Assignment: hydrodynamics

Colin P. Johnstone

1 Intro

The first two parts of this assignment are about the advection of a function over a grid using different numerical advection algorithms. In the third part of the assignment, you will construct a simple 1D hydrodynamic code and perform a Sod shock tube test.

In all of Part 1 and Part 2 of this assignment, you will have to solve the same advection problem. The function that you are to advect is $q(x,t)$ which initially has the form of

$$q(x,0) = \begin{cases} 1.0, & \text{if } 2 \leq x \leq 4, \\ 0.1, & \text{otherwise} \end{cases} \tag{1}$$

Assume an advection speed of 0.5 that is constant in time and uniform in space (i.e. $u = 0.5$).

You should advect this quantity over a 1D grid of 100 cells, going from $x = 0$ and $x = 10$. In addition to the 100 cells, you should have two ghost cells on each side of the domain where the quantities are reset every time step to make the spatial gradient of q zero (e.g. for the grid cells on the right side of the domain, you should have $q^n_{102} = q^n_{101} = q^n_{100}$).

Given the above parameters, what is the maximum time step, Δt, that you can use for this problem? (1 mark)

2 Part 1 (7 marks)

In Part 1, you should write a code that advects the function over the grid using the simple techniques of breaking the domain down into finite points based on three different methods for spatial discreetisation and using the Forward Euler Method for time discreetisation. All three methods used should be implemented in one code and you should be able to easily change between the methods by changing one line of code at the top of the file (e.g. maybe you have a variable called 'SPATIAL_METHOD', and when it is set to a certain value, the code uses the corresponding method).

1. Firstly, write the code such that it uses the central difference scheme. *Show and describe what happens when you try to advect the function using this scheme.*

2. Extent the code to use the upwind scheme. *Show and describe what happens when you try to advect the function using this scheme. Show and describe what happens when you make the time step, Δt, larger than the maximum allowed value?*

3. By adding artificial diffusion to the advection algorithm, extend the code to follow the Lax-Wendroff method. *Show and describe what happens when you try to advect the function using this scheme.*

3 Part 2 (7 marks)

In Part 2, you will write a code to advect the same function across the same grid. This time, the advection should use a flux conserving scheme, such that you break the spatial domain down into cells instead of discreet points and calculate the update in q_i^n by calculating the fluxes at the cell boundaries. Once again, you will have to use several different methods and you should write a single code that can be used to run all of them. The methods that you should use are the donor-cell method, Fromm's method, the Beam-Warming method, and the Lax-Wendroff method.

1. Write a code that can solve the advection problem using the four different methods. *Show and describe what happens when each of these methods are used.*

2. Improve the code such that it can handle $u = -0.5$ instead. Hint: basically you are using the equations for the four methods derived in the lecture, but when calculating the flux at the cell interface, you need to make the code decide whether to use the q value on the left or right side depending on which direction the information is travelling.