AIAA JOURNAL
Vol. 47, No. 4, April 2009

Implicit Harmonic Balance Solver for Transonic Flow with Forced Motions

M. A. Woodgate* and K. J. Badcock*
University of Liverpool, Liverpool, England L69 3BX, United Kingdom

DOI: 10.2514/1.36311

The computation of the aerodynamic forces arising from forced periodic motions is required for the generation of dynamic terms in models for flight simulation. The periodicity can be used to avoid using fully unsteady calculations by using the harmonic balance method. The current paper develops an implicit solver for the harmonic balance equations. The method is tested on two transonic test cases and evaluation is made against the unsteady simulation results. The first case is for the pitching NACA 0012 aerofoil. The second is for forced pitching of the F-5 wing with a wing tip launcher and missile. A reduction in computational time by one order of magnitude compared with the unsteady solver is obtained.

Nomenclature

A	=	matrix in frequency domain equation
c	=	chord
D	=	matrix in harmonic balance equation
E	=	transformation matrix between frequency and time domains
e	=	energy
$\mathbf{F}, \mathbf{G}, \mathbf{H}$	=	convective fluxes
$\mathbf{I}$	=	residual of semidiscrete system
I	=	identity matrix
k	=	reduced frequency
n_H	=	number of harmonics
p	=	pressure
$\mathbf{R}$	=	residual vector
T	=	period
t	=	time
u, v, w	=	Cartesian velocity components
$\mathbf{W}$	=	conserved variables
α	=	angle of attack
Δt	=	pseudo time step
Δt^*	=	real time step
ρ	=	density
ω	=	frequency

Subscripts

m	=	mean value
hb	=	harmonic balance
∞	=	freestream value
0	=	amplitude

Superscripts

n	=	time level
$\hat{}$	=	Fourier coefficient

Introduction

AERODYNAMIC models for flight dynamics and control analysis are normally written in a coefficient or look-up table form. In both cases raw data are required, which are used to relate the aircraft state with the forces and moments generated by the airflow. Dynamic terms can play a very significant role in the model, arising from angular rates of the motion. These terms have traditionally been generated using measurements from forced periodic motions in wind tunnels.

With advances in computational fluid dynamics (CFD) it seems likely that flight dynamics aerodynamic models will be increasingly generated using simulation. For dynamic terms this implies unsteady calculations. To populate a model with multiple parameters, a large amount of computation is implied. Murman [1] put forward the idea of exploiting periodicity to reduce the cost of computing dynamic derivatives. The nonlinear frequency domain method of McMullen et al. [2] solves for periodic flows. In this method the CFD unknowns are expanded in a truncated Fourier series. The latest approximation to the Fourier coefficients is used to reconstruct the corresponding solution in the time domain. The residual is then formed, and the Fourier transform of this residual is calculated. Finally, the transformed residual is used to update the Fourier coefficients of the solution.

An alternative is the harmonic balance of Hall et al. [3]. Here the nonlinear frequency domain equations are transformed back into the time domain where the solution is computed at a small number of points around the cycle. The harmonic balance equations are solved using multigrid by Gopinath and Jameson [4] for forced pitching of the NACA 64A010 and the LANN wing.

The current paper investigates the efficient solution of the harmonic balance equations using an implicit method. Previous work by the authors [5] has investigated the use of approximate Jacobians and Krylov-type sparse matrix solvers for steady and unsteady solutions of the Euler and RANS equations. These techniques are applied to the calculation of forced periodic solutions of the sort necessary for the development of dynamic contributions to aerodynamic models for flight dynamics. The paper continues with the formulation of the underlying CFD solver, and then the implicit harmonic balance solver is described. Results are presented for the NACA 0012 aerofoil and the F-5 wing in forced pitch. Finally, conclusions are drawn.

Time Domain Formulation

The three-dimensional Euler equations can be written in nondimensional conservative form as

$$\frac{\partial \mathbf{W}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial z} = 0 \tag{1}$$

Here, $\mathbf{W}$ is the vector of conserved variables $\mathbf{W} = [\rho, \rho u, \rho v, \rho w, e]^T$. The flux vectors $\mathbf{F}$, $\mathbf{G}$ and $\mathbf{H}$ are

Received 21 December 2007; revision received 13 September 2008; accepted for publication 2 December 2008.

*Computational Fluid Dynamics Laboratory, Flight Sciences and Technology, Department of Engineering.

$$\mathbf{F}=\begin{pmatrix}\rho u\\ \rho uu+p\\ \rho vu\\ \rho wu\\ u(\rho e+p)\end{pmatrix},\quad \mathbf{G}=\begin{pmatrix}\rho v\\ \rho uv\\ \rho vv+p\\ \rho wv\\ v(\rho e+p)\end{pmatrix},\quad \mathbf{H}=\begin{pmatrix}\rho w\\ \rho uw\\ \rho vw\\ \rho ww+p\\ w(\rho e+p)\end{pmatrix}$$

The main features of the CFD solver are described in [5]. A fully implicit steady solution of the Euler equations is obtained by advancing the solution forward in time by solving the discrete nonlinear system of equations

$$\frac{\mathbf{W}^{n+1}-\mathbf{W}^{n}}{\Delta t}=-\mathbf{R}(\mathbf{W}^{n+1}) \tag{2}$$

The term on the right hand side, called the residual, is the discretization of the convective terms, given here by Osher's approximate Riemann solver [6], MUSCL interpolation [7], and Van Albada's limiter. Equation (2) is a nonlinear system of algebraic equations, which is solved by an implicit method [5], the main features of which are an approximate linearization to reduce the size and condition number of the linear system, and the use of a preconditioned Krylov subspace method to calculate the updates.

The steady state solver is applied to unsteady problems within a pseudo-time-stepping iteration [8], which at each real time step is written as

$$\left[\left(\frac{1}{\Delta t}-\frac{3}{2\Delta t^{*}}\right)I+\frac{\partial \mathbf{R}}{\partial \mathbf{W}}\right]\Delta \mathbf{W}=-\left(\mathbf{R}(\mathbf{W})+\frac{3\mathbf{W}-4\mathbf{W}^{n}+\mathbf{W}^{n-1}}{2\Delta t^{*}}\right) \tag{3}$$

Periodicity can be used to approximate the initial solution for the pseudo-time-stepping at each real time step. At each iteration a file is written to the local disk with the converged solution at that real time step. On the next cycle this file is read to provide the initial solution for the pseudo-time-stepping, and on convergence to the next real time solution the original file is overwritten with the updated solution. As the solution approaches a periodic state the pseudo-time-stepping converges quickly because it starts from an excellent initial guess. In this way, results can be obtained from time marching in a very efficient manner.

