How To Write A Hydrodynamics Code

1. Introduction

In this note, I describe how to write a Eulerian hydrodynamics code. In § 2, I show how to write a one-dimensional first-order hydrodynamics code based on an approximate Riemann solver. In § 3, I show how to extend the 1D first-order code to high order. In § 4, I describe how to write a two-dimensional high-order hydrodynamics code.

2. One-Dimensional First-Order Method

The equations of one-dimensional hydrodynamics can be written as,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = 0, \tag{1}$$

where, $U = (\rho, \rho v, E)^T$ is the conserved variable, and $\mathbf{F} = (\rho v, \rho v^2 + P, (E + P)v)^T$ is the flux, ρ is the fluid density, v is the velocity, P is the pressure, and $E = \rho e + \frac{1}{2}\rho v^2$ is the total energy density, here e is the specific internal energy. The equations are closed by an equation of state (EOS) given by $p = p(\rho, e)$. For an ideal gas, the EOS reads,

$$p = (\gamma - 1)\rho e, \tag{2}$$

where γ is the adiabatic index of the ideal gas.

To numerically solve the above equation, the time dependent evolution of Eq. 1 can be expressed in the semi-discrete form

$$\frac{d\mathbf{U}_i}{dt} = L(\mathbf{U}) = -\frac{\mathbf{F}_{i+1/2} - \mathbf{F}_{i-1/2}}{\Delta x}, \tag{3}$$

where i denotes the cell with its center at x_i, and $\mathbf{F}_{i\pm1/2}$ are the fluxes at the cell interface.

The time integration can be done using the first-order forward Euler method,

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t L(\mathbf{U}^n), \tag{4}$$

where $\mathbf{U}^n$ is the conserved variable at $t = t^n$ and $\mathbf{U}^{n+1}$ is the conserved variable after advancing one time step.

To obtain $\mathbf{F}_{i-1/2}$ and $\mathbf{F}_{i+1/2}$, the fluxes at the cell interface, we can solve the so-called Riemann problem. Given the variables at cell i and $i + 1$, we can calculate the flux at the

interface, $x = x_{i+1/2}$. The exact solution of this problem can be solved numerically. But it is very expensive because iterations are involved. Fortunately, we can use an approximate Riemann solver. There is an industry of designing approximate Riemann solvers. The HLL Riemann solver is an example of efficient approximate Riemann solvers. Given the left state $\mathbf{U}^{\rm L}$ and right state $\mathbf{U}^{\rm R}$, the HLL flux can be written as,

$$\mathbf{F}^{\rm HLL} = \frac{\alpha^+\mathbf{F}^{\rm L} + \alpha^-\mathbf{F}^{\rm R} - \alpha^+\alpha^-(\mathbf{U}^{\rm R} - \mathbf{U}^{\rm L})}{\alpha^+ + \alpha^-}, \tag{5}$$

where α^+ and α^- are related to the minimal and maximum eigenvalues of the Jacobians of the left and right states in the form,

$$\alpha^\pm = {\rm MAX}\{0, \pm\lambda^\pm(\mathbf{U}^{\rm L}), \pm\lambda^\pm(\mathbf{U}^{\rm R})\}. \tag{6}$$

Here, the minimal and maximum eigenvalues $\lambda^\pm$ are given by,

$$\lambda^\pm = v \pm c_s, \tag{7}$$

where $c_s = \sqrt{\gamma P/\rho}$ is the sound speed. The HLL flux formula can be used to calculate the flux at the cell interface. For example, to obtain $\mathbf{F}_{i+1/2}$, substitute "L" and "R" by i and $i+1$ in Eq. 5

The time step Δt needs to satisfy the Courant-Friedrich-Levy condition. Thus the following condition must be satisfied,

$$\Delta t < \Delta x/{\rm MAX}(\alpha^\pm). \tag{8}$$

To test the code, there are a serial of tests one can run. The first test is usually the Sod shock tube problem. In this test, the one-dimensional numerical region ($0 \le x \le 1$) initially consists of two constant states: $p_L = 1.0$, $\rho_L = 1.0$, $v_L = 0.0$ and $p_R = 0.125$, $\rho_R = 0.1$, $v_R = 0.0$, where L stands for the left state, and R the right state. The fluid is assumed to be an ideal gas with an adiabatic index $\gamma = 1.4$. The initial discontinuity is at $x = 0.5$. In this test problem, the evolution of the initial discontinuity gives rise to a shock, a rarefaction wave, and a contact discontinuity in between. This is a fairly easy test. All modern hydrodynamics codes should be able to capture the expected features, acquire correct positions of the shock front, contact discontinuity and rarefaction wave.

You should run the Sod problem and compare the numerical results with the exact solutions. You should also try a harder shock tube problem, with $p_L = 100$, $\rho_L = 10$, $v_L = 0$ and $p_R = 1$, $\rho_R = 1$, $v_R = 0$. To calculate the analytical solution, you can download Frank Timmes' code at `http://cococubed.asu.edu/code_pages/exact_riemann.shtml`. For more tests, see Jim Stone's webpage for examples (`http://www.astro.princeton.edu/~jstone/tests/index.html`).

