Application of an implicit dual-time stepping multi-block solver to 3D unsteady flows

R. Steijl[a], P. Nayyar[a*], M.A. Woodgate[a], K.J. Badcock[a] and G.N. Barakos[a]

[a]CFD Laboratory, Department of Aerospace Engineering, University of Glasgow, Glasgow, G12 8QQ, United Kingdom

This work discusses the application of a parallel implicit CFD method to challenging 3D unsteady flow problems in aerospace engineering: transonic cavity flows and the flow field around a helicopter rotor in forward flight. The paper discusses the computational details of simulations using the HPCx supercomputer (1600 processors) of Daresbury Lab., U.K. and a Beowulf cluster (100 processors) of the CFD Laboratory of the University of Glasgow. The results show that accurate simulations based on Large-Eddy Simulation (LES) and Detached-Eddy Simulation (DES) at realistic Reynolds numbers require impractical run times on the Beowulf cluster. A simulation of a full helicopter geometry is similarly beyond the limits of the 100-processor Beowulf cluster.

1. INTRODUCTION

Many of the present CFD applications in aerospace engineering involve unsteady three-dimensional aerodynamic problems. In contrast to steady state flows, which can typically be tackled in a matter of hours on a multi-processor machine or on a Beowulf cluster, unsteady flows require days of CPU time.

This paper presents the application of a parallel, unfactored, implicit method for the solution of the three-dimensional unsteady Euler/Navier-Stokes equations on multi-block structured meshes [1]. For time-accurate simulations, dual time-stepping is used. The solver and its performance on Linux clusters was previously discussed in refs.[2] and [3].

The application examples presented here are for three-dimensional unsteady aerodynamic problems: transonic cavity flow and flow around a helicopter rotor in forward flight. The simulations were carried out on the HPCx supercomputer of the Daresbury Lab. in the UK[6] and the local Beowulf cluster (comprising 100 Pentium 4 processors).

The CFD method and its parallelization are described in Sections 2 The application to 3D unsteady flow problems is described in Section 3, while conclusions are drawn in Section 4.

*Present address: Aircraft Research Association Ltd., Manton Lane, Bedford, Bedfordshire, England MK41 7PF

2. CFD METHOD AND PARALLELISATION

The unsteady Navier-Stokes equations are discretised on a curvilinear, multi-block, body conforming mesh using a cell-centred finite volume method. The convective terms are discretised using Osher's upwind scheme [4] and MUSCL variable extrapolation is used to provide second-order accuracy. The Van Albada limiter is used to prevent spurious oscillations around shock waves. Central differences are used for the viscous terms. The solver includes a range of one- and two-equation turbulence models as well as LES based on the Smagorinsky model and DES Spalart-Almaras model, as described by[9]. A dual-time stepping method is employed for time-accurate simulations, where the time derivative is approximated by a second-order backward difference [5]. The resulting non-linear system of equations is solved by integration in pseudo-time using a first-order backward difference. In each pseudo-time step, a linearisation in pseudo-time is used to obtain a linear system of equations, which is solved using a Generalised Conjugate Gradient method with a Block Incomplete Lower-Upper (BILU) pre-conditioner. The method is detailed in ref.[1]. Regarding parallelisation of the above method few changes were necessary:

- The flux Jacobians resulting from the linearisation in pseudo-time are employed in an approximate form that reduces the number of non-zero entries and as a result the size of the linear system. The use of the approximate Jacobian also reduces the parallel communication since only one row of halo cells is needed by the neighbouring process in the linear solver instead of two in the case of an 'exact' Jacobian.
- The communication between processes is minimised by decoupling the BILU factorisation between blocks.
- On each processor a vector is allocated that contains all the halo cells for all grid blocks.
- Inter-process communication is performed by sending a series of messages between the respective processes, each corresponding to a block connection, containing the halo cell data. The messages are sent in chunks of $10,000$ double precision numbers using non-blocking send and receive MPI functions.

The parallel implementation was presented previously in refs. [2] and [3]. and the solver has been used on a range of platforms, including Beowulf clusters consisting of various generations of Pentium processors and multi-processor workstations. Recently, the solver was ported to the HPCx computer at Daresbury Laboratory. The HPCx system comprises 50 IBM Power4+ Regatta nodes, i.e. 1600 processors, delivering a peak performance of 10.8 TeraFlops[6].

