

Ethon hydrodynamics notes

Jonas Lippuner

January 8, 2021

Ethon is a free and open source software framework to test and develop hydrodynamics methods with block-based adaptive mesh refinement on various parallel computing architectures. It is available at <https://github.com/lanl/Ethon> and described in detail in Lippuner (2021). These notes are distributed together with the Ethon source code and they describe the hydrodynamics physics currently implemented in Ethon.

Throughout these notes, we will use index notation for vectors and derivatives. The vector \vec{x} will be denoted as x^i , where i ranges from 1 to 3. Furthermore, we denote time and spacial derivatives as follows.

$$\frac{\partial}{\partial t} = \partial_t \quad \text{and} \quad \frac{\partial}{\partial x_i} = \partial_i. \quad (1)$$

Note that we are in flat space-time, hence covariant and contravariant indices are the same. Repeated indices are summed over. For example, the gradient of a scalar and divergence of a vector field are written as

$$\nabla f = \partial_i f \quad \text{and} \quad \nabla \cdot \vec{u} = \partial_i u^i. \quad (2)$$

1 Conserved and primitive variables

We define the following **primitive** variables:

- Fluid density ρ
- Fluid velocity u^i
- Fluid internal specific energy e
- Fluid stress tensor P^{ij}

For now we only consider isotropic stresses and so we write $P^{ij} = p\delta^{ij}$, where p is the usual fluid pressure and δ^{ij} is the Kronecker delta.

The **conserved** variables are the ones that are actually evolved and they are:

- Mass density ρ

- Momentum density $\mu^i = \rho u^i$
- Total energy density $\epsilon = \rho e + \frac{1}{2}\rho u_i u^i$

Note that the total energy density is the sum of the internal energy density and kinetic energy density.

2 Compressible Euler equations

We write the conservation equations for the conserved variables as follows.

$$\partial_t \rho + \partial_j(\rho u^j) = 0 \quad (3)$$

$$\partial_t \mu^i + \partial_j(\rho u^i u^j + P^{ij}) = \rho g^i \quad (4)$$

$$\partial_t \epsilon + \partial_j(\epsilon u^j + u_i P^{ij}) = \rho u_i g^i, \quad (5)$$

where g^i is the external acceleration due to gravity. Note that we can write the above also in terms of the time derivatives of the primitive variables. The mass conservation equation (3) remains the same. The momentum conservation equation (4) becomes

$$\begin{aligned} & \partial_t(\rho u^i) + \partial_j(\rho u^i u^j + P^{ij}) = \rho g^i \\ \Leftrightarrow & u^i (\partial_t + \partial_j(\rho u^j)) + \rho \partial_t u^i + \rho u^j \partial_j u^i + \partial^j p \delta^{ij} = \rho g^i \\ \Leftrightarrow & \partial_t u^i + u^j \partial_j u^i + \frac{\partial^i p}{\rho} = g^i, \end{aligned} \quad (6)$$

where we used eq. (3), $\mu^i = \rho u^i$, and $P^{ij} = p \delta^{ij}$. Finally, for the energy conservation equation (5) we find

$$\begin{aligned} & \partial_t \left(\rho e + \frac{1}{2} \rho u_i u^i \right) + \partial_j \left(\rho e u^j + \frac{1}{2} \rho u_i u^i u^j + u_i P^{ij} \right) = \rho u_i g^i \\ \Leftrightarrow & e \partial_t \rho + \rho \partial_t e + \frac{1}{2} u_i u^i \partial_t \rho + \rho u_i \partial_t u^i \\ & + e \partial_j(\rho u^j) + \rho u^j \partial_j e + \frac{1}{2} u_i u^i \partial_j(\rho u^j) + \rho u^j u_i \partial_j u^i \\ & + u_i \partial^i p + p \partial^i u_i = \rho u_i g^i \\ \Leftrightarrow & e (\partial_t \rho + \partial_j(\rho u^j)) \\ & + \frac{1}{2} u_i u^i (\partial_t \rho + \partial_j(\rho u^j)) \\ & + \rho u_i \left(\partial_t u^i + u^j \partial_j u^i + \frac{\partial^i p}{\rho} - g^i \right) \\ & + \rho \partial_t e + \rho u^j \partial_j e + p \partial^i u_i = 0 \\ \Leftrightarrow & \partial_t e + u^i \partial_i e + \frac{p}{\rho} \partial^i u_i = 0, \end{aligned} \quad (7)$$

where we used eqs. (3) and (6), $\epsilon = \rho e + \frac{1}{2}\rho u_i u^i$, and $P^{ij} = p \delta^{ij}$.

