Numerical Prediction of Flow Instabilities and Aeroelastic Effects

KERRY HOURIGAN[1], IVAN MCBEAN[1], MARK THOMPSON[1] and FENG LIU[2]

[1] Fluids Laboratory for Aeronautical and Industrial Research (FLAIR), Department of Mechanical Engineering, Monash University, Clayton 3800, Australia
[2] Department of Mechanical and Aerospace Engineering, University of California, Irvine, California 92697-3975, USA

Summary

Two lines of investigation being undertaken at Monash University and of relevance to the aeronautics industry will be presented. First, the prediction of vortex breakdown, a fluid structure that bedevils high angle of attack aircraft but may assist the breakup of tip vortices, has been undertaken using an in-house parallelized spectral element code. Second, aeroelasticity has been identified as one of the most important problems presently facing the designers of turbomachinery blades as well as aircraft control surfaces. A 3-dimensional solver implementing the compressible Navier-Stokes equations coupled with the k-ω turbulence model has been developed to solve the unsteady flow through oscillating annular or linear turbine cascades. Simulations have been performed in parallel in a time accurate manner using a Jameson-type dual time Runge-Kutta scheme. This is then used to investigate the effects of 3-dimensionality and secondary flows on the unsteady aerodynamics and flutter characteristics of turbine cascades.

1 Introduction

The control of the transition from laminar to turbulent flow is of crucial importance in various flows, including those found in swirling flows and the flows around bluff bodies. Swirling flow is a basic ingredient of many important industrial and aerospace flows. Modern high-performance aircraft generate considerable aerodynamic lift forces at high angles of attack. However, the resulting highly swirling vortex structures that result are prone to rapid dilatation called Vortex Breakdown, leading to restriction of the angle of attacks and the operational flight envelope. Similarly, turbomachinery can produce flows of sufficiently high swirl in which vortex breakdown and loss of efficiency are observed. On the other hand, vortex breakdown can be utilised as flame holders in combustion chambers and to break u p the tip vortices of large aircraft. In the case of bluff body flows, the transition to turbulence in the wake leads to significant drag changes on the body and the character of the wake that interferes with downstream bodies. Significant improvements in

processes in the aerospace, mineral processing, wind engineering and manufacturing industries can result from the control of the transitions of flow in high swirl and wakes.

As designers in the turbomachinery industry strive to design machines that are lighter, more powerful and more efficient, blade flutter has become one of the most important limiting factors on the design process. The aeroelastic response is a complex phenomenon that is not well modeled or predicted by current design techniques. Codes that implement 2-dimensional models can simulate this behaviour in a meridional plane; however the flow structures found in blade passages are generally 3-dimensional and such models provide a qualitative rather than a quantitative analysis. Furthermore, important flow phenomena are not modeled including hub and casing vortices and tip effects.

A structured 3-dimensional Navier-Stokes code is developed to solve the unsteady governing equations. These are solved using an explicit Runge-Kutta scheme, implementing residual averaging and multigrid. The problem is then solved in a time accurate manner through a fully implicit scheme as proposed by Jameson [10]. This scheme has already been used in a 2-dimensional model of aeroelasticity in turbomachinery [19, 12]. The development of the present code is an extension of the previous two-dimensional method to three dimensions. Similar algorithms have been successfully implemented in a 3-dimensional Navier-Stokes external solver that models flow over a flexible wing [24, 14].

In this paper, the numerical methods being employed to study the transition to turbulence in swirling flows are first described. Then, the numerical methods used to study the aeroelastic response of aerofoils in turbulent flow are reported.

2 Time-Dependent Simulations of Vortex Breakdown

Two- and three-dimensional time-dependent spectral-element simulations of the swirling flow in a torsionally driven cylinder were undertaken. The aim was not to impose unphysical symmetries on the solution fields; in particular, there was no enforcement of either non-axisymmetry or time-independence.

Numerical Method

The time-dependent simulations employed a spectral/spectral-element method for axisymmetric geometries. A spectral-element discretisation was used in the r–z planes and, for the three-dimensional simulations, Galerkin-Fourier expansion in the θ direction. Within each spectral-element the solution variables ($\mathbf{u}$, p) are represented by a tensor-product of Lagrangian interpolants. The internal node points are chosen to correspond to the Gauss-Legendre-Lobatto quadrature points. This leads to considerable efficiency gains when evaluating the integrals from the application

of the Galerkin weighted residual method used to obtain the discretised equations. The order of the interpolation can be selected during run-time. This is a significant advantage since the spatial convergence of the simulations can be easily validated. When run in this mode, for *smooth* problems the method achieves spatial exponential (or spectral) convergence rates usually associated with global spectral methods.

