Fluid Flow around a thin airfoil

Lance Simms

December 12, 2005

1 Introduction

The Euler equation of motion for fluid flow is obtained from the Navier-Stokes equation by setting the viscosity equal to zero. While this simplifies the analysis of the fluid motion somewhat, the equations are still non-linear, and a closed-form solution to them does not exist. For flow about a slender body in two dimensions, they are written as

$$\frac{\partial Q}{\partial t} + \frac{\partial E(Q)}{\partial x} + \frac{\partial F(Q)}{\partial y} = 0$$

with

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}, \quad E = \begin{pmatrix} \rho \\ \rho u^2 + \rho \\ \rho uv \end{pmatrix}, \quad F = \begin{pmatrix} \rho \\ \rho uv \\ \rho v^2 + u \end{pmatrix}$$

Here we will consider the flow past a thin airoil, so these equations are applicable. For the problem at hand, we assume that the flow is uniform ($\rho = 1, u = M_\infty, v = 0$) at inflow, and we use it as the referene state for local linearization. As in the project description, we rewrite the equations in quasi-linear form, using e.g.

$$\frac{\partial E}{\partial x} = \frac{\partial E}{\partial Q}\frac{\partial Q}{\partial x} = A\frac{\partial Q}{\partial x}$$

which gives us

$$\frac{\partial Q}{\partial t} + A\frac{\partial Q}{\partial x} + B\frac{\partial Q}{\partial y} = 0$$

where

$$A = \begin{pmatrix} 0 & 1 & 0 \\ -u^2 + 1 & 2u & 0 \\ -uv & v & u \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 & 1 \\ -uv & v & u \\ -v^2 + 1 & 0 & 2v \end{pmatrix}$$

To simplify the application, we freeze A and B at the reference state $\rho = 1, u = M_\infty, v = 0$ so that they become matrices with constant elements

$$A = \begin{pmatrix} 0 & 1 & 0 \\ -M_\infty^2 + 1 & 2M_\infty & 0 \\ 0 & 0 & M_\infty \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & M_\infty \\ 1 & 0 & 0 \end{pmatrix}$$

Now the elements of our vector Q represent small perturbations from a uniform flow in the x direction. The Mach number is M_∞ and equations can be used to study subsonic to supersonic small disturbance flow over slender bodies or past surfaces with small surface variations. The matrix A has real distinct eigenvalues $M_\infty, M_\infty + 1, M_\infty - 1$ and B has the eigenvalues $0, 1, -1$ so the system is hyperbolic in time. It is this form we will use to approximate the flow of air around the foil

2 Computational Methods

2.1 Spatial Difference Approximation: Finite Differences

We first begin by turning the partial differential equations (PDEs) describing our system into a coupled system of ordinary differential equations. This is accomplished by approximating the spatial derivative

$$R(Q) = -A\frac{\partial Q}{\partial x} - B\frac{\partial Q}{\partial y}$$

with a finite difference method. Two methods will be considered here: a flux splitting scheme with 2^{nd} order one-sided differencing (with 1^{st} order one-sided differencing at the boundaries) and a compact differencing scheme.

The first scheme uses flux splitting with backward (δ^b) and forward differencing (δ^f) and is given by

$$R(Q)^n_{j,k} = -A^+\delta^b_x Q^n_{j,k} - A^-\delta^f_x Q^n_{j,k} - B^+\delta^b_y Q^n_{j,k} - B^-\delta^f_y Q^n_{j,k} \tag{1}$$

The exact form of δ^b and δ^f are outlined in the text. 2^{nd} order one-sided differencing is used in the grid interior while 1^{st} order one-sided differencing is used near the boundaries. Despite the use of the first-order differencing, 2^{nd} order global spatial accuracy is maintained. We know that for the x direction, the taylor error, er_t goes like

$$er_t = \frac{\Delta x^2}{3}(\frac{\partial^3 u}{\partial x^3})_j$$

And the same relation holds for the y direction. The modified wave numbers for the first order method are

$$ik_x^{*b,f} = \frac{\pm(1 - \cos(k_x\Delta x)) + i\sin(k_x\Delta x)}{2\Delta x}$$

and those for the second order method are

$$ik_x^{*b,f} = \frac{\pm(3 - 4\cos(k_x\Delta x) + \cos(2k_x\Delta x)) + i(4\sin(k_x\Delta x) - \sin(2k_x\Delta x))}{\Delta x}$$

again, the same relations hold in the y-direction.

