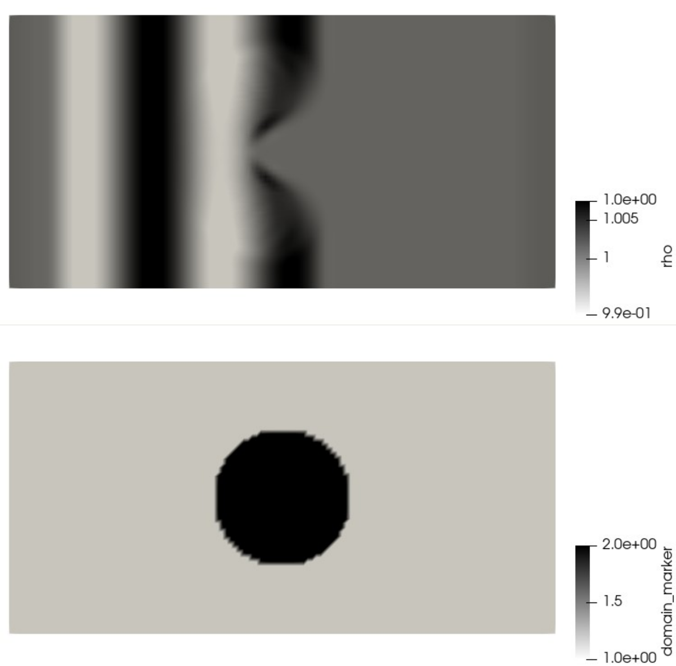
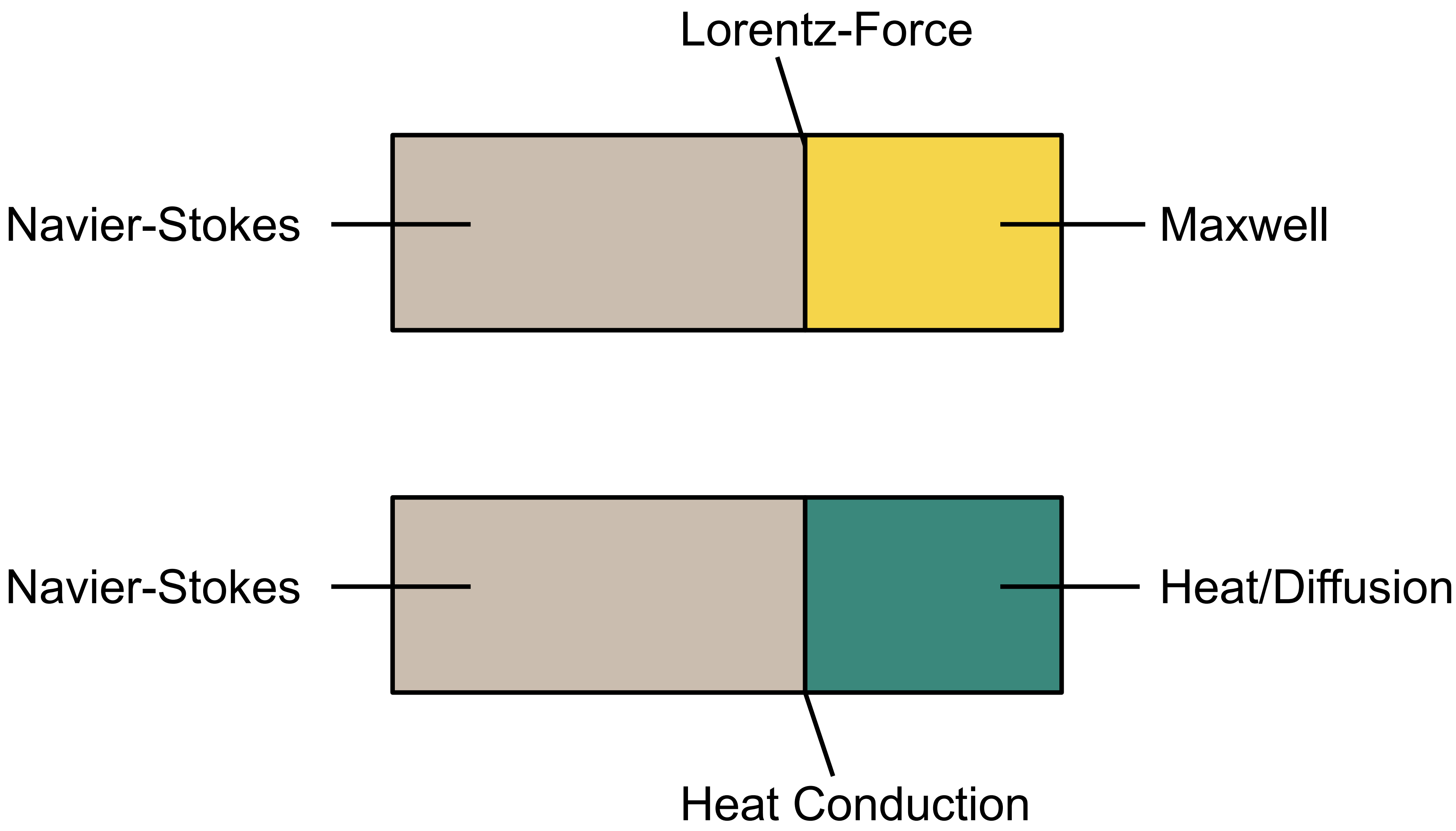
SNuBIC C2
On the Coupling of Multiphysics Systems