If shocks are present, non-conservative methods do not converge to the correct solution, so one must use conservative methods in those cases (which do converge to the correct solution, if they converge at all), see Toro (2013, §5.3).

For notational simplicity, we shall write the conservative form of the compressible Euler equations (3) to (5) as follows

$$\partial_t \mathbf{U} + \partial_j \mathbf{F}^j(\mathbf{U}) = \mathbf{S}(\mathbf{U}), \quad (8)$$

where the state vector \mathbf{U} , the flux tensor $\mathbf{F}^j(\mathbf{U})$, and the source term $\mathbf{S}(\mathbf{U})$ are given by

$$\mathbf{U} = \begin{bmatrix} \rho \\ \mu^1 \\ \mu^2 \\ \mu^3 \\ \epsilon \end{bmatrix}, \quad \mathbf{F}^j(\mathbf{U}) = \begin{bmatrix} \rho u^j \\ \rho u^1 u^j + P^{1j} \\ \rho u^2 u^j + P^{2j} \\ \rho u^3 u^j + P^{3j} \\ \epsilon u^j + u_i P^{ij} \end{bmatrix}, \quad \mathbf{S}(\mathbf{U}) = \begin{bmatrix} 0 \\ \rho g^1 \\ \rho g^2 \\ \rho g^3 \\ \rho u_i g^i \end{bmatrix}. \quad (9)$$

Note that to evaluate $\mathbf{F}^j(\mathbf{U})$ we need an additional Equation of State (EOS) that allows us to recover the primitive variables from the conservative ones, which are contained in \mathbf{U} . Note that all of the above quantities are functions of time and space.

3 Finite-volume scheme

To discretize the non-linear, partial-differential Euler equations (3) to (5) we introduce a Cartesian mesh. For simplicity, we initially consider a uniform mesh with cells of size Δx^i , where the sizes in the three dimensions can vary, in principle (in practice they are usually the same, though). Furthermore, we consider a cell-centered method where we have grid points at the centers of the cells and the cell-averaged quantities are stored at those grid points. Additionally, we initially restrict ourselves to the case without source terms, i.e. $\mathbf{S} = \mathbf{0}$ and so we have the system

$$\partial_t \mathbf{U} + \partial_j \mathbf{F}^j(\mathbf{U}) = \mathbf{0}. \quad (10)$$

Now consider a cell with index α and volume

$$V_\alpha = \Delta x^1 \Delta x^2 \Delta x^3. \quad (11)$$

We define the cell average of the state vector in cell α as

$$\bar{\mathbf{U}}_\alpha(t) = \frac{1}{V_\alpha} \int_{V_\alpha} \mathbf{U}(x^i, t) dV. \quad (12)$$

Since $\partial_t \mathbf{U} = -\partial_j \mathbf{F}^j(\mathbf{U})$, we have

$$\mathbf{U}(x^i, t_2) = \mathbf{U}(x^i, t_1) - \int_{t_1}^{t_2} \partial_j \mathbf{F}^j(\mathbf{U}(x^i, t)) dt, \quad (13)$$

and so

$$\begin{aligned}\bar{\mathbf{U}}_\alpha(t_2) &= \frac{1}{V_\alpha} \int_{V_\alpha} \left(\mathbf{U}(x^i, t_1) - \int_{t_1}^{t_2} \partial_j \mathbf{F}^j(\mathbf{U}(x^i, t)) dt \right) dV \\ &= \bar{\mathbf{U}}_\alpha(t_1) - \frac{1}{V_\alpha} \int_{t_1}^{t_2} \int_{\partial V_\alpha} n_j \mathbf{F}^j(\mathbf{U}(x^i, t)) dS dt,\end{aligned}\quad (14)$$

where we assumed that the flux \mathbf{F} is well behaved so that we can change the order of integration, and we used the divergence theorem to replace the volume integral of the divergence of \mathbf{F} with the surface integral over the cell α , where n_j is the normal vector of the cell surface. Note that eq. (14) is exact since we have made no approximations yet.

