Code construction – 4: Surface fitted coordinate for the fluid and hydrostatic eq.

On top of the spherical coordinates for the field (r, θ, ϕ), we introduce surface fitted coordinates for the fluid as follows.

$$\begin{cases} r_f = \dfrac{r}{R(\theta,\phi)} \\ \theta_f = \theta \\ \phi_f = \phi \end{cases}$$

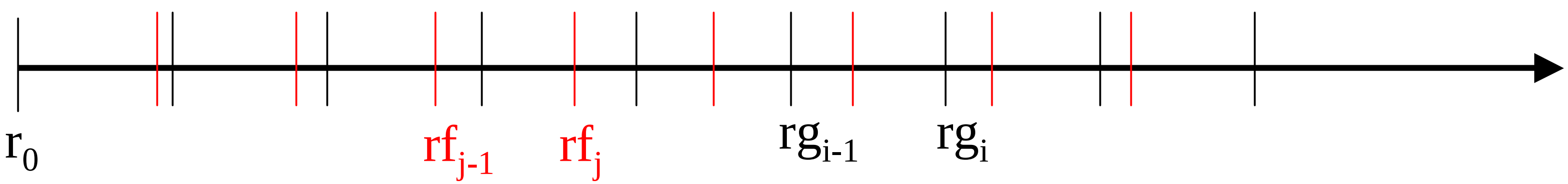
where R(θ,ϕ) is the surface of a star.

$$r_f \in [0,1], \quad \theta_f \in [0,\pi], \quad \phi_f \in [0,2\pi].$$

The surface R(θ, ϕ) is normalized by the value at θ = π/2, and ϕ = 0.
$R_0 := R(\pi/2, 0)$.

subroutine interpolation_grav_to_fluid

θ, and ϕ coordinates are the same for the field and fluid coordinates. So, an interpolation along the radial coordinate (1D interpolation) is required.
It is recommended to use an interpolating formula higher than the 2nd order.



end interpolation_grav_to_fluid

subroutine coordinate_patch_kit_fluid

We use the same grid spacing for the fluid coordinate (r_f, θ_f, ϕ_f) as the coordinate grids for the gravitational field.

Radial coordinate input parameters: nrf. ($r_f = r_g$ up to $r_f = 1$)

$R(\theta,\phi)$: $r_f \in [0, 1]$, the grid spacing rule is the same as r_g.

nrf : Total number of radial grid points (-1).

rf(ir), ir = 0, nrf : Radial coordinate grid points r_i .
hrf(ir) : Mid-point of radial grids.
drf(ir) : Radial grid spacing $\Delta r_i := r_i - r_{i-1}$.

θ coordinate input parameter: ntf.

$\theta \in [0, \pi]$, the grid spacing is equidistant.

ntf : Total number of θ grid points (-1).

thf(it), it = 0, ntf : θ grid points θ_i .
hthf(it) : Mid-point of θ grids $\theta_{i+1/2}$.
dthf : θ grid spacing $\Delta\theta$.

Functions associated with θ coordinate.

Trigonometric funcitons.
sinthf(it), it = 0, ntf : $\sin\theta_i$. hsinthf(it) : $\sin\theta_{i+1/2}$ sin at mid-point of θ grids.
costhf(it), it = 0, ntf : $\cos\theta_i$. hcosthf(it) : $\cos\theta_{i+1/2}$ cosin at mid-point of θ grids.

ϕ coordinate input parameter: npf.

$\phi \in [0, 2\pi]$, the grid spacing is equidistant.

npf : Total number of ϕ grid points (-1).

phif(ip), ip = 0, npf : ϕ grid points ϕ_i .
hphif(ip) : Mid-point of ϕ grids $\phi_{i+1/2}$.
dphif : ϕ grid spacing $\Delta\phi$.

Functions associated with ϕ coordinate.

Trigonometric funcitons.
sinphif(ip), ip = 0, npf : $\sin\phi_i$. hsinphif(ip) : $\sin\phi_{i+1/2}$ sin at mid-point of ϕ grids.
cosphif(ip), ip = 0, npf : $\cos\phi_i$. hcosphif(ip): $\cos\phi_{i+1/2}$ cos at mid-point of ϕ grids.

