Newton–Krylov Strategy for Compressible Turbulent Flows on Unstructured Meshes

Philippe Geuzaine*
University of Colorado at Boulder, Boulder, Colorado 80309-0429

Introduction

THE objective of this Note is to investigate the performance of Newton–Krylov methods applied to the solution of compressible high Reynolds number flows around two-dimensional geometries. Although these methods have already been employed successfully for the computation of inviscid and viscous flows,[1–3] very few attempts have been made to apply these techniques to the simulation of turbulent flows with partial differential equation turbulence models.[4] In this Note, which summarizes the results presented in Refs. 5 and 6, we consider the one-equation Spalart–Allmaras model[7] and the two-equation k–ω model,[8] and we examine the performance of the Newton–Krylov solver for the computation of the turbulent flow over a NACA 0012 airfoil, an RAE 2822 airfoil, and a two-element airfoil.

Spatial Discretization

The spatial discretization of the averaged Navier–Stokes equations is obtained by a cell-centered finite volume technique. The unknowns are stored at the cell centroids, and the mesh cells, which can be arbitrary polygonals, are taken as the control volumes. The discretization of the advective term is performed by an upwind scheme based on Roe's approximate Riemann solver. To obtain a higher-order scheme, a piecewise discontinuous linear reconstruction of the mean flow variables is employed in each control volume.[9,10] The monotonicity of the solution is achieved by introducing a discontinuity detector[9] in the reconstruction procedure when the solution contains steep gradients. The viscous term is discretized by a compact linearity preserving scheme based on the diamond path stencil.

The modeling of the turbulence is performed by solving either the one-equation Spalart–Allmaras model[7] or the two-equation k–ω model.[8] Because the governing equations of these turbulence models are similar to the mean flow equations, the finite volume discretization described earlier can be equally applied to these equations. Notice that to ensure the positivity of the turbulence variables, a constant reconstruction is selected for them.

Implicit Time Integration

Because we are only interested in computing steady-state solutions, we could apply Newton's method directly to the nonlinear equations $f(u) = 0$. However, to widen the domain of convergence of the method, a pseudotransient form where the time derivative is approximated by backward differencing and a mesh sequencing procedure are considered. The sequence of iterates u^n is defined by

$$u^{n+1} = u^n - [(I/\Delta t^n) + J(u^n)]^{-1} f(u^n) \tag{1}$$

with the Jacobian matrix $J(u) = \partial f/\partial u$. The preceding equation is nothing but the result of one backward Euler step and one Newton correction step. Because m unknowns are stored per node [in two dimensions, $m = 5$ or 6 depending on whether the computation is performed with the Spalart–Allmaras[7] or the k–ω (Ref. 8) turbulence model] and because the mean flow equations and the turbulence equations are solved in a form fully coupled, a Jacobian entry $J_{i,j}$ is a small dense $m \times m$ matrix. To recover a full Newton method at the last stage of the time integration process, the timestep Δt^n is increased inversely to the residual reduction.

A restarted version of the GMRES[11] algorithm solves the linear system. One of the most important aspects of all Krylov methods is that the Jacobian matrix $J(u)$, arising from the linearization of the function $f(u)$, is never needed explicitly. The only operations where J is required are matrix–vector multiplications $w = Jv$, which can be approximated by a first-order forward finite difference formula

$$J(u)v = [f(u + \epsilon v) - f(u)]/\epsilon \tag{2}$$

where ϵ is a perturbation. This approach is very appealing because there is no need to compute the Jacobian matrix and to store it in memory. This saving in memory is, however, obtained at the expense of increased computational time because each matrix–vector product requires an additional evaluation of the nonlinear function f. For noncompact spatial discretization schemes, which make the computation and the storage of the complete Jacobian matrix prohibitive, Jacobian-free methods are very attractive because they retain a Newton-type convergence without having to compute and to store the higher-order Jacobian matrix. Note that this approach is not really fully matrix-free because it still requires a preconditioning matrix for efficiency purposes. Nevertheless, the latter can be based on a lower-order discretization scheme, thus limiting both computational and storage costs.

