A Parallel Implicit Harmonic Balance Solver for Forced Motion Transonic Flow

Ken Badcock & Mark Woodgate

Department of Engineering

University of Liverpool

Liverpool L69 3GH

WCCM8 ECCOMAS 2008
June 30 - July 4 2008
Venice Italy

Time domain VS Frequency domain solvers

- If the solution is required only once periodic steady state is reached Frequency domain solvers can be used
  - Boundary conditions force the unsteadiness
  - Unsteadiness due to the flow field
- Time domain solver can capture arbitrary time histories vs Frequency domain periodic steady state
- Time domain is unsteady vs Frequency domain steady state
  - 32 points per period 15 inner iterations per point for N cycles vs M frequencies steady state calculations

Time domain Calculation with periodic solutions

Solver is parallel implicit dual time cell centred scheme (Badcock et al Progress in Aerospace Sciences 2000)

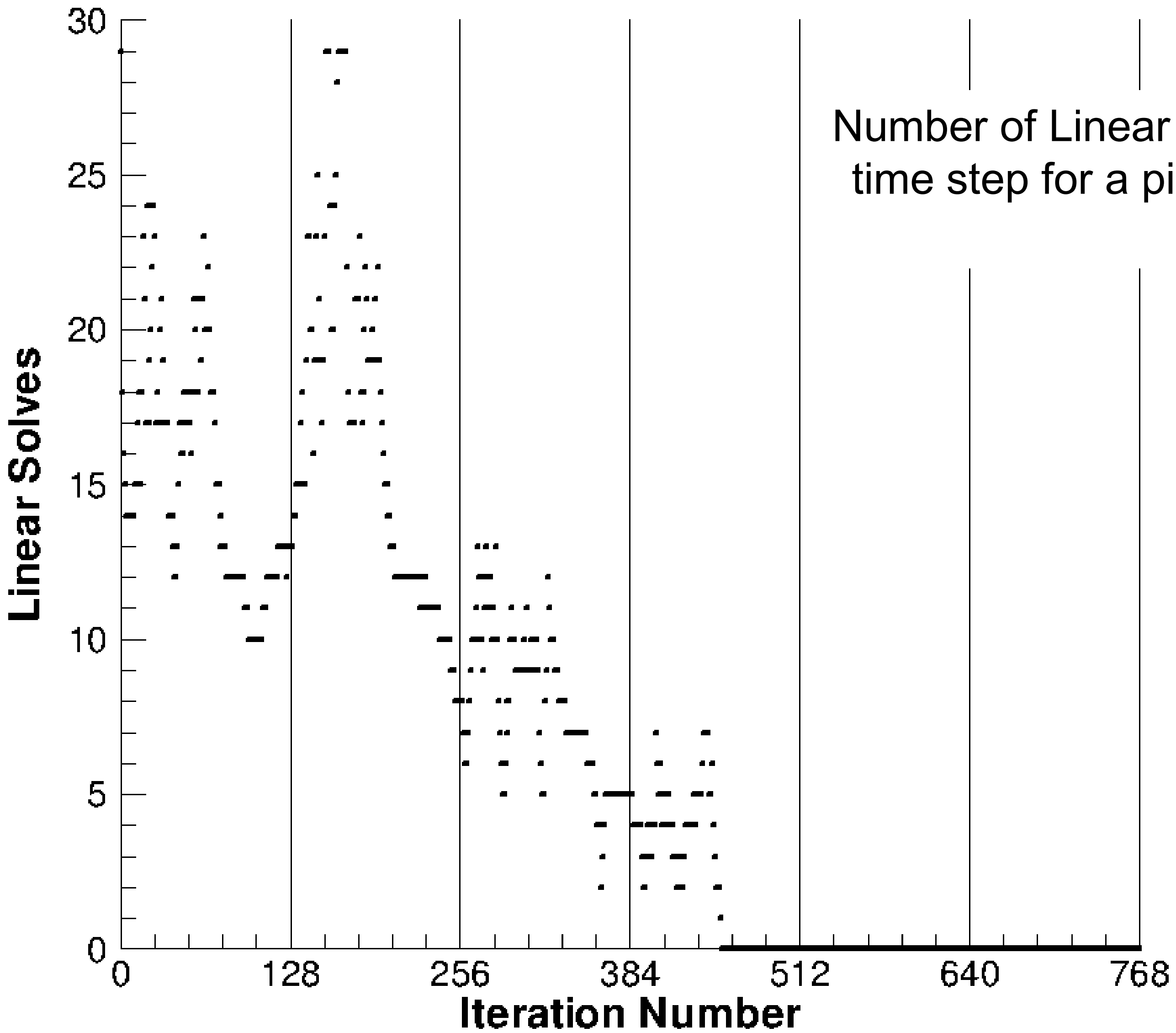
- MUSCL + Osher's scheme + approximate Jacobian.
- Krylov Subspace Method with BILU(k) Preconditioning

Its possible to use the periodic nature in time domain solutions

- At each time level store the complete solution
- After 1 ½ cycles read in the solution from the N-1 time level
- After few cycles the initial guess is the exact answer

Possible improvement use a variable convergence tolerance

- Base it on the change in the unsteady residual?



