

I worked with John Pharo on this worksheet. It took about four hours.

A Simple 1D Finite Volume Hydro Code: The Shock Tube Problem

1.

I first implemented `con2prim`, `prim2con`, the piecewise constant reconstruction, and the HLLE Riemann solver, all in `ws16.py`. A description of the structure of that code follows.

First, a one-dimensional grid of some number of cells, each of which representing the same width of physical space, is set up. In each cell, the density, velocity, internal energy, and pressure of the fluid is stored at the center, left boundary, and right boundary. Additionally, an array of conserved quantities (density, momentum density, and energy density) is stored at each cell, these being functions of the density, velocity, and internal energy. An initial time step is calculated based on the calculated speed of sound in the fluid to ensure that information cannot travel further than a fraction of one grid-spacing per time step. Then in the main loop, the time step is recalculated, then the conserved quantities are updated via the Euler FD equations and the RK3 method. In each intermediate step of the RK3 method, the conservative quantities are transformed back to primitive quantities and then the boundary conditions are enforced on them. Then the loop repeats. Some representative figures of the output at different times for piecewise-constant reconstruction follow.

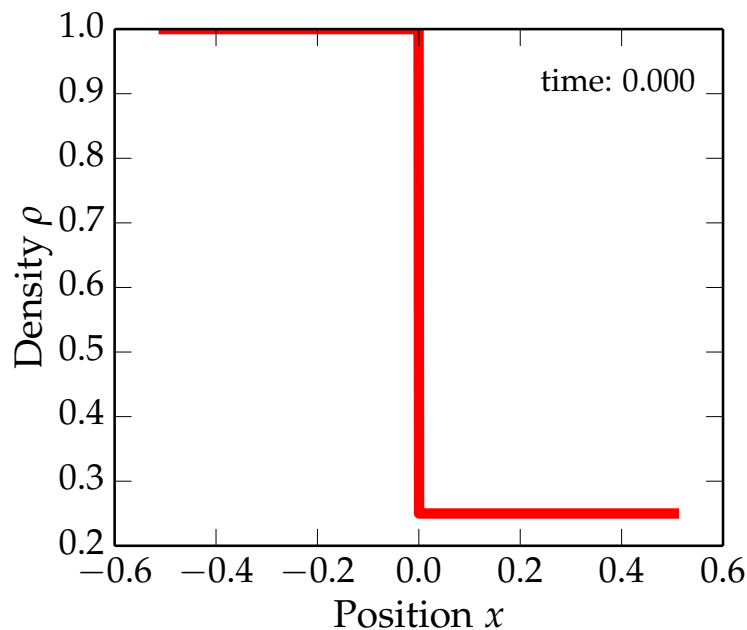


Figure 1: Initial density profile.

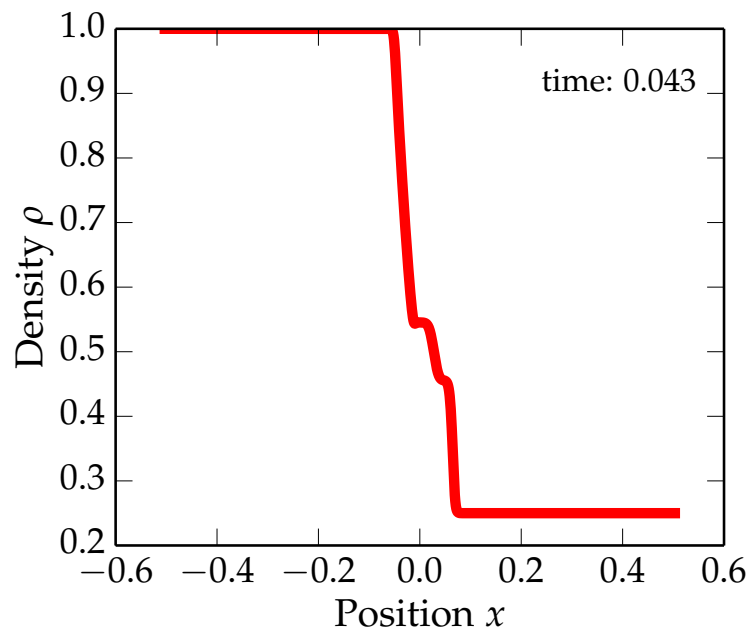


Figure 2: Density profile at the 200th simulation iteration, which corresponds to a time of 0.043.