Simon Candelaresi

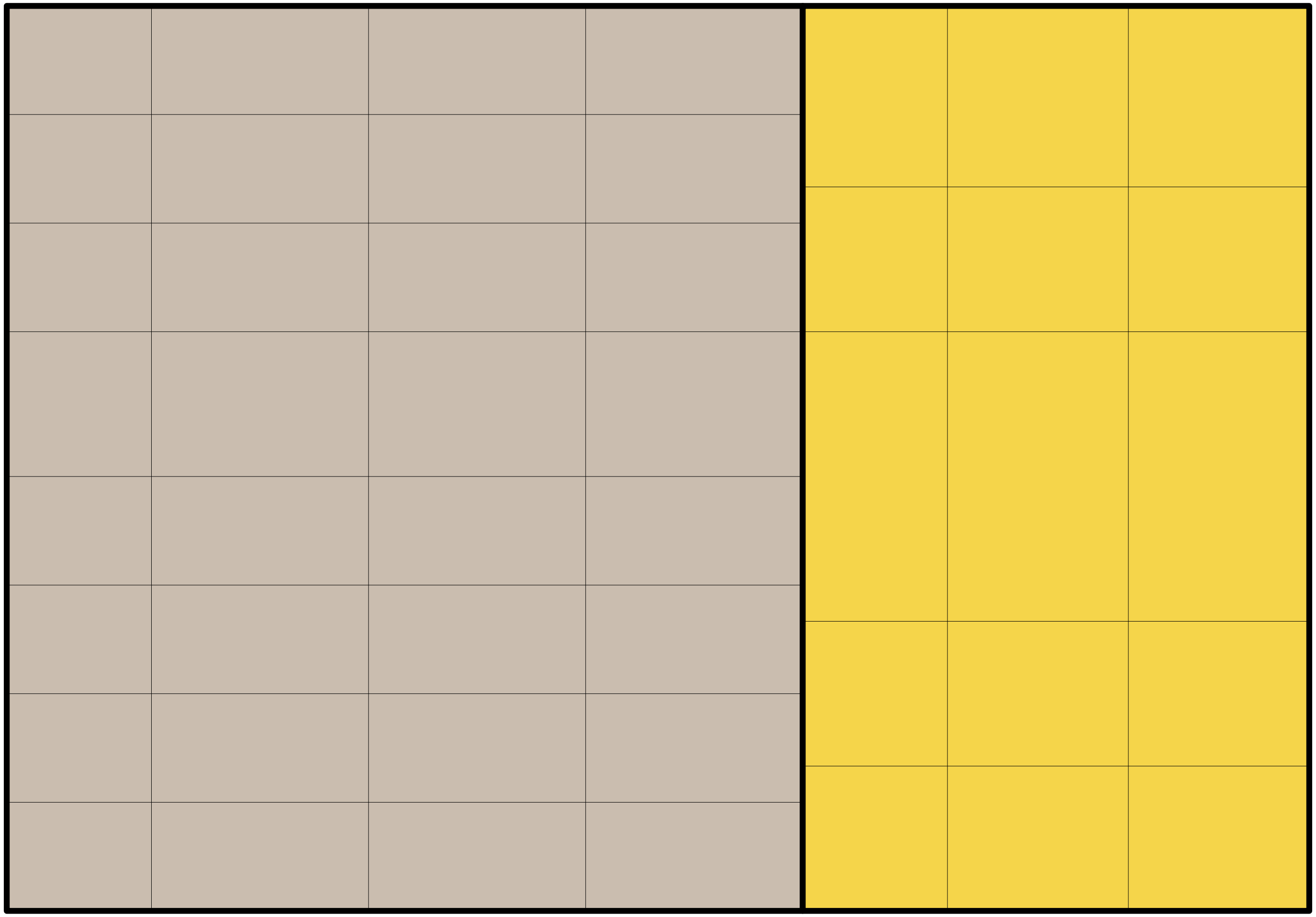
High-Performance Computing Center Stuttgart



