Assignment #2: Effects of the Numerical Approximations

Part I: Linear Stability and Implicit Numerical Damping

Due: 22 October 2019 (By the start of class, preferably via e-mail. *Include your code!*)

Objective

Through practical experiments using a highly simplified numerical model, quantify the effects of finite differencing schemes – particularly their stability, dispersion, and damping characteristics – on forecast quality and accuracy.

A Short Introduction to Fourier Analysis

The forward Fourier transform allows us to convert function values in physical space to frequency or wavenumber space. In discrete form for a real-valued function *f*, the forward Fourier transform can be written as:

$$\hat{f}_k = \sum_{n=0}^{N-1} f_n e^{-\frac{2\pi ikn}{N}}$$

Here, f_n is the value of the function *f* at grid point *n*, *N* is the grid or domain length, *n* is grid point, *k* is a wavenumber, and $\hat{f}_k$ is the value of the transformed function $\hat{f}$ at wavenumber *k*.

Due to Euler's relationship, the exponential in the transform above can be viewed in terms of sine and cosine functions. Thus, the f_n are the amplitudes of those waves in physical space, and the $\hat{f}_k$ document both the amplitude and phase of the summed waves in wave space.

Think about the transformed function: it is the sum of sine and cosine waves, each with a different wavenumber *k*. Because wavenumber is inversely related to wavelength, we can view the function as the sum of sine and cosine waves of different wavelengths. We are often interested in knowing the amplitudes of these waves, which provide a measure of their contribution to the final solution. Both the amplitude and power spectra of the transformed function provide information about wave amplitude, with the former given by $\left|\hat{f}_k\right|$ (the absolute value of the transformed function) and the latter given by $\left|\hat{f}_k\right|^2$ (the square of the amplitude spectra). Both are functions of wavenumber (and thus inverse wavelength).

Let us focus on the power spectrum and consider how to calculate it from model output. If *answer* is an array of *N* grid points separated by a horizontal grid spacing given by *dx*, then the following Python code can be used to obtain the power spectrum (*ps*) and associated frequencies (*freqs*) from the result of the real-valued fast Fourier transform:

```
ps = np.abs(np.fft.rfft(answer))**2
freqs = np.fft.rfftfreq(answer.size, dx)
```

The above assumes that the *numpy* module has been imported as *np*. Note that *answer.size* = *N*.

The frequency helper function (*np.fft.rfftfreq*) creates an array of $N/2 + 1$ (if N is even) or $(N-1)/2 + 1$ (if N is odd) points with equally spaced values between 0 and 0.5. These values are then scaled by the grid spacing (*dx*) to give a frequency array *freqs* with units of m^{-1}. In our one-dimensional example, this can be viewed as wavenumber. The result of *1/freqs* is wavelength with units of m. For our example, the returned wavelengths range between 20000 ($2\Delta x$) and 1000000 ($100\Delta x$), or from the shortest-resolvable wave to the largest-resolvable wave on the grid. The power spectrum itself has units that are the square of those of *answer*.

A similar approach can be used in any programming language. MATLAB works similarly to Python, whereas FORTRAN requires external functions for the fast Fourier transform. (Here, we ignore any data scaling that the fast Fourier transform algorithm may perform. For Python and MATLAB, no scaling is done on the forward Fourier transform, such that scaling of $1/N$ is needed on the inverse Fourier transform to return the original function.)

Introduction

For this assignment and the next, you will conduct several experiments with a numerical model based upon a one-dimensional linear advection equation, i.e.,

$$\frac{\partial h}{\partial t} = -U\frac{\partial h}{\partial x}$$

In the real world, any field h will simply be advected, without change in amplitude or wavelength, by the constant advecting velocity U. As we have discussed in class, however, this is not true for modeled waves, with solution properties reliant upon the chosen finite differencing scheme.

Your model may be programmed and its output visualized in any programming languages of your choice. If you have rudimentary experience with or a willingness to learn MATLAB or Python, you may find either to be ideal for this assignment. As with all assignments in this class, while you are free to talk about this assignment with your classmates, all work that you complete and turn in must be your own.

