Journal of Computational Physics **178,** 498–519 (2002)
doi:10.1006/jcph.2002.7038, available online at http://www.idealibrary.com on IDEAL®

Stability Analysis of Preconditioned Approximations of the Euler Equations on Unstructured Meshes

P. Moinier and M. B. Giles

Computing Laboratory, Numerical Analysis Group, Oxford University, Wolfson Building, Parks Road, Oxford OX1 3QD, United Kingdom
E-mail: moinier@comlab.ox.ac.uk

Received September 28, 2000; revised October 1, 2001

This paper analyses the stability of a discretisation of the Euler equations on 3D unstructured grids using an edge-based data structure, first-order characteristic smoothing, a block-Jacobi preconditioner, and Runge–Kutta timemarching. This is motivated by multigrid Navier–Stokes calculations in which this inviscid discretisation is the dominant component on coarse grids.

The analysis uses algebraic stability theory, which allows, at worst, a bounded linear growth in a suitably defined "perturbation energy" provided the range of values of the preconditioned spatial operator lies within the stability region of the Runge–Kutta algorithm. The analysis also includes consideration of the effect of solid wall boundary conditions, and the addition of a low Mach number preconditioner to accelerate compressible flows in which the Mach number is very low in a significant portion of the flow.

Numerical results for both inviscid and viscous applications confirm the effectiveness of the numerical algorithm and show that the analysis provides accurate stability bounds.

1. INTRODUCTION

The motivation for the analysis in this paper was the use of a combination of multigrid and Jacobi preconditioning to accelerate the solution of an approximation of the steady compressible Reynolds-averaged Navier–Stokes equations on unstructured grids. Multigrid is well known to be very effective in accelerating iterative convergence, provided that one has suitable restriction and prolongation operators for transferring the residual and correction, respectively, between different grid levels, and a smoothing operator which eliminated high-frequency errors on each grid level. With unstructured grids, there are difficulties in formulating restriction and prolongation operators of sufficiently high order to satisfy the

0021-9991/02 $35.00

theoretical conditions identified by Hackbusch [11], but the more significant challenge is developing an efficient smoothing operator.

Pierce [22] analysed the behaviour of explicit timemarching methods for the Navier–Stokes equations on structured grids and identified the main problem as being the stiffness in the discrete equations due to the disparity in the propagative speed of convective and acoustic waves. This is greatly exacerbated for high Reynolds number flows due to the highly stretched grids which are required to resolve the flow gradients in the boundary layer. The numerical stiffness in this case is related to the fact that the timescale for viscous diffusion across a high Reynolds number boundary layer is much greater than the timescale for the propagation of a pressure wave across the boundary layer. Using standard explicit solvers restricted by the acoustic timescale leads to very slow convergence for the convection/diffusion of streamwise momentum and temperature. To cope with this problem, Pierce and Giles [23] analysed different combinations of preconditioner and multigrid methods for both inviscid and viscous flow applications. For turbulent Navier–Stokes calculations, a block-Jacobi preconditioner and a semicoarsening multigrid method provides an effective damping of all modes inside the boundary layer. The preconditioner provides effective damping of the convective/diffusive modes, while the multigrid strategy, in which the grids are coarsened only in the direction normal to the boundary layer, ensures that all acoustic modes are eliminated efficiently. Together, very good multigrid convergence rates have been achieved for a range of applications [23].

For unstructured grids, we have followed the same approach, combining block-Jacobi preconditioning with a semicoarsening multigrid strategy [19, 20]. This has been effective in giving good multigrid convergence rates, which led to the present work to analyse its behaviour. With structured grids, one uses Fourier (or von Neumann) analysis, knowing that the eigenmodes of the iteration for small perturbations to a uniform flow on a regular periodic grid are Fourier modes. One can then calculate the corresponding eigenvalues and from these determine the stability of the iterative procedure. With unstructured grids, the spatial variation in the eigenmodes is not known a priori. Furthermore, even if one could compute the eigenvalues of the iterative process, their values can be misleading: there are well-known examples such as the first-order upwinding of the convection equation on a finite 1D domain (e.g., [24]) for which the eigenvalues have modulus less than unity, ensuring eventual exponential decay, but allowing an unacceptable large transient growth due to the extreme nonorthogonality of the eigenvectors.

In the present work, we analyse the stability using a variation of the energy method [25], which relies on the construction of a suitable defined "energy" which can be proven to monotonically decrease. In [9], Giles analyses the semidiscrete and fully discrete Navier–Stokes equations arising from a Galerkin discretisation on a tetrahedral grid. The use of the energy method shows that the semidiscrete equations are stable, with the "energy" decreasing monotonely. The analysis of the fully discrete equations using Runge–Kutta time integration is based on algebraic stability analysis [16], which ensures bounded transient growth before eventual exponential decay. The analysis in this paper uses the same approach, but applied to our edge-based discretisation which is a very common approach to discretising the Navier–Stokes equations on unstructured grids [2, 3, 18, 21]. The inclusion of higher order numerical smoothing and viscous terms in the stability analysis appears too difficult, unfortunately, and so only the first-order discretisation of the Euler equations is considered. However, this is the dominant component of the discretisation of the Navier–Stokes equations on the coarser grids used within the multigrid procedure. Indeed, in addition to the

use of a first-order discretisation on all coarse levels, the semicoarsening strategy applied in the direction normal to the wall and used to generate the multigrid sequence [20] doubles the cell size in the boundary layer from one level to the next. This creates cells for which the viscous terms are small components compared to the convective terms (typically, starting with 20 points across the fine grid boundary layer, this number drops down to 5 for the second level). When convergence of the Navier–Stokes equations is achieved on the fine grid, the conclusions that are drawn from the stability analysis remain therefore pertinent enough to explain the good behaviour of the same calculation using multigrid.

The present analysis also extends the previous analysis of Giles in two important respects. The first is the inclusion of the Jacobi preconditioning which is used, as explained above, to reduce the disparity in propagation speeds and local timescales. The second is the inclusion of low Mach preconditioning, in addition to the Jacobi preconditioning, to accelerate the convergence and improve the accuracy when a significant portion of the flow field has a very low Mach number. Both of these are now becoming common features of compressible Navier–Stokes discretisations on both structured and unstructured grids [6, 15, 27–29, 31].

The paper is organised as follows. Section 2 presents the nonlinear discretisation of the Navier–Stokes equations, and the construction of the Jacobi preconditioner which is based on its linearisation. Section 3 presents the stability analysis for small perturbations from a uniform flow, starting with a periodic grid and then considering solid wall boundary conditions. Section 4 outlines the low Mach number preconditioning and how it is included within the stability analysis. Section 5 has numerical results showing the effectiveness of the multigrid procedure, and that the stability analysis gives stability bounds which are sufficient and almost necessary. Section 6 presents conclusions.

2. NUMERICAL METHOD

2.1. Overview

Writing the 3D compressible Reynolds-averaged Navier–Stokes equations as

$$\frac{\partial \mathcal{Q}}{\partial t} + \nabla \cdot \mathcal{F}(\mathcal{Q}, \nabla \mathcal{Q}) = 0, \tag{1}$$

where $\mathcal{Q}(\mathbf{x})$ is the vector of conserved variables, $(\rho, \rho u, \rho v, \rho w, \rho E)^T$ and $\mathcal{F}(\mathcal{Q}, \nabla \mathcal{Q})$ is the total flux, which can be split into an inviscid flux, $\mathcal{F}^{\boldsymbol{I}}(\mathcal{Q})$, and a viscous flux, $\mathcal{F}^V(\mathcal{Q}, \nabla \mathcal{Q})$,[1] the preconditioned semidiscrete equations appear as

$$P^{-1} \frac{dQ}{dt^*} + R(Q) = 0, \tag{2}$$

with $R(Q)$ being the residual vector of the spatial discretisation and P^{-1} the local preconditioner. It is important to note that the timestep in Eq. (2) is, compared to the timestep in Eq. (1), nondimensional to be consistent with the other terms.

The turbulent viscosity is modeled using the Spalart and Allmaras turbulence model [26], which can be included with the above equations with the addition of a source on the right-hand side of Eq. (1).

[1] In this paper alphabetical letters are used to denote discrete quantities, whereas calligraphic letters are used to denote analytic functions and variables. Boldfaced quantities are vectors in Cartesian coordinates.