The spatially-discrete equations are then independently discretised in time using a classical three-step splitting scheme described in [13]. This involves splitting each time-step into substeps accounting separately for convection, pressure gradient/continuity and diffusion. The convection substep employs the third-order Adams-Bashforth extrapolation and the diffusion substep uses the second-order Crank-Nicolson method (with the θ correction to improve high-frequency stability (see [4]). Overall second-order accuracy is achieved by using the higher-order pressure boundary condition described in [13].

The implementation has been validated on numerous problems and the expected convergence behaviour confirmed. For example, it has been used to predict the two-shedding modes occurring in a circular cylinder wake [22], the flow transitions in the wake of a sphere [23], and the feedback stability mechanism governing lock-on states for the flow past long rectangular plates ([8, 21]). More details of the implementation can be found in these papers and references therein.

A combination of temporal and spatial resolution studies were undertaken for the three-dimensional simulations described in the following section.

Parallelisation Issues

Three-dimensional simulations are inherently expensive and put considerable strain on computational resources. Fortunately, the spectral/spectral-element method described here can be parallelised efficiently allowing accurate simulations to be performed on clusters using the MPI (message passing interface) routines.

The application of the time-splitting scheme to the Navier-Stokes equations results in the following set of equations

$$\frac{\hat{\mathbf{u}} - \mathbf{u}^n}{\Delta t} = \mathrm{N}(\mathbf{u}), \tag{1}$$

$$\frac{\hat{\hat{\mathbf{u}}} - \hat{\mathbf{u}}}{\Delta t} = -\nabla P/\rho, \tag{2}$$

$$\frac{\mathbf{u}^{n+1} - \hat{\hat{\mathbf{u}}}}{\Delta t} = \mathrm{L}(\mathbf{u}). \tag{3}$$

Here, N(**u**) and L(**u**) represent the nonlinear (convection) and linear (diffusion) terms in the Navier-Stokes equations, P is the pressure, ρ the fluid density, $\mathbf{u} = (u_z, u_r, u_\theta)$ is the velocity vector and, at the completion of the three steps, the time

has been advanced from time t^n to t^{n+1}. Further details are given in ([13])

For a Cartesian coordinate system, if a (Galerkin) Fourier expansion is used to describe the dependence of a variable on the azimuthal coordinate, the second and third substeps decouple into a set of equations for each Fourier plane. This means these substeps can be done efficiently in Fourier space. In addition, the equations for each Fourier mode (consisting of two planes) can be solved on a separate processor and the associated memory requirements shared between the processors. The introduction of axisymmetric coordinates introduces some complications. In this case, the equations for u_r and u_θ from the diffusion step are coupled; however, decoupling can be achieved by introducing the complex variables $u_1 = u_r + iu_\theta$ and $u_2 = u_r - iu_\theta$. Once this is substitution is performed, the pressure and diffusion substeps can still be solved in Fourier space with each Fourier mode evaluated on a different processor.

The convection substep in usually performed in physical space (because it is nonlinear) using an explicit scheme such as Adams-Bashforth. The transformation to physical space means that significant interprocess communication is required. Fortunately, optimised MPI routines exist to make this process reasonably efficient. The physical space domain is split into a number of approximately equal parts corresponding to the number of processors used. The different Fourier mode components are collected on each of these parts and a fast Fourier transform used to convert to physical space velocity components. The velocity components are then updated. To take account of the convection substep of the time-splitting algorithm, a Fourier transform is applied to transform back to Fourier space and the Fourier components are again distributed amongst the processors so that each processor once again has one (or more) Fourier mode(s). Overall, the interprocess communication between the processors is considerable, and hence the method requires high-speed intercommunication. Tests on a Compaq Alpha cluster built from ES40 processor units with a low-latency high-bandwidth Quadrics interconnect indicate typical parallel efficiencies of 80% or greater are achieved on 32 nodes. In addition, memory requirements are shared between the processors and hence order-of-magnitude larger simulations are possible using multi-processor machines. The code has been validated for axisymmetric and non-axisymmetric problems (*e.g.*, [23].

Three-Dimensional Simulation Results

Two cases are examined in this section. The first is vortex breakdown in a confined circular cylinder where rotation is induced by a spinning lid. The second case is vortex breakdown in a swirling pipe flow. Both cases lead to many different flow behaviours depending on the governing parameters.