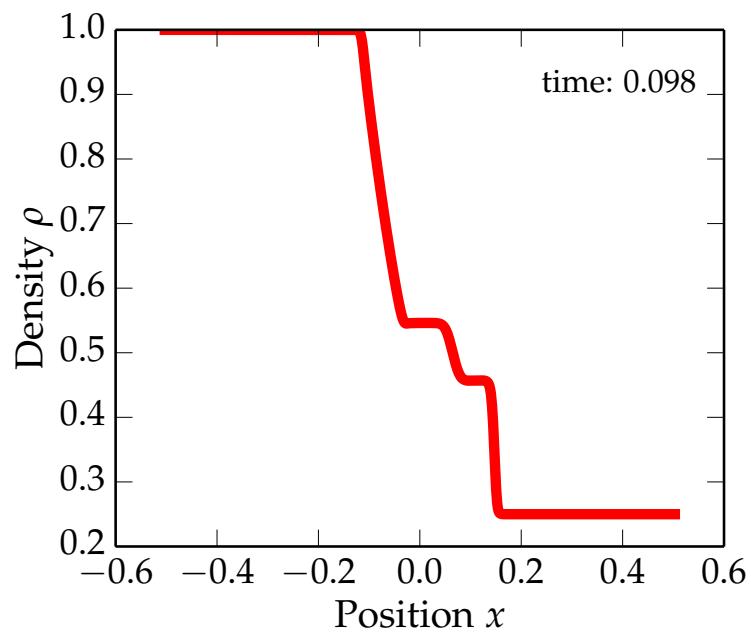


Figure 3: Density profile at the 400th simulation iteration, which corresponds to a time of 0.098.

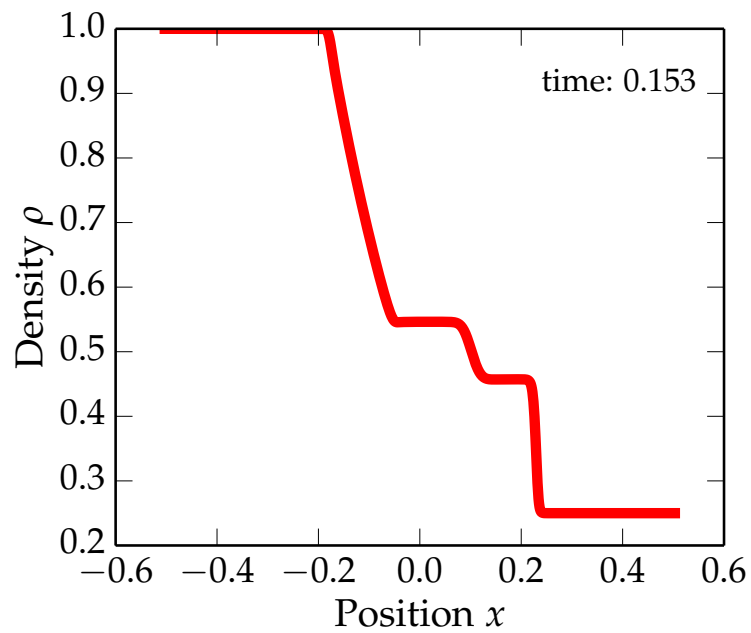


Figure 4: Density profile at the 600th simulation iteration, which corresponds to a time of 0.153.

