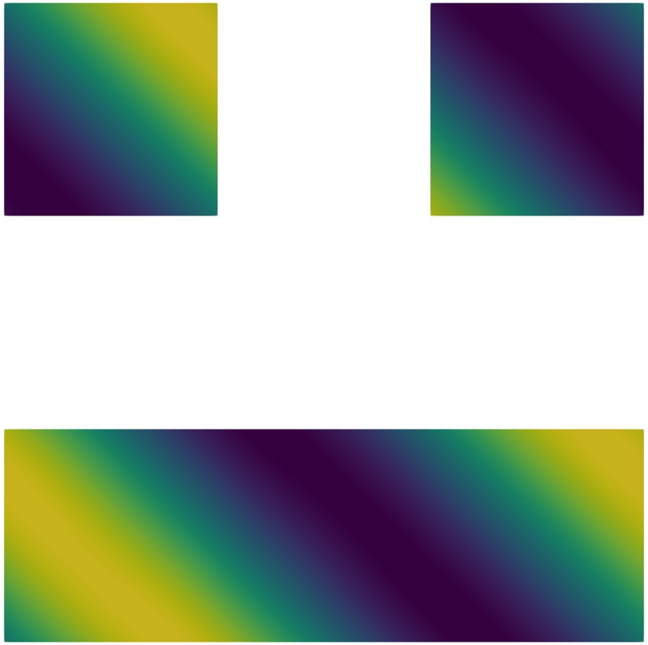
Multiphysics Coupling with Trixi.jl

Simon Candelaresi, Michael Schlottke-Lakemper

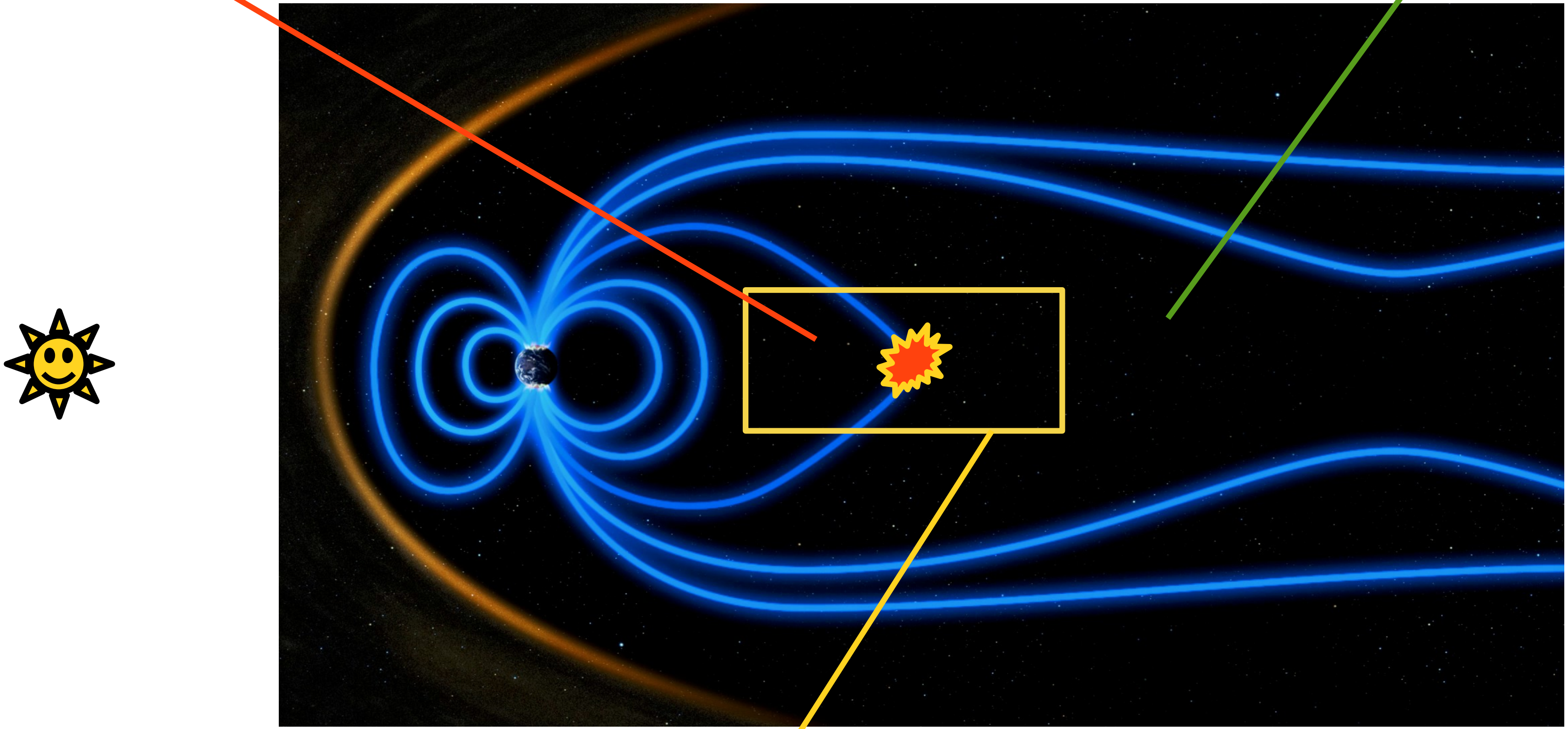


Universität
Augsburg
University

Modeling

kinetic model
(expensive)

magnetohydrodynamics (MHD)
(cheap)



interface coupling

(ESA)