Case A: Spinning-Lid Rig Three-dimensional simulations of the confined flow in a circular cylinder at $Re \geq 1850$ were undertaken to establish the final asymptotic

three-dimensional time-dependent state. This study was motivated by a similar numerical investigation of the flow using a non-axisymmetric mesh which showed that the flow evolved to an asymptotic state with four-fold rotational symmetry ([20]). Other studies based on stability analyses, (e.g., [6]) have suggested that the flow should remain axisymmetric to much higher Reynolds numbers.

The mesh used was the same as for a two-dimensional resolution study extended to three dimensions using a Fourier expansion in the azimuthal direction. Typically, 32 Fourier planes were used. Simulation undertaken evolved the flow from an initially-stationary state in a confined cylinder. In order to accelerate the development of any three-dimensionality, a white noise perturbation of amplitude 10^{-3} was added to each velocity component at each node after the first few time-steps. The flow field was subsequently evolved for a time corresponding to approximately 100 lid revolutions. All Fourier components (except the zeroth mode) decayed quickly leaving the flow in the same axisymmetric state as calculated with the axisymmetric code. In addition, an independent stability analysis indicated the stability of the flow to non-axisymmetric perturbations verifying the results of ([6]).

Case B: Swirling Flow in a Pipe A three-dimensional simulation of the swirling flow in a pipe is shown in Figure 1. In this case $Re = 1280$ with the pipe geometry and inlet velocity profile based on the experiments of Faler and Lebovich ([5]).

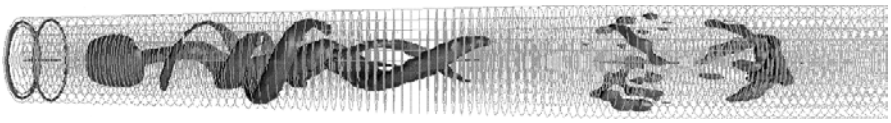

Figure 1 Transition state of swirling flow in a pipe showing vortex breakdown bubble and downstream spiral structure. Vortical flow structures are highlighted using the method of Jeong and Hussain. ([11])

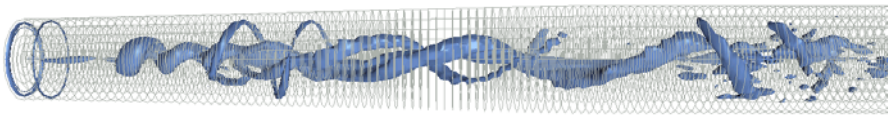

Figure 2 Final spiral breakdown state.

In the first part of the figure, the transition stage is shown with a vortex breakdown bubble apparent and spirals developing downstream. In the second figure, the bubble

has disappeared and has been replaced by a spiral breakdown. Note that the isosurface structures near the pipe entrance are due to the developing boundary layer and initial adjustment of the flow. An interesting feature is the outer spirals close to the pipe outer wall. These may be a transient feature indicating a convective instability although they appear both during the initial development and after the flow has evolved to its asymptotic state.

3 Fluid Model for the Aeroelastic Problem

The present three-dimensional multiblock and parallel code has been developed from a proven steady solver designed to model turbomachinery cascade flow [15, 18, 16]. The governing equations for the unsteady fluid problem in a Eulerian reference frame with a moving mesh.

$$\frac{\partial}{\partial t}\iint_{\Omega} \mathbf{w}\, d\Omega + \oint \mathbf{f}\, dS_x + \mathbf{g}\, dS_y + \mathbf{h}\, dS_z = 0 \tag{4}$$

where

$$\mathbf{w} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix} \tag{5}$$

$$\mathbf{f} = \begin{pmatrix} \rho\bar{u} \\ \rho u\bar{u} + p \\ \rho v\bar{u} \\ \rho w\bar{u} \\ \rho E\bar{u} + pu \end{pmatrix}, \quad \mathbf{g} = \begin{pmatrix} \rho\bar{v} \\ \rho u\bar{v} \\ \rho v\bar{v} + p \\ \rho w\bar{v} \\ \rho E\bar{v} + pv \end{pmatrix}, \quad \mathbf{h} = \begin{pmatrix} \rho\bar{w} \\ \rho u\bar{w} \\ \rho v\bar{w} \\ \rho w\bar{w} + p \\ \rho E\bar{w} + pw \end{pmatrix} \tag{6}$$

The time dependent and semi-discrete form of the governing equations may be written as

$$\frac{d\mathbf{w}}{dt} + R(\mathbf{w}) = 0 \tag{7}$$

A dual time stepping scheme [10] is used to calculate the unsteady flow problem. A second order accurate, fully implicit scheme is used to integrate Equation (7) to evolve the unsteady problem in a time accurate manner.

