14th Australasian Fluid Mechanics Conference
Adelaide University, Adelaide, Australia
10-14 December 2001

Navier-Stokes Simulation of 2-D Unsteady Aerodynamics of a Turbine Cascade

I. W. McBean[1], F. Liu[2], K. Hourigan[1] M. Thompson[1]

[1]Department of Mechanical Engineering
Monash University Clayton 3800, Australia

[2]Department of Mechanical and Aerospace Engineering
University of California Irvine, CA 92697-3975

Abstract

A Navier-Stokes solver coupled with the k-ω turbulence model is developed to solve the unsteady flow through an oscillating turbine cascade. Calculations are performed in parallel in a time-accurate manner. A coupled simulation is performed for the Isogai wing model.

Introduction

Aeroelasticity in turbomachinery has been recognised as one of the most important problems presently facing the designers of turbomachinery blades [18]. The structural instability is due to the interaction between the unsteady aerodynamics and the structural dynamics of the blades. As the demand for lighter and more efficient machines develops, engineers seek to design blades that are increasingly more slender and subject to greater loading. As a consequence, turbine blades are more likely to be subject to the effects of dynamic loading due to unsteady aerodynamics. The coupling between the fluid and structure can lead to blade failure if, in the design phase, attention is not paid to the aeroelastic effects in the turbomachinery blade rows.

Although there have been a large number of 2-dimensional studies into unsteady aerodynamics in turbomachinery, these omit important 3-dimensional viscous and other effects. There are a number of review papers in the literature that list both computational and experimental simulations of aeroelasticity in turbomachinery [6, 18, 15, 19]. Whilst experimental studies play an important role in research into this phenomenon, computational simulations provide a number of key advantages. These include the ability to represent the flow over the whole flow-field at significantly lower cost. Thus results may be studied in detail, providing insights into flow behaviour and flow structures. In general, computational studies of unsteady aerodynamics require resources that are well beyond a single processor. However the development of multiple processor systems has greatly reduced simulation times through the calculation of problems in parallel.

The solution of field and fluid problems lend themselves easily to solution in parallel, as the computational domain may be divided into blocks and the field equations solved for each block on separate processors. A 3-dimensional unsteady Navier-Stokes code has been developed from code that calculated the steady state solution for single blade passages in turbine cascades [12, 14, 13]. The new implementation includes an unsteady solver, moving grid, multiple processor capability and structural model. The code has been described previously in more detail and validated [16]. The results presented are for both coupled and uncoupled configurations. In the first case modal equations are solved to simulate fluid structure interaction. The second case is a turbine model where the motion is prescribed.

Ultimately the code is intended to model 3-dimensional coupled aeroelasticity; however the present results will be used to validate and evaluate the effectiveness of the unsteady and moving grid parts of the fluid dynamics code.

Governing Equations and Numerical Method

The fluid field equations solved are the 3-dimensional, Favre-averaged Navier-Stokes equations coupled with the energy equation, and Wilcox's [20] k-ω turbulence model for closure. These are descretized on a structured hexahedral grid using the finite-volume representation. Artificial diffusion is used to suppress oscillatory behaviour in the flow field [8]. The upwind method of discretisation is employed for the convective terms of the k-ω turbulence model [14]. Both sets of equations are solved explicitly in a strongly-coupled manner through a 5 stage Runge-Kutta scheme. The time-accurate unsteady solution is found through Jameson's fully implicit dual time-stepping scheme [9, 1, 2].

The structural equations for the coupled spring mass system are decoupled using modal analysis. These are then reduced to ordinary differential equations and are solved using the same 5 stage Runge-Kutta scheme as for the flow solver. Details of the implementation may be found in the aforementioned references and in [16].

Moving Grid

The moving grid is implemented through transfinite interpolation. This method has the ability to regenerate the new grid at relatively low computational cost. It also has the advantage that for internal boundaries the grid remains continuous over the interfaces, whilst only the change in position of the corner points is required for the complete specification of the grid on a face. Thus the grid may be deformed throughout the fluid domain with the minimum of communication between blocks.