The surface integral in eq. (14) has six components, namely the integral of the flux across the six faces of the cell. For cell α we define the following quantities

$$\begin{aligned}x_\alpha^i &= \text{cell center location,} \\ x_{\alpha,L}^i &= \text{coordinates of the lower cell faces, and} \\ x_{\alpha,U}^i &= \text{coordinates of the upper cell faces.}\end{aligned}\quad (15)$$

Note that

$$x_{\alpha,U}^i - x_{\alpha,L}^i = \Delta x^i \quad \text{and} \quad x_\alpha^i = \frac{x_{\alpha,L}^i + x_{\alpha,U}^i}{2}.\quad (16)$$

We can now write eq. (14) as

$$\bar{\mathbf{U}}_\alpha(t_2) = \bar{\mathbf{U}}_\alpha(t_1) + \frac{\Delta t}{\Delta x^i} (\bar{\mathbf{F}}_{\alpha,L}^i - \bar{\mathbf{F}}_{\alpha,U}^i),\quad (17)$$

where $\Delta t = t_2 - t_1$, and $\bar{\mathbf{F}}_{\alpha,L}^i$ and $\bar{\mathbf{F}}_{\alpha,U}^i$ are the average fluxes through the lower and upper faces in the i -direction. Their exact form is

$$\bar{\mathbf{F}}_{\alpha,L}^i = \frac{\Delta x^i}{\Delta t V_\alpha} \int_{t_1}^{t_2} \int_{V_\alpha} \mathbf{F}^i(\mathbf{U}(x^1, x^2, x^3, t)) \delta(x^i - x_L^i) dV dt \quad (\text{no sum over } i),\quad (18)$$

where the Dirac-Delta function reduces the integral over the cell volume to an integral over the lower cell face in direction i with coordinate x_L^i . The expression for $\bar{\mathbf{F}}_{\alpha,U}^i$ is analogous, since we have already taken care of the sign of the normal vector of the lower and upper faces in eq. (17).

Obviously, we cannot evaluate eq. (18) directly, because that would require knowledge of the solution \mathbf{U} at all times and all points in space. One of the core ingredients of a finite volume method is thus to prescribe some numeric approximations for $\bar{\mathbf{F}}_{\alpha,L}^i$ and $\bar{\mathbf{F}}_{\alpha,U}^i$ that can be readily evaluated.

3.1 Godunov scheme

The Godunov scheme is one of the simplest finite volume schemes and it is first-order accurate. We compute the fluxes by solving the Riemann Problem between the two cells, i.e.

$$\bar{\mathbf{F}}_{\alpha,L}^j = \text{RP}(\bar{\mathbf{U}}_{\alpha-1}, \bar{\mathbf{U}}_{\alpha}), \quad (19)$$

$$\bar{\mathbf{F}}_{\alpha,U}^j = \text{RP}(\bar{\mathbf{U}}_{\alpha}, \bar{\mathbf{U}}_{\alpha+1}), \quad (20)$$

where $\bar{\mathbf{U}}_{\alpha-1}$ and $\bar{\mathbf{U}}_{\alpha+1}$ are the average states of the cells adjacent to cell α in the lower and upper direction in dimension j , respectively, and we use RP to denote the solution to the Riemann Problem. We can use these fluxes in eq. (17) to update the state in cell α .

3.2 MUSCL-Hancock scheme

The MUSCL-Hancock scheme is a popular finite volume scheme that is second-order accurate in time and space. MUSCL stands for Monotone Upstream-centred Scheme for Conservation Laws. For example, FLASH uses MUSCL-Hancock for its dimensionally unsplit hydro solver. See (Toro, 2013, §13.4, §14.4, §16.5) for more details.