The discrete form of (7) is

$$\frac{3w^{n+1} - 4w^n + w^{n-1}}{2\Delta t} + R(w^{n+1}) = 0 \tag{8}$$

This equation may be recast into

$$\frac{d\mathbf{w}}{dt^*} + R^*(\mathbf{w}) = 0 \tag{9}$$

where

$$R^*(\mathbf{w}) = \frac{3w}{2\Delta t} + R(w) - \frac{2}{\Delta t}w^n + \frac{1}{2\Delta t}w^{n-1} \tag{10}$$

The steady state solution w in equation (9) is then equivalent to the time accurate solution w^{n+1} of equation (8). Any efficient algorithm may be used to obtain the steady-state solution to (9). In this paper, the above mentioned Runge-Kutta type scheme with multigrid is used. Minimum modification of the steady solver if required to make it time accurate in the above manner.

4 Multiblock and Parallel Implementation

A method using multiple blocks of structured grids is used to maximize the use of computational resources and to allow the generation of grids for complex geometries. While each block consists of a structured grid, the blocks can be connected to each other in an unstructured manner provided the mesh geometry is matched at the block interfaces.

It was decided early in the development of the code to make use of MPI and some of the high level programming constructs available in Fortran90. A number of different objects were created in the code data structure to facilitate the parallel calculation of the fluid problem. Each fluid block is treated as a single object or entity. A schematic of the multiblock data structure is shown in Figure 3. A processor may be allocated more than one fine grid block and each fine grid block will have associated a number of coarser, multiblock grids. The machine calculates for each multigrid level simultaneously, then copies the solution or interpolates the residual to the next multigrid level.

A subface is defined as another object. This is used for the interface between the present block and another block, or a region to which a single boundary condition is to be applied. These objects are cycled through each time the boundary of the blocks are to be updated, upon which communication is effected or a boundary condition is applied. So that separate boundary conditions are not required for each coordinate direction, subface objects are transformed into a single coordinate system. This increases code complexity marginally. However it maintains that the code that is relatively compact.

The structured cell numbering within each block is unimportant as transformations are used to reorient the face so that the numbering matches with the neighboring face. Consider the example of a C-grid where a single block is wrapped around the blade. At the interface downstream of the trailing edge, the cell numbering will be different on the upper and lower blocks, thus requiring reorientation of the 2-dimensional arrays.

In keeping with the use of high levels of Fortran90 code, the communication module also exploits some of the more sophisticated MPI routines. The use of MPI derived

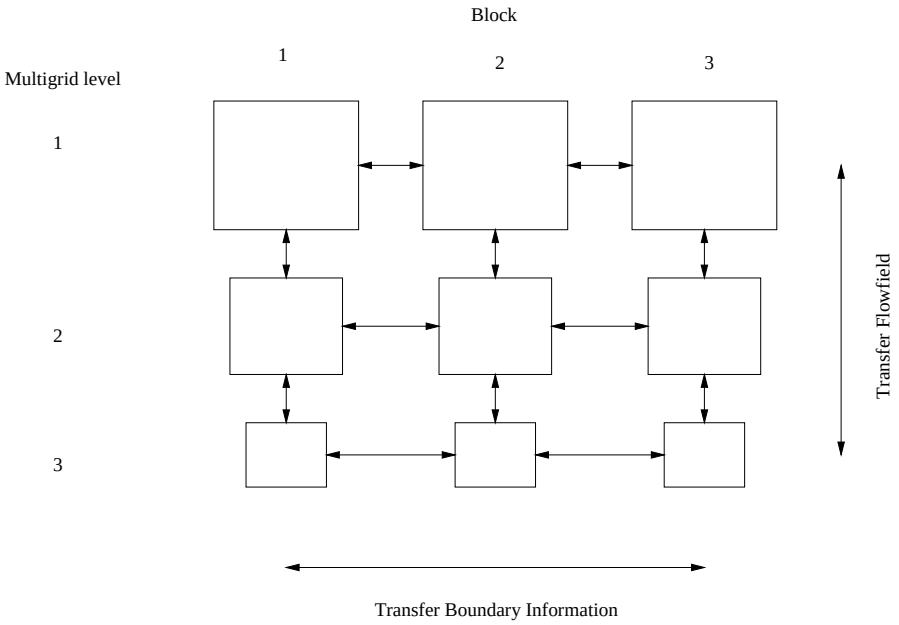

Figure 3 Schematic of multigrid and multiblock communication.

types allows the direct access of memory for the transfer of data, reducing the number of copies required during the communication of ghost cells. Due to the repetitive nature of the CFD computation, the “pipe-lining” of message passing calls is also implemented.

