Figure A.2: Shear instability at the edge of an altocumulus layer. Photograph from the International Space Station (NASA).

OC680 Homework #3 Due Tuesday Jan. 30

1. **Numerical analysis of shear instability**

    (a) Write a Matlab function to find eigenvalues σ and eigenfunctions $\hat{w}$ of the Rayleigh equation in finite difference form:

    $$\sigma A_{ij}\hat{w}_j = B_{ij}\hat{w}_j \tag{A.0.4}$$

    for $i, j = 1, 2, ..., N$ with boundary conditions

    $$\hat{w}_0 = \hat{w}_{N+1} = 0 \tag{A.0.5}$$

    The matrices $\underset{\sim}{A}$ and $\underset{\sim}{B}$ are defined by

    $$\begin{aligned} A_{ij} &= D_{ij} - \tilde{k}^2 I_{ij} \\ B_{ij} &= -ik(U_i A_{ij} - U_i'' I_{ij}) \end{aligned} \tag{A.0.6}$$

    where $\underset{\sim}{D}$ is the second derivative matrix (including the boundary conditions (A.0.5), $\tilde{k}$ is the wave vector magnitude $\sqrt{k^2 + \ell^2}$, $\vec{U}$ is the background velocity profile, $\vec{U}''$ is its second derivative, $\underset{\sim}{I}$ is the identity matrix and there is no sum on the repeated index i. After defining $\underset{\sim}{A}$ and $\underset{\sim}{B}$, use

    <div align="center">[w,e]=eig(B,A); sigma=diag(e);</div>

    to solve the generalized eigenvalue problem (A.0.4). Finish by sorting the eigenvalue/eigenvector pairs in order of descending growth rate. Output the pair with the largest growth rate. Your function should accept the vectors $\vec{z}$, $\vec{U}$ and the scalars k and ℓ as inputs and deliver σ and $\hat{w}$ for the mode with the maximum growth rate as outputs.
    Hints:

    - In Matlab, a simple way to left-multiply a vector onto a matrix, $v_i A_{ij}$ (with no sum on i), is like this: **diag(v)*A**.
    - The identity matrix of size $N \times N$ is given by the built-in function **eye(N)**
    - It's a good idea to use the 2nd derivative matrix to compute U'', but use one-sided derivatives rather than boundary conditions for the top and bottom rows, since does not obey the same boundary conditions as .
    - Make sure your z vector excludes the boundaries!
    - Sort using **sort(...,'descend')**.

    (b) Write a script to test the function you developed in part (a) for the following test case:

    $$U^* = \tanh(z^*); \ k^* = 0.45; \ \ell^* = 0; \ z^* \in (-4, 4); \ \Delta = 0.2.$$

The script should define the inputs for the function, call the function, then plot the outputs. The resulting plot should show $\hat{w}^*$ versus z^*, both as real and imaginary parts and in polar form (magnitude and phase versus z^*), and should include an annotation that gives the growth rate, e.g.

```
title(sprintf('\\sigma*=\%.3f',your_value_of_sigma)).
```

You should get $\sigma^* = 0.175$.

[Hint: Remember that your vector of z^* values should exclude the boundaries.]

2. **The piecewise-linear shear layer: numerical solution**

Here you will solve the shear layer problem numerically for comparison with the analytical solution.

(a) Repeat the derivation of $\sigma^* = \sigma^*(k*)$ for the piecewise linear shear layer as outlined in class, but with all of the algebra included.

(b) Test your result from (a) using the numerical function developed in project 1. Use the scaled variables, so that the velocity profile is:

$$U* = \begin{cases} 1, & z* > 1 \\ z*, & -1 \le z* \le 1 \\ -1, & z* < -1 \end{cases}$$

Compare plots of $\sigma*(k*)$ as well as eigenfunctions and growth rates of the fastest-growing mode for both the analytical and numerical solutions. Try a few different ranges for z, e.g. z=[-3 3]; z=[-6 6]; z=[-10 10], and plot $\sigma^*(k^*)$ for each. You should find that only when z=[-10 10] or larger is the analytical form of reproduced. That is because the boundary conditions are different. In the analytical solution, we assumed that the vertical domain is infinite, so that $\hat{w} \to e^{-\tilde{k}|z|}$ as $|z| \to \infty$.

[Hint: When comparing eigenfunctions, remember that they are only defined up to a multiplicative constant, which may be complex. As a result, eigenfunctions that should be the same can look totally different. The solution is to normalize. The easiest way is to divide the eigenfunction through by its value at some fixed height, e.g. $z = 0$.]

(c) Resolve the discrepancy in boundary conditions between (a) and (b) by deriving and implementing an asymptotic boundary condition in your code. For consistency with $\hat{w} \to e^{\mp \tilde{k}z}$, require that $\hat{w}' = +\tilde{k}\hat{w}$ and $\hat{w}' = -\tilde{k}\hat{w}$ at $z = z_1$ and $z = z_N$, respectively. At $z = z_1$:

$$\hat{w}'_1 = \frac{\hat{w}_2 - \hat{w}_0}{2\Delta} = \tilde{k}\hat{w}_1 \quad \Rightarrow \quad \hat{w}_0 = \hat{w}_2 - 2\tilde{k}\Delta\hat{w}_1.$$

Now substitute into the finite difference expression for the second derivative:

$$\hat{w}''_1 = \frac{\hat{w}_0 - 2\hat{w}_1 + \hat{w}_2}{\Delta^2} = \frac{2\hat{w}_2 - 2(1 + \tilde{k}\Delta)\hat{w}_1}{\Delta^2}$$

After a similar process at the upper boundary, you should have:

$$D_{1,1} = -2(1 + \tilde{k}\Delta)/\Delta^2 \ ; \ D_{1,2} = 2/\Delta^2 \ ; \ D_{1,j} = 0 \text{ otherwise}$$
$$D_{N,N} = -2(1 + \tilde{k}\Delta)/\Delta^2 \ ; \ D_{N,N-1} = 2/\Delta^2 \ ; \ D_{N,j} = 0 \text{ otherwise.}$$

After making this replacement in the derivative matrix, show that you can match the analytical result with a much smaller domain.