The second scheme is a compact scheme given by

$$(\frac{\partial Q}{\partial x})_{j+1} + 4(\frac{\partial Q}{\partial x})_j + (\frac{\partial Q}{\partial x})_{j-1} = -\frac{3}{\Delta x}(-u_{j-1} + u_{j+1}) \tag{2}$$

As indicated in the text, the taylor error for this method goes like

$$er_t = \frac{\Delta x^4}{120}(\frac{\partial^5 u}{\partial x^5})_j$$

making it fourth-order accurate. The modified wave number for this method was calculated as a homework problem, and is given by

$$ik^* = \frac{3i\sin(k\Delta x)}{(2+\cos(k\Delta x)\Delta x}$$

Implementing this method requires inverting a banded tri-diagonal matrix of the form B(1,4,1) in order to gain the increased spatial accuracy.

2.2 Time-Marching Methods

After approximating the spatial derivatives in the problem, we integrate the solution forward in time by turning the coupled set of ODEs into Ordinary Difference Equations (OΔEs). The methods considered here will be the Euler Explicit (EE) method and a Fourth-Order Runge-Kutta (RK4) method. These methods are defined as

$$Q_{j,k}^{n+1} = Q_{j,k}^n + hR(Q)^n \tag{3}$$

and

$$\hat{Q}_{j,k}^{n+1/2} = Q^n + \frac{1}{2}hR(Q)^n \tag{4}$$

$$\tilde{Q}_{j,k}^{n+1/2} = Q^n + \frac{1}{2}R(\hat{Q})^{n+1/2}$$

$$\overline{Q}_{j,k}^{n+1} = Q^n + hR(\tilde{Q})^{n+1/2}$$

$$Q_{j,k}^{n+1} = Q_{j,k}^n + \frac{1}{6}h[R(Q)^n + 2(R(\hat{Q})^{n+1/2} + R(\tilde{Q})^{n+1/2}) + R(\overline{Q})^{n+1}]$$

where h=Δt is the timestep used.

As discussed in class and in the text, time-marching methods approximate the value of $e^{\lambda h}$ with σ-roots. The relationship between the σ-roots and λ for the the EE and RK4 methods are

$$\sigma = 1 + \lambda h$$

and

$$\sigma = 1 + \lambda h + \frac{1}{2}\lambda^2 h^2 + \frac{1}{6}\lambda^3 h^3 + \frac{1}{24}\lambda^4 h^4$$

3 The Physical Set-Up

The spatial and temporal methods described in the previous sections can now be used to model small disturbance flows in a physical space. We will use a uniform grid in a two-dimensional range specified by -1 ≤ x ≤ 3 and 0 ≤ y ≤ 2. The surface disturbance will be a thin airfoil with a surface described by

$$y_{wall} = \tau x(1-x)/2 \quad 0 \le x \le 1$$

$$y_{wall} = 0 \qquad\qquad x < 0, x > 1$$

And the boundary conditions that will be used are the following

1. At inflow (x=-1): ρu=M_∞, ρv=0, and $\frac{\partial \rho}{\partial x} = 0$
2. At the top (y=2): ρ=1, u=M_∞, v=0
3. At outflow (x=3): $\frac{\partial Q}{\partial x} = 0$
4. At the bottom (y=0): v is specified with thin airfoil conditions using

$$v = M_\infty \frac{\partial y_{wall}}{\partial dx}$$

This set-up thus assumes we have some constant source of airflow at the inflow and that the upper portion of the domain is off at infinity. To model it numerically, we break the region up into a discrete set of grid points. The surface of the airfoil is one unit long in physical space and we divide it into nx discrete grid points. So this determines the physical spacing between grid points as

$$dx = dy = \frac{1}{nx - 1}$$

The grid will then consist of $int(4/dx + 1) \times int(2/dy + 1)$ points. They are indexed by (column, row) as (j, k).

