AIAA JOURNAL
Vol. 33, No. 6, June 1995

Anomalous Asymmetries in Flows Generated by Algorithms that Fail to Conserve Symmetry

Yuval Levy* and Lambertus Hesselink†
Stanford University, Stanford, California 94305-4035
and
David Degani‡
Technion—Israel Institute of Technology, Haifa 32000, Israel

Numerical simulations of high-angle-of-attack flows around slender bodies of revolution are carried out using four different numerical algorithms: a partially flux-split algorithm, the Beam and Warming algorithm in its original and diagonal forms, and an algorithm combining block and diagonal forms. Comparisons of laminar flows at $\alpha = 20$ and 40 deg reveal that the diagonal algorithm fails to conserve symmetry at high angles of attack and a spurious asymmetry is developed. The source of the spurious asymmetry has been traced and found to be inherent to the diagonalization process.

Introduction

ONE of the significant advantages in using numerical simulations to investigate stability properties of certain flows is the ability to obtain a disturbance-free base solution. This base solution can then be perturbed for parametric studies of the effect of controlled disturbances. Obviously, due to the high sensitivity of such flows to any disturbance, whatever numerical algorithm is used must be free of any kind of bias.

A good example is the use of numerical methods to simulate flows around slender bodies at high angles of attack. Since slender bodies of revolution are similar to aircraft forebodies and are numerically and experimentally tractable, they are often chosen as generic exemplars of forebodies for numerical simulations and experiments. It has been noted that for such bodies, there is an angle-of-attack range for which minute imperfections at the tip can cause large asymmetries in the flow pattern.[1–7] This fact suggests the existence of inherent instabilities in the expected symmetric flow. This is certainly the case as the angle of attack approaches 90 deg since the observed Karman vortex street in the wake of the body is well known to be the result of an absolute instability of the symmetric flow. In this case, the resulting perturbed flow persists even after removal of the disturbance that initiated it. However, for flow at a lower angle-of-attack range, roughly from $\alpha = 30$ to 65 deg, experiments and computations suggest the existence of a convective instability mechanism.[8–11] For example, Degani and Tobak[8] and Williams and Bernhardt[9] have shown experimentally that a small symmetry-breaking disturbance, acting on the body's surface near the tip, is sufficient to excite an asymmetric flow component which vanishes when the disturbance is removed. More recent studies support the existence of a convective instability mechanism even when it is the flow upstream of the body that is disturbed.[12,13] Theoretical work[13] suggests that the response to an impulsive disturbance grows in accordance with the convective instability mode of behavior.

All this means that numerical simulations of such flows should give the basic symmetric solutions as long as algorithms, boundary and initial conditions, and grids are perfectly symmetric. Indeed, the first successful achievement of a simulated fully three-dimensional asymmetric flow[10,14,15] pointed to the necessity of perturbing the symmetric base flow by an asymmetrically disposed time-invariant disturbance. Otherwise, the flow remained symmetric even in the case of high angle of attack where experimental results showed the presence of asymmetry.

However, some results obtained later on the basis of a conical flow assumption[16,17] showed that asymmetric conical solutions could be obtained even without perturbations, just by allowing the residual of the computation to drop below a certain value. In another case a three-dimensional solver[18] gave asymmetric solutions without the use of a perturbation. Thomas,[11] on the other hand, showed that his three-dimensional code gave a symmetric base solution as long as all boundary conditions were symmetric, but when conical restrictions were imposed the solution became asymmetric. This result suggests that the conical approximation may have altered the scope of the problem in question. Eliminating one coordinate direction causes the approximation to be quasi-two-dimensional and, therefore, makes the crossflow susceptible to the absolute instability that leads to vortex asymmetry in the flow past a circular cylinder. With the exception of Thomas, the authors who have obtained asymmetric flow solutions without the imposition of a fixed asymmetric disturbance have argued that the numerical algorithms they used were able to reveal the true nature of the flows by bypassing the symmetric base solution. The main argument against this notion is apparent in their solutions: Why should the solutions have sought a preferred orientation of the asymmetry? In experiments it seems clear that the reason lies in the practical fact that the body tip is never perfectly symmetric. But what about numerical codes? Why should one asymmetric solution have been selected rather than, say, the mirror image solution? Obviously, these contradicting arguments need to be resolved before numerical algorithms can be used freely to simulate high angle of attack or other types of flows that are sensitive to small disturbances.

Although some reasonable explanations may be given in an effort to resolve these questions, and they will be discussed in the Discussion and Conclusions section, the main emphasis of this paper is to show analytically and to demonstrate numerically that a typical and popular algorithm, the diagonal algorithm,[19] which looks innocent to the naked eye, contains an inherent asymmetry. This asymmetry of the algorithm, heretofore unrecognized, is bound not only to bypass the symmetric base solution but also to produce a spurious nonphysical asymmetric solution.

Theoretical Background

The three-dimensional conservation-law form of the Navier–Stokes equations in body-fitting curvilinear coordinates is

$$\partial_\tau \hat{Q} + \partial_\xi (\hat{E} - \hat{E}_v) + \partial_\eta (\hat{F} - \hat{F}_v) + \partial_\zeta (\hat{G} - \hat{G}_v) = 0 \qquad (1)$$

Received March 21, 1994; presented as Paper 94-2287 at the AIAA 25th Fluid Dynamics Conference, Colorado Springs, CO, June 20–23, 1994; revision received Dec. 9, 1994; accepted for publication Dec. 20, 1994.

*Postdoctoral Fellow, Aeronautics and Astronautics Department. Member AIAA.

†Professor, Aeronautics and Astronautics and Electrical Engineering Departments. Member AIAA.

‡Professor, Faculty of Mechanical Engineering. Associate Fellow AIAA.

where τ is the time, and the independent spatial variables, ξ, η, and ζ are the streamwise, circumferential, and radial coordinates, respectively. In Eq. (1) $\hat{Q}$ is the vector of dependent flow variables, mass, momentum, and energy; $\hat{E} = \hat{E}(\hat{Q})$, $\hat{F} = \hat{F}(\hat{Q})$, and $\hat{G} = \hat{G}(\hat{Q})$ are the inviscid flux vectors, whereas the terms $\hat{E}_v$, $\hat{F}_v$, and $\hat{G}_v$ are fluxes containing the viscous derivatives.

For body-conforming coordinates and flow at high Reynolds number (with ζ the coordinate leading away from the surface), the thin-layer approximation can be applied. By using dimensionless variables the Reynolds number Re can be factored out, and Eq. (1) takes the following form[20,21]:

$$\partial_\tau \hat{Q} + \partial_\xi \hat{E} + \partial_\eta \hat{F} + \partial_\zeta \hat{G} = Re^{-1}\partial_\zeta \hat{S} \tag{2}$$

The equation of state is used to compute the pressure from the known flow variables.

Applying the implicit approximately factored Beam and Warming[22,23] algorithm to Eq. (2) results in a three-factored finite difference equation where each of the factors corresponds to one of the coordinate directions. For first- or second-order accuracy in time the system of finite difference equations to be solved is

$$\left(I + h\delta_\xi \hat{A}^n\right)\left(I + h\delta_\eta \hat{B}^n\right)\left(I + h\delta_\zeta \hat{C}^n - hRe^{-1}\bar{\delta}_\zeta J^{-1}\hat{M}^n J\right) \times \Delta\hat{Q}^n = \hat{R}^n \tag{3}$$

where $\hat{R}^n$ is

$$\hat{R}^n = -\Delta t\left(\delta_\xi \hat{E}^n + \delta_\eta \hat{F}^n + \delta_\zeta \hat{G}^n - Re^{-1}\bar{\delta}_\zeta \hat{S}^n\right) \tag{4}$$

Setting the quantity h equal to Δt yields the first-order-accurate Euler implicit form, whereas setting h equal to $\Delta t/2$ yields the second-order-accurate trapezoidal form. The δ operators denote central differencing, and the $\bar{\delta}$ operator denotes a midpoint operator used in order to preserve the block tridiagonal form. Symbols $\hat{A}$, $\hat{B}$, and $\hat{C}$ stand for the Jacobian matrices resulting from the local time linearization of the flux vectors:

$$\hat{A}^n = \left(\frac{\partial \hat{E}}{\partial \hat{Q}}\right)^n, \qquad \hat{B}^n = \left(\frac{\partial \hat{F}}{\partial \hat{Q}}\right)^n \qquad \hat{C}^n = \left(\frac{\partial \hat{G}}{\partial \hat{Q}}\right)^n \tag{5}$$

The term $J^{-1}\hat{M}^n J$ is the Jacobian matrix resulting from the local time linearization of the viscous vector

$$J^{-1}\hat{M}^n J = \left(\frac{\partial \hat{S}}{\partial \hat{Q}}\right)^n \tag{6}$$

In matrix form, the system of equations for a circumferential factor with periodic boundary conditions takes the form given in Eq. (A1) of Appendix A.

