INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2007; **55**:509–536
Published online 17 April 2007 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1455

BILU implicit multiblock Euler/Navier–Stokes simulation for rotor tip vortex and wake convection

Bowen Zhong[1], Scott Shaw[1] and Ning Qin[2,*,†]

[1]*Department of Aerospace Sciences, School of Engineering, Cranfield University, Bedfordshire MK43 0AL, U.K.*
[2]*Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, U.K.*

SUMMARY

In this paper, a block incomplete lower–upper (BILU) decomposition method is incorporated with a multiblock three-dimensional Euler/Navier–Stokes solver for simulation of hovering rotor tip vortices and rotor wake convection. Results of both Euler and Navier–Stokes simulations are obtained and compared with experimental observations. The comparisons include surface pressure distributions and tip vortex trajectories. The comparisons suggest that resolution of the boundary layer is important for the accurate evaluation of the blade surface loading, but is less so for the correct prediction of the vortex trajectory. Numerical tests show that, using Courant–Friedrichs–Lewy (CFL) number of 10 or 30 with the developed BILU implicit scheme can be 6–7 times faster than an explicit scheme. The importance of solution acceleration schemes that increase the permitted time-step is illustrated by comparing the evolving wake structures at different stages of the calculation. In contrast to fixed wing simulations, the extent of the wake structures is shown to require resolution of large physical time. This observation explains the poor performance that is obtained when employing convergence acceleration strategies originally intended for solution of equilibrium problems, such as the multigrid methods. Copyright © 2007 John Wiley & Sons, Ltd.

Received 19 June 2004; Revised 4 December 2006; Accepted 29 December 2006

KEY WORDS: helicopter rotor; tip vortex and wake; vortex trajectory; BILU; convergence acceleration; implicit method; Osher's scheme

1. INTRODUCTION

The aerodynamics of helicopter rotor blades at hover and in low-speed forward flight is dominated by interactions with the trailed wake structures of preceding blades. The induced effects of such

*Correspondence to: Ning Qin, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, U.K.
†E-mail: N.Qin@sheffield.ac.uk

Contract/grant sponsor: UK Overseas Research Studentship Award (ORS)
Contract/grant sponsor: Creanfield University College of Aeronautics Research Scholarship

interactions have a substantial influence on the blade loading, aerodynamic performance, vibration and aero-acoustics of the main rotor. The ability to accurately calculate the influence of the wake generated by an isolated rotor and associated interactions in reasonable time is therefore of great practical importance in understanding and improving blade aerodynamic design.

In hovering flight the rotor wake comprises two key physical elements: a sheet of trailed vorticity associated with spanwise variations of blade loading and a pair of concentrated vortices trailed from the tip and the root of the blade. The vortex sheet is a relatively weak feature of the flow that descends in a tightening helical pattern below the rotor. The root and tip vortices follow contracting helical trajectories below the rotor disc. The tip vortex is generally much stronger than the root vortex due to the significant difference in dynamic head experienced at the tip and the hub. At later wake ages, in the 'far wake', the tip vortex dominates the flow.

The importance of the wake in determining rotor inflow and hence rotor performance, vibration and acoustics has motivated considerable efforts in the development of numerical methods for this flow. However, despite the fact that various methods for the helicopter rotor flowfield simulation have been developed, such as the lifting line/lifting surface theory, methods solving the full potential equations and more recently the Euler and Navier–Stokes equations [1–8], accurate numerical simulation of the rotor flowfield still poses considerable challenge. One of the major problems is the computed vortex wake system diffuses too rapidly due to numerical dissipation. Different techniques, including grid adaptation [9, 10], overset grids [11, 12] and unstructured grids [13, 14], have been developed to improve the accuracy of the results. Typically, large grids have to be used for simulating the evolution and the convection of the tip vortices and the vortical wake.

The problem is further compounded by the evolutionary nature of the flow. While the tip vortices generated by fixed wings and propellers are important in determining the induced velocity field and hence performance, the convection speed of the vortices, which is approximately equal to the free stream velocity, is sufficiently large that the tip vortices evolve rapidly. This behaviour has meant that algorithms originally intended for solution of equilibrium problems, such as multigrid, can be employed to accelerate convergence to the steady state, see, for example, Allen [15]. In contrast, the tip vortices of rotors are convected slowly due to the low inflow velocities induced by the rotating blades. As a consequence of this disparity in time scales, rotor simulations exhibit much slower evolution to a 'steady state' than corresponding fixed wing and propeller calculations and require numerical approaches that emphasize the need to run simulations to large physical times. This explains the poor performance that is typically obtained when using multigrid for this problem, see, for example, [15], and the surprising observation that unsteady time accurate simulations of hovering rotor flows in the inertial frame of reference may be as efficient as calculations employing local time-stepping solution acceleration strategies in a blade-attached non-inertial reference frame, see Allen [16] and Shaw [17].

The aim of the present work is to develop an accurate, robust and efficient multiblock Navier–Stokes solver based on the Osher's approximate Riemann solver and the block incomplete lower and upper decomposition (BILU) technique for the simulation of hovering helicopter rotors, and to gain an improved understanding of the rotor vortical flow. In this paper, an appropriate grid system for simulating the vortical wake flow is generated in order that the vortical wake flow can be simulated with minimum numerical dissipation. Accuracy and efficiency of the code are examined and the importance of solution acceleration schemes that emphasize large time steps is demonstrated by comparing the evolution of the computed flow at different points in the convergence history.

DOI: 10.1002/fld

2. GOVERNING EQUATIONS

The governing equations of unsteady, viscous, iso-thermal flow are the three-dimensional Navier–Stokes equations. These equations are commonly written in the fixed frame of reference of a stationary observer. In this inertial formulation the blade rotates relative to the fixed co-ordinate system and the equations must be solved using a time accurate approach. If the frame of reference is adjusted to that of an observer moving with the rotating blade, the equations can be transformed to a non-inertial system in which the problem is steady.

We assume that the blade frame of reference rotates around the z-axis with an angular velocity $\boldsymbol{\omega} = (0, 0, \Omega)$ and we define the absolute velocity $\mathbf{q} = (u, v, w)$, the relative velocity $\mathbf{q}_\mathrm{r} = (\tilde{u}, \tilde{v}, \tilde{w})$ and the rotational velocity $\mathbf{q}_\Omega = (\hat{u}, \hat{v}, \hat{w}) = \boldsymbol{\omega} \times \mathbf{r}$.

The Reynolds averaged Navier–Stokes equations can be written in terms of absolute variables in the blade reference frame in the following non-dimensional form:

$$\frac{\partial \mathbf{Q}}{\partial t} + \frac{\partial (\mathbf{E}^i + \mathbf{E}^v)}{\partial x} + \frac{\partial (\mathbf{F}^i + \mathbf{F}^v)}{\partial y} + \frac{\partial (\mathbf{G}^i + \mathbf{G}^v)}{\partial z} = \mathbf{H} \tag{1}$$

where $\mathbf{Q}$ is the vector of conservative variables, $\mathbf{E}$, $\mathbf{F}$, $\mathbf{G}$ are the Cartesian flux vectors, and $\mathbf{H}$ is a source term. The vector $\mathbf{Q}$ is defined as follows:

$$\mathbf{Q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix} \tag{2}$$

and the corresponding Cartesian variable flux vectors are given by

$$\mathbf{E}^i = \begin{pmatrix} \rho\tilde{u} \\ \rho u\tilde{u} + p \\ \rho v\tilde{u} \\ \rho w\tilde{u} \\ \rho E\tilde{u} + pu \end{pmatrix}, \quad \mathbf{F}^i = \begin{pmatrix} \rho\tilde{v} \\ \rho u\tilde{v} \\ \rho v\tilde{v} + p \\ \rho w\tilde{v} \\ \rho E\tilde{v} + pv \end{pmatrix}, \quad \mathbf{G}^i = \begin{pmatrix} \rho\tilde{w} \\ \rho u\tilde{w} \\ \rho v\tilde{w} \\ \rho w\tilde{w} + p \\ \rho E\tilde{w} + pw \end{pmatrix} \tag{3}$$

where $\tilde{u} = u - \hat{u}$, $\tilde{v} = v - \hat{v}$, $\tilde{w} = w - \hat{w}$, and

$$\mathbf{E}^v = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ \theta_x \end{pmatrix}, \quad \mathbf{F}^v = \begin{pmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ \theta_y \end{pmatrix}, \quad \mathbf{G}^v = \begin{pmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ \theta_z \end{pmatrix} \tag{4}$$

DOI: 10.1002/fld

The source term due to the contribution of the centrifugal forces and the Coriolis forces, when the system rotates around the z-axis, becomes the following:

$$\mathbf{H} = \begin{pmatrix} 0 \\ \rho\Omega v \\ -\rho\Omega u \\ 0 \\ 0 \end{pmatrix} \tag{5}$$

The viscous stress tensor is given by

$$\tau_{ij} = 2\mu S_{ij} + \lambda \frac{\partial u_k}{\partial x_k}\delta_{ij} \tag{6}$$

with the strain rate

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial u_j} + \frac{\partial u_j}{\partial x_i}\right) \tag{7}$$

where μ is the molecular viscosity, λ the second coefficient of viscosity which equals to $-\frac{2}{3}\mu$ when Stoke's hypothesis is applied, and δ_{ij} is the Kronecker symbol.

For turbulent viscous flows the coefficient of viscosity μ is computed as the sum of $\mu_l + \mu_t$, where the laminar viscosity, μ_l, is estimated from temperature T through the Sutherland's law and the turbulent viscosity, μ_t, is evaluated using the Baldwin–Lomax algebraic eddy viscosity model [18].

3. NUMERICAL ALGORITHM

The computational domain is divided into multiple blocks. Each block has six faces, with its own co-ordinate system (ξ, η, ζ) in the computational space. The grid for each block is then generated.