Harmonic Balance Method

As an alternative to time marching, the harmonic balance method allows for a direct calculation of the periodic state. Write the semidiscrete form as a system of ordinary differential equations

$$\mathbf{I}(t)=\frac{\mathrm{d}\mathbf{W}(t)}{\mathrm{d}t}+\mathbf{R}(t)=0 \tag{4}$$

Consider the solution $\mathbf{W}$ and residual $\mathbf{R}$ to be periodic in time and a function of ω, giving

$$\mathbf{W}(t)=\hat{\mathbf{W}}_0+\sum_{n=1}^{\infty}(\hat{\mathbf{W}}_{a_n}\cos(\omega nt)+\hat{\mathbf{W}}_{b_n}\sin(\omega nt)) \tag{5}$$

$$\mathbf{R}(t)=\hat{\mathbf{R}}_0+\sum_{n=1}^{\infty}(\hat{\mathbf{R}}_{a_n}\cos(\omega nt)+\hat{\mathbf{R}}_{b_n}\sin(\omega nt)) \tag{6}$$

Next, the series is truncated to a specified number of harmonics N_H

$$\mathbf{W}(t)\approx\hat{\mathbf{W}}_0+\sum_{n=1}^{N_H}(\hat{\mathbf{W}}_{a_n}\cos(\omega nt)+\hat{\mathbf{W}}_{b_n}\sin(\omega nt)) \tag{7}$$

$$\mathbf{R}(t)\approx\hat{\mathbf{R}}_0+\sum_{n=1}^{N_H}(\hat{\mathbf{R}}_{a_n}\cos(\omega nt)+\hat{\mathbf{R}}_{b_n}\sin(\omega nt)) \tag{8}$$

and Eq. (4) can also be expressed in a Fourier series as

$$\mathbf{I}(t)\approx\hat{\mathbf{I}}_0+\sum_{n=1}^{N_H}(\hat{\mathbf{I}}_{a_n}\cos(\omega nt)+\hat{\mathbf{I}}_{b_n}\sin(\omega nt)) \tag{9}$$

A Fourier transform of Eq. (9) then yields

$$\hat{\mathbf{I}}_0=\frac{\omega}{2\pi}\int_0^{2\pi/\omega}\mathbf{I}(t)\mathrm{d}t=\hat{\mathbf{R}}_0 \tag{10}$$

$$\hat{\mathbf{I}}_{a_n}=\frac{\omega}{\pi}\int_0^{2\pi/\omega}\mathbf{I}(t)\cos(\omega nt)\mathrm{d}t=\omega n\hat{\mathbf{W}}_{b_n}+\hat{\mathbf{R}}_{a_n} \tag{11}$$

$$\hat{\mathbf{I}}_{b_n}=\frac{\omega}{\pi}\int_0^{2\pi/\omega}\mathbf{I}(t)\sin(\omega nt)\mathrm{d}t=-\omega n\hat{\mathbf{W}}_{a_n}+\hat{\mathbf{R}}_{b_n} \tag{12}$$

giving a system of equations for the Fourier series coefficients

$$\hat{\mathbf{R}}_0=0 \tag{13}$$

$$\omega n\hat{\mathbf{W}}_{b_n}+\hat{\mathbf{R}}_{a_n}=0 \tag{14}$$

$$-\omega n\hat{\mathbf{W}}_{a_n}+\hat{\mathbf{R}}_{b_n}=0 \tag{15}$$

This is a system of $N_T=2N_H+1$ equations in N_T unknown harmonic terms and can be expressed in matrix form as

$$\omega\mathbf{A}\hat{\mathbf{W}}+\hat{\mathbf{R}}=\mathbf{0} \tag{16}$$

where A is a $N_T\times N_T$ matrix containing the entries $A(n+1,N_H+n+1)=n$ and $A(N_H+n+1,n+1)=-n$ and

$$\hat{\mathbf{W}}=\begin{pmatrix}\hat{\mathbf{W}}_{a_0}\\ \hat{\mathbf{W}}_{a_1}\\ \vdots\\ \hat{\mathbf{W}}_{a_{N_H}}\\ \hat{\mathbf{W}}_{b_1}\\ \vdots\\ \hat{\mathbf{W}}_{b_{N_H}}\end{pmatrix}\qquad \hat{\mathbf{R}}=\begin{pmatrix}\hat{\mathbf{R}}_{a_0}\\ \hat{\mathbf{R}}_{a_1}\\ \vdots\\ \hat{\mathbf{R}}_{a_{N_H}}\\ \hat{\mathbf{R}}_{b_1}\\ \vdots\\ \hat{\mathbf{R}}_{b_{N_H}}\end{pmatrix} \tag{17}$$

The difficulty with solving Eq. (16) is in finding a relationship between $\hat{\mathbf{R}}$ and $\hat{\mathbf{W}}$. To avoid this problem the system is converted back to the time domain. The solution is split into N_T discrete equally spaced subintervals over the period $T=2\pi/\omega$

$$\mathbf{W}_{hb}=\begin{pmatrix}\mathbf{W}(t_0+\Delta t)\\ \mathbf{W}(t_0+2\Delta t)\\ \vdots\\ \mathbf{W}(t_0+T)\end{pmatrix}\qquad \mathbf{R}_{hb}=\begin{pmatrix}\mathbf{R}(t_0+\Delta t)\\ \mathbf{R}(t_0+2\Delta t)\\ \vdots\\ \mathbf{R}(t_0+T)\end{pmatrix} \tag{18}$$

where $\Delta t=2\pi/(N_T\omega)$. Then there is a transformation matrix [9] E such that

$$\hat{\mathbf{W}}=E\mathbf{W}_{\mathrm{hb}}\quad\text{and}\quad\hat{\mathbf{R}}=E\mathbf{R}_{\mathrm{hb}}$$

and then Eq. (16) becomes

$$\begin{aligned}\omega AE\mathbf{W}_{\mathrm{hb}}+E\mathbf{R}_{\mathrm{hb}}&=0=\omega E^{-1}AE\mathbf{W}_{\mathrm{hb}}+E^{-1}E\mathbf{R}_{\mathrm{hb}}\\ &=\omega D\mathbf{W}_{\mathrm{hb}}+\mathbf{R}_{\mathrm{hb}}\end{aligned} \tag{19}$$

where $D=E^{-1}AE$ and the components of D are defined by

$$D_{i,j} = \frac{2}{N_T} \sum_{k=1}^{N_H} k \sin(2\pi k(j-i)/N_T)$$

Note that the diagonal $D_{i,i}$ is zero.