The geometry of the block structure is stored on the master processor. To maintain global grid uniformity Batina's spring network analogy [3] is used to define the corner points of the block. Each edge of the blocks is treated as a spring with its stiffness inversely proportional to its length. The location of each free internal corner of each block is determined through an iterative algorithm that establishes the equilibrium position of the nodes. Additional springs diagonally connect the corners within each block for additional control of grid deformation.

The moving grid method has been successfully applied to the modelling of aeroelasticity in wings [21, 11] and in two dimensional analysis of turbine aeroelasticity [17]. Details of the method and its implementation have been previously presented [21].

Parallel Implementation

The specific parallel implementation has been carefully considered to maximise code efficiency and flexibility. While each block consists of a structured grid, the blocks can be connected to each other in an unstructured manner, although the grids must be matched at the block interfaces. Efficient load balancing may be achieved by allocating more than one block to a single processor, with the objective that each processor computes approximately the same number of cells. Boundary conditions are separately specified for each block. The ghost cells are updated through the transfer of cells from neighboring blocks or by applying the corresponding boundary condition. The Message Passing Interface (MPI) is used to transfer variables belonging to boundary cells between neighboring blocks.

A number of high level MPI algorithms are exploited in the implementation. The interfaces between blocks are treated in a semi-structured manner. Each interface is represented as a subface of a block face. Within each sub-face the cells are stored as a 2-dimensional array and the transfer of cells is performed by a direct copy from the flow array to the MPI buffer. The new implementation is intended to increase code efficiency while also decreasing code complexity.

A multi-grid method is used to accelerate convergence as the governing fluid equations are solved in quasi-time. Each multigrid level is solved simultaneously, before the flow-field is interpolated to the next coarsest grid, or the residual is transferred to the next finest grid. At each level, boundary information is transferred between blocks so that boundary conditions may be updated.

Within the flow code, the solver module only requires information about the present block. This primarily involves the old solutions and boundary condition information. The implementation is therefore relatively compact and extremely flexible, the only drawback being the detail required in the boundary condition definition. The boundary conditions are defined only for the fine grid and the boundary condition information for the other grids in the multi-grid sequence is calculated internally.

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 [7] 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 1. 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.

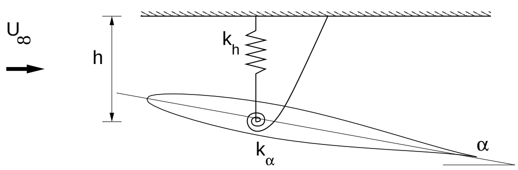

Figure 1: Isogai wing model

The important parameters for this simulation are the free stream Mach number Ma and the flutter velocity V_f. The flutter velocity is defined,

$$V_f = \frac{U_\infty}{b\omega_\alpha\sqrt{\mu}}. \quad (1)$$

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 *rho* and the structural natural frequency ω_α.

Results for the flutter boundary are shown in Figure 2 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 where carried out at fixed far field Mach number while the flutter velocity was varied. Each point required about five simulations to locate the flutter boundary. The line of best fit indicates the flutter boundary for the model.

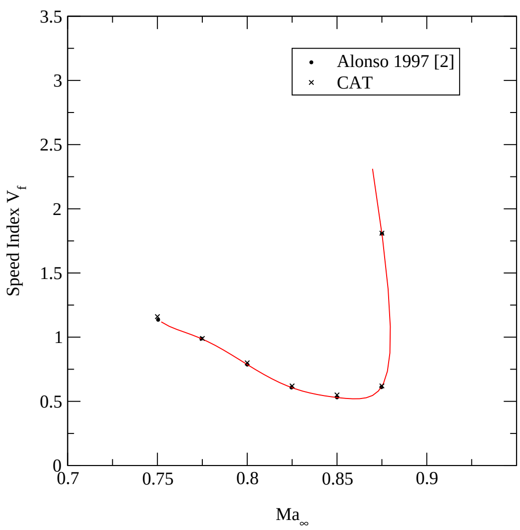

Figure 2: Flutter boundary for Isogai wing model

2-d Turbine Simulation

The geometric complexities and difficulties associated with the flow regime have led researchers in the field of experimental research to simplify and model aeroelastic and unsteady flows for turbomachinery, rather than take measurements on full scale machines. This results in experimental conditions that are better controlled and thus provides data that may be more easily analysed. Initially ten standard test case configurations [4] were compiled to accommodate a need for the research community to validate numerical models and to better understand the physics of the phenomenon.

