Computational Astrophysics, Fluid Dynamics

21777

Question 1.

The initial solver I created used the Lax-Friedrichs method, this enabled me to create the skeleton of a Riemann solver program which I could then improve on. The first improvement I implemented was the Courant-Friedrichs-Lewy (CFL) stability condition imposed onto the time step. This allowed me to increase N, the number of grid spaces, which showed that for higher values of N my Lax-Friedrichs method approached the correct solutions. The next improvement made involved altering the flux reconstruction algorithm from the Lax-Friedrichs to an HLL method. This involved estimating the fastest signal velocities emerging from the initial discontinuity at each interface. The final improvement I made was changing from a basic first order time step to a second order Runge-Kutta scheme. The results are shown in figures 1 and 2 where the grey line shows the exact solution, the blue dots are the points generated by my HLL function using N=100 and the orange dots are the points generated by the initial Lax-Friedrichs method using N=100, this highlights the significant difference that the improvements made.

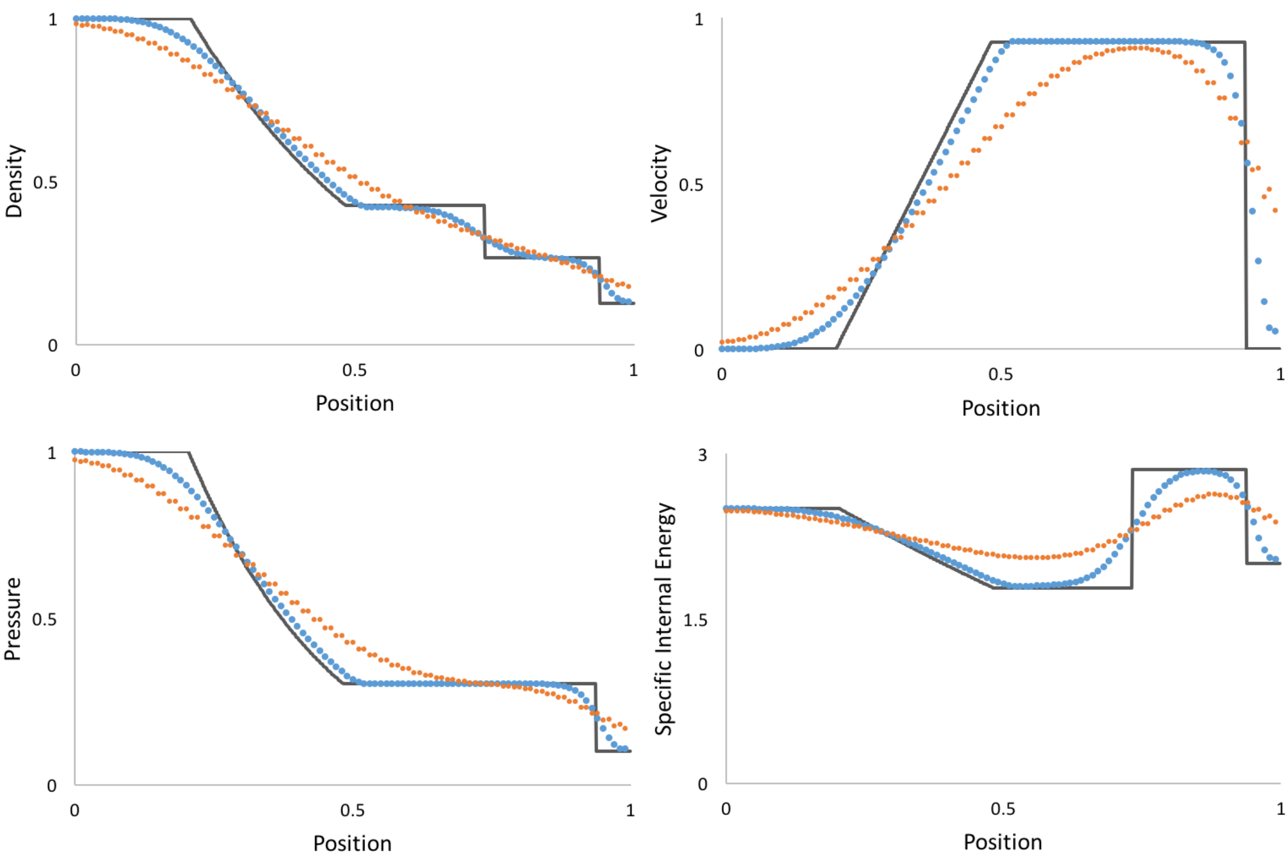

Figure 1. the exact solutions in grey, the solution generated by an HLL method with a CFL stability condition and second order time step with N=100 grid spaces in blue, and the solution generated by the original Lax-Friedrichs method with a CFL stability condition with N=100 grid spaces in orange for density, velocity, pressure and specific internal energy at time t=0.25 units, for setup A.

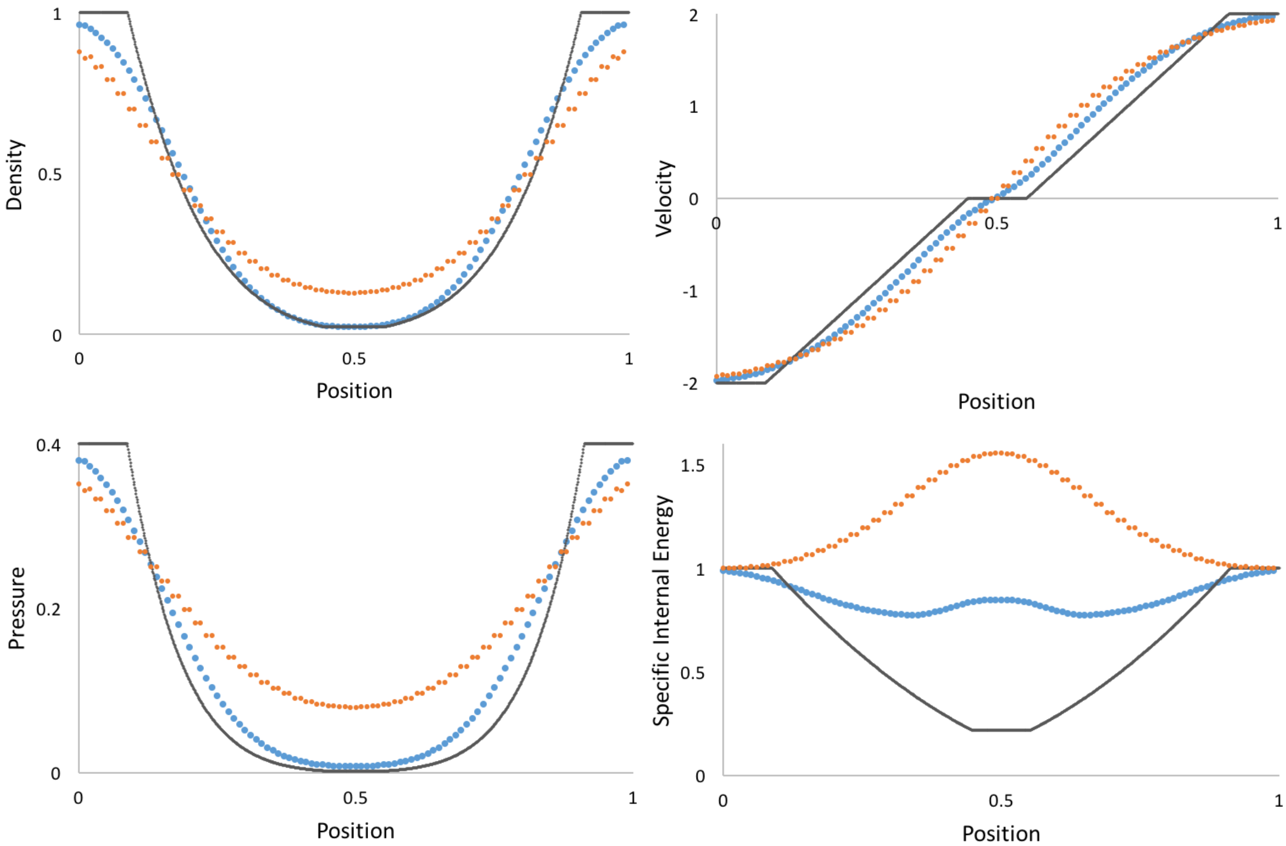

Figure 2. the exact solutions in grey, the solution generated by an HLL method with a CFL stability condition and second order time step with N=100 grid spaces in blue, and the solution generated by the original Lax-Friedrichs method with a CFL stability condition with N=100 grid spaces in orange for density, velocity, pressure and specific internal energy at time t=0.15 units, for setup B.