The solution of each of the factors of this system requires a block tridiagonal inversion. The block inversion requires 25 variables per grid point on each cross section, and the operation count per grid point is high.

The required temporary space and the high operation count for block inversions motivated Pulliam and Chaussee[19] to develop the diagonal version of the implicit form. They used a similarity transformation[19] to diagonalize $\hat{A}$, $\hat{B}$, and $\hat{C}$:

$$\hat{A} = \hat{T}_\xi \hat{\Lambda}_\xi \hat{T}_\xi^{-1}, \qquad \hat{B} = \hat{T}_\eta \hat{\Lambda}_\eta \hat{T}_\eta^{-1}, \qquad \hat{C} = \hat{T}_\zeta \hat{\Lambda}_\zeta \hat{T}_\zeta^{-1} \tag{7}$$

Applying the diagonalization to Eq. (3) and dropping the viscous Jacobian matrix $\hat{M}$ since it has different eigenvectors than $\hat{C}$, results in

$$\left[\left(\hat{T}_\xi \hat{T}_\xi^{-1}\right)^n + h\delta_\xi\left(\hat{T}_\xi \hat{\Lambda}_\xi \hat{T}_\xi^{-1}\right)^n\right]\left[\left(\hat{T}_\eta \hat{T}_\eta^{-1}\right)^n + h\delta_\eta\left(\hat{T}_\eta \hat{\Lambda}_\eta \hat{T}_\eta^{-1}\right)^n\right] \times \left[\left(\hat{T}_\zeta \hat{T}_\zeta^{-1}\right)^n + h\delta_\zeta\left(\hat{T}_\zeta \hat{\Lambda}_\zeta \hat{T}_\zeta^{-1}\right)^n\right]\Delta\hat{Q}^n = \hat{R}^n \tag{8}$$

By moving $\hat{T}_\xi$, $\hat{T}_\eta$, and $\hat{T}_\zeta$ outside of the difference operators δ_ξ, δ_η, and δ_ζ, respectively, Pulliam and Chaussee constructed the so called "diagonal" form of the algorithm

$$\hat{T}_\xi^n\left(I + h\delta_\xi \hat{\Lambda}_\xi^n\right)\left(\hat{T}_\xi^{-1}\right)^n \hat{T}_\eta^n\left(I + h\delta_\eta \hat{\Lambda}_\eta^n\right)\left(\hat{T}_\eta^{-1}\right)^n \hat{T}_\zeta^n\left(I + h\delta_\zeta \hat{\Lambda}_\zeta\right)^n \times \left(\hat{T}_\zeta^{-1}\right)^n \Delta\hat{Q}^n = \hat{R}^n \tag{9}$$

The circumferential factor of the diagonal algorithm also can be written in the matrix form that is described in Eq. (A2) of Appendix A.

Solution of this system requires five scalar tridiagonal inversions for each factor instead of the block inversion because the $\hat{\Lambda}$ matrices are diagonal and, therefore, the equations are decoupled. The inversion procedure of each factor is then the following: 1) Multiply the right-hand side by the matrix $\hat{T}^{-1}$. 2) Conduct the five scalar inversions. 3) Multiply the resulting right-hand side by the matrix $\hat{T}$.

Although two matrix multiplications are required, the operation count and the memory space required are decreased dramatically. By multiplying the matrices $\hat{T}$, $I + \delta\hat{\Lambda}$, and $\hat{T}^{-1}$, one can see exactly the form of the solution achieved by the inversion procedure. The result of that multiplication also reveals how the left-hand side of the original Beam and Warming algorithm differs from that of the diagonal form of the algorithm. The result is presented in Eq. (A3) of Appendix A. (This will be used later to detect the origin of the numerical asymmetry.) Pulliam and Chaussee[19] claimed that the steady-state spatial accuracy is not affected since the steady-state solution is determined by the type of differencing used in forming $\hat{R}^n$.

In addition to the algorithms just described, other implicit algorithms discussed here are as follows:

1) First is an algorithm combining the standard and diagonalized algorithm. By using the similarity transformation to diagonalize only the streamwise and radial Jacobian matrices nonetheless maintaining the block scheme for the circumferential direction the following scheme is formed:

$$\hat{T}_\xi^n\left(I + h\delta_\xi \hat{\Lambda}_\xi^n\right)\left(\hat{T}_\xi^{-1}\right)^n\left(I + h\delta_\eta \hat{B}^n\right)\hat{T}_\zeta^n\left(I + h\delta_\zeta \hat{\Lambda}_\zeta\right)^n \times \left(\hat{T}_\zeta^{-1}\right)^n \Delta\hat{Q}^n = \hat{R}^n \tag{10}$$

2) A flux-vector splitting algorithm reported by Steger et al.[24] is next. The two-factor implicit algorithm is obtained by using upwind differencing of the convective terms in the streamwise direction and central differencing in the radial and circumferential directions:

$$\left[I + h\delta_\xi^b(\hat{A}^+)^n + h\delta_\zeta \hat{C}^n - hRe^{-1}\bar{\delta}_\zeta J^{-1}\hat{M}^n J\right] \times \left[I + h\delta_\xi^f(\hat{A}^-)^n + h\delta_\eta \hat{B}^n\right]\Delta\hat{Q}^n = \hat{R}^n \tag{11}$$

where

$$\hat{R}^n = -h\left[\delta_\xi^b(\hat{E}^+)^n + \delta_\xi^f(\hat{E}^-)^n + \delta_\eta \hat{F}^n + \delta_\zeta \hat{G}^n - Re^{-1}\bar{\delta}_\zeta \hat{S}^n\right] \tag{12}$$

All of the described algorithms need implicit and explicit smoothing in order to suppress numerical oscillations and instabilities. In the flux-split algorithm and the Beam and Warming algorithm, second-order implicit smoothing is adopted to preserve the block tridiagonal form of the equations. The diagonal algorithm contains a combination of second- and fourth-order implicit smoothing. Given symmetric initial conditions, all smoothing-term contributions are symmetric.

Results

To provide a basis for comparison, flows were computed about a tangent ogive cylinder using computer codes implementing the described implicit algorithms. The body had a 3.0 diameter tangent ogive forebody and a 12.0 diameter cylindrical afterbody. It was considered important to carry out the computations under flow conditions for which the results could be compared with the results of experiments. A wind-tunnel model with the same body geometry was tested by Smith and Nunn.[25] Forces and moments were obtained at $Re_D = 5.0 \times 10^4$, $Re_D = 8.0 \times 10^4$, and $Re_D = 1.4 \times 10^5$, and angles of attack ranging from $\alpha = 0$ to $\alpha = 90$ deg.

Computations were carried out at angles of attack of $\alpha = 20$ and $\alpha = 40$ deg, Mach numbers of $M_\infty = 0.136$ and $M_\infty = 0.2$, with Reynolds number set at $Re_D = 8.0 \times 10^4$. The computational mesh consisted of $69 \times 121 \times 50$ points in the axial, circumferential, and radial directions, respectively. Except for one case (which was started from a converged solution) all of the computations were started from freestream initial conditions. A solution was considered converged if L_2 Norm, defined as the sum of the right-hand side over the grid points of the computational domain, dropped 3 orders

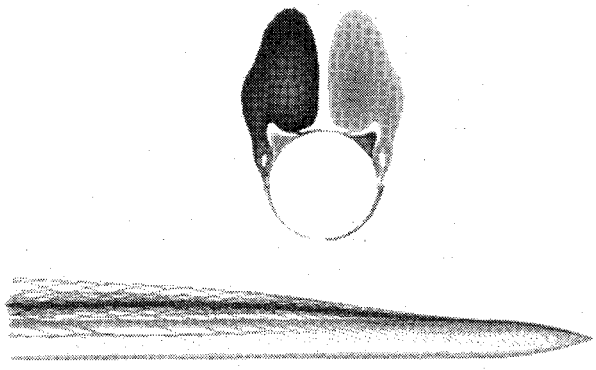


Fig. 1 Off-surface streamlines and helicity density contours at x/D = 11.4 around an ogive-cylinder body (flux-split algorithm); M_∞ = 0.136, α = 20 deg, and $Re_D = 8 \times 10^4$.