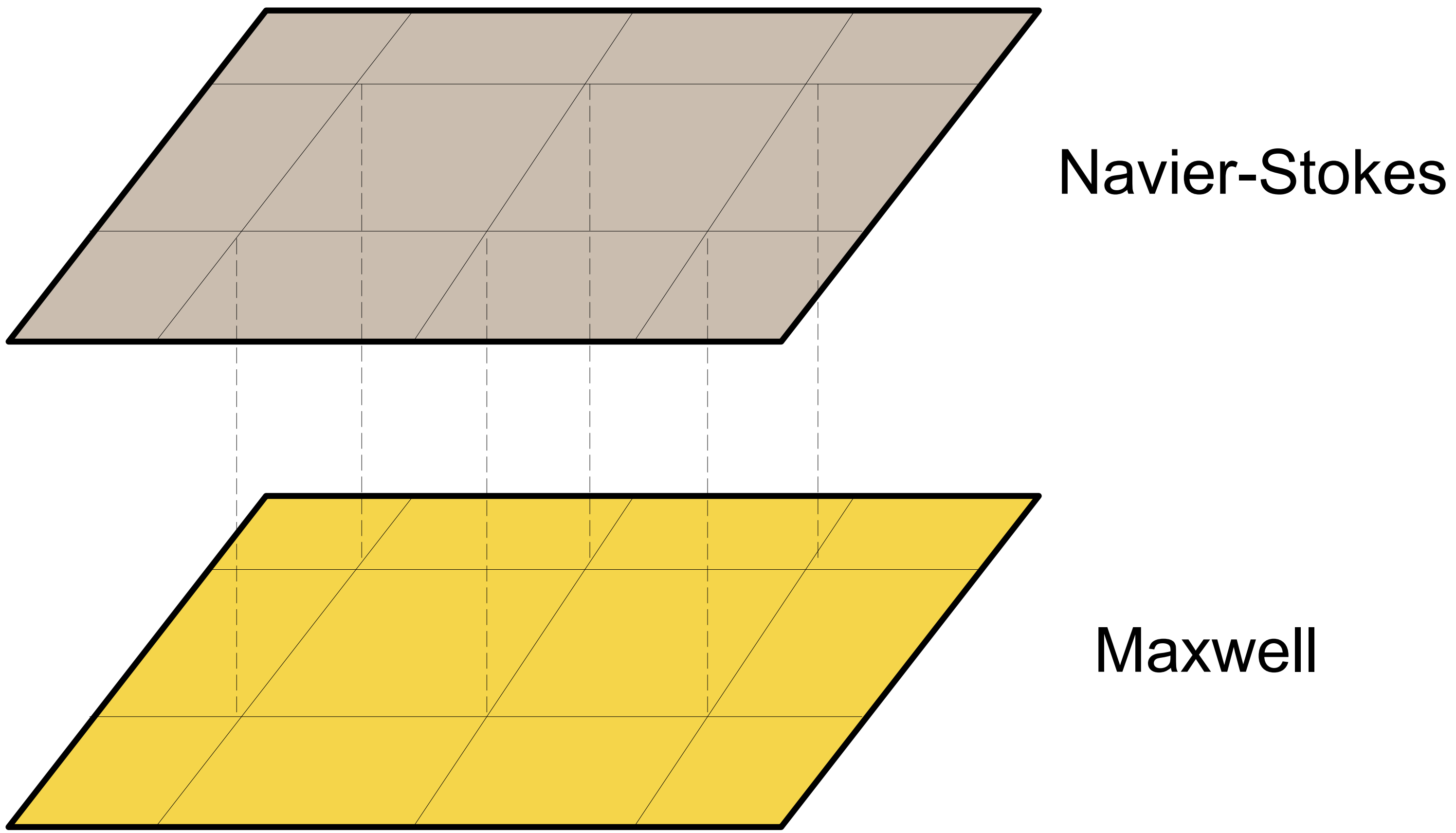
Interface Coupling



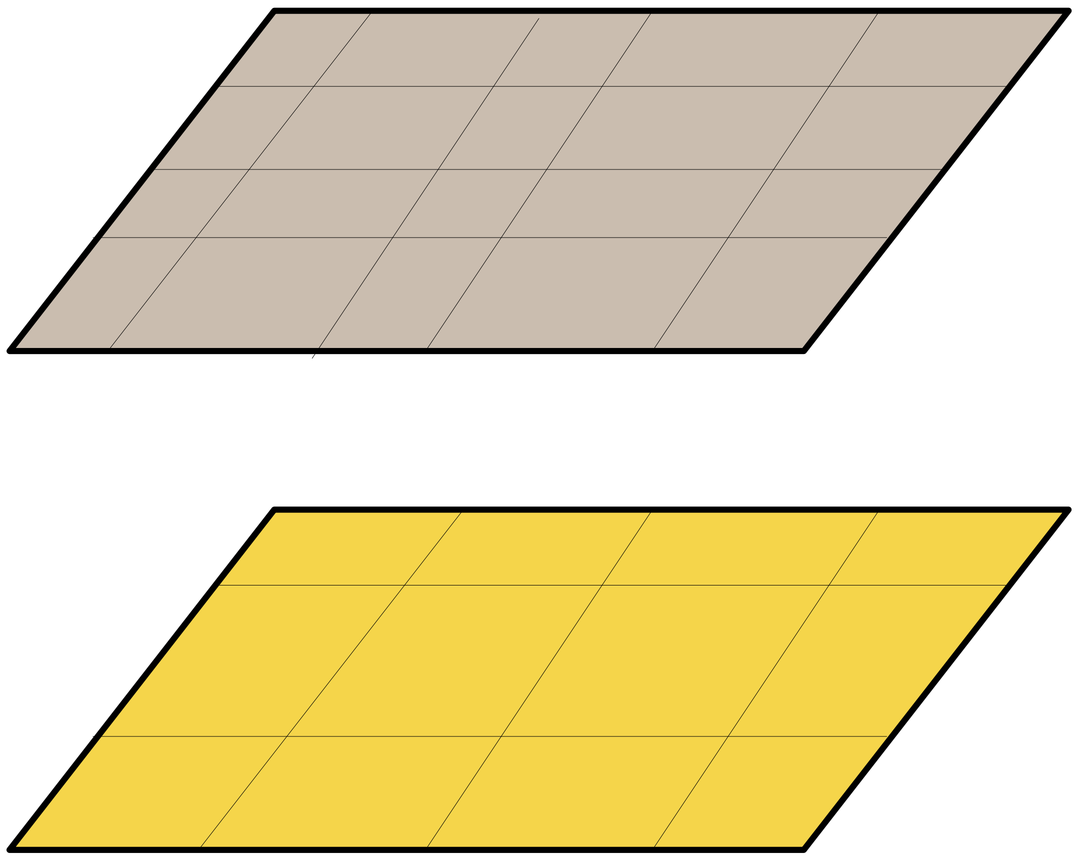
Non-Matching or Adaptive Grids



Bulk (Volume) Coupling



Non-Matching or Adaptive Grids

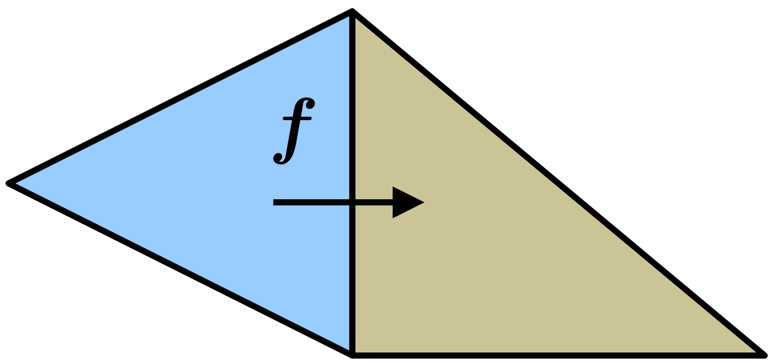


Coupling Via Fluxes

General hyperbolic problem:

$$\partial_t \boldsymbol{u}_i + \boldsymbol{\nabla} \cdot \boldsymbol{f}_i(\boldsymbol{u}_i) = 0, \quad \boldsymbol{u}_i(t=0) = \boldsymbol{u}_i^0, \quad \boldsymbol{u}_i(\partial V_i) = \boldsymbol{u}_i^{\mathrm{BV}}$$

fluxes within the domain (between computational elements)



Example: Compressible Euler equations in 2d:

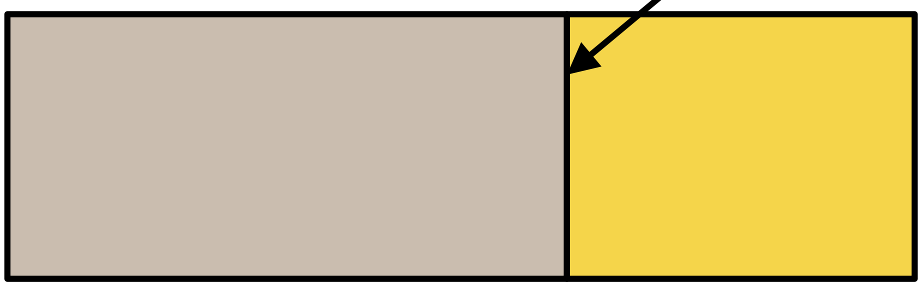
$$\partial_t \begin{pmatrix} \rho \\ \rho v_1 \\ \rho v_2 \\ e \end{pmatrix} + \partial_x \begin{pmatrix} \rho v_1 \\ \rho v_1^2 + p \\ \rho v_1 v_2 \\ (\rho e + p) v_1 \end{pmatrix} + \partial_y \begin{pmatrix} \rho v_2 \\ \rho v_1 v_2 \\ \rho v_2^2 + p \\ (\rho e + p) v_2 \end{pmatrix} = 0$$

Coupling Via Fluxes

$$\partial_t \boldsymbol{u}_a + \boldsymbol{\nabla} \cdot \left(\boldsymbol{f}_a(\boldsymbol{u}_a) + \boldsymbol{f}_{ba}(\boldsymbol{u}_a, \boldsymbol{u}_b) \right) = 0$$

$$\partial_t \boldsymbol{u}_b + \boldsymbol{\nabla} \cdot \left(\boldsymbol{f}_b(\boldsymbol{u}_b) + \boldsymbol{f}_{ab}(\boldsymbol{u}_a, \boldsymbol{u}_b) \right) = 0$$

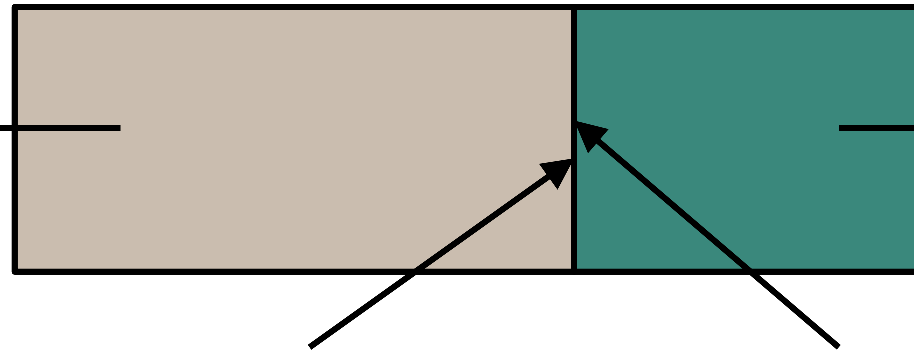
coupling fluxes



Currently under development in Trixi for the Euler and heat flux systems.

boundary: $\partial_t(\rho e) = \hat{n} \cdot (\nabla(\rho e)) = \hat{n} \cdot \nabla f_{ba}(u_b)$

Coupling via Dirichlet Boundary



- Easy to implement.
- Works for systems that have different variables.