Question 2.

To create the 2D solver I created an N by N grid of points, to move forward a step in time I performed one sweep through the grid, using my HLL solver, considering the fluxes in the x-direction to produce a new set of grid points. Using this new set of points, I then applied the HLL solver again to performed a second sweep considering the fluxes in the y-direction. This produced a new set of grid points equivalent to having taken one step in time where the change in time still obeys the CFL stability condition. In 2-dimensions there is conservation of momentum in both the x and y-directions and so the state vector and fluxes were updated to

$$\boldsymbol{q} = \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ E \end{pmatrix}, \boldsymbol{F_x} = \begin{pmatrix} \rho v_x \\ \rho {v_x}^2 + p \\ \rho v_x v_y \\ (E+p)v_x \end{pmatrix}, \boldsymbol{F_y} = \begin{pmatrix} \rho v_y \\ \rho v_x v_y \\ \rho {v_y}^2 + p \\ (E+p)v_y \end{pmatrix},$$

where $\boldsymbol{q}$ is the state vector, ρ is the density, v_x is the velocity in the x-direction, v_y is the velocity in the y-direction, E is the total energy, p is the pressure, $\boldsymbol{F_x}$ is the flux in the x-direction and $\boldsymbol{F_y}$ is the flux in the y-direction. The arguments of the 2D solver function accommodate the top and bottom nature of the initial conditions used in our test problem. The function could easily be altered to take into consideration left and right initial conditions or initial conditions split into four quadrants. Figure 3 shows the solution to our test problem generated for N=300 grid points.

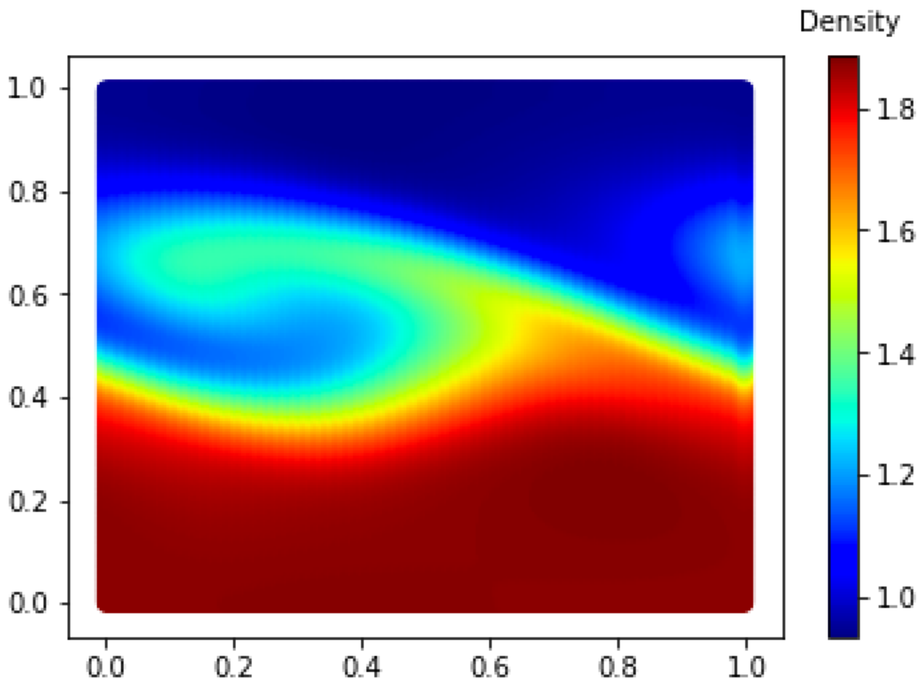

Figure 3. the solution generated by an HLL method with a CFL stability condition with N=300 grid spaces for density at time t=3.0 units showing the Kelvin-Helmholtz instability.