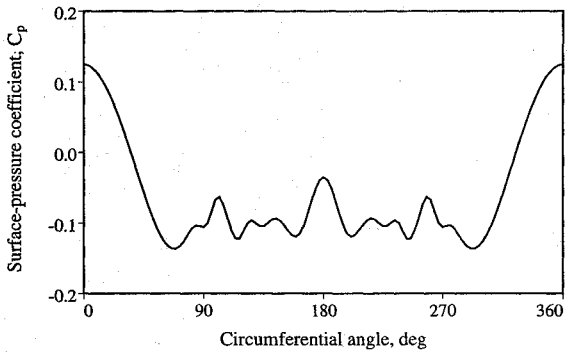


Fig. 2 Circumferential surface pressure distribution at x/D = 11.4 (flux-split algorithm); M_∞ = 0.136, α = 20 deg, and $Re_D = 8 \times 10^4$.

of magnitude for steady flows and 1.5–2 orders of magnitude for flows that exhibited slight unsteadiness. These generally accepted convergence criteria were chosen based on past experience of using the codes in question. A more detailed convergence history can be found in Ref. 26.

Ogive-Cylinder Flow at α = 20 Degrees

Two different numerical algorithms were used to simulate the flow at $\alpha = 20$ deg, the two-factor flux vector split algorithm and the diagonal algorithm. The flow conditions were $Re_D = 8.0 \times 10^4$ and $M_\infty = 0.2$ for the diagonal algorithm simulation, and $M_\infty = 0.136$ for the flux-split algorithm simulation. The solution obtained using the diagonal algorithm was already in hand, and the Mach number for the flux-split algorithm simulation was chosen to match that of the experiment.[25] As will be shown, the Mach number has no influence on flow asymmetry for these low-subsonic flow conditions. Both solutions were characterized by the existence of two primary and two secondary vortices on the leeward side of the body with some unsteadiness in the flow about the aft part of the body. The unsteadiness is reflected in oscillations of the computed normal- and side-force coefficients. The normal-force coefficients oscillate around mean values of $C_N \approx 2.74$ for the flux-split algorithm simulation and $C_N \approx 2.65$ for the diagonal algorithm simulation. Both mean values compare reasonably well with an experimentally measured normal-force coefficient of $C_N = 2.77$.

Figure 1 (flux-split algorithm simulation) shows the computed off-surface streamlines for $Re_D = 8.0 \times 10^4$ and $M_\infty = 0.136$, and computed helicity density[27] contours on a crossflow plane located at $x/D = 11.4$ for a dimensionless time of $\hat{t} = 95.0$ ($\hat{t} = ta_\infty/D$). The computed off-surface streamlines and helicity density contours show that the computed flowfield is symmetric. Further comparisons of some representative values of helicity density on opposite sides of the symmetry plane show that the differences are zero to machine accuracy. The computed circumferential surface pressure-coefficient distribution at $x/D = 11.4$ that is shown in Fig. 2 confirms that the flowfield is symmetric close to the body surface as well as in its wake.

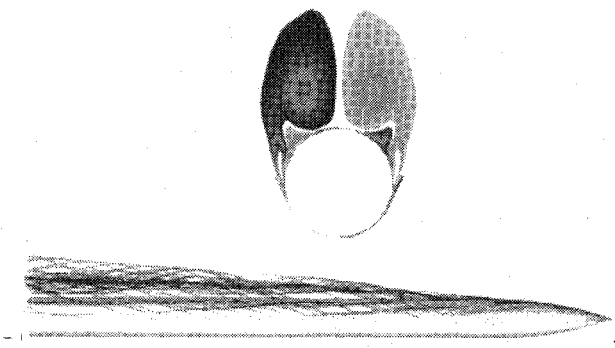


Fig. 3 Off-surface streamlines and helicity density contours at x/D = 11.4 around an ogive-cylinder body (diagonal algorithm); M_∞ = 0.2, α = 20 deg, and $Re_D = 8 \times 10^4$.

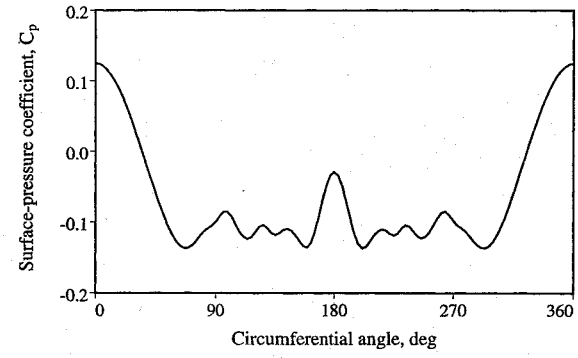


Fig. 4 Circumferential surface pressure distribution at x/D = 11.4 (diagonal algorithm); M_∞ = 0.2, α = 20 deg, and $Re_D = 8 \times 10^4$.

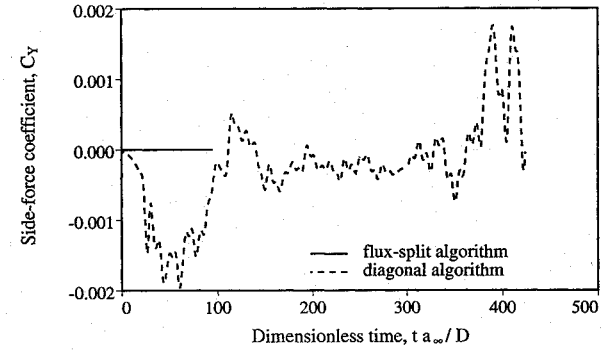
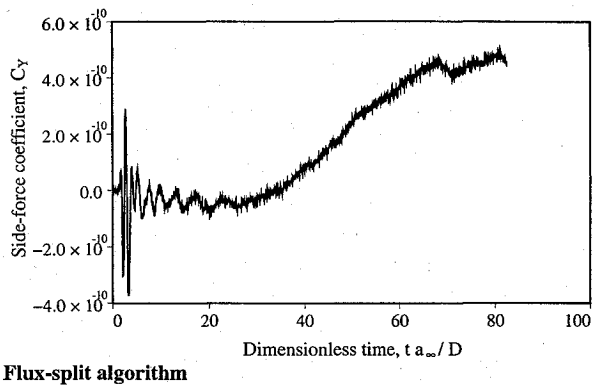


Fig. 5 Side-force coefficient history (flux-split algorithm M_∞ = 0.136, diagonal algorithm M_∞ = 0.2); α = 20 deg and $Re_D = 8 \times 10^4$.

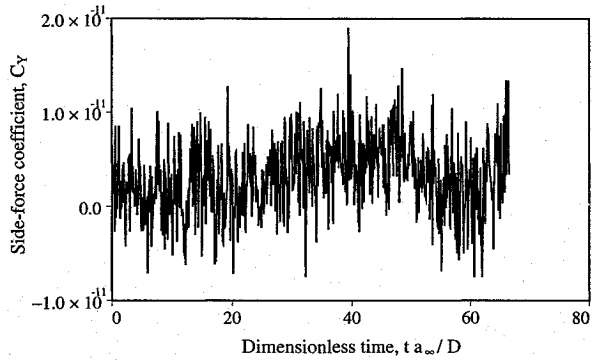
Figure 3 (diagonal algorithm simulation) shows the computed off-surface streamlines for $Re_D = 8.0 \times 10^4$ and $M_\infty = 0.2$ and computed helicity density contours at a crossflow plane located at $x/D = 11.4$ for a dimensionless time of $\hat{t} = 424.0$. Although the flowfield appears to be symmetric, numerical values of the computed helicity density show some difference in vortex strength between the left and right side of the body. The computed circumferential surface pressure-coefficient distribution ar $x/D = 11.4$ is shown in Fig. 4. Once again, the small differences between results for both sides of the body are not observable.

Figure 5 shows a comparison of the side-force coefficient time history of the solutions obtained. As a result of the symmetric flow the side-force coefficient of the solution obtained by the flux-split algorithm has oscillations of the order of machine zero for all time steps. The side-force coefficient of the flowfield obtained by using the diagonal algorithm has oscillations of ± 0.002. Those oscillations do not die away even after a dimensionless time of $\hat{t} = 424.0$. The origin of the differences in the orders of magnitude of side-force coefficient oscillations could be attributed either to differences in code accuracy or to asymmetry in the diagonal algorithm.