Number of Linear solves per real time step for a pitching aerofoil

Fourier Series Expansion

$$x(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\left[a_n \cos\left(\omega_n nt\right) + b_n \sin\left(\omega_n nt\right)\right]$$

$$\omega_n = n\frac{2\pi}{T}$$

The nth Harmonic of the function

Assume we know the time period

$$a_n = \frac{2}{T}\int_{t_1}^{t_2} x(t)\cos(\omega_n t)dt$$

Even Fourier Coefficients

$$b_n = \frac{2}{T}\int_{t_1}^{t_2} x(t)\sin(\omega_n t)dt$$

Odd Fourier Coefficients

Transforming to the Frequency Domain

Hall et al AIAA Journal 2002

$$I(t)=\frac{\partial W(t)}{\partial t}+R(t)=0$$

Assuming the solution and residual are periodic in time and truncate

$$R(t)=\hat{R}_0+\sum_{n=1}^{N_h}\left[\hat{R}_{a_n}\cos(\omega_n t)+\hat{R}_{b_n}\sin(\omega_n t)\right]$$

$$W(t)=\hat{W}_0+\sum_{n=1}^{N_h}\left[\hat{W}_{a_n}\cos(\omega_n t)+\hat{W}_{b_n}\sin(\omega_n t)\right]$$

Using Fourier transform on the equation then yields the following

$$\hat{R}_0=0 \qquad \omega_n\hat{W}_{b_n}+\hat{R}_{a_n}=0 \qquad -\omega_n\hat{W}_{a_n}+\hat{R}_{b_n}=0$$

This is $N_t=2N_h+1$ equations for N_h harmonics

Solving the Frequency Domain Equations

$$\omega A\hat{W}_{i,j,k} + \hat{R}_{i,j,k} = 0 \qquad \text{NONLINEAR}$$

It may be impossible to determine explicit expression for $\hat{W}_{i,j,k}$ in terms of $\hat{R}_{i,j,k}$

$$W_{i,j,k} = \begin{pmatrix} W_{i,j,k}(t_0 + \Delta t) \\ W_{i,j,k}(t_0 + 2\Delta t) \\ \vdots \\ W_{i,j,k}(t_0 + T) \end{pmatrix} \qquad R_{i,j,k} = \begin{pmatrix} R_{i,j,k}(t_0 + \Delta t) \\ R_{i,j,k}(t_0 + 2\Delta t) \\ \vdots \\ R_{i,j,k}(t_0 + T) \end{pmatrix}$$

$$\hat{W} = EW$$

$$\hat{R} = ER$$

Hence we can rewrite the Frequency domain equations in the time domain

$$\frac{\partial W_{i,j,k}}{\partial t} + \omega E^{-1} AEW_{i,j,k} + R_{i,j,k} = 0$$

Calculation of Derivatives

Assume a vector

$$X = \begin{pmatrix} X(t_0 + \Delta t) \\ X(t_0 + 2\Delta t) \\ \vdots \\ X(t_0 + T) \end{pmatrix}$$

How do you calculate the vector

$$\dot{X} = \begin{pmatrix} \dot{X}(t_0 + \Delta t) \\ \dot{X}(t_0 + 2\Delta t) \\ \vdots \\ \dot{X}(t_0 + T) \end{pmatrix}$$

Use the relationship

$$\dot{X} = \omega E^{-1} A E X = \omega D X$$

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

$$\begin{bmatrix} 0 & d_1 & -d_2 & d_3 & -d_3 & d_2 & -d_1 \\ -d_1 & 0 & d_1 & -d_2 & d_3 & -d_3 & d_2 \\ d_2 & -d_1 & 0 & d_1 & -d_2 & d_3 & -d_3 \\ -d_3 & d_2 & -d_1 & 0 & d_1 & -d_2 & d_3 \\ d_3 & -d_3 & d_2 & -d_1 & 0 & d_1 & -d_2 \\ -d_2 & d_3 & -d_3 & d_2 & -d_1 & 0 & d_1 \\ d_1 & -d_2 & d_3 & -d_3 & d_2 & -d_1 & 0 \end{bmatrix}$$

Computational cost of Method

For the 3D Euler Equations and the current formulation

Number of Harmonics	0	1	2	3	4	8
Memory compared to steady solver	1	3.85	7.86	13.0	19.3	55.9

- It is possible to reduce memory requirements with different storage.
- Three possible initial guesses
  - Free stream for all time levels – Very low cost and low robustness
  - The mean steady state for all time levels – Low cost robust
  - The steady state for each time level – High cost most robust
- The Matrix is HARDER to solve than the steady state matrix and there are also lower convergence tolerances on steady state solves.
  - Systems becomes harder to solve as number of harmonics increases

Parallel Implementation

- The parallel implementation is exactly the same as the time marching solver
- The Halo cells are numbered in an analogous way
  - Each halo cell now has $2N_h + 1$ lots of flow data
- The BILU(k) preconditioner is block across processors
  - Hence the preconditioning deteriorates as processors increase

