Numerical simulation of twin jets of matter emerging from an active galactic nucleus using the equations of special relativistic hydrodynamics

Darran Furnival

December 18, 2002

Reminder

- Physics: Active galactic nuclei inject relativistic jets of matter into the surrounding intracluster medium.

- Project goal: To develop a suite of software that can be used to investigate this process taking special relativity into account.

- Model: To solve the equations of special relativistic hydrodynamics in a region of space gravitationally bound by the ambient dark matter.

Equations of Special Relativistic Hydrodynamics

- Equations in *conservation law* form:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}^i(\mathbf{U})}{\partial x^i} = 0$$

$$\mathbf{U} = \begin{bmatrix} D \\ S^1 \\ S^2 \\ S^3 \\ \tau \end{bmatrix} \quad \mathbf{F}^i = \begin{bmatrix} Dv^i \\ S^1 v^i + p\delta^{1i} \\ S^2 v^i + p\delta^{2i} \\ S^3 v^i + p\delta^{3i} \\ S^i - Dv^i \end{bmatrix}$$

$D \equiv$ rest mass density

$S^i \equiv$ momentum density

$\tau \equiv$ energy density

- Conserved variables related to primitive variables by:

$$D = \rho\Gamma$$
$$S^i = \rho h \Gamma^2 v^i$$
$$\tau = \rho h \Gamma^2 - p - D$$
$$h = 1 + \epsilon + \frac{p}{\rho}$$

$$\rho \equiv \text{proper rest mass density}$$
$$h \equiv \text{specific enthalpy}$$
$$v^i \equiv \text{velocity}$$
$$p \equiv \text{pressure}$$
$$\epsilon \equiv \text{internal energy}$$

$\Gamma = (1 - v^2)^{-1/2}$ is the Lorentz factor.

- Equation of state:

$$p = p(\rho, \epsilon)$$

- Also need to include gravitational potential due to dark matter.

Method of lines - Discretize in space to leave a system of ordinary differential equations.

$$\begin{aligned}\frac{d\mathbf{U}_{ij}}{dt} &= -\frac{1}{dx}(\mathbf{F}_{i+1/2,j} - \mathbf{F}_{i-1/2,j}) \\ &\quad - \frac{1}{dy}(\mathbf{F}_{i,j+1/2} - \mathbf{F}_{i,j-1/2}) \\ &= L(\mathbf{U}_{ij})\end{aligned}$$

- Present code uses explicit Euler method.

- Future improvement will use RK4.

Calculation of $L(\mathbf{U}_{ij})$

- Evaluate flux at cell centers. Need primitive variables as well as conserved variables at each time step to do this.

- Evaluate flux at cell boundaries using a function of the form:

$$\mathbf{F}_{BDY} = \mathbf{F}_{BDY}(\mathbf{F}_L, \mathbf{F}_R).$$

Choice of function F

- Artificial viscosity:

$$\mathbf{F}_{BDY} = \frac{1}{2}(\mathbf{F}_L + \mathbf{F}_R) - \frac{h}{2\Delta t}(\mathbf{U}_R - \mathbf{U}_L)$$

  Used with explicit Euler method gives Lax-Friedrichs scheme. Present code uses this choice of F.

- Future code will use Roe-type relativistic Riemann solver which can capture shocks at ultra-relativistic speeds.

Boundary conditions

- Boundary conditions enforced by use of ghost cells. Present code supports inflow, outflow, and reflective boundary conditions.

- In present code user defines:
  - Computational domain.
  - Number of uniform cells in each direction.
  - Boundary condition in each ghost cell.

- Future code should allow non-uniform grids to allow higher resolution where needed and lower resolution elsewhere.

Equation of state

- Present code uses the equation of state for an ideal polytropic gas

$$p = (\gamma - 1)\epsilon$$

with $\gamma = 5/3$.

- Future code should discern relativistic regime $\gamma = 4/3$ from non-relativistic regime $\gamma = 5/3$.

Recovery of primitive variables

- Use Newton-Raphson method to obtain root of

$$f(p) = (\gamma - 1)\rho\epsilon - p$$

and then

$$\begin{aligned} v^i &= \frac{S^i}{\tau + D + p} \\ \rho &= \frac{D}{\Gamma} \\ \epsilon &= \frac{\tau + D(1 - \Gamma) + p(1 - \Gamma^2)}{D\Gamma} \end{aligned}$$

- This is not yet implemented in present code.

Gravitation

- Initially solve Poisson equation

$$\nabla^2 \Phi = 4\pi G \rho_{DM}$$

where $\rho_{DM} \equiv$ dark matter density is known and remains constant.

- Non-dark matter also contributes to gravitational field but this will be neglected in code.

Parallelization

- Domain decomposition will be used, i.e. computational domain will be divided into the subdomains which will be distributed to the different processors.

- This is not yet implemented in present code.

Summary

- Where am I now: Almost have 0th order 2-dimensional code - needs Newton-Raphson module.

- What has changed: The general goal has stayed the same. One new idea is to incorporate Euler equations and discern between relativistic regimes and non-relativistic regimes.

- Future plan:
  - Complete 0th order 2-dimensional code.
  - Parallelize code.
  - Improve modules.
  - Generalize to 3-dimensions.