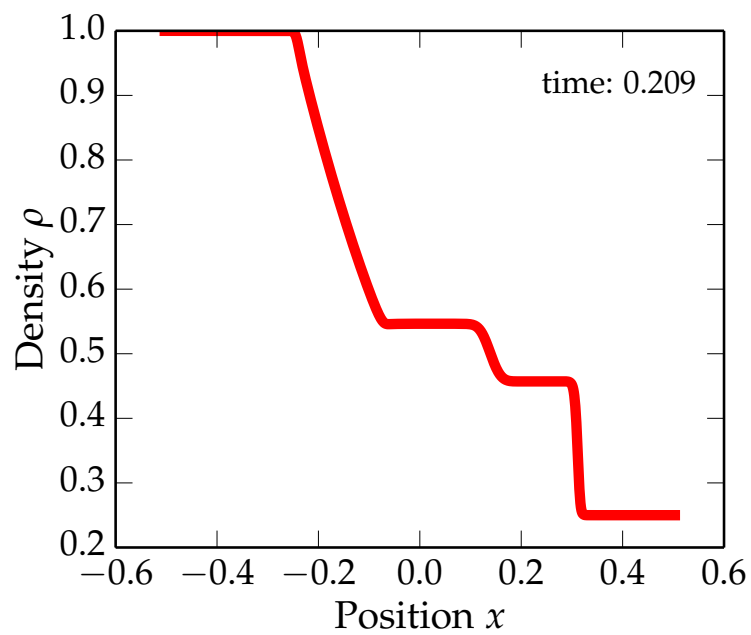


Figure 5: Density profile at the 800th simulation iteration, which corresponds to a time of 0.209.

2.

I then compared the simulation results at time $t = 0.2$ for the piecewise-constant, TVD-minmod, and TVD-MC2 reconstruction methods, as shown in Figure 6.

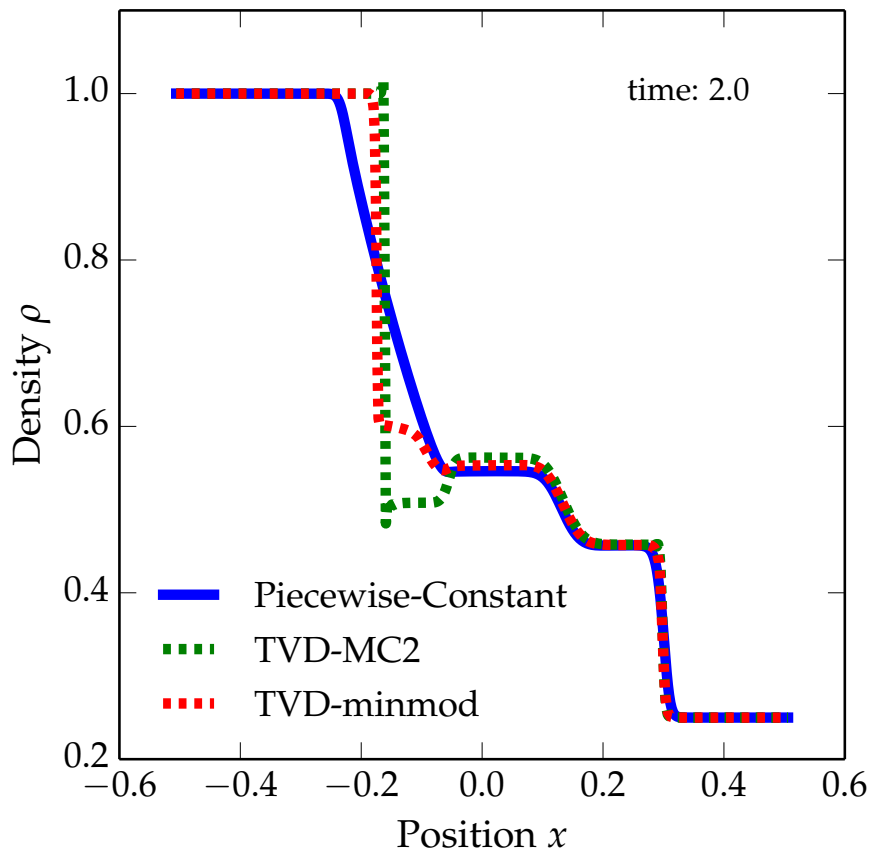


Figure 6: Density profile at time $t = 0.2$ for simulations run with the piecewise-constant, TVD-MC2, and TVD-minmod reconstruction methods. It is clear that the two TVD methods produce more features in the density profile, such as the rarefaction layer.

3.