We can then apply pseudo time marching to the harmonic balance equation

$$\frac{\mathrm{d}\mathbf{W}_{\mathrm{hb}}}{\mathrm{d}t} + \omega D \mathbf{W}_{\mathrm{hb}} + \mathbf{R}_{\mathrm{hb}} = 0$$

This equation is solved using an implicit method, of which one step is written as

$$\frac{\mathbf{W}_{\mathrm{hb}}^{n+1} - \mathbf{W}_{\mathrm{hb}}^{n}}{\Delta t} = -[\omega D \mathbf{W}_{\mathrm{hb}} + \mathbf{R}_{\mathrm{hb}}(\mathbf{W}^{n+1})] \tag{20}$$

The right-hand side is linearized as

$$\mathbf{R}_{\mathrm{hb}}(\mathbf{W}_{\mathrm{hb}}^{n+1}) \approx \mathbf{R}_{\mathrm{hb}}(\mathbf{W}_{\mathrm{hb}}^{n}) + J(\mathbf{W}_{\mathrm{hb}}^{n+1} - \mathbf{W}_{\mathrm{hb}}^{n})$$

The Jacobian matrix J is

$$J = \begin{bmatrix} \frac{\partial \mathbf{R}}{\partial \mathbf{W}}|_{t_0+\Delta t} & \omega D_{1,2} & \cdots & \omega D_{1,N_T} \\ \omega D_{2,1} & \frac{\partial \mathbf{R}}{\partial \mathbf{W}}|_{t_0+2\Delta t} & & \\ \vdots & & \ddots & \\ \omega D_{N_T,1} & \omega D_{N_T,2} & & \frac{\partial \mathbf{R}}{\partial \mathbf{W}}|_{t_0+T} \end{bmatrix} \tag{21}$$

where $\partial \mathbf{R}/\partial \mathbf{W}$ is the Jacobian matrix of the CFD residual, which appears in Eq. (3).

There are two considerations when solving Eq. (19). First, for solving CFD systems it is normally more efficient to use an approximate Jacobian matrix based on a lower-order spatial discretization of the residual function. This results in a linear system that has less terms in the coefficient matrix and is better conditioned. Second, a sparse matrix solver is used to calculate the updates from the solution of the linear system. The key issue is normally in the preconditioning used, and block incomplete lower upper (BILU) factorization has proved effective for systems arising from CFD systems [5].

For solving the harmonic balance system several experiments were made based on experience with solving for a CFD steady state. First, for the terms on the diagonal of J, arising from the CFD residual, an approximate Jacobian matrix arising from the first-order discretization of R is used. The linear system is solved using a Krylov subspace method with BILU factorization with no fill-in (i.e., the sparsity pattern of the factorization is the same as the coefficient matrix). Experiments using this approximate Jacobian are shown for the two test cases following. Further approximations were attempted by neglecting all off-diagonal terms. This has the advantage of decoupling the linear systems for each time step. However, the pseudo time stepping was found to diverge with this approximation.

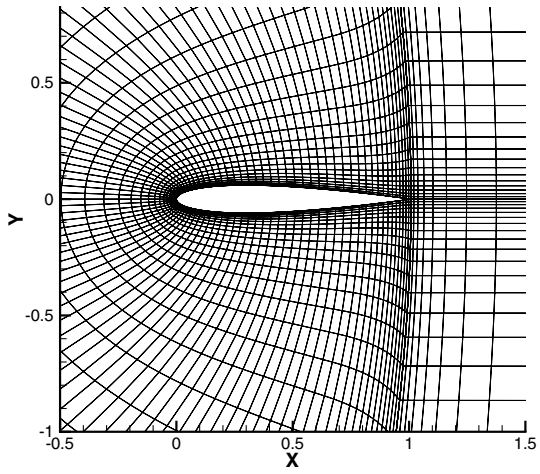


Fig. 1 View of the block structured grid used for test case 1.

Table 1 Memory requirements for test case 2 as the number of harmonics is increased, where M_{st} is the memory required for the steady state solver and M_{hb} is the memory required by the implicit harmonic balance solver

Modes	$M_{\mathrm{hb}}/M_{\mathrm{st}}$
0	1.0
1	3.85
2	7.86
3	13.0
4	19.3
8	55.9

Table 2 Summary of the linear solver convergence for test case 1 as the number of harmonic balance modes is increased

Modes	Linear Solver Residual Obtained	Linear Solver Iterations
1	0.0001	5–10
2	0.0001	5–10
3	0.0001	5–10
4	0.0001	10–20
5	0.0001	14–40
6	0.001–0.0001	40
7	0.3 – 0.1	40
8	0.9 – 0.5	40

The memory required to store the linear system for the 3-D case scales like

$$\frac{4N_H^2 + 16N_H + 7}{7} \tag{22}$$

The memory required for test case 2 (described next) is shown in Table 1 for an increasing number of modes. Here, the memory is expressed as a fraction of the memory required for the implicit steady state solver on the same grid. The requirements quickly grow and become significant beyond a small number of modes.

The performance of the linear solver for test case 1 (see following) is given in Table 2. As the number of harmonic modes is increased, the linear system becomes harder to solve. The performance is acceptable up to about 5 modes. Beyond this number the linear solver fails to reach the requested four orders of residual reduction at some pseudo time steps. One level of fill-in in the BILU factorization was tried for these cases without much improvement. There are a few

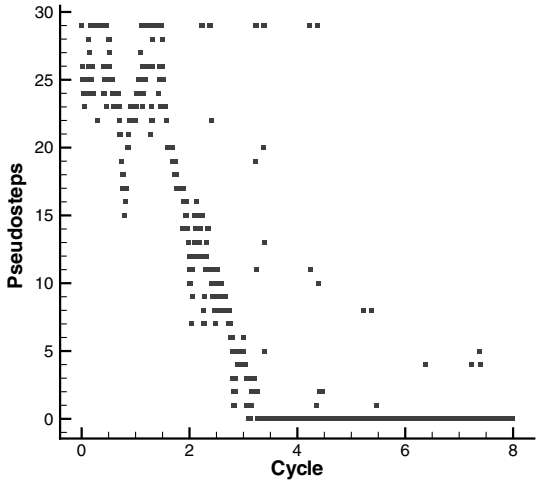


Fig. 2 Number of pseudo time steps at each real time step for test case 1, with 128 real time steps per cycle.