Number of Procs	CPU time	Efficiency
1	3134	N/A
2	1588	98.6%
4	841	93.1%
8	469	83.5%

3D Test wing with 200K cells. Beowulf cluster of Intel P4's with 100Mbits/sec bandwidth

Timing for CT1 test case

AGARD Report No. 702, 1982

$$\alpha_m = 2.89 \quad \alpha_0 = 2.41 \quad M_\infty = 0.6 \quad k = 0.0808$$

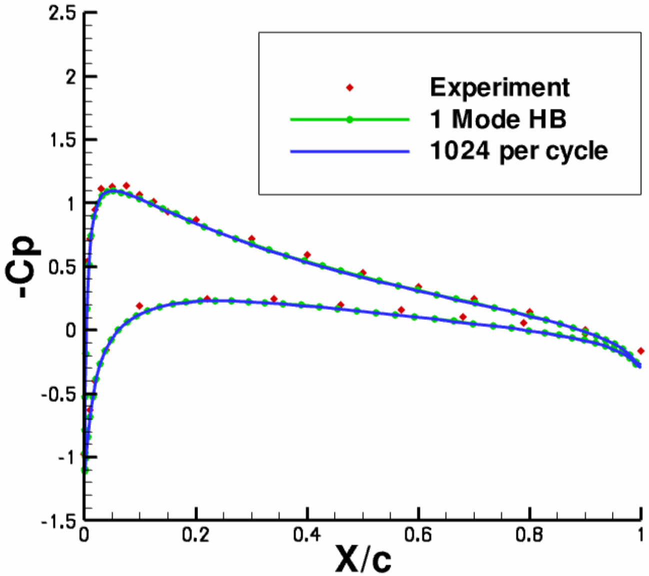
Steady state solve is 3 seconds for 128x32 cell grid for a single 3.0Ghz P4 Node

Steps per cycle	CPU time for 6 cycles
16	64
32	117
64	218
128	390
256	683
512	1205
1024	2120