5 The Moving Grid

The movement of the fluid boundary requires the fluid grid to be regenerated over the entire flow domain. Thus in the multiple block code, given that the grid for each structured block is regenerated independently, the position of the corner points of each block must be somehow defined. This is effected by using a spring network analogy as proposed by Batina [2] to maintain grid regularity, which is particularly important where Navier-Stoke calculations are performed.

The network is formulated by connecting each block corner with hypothetical springs and corner positions are determined by a solution of the static equations. This simple and efficient calculation is performed on a single processor. Initially an unstructured grid network is constructed on the root processor. This contains nodal locations for each block corner and the connecting node information. New nodal positions are determined for free nodes through a predictor-corrector scheme. These are distributed to the respective processors, where transfinite interpolation (TFI) is performed to

interpolate the local grid at the previous time step to the new position. The details of this method may be found in Wong [24].

6 Model Validation and Results

Forced Airfoil Oscillation

To demonstrate the validity of the moving mesh, multiblock and unsteady implementation, the NACA64A010 case is presented. Computational results are compared in Figure 4 for different configurations and with experiment. An unsteady Euler calculation is performed in the flow solver. In the first case, a single block O-grid is used in combination with TFI to deform the grid to the oscillating airfoil. The far field boundary remains rigid. The second case involves a mesh that is not deformed, but rotates rigidly with the displacement of the airfoil surface. For the third calculation, the same grid as used for the single block cases is divided into 32 equal blocks, with 4 blocks in the radial direction and 8 in the circumferential direction. In this case the block corners were located using the spring analogy. The results for inviscid flow compare similarly with results presented elsewhere [1, 17] and there is little difference between the results for the different configurations.

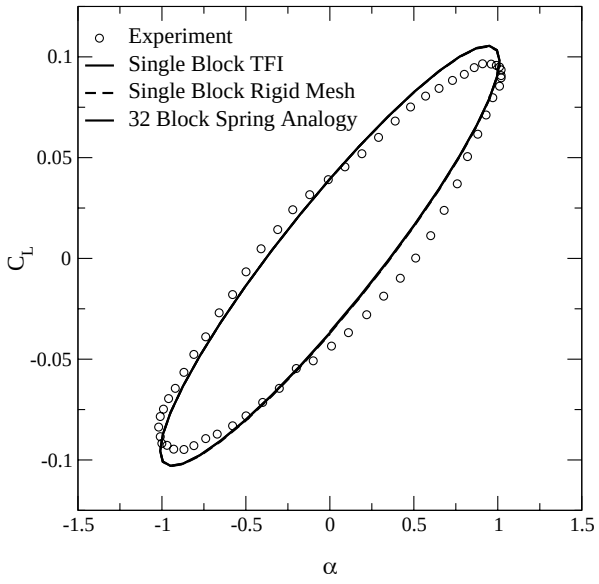

Figure 4 Comparison of unsteady NACA64A010 results with experiment.

7 Coupled Airfoil Model

With the integration of the structural solver, it was necessary to validate the implementation of the coupled model. The Isogai wing model [9] is a simple case that exhibits unsteady fluid-structure interaction and has proved useful in testing numerical models. It has been used previously by other researchers where the mesh moved in a rigid fashion [1]. The structural parameters for the case were chosen to simulate the vibrational characteristics of a swept back wing that are often used in military fighter aircraft.