The baseline preconditioning technique used in this study is an ILU(0) factorization of the distance-1 Jacobian matrix. This matrix is based on a lower-order discretization of the equations because it involves only the edge and vertex neighbors. For comparison purposes, more complex ILU(k) factorizations and the distance-2 Jacobian matrix, namely, the matrix based on the higher-order discretization, are also considered. Because the analytical computation of the entries of the Jacobian matrix used for preconditioning purposes can be complex and computationally expensive, they are evaluated by a finite difference formula. Because highly stretched grids are needed for turbulent flow simulations, all of the computations are performed in double precision. However, it has been observed that storing the preconditioning matrix in single precision does not deteriorate the performance of the Krylov solver, while allowing a significant saving in memory.

At first sight, because a finite difference Krylov method is employed for solving the linear system, the strong imposition of the boundary conditions at solid surfaces requires only the modification of the residual vector. However, the Krylov solver must be preconditioned to achieve a high efficiency. Therefore, the boundary conditions must be reflected in the Jacobian matrix used as a preconditioning matrix. This is easily done by linearizing the residual.[10]

Whereas the computations with the Spalart–Allmaras model can be directly started from a uniform flowfield with the higher-order scheme, the stiffness of the k–ω model imposes a more careful startup procedure. To obtain an initial solution from which Newton's method converges, 100 nonlinear iterations are performed at a Courant–Friedrichs–Lewy (CFL) number equal to 1 with a constant reconstruction scheme and a modified linearization of the source terms, such that only the negative source terms on the right-hand side are linearized whereas the positive source terms are treated explicitly, starting from a uniform flowfield. Keeping the same spatial discretization scheme but using an exact linearization of the source terms, the nonlinear residual can then be driven to machine accuracy.

Numerical Applications

The efficiency of the Newton–Krylov method used in conjunction with a mesh sequencing strategy is assessed by calculating the fully turbulent flow (i.e., no transition points are specified) over a NACA 0012 airfoil, an RAE 2822 airfoil, and a two-element airfoil. A complete description of the geometry of these test cases as well as a detailed comparison of the numerical solutions with experimental data is provided in Ref. 6.

Received 11 February 2000; revision received 11 August 2000; accepted for publication 10 October 2000.

*Research Associate, Department of Aerospace Engineering Sciences, Center for Aerospace Structures. Member AIAA.

Turbulent Flow over a NACA 0012 Airfoil

This test case consists of the computation of the turbulent flow over a NACA 0012 airfoil at $M_\infty = 0.502$ and 1.77 deg of incidence.[12] The Reynolds number based on the airfoil chord is equal to 2.91×10^6. A hybrid Cartesian/quad mesh containing 14,110 cells is generated for this simulation. A view of this grid is presented in Fig. 1a.

The nonlinear convergence histories for the one-equation Spalart–Allmaras (S–A) model[7] are reported in Fig. 1b in terms of the computational work. The computational work is defined as the computational time normalized by the cost of one fine mesh residual evaluation. The computation is started from a uniform flowfield with an initial CFL number equal to 10. Unless stated otherwise, GMRES(50) is used with a maximum of two restarts, and the preconditioning of the linear system is achieved by an ILU(0) factorization of the distance-1 Jacobian matrix. Because the turbulence equation is solved in form fully coupled with the mean flow equations, a Newton-type convergence is also obtained for this equation.

