Homework 5

Computational Astrophysics (ASTR660)

(Due at the start of class on December 8, 2022)

Exercise 1

[Collisionality of astrophysical systems (1 pt)]

The fluid approximation is valid for describing systems when the particles within the medium are *collisional*, i.e., when $\lambda \ll L$, where λ is the mean free path between particle collisions, and L is the size of the system. For ionized gas in the universe, the mean free path can be expressed as

$$\lambda = \frac{1}{n\sigma} = \frac{(k_{\mathrm{B}}T)^2}{ne^4} \sim 10^6 \left(\frac{T^2}{n}\right) \mathrm{cm}, \tag{1.1}$$

where T and n are the temperature (in Kelvin) and number density (in cm^{-3}) of the gas, respectively. Please determine for each of the following astrophysical system whether it can be described as a fluid.

(1) The Sun: $T \sim 10^7$ K, $n \sim 10^{26}$ cm^{-3}.

(2) Solar wind: $T \sim 10^5$ K, $n \sim 10$ cm^{-3}.

(3) Warm ionized interstellar medium: $T \sim 10^4$ K, $n \sim 10$ cm^{-3}.

(4) Intracluster medium within galaxy clusters: $T \sim 3 \times 10^7$ K, $n \sim 10^{-3}$ cm^{-3}.

Exercise 2

[Error analyses for finite-volume methods (1.5 pt)]

During the in-class exercise in Lecture 11, we implemented the Lax-Friedrichs method for solving the advection equation. Please complete the exercise if you have not done so. In this exercise, we will compare the truncation errors of the Lax-Friedrichs (LF) method and the Lax-Wendroff (LW) method for the advection test.

(1) Please implement the LW method. Make a plot to show the results for the advection of a top-hat function.

(2) Since both the LF and LW methods cannot handle discontinuities very well, let's change the initial condition to simulate a wave instead. The wave can be described as a Gaussian function centered at $x = 0.1$ at $t = 0$:

$$u(x) = \max(\exp(-1000(x-0.1)^2), \varepsilon), \tag{2.1}$$

where $\varepsilon = 10^{-99}$ is used to prevent file I/O errors when the numbers are too small. Please modify `initial.f90` as well as `IO.f90` (the analytical solution) accordingly. Run the advection test of this wave and compare the results using the LF and LW methods.

(3) Vary the number of grid cells from $N = 50$ to $N = 5000$ for both methods. Defining the error of the solutions to be $\mathcal{E} = \Sigma_i(|u_i - u_{i,anal}|)/N$, make a log-log plot for $\mathcal{E}$ versus N for both methods. Do the scalings you obtained agree with the expected truncation error of the two methods? Recall that the truncation errors for the LF and LW methods are of orders $\mathcal{O}(\Delta x^2, \Delta t)$ and $\mathcal{O}(\Delta x^2, \Delta t^2)$, respectively.

Exercise 3

[1D and 2D advection simulations using shock-capturing methods (3 pt)]

(1) In Lecture 12, we implemented the piecewise-linear method (PLM) for the advection test, which handles discontinuities better than the LF and LW methods. Please complete the exercise if you have not done so. Please show the results for advecting a top-hat function using $N = 500$, where N is the number of grid cells in the x-dimension.

(2) Please generalized your code from 1D to 2D in order to advect a square function in the diagonal direction to the upper-right corner of the 2D domain (as shown in Figure 1). Again, we will assume periodic boundary conditions for all boundaries. Please use the following parameters for this simulation: $N_x = N_y = 128$ (number of grid cells in x and y), $c_x = c_y = 1.0$ (initial x and y velocities of the square function), $t_{\rm end} = 2.5$ (final time of the simulation), and $cfl = 0.4$. At $t = 0$, please set up the initial condition as the following:

$$u(x, y) = \begin{cases} 1.0 & \text{if } 0.1 \le x \le 0.2 \text{ and } 0.1 \le y \le 0.2 \\ 0.01 & \text{otherwise.} \end{cases} \tag{3.1}$$

Note that for multi-dimensional simulations, one can use operator splitting, so we could update the solutions on the grid using fluxes in the x-direction first, and then update the solutions on the grid using the y fluxes. A pseudo code for the `update` subroutine in `evolution.f90` would look like the following:

```
subroutine update(time, dt)
    use Simulation_data
    (variable declarations)

    ! do x-direction first

    ! update B.C. in x
    ! copy the current solution to the uold array
    ! for each 2D grid cell
    !    compute x flux
    !    update solution using x flux

    ! repeat the above for y-direction

end subroutine update
```

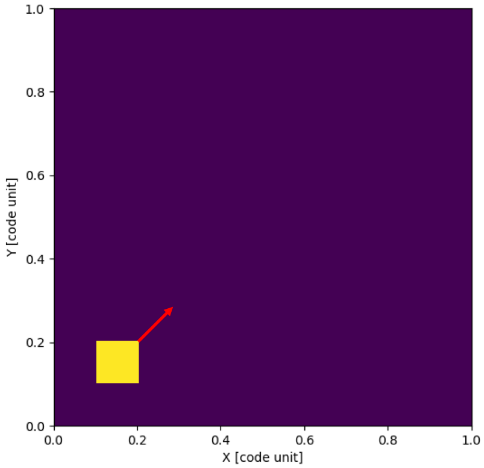

Figure 1: Initial condition of a square function moving to the upper-right corner for the 2D advection test.

Modify `IO.f90` so that the 2D data on the grid could be written to output files in a format that can be read by your favorite plotting routine. Plot the simulation result at t_{end}.

[Please use the PLM method you developed in (1) to compute the fluxes for this exercise. If your PLM implementation was not successful, you could use the LF/LW/upwind methods instead to get partial credits.]

Exercise 4

[Sod shocktube test using Ulula (1.5 pt)]

(1) Please run the Sod shocktube test using the `Ulula` code (`https://bdiemer.bitbucket.io/ulula/index.html`) and reproduce the gas profiles as shown in Figure 2.

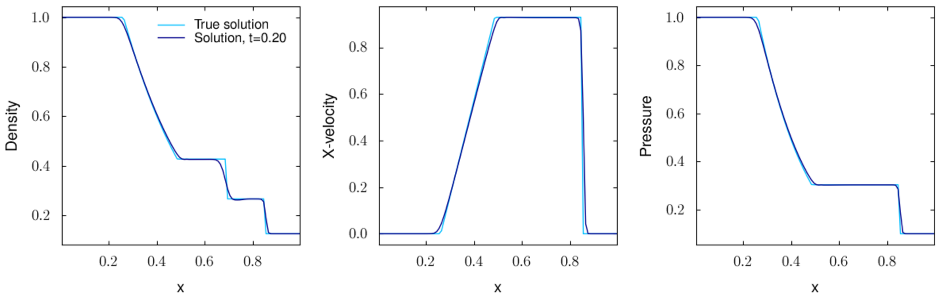

Figure 2: Profiles of gas density, x-velocity, and pressure at $t = 0.2$ for the Sod shocktube test.

(2) Please experiment with different combinations of options for the hydro schemes (e.g., reconstruction methods, slope limiters, time-stepping algorithms). Choose three cases that you tried, plot the profiles, and briefly describe their differences.