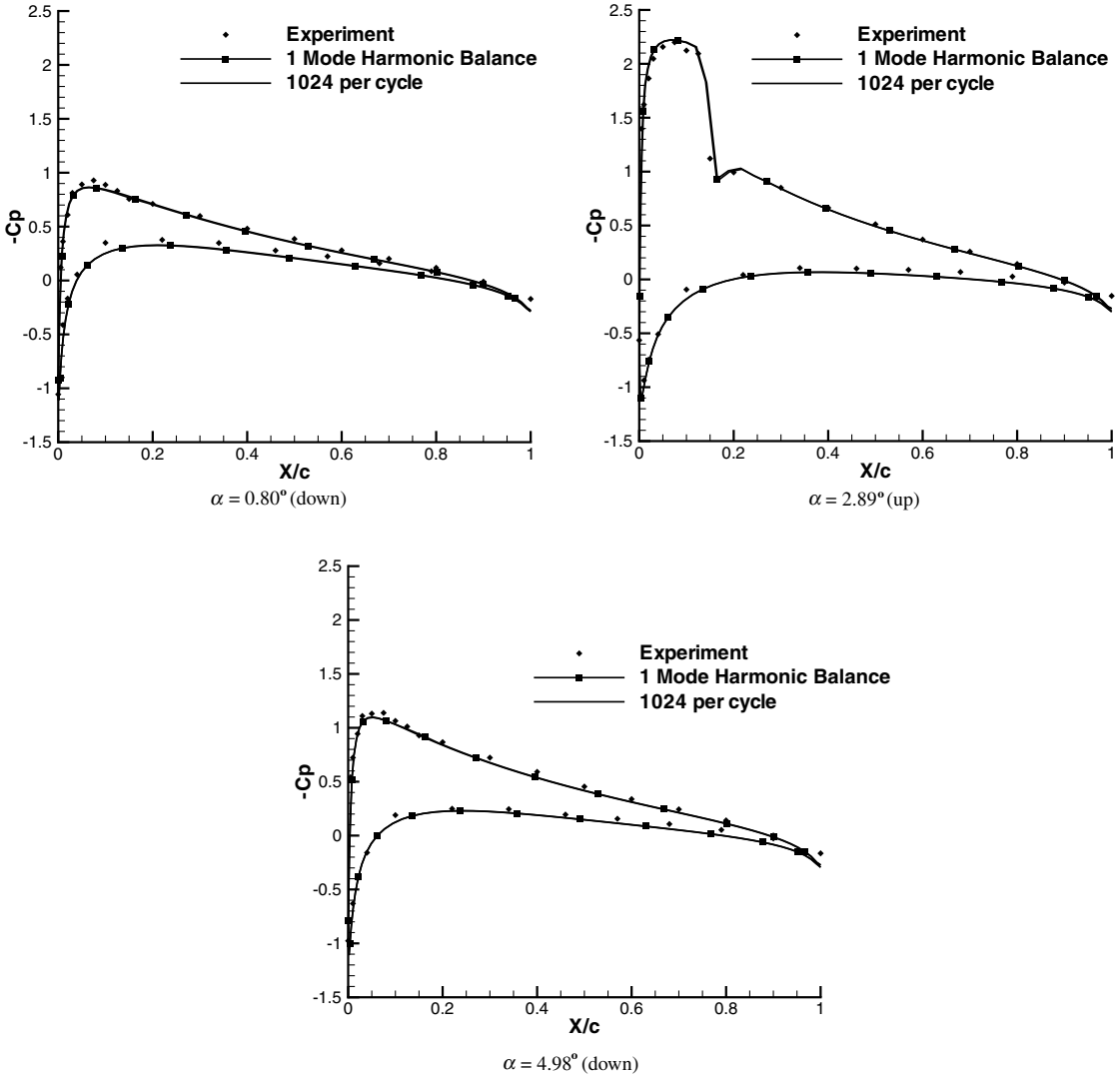


Fig. 3 Instantaneous pressure coefficient distribution for test case 1 at three points in the cycle. The terms up and down in parenthesis indicate the direction increasing or decreasing angle, respectively.

possible approaches to improving the performance of the linear solver. The first is to reorder the unknowns so that the $N_H + 1$ time slices for each cell follow on from each other. This would move the off-diagonal D blocks closer to the main diagonal and, hence, reduce the bandwidth of the system, which usually helps to improve the BILU factorization. The other option is to relax the quasi-Newton method by adding a time like term $1/\Delta t$ to the diagonal of the harmonic system.

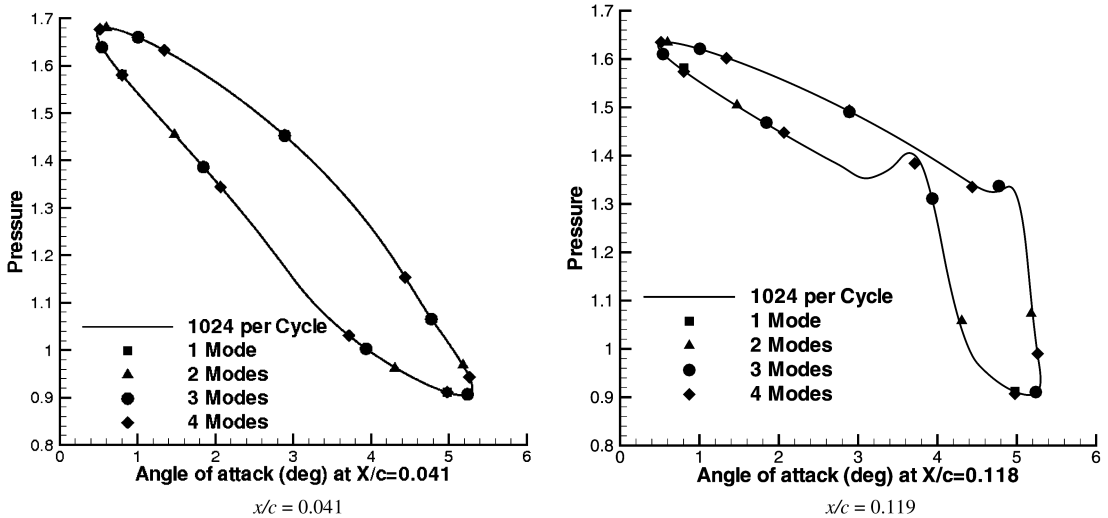


Fig. 4 Pressure coefficient for test case 1 as a function of α at two points in the aerofoil that remain subsonic (left) throughout the cycle and features transonic flow (right) at some points in the cycle.

Table 3 Run times for test case 1 with different numbers of steps per cycle for the time domain solver

Steps per cycle	CPU time, s
16	64
32	117
64	218
128	390
256	683
512	1205
1024	2120

Table 4 Run times for test case 1 with different numbers of modes per cycle for the harmonic balance solver

Modes per cycle	CPU time, s
1	15
2	25
3	42
4	75

In summary, the method finally adopted for the results presented next is to approximate the Jacobian matrix J by using the Jacobian of the first-order CFD spatial residual on its diagonal, and to use BILU with no fill-in for the linear solver preconditioner.

Results

Test Case 1: Unsteady NACA 0012 CT1

The first test case is for a pitching aerofoil selected from the AGARD database [10,11]. A periodic motion of the aerofoil is defined by the angle of attack as a function of time such that

$$\alpha(t) = \alpha_m + \alpha_0 \sin(\omega t)$$

where ω is related to the reduced frequency k by

$$k = \frac{\omega c}{2U_\infty}$$

The case has a freestream Mach number of 0.6, a mean incidence $\alpha_m = 2.89$ deg, $\alpha_0 = 2.41$ deg, $k = 0.0808$, and the pitching is about the quarter chord.