The iterative scheme used to converge the discrete residuals to zero is pseudo timestepping using the five-stage Runge–Kutta method developed by Martinelli [17]. This can be expressed as

$$\begin{aligned} Q_j^{(0)} &= Q_j^n \\ Q_j^{(k)} &= Q_j^n - \alpha_k \text{CFL}\ P_j\ R_j^{(k-1)}, \quad k = 1, 2, 3, 4, 5 \\ Q_j^{n+1} &= Q_j^{(5)}, \end{aligned}$$

where

$$\begin{aligned} R_j^{(k-1)} &= C_j\big(Q^{(k-1)}\big) - B_j^{(k-1)} \\ B_j^{(k-1)} &= \beta_k D_j\big(Q^{(k-1)}\big) + (1-\beta_k) B_j^{(k-2)}, \end{aligned}$$

with $C_j(Q^{(k-1)})$ being the convective contribution to R_j, and $D_j(Q^{(k-1)})$ being the remaining parts due to the dissipation, both physical and numerical, and the turbulence model source term.

The block-Jacobi preconditioner is based on a local linearisation of the 3D Navier–Stokes equations and constructed by extracting the terms corresponding to the central node, thereby giving a block-diagonal matrix. As the flux can be split into inviscid and viscous parts, the matrix preconditioner will have contributions coming from both. For the turbulence model there is also a contribution from the linearised source term.

2.2. Inviscid Terms

Using a finite volume approach, the integration of the inviscid flux over some control volume Ω gives, after the application of the divergence theorem,

$$R_j^I = \frac{1}{V_j} \oint_{\partial\Omega} \mathcal{F}^I(\mathcal{Q}) \cdot \boldsymbol{n}\, dS, \tag{3}$$

where V_j is the volume of the control volume associated with node j and $\boldsymbol{n}$ is an outward-pointing unit normal. As explained in [19] the discrete approximation to Eq. (3) is

$$R_j^I = \frac{1}{V_j}\left(\sum_{i \in E_j} F_{ij}^I \Delta s_{ij} + \sum_{k \in B_j} F_k^I \Delta s_k \right), \tag{4}$$

where E_j is the set of all nodes connected to node j via an edge, B_j the set of boundary faces associated with node j (e.g., wall and far-field), and Δs_{ij} and Δs_k are associated face areas. For the edge (i, j), the numerical flux toward node i from node j takes the form [19]

$$\begin{aligned} F_{ij}^I = \frac{1}{2}\Bigg(& (\mathcal{F}^I(Q_i) + \mathcal{F}^I(Q_j)) \cdot \boldsymbol{n}_{ij} \\ & - |A_{ij}|\left(\Psi\,(Q_i - Q_j) - \frac{1}{3}(1-\Psi)(L_i(Q) - L_j(Q)) \right) \Bigg). \end{aligned} \tag{5}$$

Here $\boldsymbol{n}_{ij}$ is the unit vector normal to the face, $A_{ij} = \frac{\partial \mathcal{F}^I}{\partial \mathcal{Q}} \cdot n_{ij}$, and its absolute value $|A_{ij}|$ is defined to be $T|\Lambda|T^{-1}$, with $|\Lambda|$ being the diagonal matrix of absolute eigenvalues, and T

the corresponding matrix of right eigenvectors. $L(Q)$ is a pseudo-Laplacian operator which is normalised to have unit weight with respect to the central node. The switch Ψ is defined to have a value close to zero when the flow is smooth, and close to unity when there is a discontinuity, such as at a shock.

The corresponding contribution to the block-Jacobi preconditioner comes from the linearised form of the flux with the first-order upwinding, which is

$$\frac{1}{2}(A_{ij}Q_i + A_{ij}Q_j - |A_{ij}|(Q_i - Q_j)).$$

Summing the matrices multiplying the value Q_j at the central node gives the following inviscid preconditioner for a grid point which does not lie on the boundary.

$$\left(P_j^I\right)^{-1} = \frac{1}{2V_j}\left(\sum_{i\in E_j}(A_{ij} + |A_{ij}|)\Delta s_{ij}\right).$$

However, because $\oint_{\partial\Omega}\mathcal{F}\cdot\boldsymbol{n}\,dS = 0$ when $\mathcal{F}$ is a constant vector, it follows that $\sum_{i\in E_j}(A_{ij}Q_j)$ is identically zero, and so the first term in the summation above can be dropped. Extending this to include the boundary faces gives the final of the inviscid contribution to the Jacobi preconditioner,

$$\left(P_j^I\right)^{-1} = \frac{1}{2V_j}\left(\sum_{i\in E_j}|A_{ij}|\Delta s_{ij} + \sum_{k\in B_j}|A_k|\Delta s_k\right). \tag{6}$$

2.3. Viscous Terms

The viscous flux is integrated over the same control volume, giving

$$R_j^V = \frac{1}{V_j}\oint_{\partial\Omega}\mathcal{F}^V(\mathcal{Q},\nabla\mathcal{Q})\cdot\boldsymbol{n}\,dS. \tag{7}$$

For interior grid points, this leads to the discrete form

$$R_j^V = \frac{1}{V_j}\sum_{i\in E_j}F_{ij}^V\Delta s_{ij}, \tag{8}$$

where F_{ij}^V is the numerical viscous flux associated with the edge (i, j). On all boundary faces other than walls, it is assumed that the viscous fluxes are negligible. For nodes lying on a stationary wall, the momentum residual is discarded and replaced by the no-slip condition that the velocity is zero. Therefore, there is no need to evaluate the viscous flux contribution from a wall face and so the expression above for the viscous residual is used for all grid points.

The numerical viscous flux is defined as

$$F_{ij}^V = \mathcal{F}^V(\overline{Q_{ij}}, \nabla Q_{ij})\cdot\boldsymbol{n}_{ij}.$$

$\overline{Q_{ij}}$, the value of the flow vector at the middle of the edge, is taken to be a simple average of Q_i and Q_j. The value of ∇Q_{ij} is defined as

$$\nabla Q_{ij} = \frac{Q_j - Q_i}{|x_j - x_i|} s_{ij} + (\overline{\nabla Q_{ij}} - (\overline{\nabla Q_{ij}} \cdot s_{ij}) s_{ij}), \tag{9}$$

where $s_{ij} = (x_j - x_i)/|x_j - x_i|$ and $\overline{\nabla Q_{ij}}$ is the simple average of the values of ∇Q at nodes i and j obtained by the usual application of Green's theorem on the standard control volume. The first term in Eq. (9) determines the component of ∇Q in the direction along the edge by simply differencing the values of Q at the two ends; this is clearly the most accurate and the most stable way to do so, and in a highly stretched boundary layer it leads to a viscous discretisation which has the standard three-point differencing across the boundary layer which one would use with a structured grid. The second term gets the orthogonal components of ∇Q from the average of the gradients at either end of the edge; in highly stretched orthogonal grids this component is insignificant.

The viscous contribution to the preconditioner is obtained by linearising the viscous flux expression on the basis of perturbations to a locally uniform flow, and neglecting the second term in Eq. (9). Extracting the matrix multiplying the value of Q_j at the central node then gives

$$\left(P_j^V\right)^{-1} = \frac{1}{V_j} \sum_{i \in E_j} B_{ij} \Delta s_{ij}, \tag{10}$$

where

$$B_{ij} = \frac{1}{|x_j - x_i|} \frac{\partial F_{ij}^V}{\partial (\nabla Q_{ij})} \cdot s_{ij}.$$

2.4. Turbulence Model

When the Navier–Stokes equations are augmented by one or more turbulence model equations, the preconditioner has to have an additional contribution $(P_j^S)^{-1}$ which is the Jacobian matrix corresponding to the linearisation of the turbulence model source term. The inclusion of this means that the source term is effectively treated in an implicit fashion.

The full matrix preconditioner is then given by

$$P_j^{-1} = \left(P_j^I\right)^{-1} + \left(P_j^V\right)^{-1} + \left(P_j^S\right)^{-1}. \tag{11}$$

2.5. Preconditioner Adjustment at Walls

In inviscid calculations, the component of the velocity normal to the wall boundary is set to zero. This is accomplished by discarding the normal component of the momentum flux residual and modifying the preconditioner so that this results in zero change to the normal velocity component, which is set to zero at the outset.

The modifications to the residual and preconditioner can be expressed using a symmetric projection matrix N which extracts from a general vector Q the normal components of the velocity at the wall [7, 19]. The flow tangency condition is then

$$NQ = 0,$$

and so, trivially,

$$N\frac{dQ}{dt} = 0.$$

The equation for the updating of the rest of the flow solution is

$$(I - N)P^{-1}(I - N)\frac{dQ}{dt} = (I - N)R,$$

which simply states that changes in the other components of the solution are driven by the flux residual R.