To integrate the equations, we must start with some initial value for the grid points. The initial condition that will be used is

$$Q^{(0)} = \begin{pmatrix} 1 \\ M_\infty \\ 0 \end{pmatrix}$$

representing uniform density over the whole grid, a uniform x velocity of M_∞ and zero velocity in the y-direction. We should expect that due to the presence of the airfoil, some motion of the air should transfer to the y direction over time.

When we integrate forward in time, we must choose a value for the timestep, $h = \Delta t$. This is accomplished by defining a CFL number, and for this project I have chosen

$$CFL = \frac{h(2 + M_\infty)}{min(dx, dy)}$$

This ratio has units of *time/length* (the mach number is dimensionless) and is degenerate in the sense that the same CFL number can represent different spatial and temporal scales as long as their ratio remains constant.

4 Choosing the right CFL number

The simulations of the physical situation described will be run using combinations of the spatial and time-marching methods described above. For fixed value of nx, dx, dy and M_∞ we expect different behavior if we vary the CFL number (corresponding to a change in the timestep h). For certain values of this parameter, we should expect the solution to approach a steady state where

$$R = \frac{\partial Q}{\partial t} = -A\frac{\partial Q}{\partial x} - B\frac{\partial Q}{\partial x} + bc = 0$$

while for others, we should expect the solution to blow up as time advances. Furthermore, certain values of the CFL should result in a faster convergence of the solution to a steady state than others. This behavior can be understood by doing a periodic analysis of the solutions. To do this, we first consider the equations in the following form

$$\frac{\partial Q}{\partial t} = (-A\frac{\partial}{\partial x} - B\frac{\partial}{\partial y})Q + bc = \overline{A}Q + bc$$

where $\bar{\text{A}}$ is the 'signature matrix' of the problem. It is a difference matrix that characterizes the physics and methods being used in the problem.

If we now consider that the value of $Q_{j,k}$ can be expressed as a sum over the possible wavenumbers of the grid being used, i.e.

$$Q^n_{j,k} = \sum_{k_x=-k_{x_{max}}}^{k_{x_{max}}} \sum_{k_y=-k_{y_{max}}}^{k_{y_{max}}} C_{k_x,k_y}\sigma^n_{k_x,k_y}X_{k_x,k_y} + \overline{A}Q^{-1}bc \tag{5}$$

It is evident that the transient term in (5) will blow up if the value of any of the σ_{k_x,k_y} are greater than 1. In order to check whether or not the σ_{k_x,k_y} are greater than one, we can use the modified wave numbers of the particular spatial method we are using in place of the difference operators as in

$$(-A\frac{\partial}{\partial x} - B\frac{\partial}{\partial y})Q^n_{j,k} = (-Aik^*_x - Bik^*_y)Q^n_{j,k}$$

and obtain a 3×3 matrix for each set of wavenumbers. With these, the $\sigma - \lambda$ relationship can be applied in matrix form. The magnitude of the eigenvalues of the sigma matrix will then determine whether or not the solution will be stable or not.

The file Final_Project_Sigma_Lam.m performs this analysis for a choice of spatial differncing and time-marching methods. It computes the maximum value of σ_{k_x,k_y} for a range of CFL numbers, and plots them. The plots show where the transition between stable/unstable behavior occurs (i.e. where σ_{k_x,k_y} goes above one and indicates the approximate CFL number where this occurs. It is clear from this analysis that the Euler Explicit method will be unstable for all CFL values if combined with the compact scheme or the flux splitting scheme, as show in the figure below.