A three-block grid was generated, which contains 4096 points and is shown in Fig. 1. Note that all time domain calculations had been previously repeated on a finer grid that has twice the number of points in each direction, with virtually identical results obtained. The steady state solver starts with 10 explicit steps with a CFL (Courant–Friedrichs–Lewy) number of 0.4 followed by an implicit solution with a CFL number of 500 until the L2 norm of the residual is dropped eight orders of magnitude from the starting value.

The unsteady solver uses the same tolerance and limits the number of pseudosteps to 30. Again, a CFL number of 500 is used. The periodicity is exploited to provide a good initial guess from the previous cycle as described in the section on the time domain formulation. The number of pseudosteps at each real time step is

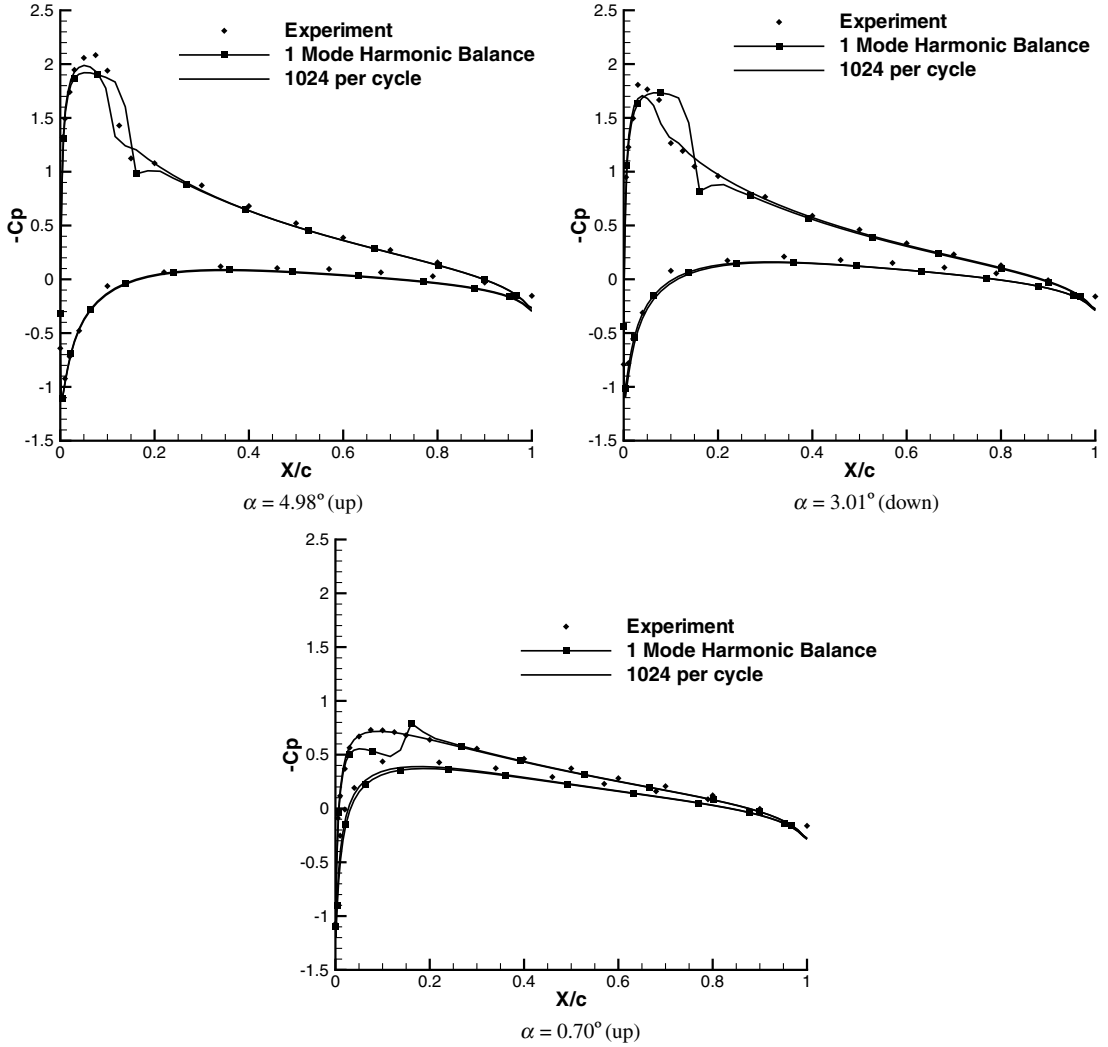


Fig. 5 Reconstructed instantaneous pressure coefficient distribution for test case 1 at three points in the cycle. The terms up and down in parenthesis indicate the direction increasing or decreasing angle, respectively.