Combining these two equations yields

$$((I - N)P^{-1}(I - N) + N)\frac{dQ}{dt} = (I - N)\,R, \tag{12}$$

giving the modified preconditioner on the left-hand side of the equation. It can be shown that this modified preconditioner is of full rank and hence is invertible, so this equation gives a well-defined specification of the evolution of the entire flow solution.

In viscous applications, a similar procedure is followed, except that it is the entire flow velocity which is set to zero at stationary walls due to the no-slip condition.

3. INVISCID STABILITY ANALYSIS

We now examine the stability of the numerical scheme. Defining a suitable "energy" of small perturbations from a uniform flow, the first step is to consider the semidiscrete equations and prove that the "energy" is monotonically decreasing and so the equations are stable. Thereafter, we consider the fully discrete equations and evaluate their stability using the theory of algebraic stability.

Key to the analysis is the use of symmetrising variables (Gustafsson and Sundstrom [10] and Abarbanel and Gottlieb [1]), which yield equations in which the main matrices are all either symmetric or antisymmetric. It is important to note that the same semidiscrete equations are obtained whether one discretises the nonlinear Euler equations using conservative variables, linearises the equations, and transforms them to symmetrised variables, or whether alternatively one first linearises and transforms the equations and then discretises them. This justifies our use of the symmetrising variables since the stability analysis remains valid independent of the choice of variables used to represent the linear equations.

3.1. Semidiscrete Equations

To avoid the complication of analysing the influence of boundary conditions, we first consider the analysis with periodic boundary conditions.

The semidiscrete preconditioned Euler equations with first-order characteristic smoothing are

$$\left(\sum_{i\in E_j}\frac{1}{2}|A_{ij}|\Delta s_{ij}\right)\frac{dQ_j}{dt} = -\sum_{i\in E_j}\frac{1}{2}((\mathcal{F}(Q_i) + \mathcal{F}(Q_i))\cdot \boldsymbol{n}_{ij} - |A_{ij}|(Q_i - Q_j))\Delta s_{ij},$$

where $A_{ij} = \frac{\partial \mathcal{F}}{\partial Q}\cdot \boldsymbol{n}_{ij}$, as defined earlier.

Linearising locally and transforming to the symmetrised variables $U = [dp/\rho c, du, dv, dw, dp - c^2 d\rho]^T$, the resulting equation is

$$\left(\sum_{i \in E_j} \frac{1}{2} |A_{ij}| \Delta s_{ij} \right) \frac{dU_j}{dt} = - \sum_{i \in E_j} \frac{1}{2} (A_{ij}(U_i + U_j) - |A_{ij}|(U_i - U_j)) \Delta s_{ij},$$

where A_{ij} is now a symmetric matrix whose eigenvalues are u_n and $u_n \pm c$, with c being the speed of sound and u_n the component of velocity in the direction normal to the face associated with the edge. The expression of A_{ij} is given in Appendix A.

Considering now the whole mesh, the combined system of o.d.e.'s may be written as

$$\mathbb{P}^{-1} \frac{dU}{dt} = -(\mathbb{A}U + \mathbb{D}U). \tag{13}$$

$\mathbb{P}$, $\mathbb{A}$, and $\mathbb{D}$ are all $N \times N$ block matrices, in which each block is a 5×5 matrix. They can be expressed as

$$\mathbb{P}^{-1} = \sum_{\text{edges}} \mathbb{P}_{ij}^{-1} = \sum_{\text{edges}} \begin{bmatrix} \frac{1}{2}|A_{ij}| \Delta s_{ij} & 0 \\ 0 & \frac{1}{2}|A_{ij}| \Delta s_{ij} \end{bmatrix}, \tag{14}$$

$$\mathbb{A} = \sum_{\text{edges}} \mathbb{A}_{ij} = \sum_{\text{edges}} \begin{bmatrix} 0 & +\frac{1}{2} A_{ij} \Delta s_{ij} \\ -\frac{1}{2} A_{ij} \Delta s_{ij} & 0 \end{bmatrix}, \tag{15}$$

and

$$\mathbb{D} = \sum_{\text{edges}} \mathbb{D}_{ij} = \sum_{\text{edges}} \begin{bmatrix} \frac{1}{2}|A_{ij}| \Delta s_{ij} & -\frac{1}{2}|A_{ij}| \Delta s_{ij} \\ -\frac{1}{2}|A_{ij}| \Delta s_{ij} & \frac{1}{2}|A_{ij}| \Delta s_{ij} \end{bmatrix}, \tag{16}$$

where an edge connects node j to node i and the four block elements correspond to entries (j, j), (j, i), (i, j), and (i, i); all other elements are zero.

The application of an entropy fix [12] to the eigenvalues ensures that each block diagonal is strictly positive definite, and therefore so is $\mathbb{P}^{-1}$.

The diagonal blocks in $\mathbb{A}$ are zero, because $\sum_{i \in E_j} \frac{1}{2} \mathcal{F}(Q_j) \cdot \boldsymbol{n}_{ij} = 0$ due to the closed control volume around node j, as explained earlier. In addition, $\mathbb{A}$ is antisymmetric because each of the blocks A_{ij} is symmetric and so $\mathbb{A}_{ij}$ is antisymmetric.

Concerning the matrix $\mathbb{D}$, the first remark to be made is that since A_{ij} is symmetric, it is diagonalisable by an orthogonal similarity transformation $A_{ij} = T_{ij} \Lambda_{ij} T_{ij}^T$ with $T_{ij}^T T_{ij} = I$. Hence, $|A_{ij}| = T_{ij} |\Lambda_{ij}| T_{ij}^T$ is symmetric, and therefore so are $\mathbb{D}_{ij}$ and the full matrix $\mathbb{D}$. In addition, for any complex vector U and its Hermitian U^*,

$$U^* \mathbb{D} U = \frac{1}{2} \sum_{\text{edges}} (U_i - U_j)^* |A_{ij}| \Delta s_{ij} (U_i - U_j).$$

The matrices $|A_{ij}|$ are real positive definite symmetric, and the quantities Δs_{ij} are positive real numbers. Consequently, $U^* \mathbb{D} U$ is a sum of nonnegative real numbers and therefore $\mathbb{D}$ is positive semidefinite.

Defining the perturbation "energy" as $E = \frac{1}{2}U^*\mathbb{P}^{-1}U$, we find its rate of change to be

$$\begin{aligned}\frac{dE}{dt} &= \frac{1}{2}\left(\frac{dU^*}{dt}\mathbb{P}^{-1}U + U^*\mathbb{P}^{-1}\frac{dU}{dt}\right)\\ &= \frac{1}{2}(-U^*(\mathbb{A}+\mathbb{D})^*U - U^*(\mathbb{A}+\mathbb{D})\,U)\\ &= -U^*\mathbb{D}U\\ &\le 0.\end{aligned}$$

Hence, the energy is nonincreasing. Since $\mathbb{P}^{-1}$ is symmetric and positive definite, this in turn proves the stability of the semidiscrete equation.

3.2. Fully Discrete Equations

Having shown that the semidiscrete equations are stable, we now consider the fully discrete equations using Runge–Kutta time integration.

Starting with Eq. (13) and defining $\mathbb{C} = -(\mathbb{A}+\mathbb{D})$, this can be rearranged by defining a new variable $W = \mathbb{P}^{-1/2}U$ to become

$$\frac{dW}{dt} = \mathbb{P}^{1/2}\mathbb{C}\,\mathbb{P}^{1/2}W.$$

Using Runge–Kutta time integration, the fully discrete equations are

$$W^{(n+1)} = L\left(k\,\mathbb{P}^{1/2}\mathbb{C}\mathbb{P}^{1/2}\right)W^{(n)}, \tag{17}$$

where $L(z)$ is the Runge–Kutta polynomial with stability region S, as defined in Appendix B, and k is the global timestep. After n timesteps Eq. (17) gives

$$W^{(n+1)} = \left(L\left(k\,\mathbb{P}^{1/2}\mathbb{C}\,\mathbb{P}^{1/2}\right)\right)^n W^{(0)}.$$

This is said to be absolute stable if there are no solutions which increase without bound as $n \to \infty$. This requires that none of the eigenvalues of $L(k\,\mathbb{P}^{1/2}\mathbb{C}\,\mathbb{P}^{1/2})$ have magnitude greater than unity (which means that the eigenvalues do not lie outside the stability region S), and any eigenvalues of unit size are simple eigenvalues.

In principle, absolute stability is sufficient to guarantee that the timemarching iteration converges to the solution of the steady equations. However, it is not sufficient in practice because if the matrix is not normal (i.e., its eigenvectors are nonorthogonal) it allows the possibility of a very large transient growth which can lead to arithmetic overflow. We therefore require a form of stability analysis which provides sufficient conditions to eliminate this possibility.