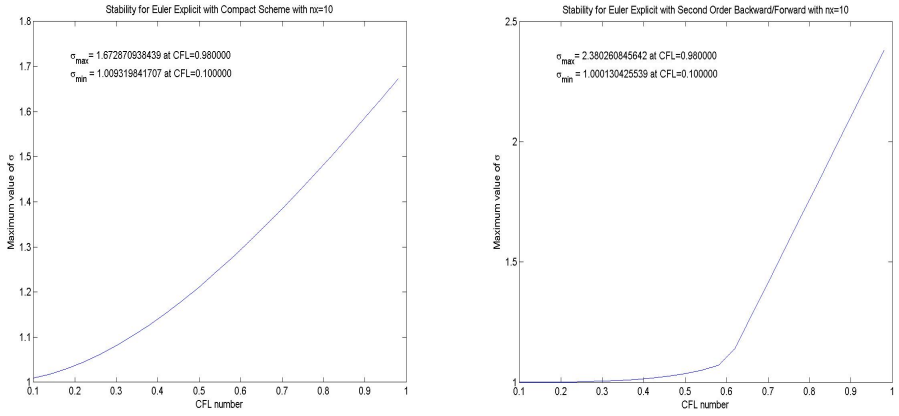

Figure 1: Plots showing that σ_{k_x,k_y} vs. CFL for the EE method when used with the flux-vector splitting or compact spatial methods

On the other hand, for the RK4 method the analysis shows that there are ranges of CFL number for which both the compact and flux vector splitting spatial methods are stable. The transition should occur around CFL=.8 for the flux splitting method and at about CFL=1.9 for the compact scheme. The file Final_Project_Sigma_Lam.m also outputs the value of the CFL number for which the smallest maximum value of σ_{k_x,k_y} occurs. According to the analysis, this CFL should produce the fastest convergence (i.e. the smallest residual at a given timestep as discussed in the text).

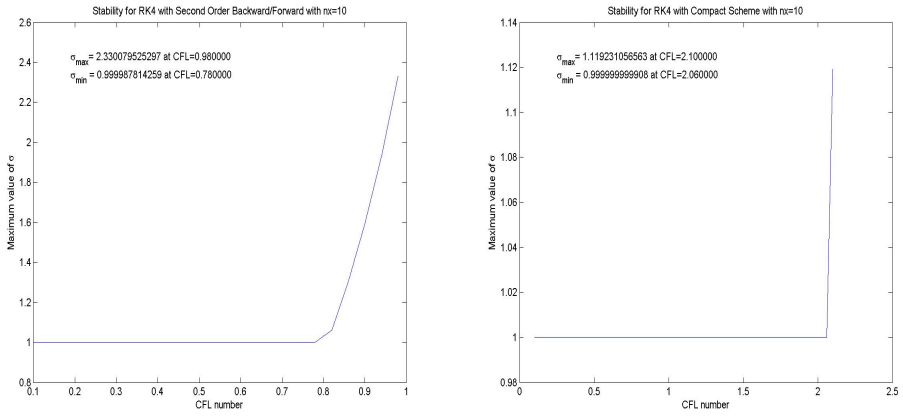

Figure 2: Plots showing σ_{k_x,k_y} vs. CFL number for the RK4 method when used with the flux-vector splitting or compact spatial methods.

5 Results

The simulations were performed in Matlab with a modified version of code provided by Thomas Pulliam. The two main files used were Final_Project_8_flux_so.m and Final_Project_8_compact.m. These pertain to the flux vector splitting and compact spatial difference methods, respectively.

The simulations were first run using the Euler Explicit method on the uniform grid discussed above. As predicted, this method is unstable for all values of CFL number when used with the compact or flux vector splitting spatial methods. For low CFL, the densities and velocities stay within a reasonable range for a while, but eventually blow up with time.

The simulations were first run for a range of CFL numbers with constant values of nx, dx, dy and M_∞ in order to verify the stability predictions made in the last section. The table below shows the results for the RK4 and second-order upwind flux-splitting schemes with values of $nx = 10$ (corresponding to $dx = .1111$), and $M_\infty = 0.8$, after 1000 time steps. The residual at timestep n, r_n, is defined as

$$r_n = \frac{\|R(Q)\|}{N_{elements}(R(Q))}$$

and the slope is simply $r_n - r_{n-1}$. These parameters give us an indication of how close to convergence the solution is and how quickly it is converging or diverging at a given timestep. A small value of r_n means the elements of the difference matrix are small, meaning that the value of the elements of Q will not change significantly at the next timestep.