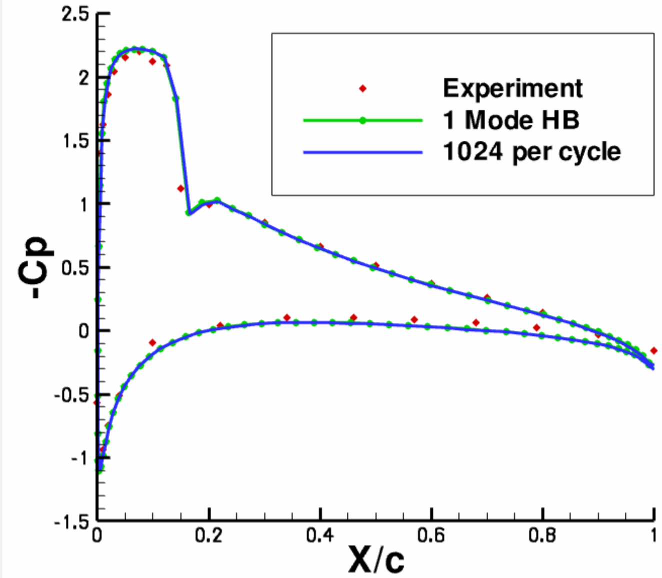
Harmonics	CPU time
1	15
2	25
3	42
4	75

15-25 implicit steps to reduce the residual 8 orders

CT1 Test Case - 1 Harmonic Mode

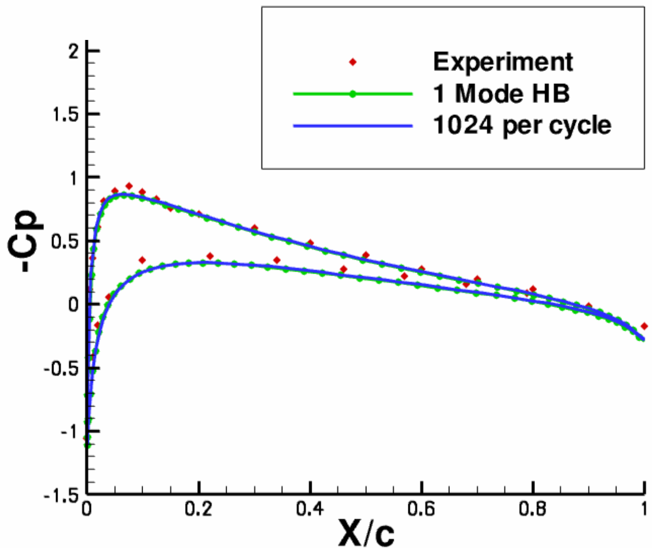


1 Harmonic gives 3 time slices



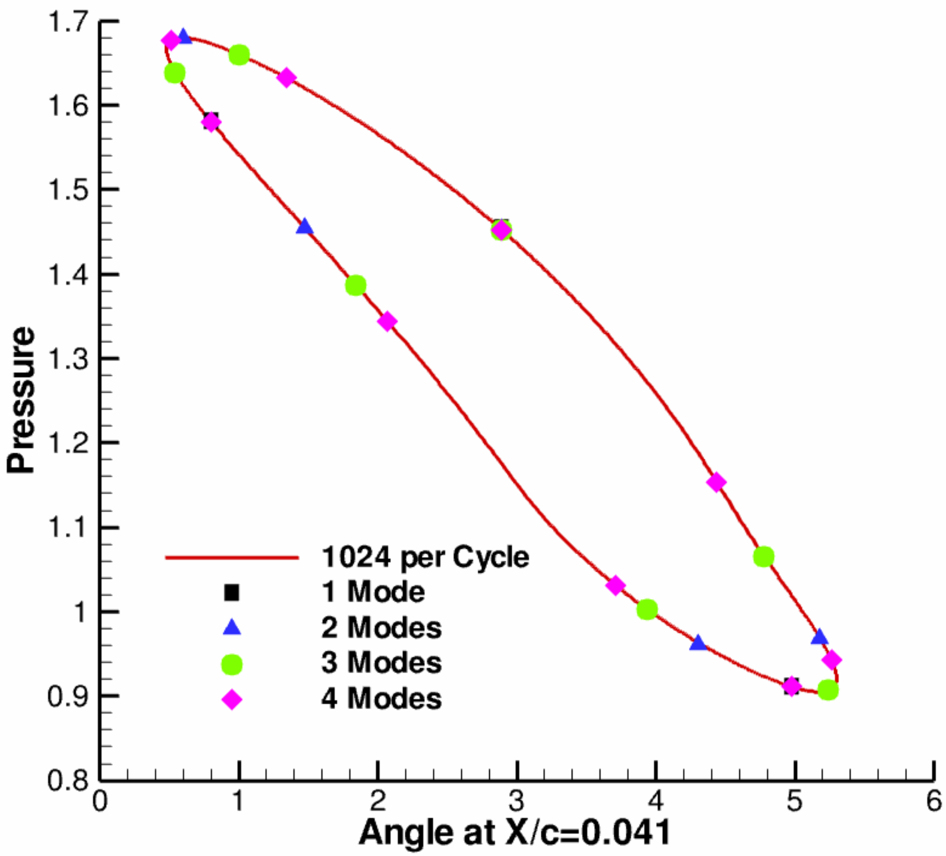
2.97 Degrees up

5.0 degrees down