Weight for the integration: assigned at the mid-points.

wrf(irr) = hrf(irr)^2 drf(irr) : weight for the radial integratoin $r_i^2 \Delta r_i$
wtf(itt) = hsinthef(itt) dthf : weight for the θ integratoin $\sin\theta_i \Delta\theta_i$
wpf(ipp) = dphif : weight for the ϕ integratoin $\Delta\phi_i$
wrtpf(irr,itt,ipp) = hR(itt,ipp)^3 $\times$ wrf(irr) $\times$ wtf(itt) $\times$ wpf(ipp)

hR(itt,ipp) is the radius of the stellar surface at the mid-points of (θ, ϕ) grids.
That is, at hth(itt), and hphi(ipp) .

$$hR(itf, ipf) = \frac{1}{4} \sum_{i=itf-1}^{itf} \sum_{j=ipf-1}^{ipf} R(i,j)$$

end subroutine coordinate_patch_kit_fluid subroutine rotating_compact_star

call interpolation_grav_to_fluid

call hydrostatic_equation

call update_matter

call update_surface

call update_parameter

end subroutine rotating_compact_star

Interpolate $\{\alpha, \psi, \beta^a\}$ which are used to compute u^t.

$$\left\{ \begin{array}{l} \dfrac{h}{u^t} = \text{const} \\ u^t = \dfrac{1}{\sqrt{\alpha^2 - \omega_a \omega^a}} = \dfrac{1}{\sqrt{\alpha^2 - \psi^4 f_{ab}\, \tilde{\omega}^a \tilde{\omega}^b}}, \\ \text{where} \quad \tilde{\omega}^a = \tilde{\beta}^a + \Omega \phi^a. \end{array} \right.$$

Same as the gravitational field (a different value for λ might make a convergence faster.)

$$\phi^{(\mathrm{N+1})} = \lambda \phi^{(\mathrm{new})} + (1 - \lambda)\phi^{(\mathrm{N})}$$

where $\phi = h - 1$ is one of choice. $R(\theta, \phi)$ is also updated in the same way.

Recall: polytropic (adiabatic) EOS.

Conservation of the energy density is written

$$d\epsilon = Tds + hd\rho.$$

This is written using the relativistic enthalpy per baryon mass

$$h := \frac{\epsilon + p}{\rho},$$

$$dh = d\left(\frac{\epsilon + p}{\rho}\right) = Tds + \frac{1}{\rho}dp.$$

When the specific entropy s is constant over the whole fluid, the flow is called isentropic, and consequently the fluid admit one-parameter EOS.

We consider a polytopic (or adiabatic) EOS

$$p = K\rho^{\Gamma},$$

where K is the adiabatic constant and Γ is the adiabatic index (with $\Gamma > 1$).

Since $dh = dp/\rho = \Gamma/(\Gamma - 1)d(p/\rho)$, and $h \to 1$ as $p/\rho \to 0$, we have

$$h = 1 + \frac{\Gamma}{\Gamma - 1}\frac{p}{\rho}.$$

It is convenient to introduce a variable q as

$$q := \frac{p}{\rho}.$$

Defining a polytropic index $n := 1/(\Gamma - 1)$ we have

$$\begin{aligned} \rho &= K^{-n}q^{n}, \\ p &= K^{-n}q^{n+1}, \\ h &= 1 + (n+1)\,q. \end{aligned}$$

subroutine update_parameter

$$\frac{h}{u^t} = \mathcal{E} = \text{const} \qquad u^t = \frac{1}{\sqrt{\alpha^2 - \omega_a \omega^a}} = \frac{1}{\sqrt{\alpha^2 - \psi^4 f_{ab} \tilde{\omega}^a \tilde{\omega}^b}},$$

Same as the Newtonian calculation, we have three parameters, $\{\Omega, \mathcal{E}, R_0\}$
Three conditions are imposed at the center, and two points at the surface.
When the length scale is updated from R_i to R_0 , the lapse and the conformal factor changes as, $\ln \psi \to \left(\frac{R_0}{R_i}\right)^2 \ln \psi, \quad \ln \alpha \to \left(\frac{R_0}{R_i}\right)^2 \ln \alpha$

(When an iteration is made, the physical size of a star may change. While in numerical computation, we normalize the size (to set R(π/2,0)=1) and update the length scale R_0.) $h^2 \left(\alpha^{2(R_0/R_i)^2} - \psi^{4(R_0/R_i)^2} f_{ab} \tilde{\omega}^a \tilde{\omega}^b\right) = \mathcal{E}^2,$

Three conditions $h=h_c$ at the center, $h=1$ at R_{eq} and R_p, are applied to

$$\ln h + R_0^2 \ln \alpha^{1/R_i^2} + \frac{1}{2} \ln \left\{ 1 - \left(\frac{\psi^4}{\alpha^2}\right)^{(R_0/R_i)^2} f_{ab} \tilde{\omega}^a \tilde{\omega}^b \right\} = \mathcal{E}^2,$$

and solved for the parameters $\{\Omega, \mathcal{E}, R_0\}$.

end subroutine update_parameter