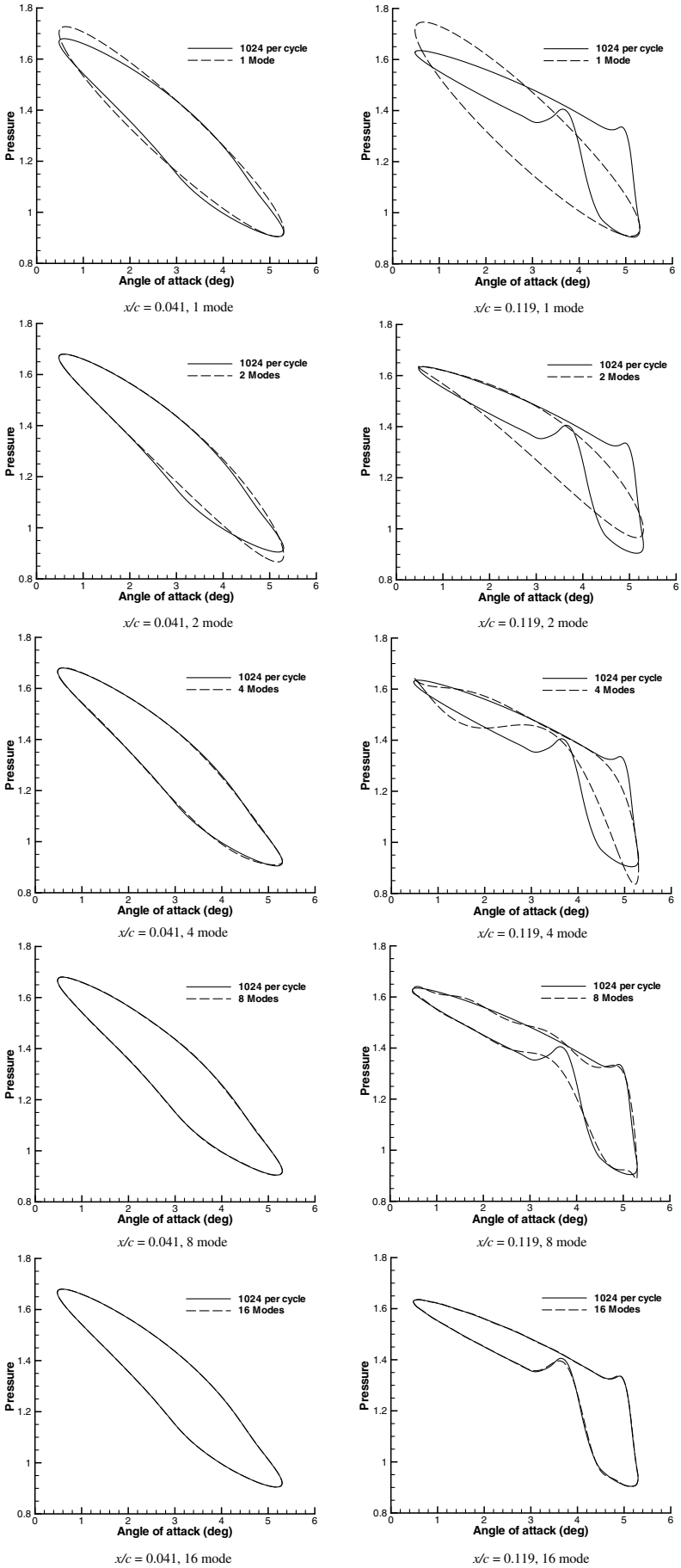


Fig. 6 Reconstructed pressure coefficient for test case 1 as a function of α at two points in the aerofoil that remain subsonic (left) throughout the cycle and features transonic flow (right) at some points in the cycle.

shown in Fig. 2 when 128 real time steps are used per cycle. By the third cycle many of the real time steps require no further convergence from the previous cycle. This significantly reduces the cost of the overall calculation.

The harmonic balance solver first solves $2N_H + 2$ steady-state calculations with 10 explicit and 10 implicit iterations to obtain an initial solution. This is followed by a solution of the full harmonic system.

Referring to Table 2, increased work associated with the linear solver as the number of modes is increased contributes to increased cost. The number of iterations to reduce the residual of the harmonic balance equations by eight orders of magnitude is between 25 and 30 for all the cases shown.

A shock appears and disappears during the pitching cycle. A very small time step, corresponding to 1024 steps per forced pitching cycle, was used with the time domain solver to generate a reference solution. The pressure coefficient distributions at three points in the cycle, which are included in the harmonic balance time intervals, are shown in Fig. 3. The appearance and disappearance of the shock toward the leading edge on the upper surface is clear. The harmonic balance solution using 1 mode is also included in the figure, and there is a close correspondence with the time domain reference solution at these time steps.

Next, the behavior of the pressure coefficient throughout a pitching cycle at two locations on the aerofoil was examined. The first position ($x/c = 0.041$) was always ahead of the shock, which appears on the upper surface, whereas the second location ($x/c = 0.119$) has the shock moving through. The pressure coefficient at these two locations as a function of α is shown in Fig. 4 with the time domain reference solution and the harmonic balance solution using 1–4 modes. There is an excellent agreement between the two sets of solutions, even when the shock significantly distorts the curve. Note that the harmonic balance predictions are marked with points at the time intervals used in the calculation only, and no reconstruction of the solution at other times is shown in this figure.

These results show that the harmonic balance method is able to reproduce, at the time intervals used in the calculation, the reference time domain solution with only 1 mode. A study was undertaken to assess the efficiency of the harmonic balance solver in comparison to the time domain solver. The time domain solver produces results very close to the reference solution with 32 steps per cycle. The cost of the time domain solutions with increasing numbers of time steps per cycle is shown in Table 3. These results were obtained using restarting from the corresponding solution on the previous cycle and represent the fastest results that could be obtained with the time domain solver for the current test case. The time to convergence of the harmonic balance solver using 1–4 modes is shown in Table 4. The time required for a good solution using the unsteady solver is 117 s, corresponding to 32 steps per cycle. The harmonic balance method gives a similar solution with 1 mode, and in 12.8% of the CPU time. For reference, the time for a steady-state solution at the mean incidence is 3 s for this case.

The results shown until now have all been for the harmonic balance solution at the time intervals used in the calculation. However, the solution at other time intervals is also required. The Fourier coefficients can be calculated from the harmonic balance time interval solutions, and then used to reconstruct the solution at any point around the motion cycle. Reconstructed pressure distributions are shown in Fig. 5. The 1-mode reconstruction is good at all points on the aerofoil except where the shock appears and moves. In this region the disagreement with the time domain solution is very large. To explain this behavior the reconstructed solution is examined at the two points on the aerofoil chosen previously. The reconstructed values of the pressure coefficients around the cycle are shown in Fig. 6. It is clear that, particularly for the second point that is traversed by the shock, the response has very significant higher harmonic content. The reconstructed 8-mode solution for this point shows significant discrepancy with the time domain solution, and the 16-mode solution has still not captured some of the details. To quantify this, the root mean squared difference between the time domain and the reconstructured harmonic balance solutions was

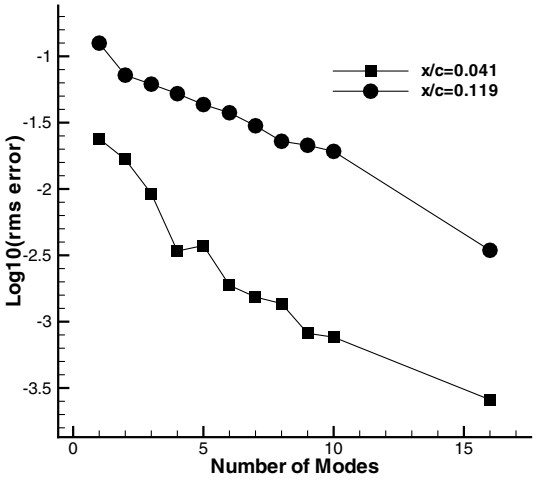


Fig. 7 Root mean square of the difference between the 1024 steps per cycle time domain solution and the reconstructed harmonic balance solutions for test case 1 at two points on the aerofoil.

calculated and is shown in Fig. 7. This figure quantifies the improvement in the harmonic balance solution as the number of modes is increased. Consistent with the plots shown in Fig. 6, the difference only decreases very slowly, particularly for the difficult second point. Comparing the 1- and 8-mode solutions at this point the difference with the time domain solution has dropped by less than one order of magnitude.

Test Case 2: F-5 Wing with Tip Launcher

The second test case is for a pitching wing selected from the NATO Research and Technology Organization's report [11] and relates to a transonic wind tunnel test from 1977 on an oscillating model of the outer part of the Northrop F-5 wing with external store. Run 355 has a freestream Mach number of 0.896, a mean incidence $\alpha_m = 0.004\,\text{deg}$, $\alpha_0 = 0.117\,\text{deg}$, $k = 0.275$, and the pitching is about the unswept line through the root half chord location. The grid generated for this configuration has 290 blocks and 168,000 cells. A view of the surface grid is shown in Fig. 8.

