**MGS 723 – Problem Set #4 – Solving the advection equation with finite differences**

*Assigned: 10/20/21*

*Due: 11/03/21*

**Background:**

We will solve the 1-D advection (/transport) equation using finite differences. By comparing numerical and analytical solutions, we will examine the accuracy and stability of various discretizations. The 1-D advection equation is the following PDE:

(1)

Note that diffusion (i.e., a 2nd spatial derivative) is absent in the equation. We will solve this subject to the following initial conditions (a gaussian pulse located at ):

(2)

We will consider the following three finite difference discretizations:

FTCS Method:

(3)

LAX method:

(4)

Staggered leapfrog:

(5)

Note: For the staggered leapfrog discretization, you will need to store two previous time-steps ( and ).

**Questions:**

1. **FTCS method:**
2. Download the skeleton Python script “1D\_Advection\_Skeleton.py”. Using the above discretization for the explicit solution, fill in the blanks (‘???’) to complete the code using an FTCS discretization (Equation 3). Does the numerical solution accurately track the analytical one? *(Note: Don’t worry about what happens as the pulse hits the boundary as this is a boundary condition issue unrelated to the FD scheme)*
3. Modify the plotting portion of the script so that you can plot three different times during a single model run (either on three panels or on one plot).
4. Play around with the time step and grid spacing values and, for each, compute the following non-dimensional parameter (the *Courant Number*). Does the model become more unstable (or “blow-up” more) as increases/decreases? Can you find an value that avoids this blow-up?

*As you can hopefully see from the above, the FTCS scheme does not work particularly well due to stability issues. Now let’s program a finite difference method that is more stable than FTCS:*

1. **LAX method:**
2. Program the LAX method (Equation 4) and revert and values back to those of the original script. Describe how the model now evolves. Does it blow-up (i.e., are there oscillations)? How about numerical diffusion?
3. Again, play around with the time step and grid spacing values and, for each, compute (the *Courant Number*). Is the numerical scheme stable for all Courant numbers? How does the amount of numerical diffusion vary with ?
4. What is the physical meaning of ? What happens for ?

*This should illustrate a trade-off between numerical stability (e.g., blow-ups) and numerical accuracy (e.g., numerical diffusion). Now let’s try a scheme that should be more accurate:*

1. **Staggered leapfrog method:**
2. Program the Staggered leapfrog method (Equation 5) assuming that at the first time-step.
3. Again, try different values of (the *Courant Number*) and compare the accuracy and stability of the solution with that exhibited by the previous methods. For what values is the scheme accurate and stable?
4. Make the width of the initial temperature perturbation smaller. Does this affect the stability of this scheme? Why might this be?
5. Considering the derivatives in Equations 5, what is the accuracy of the scheme? (i.e., order of accuracy in and ). How about the FTCS scheme?

Note: These problems are modified Becker and Kaus, Numerical Modeling of Earth Systems