3. One-Dimensional High-Order Method

It is very straightforward to extend the first-order method we discussed in the § 2 to high order. We like to achieve high order in both space and time.

The time integration can now be done with a high-order Runge-Kutta method. A very popular third-order Runge-Kutta scheme in computational gas dynamics was designed by Shu & Osher. The method combines the first-order forward Euler steps and involves prediction and correction. The method is the following,

$$\mathbf{U}^{(1)} = \mathbf{U}^n + \Delta t L(\mathbf{U}^n) \tag{9}$$

$$\mathbf{U}^{(2)} = \frac{3}{4}\mathbf{U}^n + \frac{1}{4}\mathbf{U}^{(1)} + \frac{1}{4}\Delta t L(\mathbf{U}^{(1)}) \tag{10}$$

$$\mathbf{U}^{n+1} = \frac{1}{3}\mathbf{U}^n + \frac{2}{3}\mathbf{U}^{(2)} + \frac{2}{3}\Delta t L(\mathbf{U}^{(2)}), \tag{11}$$

where $L(\mathbf{U})$ is the right hand side of Eq. 3, $\mathbf{U}^{n+1}$ is the final value after advancing one time step from $\mathbf{U}^n$.

To achieve high order in space, the left and right states at the cell interface must be reconstructed using high-order interpolation methods instead of using the values at the cell center. A simple high-order reconstruction method is the piecewise linear method (PLM) with a generalized minmod slope limiter. To obtain pressure, density and velocity of the left and right states at the cell interface $i + 1/2$, we need the states at $i - 1$, i, $i + 1$, and $i + 2$. Note that two ghost cells are needed at the boundaries. Given c_{i-1}, c_i, and c_{i+1}, the left-biased interface value of the left state reads,

$$c^{\mathrm{L}}_{i+1/2} = c_i + 0.5\,\mathrm{minmod}(\theta\,(c_i - c_{i-1}), 0.5\,(c_{i+1} - c_{i-1}), \theta\,(c_{i+1} - c_i)), \tag{12}$$

where c denotes pressure, density or velocity ,$1 \le \theta \le 2$, and the minmod function reads,

$$\mathrm{minmod}(x, y, z) = \frac{1}{4}|\mathrm{sgn}(x) + \mathrm{sgn}(y)|\,(\mathrm{sgn}(x) + \mathrm{sgn}(z))\,\min(|x|, |y|, |z|), \tag{13}$$

here the sgn function returns the sign of the number. This becomes the more diffusive normal minmod limiter when $\theta = 1$, and becomes the monotonized central-difference limiter when $\theta = 2$. We usually use $\theta = 1.5$. To obtain the right state at $i + 1/2$, variables c_i, c_{i+1} and c_{i+2} are used in the right-biased reconstruction with an expression similar to Eq. 12. More specifically, the right state at interface $i + 1/2$ reads,

$$c^{\mathrm{R}}_{i+1/2} = c_{i+1} - 0.5\,\mathrm{minmod}(\theta\,(c_{i+1} - c_i), 0.5\,(c_{i+2} - c_i), \theta\,(c_{i+2} - c_{i+1})). \tag{14}$$

4. Two-Dimensional High-Order Method

It is also very straightforward to extend the method to multi-dimensions. We will use 2D as an example. The 2D gas dynamics equations can be written as,

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = 0, \tag{15}$$

where, the conserved variable is $\mathbf{U} = (\rho, \rho v_x, \rho v_y, E)^T$, the flux in x-direction is $\mathbf{F} = (\rho v_x, \rho v_x^2 + P, \rho v_x v_y, (E+P)v_x)^T$, and the flux in x-direction is $\mathbf{G} = (\rho v_y, \rho v_x v_y, \rho v_y^2 + P, (E+P)v_y)^T$, here the total energy density is given by $E = \rho e + \frac{1}{2}\rho(v_x^2 + v_y^2)$.

Using the method of lines, the time dependent evolution of Eq. 15 can be expressed in the semi-discrete form

$$\frac{d\mathbf{U}_{i,j}}{dt} = L(\mathbf{U}) = -\frac{\mathbf{F}_{i+1/2,j} - \mathbf{F}_{i-1/2,j}}{\Delta x} - \frac{\mathbf{G}_{i,j+1/2} - \mathbf{G}_{i,j-1/2}}{\Delta y}, \tag{16}$$

where $\mathbf{F}_{i\pm1/2,j}$, and $\mathbf{G}_{i,j\pm1/2}$ are the fluxes at the cell interface for x- and y-direction respectively.

The methods described in previous sections can be used to obtain $\mathbf{F}_{i\pm1/2,j,k}$ and $\mathbf{G}_{i,j\pm1/2,k}$. Also the Runge-Kutta scheme can be used for time integration. You should be able to write a 2D hydro code based on this note.

For some 2D tests, see Jim Stone's webpage (`http://www.astro.princeton.edu/~jstone/tests/index.html`).