I then compared the results of this 1D finite volume hydro code with the results of the 1D smooth particle hydrodynamics code from WS15 set up with the same initial conditions. The reconstruction method for the finite volume hydro code that produces a density profile closest to the analytic result is TVD-minmod, as can be seen from Figure 6. I compare the TVD-minmod result (Figure 7) to the SPH result (Figure 8) at $t = 0.2$ below. These produce extremely disparate results; while the finite volume hydro code decreases in density monotonically moving from left to right, the SPH code results in a large local maximum in density at the location where the shock discontinuity is in the finite volume code.

I couldn't get my Fortran compiler to work, so I'm skipping that part.

4.

I sped up my code by using `numpy` array operations instead of slow loops; the result is in `ws16_sped-up.py`. The run-time of the old code is 216.5 ± 4.0 seconds, whereas the run-time of the new code is 163.0 ± 1.5 seconds, a speed-up of 25%.

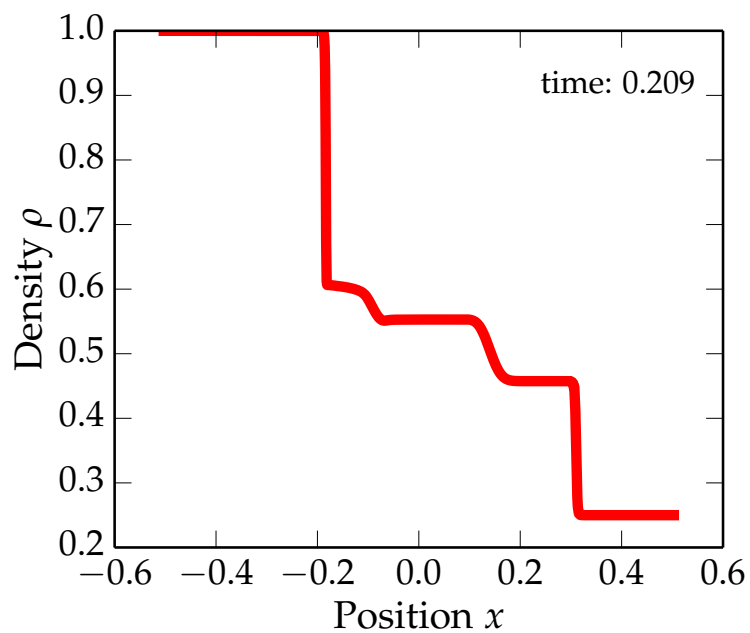


Figure 7: Density profile at time $t = 0.2$ for the finite volume hydro code with TVD-minmod reconstruction.

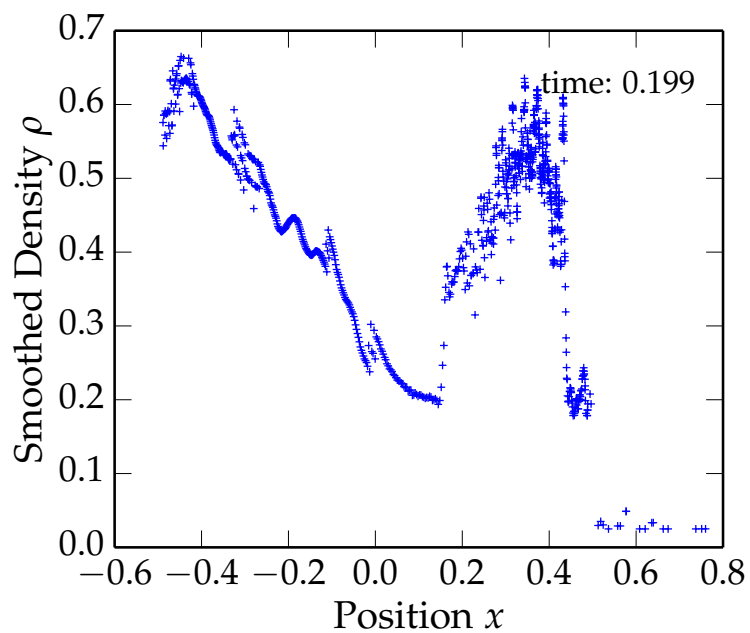


Figure 8: Density profile at time $t = 0.2$ for the SPH code.