Ideally, one would like to prove strong stability, which using the L_2 vector norm is expressed as

$$\left\|W^{(n)}\right\| \le \gamma \left\|W^{(0)}\right\|,$$

where γ is a constant which applies uniformly to all matrices in the family of spatial discretisations for different average mesh spacings h. However, in practice it is often not

possible to prove strong stability. Instead, what can be more easily proved, and is used here, is a weaker form of stability called algebraic stability [9, 14, 16, 24]. This allows, at worst, a linear growth in the transient solution of the form

$$\left\|W^{(n)}\right\| \leq \gamma n \left\|W^{(0)}\right\|,$$

where γ is a constant. Together with absolute stability which ensures eventual exponential decay of the transient solution, this provides very practical stability criteria. It can also be proved that it provides a sufficient condition for convergence of consistent discretisations of initial value p.d.e.s, subject to very mild restrictions on the smoothness of the initial conditions [8]. This shows the practical usefulness of the concept of algebraic stability. As discussed in [9], a sufficient condition for algebraic (and absolute) stability is that

$$\tau\left(k\mathbb{P}^{1/2}\mathbb{C}\mathbb{P}^{1/2}\right) \subset S,$$

where the field of values τ is defined as

$$\tau\left(k\mathbb{P}^{1/2}\mathbb{C}\mathbb{P}^{1/2}\right) = \left\{ k\frac{W^*\mathbb{P}^{1/2}\mathbb{C}\mathbb{P}^{1/2}W}{W^*W} : W \neq 0 \right\}.$$

The first step in applying this stability analysis to the current problem is to prove that when $k = 1$ the field of values is bounded by a unit circle centred on $z = -1$. Making the substitution $U = \mathbb{P}^{1/2}W$, consider

$$\tau\left(\mathbb{P}^{1/2}\mathbb{C}\mathbb{P}^{1/2}\right) = \left\{ \frac{U^*\mathbb{C}U}{U^*\mathbb{P}^{-1}U} : U \neq 0 \right\} = \left\{ -\frac{U^*(\mathbb{A}+\mathbb{D})U}{U^*\mathbb{P}^{-1}U} : U \neq 0 \right\}.$$

Looking at the contributions from a single edge (i, j), let H be the orthonormal matrix of eigenvectors of A_{ij}, so that $\frac{1}{2}A_{ij}\Delta s_{ij} = H^T\Lambda H$ with Λ being a diagonal matrix. Introducing the definitions $V_i = HU_i$ and $V_j = HU_j$, and with $v_i^{(m)}$ and $v_j^{(m)}$ denoting the mth component of these vectors, we obtain

$$\begin{aligned}
U^*\mathbb{P}_{ij}^{-1}U &= \begin{pmatrix} V_i \\ V_j \end{pmatrix}^* \begin{bmatrix} |\Lambda| & 0 \\ 0 & |\Lambda| \end{bmatrix} \begin{pmatrix} V_i \\ V_j \end{pmatrix} \\
&= \sum_m \left|\lambda^{(m)}\right| \left(\left|v_i^{(m)}\right|^2 + \left|v_j^{(m)}\right|^2 \right), \\
U^*\mathbb{D}_{ij}U &= \begin{pmatrix} V_i \\ V_j \end{pmatrix}^* \begin{bmatrix} |\Lambda| & -|\Lambda| \\ -|\Lambda| & |\Lambda| \end{bmatrix} \begin{pmatrix} V_i \\ V_j \end{pmatrix} \\
&= \sum_m -\left|\lambda^{(m)}\right| \left(v_i^{(m)*}v_j^{(m)} + v_j^{(m)*}v_i^{(m)} \right) + \left|\lambda^{(m)}\right| \left(\left|v_i^{(m)}\right|^2 + \left|v_j^{(m)}\right|^2 \right),
\end{aligned}$$

and

$$\begin{aligned}
U^*\mathbb{A}_{ij}U &= \begin{pmatrix} V_i \\ V_j \end{pmatrix}^* \begin{bmatrix} 0 & +\Lambda \\ -\Lambda & 0 \end{bmatrix} \begin{pmatrix} V_i \\ V_j \end{pmatrix} \\
&= \sum_m \lambda^{(m)} \left(v_i^{(m)*}v_j^{(m)} - v_j^{(m)*}v_i^{(m)} \right),
\end{aligned}$$

Hence

$$
\begin{aligned}
&\left|U^*\mathbb{P}_{ij}^{-1}U - U^*(\mathbb{A}_{ij}+\mathbb{D}_{ij})U\right| \\
&\quad= \sum_m \left|\left(-\lambda^{(m)} + \left|\lambda^{(m)}\right|\right) v_i^{(m)*} v_j^{(m)} + \left(\lambda^{(m)} + \left|\lambda^{(m)}\right|\right) v_j^{(m)*} v_i^{(m)}\right| \\
&\quad\le \sum_m 2\left|\lambda^{(m)}\right| \left|v_i^{(m)}\right| \left|v_j^{(m)}\right| \\
&\quad\le \sum_m \left|\lambda^{(m)}\right| \left(\left|v_i^{(m)}\right|^2 + \left|v_j^{(m)}\right|^2\right), \\
&\quad\Rightarrow \left| U^*\mathbb{P}_{ij}^{-1}U - U^*(\mathbb{A}_{ij}+\mathbb{D}_{ij})U\right| \le U^*\mathbb{P}_{ij}^{-1}U.
\end{aligned}
$$

Summing over all edges then gives the desired result.

$$
|U^*\mathbb{P}^{-1}U - U^*(\mathbb{A}+\mathbb{D})U| \le U^*\mathbb{P}^{-1}U, \quad \Rightarrow \quad \left|1 - \frac{U^*(\mathbb{A}+\mathbb{D})\,U}{U^*\mathbb{P}^{-1}U}\right| \le 1. \tag{18}
$$

When $k \neq 1$ the field of values is bounded by a circle of radius k centred on $z = -k$. Choosing the largest such circle lying inside the stability region S gives a timestep k which is guaranteed to be algebraically stable. Numerical results will later establish that this is close to being a necessary condition for stability as well as sufficient. Appendix A illustrates, for three popular multistage integration schemes, the stability region S within which $|L(z)| \le 1$. It also shows for each case the largest circle which lies inside S and which corresponds to the sufficient stability limits of the scheme being analysed.

3.3. Wall Boundary Condition

In symmetrised form, the semidiscrete equations with the modified residual and preconditioner can be written as

$$
[(I-N)\mathbb{P}^{-1}(I-N)+N]\frac{dU}{dt} = -(I-N)(\mathbb{A}+\mathbb{D})(I-N)U.
$$

Note the introduction of the additional factor $(I-N)$ on the right-hand side, which is valid since $(I-N)U = U$. Defining $U_\perp = NU$ and $U_\parallel = (I-N)\,U$, and noting that $N^2 = N$, one obtains

$$
\begin{aligned}
&|U^*[(I-N)\mathbb{P}^{-1}(I-N)+N]U - U^*(I-N)(\mathbb{A}+\mathbb{D})(I-N)\,U| \\
&\quad= |U_\parallel^*\mathbb{P}^{-1}U_\parallel + U_\perp^*U_\perp - U_\parallel^*(\mathbb{A}+\mathbb{D})\,U_\parallel| \\
&\quad\le U_\parallel^*\mathbb{P}^{-1}U_\parallel + U_\perp^*U_\perp \\
&\quad= U^*[(I-N)\mathbb{P}^{-1}(I-N)+N]\,U,
\end{aligned}
$$

and hence

$$
\left|1 - \frac{U^*(I-N)(\mathbb{A}+\mathbb{D})(I-N)\,U}{U^*[(I-N)\mathbb{P}^{-1}(I-N)+N]\,U}\right| \le 1,
$$

proving that the field of values lies within the unit circle centred on $z = -1$, as before.

4. EXTENSION TO LOW MACH NUMBER PRECONDITIONING

At low Mach number the disparity between the acoustic and convective wave speeds cannot be adequately handled by block-Jacobi preconditioning on its own, and a slowdown of the convergence is observed. Furthermore, the numerical solution produced is often of poor quality, with significant errors in the pressure distribution due to the relative scaling of the different numerical smoothing terms. To address these difficulties, a low Mach number preconditioner can be incorporated into the numerical dissipation and hence into the block-Jacobi preconditioner. Again, the results obtained show good multigrid performances and the question of interest concerns now the implication of the modifications on the timestep stability limits presented in the previous sections. Before addressing this question, we first present the low Mach number preconditioning technique employed.