Coupling via Converter Functions

Two system with any number of shared variables, including 0:

$$\partial_t u_a + \nabla \cdot f_a(u_a) = 0 \qquad \partial_t u_b + \nabla \cdot f_b(u_b) = 0$$

$$c_{ba}(x, u_b) \qquad c_{ab}(x, u_a)$$

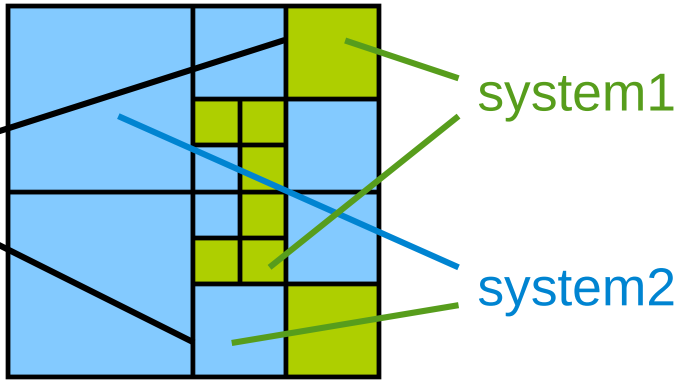
2.5d MHD $\rho, v_1, v_2, v_3, p, B_1, B_2, B_3, \Psi$	ρ, v_1, v_2, p 2d Euler

```
coupling_function12 = (x, u, equations_other, equations_own)
                     -> SVector(u[1], u[2], u[3], 0.0, u[4], 0.0, 0.0, 0.0, 0.0)
coupling_function21 = (x, u, equations_other, equations_own) -> SVector(u[1], u[2], u[3], u[5])
```

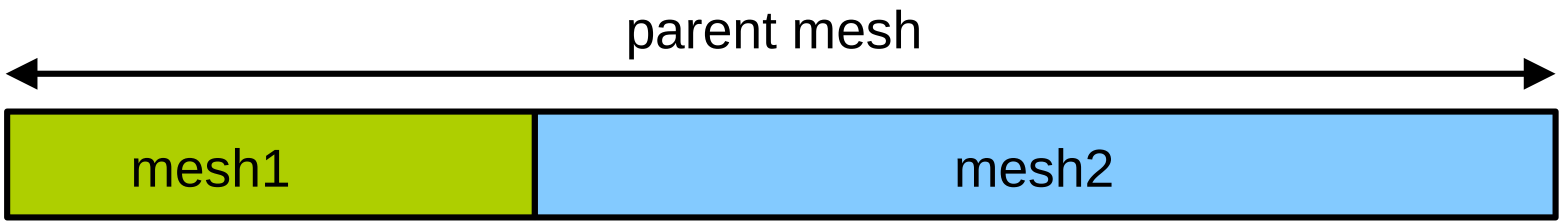
- User can define converter functions.
- Any pair of systems can be coupled.

p4est Meshes

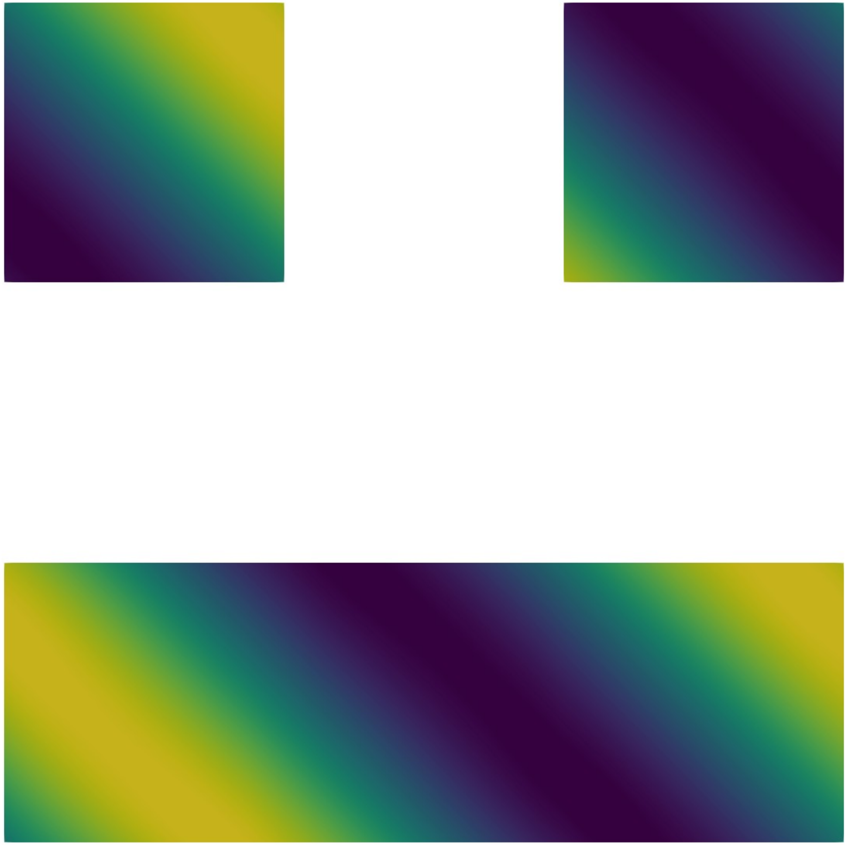
We can define the b.c. for each cell boundary.