The model is shown in Figure 5. Note the springs attached to both the plunging and pitching axes and the axis of rotation is actually well forward of the airfoil leading edge. Initially the airfoil is forced to oscillate for one period in pitch and then released. Once released the structural equations are applied to ascertain the new airfoil location at every time step. The aerodynamics force the response of the structural system. Inner iterations are used whereby the structural equations are updated within each real fluid time step. The initial oscillation was necessary to perturb the model from rest, as some disturbance is required to move the model from a stable configuration.
The important parameters for this simulation are the free stream Mach number Ma and the flutter velocity V_f. The flutter velocity is defined as

$$V_f = \frac{U_\infty}{b\omega_\alpha\sqrt{\mu}}. \tag{11}$$

This is used to determine the effect of freestream velocity U_∞ on the stability and involves the ratio of fluid momentum to structural inertial terms,

$$\mu = \frac{m}{\pi\rho b^2},$$

the airfoil chord b, the fluid density ρ and the structural natural frequency ω_α.
Predicted results for the flutter boundary are shown in Figure 6 and compare well with those of Alonso [1]. The flutter boundary is defined as where the amplitude of oscillation neither increases nor decreases with time. Simulations were carried

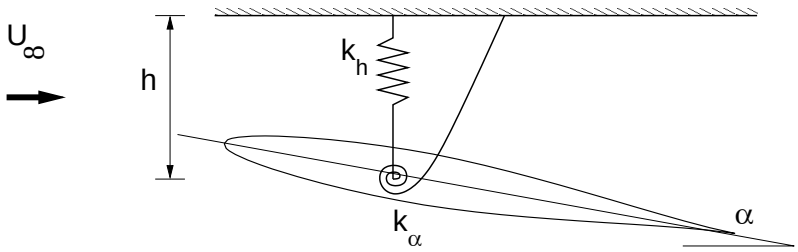

Figure 5 Isogai wing model.

out at fixed far field Mach number while the flutter velocity was varied. Each point required approximately five simulations to locate the flutter boundary. The line of best fit indicates the flutter boundary for the model.

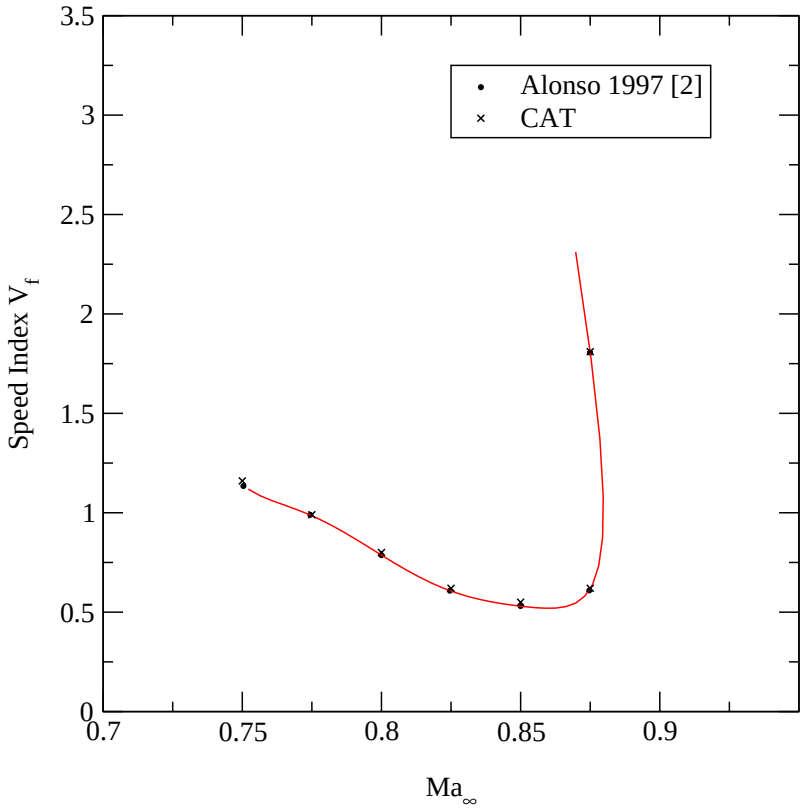

Figure 6 Flutter boundary for Isogai wing model

8 Turbomachinery Cascade

Few 3-dimensional experimental measurements exist in the field of unsteady aerodynamics in turbomachinery. A large effort has been made to compile measurements for oscillating cascades that are typical of those found in industry through the Workshop on Aeroelasticity in Turbomachines [3], however measurements in these cases are made at mid span and blade motion is symmetric in the radial plane to minimise the 3-dimensional effects on results.

The Standard Test Case 4 is described as a highly loaded turbine rotor, involving typical sections of modern free standing turbine blades [3]. The flow is high subsonic and the blade normally exhibits flutter in the first bending mode. To simulate the unsteady flow, the blade is forced to oscillate in this bending mode by translation at an angle to the axial axis in the radial plane. Viscous effects in two-dimensional models have been simulated numerically for this case by other authors using the Navier-Stokes equations with the algebraic Baldwin Lomax Model [7] and the k-ω turbulence model [12].