Recent study on the effect of geometrical disturbance on vortex asymmetry by Degani[10] shows that in the angle-of-attack range of



a) Flux-split algorithm



b) Beam and Warming algorithm

Fig. 6 Side-force coefficient history (flux-split algorithm $M_\infty = 0.136$, Beam and Warming algorithm $M_\infty = 0.2$); $\alpha = 40$ deg and $Re_D = 8 \times 10^4$.

$0 < \alpha < 30$ deg the introduction of a symmetry-breaking disturbance near the tip of the body has only a small effect on the symmetry of the flow. Furthermore, the study shows that in the angle-of-attack range of $30 < \alpha < 50$ deg the flowfield sensitivity to geometrical disturbances is larger and an asymmetric flowfield with a large side force is developed once the disturbance is introduced. Resolving the question of the origin of the asymmetry requires then that a computation of a flowfield be carried out for an angle of attack higher than 20 deg. An angle of attack $\alpha = 40$ deg was chosen.

Ogive-Cylinder Flow at $\alpha = 40$ Deg

Four different algorithms were used in this case: the flux-split algorithm, the diagonal algorithm, the Beam and Warming standard algorithm with the exclusion of the viscous Jacobian matrix from the left-hand side, and an algorithm using diagonalization only in the streamwise and radial directions as described in Eq. (10). By not including the viscous Jacobian matrix in the left-hand side of the Beam and Warming algorithm, the only difference between it and the diagonal version was the diagonalization, making the latter the only possible reason for different results. The flow conditions were $Re_D = 8.0 \times 10^4$ and $M_\infty = 0.2$, except for the flux-split algorithm simulation where the Mach number was set to $M_\infty = 0.136$ to match that of the experiment.[25]

The side-force-coefficient time history of the results obtained by both the flux-split algorithm and the Beam and Warming algorithm are presented in Figs. 6a and 6b, respectively. The difference between the plots reflects the different characteristics of the algorithms, but neither side-force coefficient ever grows beyond negligible values. For both cases, for the duration of the simulation, start up to convergence, the values of the side-force coefficients remain at machine zero. The computed off-surface streamlines and computed helicity density contours for the cross section at $x/D = 11.4$ are shown in Figs. 7 (flux-split algorithm, dimensionless time of $\hat{t} = 82.5$) and 8 (Beam and Warming algorithm, dimensionless time of $\hat{t} = 66.5$). Figures 9a and 9b show the computed circumferential surface pressure-coefficient distribution for both cases at the cross section $x/D = 11.4$. From Figs. 7–9, one can see that the strength of the primary and secondary vortices and the location of the primary and secondary separation lines are perfectly symmetric, thus confirming that both solutions are perfectly symmetric. Although the Mach number for the two cases was different the results are similar. The only differences are in the levels of the local pressure minima and maxima on the bodies' surfaces and in the levels of helicity density reflecting the different strengths of the vortices. A comparison of the computed normal-force coefficients with the experimentally measured value of $C_N = 11.42$ showed a large disparity. The computed values from both simulations oscillate around mean values of $C_N \approx 6.75$. The reason for the discrepancy is the difference of the asymmetry levels between the results of computations and the experiment. The computational results (obtained with symmetric boundary conditions) were symmetric whereas the experimental results exhibited asymmetric flow. Experimental results presented by Lamont[6] show that the normal-force coefficient is a function of the roll angle and, therefore, of the asymmetry level, thus providing a possible explanation for the differences.

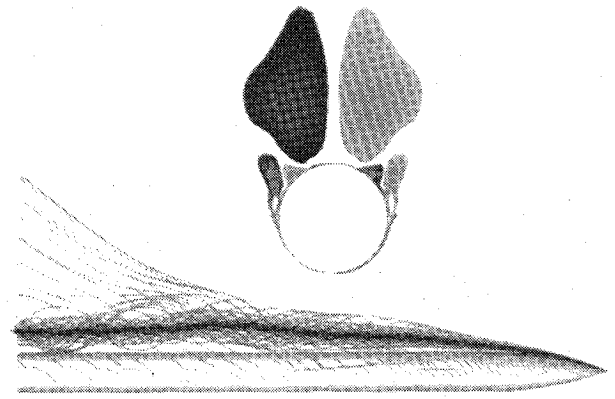


Fig. 7 Off-surface streamlines and helicity density contours at $x/D = 11.4$ around an ogive-cylinder body (flux-split algorithm); $M_\infty = 0.136$, $\alpha = 40$ deg, and $Re_D = 8 \times 10^4$.

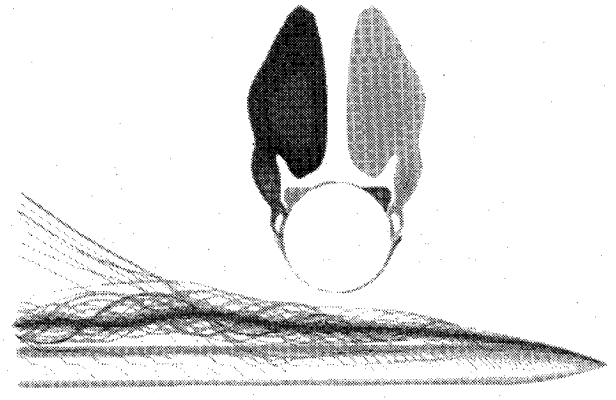
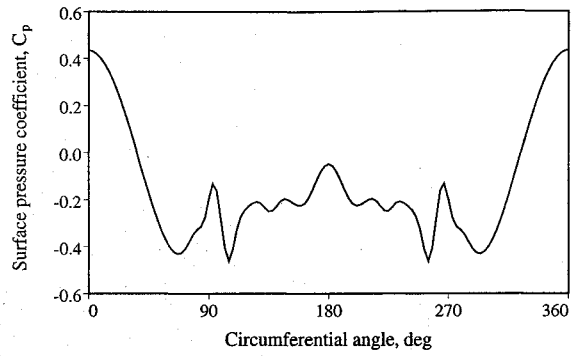
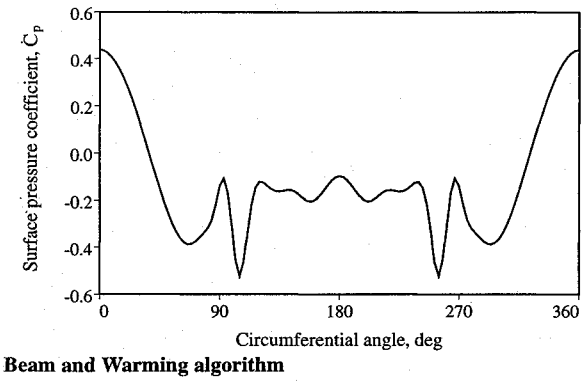


Fig. 8 Off-surface streamlines and helicity density contours at $x/D = 11.4$ around an ogive-cylinder body (Beam and Warming algorithm); $M_\infty = 0.2$, $\alpha = 40$ deg, and $Re_D = 8 \times 10^4$.

The side-force-coefficient time history of the results obtained by the diagonal algorithm is presented in Fig. 10. The flow conditions were identical to those of the solution obtained by using the original Beam and Warming algorithm. In contrast to the $\alpha = 20$-deg case, the side-force oscillations are relatively large. At one point of the computation ($\hat{t} = 164$) the side-force coefficient reaches a value of $C_Y \approx 1.07$. Figure 11 shows the computed off-surface streamlines and computed helicity density contours for the cross section at $x/D = 11.4$ at time $\hat{t} = 164$. The computed off-surface streamlines show a little asymmetry especially in the wake around the aft part of the body. The helicity density contours show some asymmetry near the body surface; in particular, the secondary vortices differ in shape and strength. One would expect that since the asymmetry in the wake is relatively small the resulting side force would also be small, but the time history of the side-force coefficient shows the contrary. The size of the side-force coefficient indicates that although the asymmetry in the wake is small, the wall values of the flow variables (especially the surface pressure coefficient) differ



a) Flux-split algorithm



b) Beam and Warming algorithm

Fig. 9 Circumferential surface pressure distribution at x/D = 11.4 (flux-split algorithm M_∞ = 0.136, Beam and Warming algorithm M_∞ = 0.2); α = 40 deg and $Re_D = 8 \times 10^4$.