3. EXAMPLES OF 3D UNSTEADY APPLICATIONS

3.1. Transonic cavity flows

This section presents results from a computational study of transonic cavity flows, in which the formation of highly unsteady turbulent flow structures and the resulting noise production is the main interest[7]. In cavity flows, the flow separates at the sharp edge at the front of the cavity while further downstream two flow patterns may be encountered.

For the first pattern, the shear layer formed by the separation at the cavity front spans the entire cavity and re-attaches at the rear of the cavity. This is referred to as *open* cavity. In contrast, the shear layer in a *closed* cavity re-attaches at the cavity floor, then separates from the cavity floor further downstream, forming a shear layer that re-attaches at the rear cavity edge. The conditions considered here result in an *open* cavity flow.

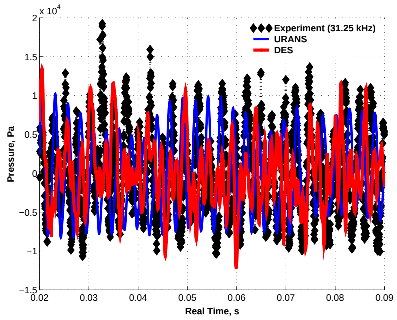
Three approaches for the turbulence modelling were used: unsteady RANS (URANS) using the $k - \omega$ model, Large-Eddy Simulation (LES) and Detached-Eddy Simulation (DES). LES works by filtering the flow structures in terms of scale size, with the larger scales explicitly resolved and the smaller ones modelled using a sub-grid scale (SGS) model. Pure LES can still be expensive, however, and recent endeavours have looked at developing hybrids of Unsteady Reynolds-Averaged Navier-Stokes (URANS) and LES to compromise the best of both methods. One example of such developments is the DES method introduced by Spalart et al. [9].

Here, a clean rectangular cavity with a length-to-depth ratio (L/D) of 5 and a width-to-depth ratio (W/D) of 1 is considered for two cavity configurations: one with doors-on and another with doors-off. The free-stream Mach number is 0.85 and a Reynolds number of one million based on the cavity length. These conditions result in an *open* cavity flow for both configurations.

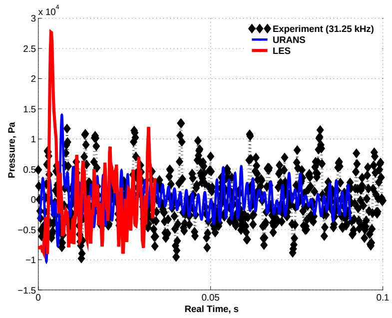
Pressure traces and visualisation of the flow-field inside the cavity from DES simulation for the doors-on configuration and LES simulation for the doors-off configuration are illustrated in Figure 1. Experimental pressure signals (provided by Ross et al.[10] and sampled at 31.25 kHz for doors-on and 6 kHz for doors-off) and numerical results with URANS for both doors-on (Figure 1(a)) and doors-off (Figure 1(b)) are also included for reference. The high Reynolds number considered here requires the use of high density grids. Small time-steps are required as a result of the high frequency unsteady flow features, with frequencies as high as 1 kHz. This results in an overall number of time-steps of approximately 50,000 required to simulate 0.2 seconds of the flow, which is just enough for gathering the flow statistics required for LES. The high density grids combined with the large number of time steps makes this flow computation very demanding, which is why the HPCx super-computing facility was exploited.

At a Reynolds number of 1 million, the flow in the cavity is turbulent. Combined with the presence of walls and the presence of a shear layer that separates the external (fast-flowing) fluid with the internal (slow-moving) cavity fluid, high levels of dissipation exist signifying that a large number of turbulent length scales are present. Good resolution of this turbulent spectrum is important in order to understand the function of turbulent processes and the source of acoustics inside the cavity. Without the use of massively parallel computers such as the HPCx, simulation of such turbulent flow-fields within realistic run times becomes impossible.

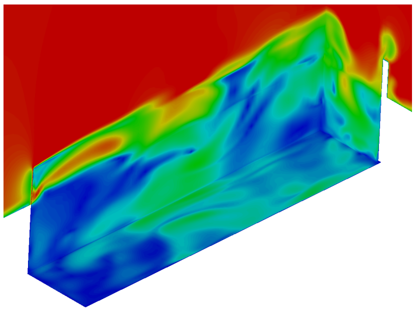
Table 1 shows details of three calculations: A DES on a grid with 4.5 million points on 320 processors (HPCx), an LES on a grid with 4.5 million points (24 processors on Beowulf cluster) and an URANS ($k - \omega$) simulation on a 1.5 million point grid on 19 processors of the local Beowulf cluster. A proper spectral decomposition of the DES and LES flow-field requires the calculation to run for long durations (at least 0.1s) to obtain sufficient samples (after sampling at either 31.25 kHz (for doors-on) and 6 kHz (for doors-off)) for analysis of the frequency content inside the cavity. Even after approximately