The conservation laws expressed by Equation (1) are re-written in an integral formulation and solved using a finite-volume scheme. For each local discretized finite-volume cell labelled by (i, j, k) the discretized form of Equation (1) becomes

$$\frac{\partial}{\partial t}\left(V_{i,j,k}\mathbf{Q}_{i,j,k}\right) + \tilde{\mathbf{E}}_{i+1/2,j,k} - \tilde{\mathbf{E}}_{i-1/2,j,k} + \tilde{\mathbf{F}}_{i,j+1/2,k} - \tilde{\mathbf{F}}_{i,j-1/2,k} + \tilde{\mathbf{G}}_{i,j,k+1/2} - \tilde{\mathbf{G}}_{i,j,k-1/2} + V_{i,j,k}\mathbf{H}_{i,j,k} = 0 \tag{8}$$

The convective fluxes at the cell interfaces are evaluated using Osher's approximate Riemann solver [19] given as, for example:

$$\tilde{\mathbf{E}}_{i+1/2,j,k} = \frac{1}{2}[\overline{\mathbf{E}}(\mathbf{Q}^-, \mathrm{d}\mathbf{S}_{i+1/2,j,k}) + \overline{\mathbf{E}}(\mathbf{Q}^+, \mathrm{d}\mathbf{S}_{i+1/2,j,k})] - \frac{1}{2}\int_{\mathbf{Q}^-}^{\mathbf{Q}^+} \left|\frac{\partial \overline{\mathbf{E}}}{\partial \mathbf{Q}}\right| \mathrm{d}\mathbf{Q} \tag{9}$$

DOI: 10.1002/fld

where

$$\overline{\mathbf{E}}(\mathbf{Q}, \mathrm{d}\mathbf{S}_{i+1/2,j,k}) = (\mathbf{i}\cdot\mathrm{d}\mathbf{S}_{i+1/2,j,k})\mathbf{E} + (\mathbf{j}\cdot\mathrm{d}\mathbf{S}_{i+1/2,j,k})\mathbf{F} + (\mathbf{k}\cdot\mathrm{d}\mathbf{S}_{i+1/2,j,k})\mathbf{G} \tag{10}$$

The $\mathbf{Q}^-$ and $\mathbf{Q}^+$ values are calculated using a modified MUSCL [20] scheme proposed by Prince *et al.* [21] to tackle the numerical phantom vorticity in Euler solutions on highly stretched grids. The scheme uses the primitive variable vector $\mathbf{q} = (\rho, u, v, w, T)^{\mathrm{T}} \cdot \mathbf{q}^L$ and $\mathbf{q}^R$ are the primitive forms of $\mathbf{Q}^-$ and $\mathbf{Q}^+$, respectively, which can be written as

$$\mathbf{q}^L_{i+1/2} = \mathbf{q}_i + \frac{s}{\dfrac{d_i + d_{i-1}}{d_i} + \dfrac{d_i + d_{i+1}}{d_i}} \left[\left(\frac{d_i + d_{i-1}}{2d_i} + \kappa s \right) \overline{\Delta}_+ + \left(\frac{d_i + d_{i+1}}{2d_i} - \kappa s \right) \overline{\Delta}_- \right] \tag{11a}$$

$$\mathbf{q}^R_{i+1/2} = \mathbf{q}_{i+1} - \frac{s}{\dfrac{d_{i+1} + d_i}{d_{i+1}} + \dfrac{d_{i+1} + d_{i+2}}{d_{i+1}}} \left[\left(\frac{d_{i+1} + d_i}{2d_{i+1}} - \kappa s \right) \overline{\overline{\Delta}}_+ + \left(\frac{d_{i+1} + d_{i+2}}{2d_{i+1}} + \kappa s \right) \overline{\overline{\Delta}}_- \right] \tag{11b}$$

where

$$\overline{\Delta}_+ = \frac{2d_i(\mathbf{q}_{i+1} - \mathbf{q}_i)}{d_{i+1} + d_i}, \quad \overline{\Delta}_- = \frac{2d_i(\mathbf{q}_i - \mathbf{q}_{i-1})}{d_i + d_{i-1}}$$

$$\overline{\overline{\Delta}}_+ = \frac{2d_{i+1}(\mathbf{q}_{i+2} - \mathbf{q}_{i+1})}{d_{i+2} + d_{i+1}}, \quad \overline{\overline{\Delta}}_- = \frac{2d_{i+1}(\mathbf{q}_{i+1} - \mathbf{q}_i)}{d_{i+1} + d_i}$$

and $d_{i-1}, d_i, d_{i+1}, d_{i+2}$ are the cell sizes of the cells $(i-1, j, k)$, (i, j, k), $(i+1, j, k)$, $(i+2, j, k)$ in i direction, respectively. The parameter κ determines the spatial accuracy of the interpolation scheme, while the parameter s represents a limiter, which is used to eliminate spurious oscillations close to discontinuities. The value of κ was set to $\frac{1}{3}$ in the present calculations, which results in a third-order upwind scheme for the convective part of the equations. The s value is found using the formulation of Anderson *et al.* [22] written as

$$s = \frac{2\Delta_+\mathbf{q}\Delta_-\mathbf{q} + \varepsilon}{(\Delta_+\mathbf{q})^2 + (\Delta_-\mathbf{q})^2 + \varepsilon} \tag{12}$$

where ε is a small constant preventing division by zero in regions of null gradients.

In order to improve the robustness of the resulting method, a transformation from the primitive variables $\mathbf{q} = [\rho, u, v, w, p]^{\mathrm{T}}$ to the characteristic variables $\tilde{\mathbf{q}} = [c, U, V, W, z]^{\mathrm{T}}$ is made. The transformation between the Cartesian velocities (u, v, w) and the contravariant velocities (U, V, W) is as follows:

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \mathbf{M} \begin{pmatrix} u - \hat{u} \\ v - \hat{v} \\ w - \hat{w} \end{pmatrix} \tag{13}$$

DOI: 10.1002/fld

where

$$\mathbf{M} = \begin{pmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{pmatrix} \tag{14}$$

and $\hat{u}$, $\hat{v}$, $\hat{w}$ are the Cartesian components of the rotational velocity for the hovering rotor.

For the calculation of a fixed wing in a steady flow, the values of $\hat{u}$, $\hat{v}$, $\hat{w}$ are set to zero. For unsteady flow calculation with a moving grid, $\hat{u}$, $\hat{v}$, $\hat{w}$ correspond to the grid speeds $\mathrm{d}x/\mathrm{d}t$, $\mathrm{d}y/\mathrm{d}t$ and $\mathrm{d}z/\mathrm{d}t$, respectively.

From the finite-volume method point of view, the inviscid flux in i direction can be calculated as

$$\mathbf{E}^i = \begin{bmatrix} \rho U \\ \rho U u + g_1 p \\ \rho U v + g_2 p \\ \rho U w + g_3 p \\ U(\rho E + p) + (g_1 \hat{u} + g_2 \hat{v} + g_3 \hat{w}) p \end{bmatrix} \tag{15}$$

It should be noted that, due to the changes in the convective terms, for a hovering rotor calculation with an absolute velocity formulation, the eigenvalues of the flux matrix are changed to $\lambda_1 = \lambda_2 = \lambda_3 = u - \hat{u}$, $\lambda_4 = u - \hat{u} + c$ and $\lambda_5 = u - \hat{u} - c$. Hence, the positions of the sonic points, which are important in the Osher's scheme, are changed. The transformation (13) including the rotational speed is crucial to ensure a correct implementation.

As mentioned by some researchers [3, 23], in order that the general geometrical identities deduced from the conservation laws can be satisfied, the geometric quantities involved in the numerical flux calculation have to be properly calculated. An area moment vector is introduced for each cell face as follows:

$$\mathbf{M} = \int_{\partial V} \mathbf{r} \times \mathbf{n} \, \mathrm{d}S \tag{16}$$

If this vector is normalized by cell face area

$$\overline{\mathbf{M}} = \mathbf{M}/S \tag{17}$$

the normal contravariant velocity should be evaluated as follows:

$$U = g_1 \cdot u + g_2 \cdot v + g_3 \cdot w - \boldsymbol{\omega} \cdot \overline{\mathbf{M}} \tag{18}$$

The calculation of the inviscid flux requires only that the normal velocity be accurately evaluated while the tangential contravariant velocities can be determined arbitrarily, as long as the transformation and the inverse transformation can be properly carried out. Hence, instead of using the

DOI: 10.1002/fld

transformation of Equation (13), the following transformation will be used:

$$\begin{pmatrix} U \\ V \\ W \end{pmatrix} = \mathbf{M} \begin{pmatrix} u \\ v \\ w \end{pmatrix} - \begin{pmatrix} \boldsymbol{\omega} \cdot \overline{\mathbf{M}} \\ 0 \\ 0 \end{pmatrix} \tag{19}$$

and the inverse transformation is

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \mathbf{M}^{-1} \begin{pmatrix} U + \boldsymbol{\omega} \cdot \overline{\mathbf{M}} \\ V \\ W \end{pmatrix} \tag{20}$$

The flux in i direction can then be calculated as

$$\mathbf{E}^i = \begin{bmatrix} \rho U \\ \rho U u + g_1 p \\ \rho U v + g_2 p \\ \rho U w + g_3 p \\ U(\rho E + p) + p \cdot \boldsymbol{\omega} \cdot \overline{\mathbf{M}} \end{bmatrix} \tag{21}$$

where g_1, g_2, and g_3 are the normals of the surface where the flux needs to be calculated.

3.1. Temporal discretization

After the spatial discretization, the Navier–Stokes equations can be written in a semi-discretized form of Equation (8), which can be further expressed as

$$V \frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} = -\mathbf{R}(\mathbf{Q}) \tag{22}$$

where the right-hand side term is called the residual. The subscripts i, j, k have been dropped here for convenience.

Although we only consider steady flow in this paper, it is efficient to use time marching to drive the residual towards a steady-state converged solution. The solution reaches the steady state when the residual becomes zero.

There are commonly two methods available, namely the explicit method and the implicit method to integrate Equation (22) in time.

3.2. Explicit method

One possible explicit temporal discretization of Equation (22) is

$$V \frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t} = -\mathbf{R}(\mathbf{Q}^n) \tag{23}$$

DOI: 10.1002/fld

The update of the dependent variables is straightforward since the residual vector only depends on the current value of the vector $\mathbf{Q}$.

$$\mathbf{Q}^{n+1} = \mathbf{Q}^n - \frac{\Delta t}{V}\mathbf{R}(\mathbf{Q}^n) \tag{24}$$

For unsteady flow calculations global time accuracy is required and the time step that may be used for each cell is limited by the maximum allowed by stability at any point in the solution domain. However, for steady flow calculation, time accuracy is not required and a time step based on local stability analysis may be used to accelerate convergence towards the steady-state solution. Due to the numerical stability reasons, the time step is restricted for the explicit method. Hence, the explicit method is simple to implement but not efficient.

3.3. Implicit method

In an implicit method, the residual vector is evaluated at the future time level and hence the discretization of the equation becomes

$$V\frac{\mathbf{Q}^{n+1} - \mathbf{Q}^n}{\Delta t} = -\mathbf{R}(\mathbf{Q}^{n+1}) \tag{25}$$

where the spatial indexing (i, j, k) is omitted for simplicity. Employing a first-order linearization of the flux residual $\mathbf{R}(\mathbf{Q}^{n+1})$ Equation (25) can be written as the following system of linear equations:

$$\left(\frac{V}{\Delta t} + \frac{\partial \mathbf{R}(\mathbf{Q}^n)}{\partial \mathbf{Q}}\right)\Delta \mathbf{Q}^n = -\mathbf{R}(\mathbf{Q}^n) \tag{26}$$

In the calculation, the primitive variables $\mathbf{q} = [\rho, u, v, w, p]^{\mathrm{T}}$ are used. With this change of variables the implicit scheme becomes

$$\left(\frac{V}{\Delta t}\frac{\partial \mathbf{Q}}{\partial \mathbf{q}} + \frac{\partial \mathbf{R}(\mathbf{Q}^n)}{\partial \mathbf{q}}\right)\Delta \mathbf{q}^n = -\mathbf{R}(\mathbf{Q}^n) \tag{27}$$

Once this linear system is solved, the update is simply

$$\mathbf{q}^{n+1} = \mathbf{q}^n + \Delta \mathbf{q}^n \tag{28}$$

3.4. Structure of flux Jacobian matrix

The linear system that must be solved at each time step can be written in the form

$$\left(\frac{V}{\Delta t}\frac{\partial \mathbf{Q}_l}{\partial \mathbf{q}_l} + \frac{\partial \mathbf{R}_l^n}{\partial \mathbf{q}_l}\right)\Delta \mathbf{q}_l^n = -\mathbf{R}_l^n \tag{29}$$

where $l = 1, 2, \ldots, (in-1) \times (jn-1) \times (kn-1)$.