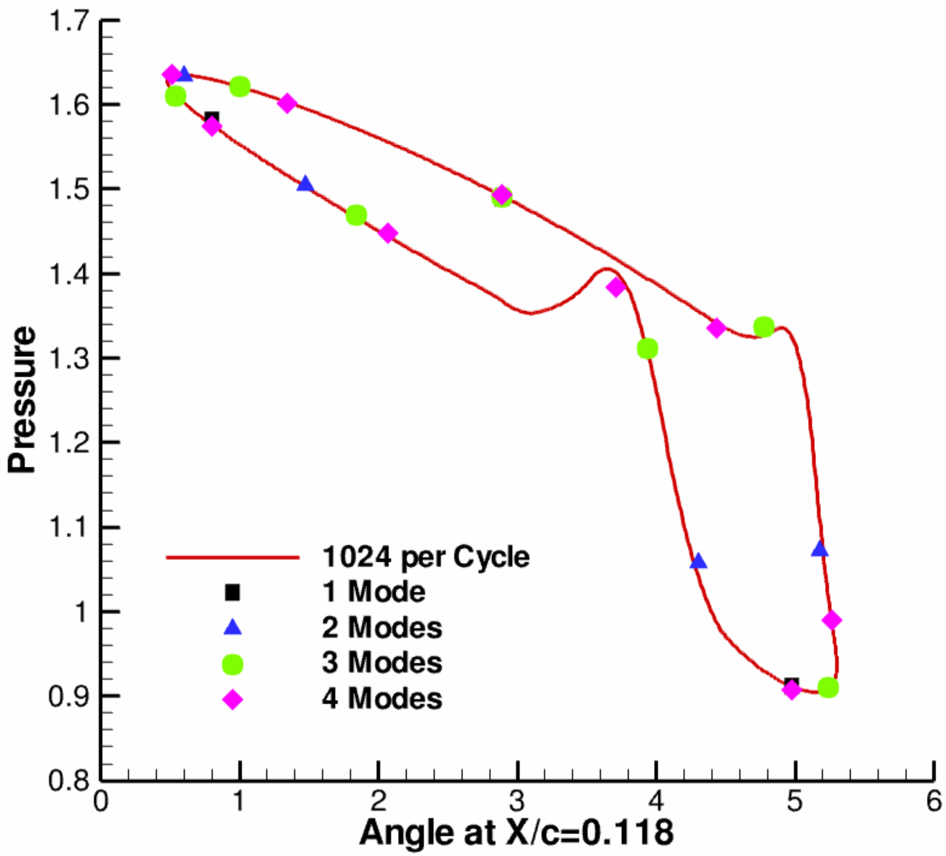


0.85 Degrees down

CT1 Test Case Pressure next to surface

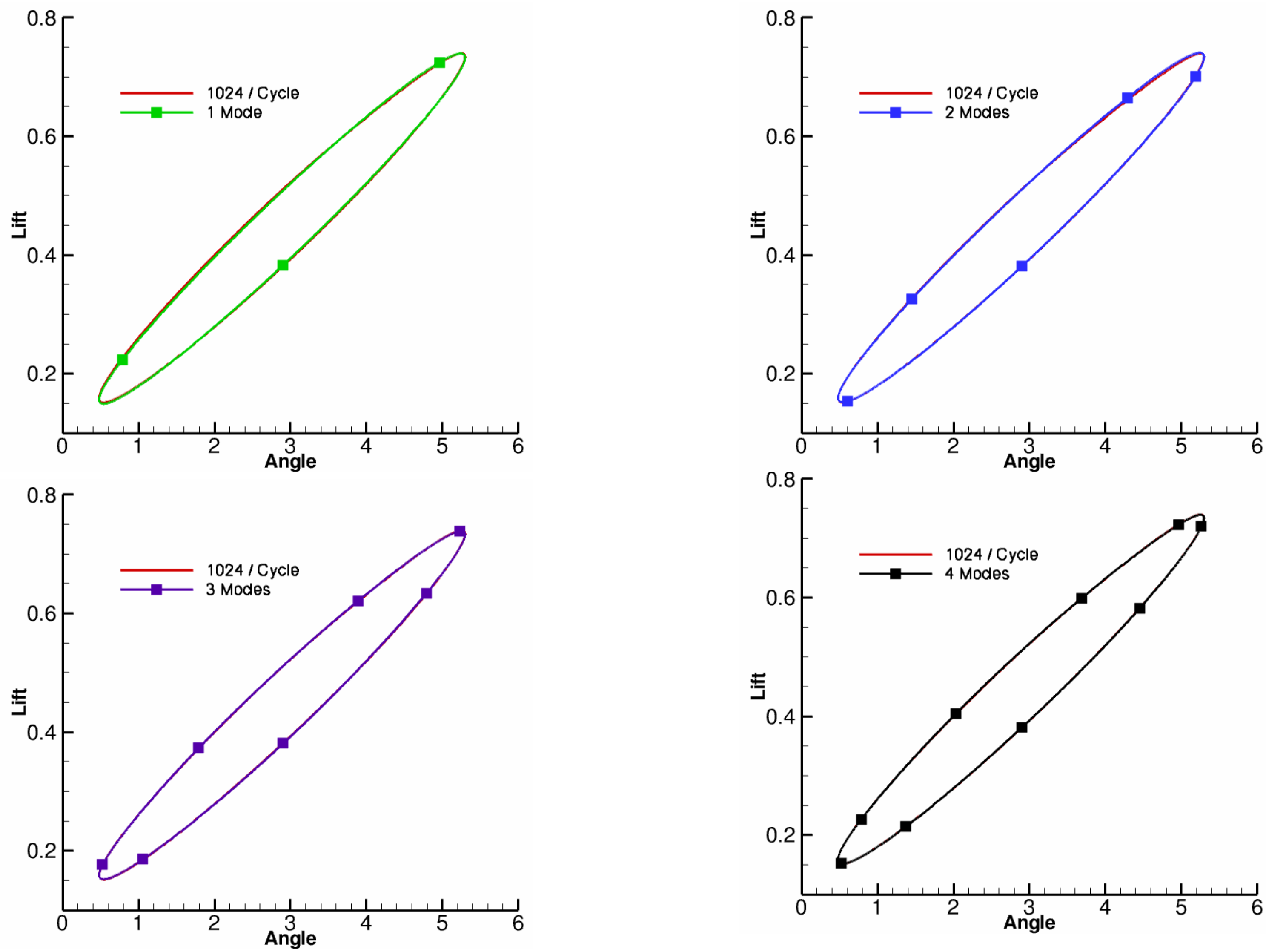


Forward of shock

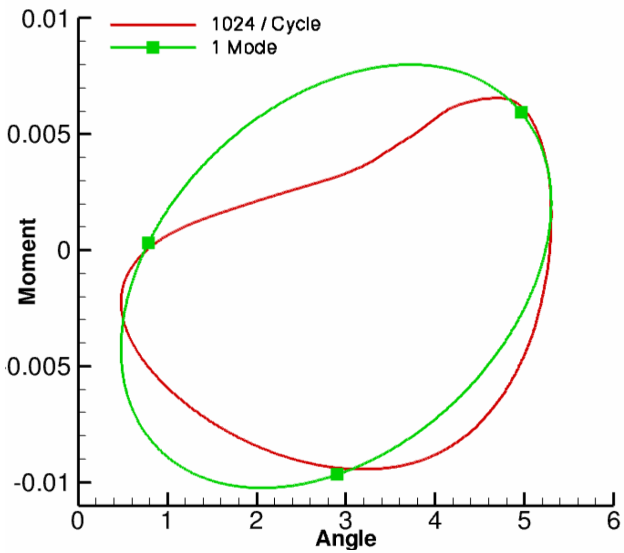


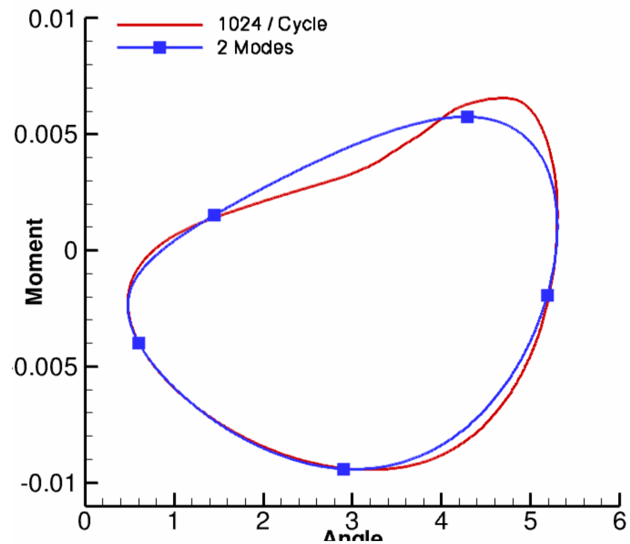
Shock passes through this point

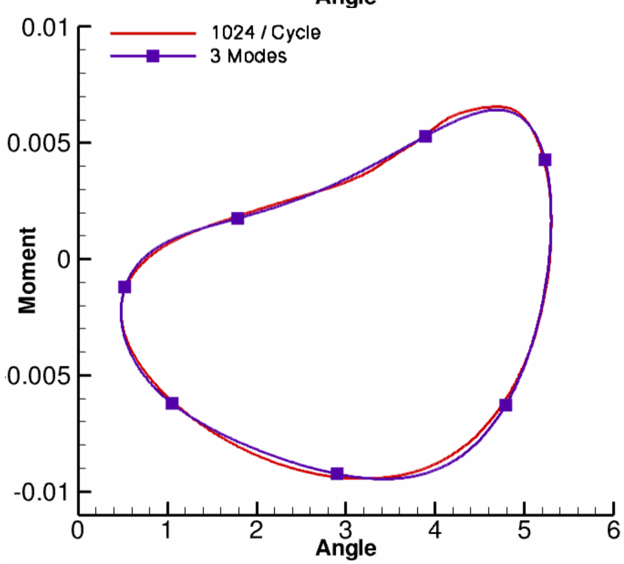