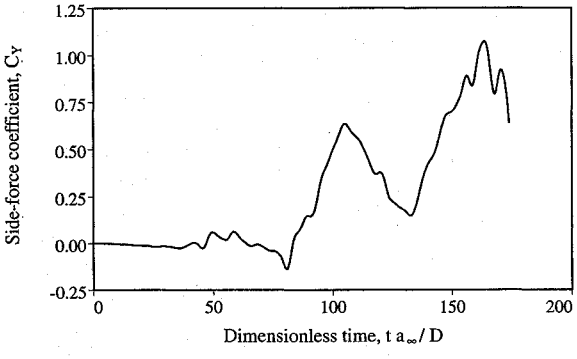


Fig. 10 Side-force coefficient history (diagonal algorithm); M_∞ = 0.2, α = 40 deg, and $Re_D = 8 \times 10^4$.

by large amounts between points on either side of the symmetry plane. Figure 12 shows the computed circumferential surface pressure-coefficient distribution for the cross section at $x/D = 11.4$. The differences are more evident from that figure. The strengths of the suction peaks created by primary and secondary vortices differ substantially. Moreover, an extra suction peak appears on one side of the body, and the surface pressure behavior indicates the existence of other spurious secondary structures. The fact that a rather large pressure difference exists between both sides of the body even though the apparent difference between the vortices is small stands in contradiction to the general belief that the large side forces are generated by the large differences in the vortices and cries out for an explanation. The discussion section contains an analysis of the algorithms and points to the differences that cause the spurious asymmetry.

To locate the origin of the asymmetry, another computation was conducted. The hybrid algorithm described in Eq. (10) was used with the flow-conditions set at $Re_D = 8.0 \times 10^4$ and $M_\infty = 0.2$. The motivation was the assumption that the diagonalization of the circumferential factor was the lone cause of the asymmetry. The development of the side force and the small asymmetry of the wake were delayed, but the trends were similar indicating that every one of the diagonalized factors contributes to the error.

Following the procedure recommended by Degani,[10] a geometrical disturbance was added near the nose tip of the model, and a

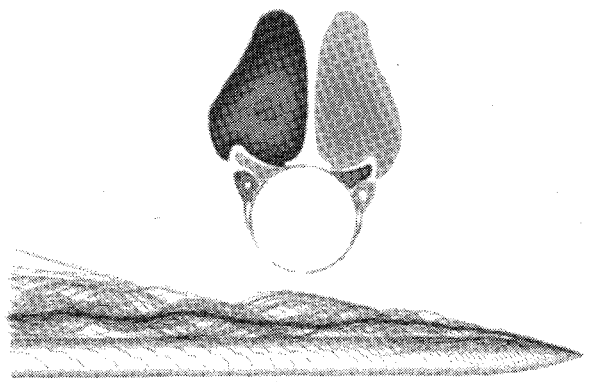


Fig. 11 Off-surface streamlines and helicity density contours at x/D = 11.4 around an ogive-cylinder body (diagonal algorithm); M_∞ = 0.2, α = 40 deg, and $Re_D = 8 \times 10^4$.

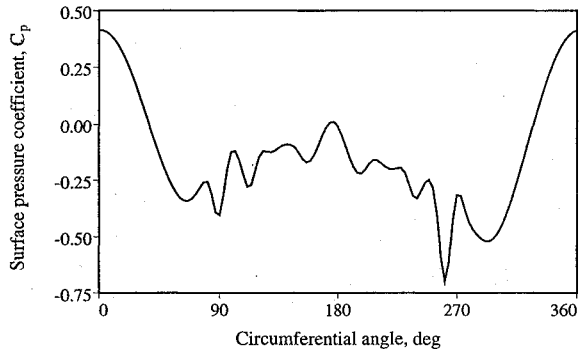


Fig. 12 Circumferential surface pressure distribution at x/D = 11.4 (diagonal algorithm); M_∞ = 0.2, α = 40 deg, and $Re_D = 8 \times 10^4$.

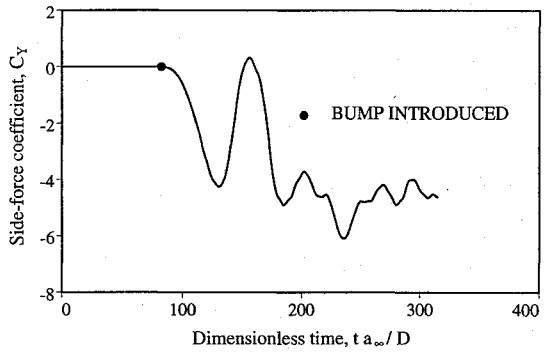


Fig. 13 Side-force coefficient history (flux-split algorithm); M_∞ = 0.136, α = 40 deg, and $Re_D = 8 \times 10^4$.

computation using the flux-split algorithm was continued from a converged solution. Figure 13 shows the side-force-coefficient time history. The side-force coefficient departs from zero and after a few oscillations reaches an equilibrium state consisting of a nonzero mean value of $C_Y \approx -4.5$ and small amplitude fluctuations. The introduction of the disturbance and the subsequent asymmetry has influenced the normal-force coefficient as well. The normal-force coefficient oscillates around $C_N \approx 10.0$, confirming that both normal and side force are functions of the asymmetry level. These confirmations are on a qualitative basis only since the experimental results are limited. Quantitative comparisons of computational results obtained with the use of the flux-split algorithm and experimental results are given in Ref. 28. In this paper, results from computational simulations of flows at angles of attack of $\alpha = 30$ and $\alpha = 40$ deg and Reynolds numbers of $Re_D = 3.0 \times 10^6$ and $Re_D = 4.0 \times 10^6$ were compared with experimental results. A very good agreement between computed and measured surface pressure distributions was demonstrated. Figure 14 shows the computed off-surface streamlines and computed helicity density contours. The asymmetry is large and resembles that observed in experiments.[2,4,5,7] Figure 15

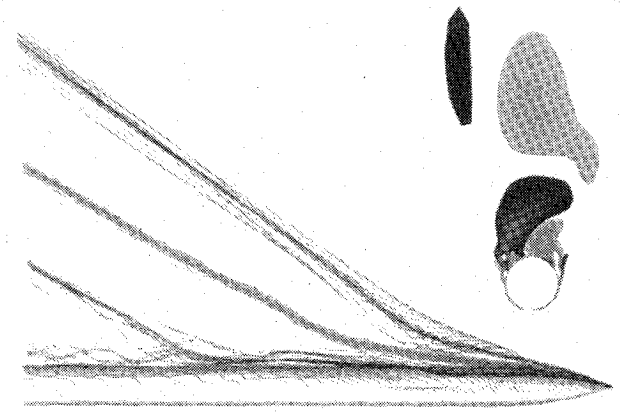


Fig. 14 Off-surface streamlines and helicity density contours at x/D = 11.4 around an ogive-cylinder body with a bump near the tip (flux-split algorithm); M_∞ = 0.136, α = 40 deg, and $Re_D = 8 \times 10^4$.

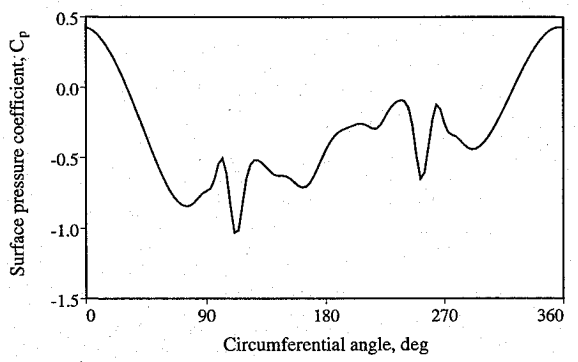


Fig. 15 Circumferential surface pressure distribution at x/D = 11.4 (flux-split algorithm); M_∞ = 0.136, α = 40 deg, and $Re_D = 8 \times 10^4$.

shows the computed circumferential surface pressure-coefficient distribution for the cross section at $x/D = 11.4$. In contrast to the behavior of the solution obtained using the diagonal algorithm (Fig. 12), the pattern of the surface pressure distribution reflects the asymmetry in the wake and the peaks reflect the effects of the primary and secondary vortices.

Discussion and Conclusions

The fact that results obtained by different computer codes contain differences is not surprising. The differences emerge primarily from the different algorithms implemented in the codes. For example, the right-hand side of the flux-split algorithm is different from the other algorithms used in the current work. Other differences may be the result of the different smoothings used for stability.