4.1. Preconditioned Numerical Dissipation

4.1.1. *1D Preliminary*

Expressed in symmetrised variables U, the Euler equations are preconditioned for low Mach number applications through the introduction of an invertible matrix Γ to become

$$\frac{\partial U}{\partial t} + \Gamma A \frac{\partial U}{\partial x} = 0.$$

The standard first-order upwind spatial discretisation gives the semidiscrete equation

$$\frac{dU}{dt} + \frac{1}{2}\Gamma A\, \delta_{2x} U - \frac{1}{2}|\Gamma A|\delta_x^2 U = 0,$$

which can be rewritten as

$$\Gamma^{-1}\frac{dU}{dt} + \frac{1}{2}A\delta_{2x}U - \frac{1}{2}\Gamma^{-1}|\Gamma A|\delta_x^2 U = 0,$$

where the δ_{2x} and δ_x^2 operators are defined as

$$\delta_{2x}U_j = U_{j+1} - U_{j-1}$$
$$\delta_x^2 U_j = U_{j+1} - 2U_j + U_{j-1}.$$

This shows that the introduction of the low Mach number preconditioning changes the numerical smoothing, as well as the unsteady evolution of the flow field.

4.1.2. *3D Generalisation*

Following the 1D approach, the flux function defined in Eq. (5) is modified to incorporate the low Mach number preconditioner, becoming

$$F_{ij}^I = \frac{1}{2}\Bigg(\mathcal{F}_{ij}^I(Q_i) + \mathcal{F}_{ij}^I(Q_j) - \Gamma_{ij}^{-1}|\Gamma_{ij}A_{ij}| \times \left(\Psi(Q_i - Q_j) - \frac{1}{3}(1-\Psi)(L_i(Q) - L_j(Q))\right)\Bigg). \tag{19}$$

Here the low Mach number preconditioning matrix Γ_{ij} has been transformed from symmetrised variables to conservative variables; see [19] for full details.

In [15] Lee gives a broad overview of the current state of preconditioning. As demonstrated in [5], and because of their highly nonnormal feature for low Mach number, many local preconditioners can transiently amplify perturbations by a factor of $1/M$ as $M \to 0$. Taking this fact into account, the preconditioner used in the current work is that developed by Weiss and Smith [31] and also used by Darmofal and Siu [6]. This preconditioner, expressed in symmetrised variables, is simply

$$\Gamma = \begin{bmatrix} \epsilon & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

ϵ is a free parameter whose role is to equilibrate the eigenvalues. This preconditioner alters only the eigenvalues corresponding to acoustic waves. Choosing $\epsilon = O(M^2)$ ensures that the convective and acoustic wave speeds are of a similar magnitude, proportional to the flow speed. In some applications, ϵ is chosen to be equal to some multiple of the square of the freestream Mach number [13, 28, 29]. However, to handle internal flows in which there can be extensive regions of low Mach number flow even though the inflow and outflow may be transonic, we instead choose to follow Darmofal and Siu [6] in defining ϵ to be

$$\epsilon_{ij} = \min\left[1, \eta M_{\max}^2\right],$$

where $M_{\max}$ is the maximum Mach number in a neighbourhood of the edge [19] and η is a free parameter set to 3.0 [13], which switches off the low Mach number preconditioning when the Mach number exceeds $1/\sqrt{3}$.

Modification of the artificial dissipation automatically implies modification of the block-Jacobi preconditioner. The adjustment is straightforward and only concerns the inviscid part. Thus Eq. (6) becomes

$$\left(P_j^I\right)^{-1} = \frac{1}{2V_j}\left(\sum_{i\in E_j} \Gamma_{ij}^{-1}|\Gamma_{ij}A_{ij}|\Delta s_{ij} + \sum_{k\in B_j} \Gamma_k^{-1}|\Gamma_k A_k|\Delta s_k\right). \tag{20}$$

4.2. Stability Analysis for Low Mach Number Preconditioning

Using symmetrised variables, the linearised 3D Euler equations are of the form

$$\frac{\partial U}{\partial t} + A_x\frac{\partial U}{\partial x} + A_y\frac{\partial U}{\partial y} + A_z\frac{\partial U}{\partial z} = 0,$$

with A_x, A_y, and A_z being uniform symmetric matrices. Introducing the low Mach number preconditioning gives

$$\Gamma^{-1}\frac{\partial U}{\partial t} + A_x\frac{\partial U}{\partial x} + A_y\frac{\partial U}{\partial y} + A_z\frac{\partial U}{\partial z} = 0,$$

and performing a change of variables to $V = \Gamma^{-1/2}U$ produces

$$\frac{\partial V}{\partial t} + \Gamma^{1/2}A_x\Gamma^{1/2}\frac{\partial V}{\partial x} + \Gamma^{1/2}A_y\Gamma^{1/2}\frac{\partial V}{\partial y} + \Gamma^{1/2}A_z\Gamma^{1/2}\frac{\partial V}{\partial z} = 0.$$

This is very similar to the original symmetrised Euler equations in that the three coefficient matrices are symmetric.

It can be shown (see [19] for full details) that the numerical scheme obtained from this p.d.e. by using first-order upwinding and block-Jacobi preconditioning is exactly the same as that obtained by constructing the nonlinear discretisation of the Euler equations using the preconditioned fluxes in Eq. (19) and the block-Jacobi preconditioning in Eq. (20) and then linearising these equations around a uniform flow field and transforming the resulting linear equations into the modified symmetrised variables V. Consequently, the numerical stability analysis is exactly the same as before, but with the symmetric matrix A replaced throughout by the symmetric matrix $\Gamma^{1/2}A\Gamma^{1/2}$. This changes the eigenvalues in the analysis, but does not change the conclusion that the semidiscrete equations are stable and the fully discrete equations satisfy the conditions for algebraic stability.

In general the requirement for the analysis to remain valid is that the low Mach number preconditioner Γ must be symmetric and positive definite when expressed in symmetrised variables; $\Gamma^{1/2}$ is then well defined. The van Leer–Lee–Roe matrix satisfies this condition [30], but those due to Turkel [27] and Lee [15] do not.

5. NUMERICAL RESULTS

To test the accuracy of the predicted stability limits, we have performed a number of numerical experiments for both inviscid and viscous flows on triangular and tetrahedral grids. As indicated in Table I, cases 1 and 2 are based on a 2D grid around a NACA0012 airfoil, case 3 is based on a 2D highly stretched grid around an RAE2822 airfoil, and case 4 consists of a half complete aircraft configuration bounded by a symmetry plane.

In all of the computations, the four-stage Runge–Kutta method described in Appendix B is used for the timemarching. For the preconditioned Euler equations with a first-order discretisation, the theoretical CFL limit guaranteeing algebraic stability is 1.39. In all four cases, we present the convergence history for a multigrid calculation which uses a second-order discretisation on the finest grid. The purpose of this is to demonstrate the good multigrid convergence achieved by the Jacobi preconditioning, and also the low Mach

TABLE I
Test Cases—Geometry, Free Stream Mach Number, Angle of Attack, Grid Type, and Number of Vertices

Test	Geometry	M_∞	α	Mesh	# Nodes
Case 1	NACA0012	0.8	1.25	Triangular	5700
Case 2 (a)	NACA0012	0.1	0.0	Triangular	5700
Case 2 (b)	NACA0012	0.01	0.0	Triangular	5700
Case 2 (c)	NACA0012	0.001	0.0	Triangular	5700
Case 2 (d)	NACA0012	0.0001	0.0	Triangular	5700
Case 3	RAE2822	0.73	2.8	Triangular	11200
Case 4	FALCON	0.85	2.0	Tetrahedral	156000

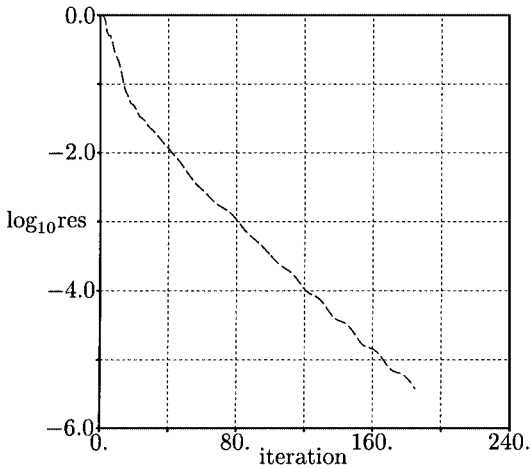


FIG. 1. Case 1, multigrid convergence history, CFL = 1.58.

number preconditioning in case 2. In addition, for the first three cases we investigate the CFD stability limit by performing a single grid calculation using the first-order discretisation, starting from the converged second-order solution on the fine grid. For all the convergence histories, what is plotted is the L_2 norm of the residual vector on the finest mesh level, normalised by the initial residual.