CFL	ρ_{max}	ρu_{max}	ρv_{max}	Slope	Residual
0.1	1.007	0.802	0.031	-0.000001356120	0.000452865486
0.2	1.006	0.802	0.031	-0.000000547674	0.000378402891
0.3	1.006	0.802	0.031	-0.000001158779	0.000265631962
0.4	1.006	0.802	0.031	-0.000001382082	0.000126390302
0.5	1.006	0.802	0.031	-0.000000156949	0.000036812463
0.6	1.006	0.802	0.031	-0.000000041620	0.000020786115
0.7	1.006	0.802	0.031	+0.000000005969	0.000009955311
0.8	1.006	0.802	0.031	+0.000000009069	0.000006183199
0.9	1.006	0.802	0.031	+0.000000002887	0.000003718333
1.0	939.9	1141	217.3	+9.773625746298	2847.928039149

Table 1: Values of the physical and computational quantities at a timestep of 1000 for the RK4 method used with Second-order flux splitting in space. Simulations were run at $nx = 10$ (corresponding to $dx = .1111$), and $M_\infty = 0.8$

It is evident from this table that somewhere between CFL=0.9 and CFL=1.0, the behavior goes from being stable to unstable. Upon further inspection of the CFL number in this range, we find that the transition occurs close to CFL=0.98. This does not agree with the predicted value, but some discrepancy is to be expected since the analysis performed assumes that we can describe the values at a

given grid point with periodic functions. Also, the definition of the CFL number being used is a conservative one, and this creates some error in our estimate. The transition from stability to instability is very interesting. As mentioned in the text and in class, increasing the timestep is useful for advancing the solution to a steady state, but it also increases the error during the intermediate times. It is apparent that to some degree, the smaller the timestep, the less error there will be at a given timestep. For instance, if we use CFL=.32 and 60 timesteps, we advance to the same physical time as if we use CFL=.96 and 20 timesteps. However, the behavior is radically different. This is illustrated in figure 3.

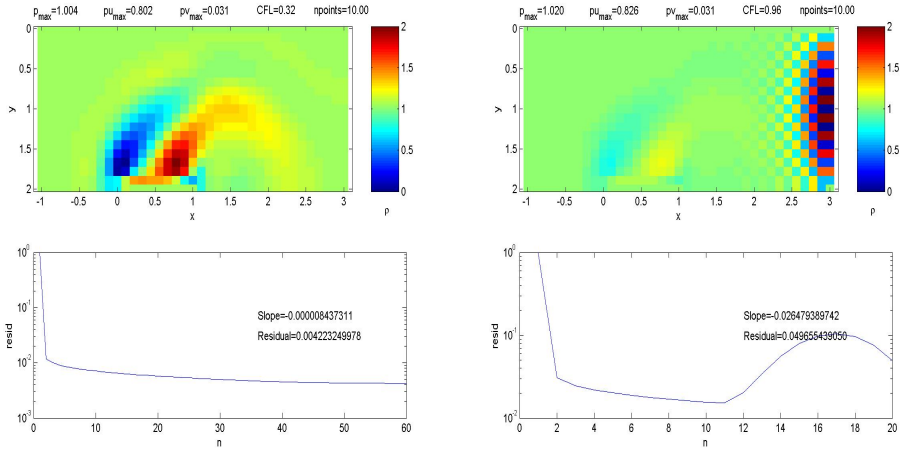

Figure 3: Plots showing results for the same physical time using different CFL numbers. The upper plots are color-density plots and the lower plots indicate the residuals.

We see that for the case of CFL=.32 and 60 timesteps, the density looks like a valid approximation for wind moving over an airfoil. On the other hand, using CFL=.96 produces a non-physical result at 20 timesteps. Both of these choices will eventually converge to a physical steady-state, but the former provides a better simulation of what happens in the meantime than the latter.

Results for the Compact scheme discussed above are full of errors, so they will not be presented here.

6 Concluding Remarks

The simulations used here are a drastic simplification of what would really happen if wind was to flow over an airfoil; they use linearized versions of the Euler Equations and do not conserve physical quantities such as energy. However, they still provide a reasonable model for air moving over an airfoil. Moreover, the analysis performed here provides insight into the subtleties of modeling physics

on a computer. This project showed the ways in which error can be induced in a simulation and the accuracy to which one can work by employing different computational methods. With further education in the subject, I hope to gain a better understanding of how to employ these methods and how to diagnose errors in my simulations. This project seems to be a good start.