Reconstruction of full lift cycle

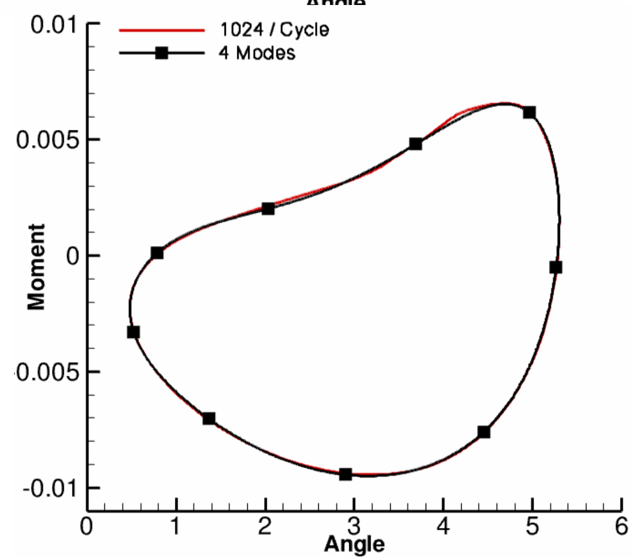


Reconstruction of full moment cycle



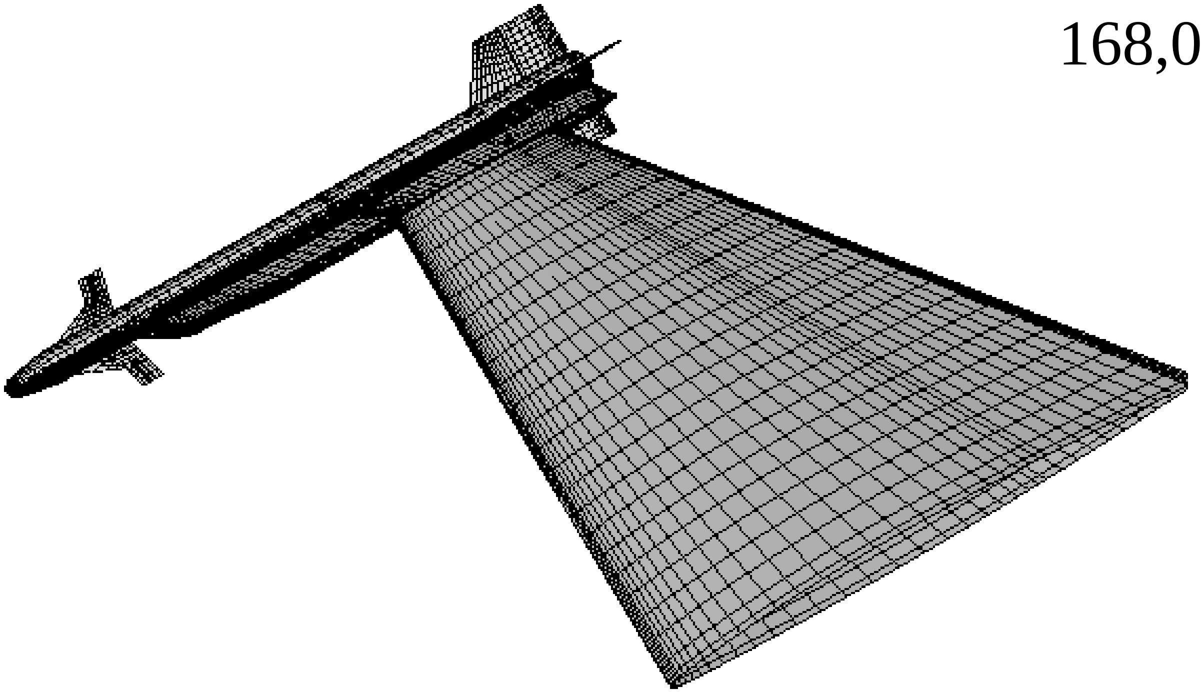






Surface grid for F5WTMAC case

Research and Technology Organization RTO-TR-26 2000



168,000 cells and 290 blocks

Timings for F5WTMAC Run355