If one adopts the assumption used by Pulliam and Chaussee[19] that the steady-state solution is unique and is determined by the right-hand side of the algorithm only, and if the explicit smoothing is also identical, the converged solutions should show no substantial differences between each other. The diagonal algorithm results and the original Beam and Warming algorithm results were obtained by using the exact-same code with identical right-hand sides. The only differences were on the left-hand side due to the different inversion methods. In contrast with the aforementioned assumptions, the results obtained for high-angle-of-attack flows around slender bodies using a code implementing the diagonal algorithm were different in nature from the results obtained by using the Beam and Warming algorithm or the flux-split algorithm. The most significant difference was the existence in the former results of a rather large asymmetry while using a symmetric grid, symmetric boundary conditions, and no geometrical perturbations. Although the differences between flux-split algorithm results and the diagonal algorithm results could be attributed to the large differences between the algorithms, the differences between the diagonal algorithm and the original Beam and Warming results indicate that for high angle-of-attack flows the assumption that the solution depends only on the right-hand side does not hold.

The asymmetry developed by using the diagonal algorithm shows that the left-hand side of the algorithm affects the results at high-angle-of-attack regimes where asymmetry was experimentally observed. At moderate angles of attack, ($\alpha \leq 20$ deg), the asymmetry as well as the side-force coefficient associated with it are small, and the solutions obtained are comparable to experimental results or solutions obtained with other algorithms. At higher angles of attack, ($\alpha \geq 30$ deg), the asymmetry is large enough to be detected and the side force is of the order of the normal force, but the solution fails to conform to the experimentally observed pattern of vortices that leave the body from alternate sides and curve away from the body to virtually align with the freestream.[2]

Previous simulations for laminar and turbulent flows[10,28] showed that for the higher angle-of-attack regime a geometrical perturbation was required in order to achieve asymmetry similar to that observed in experiments and that the asymmetrical flow returned to a symmetric state when the perturbation was removed. On that basis it was suggested that the instability was of the convective type.

All the results suggest that the diagonalization process breaks the symmetry-preserving properties of the original Beam and Warming algorithm and, thus, initiates an asymmetry in the flow that is spurious. The following analysis reveals how the diagonalization process breaks the symmetry-preserving properties of the original algorithm and prevents the solution from maintaining its symmetry.

The symmetry of a flow can be represented by a set of conditions on the flow variables. All scalar state variables must be even functions with respect to a symmetry plane, in this work the plane $y = 0$ (xz plane) or $\eta = 1$ in the computational space. In addition, components of vector variables that are parallel to the symmetry plane must be even functions of y, whereas components that are normal to that plane must be odd functions of y. These conditions mean that, except for the velocity component in the y-direction coordinate, all flow variables are even functions with respect to y.

To reveal the properties of the Beam and Warming algorithm, one needs to analyze both the right- and left-hand sides of Eq. (3). The right-hand side is a three-dimensional array of the size of the computational domain, where each of the elements is a set of five functions, each function corresponding to one flow variable. It is easily seen that, for symmetric initial flow conditions, these functions obey the same rules as are obeyed by the flow variables. Except for the third function, which corresponds to the momentum in the η-direction coordinate and which is odd, the remaining four functions are even.

The next step is to analyze the inversion process. A symmetry-preserving scheme will produce a $\Delta\hat{Q}^n$ that obeys the same symmetry rules as are obeyed by the initial flow variables. The inversion process is composed of series of three one-dimensional inversions, and each of these must preserve symmetry in order to have a symmetric right-hand side for the next inversion.

The circumferential factor is represented as a single-matrix operator in Eq. (A1). The implicit periodic boundary condition that is implemented in that direction results in a series of periodic linear systems. Each of the inversions corresponds to a circle around the body. Therefore, points from opposite sides of the symmetry plane are included in each inversion. Equations corresponding to points that are on opposite sides of the symmetry plane must obey the rules of symmetry. For example, the equation corresponding to $\eta = 2$, the second row in Eq. (A1), has to be a mirror image of the equation corresponding to $\eta = k$, the last row in Eq. (A1). (The points $\eta = 2$ and $\eta = k$ lie on opposite sides of the symmetry plane.) Extending this argument to all circumferential points results in a system of equations obeying a set of symmetry-preserving relations. For example, the values of the second and last row elements of the first column of the periodic block matrix are the opposites of each other.

The symmetry-preserving relations, applied to the linear systems, are necessary but not sufficient conditions for a symmetric solution of the periodic factor. Additional conditions must be met by the elements of the submatrices, in order to guarantee a symmetric solution. Denoting the elements of $\hat{B}$ as $\hat{b}_{ij}$ and applying the symmetry

relations, one sees that for all elements where either $i \neq 3$ or $j \neq 3$, $\hat{b}_{ij}$ are odd functions of y, whereas the rest are even functions ($\hat{b}_{33}$ is odd).

The properties of the operators associated with the Beam and Warming algorithm that are described herein are the ones that preserve the symmetry in the case of a symmetric initial condition. The series of circumferential inversions then results in another three-dimensional array that obeys the conditions for symmetry. This array is the correction term that is used to advance the solution or to supply the source term for the next series of inversions. The analysis of the axial and radial directions is similar and can be found in Ref. 26.

The analysis of the diagonal algorithm is also based on symmetric initial conditions, and since the right-hand side of the algorithm is identical to the right-hand side of the Beam and Warming algorithm, the first part of the analysis holds for both algorithms. However, due to the diagonalization, the form of the left-hand side largely differs from the left-hand side of the original algorithm. Consequently, the inversion process is different, and a few clarifications are needed before the analysis can be carried out.

Following the results of the analysis of the Beam and Warming algorithm, one expects that the eigensystems would behave in a manner similar to that of the Jacobian matrices associated with them. However, application of the symmetry condition to the eigenvalue and eigenvector matrices [Eqs. (B2) and (B3) in Appendix B] reveals that there is no consistent pattern of even and odd functions. Moreover, some of the functions are neither odd nor even.

The inversion process itself is composed of nine steps that for clarity can be grouped into three factors, each corresponding to one coordinate direction. The factors are composed of three matrices, a block-diagonal eigenvector matrix, a diagonal-block tridiagonal matrix, and a block-diagonal inverse-eigenvector matrix [see Eqs. (9) and (A2)]. The factor corresponding to the circumferential direction can also be represented as a single-matrix operator [Eq. (A3)] that is equivalent to the Beam and Warming single-matrix operator [Eq. (A1)], thus allowing a direct comparison between the two schemes. Although the inversion process is carried out as described in the text, the solution for the inversion process is identical to the solution for the single-matrix operator.

Written in a single-matrix form, the inversion process of the diagonal algorithm can then be analyzed. Examination of the operator reveals that the matrix lacks the set of symmetry-preserving relations that exist in the matrix representing the circumferential factor of the Beam and Warming algorithm. For example, the second and last elements of the first column are not the opposites of each other because $\hat{T}_2 \neq \hat{T}_k$. Therefore, the differences between equations that correspond to points on opposite sides of the symmetry plane would lead to flow asymmetry even if the eigenvectors themselves obeyed the symmetry conditions.

The preceding analysis has shown that the diagonalization process breaks the symmetry-preserving properties of the original Beam and Warming algorithm and thus initiates an asymmetry in the flow that is spurious. Every time an inversion is performed, in any cross section, at any radial distance from the body, and in each of the three factors, a small asymmetric error is introduced. The presence of that numerical error affects different flows in different ways. For low-angle-of-attack flows, the error is similar in behavior to that of round-off error and it does not grow. For high-angle-of-attack flows where instability mechanisms exist, the error is similar in behavior to that of a distribution of geometrical perturbations. The perturbations change their position between time steps, and they extend over the whole flowfield, as opposed to being located only on the body. As a result, the developed flowfield becomes asymmetric in the absence of a real geometrical perturbation, but the asymmetric pattern that emerges fails to conform to the pattern that is observed in experiments. It is, in fact, spurious.

The analyses of the symmetry-preserving properties of the diagonal form and the original form of the Beam and Warming algorithm provide a clear explanation for the results presented in this work. It was demonstrated that a certain algorithm, in this case the diagonal algorithm, contains an inherent bias that triggers a spurious asymmetric flow. This fact should alert researchers who use numerical algorithms that include diagonalization for the simulation of sensitive flows to the possibility of similar biases. It should also encourage a search for possible numerical biases in other innocent looking algorithms that produce solutions with preferred orientations without the introduction of disturbances.