DOI: 10.1002/fld

The linear system will be of the following form:

$$\left(\frac{V}{\Delta t} \begin{bmatrix} \frac{\partial \mathbf{Q}_1}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{Q}_1}{\partial \mathbf{q}_2} & \cdots & \cdots & \frac{\partial \mathbf{Q}_1}{\partial \mathbf{q}_l} \\ \frac{\partial \mathbf{Q}_2}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{Q}_2}{\partial \mathbf{q}_2} & \cdots & \cdots & \frac{\partial \mathbf{Q}_2}{\partial \mathbf{q}_l} \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ \frac{\partial \mathbf{Q}_l}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{Q}_l}{\partial \mathbf{q}_2} & \cdots & \cdots & \frac{\partial \mathbf{Q}_l}{\partial \mathbf{q}_l} \end{bmatrix} + \begin{bmatrix} \frac{\partial \mathbf{R}_1}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{R}_1}{\partial \mathbf{q}_2} & \cdots & \cdots & \frac{\partial \mathbf{R}_1}{\partial \mathbf{q}_l} \\ \frac{\partial \mathbf{R}_2}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{R}_2}{\partial \mathbf{q}_2} & \cdots & \cdots & \frac{\partial \mathbf{R}_2}{\partial \mathbf{q}_l} \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ \frac{\partial \mathbf{R}_l}{\partial \mathbf{q}_1} & \frac{\partial \mathbf{R}_l}{\partial \mathbf{q}_2} & \cdots & \cdots & \frac{\partial \mathbf{R}_l}{\partial \mathbf{q}_l} \end{bmatrix} \right) \begin{bmatrix} \Delta \mathbf{q}_1 \\ \Delta \mathbf{q}_2 \\ \vdots \\ \vdots \\ \Delta \mathbf{q}_l \end{bmatrix} = - \begin{bmatrix} \mathbf{R}_1 \\ \mathbf{R}_2 \\ \vdots \\ \vdots \\ \mathbf{R}_l \end{bmatrix} \tag{30}$$

In the above equation, only the right-hand side carries the physics of the problem. Provided the solution has converged, $\Delta \mathbf{q} = 0$, the Jacobians do not have to be computed accurately as the terms on the left-hand side are always equal to zero. In the calculation, the Jacobian of residual vector $\partial \mathbf{R}^n_{i,j,k}/\partial \mathbf{q}_{i,j,k}$ on the left-hand side of the equation will only involve first-order accuracy. It means that the MUSCL scheme is not used there and the residual vector $\mathbf{R}^n_{i,j,k}$ on the left-hand side is a function of the state variables of the neighbouring seven stencils:

$$\mathbf{R}^n_{i,j,k} = f(\mathbf{q}_{i,j,k}, \mathbf{q}_{i+1,j,k}, \mathbf{q}_{i-1,j,k}, \mathbf{q}_{i,j+1,k}, \mathbf{q}_{i,j-1,k}, \mathbf{q}_{i,j,k+1}, \mathbf{q}_{i,j,k-1}) \tag{31}$$

Using first-order accuracy, the difficulty of evaluating the Jacobians and the computer memory requirements are reduced dramatically.

If (l, m, n) is not part of the above seven stencils $\partial \mathbf{R}^n_{i,j,k}/\partial \mathbf{q}_{l,m,n} = 0$. The Jacobian matrix becomes a block-banded matrix of seven bands of blocks.

3.5. Evaluation of flux Jacobians

The flux Jacobian can be obtained using a finite difference approximation. However, it is computationally expensive. In the code, a more efficient method proposed by Sperijse [23] is employed. Badcock [24, 25] employed this method for Navier–Stokes solution and Shaw and Qin [26, 27] implemented it in an implicit solver for solving the parabolized Navier–Stokes equations. The procedure has been described in Shaw and Qin [26, 27].

In Sperijse's method, the chain rule is used to obtain the flux Jacobians analytically:

$$\frac{\partial \mathbf{E}}{\partial \mathbf{q}_i} = \frac{\partial \mathbf{E}}{\partial \mathbf{q}_0} \frac{\partial \mathbf{q}_0}{\partial \mathbf{q}_i} + \frac{\partial \mathbf{E}}{\partial \mathbf{q}_1} \frac{\partial \mathbf{q}_1}{\partial \mathbf{q}_i} \tag{32}$$

where $\mathbf{q}$ is the value of the primitive variables $[\rho, u, v, w, p]^{\mathrm{T}}$ at the cell centre, $\partial \mathbf{E}/\partial \mathbf{q}_0$ and $\partial \mathbf{E}/\partial \mathbf{q}_1$ are the Jacobians of the flux vector with respect to the left and right states used in the approximate Riemann solver. The terms $\partial \mathbf{q}_0/\partial \mathbf{q}_i$ and $\partial \mathbf{q}_1/\partial \mathbf{q}_i$, arising from the MUSCL interpolation of the primitive variables, are the derivative of the state variables with respect to the variables on the computational stencils. The expressions of $\partial \mathbf{E}/\partial \mathbf{q}_0$ and $\partial \mathbf{E}/\partial \mathbf{q}_1$ can be deduced from Osher's approximate Riemann solver [23].

The whole procedure for calculating the convective flux Jacobian can be found in [8, 26, 27].

DOI: 10.1002/fld

The diffusive flux Jacobians can be evaluated relatively straightforwardly since the fluxes are calculated using central difference scheme. The procedure can be found in [26, 27].

3.6. Block incomplete lower–upper method

We can see that the implicit method results in a sparse linear system (27), which can be expressed in a more general form as

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{33}$$

As mentioned before, only the right-hand side of the above equation carries the physics of the problem, hence, matrix $\mathbf{A}$ on the left-hand side does not have to be accurate. The matrix $\mathbf{A}$ is approximated by a BILU decomposition method:

$$\mathbf{A} \approx \mathbf{L}\mathbf{U} \tag{34}$$

where $\mathbf{L}$ is a lower triangular matrix whilst $\mathbf{U}$ is an upper triangular matrix. The matrices $\mathbf{L}$ and $\mathbf{U}$ are required to keep the same pattern as the lower and upper parts of the original matrix $\mathbf{A}$, respectively, and each non-zero element in $\mathbf{A}$ is equal to the element of the product $\mathbf{LU}$ in the same location. Hence, $\mathbf{LU}$ can only be an approximation of the original matrix $\mathbf{A}$.

Instead of solving the original linear system (33), we solve the following system:

$$\mathbf{L}\mathbf{U}\mathbf{x} = \mathbf{b} \tag{35}$$

Since the matrices $\mathbf{L}$ and $\mathbf{U}$ are triangular, their inversions are much simpler. The linear system can then be solved as follows:

$$\mathbf{x} = \mathbf{U}^{-1}\mathbf{L}^{-1}\mathbf{b} \tag{36}$$

4. GRID GENERATION FOR ROTOR

Caradonna and Tung's 2-bladed model helicopter rotor [28] is selected for the computation. The rotor is constructed by constant NACA0012 airfoil sections, untwisted and untapered with an aspect ratio of 6. C-H cylindrical-type grids were generated using the commercial software Gridgen [29].

The grid topology is shown in Figure 1. The grids at the periodic, top and bottom planes are shown. As the grid points on both periodic planes are identical, there is no need to do any interpolation in enforcing the periodic boundary conditions. The grid is split into 12 blocks for the calculations. Figure 2 shows the block boundaries and the interfaces. The angles at upstream and downstream were chosen to be unequal so that the C-type grid at the upstream region is easier to control.

The fine grid used here has 321 grid points in the wraparound chordwise direction with 241 grids on the body, 63 points in the normal direction, and 65 points in the spanwise direction with 35 points on the blade surface. The coarse grid used in the calculation is of $161 \times 32 \times 65$ points, obtained by eliminating every second line of the fine mesh in the chordwise and normal directions. The grid points along spanwise direction were kept the same as that of the fine grid. Figure 3 shows the grids and the close-up view of the grids near the blade surface at section $k = 26$. It shows that the grids near the wall boundary are well controlled.

DOI: 10.1002/fld

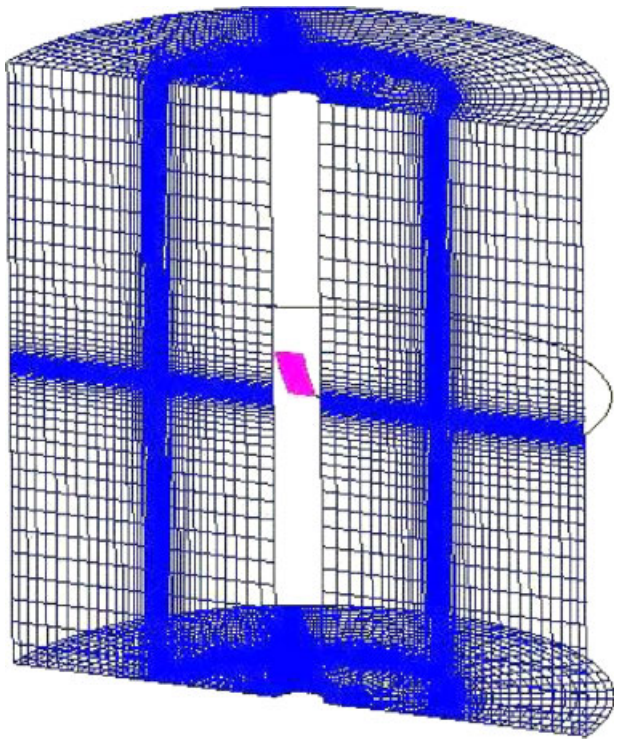

Figure 1. Grids at the periodic, top and bottom boundaries for the C-H cylindrical grid.

5. BOUNDARY AND INITIAL CONDITIONS

Part of the boundary conditions used for rotor calculations are similar to those for wing calculations. However, due to the rotating system, differences in the velocity formulations of the governing equations, grid topologies and methodologies used for the rotor calculations and boundary conditions become quite complicated and required particular attention. In general, the boundary conditions for the rotor calculations can also be divided into two major categories: (1) physical boundary, e.g. inflow, outflow, body surface, periodic boundary and (2) block interface boundaries. In the first category, the physical boundary conditions are applied. In the far-field boundaries, non-reflecting boundary conditions based on Riemann invariants are used. For the absolute velocity formulation, the unperturbed velocity is zero for hovering rotor. At the plane containing the blade root, a slip condition similar to the inviscid flow wall condition is implemented. The treatment of the second category of boundary conditions has no difference as those applied in the inertial system.