Model Configuration

Some elements of your model will not change between individual experiments. These include:

- **Grid:** 100 grid points, with $\Delta x = 10$ km and periodic lateral boundary conditions (i.e., what goes out of the domain at one end enters at the other end).

- **Advective velocity:** 10 m s^{-1}.
- **Duration of integration:** 100,000 s (the time that it takes an initial feature to be advected over the entire domain back to its starting location, such that the physical wave at the final time is identical to that at the start time).
- **Initial wave center point:** grid point 50.

Other elements of your model may change between individual experiments.

Experiment: Advection of a 1-D Gaussian Wave

A 1-D Gaussian wave may be defined generically as:

$$h(x) = a \exp\left(\frac{-(x-b)^2}{2c^2}\right)$$

For this wave, *a* defines the initial amplitude of the wave, *b* defines the grid point where the initial wave is centered, and *c* is related to the initial width of the wave. For the following questions, a = 100, c = 4, and *b* is as defined earlier in the assignment. Unless otherwise stated, C = 0.8, with the time step Δt and number of time steps left for you to determine from *C*, the advective velocity, and the duration of integration stated above.

1) (30 pts) Using a *forward-in-time*, *backward-in-space* finite difference scheme, integrate your model forward for 100,000 s.

    a. (8 pts) Create a plot of the model solution at five times: the initial time, the second time, the 40^{th} time, the 80^{th} time, and the final time. Use different line colors and/or line styles for each time and ensure that each time is labeled appropriately (such as in a legend). Describe how the model solution evolves with time.

    b. (10 pts) Compute and plot the power spectrum as a function of wavelength at four times: the initial time, the 40^{th} time, the 80^{th} time, and the final time. Describe how the power spectrum changes with time, particularly as a function of wavelength.

    c. (12 pts) Refer to Fig. 1 in the Linear Numerical Stability lecture notes. Considering the information in these figures in light of your answers to (a) and (b), describe how implicit damping is manifest in the model solution for this differencing scheme.

2) (35 pts) Using a *forward-in-time, second-order centered-in-space* finite difference scheme, integrate your model forward for 100,000 s.

    a. (9 pts) Create a plot with the model solution at five times: the initial time, the second time, the 20^{th} time, the 25^{th} time, and the 35^{th} time. Describe how the model solution evolves with time.

b. (13 pts) The stability criterion for this finite difference scheme is given by the equation at the bottom of p. 65 of the textbook. Determine the resolved wavelength at which the exponential is maximized. What is the significance of this wavelength? Do you see evidence of this significance in your plot? Describe.

c. (13 pts) Compute and plot the power spectrum as a function of wavelength at four times: the initial time, the 20th time, the 25th time, and the 35th time. Describe how the power spectrum changes with time. Does this plot support your answer to 2(b)? Reinterpret the model solution in 2(a) in light of your answers to 2(b) and 2(c).

3) (35 pts) Using a *second-order centered-in-time, second-order centered-in-space* finite difference scheme with a Courant number of 1.1, integrate your model forward for 100,000 s, making sure to change the time step and number of time steps accordingly.

a. (7 pts) Create a plot with the model solution at five times: the initial time, the second time, the 34th time, the 38th time, and the 40th time. Describe how the model solution evolves with time.

b. (9 pts) The stability criterion for this finite difference scheme is given by equation (3.54) of the textbook. For a Courant number of 1.1, between what wavelengths is this finite difference scheme numerically unstable? (Hint: work backward around the unit circle to find the second wavelength once you've found the first.)

c. (9 pts) How does the solution differ from that in Question 2? Given your answer to 3(b), discuss why these differences may arise.

d. (10 pts) Compute and plot the power spectrum as a function of wavelength at three times: the initial time, the 34th time, the 38th time, and the 40th time. Describe how the power spectrum changes with time. Does this agree with your answer to 3(b)? Discuss.