mesh views:

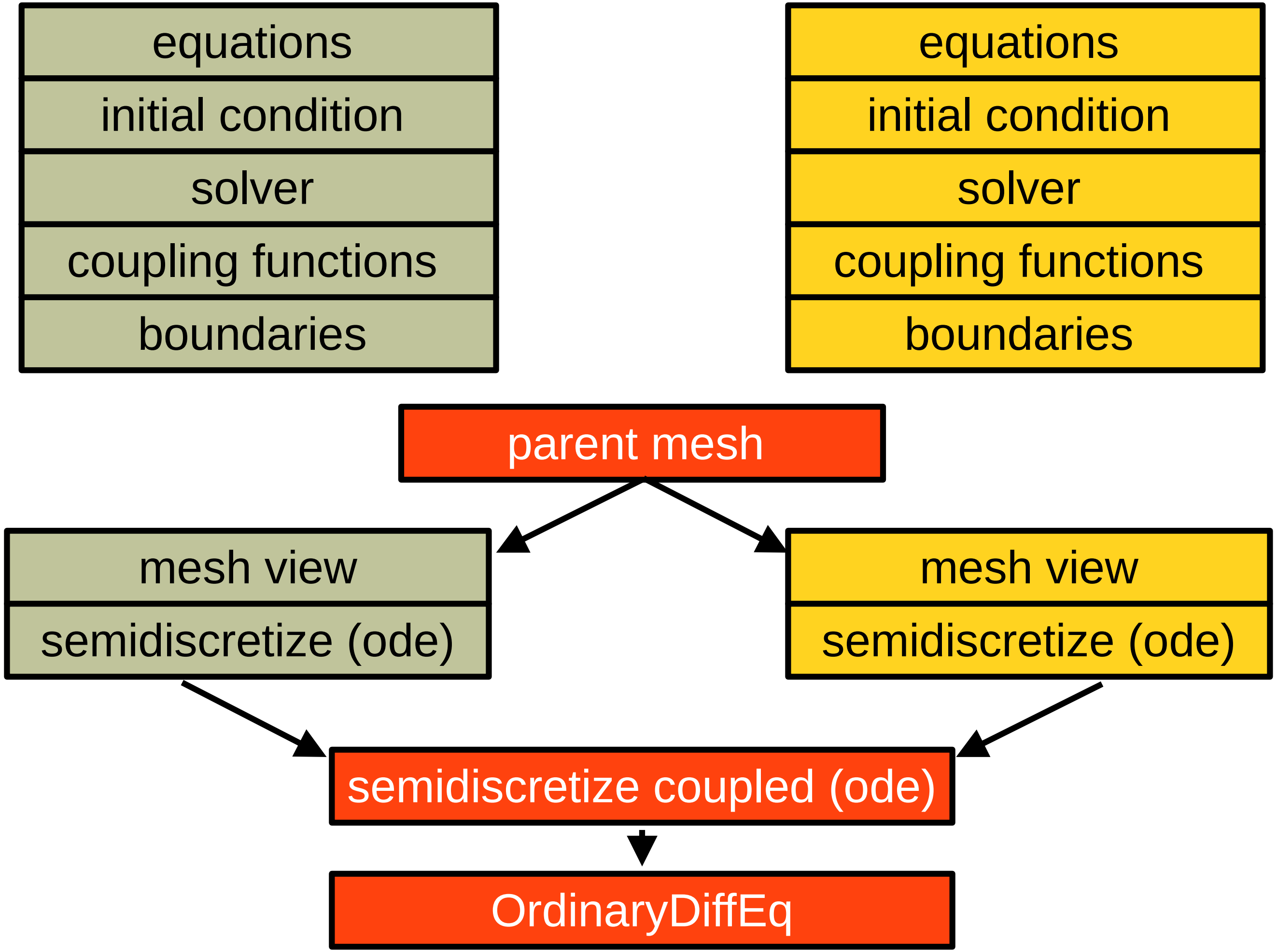


p4est Mesh Views

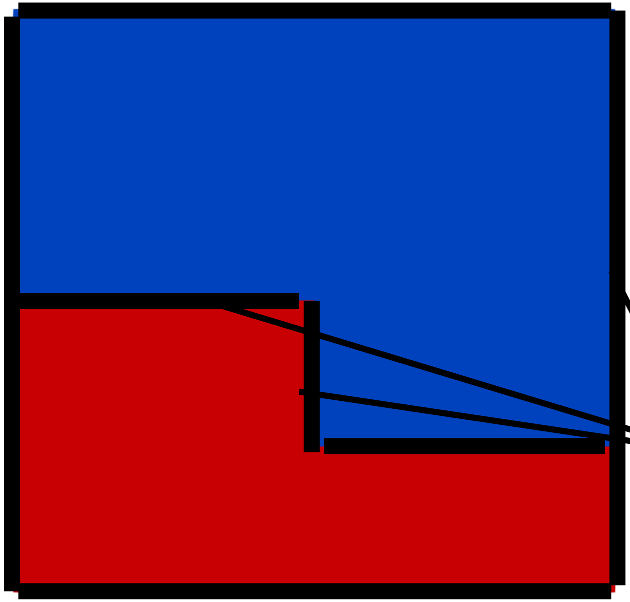


- p4est mesh views in Trixi.jl
- arbitrary topology
- views on smallest (cell) scale