Case 1 is a standard transonic NACA0012 test case with a strong shock on the upper surface and a weak shock on the lower surface. Figure 1 presents the multigrid convergence history using five grid levels. The calculation is stable up to CFL = 1.58, which corresponds to the actual stability boundary. Running the code on the finest grid using the first-order discretisation, starting from the already converged second-order solution, the calculation becomes unstable for CFL = 1.59, as shown in Fig. 2. Thus the theory underpredicts the stability boundary by approximately 14%.

In Case 2, we consider four low Mach number flows around the NACA0012 using the low Mach number preconditioner in addition to the Jacobi preconditioner (Fig. 3 for case 2b) with $M = 0.01$ shows that a calculation without the low Mach preconditioner code requires

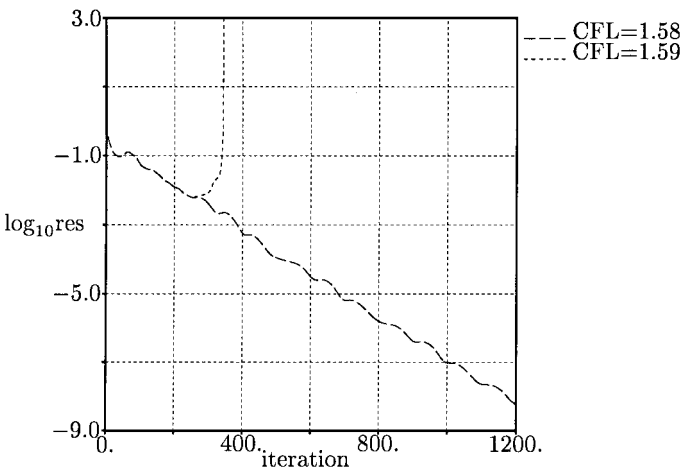


FIG. 2. Limit of stability for Case 1.

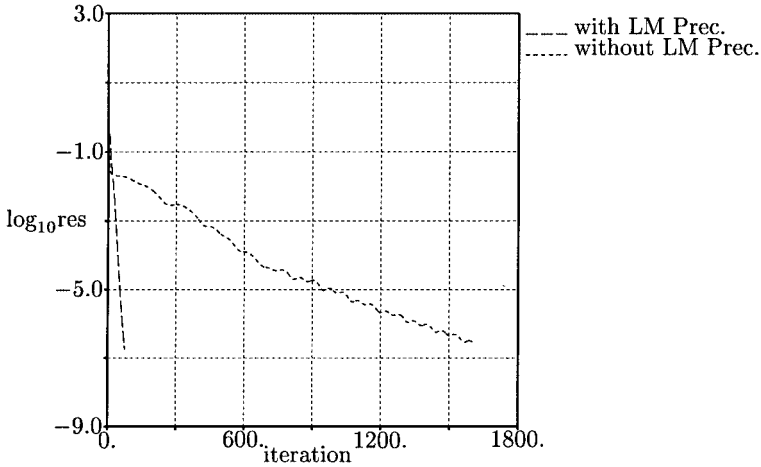


FIG. 3. Convergence history with and without low Mach number preconditioner for $M_\infty = 0.01$ with multigrid.

approximately 20 times more iterations. In both cases, five levels of multigrid were used with CFL = 1.63. The results for the other Mach numbers were very similar. In addition, the use of the low Mach number preconditioner made the accuracy of the computed flow fields almost independent of the Mach number, unlike the unpreconditioned results which deteriorate badly at low Mach numbers, as shown in [19]. In Fig. 4 we investigate the CFL stability limit for the single grid calculation. The results for the four subcases show that the actual stability boundary is at CFL = 1.63 so the theory now underpredicts the stability boundary by 17%.

In case 3, a viscous calculation is performed for the flow around a 2D RAE2822 airfoil. The case investigated is the standard AGARD Case 9 [4]. The Reynolds number based on a cord of unit length is 6.5×10^6. The fine grid counts 23 points accross the bounday layer, whereas the coarsest grid has only one point left. Again, good multigrid convergence is achieved, as shown in Fig. 5, but this time the theory overpredicts the CFL limit since

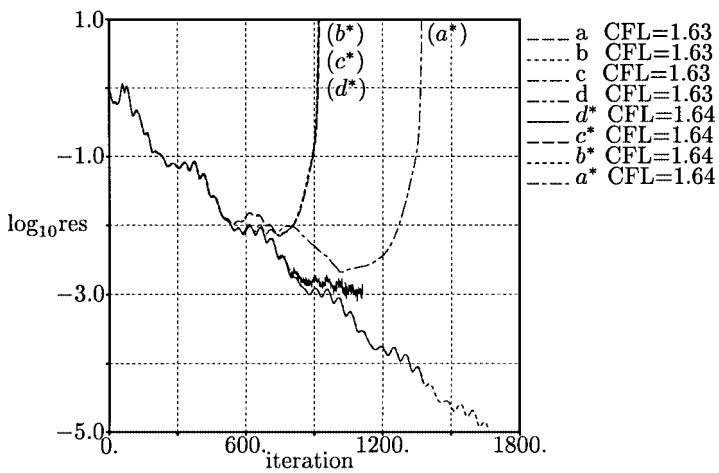


FIG. 4. Convergence history comparison for low Mach numbers and stability boundaries.

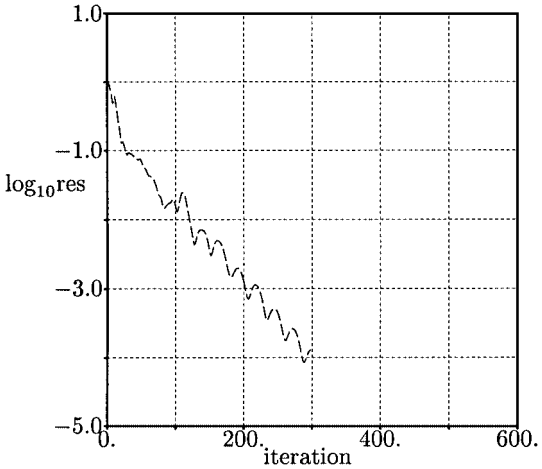


FIG. 5. Multigrid convergence history for Case 3, CFL = 1.25.

the stability limit was found to be CFL = 1.25. We attribute this overprediction to the fact that our analysis does not take into account several characteristics of this test case. One is that the theory is based on small perturbations to a uniform flow field, which is far from being the situation in this turbulent viscous case where we have a strong shock and recirculation. Also, we are using multigrid with a second-order spatial discretisation on the finest grid level, neither of which is accounted for in the theory. The results in Fig. 6 for a single grid calculation with the first-order discretisation show the stability boundary is CFL = 1.32, so the theory now overpredicts the stability limit by less than 4%. To get closer to the conditions for our analysis, we show in Fig. 7 a similar test, but this time applied to the first coarsened level that is used in the multigrid calculation. In this case, the inviscid terms in the discretisation are more dominant and the actual stability boundary stands at CFL = 1.56 so the theory, as in the two previous cases, underpredicts the stability boundary by approximately 12%.

Finally, case 4 concerns the inviscid transonic flow over a Falcon jet designed by Avions Marcel Dassault, France. This case is included to demonstrate the effectiveness of the Jacobi

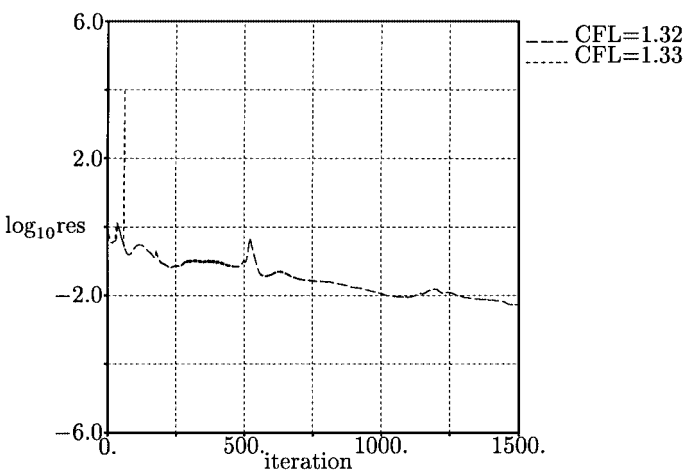


FIG. 6. Limit of stability for Case 3; fine mesh.