Coupling via Converter Functions

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

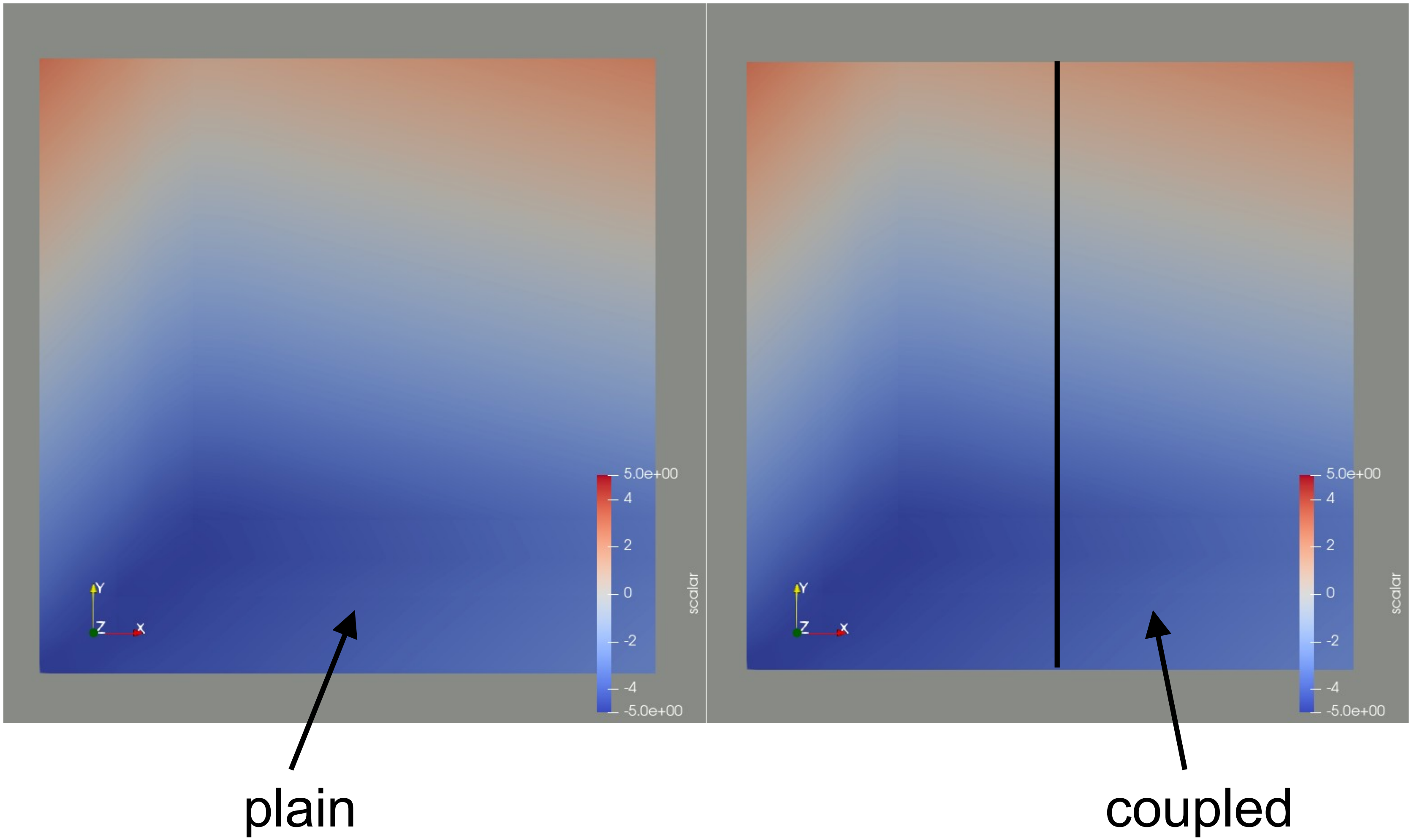
$$\partial_t u_a = f_a(u_a) \qquad \partial_t u_b = f_b(u_b)$$

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

```
function coupling_converter(equations::AbstractEquations)
    return (x, u) -> u
end
```