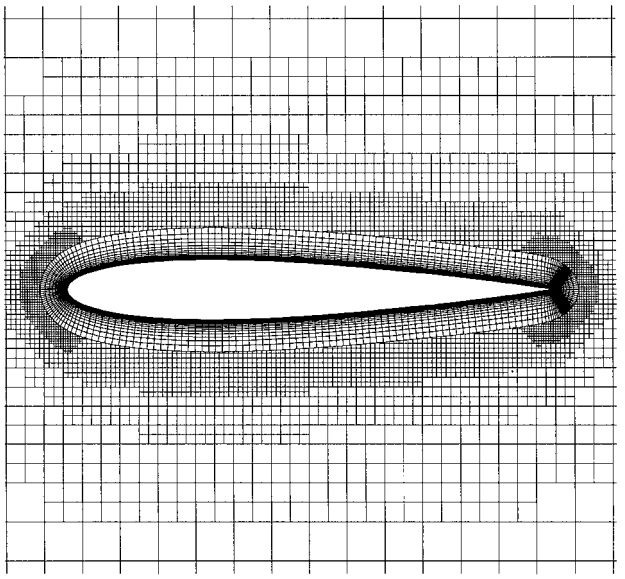
A general trend that can be observed in Fig. 1b is that many iterations are required before the turbulence is sufficiently developed, after which a fast convergence is obtained. To accelerate the development of the turbulence field, a mesh sequencing procedure can be used. For that purpose, two coarse meshes (3896 and 1184 cells) are obtained from the finest grid by a full coarsening procedure. Figure 1c reports the convergence histories of the mesh sequencing procedure. A linear reconstruction scheme is used on the three meshes. The computation is started from a uniform flowfield on the coarsest mesh with an initial CFL number equal to 10. The initial CFL number on the other meshes is equal to 100 because the initial guess on these grids is closer to the final solution. Although the residual still exhibits a plateau on the coarsest mesh, the overall computing time is approximately reduced by a factor 2.5 compared to the single-grid computation. Whereas with a single-grid approach about 160 nonlinear iterations are necessary to drive the residual norm to machine accuracy on the finest mesh, only 14 nonlinear iterations are required on the finest mesh with the mesh sequencing strategy.

Table 1 summarizes the computational costs of different preconditioning techniques on the finest mesh. In Table 1, t_{eval} and t_{fac} denote, respectively, the computational time required to evaluate and to factorize the preconditioning matrix. The memory size necessary to store the factorization is denoted by size_{fac}. The computational times required to perform 50 GMRES iterations and to reduce the nonlinear residual by 10 orders of magnitude are denoted, respectively, by $t_{\text{GMRES(50)}}$ and t_{total}. Finally, N_{total} and N_{FE} denote, respectively, the number of nonlinear iterations and the number of function evaluations (FE) to reduce the nonlinear residual norm by 10 orders of magnitude. Because we use a finite difference version of GMRES, N_{FE} also represents the total number of GMRES iterations. As shown in Table 1, the computational time on the finest mesh can be reduced by increasing the level of fill-in of the ILU(k) factorization. However, it turns out that an optimum value for k exists. The performance of the distance-2 preconditioning matrix is also reported in Table 1. Its behavior is different from inviscid flow computations[5] because its use with ILU(0) results in a stalled convergence after a reduction of the residual norm that is less than two orders of magnitude. This stalled convergence arises from the inaccuracy in the solution of the linear system. In addition to the prohibitive memory requirements of the distance-2 preconditioning matrix, this behavior precludes its use for practical applications.

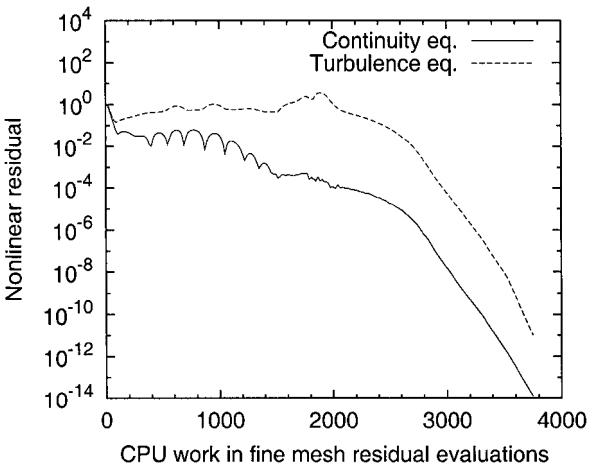
Table 1 Computational cost and performance of different preconditioning techniques (NACA 0012 airfoil)

	Distance-1				Distance-2	
Parameter	ILU(0)	ILU(1)	ILU(2)	ILU(5)	ILU(0)	ILU(1)
t_{eval}, s	4.42	4.42	4.42	4.42	8.29	8.29
t_{fac}, s	0.64	1.89	3.58	12.70	2.58	10.97
Size_{fac}, MB	11.9	17.3	25.9	52.5	27.9	47.7
$t_{\text{GMRES(50)}}$, s	49.62	51.82	55.09	61.15	55.22	58.17
t_{total}, s	611	483	387	436	——	482
N_{total} –	13	12	12	12	——	13
N_{FE} –[a]	552	394	271	187	——	209