In the study of turbomachinery, symmetry allows each blade row to be considered as a cascade in two or three-dimensions. Lane [10] was one of the first to identify and simplify the vibrational characteristics of turbine rotors. He reduced the number of possible system modes to one by considering a single equivalent blade. In this assumption the blade vibrational mode shape for each blade in the rotor is identical, with a phase shift between adjacent blades. The shape of the mode may be quite complex in that it involves a combination of bending and twist.

One result of this analysis is that the phase shift between blades, known as the inter-blade phase angle (IBPA) must be a multiple of the circumferential angle between blades. For a finite rotor, there are a finite number of possible IBPA's. This is due to the fact that a single blade must be in phase with itself in sustained vibration. The analysis was applied to compressor blade rows, but may be equally be applied to turbine blade rows.

In the computational model, assuming a particular IBPA allows a reduced number of blade passages to be considered. For example, for an IBPA of 90 degrees, four passages would be modelled with periodic conditions applied at the boundaries of the upper-most and lowest passages.

An example of two passages is shown in Figure 3. The grid was generated by assuming a spring network analogy and enforcing orthogonality at the blade surfaces. Geometric stretching was applied in the blade to blade direction and hyperbolic tangent stretching in the stream-wise direction between the blades. This produced an orthogonal H-grid.

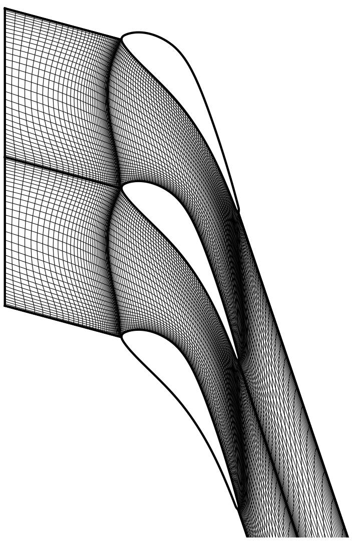

Figure 3: Geometry and computational grid for simulation of Standard Test Case 4

The Standard Test Case 4 is described as a cambered turbine cascade in transonic flow [4]. The experimental apparatus is an annular cascade and unsteady measurements were taken at mid-span at a variety of inlet and outlet conditions and IBPA's. Other authors have performed 2-dimensional simulations of this case [5] and made comparisons between Navier-Stokes calculations with the Baldwin-Lomax turbulence model and inviscid Euler simulations.

The same 2-dimensional simulation for the sub-case Test 552-B was performed with the present method both for the Navier-Stokes with k-ω turbulence model and inviscid Euler solvers. The case involves a simulated bending mode that is modelled by a periodic translation of the blades at an angle of 63 degrees from the axis, at a reduced frequency of $k_c = 0.1187$ and an amplitude of $b_c = 3.8 \times 10^{-3}$ non-dimensionalised with chord. Results are presented in Figure 4 for magnitude and Figure 5 for phase for the pressure coefficient for an IBPA of 180 degrees. These compare well with those presented by previous authors. The case is 2-dimensional and this was performed in the 3-dimensional code by using a small number of cells in the span-wise direction and applying inviscid boundary conditions at the hub and casing surfaces.

The unsteady surface pressure coefficient is defined as,

$$\tilde{c}_p = \frac{\tilde{p}(x,t)}{b_c(p_0 - p)} \tag{2}$$

where $\tilde{p}(x,t)$ is the unsteady pressure, p_0 is the total pressure defined at the inlet and p is the static pressure defined at the cascade inlet.