For the steady calculations 50 explicit steps at a CFL number of 0.4, followed by 15 implicit steps at a CFL number of 500 are used. These parameters are also used to provide the initial guess for the harmonic balance solver.

Forced motion calculations were run using 16, 32, and 64 points per cycle. The surface pressure coefficient is shown in Fig. 9 when the wing is pitching up through 0.004 deg. The shock around midchord is clear in the figure. The solution from the harmonic balance method using 1 mode is also shown in the figure, and the two results look very close. The pressure distribution along two streamwise cuts is shown in detail in Fig. 10, and comparison is made with available measurements [11]. The predictions are inviscid and

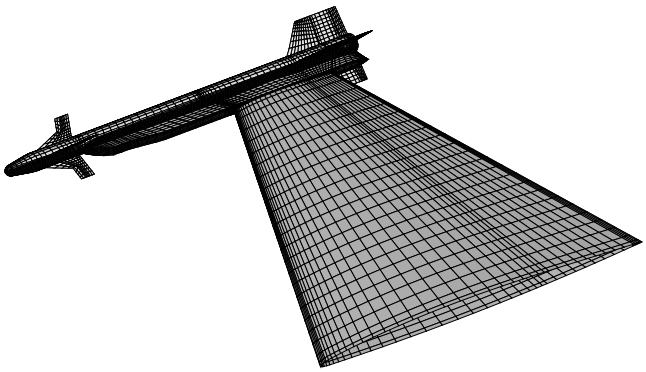
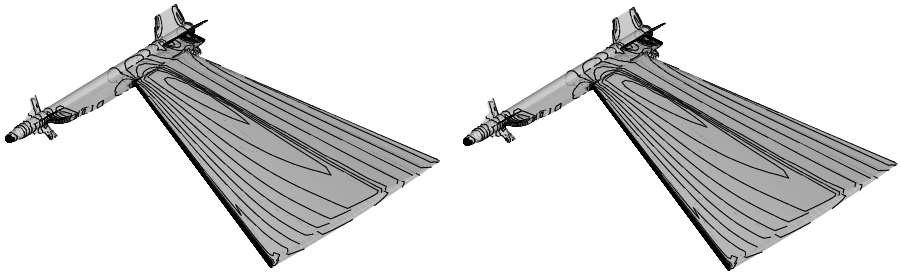
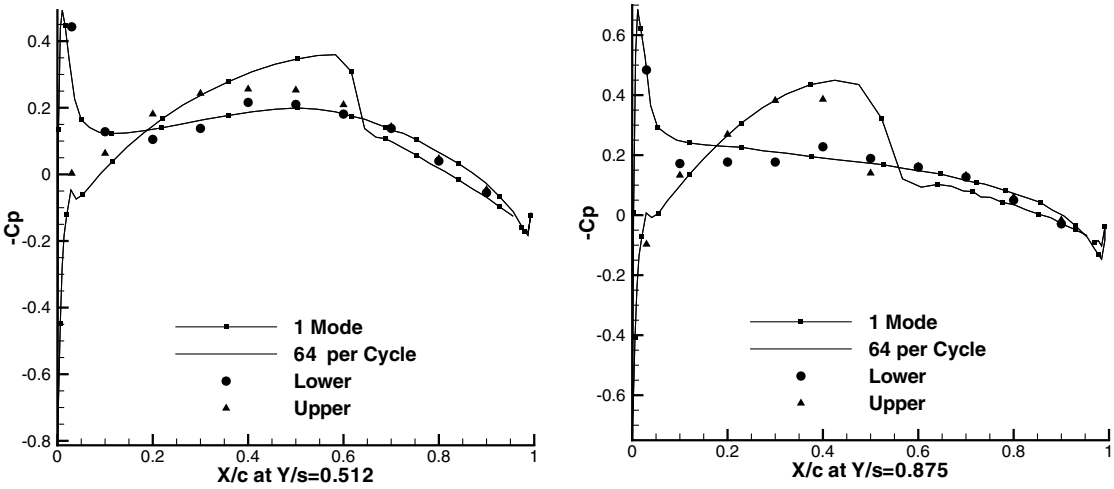


Fig. 8 Surface grid for the F-5 wing with wing, tip, and launcher used in test case 2.



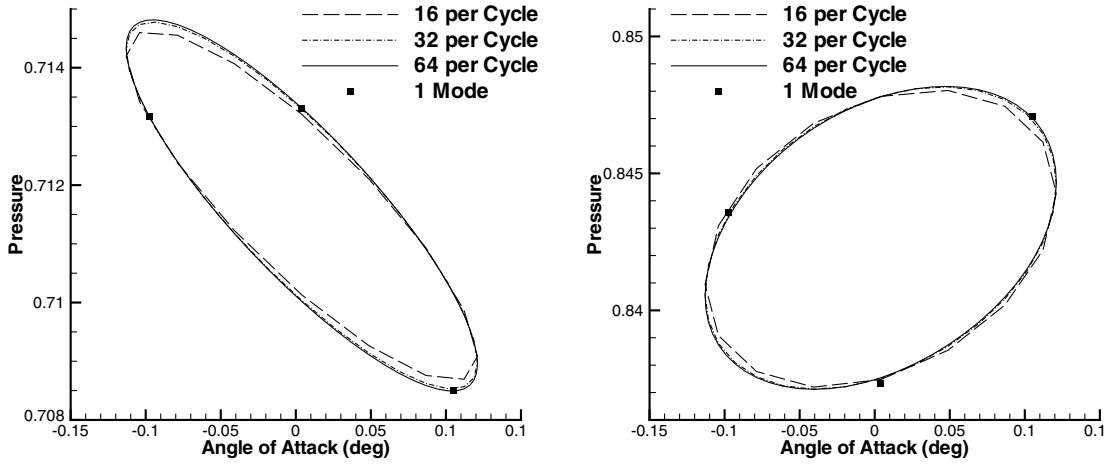
Time Domain Harmonic Balance

Fig. 9 Upper surface pressure distribution for test case 2 at $\alpha = 0.004$ deg with α increasing.



51% space 87.5% span

Fig. 10 Pressure distribution along two spanwise slices for test case 2 at $\alpha = 0.004$ deg with α increasing.



x/c = 0.35 x/c = 0.65

Fig. 11 Upper surface pressure coefficient values at two points taken at 87.5% span for test case 2.

show expected discrepancies with the measurements. The shock in the computed results persists too far inboard and is located too far downstream. However, the computed results are qualitatively similar to the measurements, and even quantitatively the comparison is reasonable. This makes the case a realistic test for the harmonic balance solver, which is the topic of this paper.