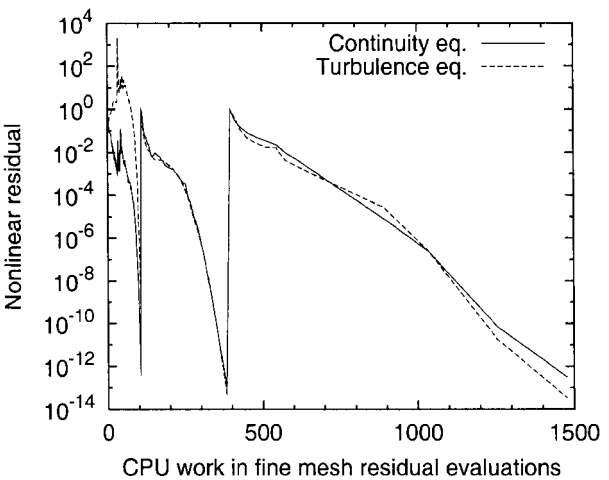
[a]Cost of one FE is 0.69 s [HP9000/780].



a) Fine grid (14,110 cells)



b) Nonlinear convergence (single grid)

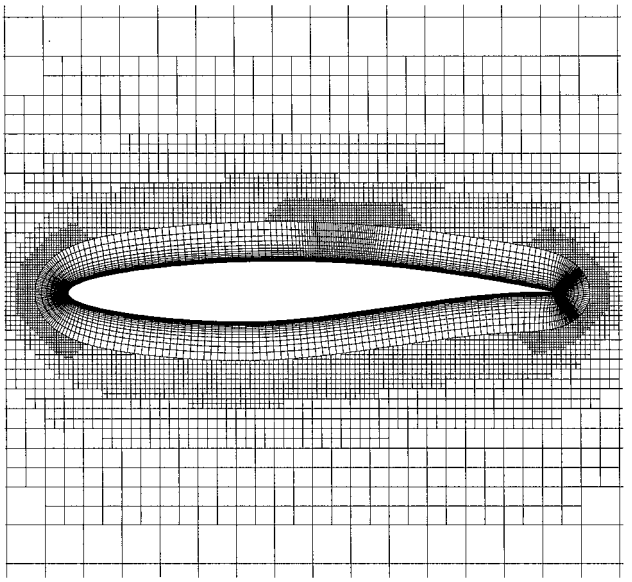


c) Nonlinear convergence (mesh sequencing)

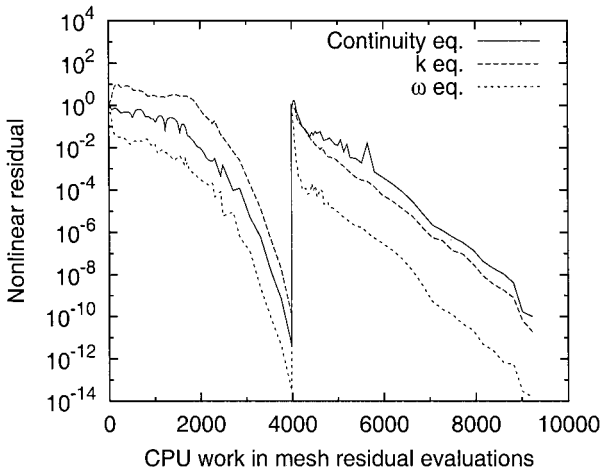
Fig. 1 NACA 0012 airfoil (S–A model[7]).