Work Flow Coupling

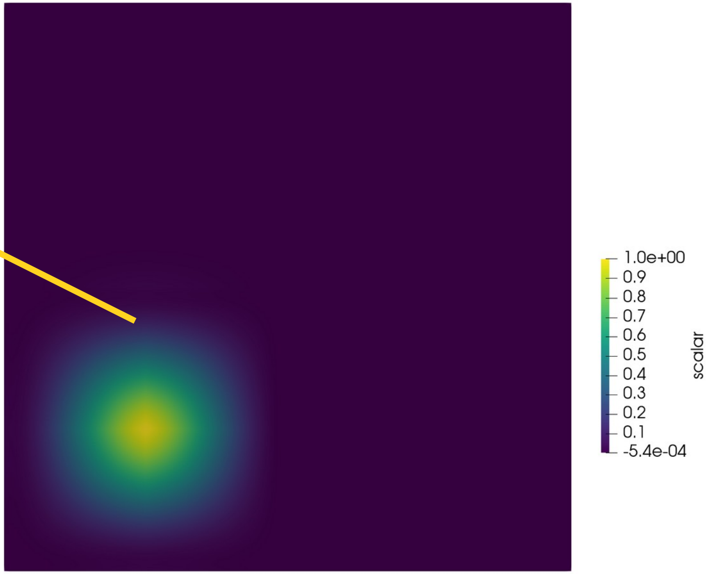


Linear Advection



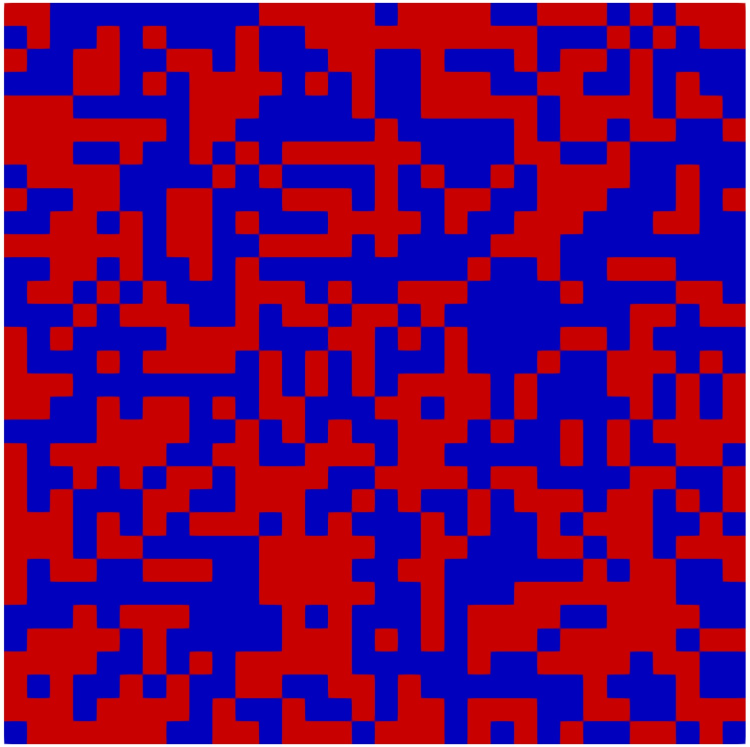
scalar advection

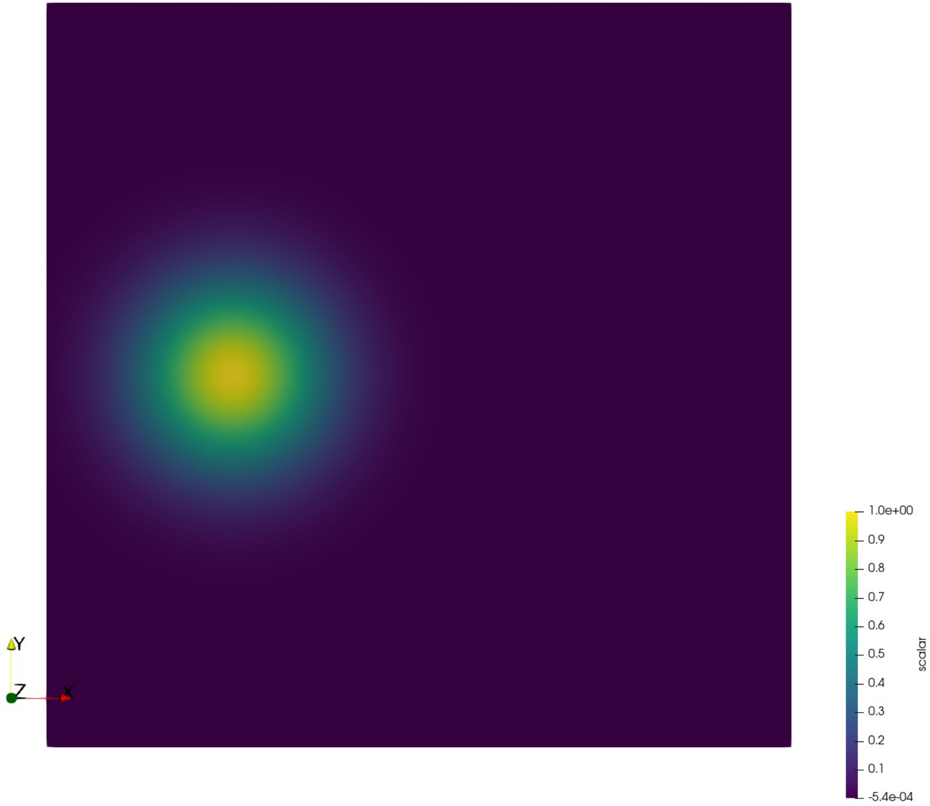
coupled interfaces



$$\partial_t u + a_x \partial_x u + a_y \partial_y u = 0 \qquad (a_x, a_y) = (0.05, 0.02)$$

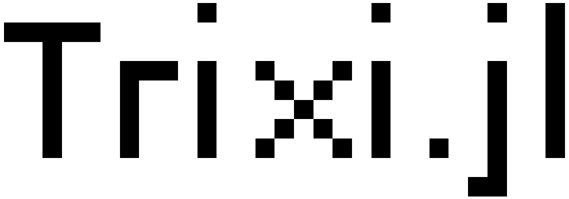
Linear Advection



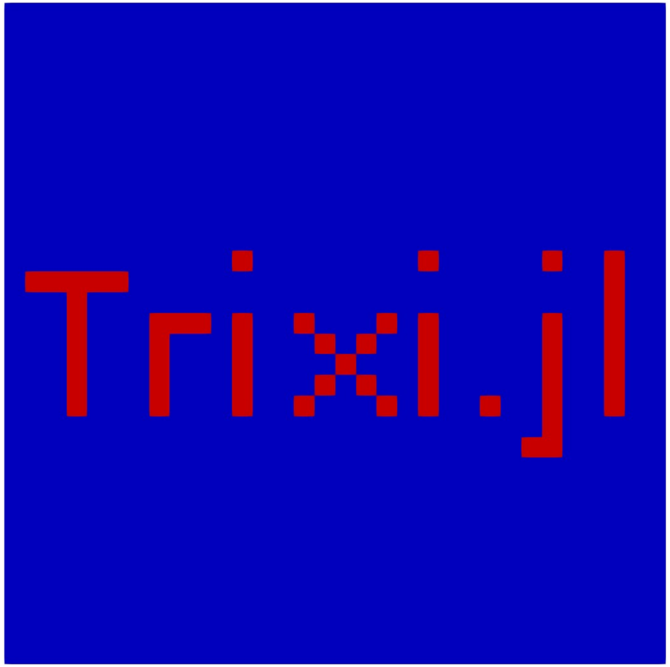


Linear Advection

BMP file:

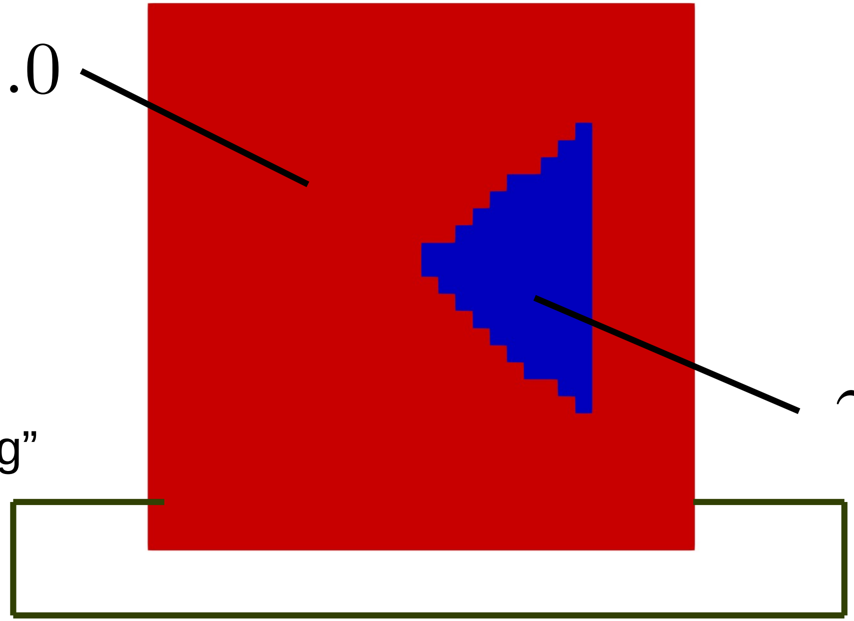


p4est domains:



Isothermal-Polytropic

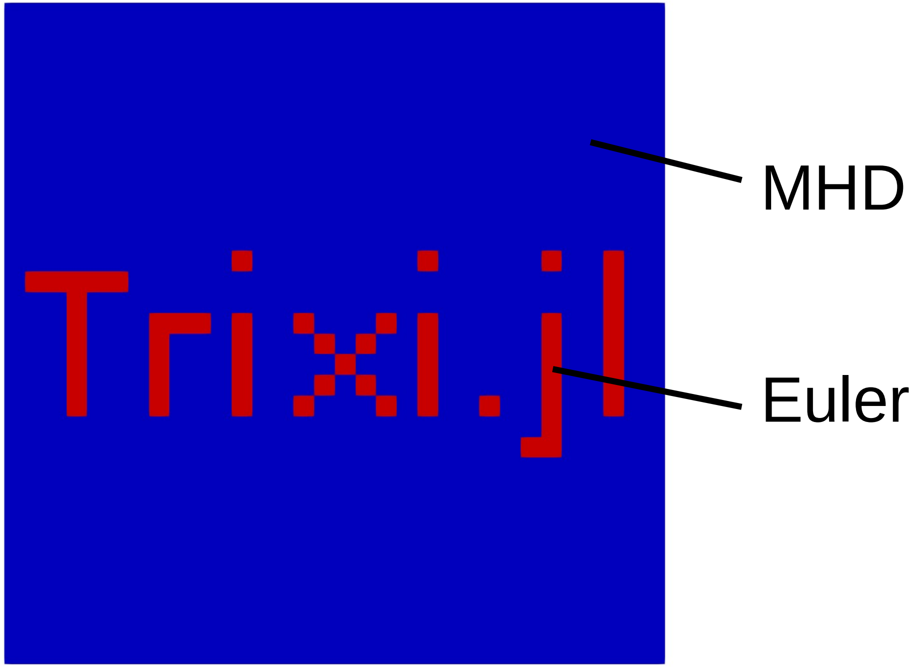
$$\partial t \begin{pmatrix} \rho \\ \rho v_1 \\ \rho v_2 \end{pmatrix} + \partial x \begin{pmatrix} \rho v_1 \\ \rho v_1^2 + \rho^\gamma \\ \rho v_1 v_2 \end{pmatrix} + \partial y \begin{pmatrix} \rho v_2 \\ \rho v_1 v_2 \\ \rho v_2^2 + \rho^\gamma \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