6. NUMERICAL RESULTS

All the calculations are based on the application of the multiblock implicit code. Euler and Navier–Stokes simulations with C-H cylindrical grid are performed on one subsonic and two transonic

DOI: 10.1002/fld

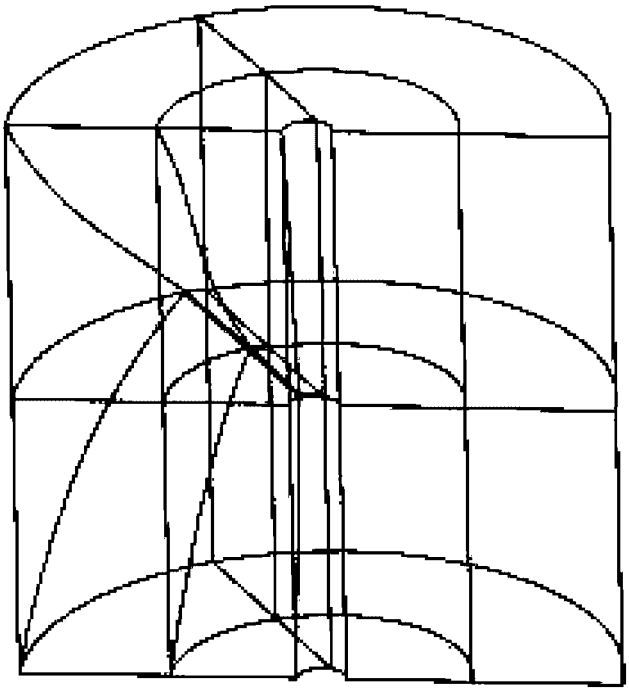

Figure 2. The block boundaries and interfaces for the C-H cylindrical grid.

lifting cases:

(1) Tip Mach number = 0.44, collective pitch angle = 8°, $Re = 1.9$ million
(2) Tip Mach number = 0.794, collective pitch angle = 8°, $Re = 3.5$ million
(3) Tip Mach number = 0.877, collective pitch angle = 8°, $Re = 3.9$ million

Figure 4 shows the evolution and the convection of the vortex wake system of the Euler solution in the case that the tip Mach number is 0.794. Figure 4(a) shows the tip vortex is formed on the upper surface of the blade and the tip vortices from this blade and the preceding blade go underneath the blade. Figure 4(b) shows the whole vortex wake system including a vortex sheet formed behind the blade. Figure 4(c)–(f) shows the interaction and the evolution of the tip vortices and vortex sheet. It shows that the tip vortex core can be identified up to about 400° of vortex age. Beyond this, the tip vortex cores are diffused forming a weak vortex under the rotor disk. These figures also show that the strength of the vortex sheet diffused even quicker than the tip vortices. This is because the grid is even coarser in the regions where the vortex sheet is located, hence the numerical dissipation is also larger there.

Figure 5 shows the convergence history for Euler calculation with the C-H cylindrical coarse grid. Figure 5(a) shows the comparison of the convergence histories using an explicit scheme with a Courant–Friedrichs–Lewy (CFL) number of 0.15 and an implicit scheme with a CFL number of 10. It is found that the computational speed-up is about an order. Figure 5(b) shows the comparison of the convergence histories using an implicit scheme with CFL numbers 10, 30 and 50. It shows that quicker converged solution can be obtained with a CFL number 30. However, with a CFL number

DOI: 10.1002/fld

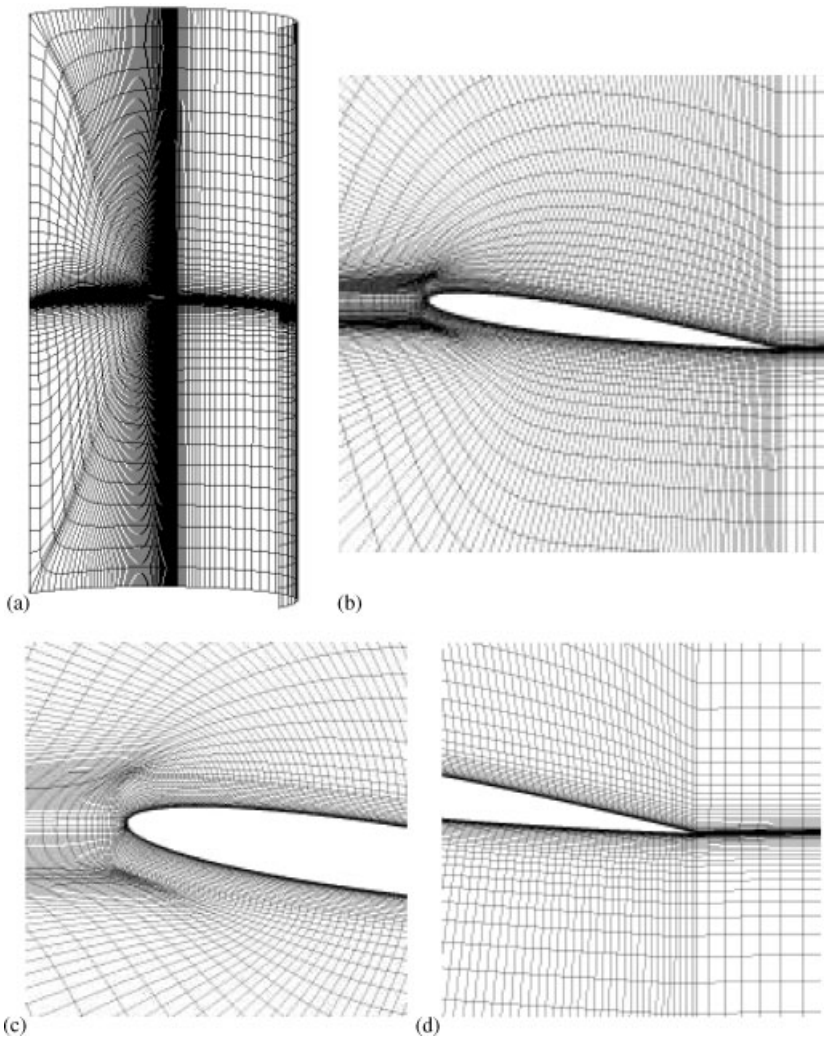

Figure 3. (a) The grid of $k = 26$ for the C-H cylindrical grid; (b) close-up view of the grid at $k = 26$; (c) close-up view of the grid near the leading edge of the airfoil at $k = 26$; and (d) close-up view of the grid near the trailing edge of the airfoil at $k = 26$.

50 the residual may go up at a certain point and a converged solution may not be achievable. These figures also show that, for this subsonic case with a coarse grid, a converged solution can be obtained in less than 30 h CPU time with a DEC Alpha EV6 workstation.

Figure 6 shows the convergence history for the transonic tip Mach number case (Mach number 0.877). Figure 6(a) shows again that substantial improvement has been achieved with the developed BILU implicit scheme when compared to the explicit scheme. Figure 6(b) shows the comparison of the convergence histories with CFL number of 10, 30 and 50. It shows that converged solutions can be obtained using CFL number of 10 and 30. However, for this case, the convergence rate

DOI: 10.1002/fld

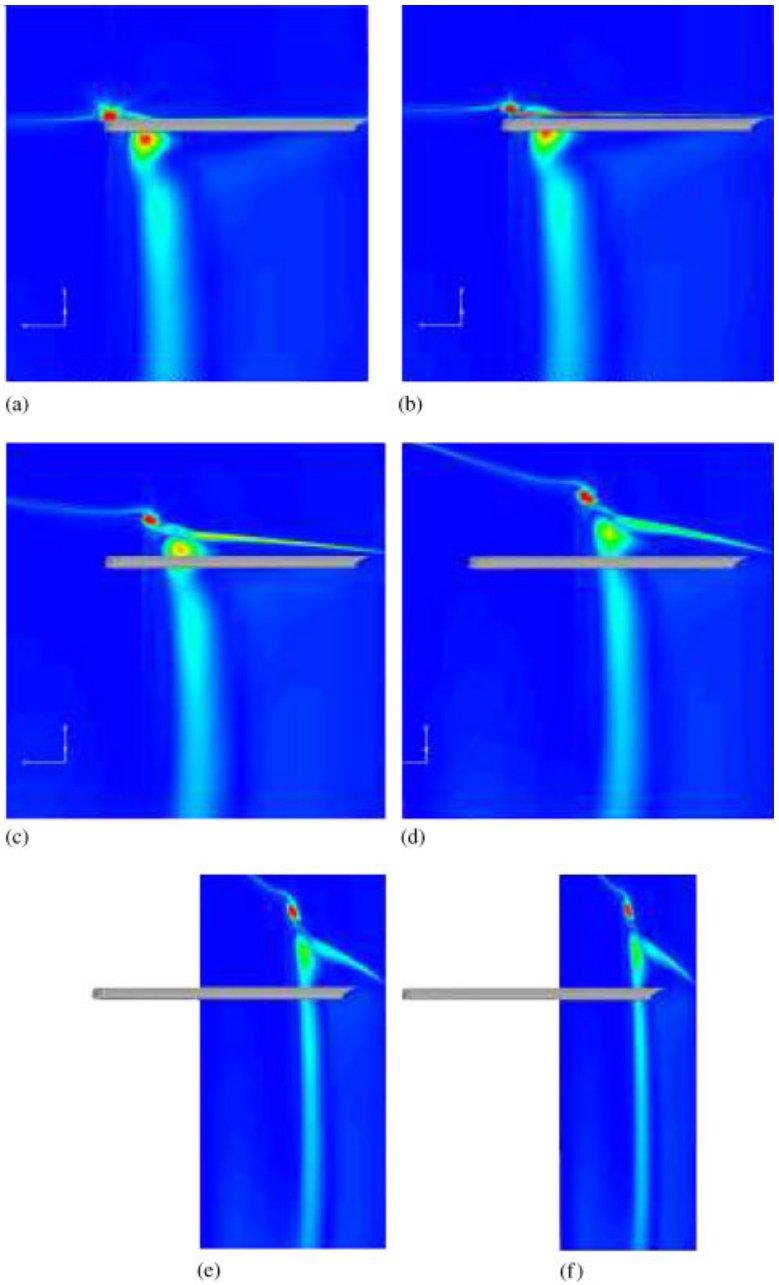

Figure 4. Tip vortex shadings with C-H cylindrical grid (collective pitch angle = 8°, Mtip = 0.794, Euler solution): (a) $\psi = 6.9°$; (b) $\psi = 11.2°$; (c) $\psi = 28.9°$; (d) $\psi = 50.1°$; (e) $\psi = 72.4°$; and (f) $\psi = 78.2°$.

DOI: 10.1002/fld

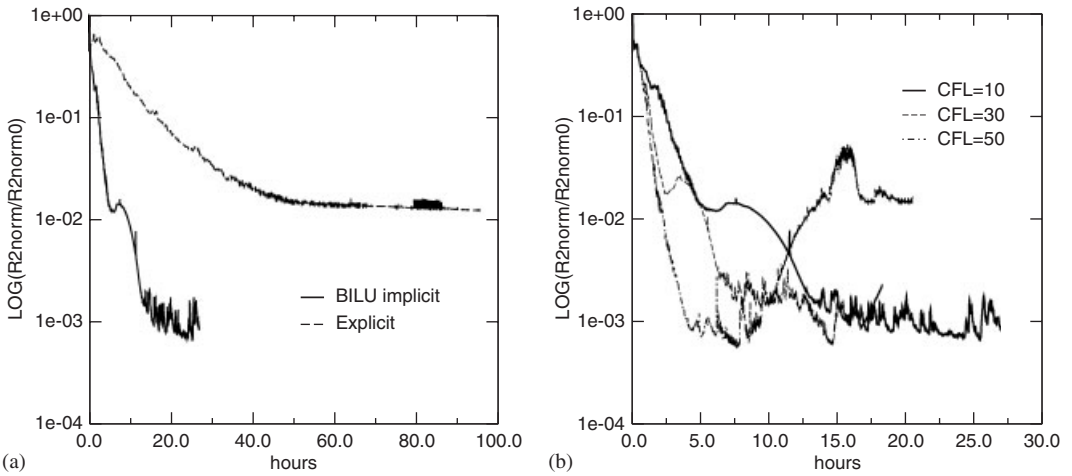

Figure 5. Convergence history with C-H cylindrical coarse grid (collective pitch angle = 8°, Mtip = 0.44, Euler solution).