The central idea in the MUSCL-Hancock scheme is to replace the constant cell averages by a linearly varying quantity within the cell. Considering only one dimension for a moment, we define the conserved quantity $\mathbf{U}_{\alpha}(x)$ in the cell α as

$$\mathbf{U}_{\alpha}(x) = \bar{\mathbf{U}}_{\alpha} + \frac{x - x_{\alpha}}{\Delta x} \Delta_{\alpha} \quad \text{for } x \in [x_{\alpha,L}, x_{\alpha,U}], \quad (21)$$

where Δ_{α} is a suitably chosen slope vector (of the five conserved quantities) of the solution inside the cell (see section 3.3 for details on how to construct the slope vector). The above is at a fixed time. We now evaluate the above at the lower and upper boundaries to get the extrapolated boundary values of \mathbf{U} :

$$\mathbf{U}_{\alpha}^L = \mathbf{U}_{\alpha}(x_{\alpha,L}) = \bar{\mathbf{U}}_{\alpha} - \frac{\Delta_{\alpha}}{2} \quad \text{and} \quad \mathbf{U}_{\alpha}^U = \mathbf{U}_{\alpha}(x_{\alpha,U}) = \bar{\mathbf{U}}_{\alpha} + \frac{\Delta_{\alpha}}{2}. \quad (22)$$

We now evolve the boundary extrapolated conserved quantities for a half timestep $\Delta t/2$ according to eq. (17) by evaluating \mathbf{F} for the boundary extrapolated values \mathbf{U}_{α}^L and \mathbf{U}_{α}^U and using these as the approximate fluxes. Hence we get

$$\begin{aligned} \hat{\mathbf{U}}_{\alpha}^L &= \mathbf{U}_{\alpha}^L + \frac{\Delta t}{2\Delta x} [\mathbf{F}(\mathbf{U}_{\alpha}^L) - \mathbf{F}(\mathbf{U}_{\alpha}^U)], \\ \hat{\mathbf{U}}_{\alpha}^U &= \mathbf{U}_{\alpha}^U + \frac{\Delta t}{2\Delta x} [\mathbf{F}(\mathbf{U}_{\alpha}^L) - \mathbf{F}(\mathbf{U}_{\alpha}^U)], \end{aligned} \quad (23)$$

where all the right-hand side terms are evaluated at the current time t_1 . Finally, to obtain the actual approximate fluxes at the cell boundaries, we solve the one-dimensional local Riemann problem at the cell boundaries using the

evolved boundary extrapolated quantities. A thorough discussion of the Riemann Problem is beyond the scope of these notes. Simply put, the solution to Riemann Problem roughly provides the flux across a discontinuity namely a cell face. For more details, see Toro (2013). The Riemann Problem requires left and right initial states \mathbf{U}_L and \mathbf{U}_R and produces a similarity solution $\mathbf{U}(x/t)$, which we will denote by

$$\mathbf{U}(x/t) \equiv \text{RP}(\mathbf{U}_L, \mathbf{U}_R). \quad (24)$$

At the lower cell boundary of cell α , the left initial state for the Riemann problem is $\hat{\mathbf{U}}_{\alpha-1}^U$ and the right initial state is $\hat{\mathbf{U}}_{\alpha}^L$, where $\alpha - 1$ denotes the cell to the left of cell α in the x -direction. The initial states for the Riemann Problem at the upper cell boundary of cell α are analogous. Hence we have

$$\begin{aligned} \mathbf{U}_{\alpha}^L(x/t) &= \text{RP}(\hat{\mathbf{U}}_{\alpha-1}^U, \hat{\mathbf{U}}_{\alpha}^L), \\ \mathbf{U}_{\alpha}^U(x/t) &= \text{RP}(\hat{\mathbf{U}}_{\alpha}^U, \hat{\mathbf{U}}_{\alpha+1}^L), \end{aligned} \quad (25)$$

where $\alpha + 1$ is the cell to the right of cell α in the x -direction. We now use the Riemann similarity solutions evaluated at $x/t = 0$ to get the final approximate fluxes for eq. (17), thus we have

$$\bar{\mathbf{F}}_{\alpha,L} = \mathbf{F}(\mathbf{U}_{\alpha}^L(0)) \quad \text{and} \quad \bar{\mathbf{F}}_{\alpha,U} = \mathbf{F}(\mathbf{U}_{\alpha}^U(0)). \quad (26)$$

Note that in principle, we could solve the Generalized Riemann Problem where the left and right states are not constant. We could use a linear form similar to eq. (21) to describe the left and right states for the Riemann Problems at the boundaries. However, the Generalized Riemann Problem is exceedingly difficult and not typically employed (Toro, 2013, §13.4.1).