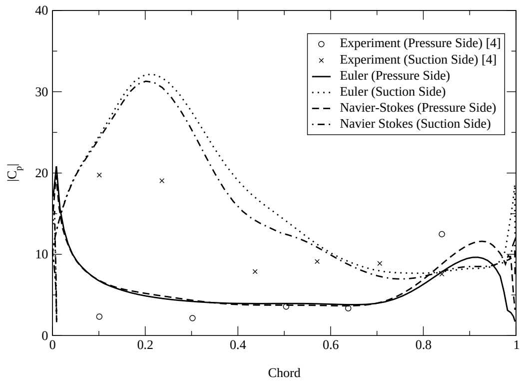

Figure 4: Magnitude of unsteady pressure coefficient for Test 552-B

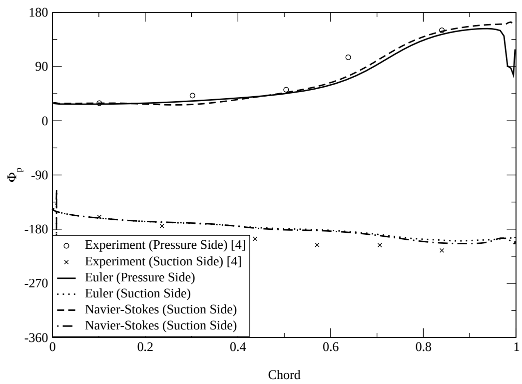

Figure 5: Phase of unsteady pressure coefficient for Test 552-B

Conclusions

A Navier-Stokes code has been developed to model aeroelasticity in turbomachinery. Two cases have been presented to demonstrate the ability of the code to model coupled structural fluid interaction and unsteady fluid dynamics. It is planned in the future to develop the code further to model 3-dimensional aeroelasticity. The 3-dimensional simulations require minimal changes to the code as the 3-dimensional geometry and governing equations are already implemented.

Acknowledgments

The authors gratefully acknowledge NPACI and the NSF for the use of Blue Horizon at San Diego under the NPACI award scheme, VPAC for the use of the state facility Compaq Alpha in Melbourne and APAC for the use of the national facility Compaq Alpha in Canberra under project d93 in the merit allocation scheme.

References

[1] Juan Alonso and Antony Jameson. Fully-implicit time-marching aeroelastic solutions. In *AIAA 32nd Aerospace Sciences Meeting*. AIAA, January 1994. 94-0056.

[2] Juan Jose Alonso. *Parallel Computation of Unsteady and Aeroelastic Flows Using an Implicit Multigrid-Driven Algorithm*. PhD thesis, Pinceton University, June 1997.

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

[4] A. Boelcs and T. H. Fransson. *Aeroelasticity in Turbomachines: Comparison of Theoretical and Experimental Cascade Results*. EPFL, Lausanne, 1986.

[5] B. Grueber and V. Carstens. Computation of the unsteady transonic flow in harmonically oscillating turbine cascades taking into account viscous effects. *Journal of Turbomachinery*, 120:104–111, January 1998.

[6] M. Imregun. *Recent Developments in Turbomachinery Aeroelasticity*. Wiley and Sons, 1998.

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

[8] A. Jameson, W. Schmidt, and E. Turkel. Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes. In *AIAA 14th Fluid and Plasma Dynamics Conference*. AIAA, June 1981.

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

[10] Frank Lane. System mode shapes in the flutter of compressor blade rows. *Journal of the Aeronautical Sciences*, pages 54–66, January 1956.

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

[12] Feng Liu. *Numerical Calculation of Turbomachinery Cascade Flows*. PhD thesis, Pinceton University, May 1991.

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

[14] Feng Liu and Xiaoqing Zheng. Staggered finite volume scheme for solving cascade flow with a k-ω turbulence model. *AIAA Journal*, 32(8):1589–1597, August 1994.

[15] J. G. Marshall and M. Imregun. An analysis of the aeroelastic behavior of a typical fan-blade with emphasis on the flutter mechanism. In *International Gas Turbine and Aeroengine Congress and Exhibition*. ASME, June 1996. 96-GT-78.

[16] I. W. McBean, F. Liu, and M. Thompson. A three dimensional navier-stokes code for aeroelasticity in turbomachinery. In *International Symposium on Unsteady Aerodynamics, Aeroacoustics and Aeroelasticity of Turbomachines*, September 2000. to be published.

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

[18] A. V. Srinivasan. Flutter and resonant vibration characteristics of engine blades. *Journal of Engineering for Gas Turbines and Power*, 119, October 1997.

[19] Joseph Verdon. Review of unsteady aerodynamic methods for turbomachinery aeroelastic and aeroacoustic applications. *AIAA Journal*, 31(2):235–250, February 1993.

[20] D. C. Wilcox. Reassessment of the scale-determining equation for avanced turbulence models. *AIAA Journal*, 26(11):1299–1310, 1988.

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