Appendix A: Circumferential Factors in Matrix Form

In reference to Eq. (3) of the text a periodic factor of the Beam and Warming algorithm has the form

$$\begin{bmatrix} I & \frac{h}{2}\hat{B}_2^n & 0 & \dots & \dots & 0 & -\frac{h}{2}\hat{B}_k^n \\ -\frac{h}{2}\hat{B}_1^n & I & \frac{h}{2}\hat{B}_3^n & 0 & \dots & \dots & 0 \\ 0 & -\frac{h}{2}\hat{B}_2^n & I & \frac{h}{2}\hat{B}_4^n & 0 & \dots & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ 0 & \dots & 0 & -\frac{h}{2}\hat{B}_{k-3}^n & I & \frac{h}{2}\hat{B}_{k-1}^n & 0 \\ 0 & \dots & \dots & 0 & -\frac{h}{2}\hat{B}_{k-2}^n & I & \frac{h}{2}\hat{B}_k^n \\ \frac{h}{2}\hat{B}_1^n & 0 & \dots & \dots & 0 & -\frac{h}{2}\hat{B}_{k-1}^n & I \end{bmatrix} \begin{Bmatrix} \Delta\hat{Q}_1 \\ \Delta\hat{Q}_2 \\ \Delta\hat{Q}_3 \\ \vdots \\ \Delta\hat{Q}_{k-2} \\ \Delta\hat{Q}_{k-1} \\ \Delta\hat{Q}_k \end{Bmatrix} = \begin{Bmatrix} \vdots \\ \vdots \\ \vdots \\ \text{RHS} \\ \vdots \\ \vdots \\ \vdots \end{Bmatrix} \tag{A1}$$

In reference to Eq. (9) of the text a periodic factor of the diagonal algorithm has the form

$$\begin{bmatrix} \hat{T}_1^n & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \hat{T}_2^n & 0 & \dots & \dots & \dots & 0 \\ 0 & 0 & \hat{T}_3^n & 0 & \dots & \dots & 0 \\ 0 & \dots & 0 & \ddots & 0 & \dots & 0 \\ 0 & \dots & \dots & 0 & \hat{T}_{k-2}^n & 0 & 0 \\ 0 & \dots & \dots & \dots & 0 & \hat{T}_{k-1}^n & 0 \\ 0 & \dots & \dots & \dots & \dots & 0 & \hat{T}_k^n \end{bmatrix} \begin{bmatrix} I & h\hat{\Lambda}_2^n & 0 & \dots & \dots & 0 & -h\hat{\Lambda}_k^n \\ -h\hat{\Lambda}_1^n & I & h\hat{\Lambda}_3^n & 0 & \dots & \dots & 0 \\ 0 & -h\hat{\Lambda}_2^n & I & h\hat{\Lambda}_4^n & 0 & \dots & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ 0 & \dots & 0 & -h\hat{\Lambda}_{k-3}^n & I & h\hat{\Lambda}_{k-1}^n & 0 \\ 0 & \dots & \dots & 0 & -h\hat{\Lambda}_{k-2}^n & I & h\hat{\Lambda}_k^n \\ h\hat{\Lambda}_1^n & 0 & \dots & \dots & 0 & -h\hat{\Lambda}_{k-1}^n & I \end{bmatrix}$$

$$\begin{bmatrix} (\hat{T}_1^{-1})^n & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & (\hat{T}_2^{-1})^n & 0 & \dots & \dots & \dots & 0 \\ 0 & 0 & (\hat{T}_3^{-1})^n & 0 & \dots & \dots & 0 \\ 0 & \dots & 0 & \ddots & 0 & \dots & 0 \\ 0 & \dots & \dots & 0 & (\hat{T}_{k-2}^{-1})^n & 0 & 0 \\ 0 & \dots & \dots & \dots & 0 & (\hat{T}_{k-1}^{-1})^n & 0 \\ 0 & \dots & \dots & \dots & \dots & 0 & (\hat{T}_k^{-1})^n \end{bmatrix} \begin{Bmatrix} \Delta\hat{Q}_1 \\ \Delta\hat{Q}_2 \\ \Delta\hat{Q}_3 \\ \vdots \\ \Delta\hat{Q}_{k-2} \\ \Delta\hat{Q}_{k-1} \\ \Delta\hat{Q}_k \end{Bmatrix} = \begin{Bmatrix} \vdots \\ \vdots \\ \vdots \\ \text{RHS} \\ \vdots \\ \vdots \\ \vdots \end{Bmatrix} \tag{A2}$$

Written in a block form a periodic factor of the diagonal algorithm is

$$\begin{bmatrix} I & \frac{h}{2}\hat{T}_1\hat{\Lambda}_2\hat{T}_2^{-1} & 0 & \dots & \dots & 0 & -\frac{h}{2}\hat{T}_1\hat{\Lambda}_k\hat{T}_k^{-1} \\ -\frac{h}{2}\hat{T}_2\hat{\Lambda}_1\hat{T}_1^{-1} & I & \frac{h}{2}\hat{T}_2\hat{\Lambda}_3\hat{T}_3^{-1} & 0 & \dots & \dots & 0 \\ 0 & -\frac{h}{2}\hat{T}_3\hat{\Lambda}_2\hat{T}_2^{-1} & I & \frac{h}{2}\hat{T}_3\hat{\Lambda}_4\hat{T}_4^{-1} & 0 & \dots & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & 0 & 0 \\ 0 & \dots & 0 & -\frac{h}{2}\hat{T}_{k-2}\hat{\Lambda}_{k-3}\hat{T}_{k-3}^{-1} & I & \frac{h}{2}\hat{T}_{k-2}\hat{\Lambda}_{k-1}\hat{T}_{k-1}^{-1} & 0 \\ 0 & \dots & \dots & 0 & -\frac{h}{2}\hat{T}_{k-1}\hat{\Lambda}_{k-2}\hat{T}_{k-2}^{-1} & I & \frac{h}{2}\hat{T}_{k-1}\hat{\Lambda}_k\hat{T}_k^{-1} \\ \frac{h}{2}\hat{T}_k\hat{\Lambda}_1\hat{T}_1^{-1} & 0 & \dots & \dots & 0 & -\frac{h}{2}\hat{T}_k\hat{\Lambda}_{k-1}\hat{T}_{k-1}^{-1} & I \end{bmatrix}$$

$$\times \begin{Bmatrix} \Delta\hat{Q}_1 \\ \Delta\hat{Q}_2 \\ \Delta\hat{Q}_3 \\ \vdots \\ \Delta\hat{Q}_{k-2} \\ \Delta\hat{Q}_{k-1} \\ \Delta\hat{Q}_k \end{Bmatrix} = \begin{Bmatrix} \vdots \\ \vdots \\ \vdots \\ \text{RHS} \\ \vdots \\ \vdots \\ \vdots \end{Bmatrix} \tag{A3}$$

Appendix B: Circumferential Jacobian Matrix and its Eigensystem

The elements of the Jacobian matrix $\hat{B}$ appearing in Eqs. (3) and (5) of the text are

$$\hat{B} = \begin{bmatrix} \eta_t & \eta_x & \eta_y & \eta_z & 0 \\ \eta_x\phi^2 - u\theta & \eta_t + \theta - \eta_x(\gamma-2)u & \eta_y u - (\gamma-1)\eta_x v & \eta_z u - (\gamma-1)\eta_x w & \eta_x(\gamma-1) \\ \eta_y\phi^2 - v\theta & \eta_x v - \eta_y(\gamma-1)u & \eta_t + \theta - \eta_y(\gamma-2)v & \eta_z v - (\gamma-1)\eta_y w & \eta_y(\gamma-1) \\ \eta_z\phi^2 - w\theta & \eta_x w - \eta_z(\gamma-1)u & \eta_y w - \eta_z(\gamma-1)v & \eta_t + \theta - \eta_z(\gamma-2)w & \eta_z(\gamma-1) \\ \theta\left(2\phi^2 - \frac{\gamma e}{\rho}\right) & \eta_x\left(\frac{\gamma e}{\rho} - \phi^2\right) - (\gamma-1)u\theta & \eta_y\left(\frac{\gamma e}{\rho} - \phi^2\right) - (\gamma-1)v\theta & \eta_z\left(\frac{\gamma e}{\rho} - \phi^2\right) - (\gamma-1)w\theta & \eta_t + \gamma\theta \end{bmatrix} \tag{B1}$$

where

$$\phi^2 = 0.5(\gamma-1)(u^2+v^2+w^2), \qquad \theta = \eta_x u + \eta_y v + \eta_z w$$

The eigenvalue matrix has elements

$$\hat{\Lambda}_\eta = D\left[\, V,\ V,\ V,\ V + c\left(\eta_x^2+\eta_y^2+\eta_z^2\right)^{\frac{1}{2}},\qquad V \quad -c\left(\eta_x^2+\eta_y^2+\eta_z^2\right)^{\frac{1}{2}} \,\right] \tag{B2}$$