Generalizing the above to three dimensions is straight forward. We still solve one-dimensional, piece-wise constant Riemann Problems at the cell boundaries (now there are six instead of two), but for the time evolution in eq. (23) we incorporate the fluxes from all six faces. We now have a different slope vector (see section 3.3) in each dimension and we'll denote the component in dimension i by Δ_{α}^i . We have six boundary extrapolated values, namely

$$\mathbf{U}_{\alpha}^{L,i} = \bar{\mathbf{U}}_{\alpha} - \frac{\Delta_{\alpha}^i}{2} \quad \text{and} \quad \mathbf{U}_{\alpha}^{U,i} = \bar{\mathbf{U}}_{\alpha} + \frac{\Delta_{\alpha}^i}{2}, \quad (27)$$

which are evolved forward in time by $\Delta t/2$ to yield

$$\hat{\mathbf{U}}_{\alpha}^{[L|U],i} = \mathbf{U}_{\alpha}^{[L|U],i} + \frac{\Delta t}{2\Delta x^j} [\mathbf{F}^j(\mathbf{U}_{\alpha}^{L,j}) - \mathbf{F}^j(\mathbf{U}_{\alpha}^{U,j})]. \quad (28)$$

Now we solve the Riemann Problem at all six faces using one evolved boundary extrapolated value from this cell and one from the adjacent cell, just like in eq. (25). This yields a similarity solution at each face, which we plug into the flux function \mathbf{F} to get the approximate flux $\bar{\mathbf{F}}_{\alpha,[L|U]}^i$ through that face. The cell averaged conserved quantity $\bar{\mathbf{U}}_{\alpha}$ is then updated according to eq. (17).

3.3 Slope construction and limiter methods

We will restrict ourselves again to one dimension in this section, since it drastically simplifies notation. The components of the slope vector in the different dimensions are all determined independently from each other, so generalizing this discussion to multiple dimension is trivial. We use $\alpha + 1$ to denote the neighboring cell in the positive direction and $\alpha - 1$ the neighbor in the negative direction. We follow (LeVeque, 2002, §6). A general form of the slope is

$$\Delta_\alpha = \frac{1}{2}(1 + \omega) (\bar{U}_\alpha - \bar{U}_{\alpha-1}) + \frac{1}{2}(1 - \omega) (\bar{U}_{\alpha+1} - \bar{U}_\alpha), \quad (29)$$

where ω is a free parameter in the interval $[-1, 1]$. Three common choices for ω are

$$\text{Upwind } (\omega = 1): \quad \Delta_\alpha^{\text{up}} = \bar{U}_\alpha - \bar{U}_{\alpha-1} \quad (\text{Beam-Warming}). \quad (30)$$

$$\text{Centered } (\omega = 0): \quad \Delta_\alpha^{\text{cent}} = \frac{\bar{U}_{\alpha+1} - \bar{U}_{\alpha-1}}{2} \quad (\text{Fromm}), \quad (31)$$

$$\text{Downwind } (\omega = -1): \quad \Delta_\alpha^{\text{down}} = \bar{U}_{\alpha+1} - \bar{U}_\alpha \quad (\text{Lax-Wendroff}), \quad (32)$$