Turbulent Flow over an RAE 2822 Airfoil

For this numerical application, the computation of the turbulent transonic flow over an RAE 2822 airfoil is considered. The inlet Mach number equals 0.73, the Reynolds number based on the airfoil chord is 6.5×10^6, and the angle of incidence is equal to 2.79 deg. These flow conditions correspond to case 9 in the experimental



a) Adapted grid (18,010 cells)



b) Nonlinear convergence

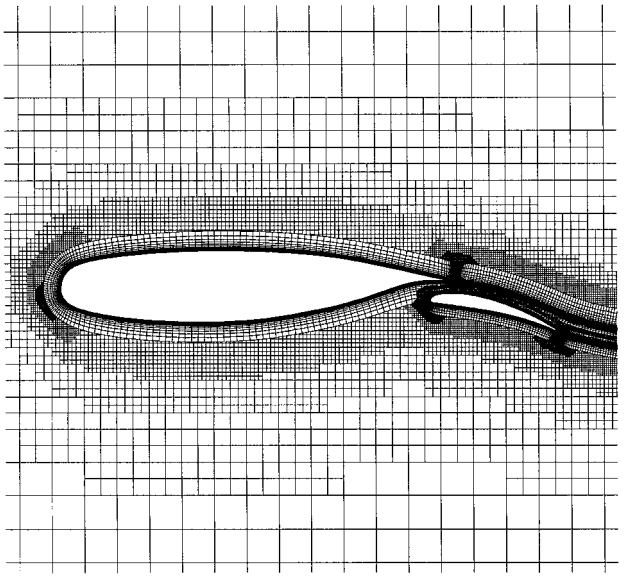
Fig. 2 RAE 2822 airfoil (k–ω model[8]).

report by Cook et al.[13] The initial hybrid mesh generated for this test case contains 15,336 cells. Because of the relatively poor streamwise grid resolution in the shock region, the shock is expected to be somewhat smeared. Therefore, a second mesh (18,010 cells), for which a point source is used to obtain a clustering of the grid lines in the shock region, is considered. This mesh is depicted in Fig 2a.

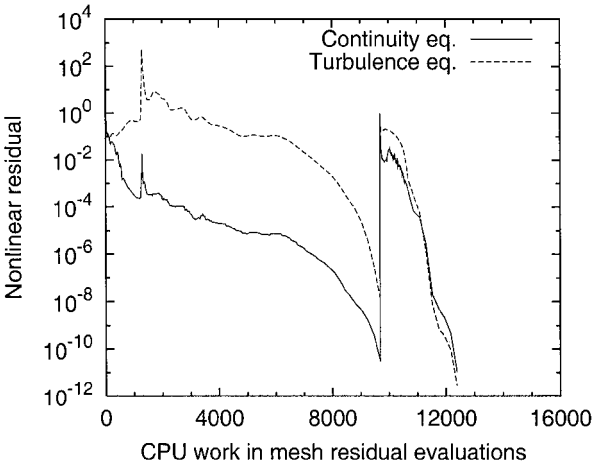
Figure 2b reports the nonlinear convergence histories for the k–ω model[8] on both grids. The computation on the unadapted grid is started from a solution obtained with a constant reconstruction scheme. The initial solution on the adapted grid is obtained by interpolation. The initial CFL number on both meshes is equal to 10. GMRES(50) is employed with a maximum of two restarts and an ILU(0) factorization of the distance-1 Jacobian matrix. As shown in Fig. 2b, a Newton-type convergence is achieved.

Turbulent Flow over a Two-Element Airfoil

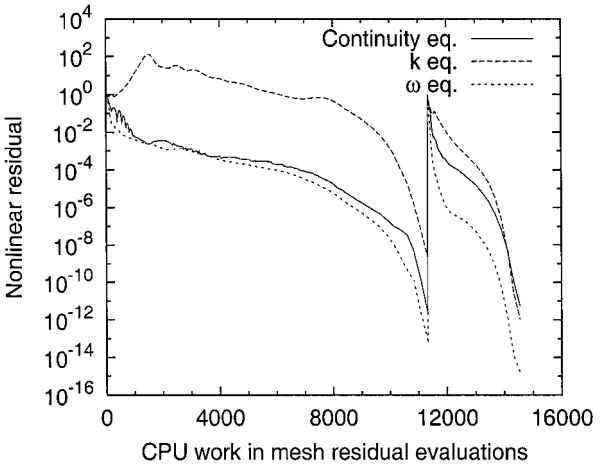
The last test case investigates the simulation of the turbulent flow over a complex geometry, namely, a modified NLR 7301 supercritical airfoil with a slotted trailing-edge flap. This configuration was studied experimentally by van den Berg.[14] The tests were performed at an inlet Mach number of 0.185, an angle of attack of 6 deg, and a Reynolds number based on the main airfoil chord of 2.51×10^6. The initial hybrid mesh generated for this computation has 19,482 cells. Streamlines from a solution computed on this grid are used to determine the wake emanating from the trailing edges of the main airfoil and the flap. The adapted mesh including the wake traces contains 25,661 cells and is depicted in Fig. 3a.