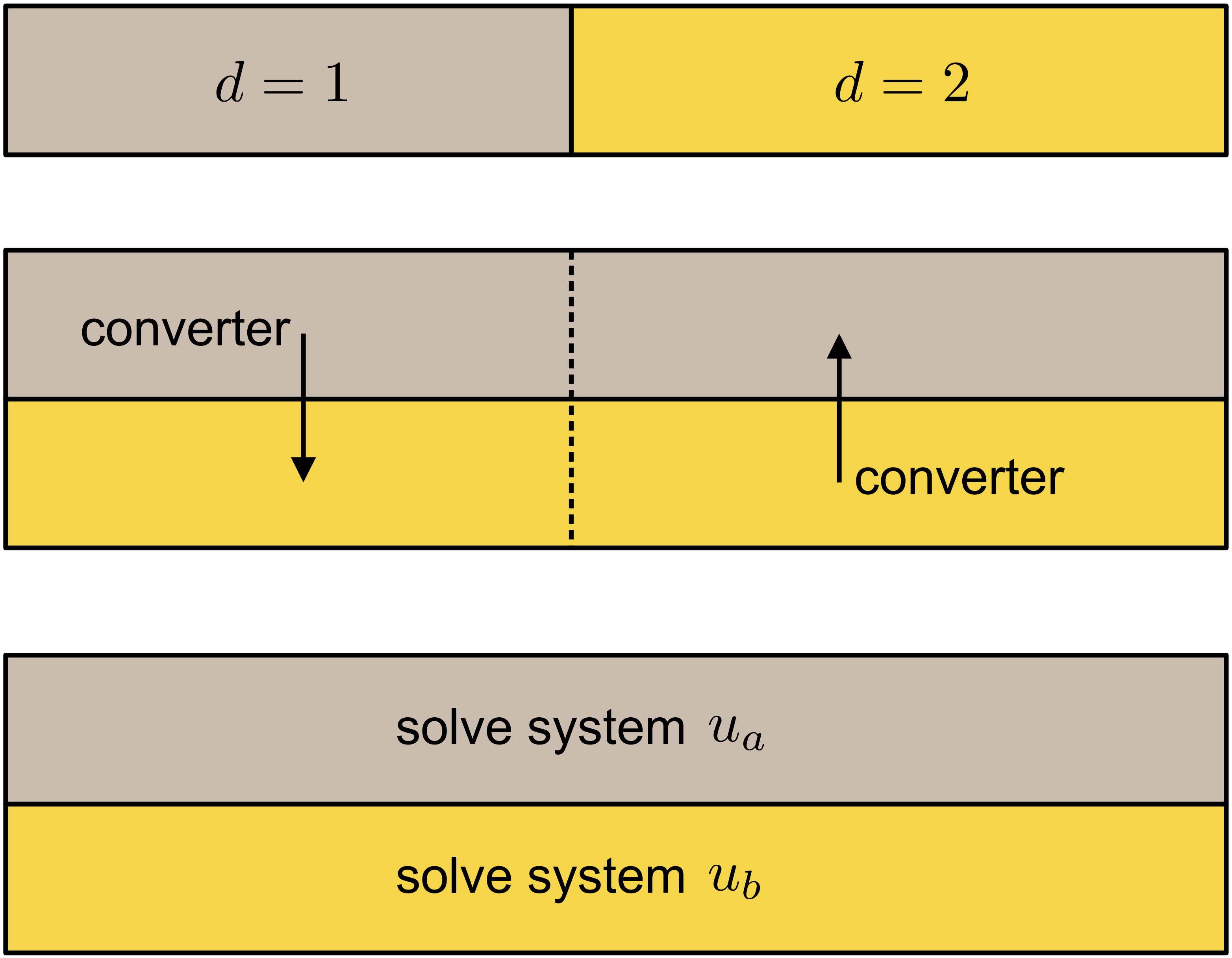
- User can define converter functions.
- Any two systems can be coupled.
- Should the functions depend on time?