WTMAC: Wing with tip launcher + missile body + aft fins + canard fins

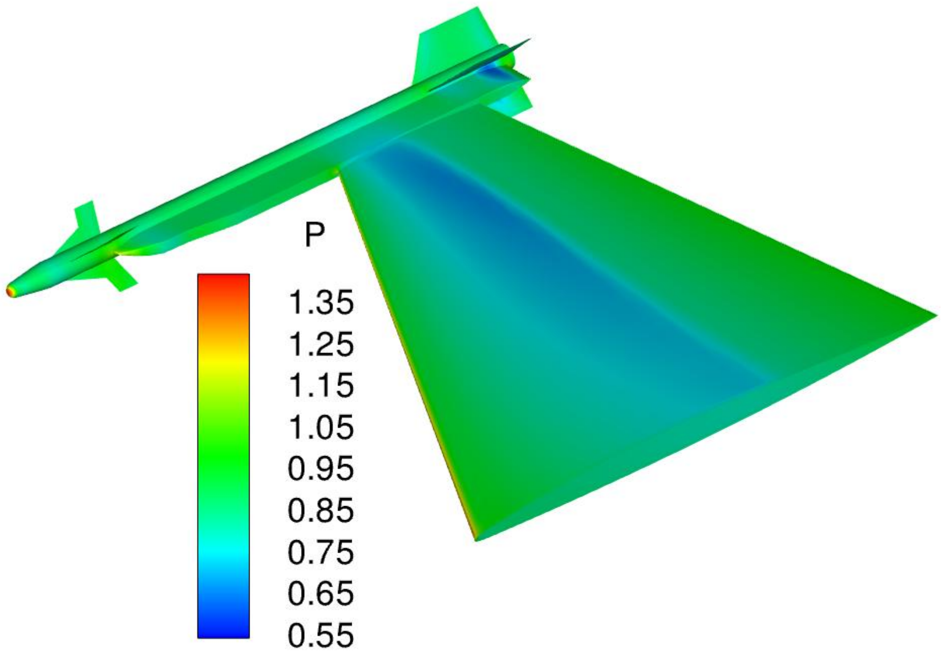
$$\alpha_m = 0.004 \quad \alpha_0 = 0.117 \quad M_\infty = 0.896 \quad k = 0.275$$

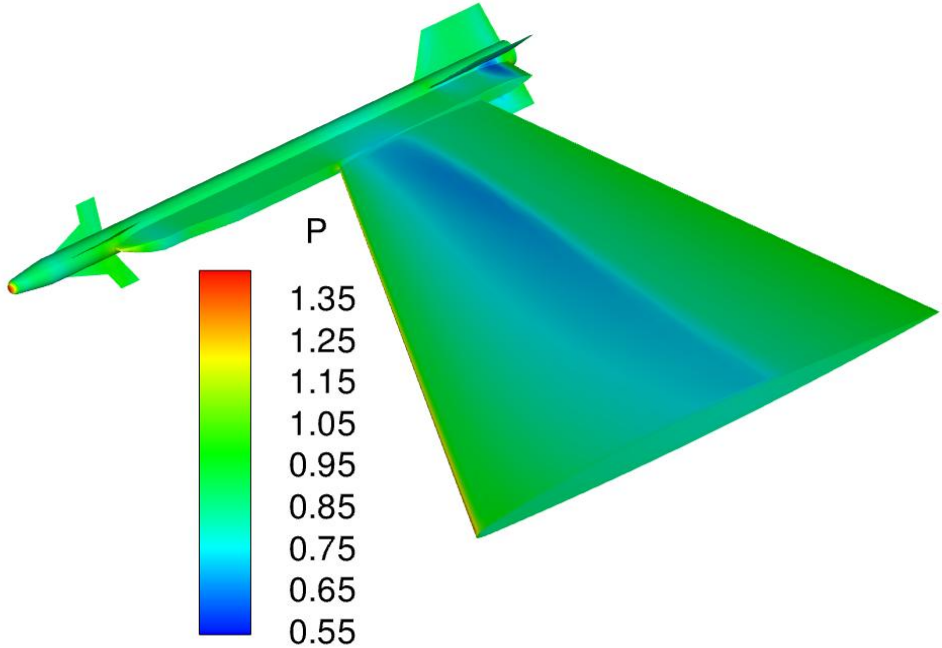
Steps per cycle	CPU time Minutes for 6 cycles
16	160
32	246
64	391