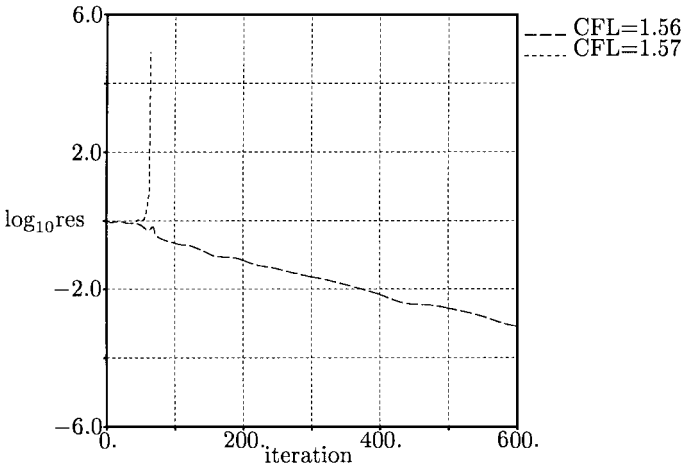


FIG. 7. Limit of stability for Case 3; first coarse level.

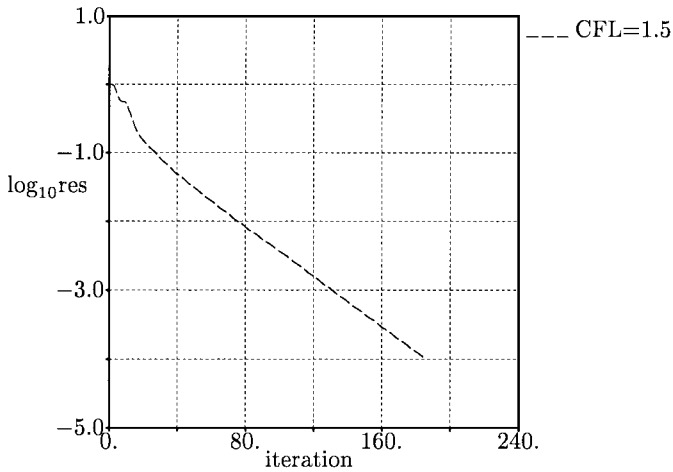


FIG. 8. Case 4, business jet: multigrid convergence history.

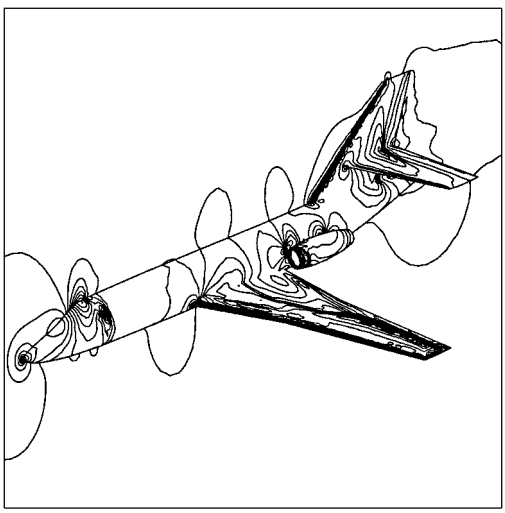


FIG. 9. Case 4, business jet: Mach number contours.

preconditioner for a 3D application. Figures 8 and 9 show the multigrid convergence history and the Mach contour plot obtained with the second-order discretisation on the finest grid. Convergence to engineering accuracy is achieved in 185 multigrid cycles with a total CPU time of approximately 3 h on a 250-MHz SGI Octane. This case exemplifies the effectiveness of the numerical algorithm for a real 3D application, with the multigrid behaving well as long as the sufficient CFL condition for algebraic stability is satisfied. More details and examples can be found in [19].

6. CONCLUSIONS

In this paper, we have analysed the numerical stability of discrete approximations of the Euler equations on unstructured grids. It has been shown that the use of first-order characteristic smoothing and block-Jacobi preconditioning results in a semidiscrete system of equations which is stable when analysed for perturbations to a uniform flow. The corresponding fully discrete equations satisfy the conditions for algebraic stability when the timestep satisfies the appropriate CFL condition. The analysis has also been extended to include the class of symmetric low Mach number preconditioners which greatly accelerate convergence for low Mach number flows and improve the steady-state accuracy.

APPENDIX A: SEMIDISCRETE DISCRETISATION

The equations that we consider in this Appendix are the linear semidiscrete preconditioned Euler equations from Section 3.1 using the symmetrised variables and written as

$$\left(\sum_{i\in E_j}\frac{1}{2}|A_{ij}|\Delta s_{ij}\right)\frac{dU_j}{dt} = -\sum_{i\in E_j}\frac{1}{2}\big(A_{ij}(U_i+U_j)-|A_{ij}|(U_i-U_j)\big)\Delta s_{ij}.$$

The Jacobian matrix A is defined for each edge (i, j), such that $A = A_x n_x + A_y n_y + A_z n_z$, where A_x, A_y, A_z are the linear uniform flux Jacobians written for the symmetrising variables and n_x, n_y, n_z the components of the normal vector along which the fluxes are evaluated.

$$A_x = L^{-1}A^x L,$$
$$A_y = L^{-1}A^y L,$$
$$A_z = L^{-1}A^z L,$$

with the transformation matrix between the conservative variables and the symmetrising variable

$$L = \begin{bmatrix} \sqrt{\gamma}\frac{\rho}{c} & 0 & 0 & 0 & 0 \\ \sqrt{\gamma}\frac{\rho u}{c} & \rho & 0 & 0 & 0 \\ \sqrt{\gamma}\frac{\rho v}{c} & 0 & \rho & 0 & 0 \\ \sqrt{\gamma}\frac{\rho w}{c} & 0 & 0 & \rho & 0 \\ \sqrt{\gamma}\frac{\rho E}{c} & \rho u & \rho v & \rho w & \sqrt{\frac{\gamma}{\gamma-1}}\frac{p}{c} \end{bmatrix},$$

and also,

$$A^x = \begin{bmatrix} u & \rho & 0 & 0 & 0 \\ 0 & u & 0 & 0 & \frac{1}{\rho} \\ 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & u & 0 \\ 0 & \rho c^2 & 0 & 0 & u \end{bmatrix}, \quad A^y = \begin{bmatrix} v & 0 & \rho & 0 & 0 \\ 0 & v & 0 & 0 & 0 \\ 0 & 0 & v & 0 & \frac{1}{\rho} \\ 0 & 0 & 0 & v & 0 \\ 0 & 0 & \rho c^2 & 0 & v \end{bmatrix},$$

$$A^z = \begin{bmatrix} w & 0 & 0 & \rho & 0 \\ 0 & w & 0 & 0 & 0 \\ 0 & 0 & w & 0 & 0 \\ 0 & 0 & 0 & w & \frac{1}{\rho} \\ 0 & 0 & 0 & \rho c^2 & w \end{bmatrix}.$$

A is then a symmetric matrix whose eigenvalues are $u_n = un_x + vn_y + wn_z$ and $u_n \pm c$.