The eigenvector matrix has elements

$$\hat{T}_\eta = \begin{bmatrix} \tilde{\eta}_x & \tilde{\eta}_y & \tilde{\eta}_z & \alpha & \alpha \\ \tilde{\eta}_x u & \tilde{\eta}_y u - \tilde{\eta}_z \rho & \tilde{\eta}_z u + \tilde{\eta}_y \rho & \alpha(u + \tilde{\eta}_x c) & \alpha(u - \tilde{\eta}_x c) \\ \tilde{\eta}_x v + \tilde{\eta}_z \rho & \tilde{\eta}_y v & \tilde{\eta}_z v - \tilde{\eta}_x \rho & \alpha(v + \tilde{\eta}_y c) & \alpha(v - \tilde{\eta}_y c) \\ \tilde{\eta}_x w - \tilde{\eta}_y \rho & \tilde{\eta}_y w + \tilde{\eta}_x \rho & \tilde{\eta}_z w & \alpha(w + \tilde{\eta}_z c) & \alpha(w - \tilde{\eta}_z c) \\ \tilde{\eta}_x \dfrac{\phi^2}{\gamma - 1} + \rho(\tilde{\eta}_z v - \tilde{\eta}_y w) & \tilde{\eta}_y \dfrac{\phi^2}{\gamma - 1} + \rho(\tilde{\eta}_x w - \tilde{\eta}_z u) & \tilde{\eta}_z \dfrac{\phi^2}{\gamma - 1} + \rho(\tilde{\eta}_y u - \tilde{\eta}_x v) & \alpha\left(\dfrac{\phi^2 + c^2}{\gamma - 1} + c\tilde{\theta}\right) & \alpha\left(\dfrac{\phi^2 + c^2}{\gamma - 1} - c\tilde{\theta}\right) \end{bmatrix} \tag{B3}$$

where

$$\tilde{\eta}_x = \frac{\eta_x}{\left(\eta_x^2 + \eta_y^2 + \eta_z^2\right)^{\frac{1}{2}}}, \qquad \tilde{\eta}_y = \frac{\eta_y}{\left(\eta_x^2 + \eta_y^2 + \eta_z^2\right)^{\frac{1}{2}}}$$

$$\tilde{\eta}_z = \frac{\eta_z}{\left(\eta_x^2 + \eta_y^2 + \eta_z^2\right)^{\frac{1}{2}}}$$

$$\tilde{\theta} = \tilde{\eta}_x u + \tilde{\eta}_y v + \tilde{\eta}_z w, \qquad \alpha = \rho/\sqrt{2c} \tag{B4}$$

Acknowledgments

This work has been supported in part by NASA Grant NCA2-765. The work of the third author was supported by the Israel Science Foundation administered by the Israel Academy of Sciences and Humanities.

References

[1]Allen, H. J., and Perkins, E. W., "A Study of Effects of Viscosity on Flow Over Slender Inclined Bodies of Revolution," NACA Rept. 1048, 1951.

[2]Thomson, K. D., and Morrison, D. F., "The Spacing, Position and Strength of Vortices in the Wake of Slender Cylindrical Bodies at Large Incidence," *Journal of Fluid Mechanics*, Vol. 50, Dec. 1971, pp. 751–783.

[3]Hunt, B. L., and Dexter, P. C., "Pressures on a Slender Body at High Angle of Attack in a Very Low Turbulence Level Airstream," AGARD CP 247, Paper 17, 1978.

[4]Dexter, P. C., and Hunt, B. L., "The Effect of Roll Angle on the Flow over a Slender Body of Revolution at High Angles of Attack," AIAA Paper 81-0358, Jan. 1981.

[5]Hunt, B. L., "Asymmetric Vortex Forces and Wakes on Slender Bodies," AIAA Paper 82-1336, Aug. 1982.

[6]Lamont, P. J., "Pressures Around an Inclined Ogive Cylinder with Laminar, Transitional, or Turbulent Separation," *AIAA Journal*, Vol. 20, No. 5, 1982, pp. 1492–1499.

[7]Zilliac, G. G., Degani, D., and Tobak, M., "Asymmetric Vortices on a Slender Body of Revolution," *AIAA Journal*, Vol. 29, 1991, pp. 667–675.

[8]Degani, D., and Tobak, M., "Experimental Study of Controlled Tip Disturbance Effect on Flow Asymmetry," *Physics of Fluids A*, Vol. 4, No. 12, 1992, pp. 2825–2832.

[9]Williams, D., and Bernhardt, J., "Proportional Control of Asymmetric Forebody Vortices with the Unsteady Bleed Technique," AIAA Paper 90-1629, June 1990.

[10]Degani, D., "Effect of Geometrical Disturbance on Vortex Asymmetry," *AIAA Journal*, Vol. 29, No. 4, 1991, pp. 560–566.

[11]Thomas, J. L., "Reynold Number Effects on Supersonic Asymmetric Flow Over a Cone," *Journal of Aircraft*, Vol. 30, No. 4, 1993, pp. 488–495.

[12]Degani, D., and Tobak, M., "Effect of Upstream Disturbance on Flow Asymmetry," AIAA Paper 92-0408, Jan. 1992.

[13]Degani, D., and Tobak, M., "Numerical Simulation of Upstream Disturbance on Flows Around a Slender Body," AIAA Paper 93-2956, July 1993.

[14]Degani, D., "Numerical Investigation of the Origin of Vortex Asymmetry," AIAA Paper 90-0593, Jan. 1990.

[15]Degani, D., and Schiff, L. B., "Numerical Simulation of the Effect of Spatial Disturbances on Vortex Asymmetry," *AIAA Journal*, Vol. 29, No. 3, 1991, pp. 344–352.

[16]Siclari, M. J., and Marconi, F., "The Computation of Navier–Stokes Solutions Exhibiting Asymmetric Vortices," AIAA Paper 89-1817, June 1989.

[17]Kandil, O. A., Wong, T. C., and Liu, C. H., "Navier–Stokes Computations of Symmetric and Asymmetric Vortex Shedding Around Slender Bodies," AIAA Paper 89-3397, Aug. 1989.

[18]Hartwich, P. M., Hall, R. M., and Hemsch, M. J., "Navier–Stokes Computations of Vortex Asymmetries Controlled by Small Surface Imperfections," AIAA Paper 90-0385, Jan. 1990.

[19]Pulliam, T. H., and Chaussee, D. S., "A Diagonal Form of an Implicit Approximate-Factorization Algorithm," *Journal of Computational Physics*, Vol. 39, No. 2, 1981, pp. 347–363.

[20]Baldwin, B. S., and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper 78-257, Jan. 1978.

[21]Steger, J. L., "Implicit Finite-Difference Simulation of Flow About Arbitrary Two-Dimensional Geometries," *AIAA Journal*, Vol. 16, No. 7, 1978, pp. 679–686.

[22]Beam, R. M., and Warming, R. F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation Law Form," *Journal of Computational Physics*, Vol. 22, No. 9, 1976, pp. 87–110.

[23]Beam, R. M., and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier–Stokes Equations," *AIAA Journal*, Vol. 16, No. 4, 1978, pp. 393–402.

[24]Steger, J. L., Ying, S. X., and Schiff, L. B., "A Partially Flux-Split Algorithm for Numerical Simulation of Unsteady Viscous Flows," Proceedings of a Workshop on Computational Fluid Dynamics, Univ. of California, Davis, CA, June 1986.

[25]Smith, L. H., and Nunn, R. H., "Aerodynamic Characteristics of an Axisymmetric Body Undergoing a Uniform Pitching Motion," AIAA Paper 75-838, June 1975.

[26]Levy, Y., "Numerical Analysis of Three-Dimensional Flow Solvers Applied to High-Angle-of-Attack Flows," Ph.D. Thesis, Dept. of Aero. & Astro. SUDAAR 652, Stanford Univ., Stanford, CA, May 1994.

[27]Levy, Y., Degani, D., and Seginer, A., "Graphical Visualization of Vortical Flows by Means of Helicity," *AIAA Journal*, Vol. 28, No. 8, 1990, pp. 1347–1352.

[28]Degani D., and Levy Y., "Asymmetric Turbulent Vortical Flows over Slender Bodies," *AIAA Journal*, Vol. 30, No. 9, 1992, pp. 2267–2273.