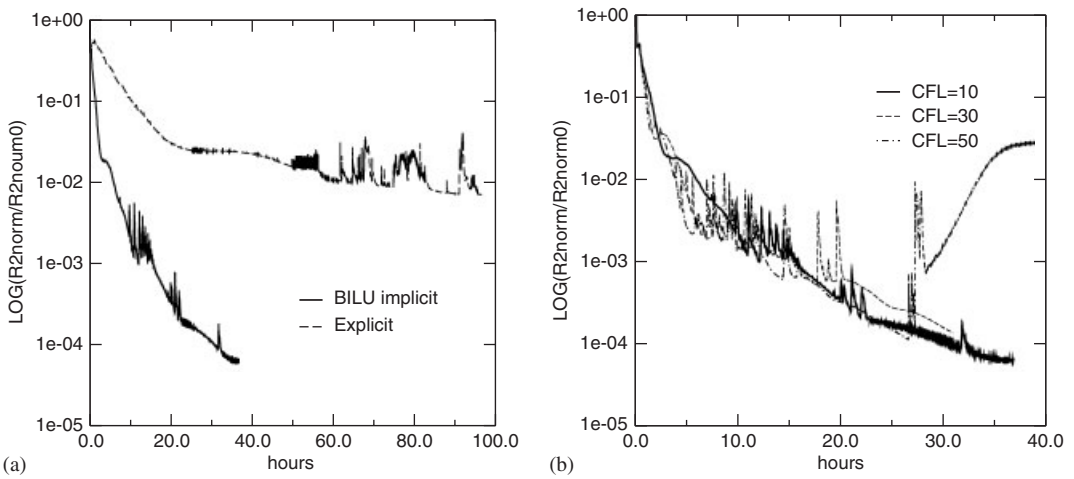

Figure 6. Convergence history with C-H cylindrical coarse grid (collective pitch angle = 8°, Mtip = 0.877, Euler solution).

with CFL number 30 is not any quicker than that with a CFL number of 10. The residual with a CFL number of 50 has gone up at a certain point and may possibly cause the process to blow up eventually. These figure show that a converged solution can be obtained within 40 h CPU time of a DEC Alpha machine.

Figure 7 shows the convergence history with a C-H cylindrical fine grid when the BILU implicit scheme is employed with a CFL number of 10 in the case that the tip Mach number is 0.877. It

DOI: 10.1002/fld

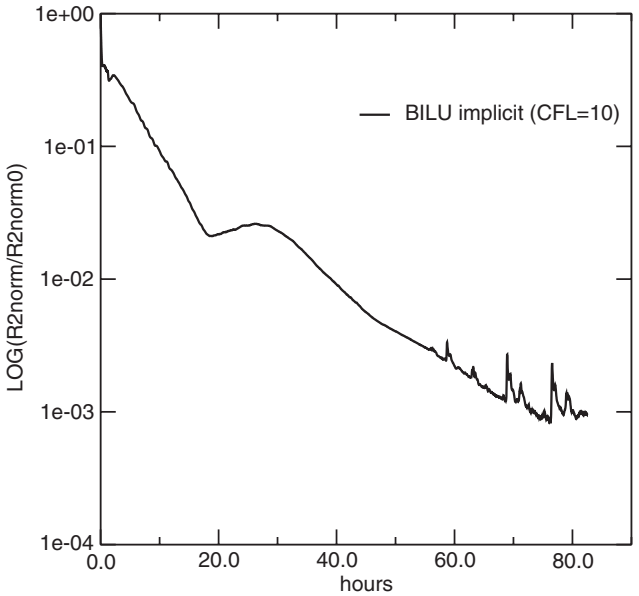

Figure 7. Convergence history with C-H cylindrical fine grid (collective pitch angle = 8°, Mtip = 0.877, Euler solution).

shows that, for a fine grid with over 1.2 million grid points, a converged Euler solution can be obtained with 80 h CPU time.

Figure 8 shows the convergence history with a fine grid in the case that the tip Mach number is 0.794. The jump shown in the middle is due to a restarted process with changes of the parameters used.

Figure 9 gives the comparison of the calculated trajectory of the tip vortex up to 360° of vortex age with the experimental data provided by Caradonna and Tung [28] and the results obtained by using Kocurek and Tangler's [30] prescribed wake model in the case that the tip Mach number is 0.44. It shows that, up to about a hundred degrees of vortex age, the vertical descent of the tip vortex matches the experimental data very well, while the contraction of the tip vortex follows the data obtained using prescribed wake model instead of the experimental data. Similar phenomena can be observed in the case that the tip Mach number is 0.794, which is shown in Figure 10. Kocurek and Tangler's prescribed wake model is an empirical model based upon a large database of sub-scale rotor experiments. The apparent discrepancy between the experiment of Caradonna and Tung and the prescribed wake model suggests that there may be problems with their experimental method. In the experiment, a suction device was employed in the wind tunnel to prevent wake recirculation. This device may have introduced a small vertical velocity that would change the trajectory of the vortex. There is insufficient information in experimental report to develop an appropriate boundary condition to represent this effect in the computational simulation.

Figure 11 shows comparisons of the pressure distributions obtained solving Navier–Stokes and Euler equations in the case that the tip Mach number is 0.44 with a C-H cylindrical fine

DOI: 10.1002/fld

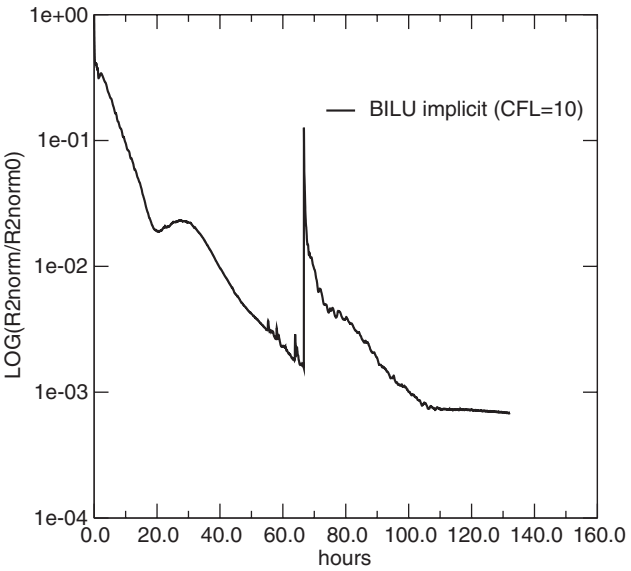

Figure 8. Convergence history with C-H cylindrical fine grid (collective pitch angle = 8°, Mtip = 0.794, Euler solution).

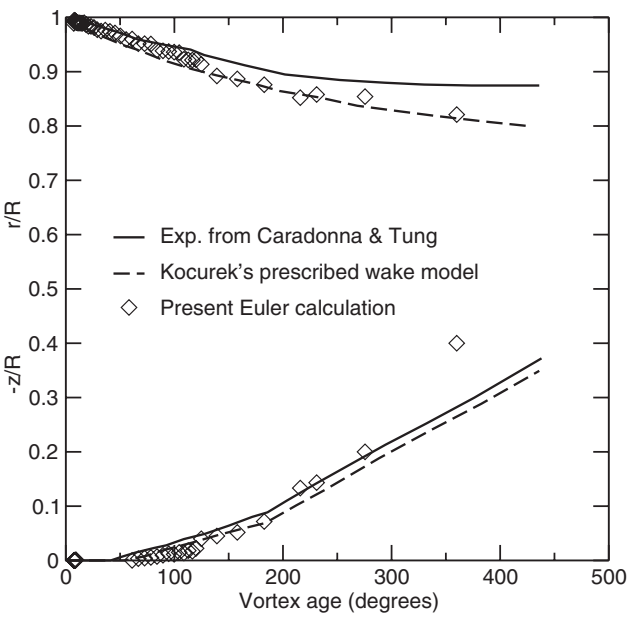

Figure 9. Tip vortex age with C-H cylindrical fine grid (collective pitch angle = 8°, Mtip = 0.44, Euler solution).

DOI: 10.1002/fld

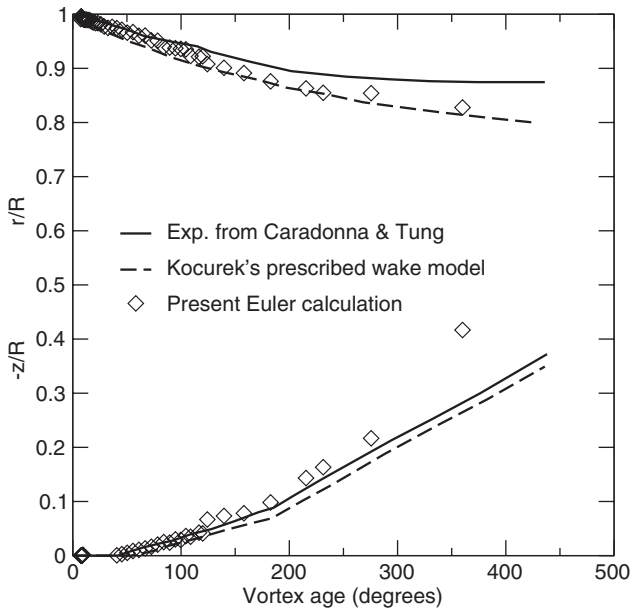

Figure 10. Tip vortex age with C-H cylindrical fine grid (collective pitch angle = 8°, Mtip = 0.794, Euler solution).

grid. It shows that the results obtained by solving Navier–Stokes equations match the experimental data very well whilst the Euler results do not. Figures 12 and 13 show the pressure distributions of Navier–Stokes solutions compared with Euler solutions and experimental data at tip Mach numbers of 0.794 and 0.877, respectively. The differences between Euler and Navier–Stokes results can clearly be identified. Overall, the Navier–Stokes solutions give much better results than the Euler solutions, especially at those stations where strong shock waves appear. The shock positions and strengths predicted by solving the Navier–Stokes equations are particularly good when compared to that obtained from Euler solutions. It shows that Euler solutions predicted stronger shocks and the positions of these shocks are too far away from the leading edge. These figures also show that the Baldwin–Lomax model can handle the mild shock-induced separations well, which appeared in the cases that the tip Mach numbers are 0.794 and 0.877.

Figure 14 shows the evolution and convection of the vortex wake system obtained from the Navier–Stokes solution in the case that the tip Mach number is 0.794. Compared to the results obtained from the Euler solution (Figure 4), the vortex wake system is diffused quicker in the Navier–Stokes solution, despite the fact that the near wake in the Navier–Stokes solution appears to be much stronger.