Recently, complete conditions at the cascade inlet and outlet plane have been made available for Standard Test Case 4. This allows the authors to validate the 3- dimensional implementation and investigate the difference between a number of different cascade models. Blade stability will be calculated by way of the energy method.

Inviscid and Navier-Stokes simulations were performed for both 2-dimensional and 3-dimensional configurations of Standard Test Case 4. Within Test Case 4 there are a number of measurements; results here are compared with Test 627. In this case, the passage flow was in the high subsonic regime with a reduced frequency of $k_c = 0.1187$ and involving a bending amplitude of $b_c = 3.8 \times 10^{-3}$. At the inlet $Ma_{in} = 0.18$ and at the outlet $Ma_{out} = 0.9$. It was found that using the present method approximately 4 oscillations were required for a converged unsteady solution as described in previous work [12].

The unsteady results were investigated along the span of the blade, as shown in Figure 7. The effects of the passage vortices on the unsteady pressure coefficient appear minimal and the coefficient on the suction side collapses onto each other towards the trailing edge. However at 10 percent span for the inviscid result, the suction side pressure deviates from the viscous results. Unsteady pressure coefficients in this figure are referenced to the static pressures at the mid-span of the passage inlet.

9 Concluding Remarks

Two lines of research, involving different numerical techniques for parallelised processing, are looking at the problem of transition to turbulence and of aeroelastic response. The spectral element method allows high order prediction of flow instabilities; in particular, it has been applied to the problem of highly swirling flows and vortex breakdown. A novel multiblock and parallel, integrated structural and fluid solver has been used for the investigation of aeroelastic response in compressible flows. The implementation is general and is not limited to particular geometries and thus is flexible in that it may be applied to a broad range of problems. The moving mesh and structural model allow for the coupled solution of aeroelastic problems. A number of different cases have been presented that compare computed results with experiment or other numerical results.

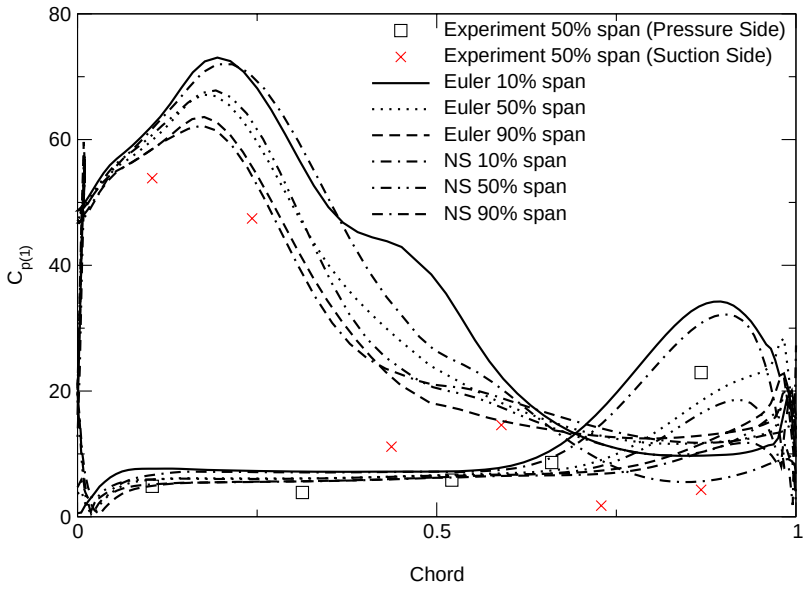

Figure 7 Comparison of unsteady pressure coefficient at different span-wise locations for 3D model for IBPA=180.

Acknowledgments

Ivan McBean acknowledges the support of an Australian Postgraduate Award. Computing support from VPAC and APAC are gratefully acknowledged. Michael Jones is thanked for running the spectral element simulations.

References

[1] Alonso, J. & Jameson, A. (1994) Fully-implicit time-marching aeroelastic solutions. In *AIAA 32nd Aerospace Sciences Meeting*. AIAA, January. 94-0056.

[2] Batina, J. T. (1990) Unsteady Euler airfoil solutions using unstructured dynamic meshes. *AIAA Journal*, **28**(8), 1381–1388.