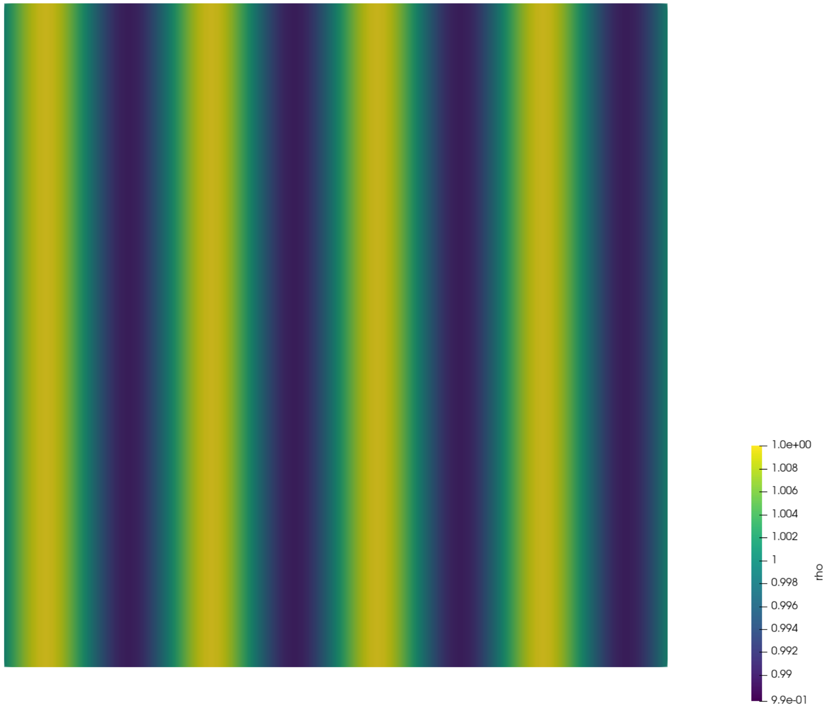


coupling functions:

$$c_{ab} = u^b$$
$$c_{ba} = u^a$$

MHD-Euler

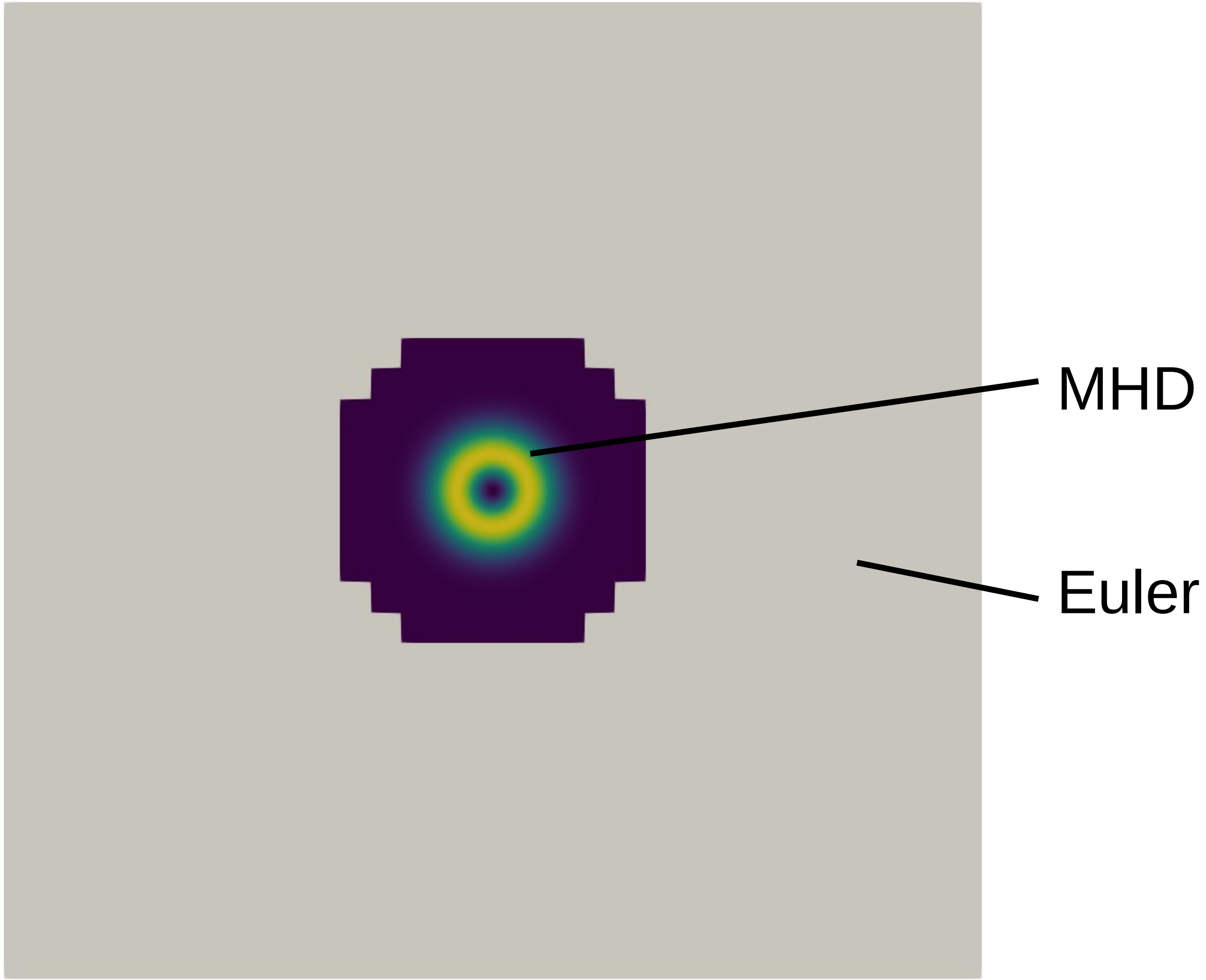

Trixi.jl
MHD
Euler


1.0e+00
1.008
1.006
1.004
1.002
1
0.998
0.996
0.994
0.992
0.99
9.9e-01
rho

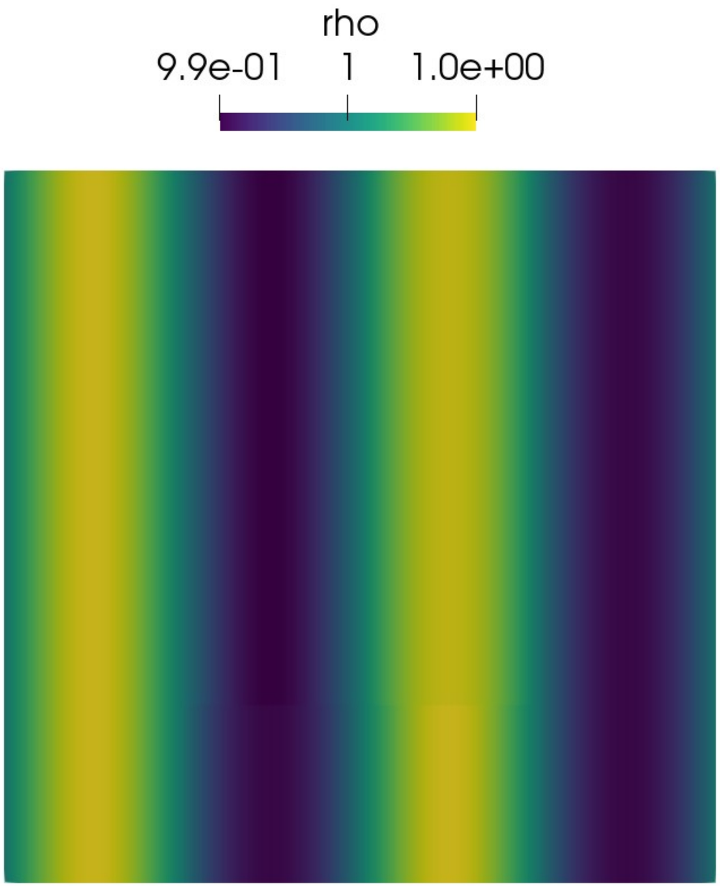
Adaptive Model Selection

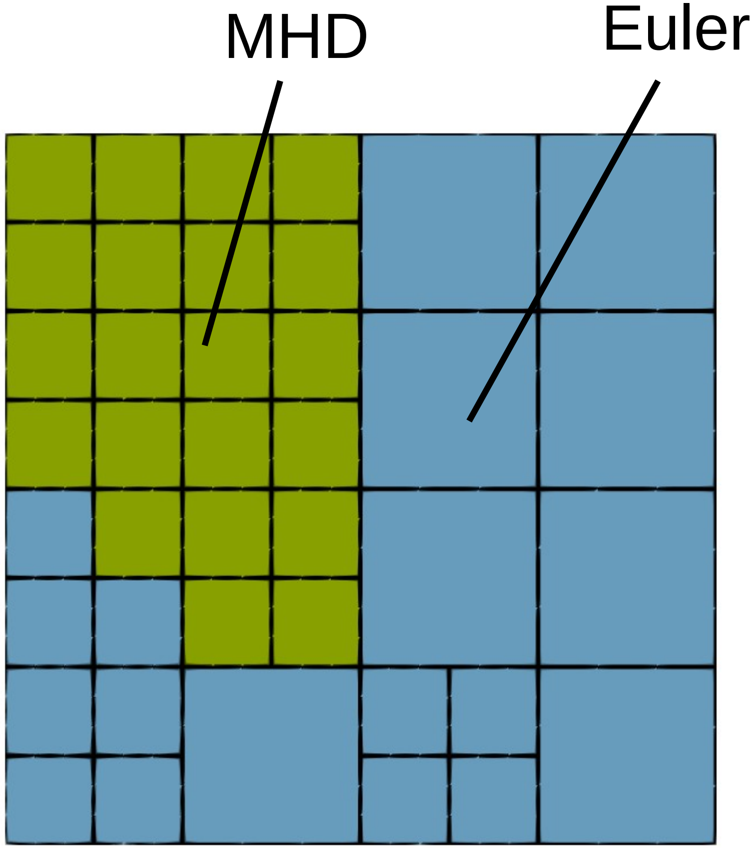
1. Generate the new grid (mesh views).
2. Write new u-solution vectors
3. Generate new ODE for OrdinaryDiffEq (integrator).
4. Reinitialize ODE integrator with new problem and new solution vector.

MHD-Euler AMS



Hanging Nodes





Conclusion

- Free domain definitions.
- Import meshes from an image.
- Coupled p4est simulations.
- Non-simply connected meshes in p4est.
- Adaptive model selection with arbitrary criteria.
- Static AMR allows atomistic refinement

simon.candelaresi@uni-a.de, simon.candelaresi@gmail.com