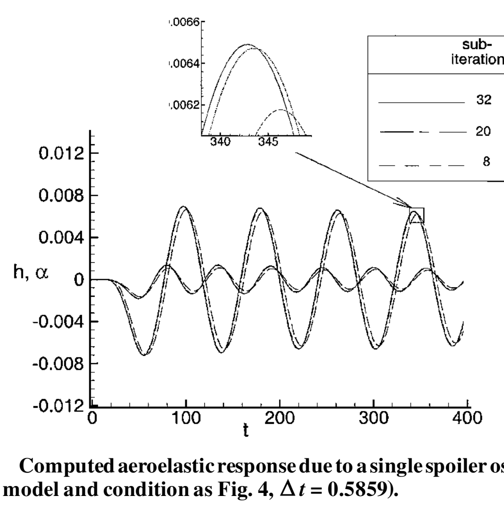

Fig. 5 Computed aeroelastic response due to a single spoiler oscillation (same model and condition as Fig. 4, Δt = 0.5859).

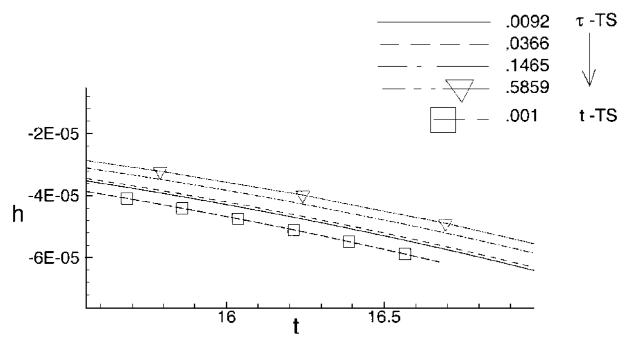

Fig. 6 Comparison of t-TS and τ-TS time stepping for aeroelastic response (same model and condition as Fig. 4).

in Fig. 2 was computed, it was found that the aeroelastic transient response was sensitive to grid spacing asymmetry between the upper and lower surfaces. A mesh that is symmetric and has the same spacing over both the upper and lower surface was used as the final grid. This grid had dimensions 319×105. Furthermore, the quality of the grid especially around the spoiler region was found to considerably impact the time-step convergence of the aeroelastic solutions. After some effort at improving grid quality, the described solutions were obtained.

The solutions of Fig. 4 give transient pitch and plunge response calculated with four time-step sizes. Note that the maximum CFL number for the largest time step ($\Delta t = 0.5859$) is approximately 2×10^6. The convergence of the plunge mode response with time step size reduction can be clearly seen in the detailed inset in Fig. 4. Table 1 summarizes data from these runs. The fluid-domain subiterations shown in Table 1 are what were required to converge the solution (L_2 norm of density) to around 10^{-7} at each time step. Between 2048 and 8192 time steps/plunge cycle, there is virtually no difference in the damping ratios. Between the largest and smallest time steps, the difference is less than 1%. The convergence of the aeroelastic response with number of fluid-domain subiterations at the largest time step can be seen in Fig. 5. It appears from Fig. 5 that 32 subiterations of the fluid domain per time step results in a very nearly converged solution. One can also see from Fig. 5 and from Table 2 that if less accuracy is required, fewer subiterations will also result in a solution. On the other hand, a computation that was performed using only a physical time step (t-TS subiteration) was unstable for time steps larger than $\Delta t = 0.001$. The solution at $\Delta t = 0.0012$, shown in Fig. 6, diverged shortly after the final time step shown. Even if it had not diverged, this small time step makes this problem solution infeasible with the t-TS method, and necessitates CFL-based local time step subiteration (identified as τ-TS in Fig. 6). The accuracy of the t-TS solution is clearly not established with only one computation. However, if the solutions with τ-TS subiterations are converging to the correct solution, as it would appear, the t-TS solution even at $\Delta t = 0.001$ does not appear to be more accurate than the τ-TS solution at a time step 100 times larger.