Coupling via Converter Functions

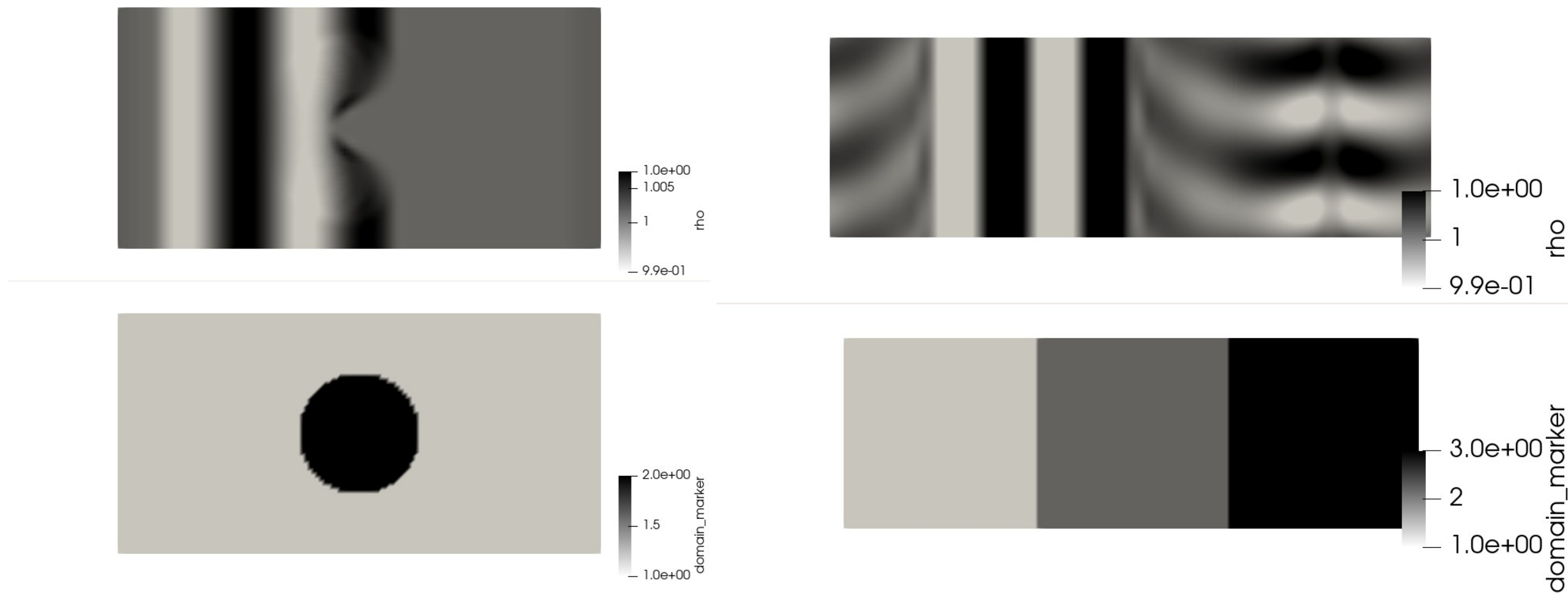


Same order of convergence.

Fixed Domain Markers

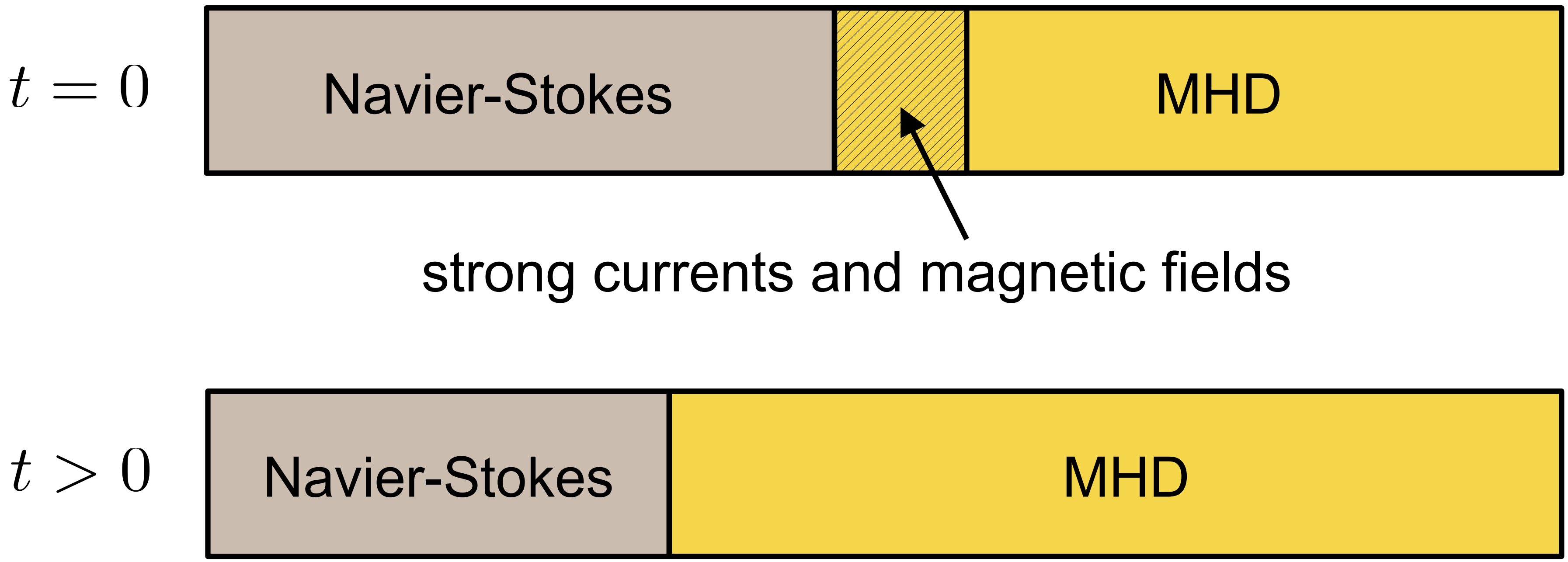


Fixed Domain Markers



- Can use self-defined converter functions.
- Flexible geometry.
- Somewhat wasteful usage of resources.

Domain Shifting



- Work with progress.
- What are the criteria?

Conclusion

- Flexible coupling through converter functions.
- Free domain definitions.
- Dynamic domains. What are the criteria?