Using any of these slopes leads to a second-order method. Unfortunately, with the increased accuracy also come spurious oscillations near discontinuities. To avoid these discontinuities, the slope computed above needs to be limited in a way that ensures that the total variation of the solution does not increase. The resulting scheme is said to be total variation diminishing (TVD). Let $\bar{\Delta}_\alpha$ denote the limited slope and define the minmod and maxmod functions as

$$\text{minmod}(a, b) = \begin{cases} 0 & \text{if } ab \leq 0, \\ a & \text{if } |a| \leq |b| \text{ and } ab > 0, \\ b & \text{if } |b| \leq |a| \text{ and } ab > 0. \end{cases} \quad (33)$$

$$\text{maxmod}(a, b) = \begin{cases} 0 & \text{if } ab \leq 0, \\ a & \text{if } |a| \geq |b| \text{ and } ab > 0, \\ b & \text{if } |b| \geq |a| \text{ and } ab > 0. \end{cases} \quad (34)$$

Some popular limiters are the following.

$$\text{minbee: } \bar{\Delta}_\alpha = \text{minmod}(\Delta_\alpha^{\text{up}}, \Delta_\alpha^{\text{down}}) \quad (35)$$

$$\text{superbee: } \bar{\Delta}_\alpha = \text{maxmod}[\text{minmod}(\Delta_\alpha^{\text{up}}, 2\Delta_\alpha^{\text{down}}), \text{minmod}(2\Delta_\alpha^{\text{up}}, \Delta_\alpha^{\text{down}})] \quad (36)$$

$$\text{MC: } \bar{\Delta}_\alpha = \begin{cases} 0 & \text{if } \Delta_\alpha^{\text{up}} \Delta_\alpha^{\text{down}} \leq 0, \\ \min(2|\Delta_\alpha^{\text{up}}|, 2|\Delta_\alpha^{\text{down}}|, |\Delta_\alpha^{\text{cent}}|) & \text{if } \Delta_\alpha^{\text{cent}} > 0, \\ -\min(2|\Delta_\alpha^{\text{up}}|, 2|\Delta_\alpha^{\text{down}}|, |\Delta_\alpha^{\text{cent}}|) & \text{if } \Delta_\alpha^{\text{cent}} < 0. \end{cases} \quad (37)$$

$$\text{van Leer: } \bar{\Delta}_\alpha = \begin{cases} 0 & \text{if } \Delta_\alpha^{\text{up}} \Delta_\alpha^{\text{down}} \leq 0, \\ \frac{2\Delta_\alpha^{\text{up}} \Delta_\alpha^{\text{down}}}{\Delta_\alpha^{\text{up}} + \Delta_\alpha^{\text{down}}} & \text{otherwise.} \end{cases} \quad (38)$$

Note that the minbee limiter is also called minmod limiter.

4 γ -Law Equation of State

Let γ be the adiabatic index (ratio constant-pressure heat capacity to constant-volume heat capacity). The pressure is given by

$$P = (\gamma - 1)\rho e, \quad (39)$$

where ρ is the mass density and e is the specific internal energy. We have $\gamma = 5/3$ for monoatomic ideal gases and $\gamma = 7/5$ for diatomic ideal gases. From the ideal gas law we also have

$$PV = Nk_B T \Rightarrow P = \frac{N}{V} k_B T = nk_B T = \frac{\rho}{m} k_B T, \quad (40)$$

where V is the volume, N is the number of particles, $n = N/V$ is the number density, and m is the mass of a particle. Equating the above gives

$$e = \frac{1}{\gamma - 1} \frac{k_B T}{m}. \quad (41)$$

For an isentropic (constant entropy, i.e. adiabatic and reversible) process, one can derive

$$P = \alpha \rho^\gamma, \quad (42)$$

for some constant γ . The sound speed can now be computed as

$$c_s^2 = \left. \frac{dP}{d\rho} \right|_S = \gamma \alpha \rho^{\gamma-1} = \gamma \frac{P}{\rho} = \gamma(\gamma - 1)e. \quad (43)$$

Acknowledgements

The development of Ethon was funded by the Laboratory Directed Research and Development program of Los Alamos National Laboratory under project number 20190519ECR. The development used LANL Institutional Computing Program resources. LANL is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the U.S.DOE (Contract No. 89233218CNA000001).

References

- LeVeque, R. J. 2002, Finite volume methods for hyperbolic problems, Vol. 31 (Cambridge university press)
- Lippuner, J. 2021, (in preparation)
- Toro, E. F. 2013, Riemann solvers and numerical methods for fluid dynamics: a practical introduction (Springer Science & Business Media)