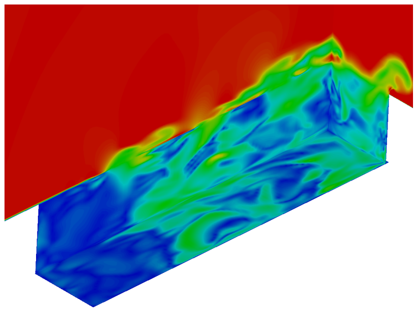
(a) Doors-On Pressure (DES)



(b) Doors-Off Pressure (LES)

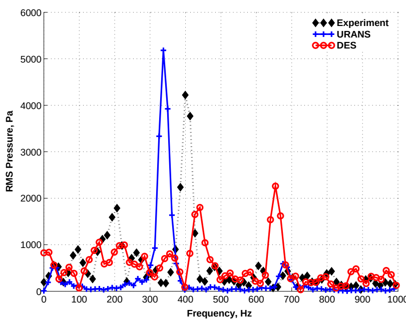


(c) Doors-On flow field (DES)

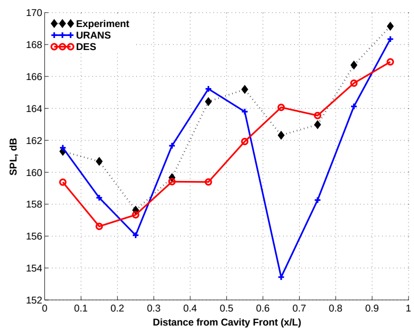


(d) Doors-Off flow field (LES)

Figure 1. Pressure traces and visualisation of the flow features inside the 3D, L/D=5, W/D=1 cavity with doors-on using DES and doors-off using LES. Pressure traces contain experimental signal (black with diamond symbols) and DES results (red). For reference, results from URANS are also included (blue). Flow-field plot consist of Mach contours normalised by free-stream Mach number of 0.85.



Power spectral density



Sound presssure level

Figure 2. Power spectral density and sound pressure level versus distance along cavity floor. Doors-on case, L/D=5, W/D=1 cavity, free-stream Mach number 0.85.

40,000 CPU hours of run-time, the 4.5 million LES calculation is far from complete as shown in the pressure signal in Figure 1(b).

For the doors-on cavity configuration, the power spectral density and the sound pressure level on the cavity floor versus distance from the cavity front are presented in Figure 2. The CFD results are compared with the experimental data of Ross et al.[10]. The comparison shows a good agreement for the DES results. The unsteady RANS results show poor correlation with experimental data, especially for the high-frequency modes. The poor predictions for power spectral density and the sound pressure level from the URANS simulation can be explained by the failure of this simulation to resolve the break-up of the shear layer that spans the cavity. This shear layer break-up is resolved in the LES and DES simulations.

Table 1
DES, LES and URANS calculation details on HPCx and Beowulf cluster

Calculation Details	DES	LES	URANS
Platform	HPCx	Beowulf cluster	Beowulf cluster
Cavity Configuration	Doors-On	Doors-Off	Doors-On
Grid Size	4.5×10^6	4.5×10^6	1.5×10^6
Processors	320	24	19
Time-Step (s)	1.81×10^{-6}	1.81×10^{-6}	1.81×10^{-5}
Pseudo-Steps/Time-step	6	4	39
Time-Steps/min.	9.72	0.723	0.425
Total Time-Steps	50,200	50,000	5,506
Total CPU Hours	28,100	1,565	3,121
Signal Duration	0.1 s	0.1 s	0.1 s
Total Run-time	3.46 days	48 days	9 days

3.2. Helicopter rotor in forward flight

This example combines a complex geometry with a flow field rich in fluid mechanics phenomena including strong interacting vortices, the formation of a vortex wake that spirals down below the rotor disk, transition to turbulence and a wide variation of the Mach and Reynolds numbers in the radial direction and around the azimuth[11]. An additional difficulty, is the strong link between the aerodynamics and the aeromechanics, i.e. to achieve a level flight, the rotor requires a blade pitch that changes periodically during the rotor revolution. The forward flight velocity leads to one side of the rotor disk with high blade-normal velocities (advancing side) and one with lower blade-normal velocities (retreating side). Using a lower blade pitch on the advancing side and an increased blade pitch on the retreating side, the rotor revolution-averaged roll and pitching moments can be canceled out. Furthermore, the rotor blades are hinged to allow for flapping (blade motion normal to vertical plane) and a lead-lag deflection (motion in the