a) Adapted grid (25,661 cells)



b) Nonlinear convergence (S–A model[7])



c) Nonlinear convergence (k–ω model[8])

Fig. 3 NLR 7301 airfoil.

Figures 3b and 3c report the nonlinear convergence histories for the S–A model[7] and the k–ω model.[8] For both models a computation is first performed on the unadapted grid with an initial CFL number equal to 10. For the S–A model the computation is started from a uniform flowfield, whereas a solution obtained with a constant reconstruction scheme is used as an initial guess for the k–ω model. The solution is then interpolated on the adapted grid. The initial CFL number on this grid equals 100. For all of the computations, GMRES(50) is selected with a maximum of two restarts,

and an ILU(0) factorization of the distance-1 Jacobian matrix is used as a preconditioner for the linear system. As shown in Figs. 3b and 3c, a fast convergence is achieved on both grids for the two turbulence models, thus demonstrating the high efficiency of the Newton–Krylov approach.

Conclusions

An efficient pseudotransient Newton–Krylov algorithm has been described and applied to the solution of steady-state compressible turbulent flows around two-dimensional wing configurations. Computations of several turbulent flows have shown that 1) the strong coupling of the turbulence model with the averaged Navier–Stokes equations permits a Newton-type convergence for all the equations; 2) the use of a mesh sequencing strategy greatly reduces the number of nonlinear iterations and the computational time; 3) the choice of an ILU(0) factorization of the distance-1 Jacobian matrix as a preconditioner generally results in a good compromise between computational time and memory requirements; 4) if the memory requirements are not essential, the choice of a higher value of the level of fill-in can greatly increase the performance of the Krylov solver; and 5) the computational time required for one steady-state calculation is equal to the cost of a few thousand evaluations of the nonlinear residual. This makes the present solution strategy highly competitive with state-of-the-art multigrid techniques.

Acknowledgments

This research was supported by the Belgian Industry and Agriculture Research Training Fund while the author was Research Assistant at the Aerodynamics Group of the University of Liège, Belgium.

References

[1]McHugh, P. R., and Knoll, D. A., "Comparison of Standard and Matrix-Free Implementations of Several Newton–Krylov Solvers," *AIAA Journal*, Vol. 32, No. 12, 1994, pp. 2394–2400.

[2]Nielsen, E. J., Anderson, W. K., Walters, R. W., and Keyes, D. E., "Application of Newton–Krylov Methodology to a Three-Dimensional Unstructured Euler Code," AIAA Paper 95-1733, 1995.

[3]Pueyo, A., and Zingg, D. W.,"Efficient Newton–Krylov Solver for Aerodynamic Computations," *AIAA Journal*, Vol. 36, No. 11, 1998, pp 1991–1997.

[4]Barth, T. J., and Linton, S. W., "An Unstructured Mesh Newton Solver for Compressible Fluid Flow and its Parallel Implementation," AIAA Paper 95-0221, 1995.

[5]Geuzaine, P., Lepot, I., Meers, F., and Essers, J.-A., "Multilevel Newton–Krylov Algorithms for Computing Compressible Flows on Unstructured Meshes," AIAA Paper 99-3341, 1999.

[6]Geuzaine, P., "An Implicit Upwind Finite Volume Method for Compressible Turbulent Flows on Unstructured Meshes," Ph.D. Dissertation, University of Liège, Liège, Belgium, 1999.

[7]Spalart, P. R., and Allmaras, S. R., "One-Equation Turbulence Model for Aerodynamic Flows," AIAA Paper 92-0439, 1992.