Again, the time evolution of the pressure coefficient at two points is shown in detail. The points chosen are at 87.5 % span and at 35 and 65 % chord. The evolution is shown as a function of incidence in Fig. 11. For this case only the 1-mode harmonic balance solution could be computed due to memory restrictions. This calculation required 3.3 GB of memory. The time domain calculation required 64 steps per cycle to be time accurate, and this solution agrees well with the 1-mode harmonic balance solution. The harmonic balance solution took 39 min to convergence, compared with 160, 246, and 391 min for 16, 32, and 64 steps per cycle, respectively. As in test case 1, for a comparable accuracy, the Harmonic Balance method is at least 10 times faster.

In this case the shock motion does not introduce as large a distortion from a sinusoidal time response at points 1 and 2, when compared with test case 1. The reconstructions for the time response around the cycle are shown in Fig. 12, and good agreement is obtained between the time domain and harmonic balance 1-mode predictions throughout the cycle.

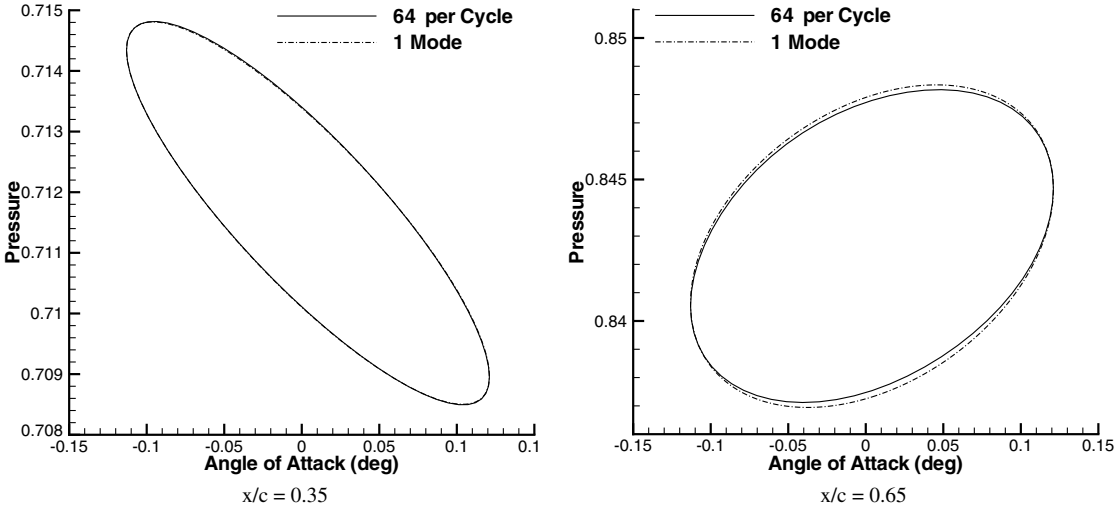


Fig. 12 Reconstructed upper surface pressure coefficient values at two points taken at 87.5% span for test case 2.

Conclusions

An implicit solver for the harmonic balance equations has been presented. The method uses approximate Jacobian matrices for the diagonal blocks arising from the linearization of the CFD residual. Experiments showed that off-diagonal terms could not be dropped from the harmonic balance Jacobian, with significant implications for the memory requirements of the method. The linear system is solved using a Krylov subspace method with BILU factorization. The performance of the linear solver was seen to fall off with increasing number of modes.

The method proved to be effective for the test cases considered, with a reduction of CPU time by at least a factor of 10 in each case when compared with the unsteady solver. A restarting method, which uses the corresponding solution from the previous cycle, is used to obtain an efficient unsteady solver for comparison. In both cases the time step converged unsteady solution was effectively reproduced, at the time intervals used in the harmonic balance calculation, by retaining 1 mode in the harmonic balance solver. However, significantly more modes were needed to reconstruct the nonlinear behavior at other points in the cycle for test case 1. If the detailed behavior is required around the cycle, then the cost benefits of the harmonic balance solver when compared with the time domain solver may be lost. A full evaluation of this point is required and will be application dependent.

Future work involves the development of a parallel version of the solver and the application of the solver to compute dynamic derivatives.

Acknowledgments

This work is supported by the Engineering and Physical Sciences Research Council and the Ministry of Defence under EP/D504473/1. The reviewers are thanked for their helpful suggestions, which particularly contributed to consideration of the reconstructed solutions.

References

[1] Murman, S. M., "Reduced-Frequency Approach for Calculating Dynamic Derivatives," *AIAA Journal*, Vol. 45, No. 6, June, 2007, pp.1161–1168. doi:10.2514/1.15758

[2] McMullen, M., Jameson, A., and Alonso, J., "Demonstration of Nonlinear Frequency Domain Methods," *AIAA Journal*, Vol. 44, No. 7, July, 2006, pp. 1428–1435. doi:10.2514/1.15127

[3] Hall, K., Thomas, J., and Clark, W., "Computation of Unsteady Nonlinear Flows in Cascades Using a Harmonic Balance Technique," *AIAA Journal*, Vol. 40, No. 5, May, 2002, pp. 879–886. doi:10.2514/2.1754

[4] Gopinath, A. K., and Jameson, A., "Time Spectral Method for Periodic Unsteady Computations Over Two- and Three- Dimensional Bodies," *43rd AIAA Aerospace Sciences Meeting and Exhibit*, AIAA Paper 2005-1220, Jan. 2005.

[5] Badcock, K. J., Richards, B. E., and Woodgate, M. A., "Elements of Computational Fluid Dynamics on Block Structured Grids Using Implicit Solvers," *Progress in Aerospace Sciences*, Vol. 36, Nos. 5–6, 2000, pp. 351–392. doi:10.1016/S0376-0421(00)00005-1

[6] Osher, S., and Chakravarthy, S., "Upwind Schemes and Boundary Conditions with Applications to Euler Equations in General Geometries," *Journal of Computational Physics*, Vol. 50, No. 3, 1983, pp 447–481. doi:10.1016/0021-9991(83)90106-7

[7] Van Leer, B., "Towards the Ultimate Conservative Difference Scheme II: Monotonicity and Conservation Combined in a Second Order Scheme," *Journal of Computational Physics*, Vol. 14, No. 4, 1974, pp. 361–374. doi:10.1016/0021-9991(74)90019-9

[8] Jameson, A., "Time Dependent Calculations Using Multigrid, with Applications to Unsteady Flows Past Airfoils and Wings," AIAA Paper 91-1596, 1991.

[9] Thomas, J. P., Dowell, E. H., Hall, K. C., and Denegri, C. M., "Further Investigation of Modeling Limit Cycle Oscillation Behaviour of the F-16 Fighter Using a Harmonic Balance Approach," *46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials (SDM) Conference*, AIAA Paper 2005-1917, April, 2005.

[10] "Compendium of Unsteady Aerodynamic Measurements," AGARD Rept. 702, 1982.

[11] "Verification and Validation Data for Computational Unsteady Aerodynamics," Research and Technology Organization, RTO-TR-26, France, Oct. 2000.

P. Beran
Associate Editor