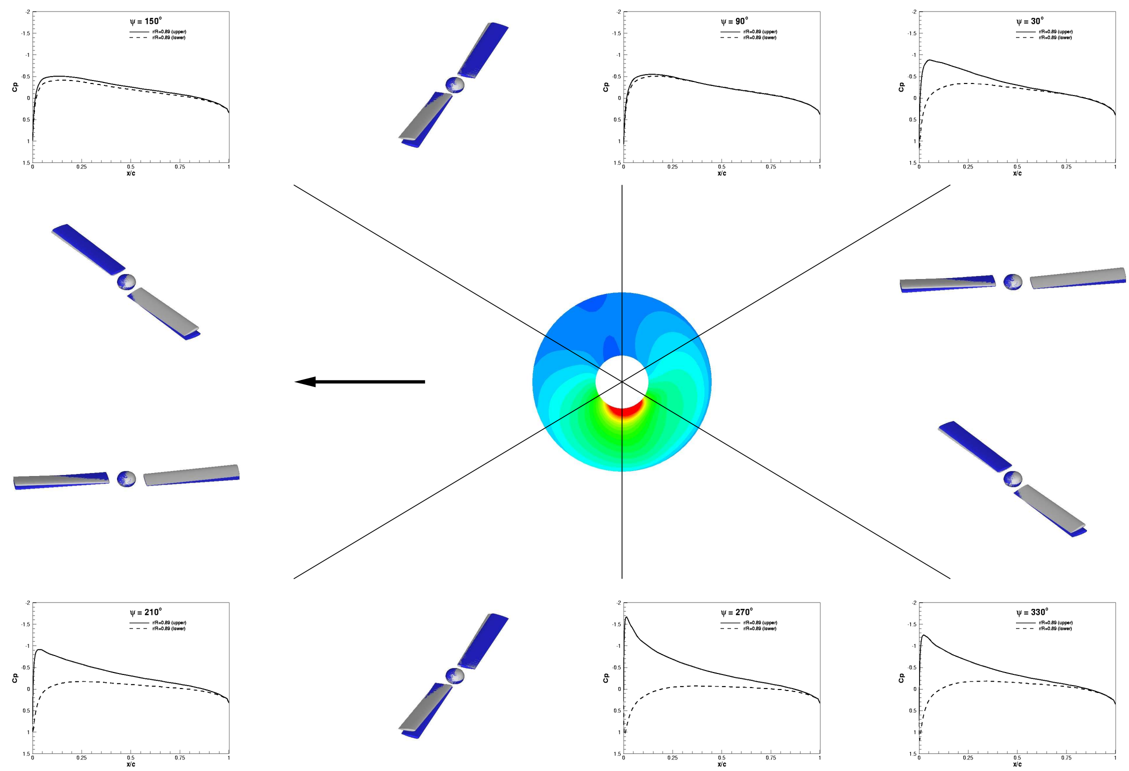


Figure 3. Geometry and chordwise pressure distribution for a fully articulated 2-bladed rotor in forward flight. The grey shade shows the rotor surface with periodic changes in pitch, flapping and lead-lag deflection. The blue shade shows the original blade position.

horizontal plane). The required control input (the blade pitch) and the resulting blade motion form part of the solution. This is known as the *trimming problem* and good CFD investigations are described in refs.[12], [13] amongst others.

The test case considered here is a two-bladed rotor with low-aspect ratio blades. The tip and forward flight Mach numbers are 0.6 and 0.09, respectively. The simulation involves periodic blade pitching, flapping and lead-lag motions. Figure 3 shows how the rotor geometry changes during a rotor revolution. The grey shadings show the geometry at various azimuthal positions compared to an equivalent rotor geometry without blade motions (blue). Also shown is the chordwise surface pressure distribution for a radial station at 89% of the rotor radius. Table 2 shows details of the forward flight case shown in Figure 3 and estimates of CPU times for forward flight cases currently underway. As shown, an affordable Beowulf cluster ($\leq$ 100 processors) does not provide the capability to simulate the viscous flow around a full helicopter configuration in forward flight (mesh size 15-30 $\cdot 10^6$ points) within realistic time. Supercomputing facilities, such as the HPCx may thus be required for simulations of such flows.