Figure 15 gives the vortex trajectories of the Navier–Stokes solution in the case where the tip Mach numbers is 0.44. The vertical descent of the tip vortex in the Navier–Stokes solution matches the experimental data very well before 300° of vortex age. Compared to the Euler solution (Figure 9), the vertical descent of the tip vortex of the Navier–Stokes solution gives better agreement with the experimental data than that in Euler solution. The contractions of the tip vortices in

DOI: 10.1002/fld

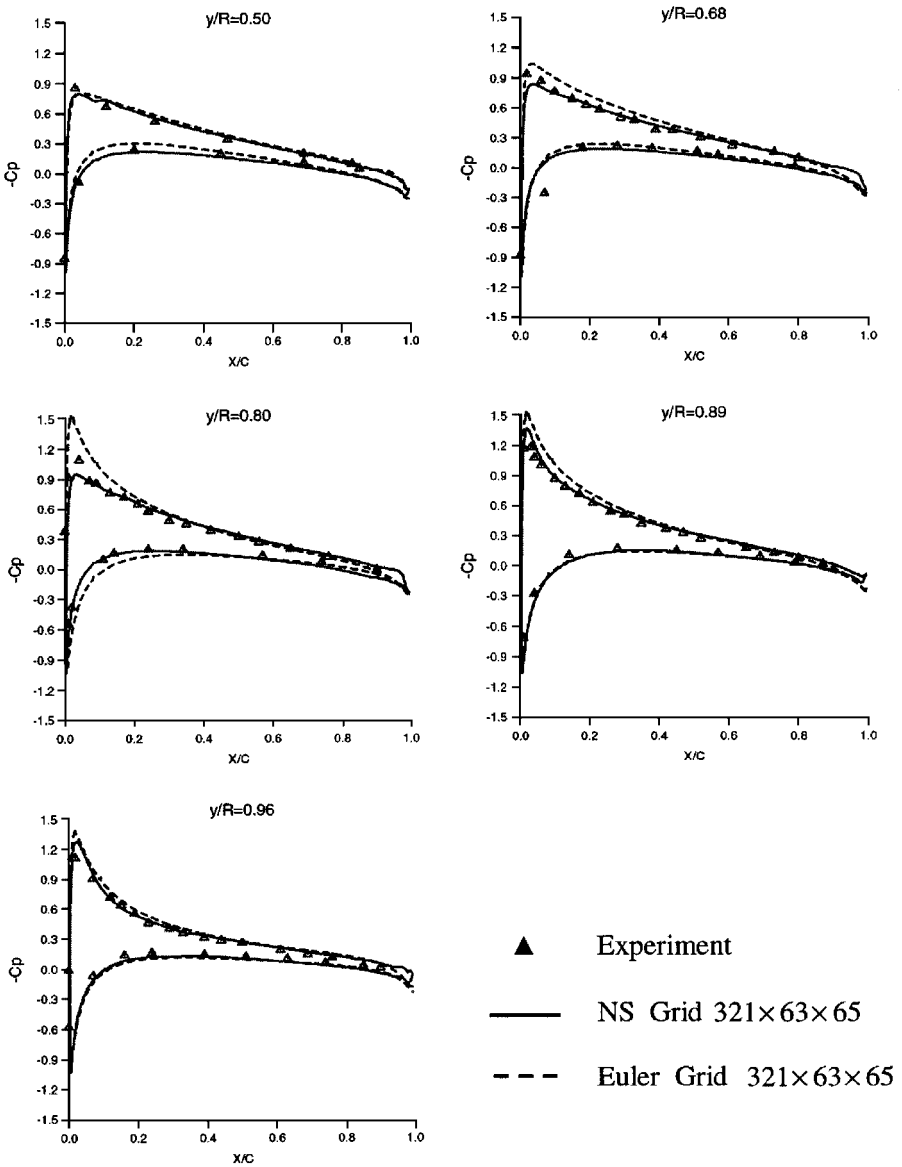

Figure 11. Surface pressure distributions with C-H cylindrical grid (collective pitch angle = 8°, Mtip = 0.44).

Navier–Stokes and Euler solutions are quite similar. It is interesting to note that, instead of matching the experimental data, the contractions seem to follow the data obtained from Kocurek and Tangler's [30] prescribed wake model.

Figure 16 shows the vortex trajectories for the case that the tip Mach number is 0.794. In this case, the vertical descent of the tip vortices in the Navier–Stokes solution gives good agreement

DOI: 10.1002/fld

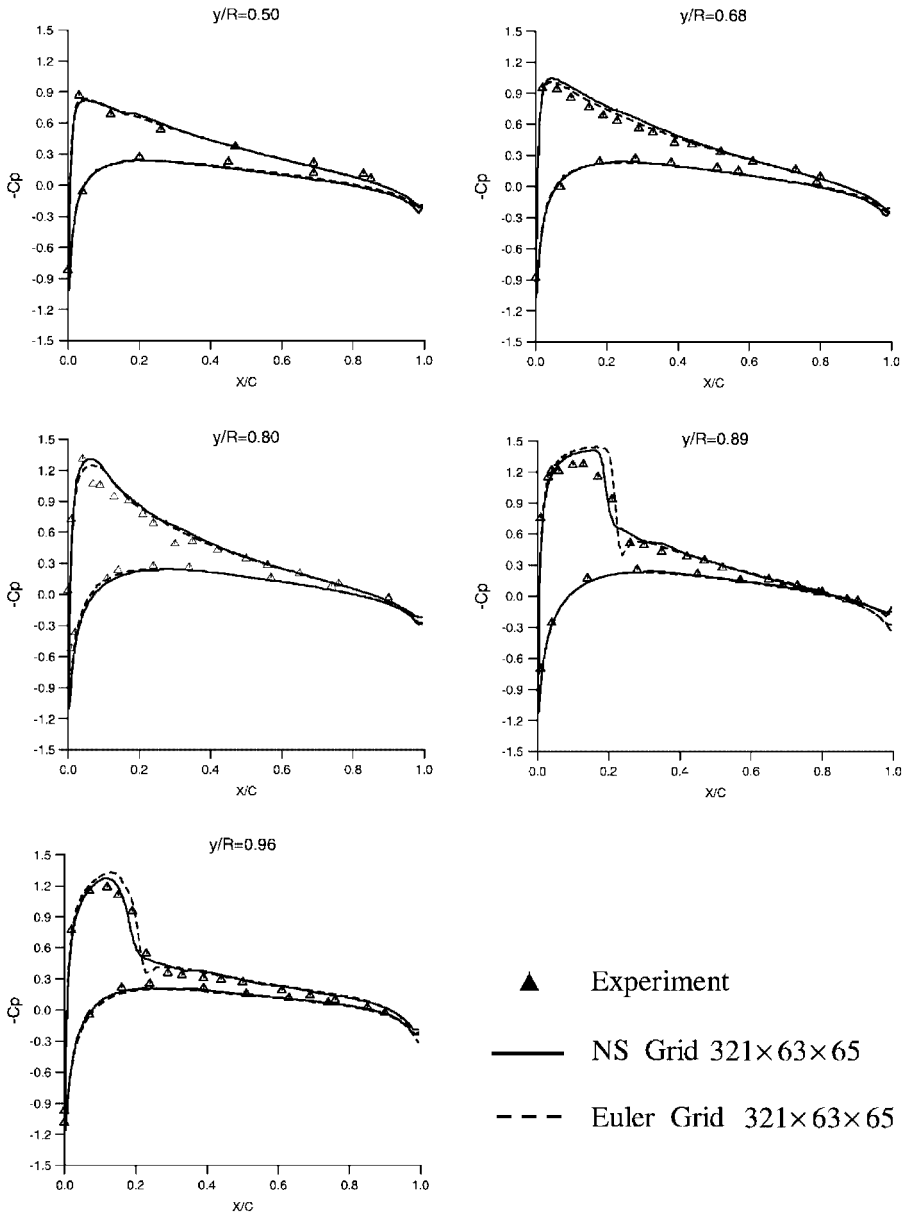

Figure 12. Surface pressure distributions with C-H cylindrical grid (collective pitch angle = 8°, Mtip = 0.794).

with experiment data before 300° of vortex age and is slightly better than that in the Euler solution, see Figure 10. The contraction in the Navier–Stokes solution appears to be slightly stronger than that in the Euler solution.

DOI: 10.1002/fld

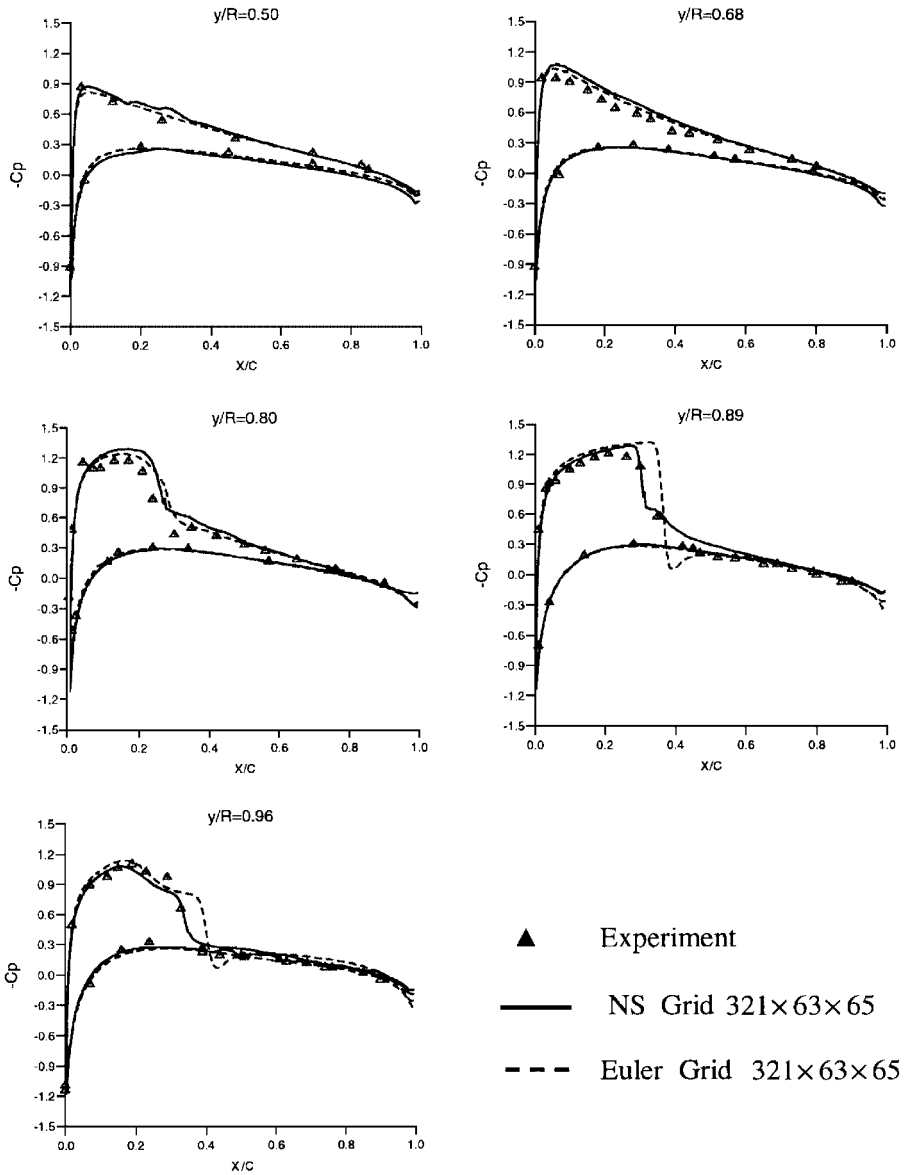

Figure 13. Surface pressure distributions with C-H cylindrical grid (collective pitch angle = 8°, Mtip = 0.877).

