arXiv:2303.04679

Flow reconstruction by multiresolution optimization of a discrete loss with automatic differentiation

Petr Karnakov

with Sergey Litvinov, Petros Koumoutsakos

Harvard University

Harvard John A. Paulson School of Engineering and Applied Sciences

Inverse and ill-posed problems

Inverse and ill-posed problems for PDEs

- incomplete initial and boundary conditions
- sparse and noisy data
- unknown coefficients
- …

Existing methods (partial list)

- PINN (Physics-Informed Neural Networks)
- differentiable solvers, adjoints
- …

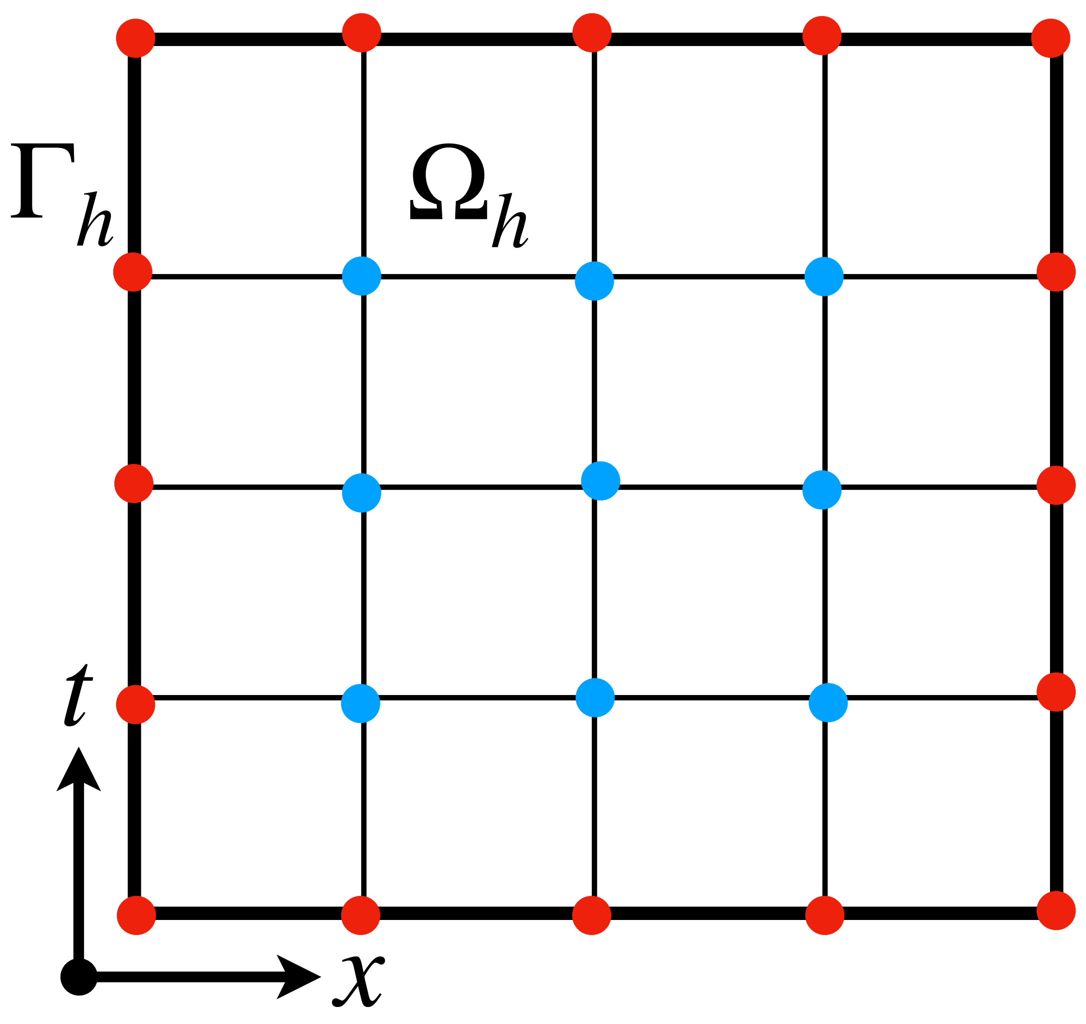
THIS WORK (Optimizing the DIscrete Loss)

1. Equations discretized on a grid - formulated as a loss function
   - faster and more accurate than PINN
   - simpler and more versatile than adjoints
2. Multigrid technique that further accelerates convergence