Table 2
Computational requirements for simulations of helicopter rotor in forward flight

Calculation Details	2-bladed rotor inviscid (measured)	2-bladed rotor RANS (estimate)	4-bladed rotor inviscid (measured)	4-bladed rotor RANS (estimate)	4-bladed rotor + fuselage RANS (estimate)
Grid Size	$1.2 \cdot 10^6$	$4.0 \cdot 10^6$	$4.0 \cdot 10^6$	$10 \cdot 10^6$	$15 \cdot 10^6$
Processors	20	40	40	100	200
Pseudo-Steps/ Time-step	40	40	40	40	40
Total Time-Steps	5000	5000	5000	5000	5000
Total CPU Hours	1,440	4,000	4,000	10,000	20,000
Total Run-time	72 hrs	100 hrs	100 hrs	100 hrs	100 hrs

4. CONCLUSIONS

The application of a parallel implicit multi-block CFD method to two challenging problems in aerodynamics is presented. The computer platforms used here are a Beowulf cluster and the HPCx supercomputer. Obtained results indicate that the parallel implementation of the solver is both robust and efficient on both platforms.

The problems considered were the transonic cavity flow and the flow around a helicopter rotor in forward flight. Using a Beowulf cluster (comprising 100 Pentium 4 processors), these flow can be analysed using CFD simulations. The run times, however, become excessive if the details of the turbulent flow must be resolved using LES or DES.

Acknowledgements
The financial support of the Engineering Physical Sciences Research Council (EPSRC) and the UK Ministry of Defense (MoD) under the Joint Grant Scheme is gratefully acknowledged for this rotorcraft project. This work forms part of the Rotorcraft Aeromechanics Defense and Aerospace Research Partnership (DARP) funded jointly by EPSRC, MoD, DTI, QinetiQ and Westland Helicopters. For the cavity work, the financial support from BAE Systems is gratefully acknowledged.

REFERENCES

1. K. Badcock, B. Richards and M. Woodgate, Elements of Computational Fluid Dynamics on Block Structured Grids Using Implicit Solvers. (2000) Progress in Aerospace Sciences, 36(5-6): 351-92.
2. M. Woodgate, K. Badcock, B. Richards and R. Gatiganti. A parallel 3D fully implicit unsteady multiblock CFD code implemented on a Beowulf cluster. In *Parallel CFD 1999*, Williamsburg, USA, 1999.

3. M. Woodgate, K. Badcock and B. Richards. The solution of pitching and rolling delta wings on a Beowulf cluster. In *Parallel CFD 2000*, Trondheim, Norway, 2000.
4. S. Osher and S. Chakravarthy. Upwind schemes and boundary conditions with applications to Euler equations in general geometries. (1983) Journal of Computational Physics, 50:447–481.
5. A. Jameson. Time dependent calculations using multigrid, with applications to unsteady flows past airfoils and wings. AIAA Paper 91-1596, 1991.
6. HPCx capability computing. http://www.hpcx.ac.uk
7. P. Nayyar, G. Barakos, K. Badcock and B. Richards. Analysis and Control of Weapon Bay Flows. NATO RTO AVT-123 Symposium on "Flow Induced Unsteady Loads and the Impact on Military Applications", Budapest, 24-28 April, 2005.
8. D. Rizzetta and M. Visbal. Large-Eddy Simulation of Supersonic Cavity Flowfields Including Flow Control. 32nd AIAA Fluid Dynamics Conference, 2002, AIAA Paper 2003-0778.
9. P.R. Spalart. Strategies for Turbulence Modelling and Simulations. (2000) International Journal of Heat and Fluid Flow, 21:252–263.
10. J. Ross. Cavity Acoustic Measurements at High Speeds. Technical Report DERA/MSS/MSFC2/TR000173, QinetiQ, March 2000.
11. R. Steijl, G. Barakos and K. Badcock. A CFD Framework for Analysis of Helicopter Rotors. AIAA Paper 2005-5124, 17th AIAA CFD Conference, Toronto, 6-9 June, 2005.
12. H. Pomin and S. Wagner. Aeroelastic Analysis of Helicopter Rotor Blades on Deformable Chimera Grids. (2004) J. Aircraft 41(3):577-584.
13. M. Potsdam, W. Yeo and W. Johnson. Rotor Airloads Prediction Using Loose Aerodynamic/Structural Coupling. American Helicopter Society 60th Annual Forum. Baltimore, MD, June 7-10, 2004.