Harmonics	CPU time Minutes
1	39
2	15*8 procs

Efficiency was low due to poor partitioning of the blocks
Impossible to run sequentially

Surface Pressure





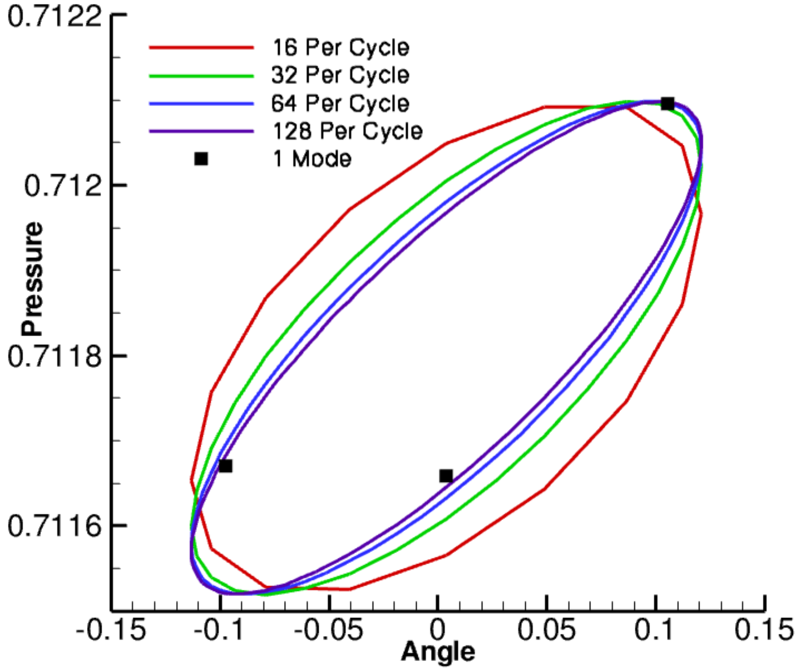
Time Marching

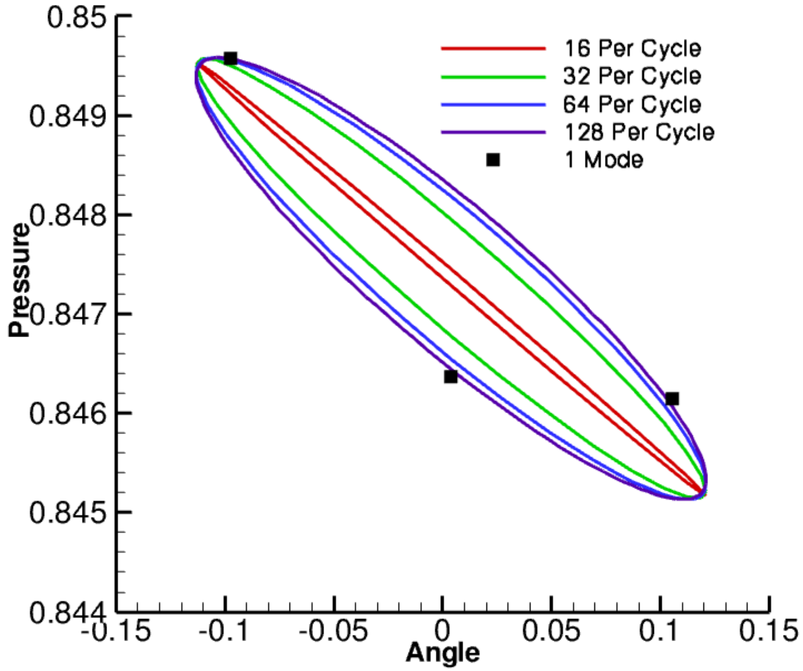
$$\alpha = 0.004 \quad \text{up}$$

1 Mode Harmonic Balance

Pressure at single points

83% of span





35% of Chord

128 steps per cycle enough to converge in time.

65% of Chord

Conclusions & Future Work

- An implicit parallel frequency domain method has be developed from an existing implicit unsteady solver
- A few Harmonic modes can be calculated at a cost of less than 50 steady state calculations

- Improvements in solving the linear system?
- Improvements the parallel efficiency
  - Better partitioning of the blocks - work and communication
  - Renumber of the internal cells
- Allow the building of aerodynamic tables used in flight mechanics