The importance of solution acceleration schemes that emphasize large time steps can be demonstrated by considering the solution at different stages of the convergence history. This is done in Figures 17 and 18, which present visualizations of the computed wake structure using iso-surfaces of vorticity, for the case with a tip Mach number of 0.794 and a collective

DOI: 10.1002/fld

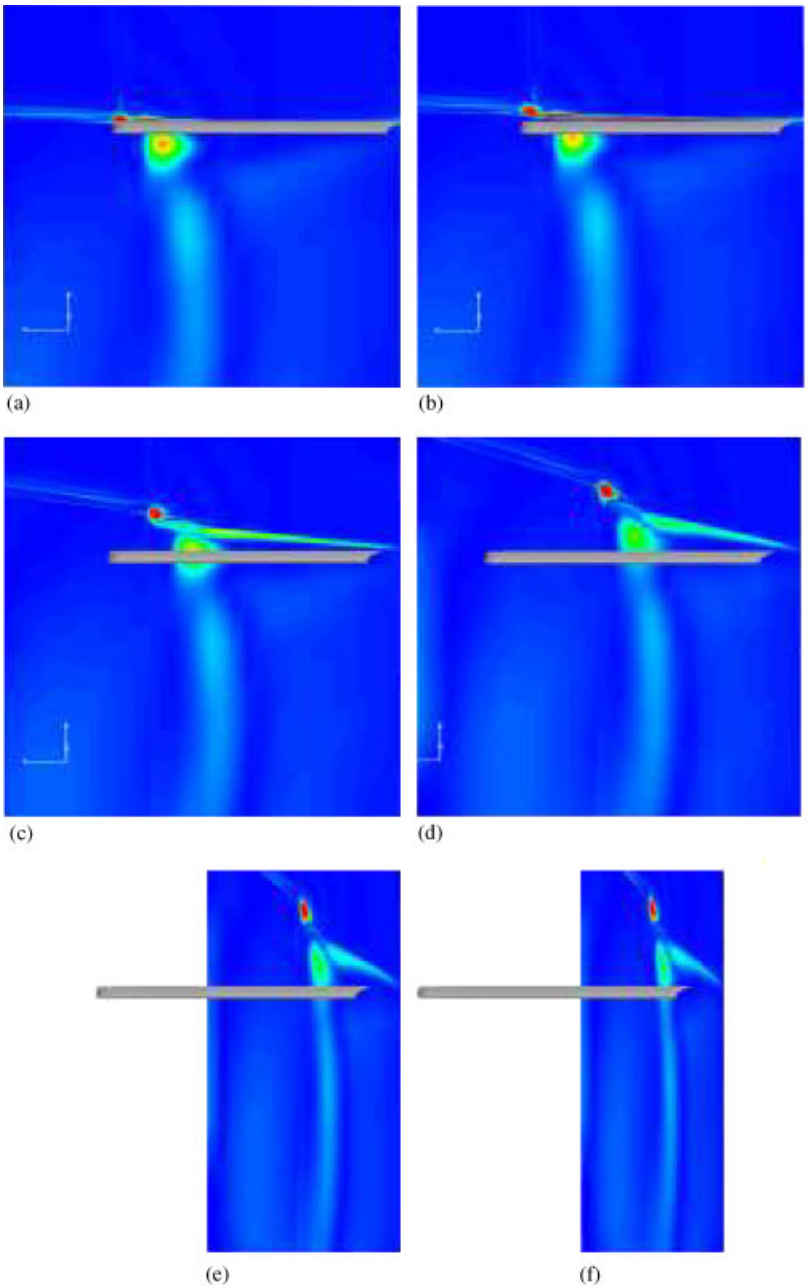

Figure 14. Tip vortex shadings with C-H cylindrical finer grid (collective pitch angle = 8°, Mtip = 0.794, NS solution): (a) $\psi = 6.9°$; (b) $\psi = 11.2°$; (c) $\psi = 28.9°$; (d) $\psi = 50.1°$; (e) $\psi = 72.4°$; and (f) $\psi = 78.2°$.

DOI: 10.1002/fld

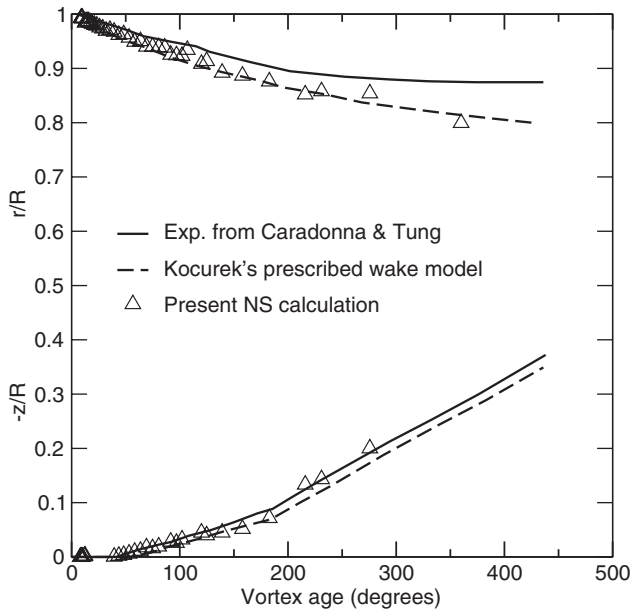

Figure 15. Tip vortex age with C-H cylindrical fine grid (collective pitch angle = 8°, Mtip = 0.44, NS solution).

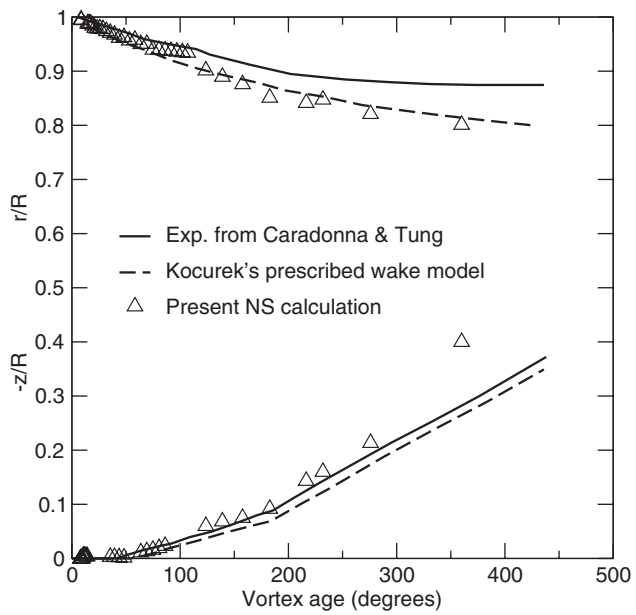

Figure 16. Tip vortex age with C-H cylindrical fine grid (collective pitch angle = 8°, Mtip = 0.794, NS solution).

DOI: 10.1002/fld

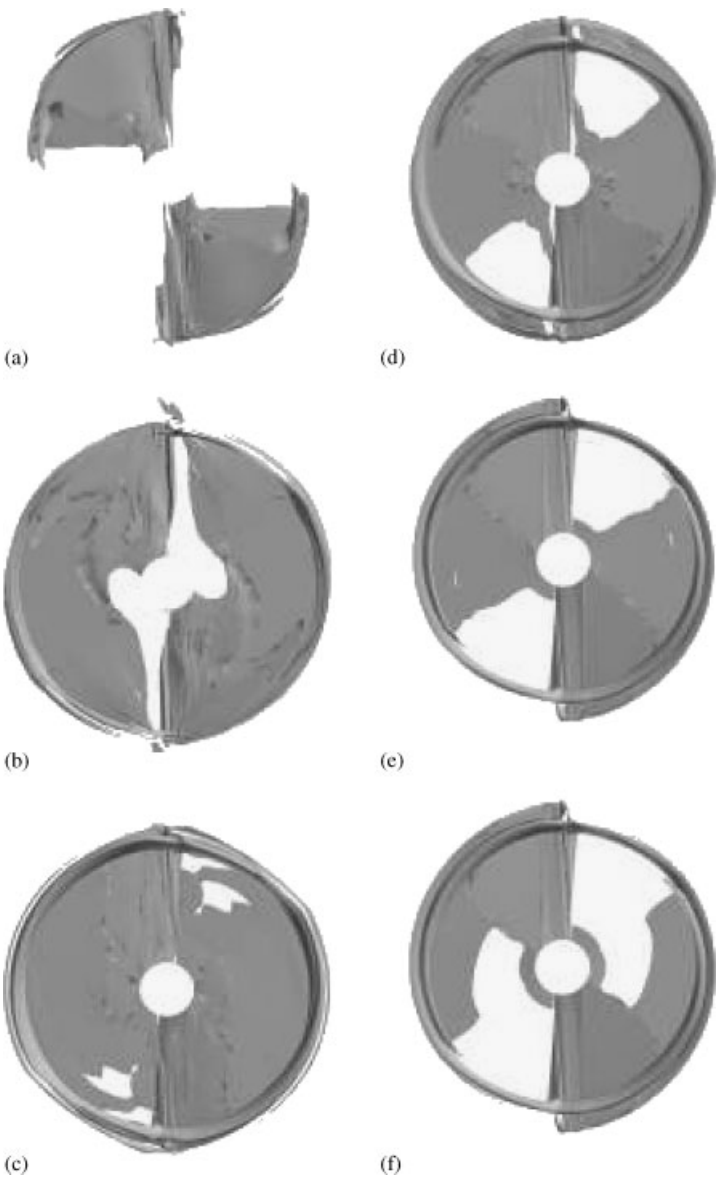

Figure 17. Evolution of the tip vortex with iteration number: (a) iteration number = 500; (b) iteration number = 1000; (c) iteration number = 2000; (d) iteration number = 3000; (e) iteration number = 5000; and (f) iteration number = 7000.

pitch angle of 8°. In this case, a CFL number of 10 is used. Early in the convergence history very little wake structure is captured. There is some evidence of the wake sheet and the tip vortex is resolved to approximately 90° wake age. After a further 500 iterations the tip vortex has developed further, now extending to a wake age corresponding to 180°. There is also