[8]Wilcox, D. C., "Reassessment of the Scale-Determining Equation for Advanced Turbulence Models," *AIAA Journal*, Vol. 26, No. 11, 1988, pp. 1299–1310.

[9]Delanaye, M., Geuzaine, P., and Essers, J.-A., "Development and Application of Quadratic Reconstruction Schemes for Compressible Flows on Unstructured Adaptive Grids," AIAA Paper 97-2120, 1997.

[10]Geuzaine, P., Delanaye, M., and Essers, J.-A., "Computations of High Reynolds Number Flows with an Implicit Quadratic Reconstruction Scheme on Unstructured Grids," AIAA Paper 97-1947, 1997.

[11]Saad, Y., and Schultz, M. H., "GMRES: a Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Problems," *SIAM Journal on Scientific and Statistical Computing*, Vol. 7, No. 3, 1986, pp. 856–869.

[12]Thibert, J. J., Grandjacques, M., and Ohman, L. H., "NACA 0012," *Experimental Data Base for Computer Program Assessment*, AR-138, AGARD, 1979, Chap. 1.

[13]Cook, P. H., McDonald, M. A., and Firmin, M. C. P., "Aerofoil RAE 2822—Pressure Distribution, and Boundary Layer, and Wake Measurements," *Experimental Data Base for Computer Program Assessment*, AR-138, AGARD, 1979, Chap. 6.

[14]van den Berg, B., "Boundary Layer Measurements on a Two-Dimensional Wing with Flap," National Aerospace Lab., TR NLR TR-79009 U, The Netherlands, 1979.

J. Kallinderis
Associate Editor

Characteristics of a Plunging Airfoil at Zero Freestream Velocity

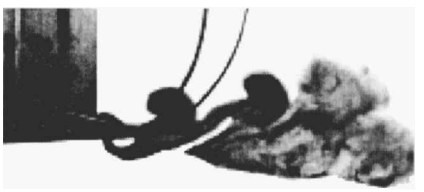
Joseph C. S. Lai*
University of New South Wales, Australian Defence Force Academy, Canberra, Australian Capital Territory 2600, Australia
and
Max F. Platzer†
Naval Postgraduate School, Monterey, California 93943-5106

Nomenclature

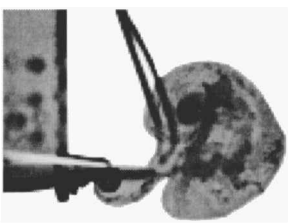
a	=	amplitude of oscillation, mm
c	=	chord of airfoil, mm
d	=	diameter of cylinder, mm
f	=	frequency of oscillation, Hz
h	=	nondimensional amplitude of oscillation, a/c
k	=	reduced frequency parameter, $2\pi fc/U_0$
kh	=	nondimensional plunge velocity, $2\pi fa/U_0$
U	=	mean streamwise velocity
$U_{\max}$	=	maximum streamwise velocity
U_0	=	freestream velocity
V_p	=	peak plunge velocity, $2\pi fa$, m/s
x	=	streamwise direction measured from the trailing edge
y	=	lateral direction measured from the centerline of the airfoil
$y_{\max}$	=	location where $U = U_{\max}$

I. Introduction

FLOWS around oscillating airfoils are relevant for the analysis of aircraft wing flutter, helicopter and turbomachine blade flutter, and for the prediction of aeroacoustic noise generation. The analysis of incompressible flow past oscillating airfoils was



a) f = 10 Hz, k = 31.4 (kh = 0.785), and U_0 = 0.2 m/s



b) f = 2.5 Hz, h = 0.025, and U_0 = 0 m/s

Fig. 1 Vortex patterns for a 100-mm NACA 0012 airfoil oscillated in plunge for h = 0.025 and various frequencies of oscillation.

Received 1 October 1999; revision received 6 November 2000; accepted for publication 7 November 2000.

*Associate Professor, School of Aerospace and Mechanical Engineering, University College. Senior Member AIAA.

†Professor, Department of Aeronautics and Astronautics; platzer@aa.nps.navy.mil. Fellow AIAA.