

# Torus 2D hydro simulation, pseudo-Newtonian

Two-dimensional (axially symmetric) hydrodynamical simulation of a RIAF without cooling. The calculations begin from an equilibrium configuration consisting of a thick torus (Papaloizou-Pringle) with constant specific angular momentum. Accretion is induced by the addition of a small anomalous azimuthal shear stress which.

## Code units

$$G = M = 1$$

$$c = \sqrt{2}$$

such that the Schwarzschild radius is  $r_S = \frac{2GM}{c^2} = 1$ .

We usually parametrize time in units of the orbital time at the radius of maximum density (or pressure),

$$t_{\text{orb}}(r_0) = \frac{2\pi r_0}{v_\phi(r_0)}$$

where  $v_\phi(r_0)$  is the Keplerian velocity for the appropriate potential at  $r_0$ .

## Viscous stress tensor

Parametrized following Stone et al. (1999) with the kinematic viscosity given by

$$\nu = \alpha \frac{c_s^2}{\Omega_k} \approx \alpha r^{-1/2}$$

as appropriate for a RIAF. The shear stress is defined in file `visc_nu.c`.

## Gravity

GR effects are incorporated via the Paczynski-Wiita potential,

$$\Phi = -\frac{GM}{r - r_S}$$

Notice that the Keplerian velocity is not simply  $\Omega_k = (\frac{GM}{r^3})^{1/2}$  with this pseudo-Newtonian potential but is instead given by

$$\Omega_k = \left(\frac{GM}{r}\right)^{1/2} \left(\frac{1}{r - r_S}\right)$$

The potential is defined in `init.c`.

## Coordinate system and grid

Coordinate system: spherical polar

Resolution:  $N_r \times N_\theta = 400 \times 200$  (cf. `pluto.ini`)

Radial range:  $r = 1.25 - 1000$  spanning three orders of magnitude in radius

## Non-uniform grid in $\theta$

A non-uniform grid in the  $\theta$ -direction is adopted, because we are not necessarily interested in having resolution towards the poles, since we are not simulating relativistic jets. According to Moscibrodzka et al. (2014), the jet is produced inside a cone of angle  $25^\circ$ . Therefore, compared to a uniform grid in  $\theta$ , our approach is to have  $1/3$  of the uniform grid resolution within the “jet cone”.

The formula for calculating the required number of elements inside one half of the jet cone is the following:

$$N_\theta(\text{jet}) = N_\theta(\text{total}) \left(\frac{25^\circ}{180^\circ}\right) (\text{reduction factor})$$

where we typically use `reduction factor = 1/3`.

For a concrete example, suppose we choose  $N_\theta = 100$  and decide to decrease the resolution to  $1/3$  within  $25^\circ$  of the poles. According to the formula above, one has  $n = 5$  inside each half of the cone. Therefore, we would have 5 elements in the upper half of the cone, 5 in the second half and 90 elements in the rest of the theta domain.

In Pluto, you can insert this modified resolution in `pluto.ini` as follows:

```
X2-grid 3      0.0  5 u 0.436 90 u 2.705 5 u 3.1415
```