DOI: 10.1002/fld

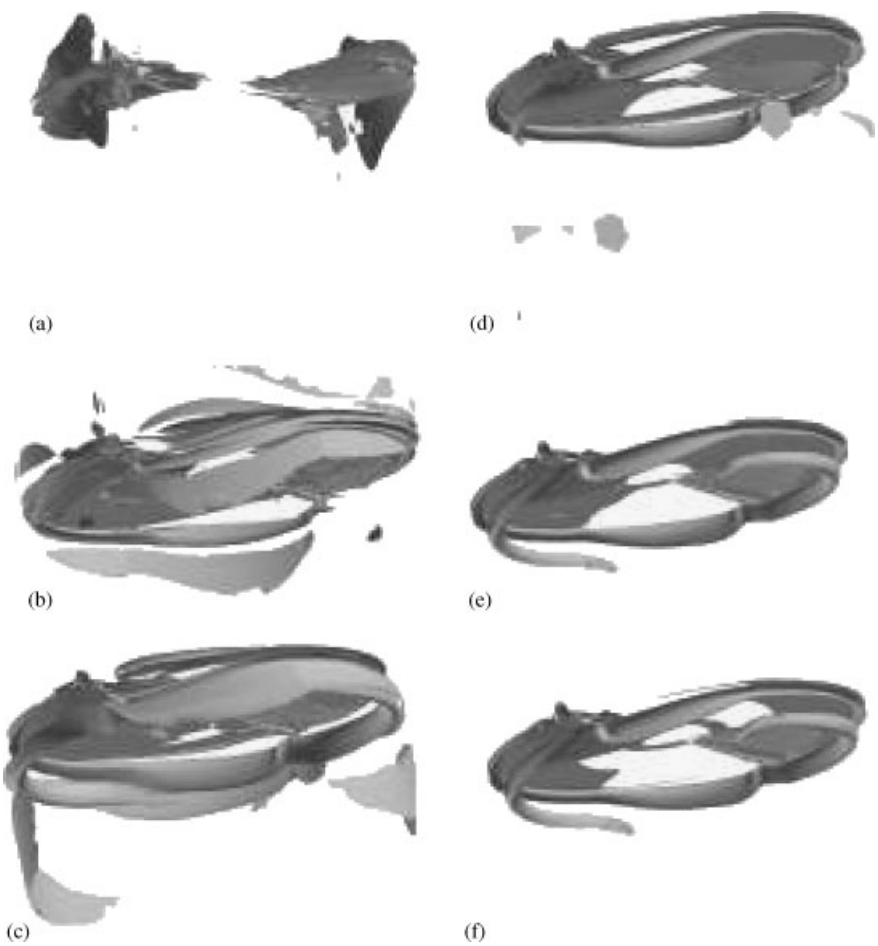

Figure 18. Evolution of the tip vortex with iteration number: (a) iteration number = 500; (b) iteration number = 1000; (c) iteration number = 2000; (d) iteration number = 3000; (e) iteration number = 5000; and (f) iteration number = 7000.

evidence of considerable noise in the solution, which is thought to be due to the use of local time stepping.

As the calculation proceeds further, more of the wake structure is revealed and the noise in the solution is reduced as the calculation converges. By 5000 iterations the visualized tip vortex extends to approximately 300° of wake age, and the wake structure does not appear to evolve significantly when further iterations are performed. Beyond this wake age numerical viscosity due to discretization error appears to overwhelm the computed flow structures causing them to be rapidly dissipated. Finer grid should demonstrate this evolutionary nature better as longer tip vortex can be captured by the simulation and longer stimulation time will be required for allowing the tip vortex to evolve.

These results help to explain the earlier observations by Allen [15] concerning the poor performance of the multigrid algorithm for the rotor problem when compared to earlier computations for

DOI: 10.1002/fld

fixed wings. The rapid convection of the tip vortex (at approximately the freestream velocity) is such that the fixed wing calculation shares many of the characteristics of the equilibrium problems for which multigrid was originally developed, allowing the full potential of the method to be developed. For the rotor problem, where the convection speed is related to the inflow velocity, the computation is dominated by the evolutionary nature of the flowfield rendering approaches developed for equilibrium problems ineffective. Indeed Allen [15] reported speed-ups are more consistent with grid sequencing than multigrid. Instead algorithms developed for evolutionary problems, such as implicit time-marching schemes, should be favoured for an efficient solution of a steady-state problem by following a physical process of the flow evolution. Further evidence for this conclusion is provided by the comparisons of the relative costs of time-marching computations performed using local time stepping in a blade-attached non-inertial reference frame and global time stepping in an unsteady calculation in an inertial frame of reference, see, for example, Allen [16] and unpublished work by Shaw [17], which show that there is little benefit in employing local time stepping as a solution acceleration scheme.

7. CONCLUSIONS

A three-dimensional implicit multiblock Navier–Stokes solver for hovering rotor simulations has been developed. The convective fluxes are discretized by using Osher's approximate Riemann solver while the viscous fluxes by a central difference scheme. A modified MUSCL scheme is employed for improving the accuracy of the discretization for the inviscid fluxes. A block incomplete lower and upper decomposition (BILU) is employed to solve the linear system. The accuracy of the solver and the efficiency of the developed BILU implicit scheme have been examined for fixed wing calculations [8]. Numerical tests show that, to achieve a converged solution, the implicit scheme, using a CFL number of 30, can be 6–7 times faster compared to the time spent when using an explicit scheme. For hovering rotor simulation, despite the changes made in the governing equations and the numerical scheme, numerical simulations demonstrate similar improvements with the BILU implicit scheme.

The fidelity of the solutions is demonstrated by comparing results from Euler and Navier–Stokes solutions, including surface pressure distributions and vortex trajectories, with the corresponding experimental measurements. The comparisons suggest that resolution of the boundary layer is important for the accurate evaluation of the blade surface loading, but is less important for the correct prediction of the vortex trajectory.

The importance of discretization approaches that consider the development of the tip vortex was demonstrated by comparing computed data at several points in the convergence history. Such comparisons explain the ineffectiveness of algorithms rooted in equilibrium problems, such as multigrid and observations concerning the use of global and local time stepping for this problem. Efficient computation of flow problems related to hovering rotors requires algorithms, such as the implicit time-marching algorithm presented in this paper, that emphasize the evolutionary nature of the problem.

ACKNOWLEDGEMENTS

The authors acknowledge the contribution from and helpful discussions with David Perigo and Alan Le Moigne of the Centre for Computational Aerodynamics, Cranfield University College of Aeronautics.

DOI: 10.1002/fld

The project was partly supported by the UK Overseas Research Studentship Award (ORS) and Cranfield University College of Aeronautics Research Scholarship.

REFERENCES

1. Hariharan N, Sankar LN. A review of computational techniques for rotor wake modelling. *AIAA Paper 2000-0114*, 2000.
2. Caradonna FX. Developments and challenges in rotorcraft aerodynamics. *AIAA Paper 2000-0109*, 2000.
3. Boniface JC, Pahlke K. Calculations of multibladed rotors in forward flight using 3-D Euler methods of DLR and ONERA. *22nd European Rotorcraft Forum*, Paper No. 58, Brighton, U.K., 1996.
4. Rouzaud O, Raddatz J, Boniface JC. Euler calculations of multibladed rotors in hover by DLR and ONERA methods and comprarison with helishape tests. *American Helicopter Society 53rd Annual Forum*, Virginia Beach, VA, U.S.A., 1997.
5. Pomin H, Wagner S. Navier–Stokes analysis of helicopter rotor aerodynamics in hover and forward flight. *AIAA Paper 2001-0998*, 2001.
6. Allen CB. Convergence of steady and unsteady formulations for inviscid hovering rotor solutions. *International Journal for Numerical Methods in Fluids* 2003; **41**(9):931–949.
7. Zhong B, Qin N. Non-initial multiblock Navier–Stokes calculation for hovering rotor flowfields using relative velocity approach. *Aeronautical Journal* 2001; **105**(1049):379–398.
8. Zhong B, Shaw ST, Qin N. BILU implicit multiblock Euler/Navier–Stokes simulations for helicopter rotor vortical flow. *AIAA Paper 2005-465*, 2005.
9. Tang L, Baeder JD. Improved Euler simulation of hovering rotor tip vortices with validation. *The 55th American Helicopter Society Annual Forum*, Montreal, Canada, 1999.
10. Tramel RW, Sheta EF. Higher order accurate adaptive multiresolution schemes for the Euler equations. *AIAA Paper 2003-3823*, 2003.
11. Ahmad J, Duque EPN. Helicopter rotor blade computation in unsteady flows using moving overset grids. *Journal of Aircraft* 1996; **33**(1):54–60.
12. Benoit C, Jeanfaivre G. Three-dimensional inviscid isolated rotor calculations using chimera and automatic Cartesian partitioning methods. *Journal of the American Helicopter Society* 2003; **48**(2):128–138.
13. Strawn RC, Barth T. Helicopter rotor flows using unstructure grid scheme. *Journal of the American Helicopter Society* 1993; **38**(2):61–67.
14. Ruffin SM, O'Brien D, Smith MJ, Hariharan N, Lee JD, Sankar L. Comparison of rotor–airframe interaction utilizing overset and unstructured grid techniques. *AIAA Paper 2004-0046*, 2004.
15. Allen CB. Multigrid convergence of inviscid fixed- and rotary-wing flows. *International Journal for Numerical Methods in Fluids* 2002; **39**(2):121–140.
16. Allen CB. Convergence of steady and unsteady formulations for inviscid hovering rotor flows. *International Journal for Numerical Methods in Fluids* 2003; **41**(9):931–949.
17. Shaw ST. Unpublished unsteady simulations of hovering rotors, 2005.
18. Baldwin BS, Lomax H. Thin-layer approximation and algebraic model for separated turbulent flows. *AIAA Paper 78-257*, 1978.
19. Allen CB. Multigrid acceleration of an upwind Euler method for hovering rotor flows. *Aeronautical Journal* 2001; **105**(1051):517–524.
20. Van Leer, B. Upwind difference methods for aerospace problems governed by the Euler equation. *Lecture Notes in Applied Mathematics* 1985; **22**:27–336.
21. Prince SA, Qin N, Ludlow DK. Phantom vorticity in Euler solution on highly stretched grids. *ICAS 2000-0.2*, Harrogate, U.K., 2000.
22. Anderson W, Thomas J, van Leer B. A comparison of finite volume flux vector splitting for the Euler equations. *AIAA Journal* 1986; **24**(9):1453–1460.
23. Sperijse SP. Multigrid solution of the steady Euler equations. *Ph.D. Dissertation*, Centrum voor Wiskunde en Informatica, Amsterdam, 1987.
24. Badcock KJ, Richards BE. Implicit time stepping methods for the Navier–Stokes equations. *AIAA Journal* 1996; **34**(3):555–559.
25. Badcock KJ, Richards BE, Woodgate MA. Elements of computational fluid dynamics on block structured grids using implicit solvers. *Progress in Aerospace Science* 2000; **36**(5–6):351–392.

DOI: 10.1002/fld

26. Shaw ST, Qin N. A preconditioned implicit Krylov-subspace method for the solution of the parabolised Navier–Stokes equations. *Cranfield University COA Report 9706 NFP*, 1997.
27. Shaw ST, Qin N. A preconditioned implicit Krylov-subspace method for the solution of the parabolised Navier–Stokes equations. *AIAA Paper 98-0111*, 1998.
28. Caradonna FX, Tung C. Experimental and analytical studies of a model helicopter rotor in hover. *NASA TM 181232*, 1981.
29. Gridgen User Manual. Pointwise, Inc.: U.S.A., 1997.
30. Kocurek JD, Tangler JL. A prescribed wake lifting surface hover performance analysis. *The 32nd AHS Annual Forum*, Washington, DC, U.S.A., 1976.

DOI: 10.1002/fld