Conclusions

A numerical scheme for the Euler and Navier–Stokes equations in a general dynamically deforming coordinate system that incorporates the geometric conservation law into the governing equations has been evaluated in this paper. By casting the equations in the form shown here, a somewhat more economical scheme is obtained for problems requiring deforming meshes. A structured mesh scheme, based on the spring analogy, has been applied to moderately complex surface geometry. Its performance has been assessed for time steps spanning a large range, applied to problems from oscillating two-dimensional spoilers to the transient response of an airfoil. The spoilers have been modeled as a ramp and backward step and the oscillation by true rigid-body rotation. The mesh scheme does not appear to result in mesh entanglement for the geometry and mesh motion considered here even with an extremely large time step ($\mathrm{CFL} \approx 2 \times 10^6$). This is due to the use of mesh initialization via transfinite interpolation, to the fluid mesh near the moving surface undergoing motion prescribed by the surface, and to the selective relaxing/stiffening of the spring stiffness in the spring analogy scheme. Even at very large time steps, the scheme appears to retain considerable accuracy if a sufficient number of τ-TS subiterations are used. In the case of strongly nonlinear flow driven by spoiler oscillation, numerical results have shown that the error due to an increasing time step that does arise is mainly in the out-of-phase component. This is expected for a second-order time-accurate scheme and has implications for the accurate computation of flutter onset and other aeroelastic phenomena. Finally, aeroelastic capability has been demonstrated for the damped transients of a wing having pitch and plunge modes. The case computed was at low dynamic pressure in which the modes were initially perturbed by a single-large-scale oscillation of the spoiler surface. The results show that accuracy at very large-time-step sizes, similar to that used in the oscillating spoiler computations, can be achieved here as well. This suggests that for these cases and this solution method, the convergence of the fluid solution is a dominant factor in the accuracy of the structure–fluid solution. Again, this is possible because of the τ-TS flowfield solution. Because of stability limitations, the largest time step possible with t-TS subiteration was at least 600 times smaller than the largest time step with τ-TS subiteration.

References

[1]Schuster, D. M., Vadyak, J., and Atta, E., “Static Aeroelastic Analysis of Fighter Aircraft Using a Three-Dimensional Navier–Stokes Algorithm,” *Journal of Aircraft*, Vol. 27, No. 9, 1990, pp. 820–825.

[2]Guruswamy, G. P., “Unsteady Aerodynamic and Aeroelastic Calculations for Wings Using Euler Equations,” *AIAA Journal*, Vol. 28, No. 3, 1990, pp. 461–469.

[3]Melville, R. B., Morton, S. A., and Rizzetta, D. P., “Implementation of a Fully Implicit, Aeroelastic Navier–Stokes Solver,” AIAA Paper 97-2039, June–July 1997.

[4]Morton, S. A., Melville, R. B., and Visbal, M. R., “Accuracy and Coupling Issues of Aeroelastic Navier–Stokes Solutions on Deforming Meshes,” *Journal of Aircraft*, Vol. 35, No. 5, 1998, pp. 798–805.

[5]Bendiksen, O. O., “A New Approach to Computational Aeroelasticity,” AIAA Paper 91-0939, 1991.

[6]Farhat, C., Lesoinne, M., and Maman, N., “Mixed Explicit/Implicit Time Integration of Coupled Aeroelastic Problems: Three-Field Formulation, Geometric Conservation and Distributed Solution,” *International Journal for Numerical Methods in Fluids*, Vol. 21, No. 10, 1995, pp. 807–835.

[7]Vahdati, M., and Imregun, M., “An Application of the ICED-ALE Methodology to Integrated Non-Linear Aeroelasticity Analyses,” *Journal of Engineering Computations*, Vol. 14, No. 3, 1997, pp. 281–305.