[3] Bölcs, A. & Fransson, T. H. (1986) *Aeroelasticity in turbomachines: Comparison of theoretical and experimental cascade results*. Lausanne: EPFL.

[4] Canuto, C., Hussaini, M.Y., Quarteroni, A. & Zang, T. (1987) *Spectral methods in fluid dynamics*. Springer-Verlag.

[5] Faler, J. H. & Leibovich, S. (1978) An experimental map of the internal structure of a vortex breakdown. *J. Fluid Mech.* **86**, 313-335.

[6] Gelfgat, A.Yu., Bar-Yoseph, P. Z. & Solan, A. (2001) Three-dimensional instability of axisymmetric flow in a rotating lid-cylinder enclosure *Journal of Fluid Mechanics*, **438**, 363-377.

[7] Grueber, B. & Carstens, V. (1998) Computation of the unsteady transonic flow in harmonically oscillating turbine cascades taking into account viscous effects. *Journal of Turbomachinery*, **120**, 104-111.

[8] Hourigan, K, Thompson, M. C. & Tan, B. T. (2001) Self-Sustained Oscillations in Flows around Long Flat Plates. *J. Fluids and Structures*, **15**, 387-398.

[9] Isogai, K. (1979) On the transonic-dip mechanism of flutter of a sweptback wing. *AIAA Journal*, **17**(7), 793–795.

[10] Jameson, A. (1991) Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. In *AIAA 10th Computational Fluid Dynamics Conference*. AIAA, June.

[11] Jeong, J. & Hussain, F. (1995) On the identification of a vortex. *J. Fluid Mech.*, **285**, 69-94.

[12] Ji, S. & Liu, F. (1999) Flutter computation of turbomachinery cascades using a parallel unsteady Navier-Stokes code. *AIAA Journal*, **37**(3), 320–327.

[13] Karniadakis, G. E., Israeli, M. & Orszag, S. A. (1991) High-Order splitting methods for incompressible Navier-Stokes equations. *J. Comp. Phys.*, **97**, 441.

[14] Liu, F., Cai,J., Zhu, Y., Tsai, H. M. & Wong, A. S. F. (2000) Calculation of wing flutter by a coupled CFD-CSD method. In *AIAA 38th Aerospace Sciences Meeting & Exhibit*. AIAA, January. AIAA 2000-0907.

[15] Liu, F. (1991) *Numerical Calculation of Turbomachinery Cascade Flows*. PhD thesis, Princeton University, May.

[16] Liu, F., Ian K. Jennions, I. K. & Jameson, A. (1998) Computation of turbomachinery flow by a convective-upwind-split- pressure (CUSP) scheme. In *36th Aerospace Sciences Meeting and Exhibit*. AIAA, January.

[17] Liu, F. & Ji, S. (1996) Unsteady flow calculations with a multigrid navier-stokes method. *AIAA Journal*, **34**(10),2047–2053.

[18] Liu, F. & Zheng, X. (1994) Staggered finite volume scheme for solving cascade flow with a k-ω turbulence model. *AIAA Journal*, **32**(8),1589–1597.

[19] Sadeghi, M. and Liu, F. (2000) Computation of mistuning effects on cascade flutter. In *AIAA 38th Aerospace Sciences Meeting & Exhibit*. AIAA, January. AIAA 2000-0230.

[20] Sotiropoulos, F. & Ventikos, Y. (2001) The three-dimensional structure of confined swirling flows with vortex breakdown. *J. Fluid Mech.*, **426**, 155-175.

[21] Tan, B. T., Thompson, M.C. & Hourigan, K. (1998) Flow Around Long Rectangular Plates under Cross Flow Perturbations. *Int. J. of Fluid Dynamics (WWW)*, **2**, Article 1.

[22] Thompson, M. C., Hourigan, K. & Sheridan, J. (1996) Three-Dimensional Instabilities in the Wake of a Circular Cylinder. *International Journal of Experimental Heat Transfer, Thermodynamics, and Fluid Mechanics*, **12**, 190-196.

[23] Thompson, M. C., Leweke, Th. & Provansal, M. (2001) Kinematics and dynamics of sphere wake transition. *Journal of Fluids and Structures*, **15**, 575-586.

[24] Wong,A. S. F., Tsai, H. M., Cai, J., Zhu, Y. & Liu, F. (2000) Unsteady flow calculations with a multi-block moving mesh algorithm. In *38th Aerospace Sciences Meeting and Exhibit*. AIAA, January. AIAA-2000-1002.