APPENDIX B: RUNGE–KUTTA STABILITY REGIONS

Let us consider a system of o.d.e.'s of the form

$$\frac{dQ}{dt} = CQ,$$

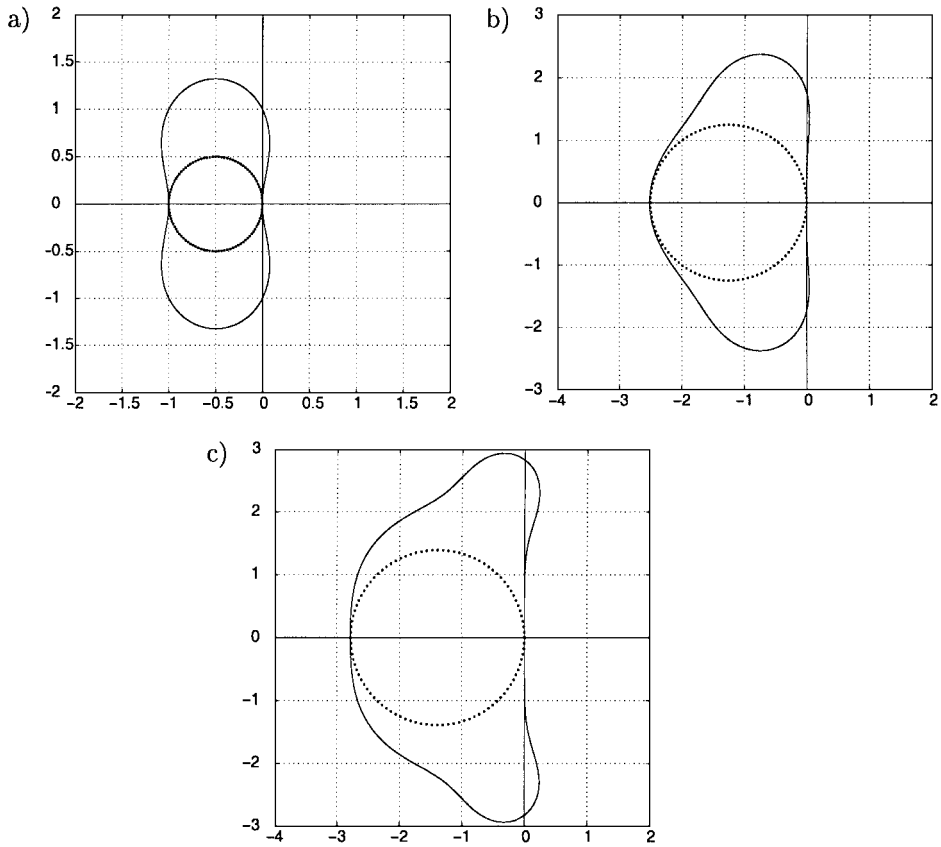
where C is a real square matrix. Runge–Kutta time integration with a timestep k gives

$$Q^{(n+1)} = L(kC)\, Q^{(n)},$$

where

$$L(z) = \sum_{m=0}^{p} a_m z^m, \quad (a_0 = a_1 = 1, a_p \neq 0)$$

is the Runge–Kutta polynomial function with stability region S. This region is shown in the following figures for three popular multistage schemes: (a) Predictor–corrector: $u^{(1)} = u^n + \lambda \Delta t u^n$; $u^{n+1} = u^n + \lambda \Delta t u^{(1)}$; $r_c = 0.5$. (b) Three-stage scheme: $u^{(1)} = u^n + \frac{1}{3}\lambda \Delta t u^n$; $u^{(2)} = u^n + \frac{1}{2}\lambda \Delta t u^{(1)}$; $u^{n+1} = u^n + \lambda \Delta t u^{(2)}$; $r_c = 1.25$. (c) Four-stage scheme: $u^{(1)} = u^n + \frac{1}{4}\lambda \Delta t u^n$; $u^{(2)} = u^n + \frac{1}{3}\lambda \Delta t u^{(1)}$; $u^{(3)} = u^n + \frac{1}{2}\lambda \Delta t u^{(2)}$; $u^{n+1} = u^n + \lambda \Delta t u^{(3)}$; $r_c = 1.39$. The radius r_c of the largest circle lying within S equals the maximum timestep for which the analysis in the main paper gives a sufficient condition for algebraic stability.



ACKNOWLEDGMENTS

We thank J. Elliott for advice on implementing the slip boundary condition within the preconditioner. The first author gratefully acknowledges the funding of the European Community. The research was also supported by funding from the EPSRC under research Grant GR/L18280.

REFERENCES

1. S. Abarbanel and D. Gottlieb, Optimal time splitting for two- and three-dimensional Navier–Stokes equations with mixed derivatives, *J. Comput. Phys.* **35**, 1 (1981).
2. W. K. Anderson, R. Rausch, and D. Bonhaus, *Implicit Multigrid Algorithms for Incompressible Turbulent Flows on Unstructured Grids*, AIAA Paper 95-1740-CP (1995).
3. T. J. Barth and S. W. Linton, *An Unstructured Mesh Newton Solver for Compressible Fluid Flow and Its Parallel Implementation*, AIAA Paper 95-0221 (January 1995).
4. P. H. Cook, M. A. McDonald, and M. C. P. Firmin, *Aerofoil RAE 2822—Pressure Distributions and Boundary Layer and Wake Measurements*, AGARD-AR-138 (1979).
5. D. L. Darmofal and P. J. Schmid, The importance of eigenvectors for local preconditioners of the Euler equations, *J. Comput. Phys.* **127**, 346 (1996).
6. D. L. Darmofal and K. Siu, A robust multigrid algorithm for the Euler equations with local preconditioning and semi-coarsening, *J. Comput. Phys.* **151**, 728 (1999).
7. J. Elliott, *Aerodynamic Optimization Based on the Euler and Navier–Stokes Equations Using Unstructured Grids*, Ph.D. thesis (Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, 1998).

8. M. B. Giles, On the stability and convergence of discretisations of initial value p.d.e.s. *IMA J. Numer. Anal.* **17**(4), 563 (1997).
9. M. B. Giles, Stability analysis of a Galerkin/Runge–Kutta Navier–Stokes discretisation on unstructured tetrahedral grids, *J. Comput. Phys.* **132**, 201 (1997).
10. B. Gustafsson and A. Sundström, Incompletely parabolic problems in fluid dynamics. *SIAM J. Appl. Math.* **35**(2), 343 (1978).
11. W. Hackbusch, *Multi-Grid Convergence Theory*, edited by Hackbusch-Trottenberg, Lecture Notes in Mathematics Springer-Verlag, Berlin/New York, 1982, Vol. 960.
12. A. Harten, High resolution schemes for conservation laws, *J. Comput. Phys.* **49**, 357 (1983).
13. D. Jesperson, T. Pulliam, and P. Buning, *Recent Enhancements to OVERFLOW*, AIAA Paper 97-0644 (1997).
14. J. F. B. M. Kraaijevanger, H. W. J. Lenferink, and M. N. Spijker, Stepsize restrictions for stability in the numerical solution of ordinary and partial differential equations, *J. Comput. Appl. Math.* **20**, 67 (November 1987).
15. D. Lee, *Local Preconditioning of the Euler Equations*, Ph.D. thesis (University of Michigan, 1996).
16. H. W. J. Lenferink and M. N. Spijker, On the use of stability regions in the numerical analysis of initial value problems, *Math. Comput.* **57**(195), 221 (1991).
17. L. Martinelli, *Calculations of Viscous Flows with a Multigrid Method*, Ph.D. thesis (Deptartment of Mechanical and Aerospace Engineering, Princeton University, 1987).
18. D. J. Mavriplis, *Multigrid Strategies for Viscous Flow Solvers on Anisotropic Unstructured Meshes*, AIAA Paper 97-1952 (1997).
19. P. Moinier, *Algorithm Developments for an Unstructured Viscous Flow Solver*, Ph.D. thesis (Oxford University, 1999).
20. P. Moinier, J.-D. Müller, and M.B. Giles, *Edge-Based Multigrid and Preconditioning for Hybrid Grids*, AIAA Paper 99-3339 (1999).
21. C. Ollivier-Gooch, Towards problem-independent multigrid convergence rates for unstructured mesh methods I: Inviscid and laminar flows, in *Proceedings of the 6th International Symposium on CFD, Lake Tahoe, September 1995*.
22. N. A. Pierce, *Preconditioned Multigrid Methods for Compressible Flow Calculations on Stretched Meshes*, Ph.D. thesis (Oxford University, 1997).
23. N. A. Pierce and M. B. Giles, Preconditioned multigrid methods for compressible flow calculations on stretched meshes, *J. Comput. Phys.* **136**, 425 (1997).
24. S. C. Reddy and L. N. Trefethen, Stability of the method of lines, *Numer. Math.* **62**, 235 (1992).
25. R. D. Richtmyer and K. W. Morton, *Difference Methods for Initial-Value Problems*, 2nd ed. (Wiley-Interscience, New York, 1967). [Reprint edition (1994) Krieger Publishing Company, Malabar]
26. P. R. Spalart and S. R. Allmaras, A one-equation turbulence model for aerodynamic flows, *La Recherche Aérospatiale* **1**, 5 (1994).
27. E. Turkel, Preconditioning methods for solving the incompressible and low speed compressible equations, *J. Comput. Phys.* **72**, 277 (1987).
28. E. Turkel, *Preconditioning-Squared Methods for Multidimensional Aerodynamics*, AIAA Paper 97-2025 (1997).
29. E. Turkel, V. Vatsa, and R. Radespiel, *Preconditioning Methods for Low-Speed Flows*, AIAA Paper 96-2460 (1996).
30. B. van Leer, W. T. Lee, and P. L. Roe, *Characteristic Time-Stepping or Local Preconditioning for the Euler Equations*, AIAA Paper 91-1552-CP (1991).
31. J. Weiss and W. Smith, Preconditioning applied to variable and constant density flows, *J. Amer. Inst. Aeronaut. Astronaut.* **33**(11), 2050 1995.