[8]Nakahashi, K., and Deiwert, G. S., “Self-Adaptive-Grid Method with Application to Airfoil Flow,” AIAA Paper 85-1525, July 1985.

[9]Farhat, C., Degand, C., Koobus, B., and Lesoinne, M., “An Improved Method of Spring Analogy for Dynamic Unstructured Fluid Meshes,” AIAA 1998 Structural Dynamics and Materials Conf., AIAA Paper 98-2070, April 20–23, 1998.

[10]Edwards, J. W., Bennett, R. M., Whitlow, W., Jr., and Seidel, D. A., “Time-Marching Transonic Flutter Solutions Including Angle-of-Attack Effects,” *Journal of Aircraft*, Vol. 20, No. 11, 1983, pp. 899–906.

[11]Thomas, P. D., and Lombard, C. K., “Geometric Conservation Law and Its Application to Flow Computations on Moving Grids,” *AIAA Journal*, Vol. 17, No. 10, 1979, pp. 1030–1037.

[12]Batina, J. T., “Unsteady Euler Airfoil Solutions Using Unstructured Dynamic Meshes,” AIAA Paper 89-0115, Jan. 1989.

[13]Lee-Rausch, E. M., and Batina, J. T., “Wing Flutter Boundary Prediction Using Unsteady Euler Aerodynamic Method,” *Journal of Aircraft*, Vol. 32, No. 2, 1995, pp. 416–422.

[14]Blom, F. J., and Leyland, P., “Analysis of Fluid–Structure Interaction by Means of Dynamic Unstructured Meshes,” *4th International Symposium on Fluid–Structure Interactions, Aeroelasticity, Flow-Induced Vibration and Noise*, Vol. 1, American Society of Mechanical Engineers, Farfield, NJ, 1997, pp. 3–10.

[15]Slone, A. K., Pericleous, K., Bailey, C., and Cross, M., “Dynamic Fluid–Structure Interactions Using Finite Volume Unstructured Mesh Procedures,” *Proceedings of the CEAS International Forum on Aeroelasticity and Structural Dynamics*, Associazione Italiana di Aeronautica ed Astronautica (AIDAA), Rome, 1997.

[16]Rumsey, C., Sanetrik, M., Biedron, R., Melson, N., and Parlette, E., “Efficiency and Accuracy of Time-Accurate Turbulent Navier–Stokes Computations,” *Computers and Fluids*, Vol. 25, No. 2, 1996, pp. 217–236.

[17]Rumsey, C. L., and Vatsa, V. N., “Comparison of the Predictive Capabilities of Several Turbulence Models,” *Journal of Aircraft*, Vol. 32, No. 3, 1995, pp. 510–514.

[18]Anderson, W. K., Thomas, J. L., and Rumsey, C. L., “Extension and Applications of Flux-Vector Splitting to Unsteady Calculations on Dynamic Meshes,” AIAA 8th Computational Fluid Dynamics Conference Proceedings, AIAA Paper 87-1152-CP, June 1987.

[19]Catherall, D., “Solution-Adaptive Grids for Transonic Flows,” *Numerical Grid Generation in Computational Fluid Mechanics*, edited by S. Sengupta, J. Hauser, P. R. Eiseman, and J. F. Thompson, Pineridge Press, Swansea, U.K., 1988, pp. 329–338.

[20]Cunningham, H. J., Batina, J. T., and Bennett, R. M., “Modern Wing Flutter Analysis by Computational Fluid Dynamics Methods,” *Journal of Aircraft*, Vol. 25, No. 10, 1988, pp. 962–968.

[21]Rivera, J. A., Dansberry, B. E., Bennett, R. M., Durham, M. H., Silva, W. A., “NACA 0012 Benchmark Model Experimental Flutter Results with Unsteady Pressure Distributions,” AIAA Paper 92-2396, April 1992 AIAA/ASME/ASCE/AHS/ASC 33rd Structures, Structural Dynamics, and Materials Conference, Dallas, TX, April 1992.