Solving equations as optimization: ODIL & PINN

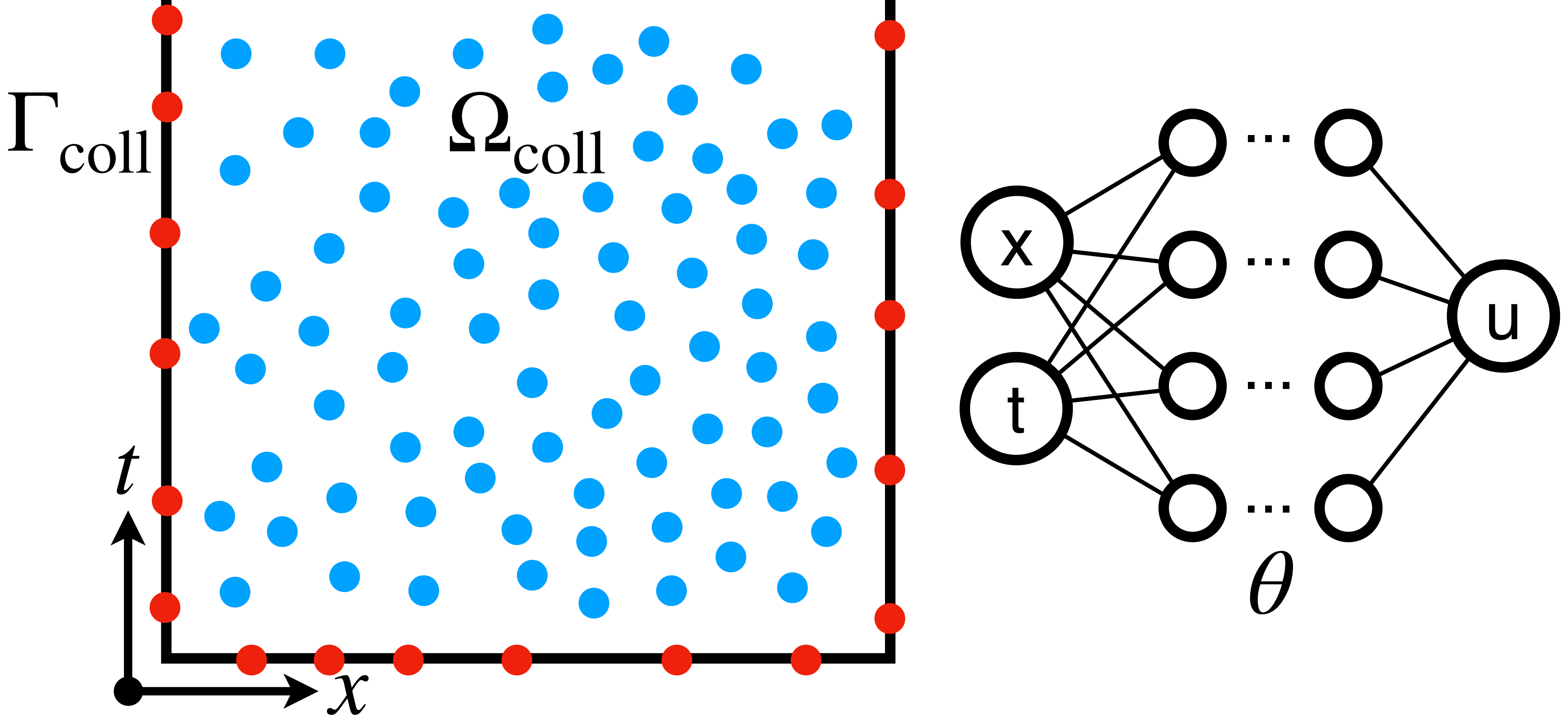
- Wave equation $\left.\dfrac{\partial^2 u}{\partial t^2} - \dfrac{\partial^2 u}{\partial x^2} = 0\right|_{\Omega}$ with conditions $\left. u = g \right|_{\Gamma}$

- **ODIL (Optimizing a DIscrete Loss)**
  solution is discrete field u_i^n



$$L(u) = \sum_{(i,n)\in\Omega_h} \left(\frac{u_i^{n-1} - 2u_i^n + u_i^{n+1}}{\Delta t^2} - \frac{u_{i-1}^n - 2u_i^n + u_{i+1}^n}{\Delta x^2} \right)^2 + \sum_{(i,n)\in\Gamma_h} \left(u_i^n - g_i^n \right)^2 \to \min_u$$

- **PINN (Physics-Informed Neural Network)**
  solution is neural network $u_\theta(x, t)$

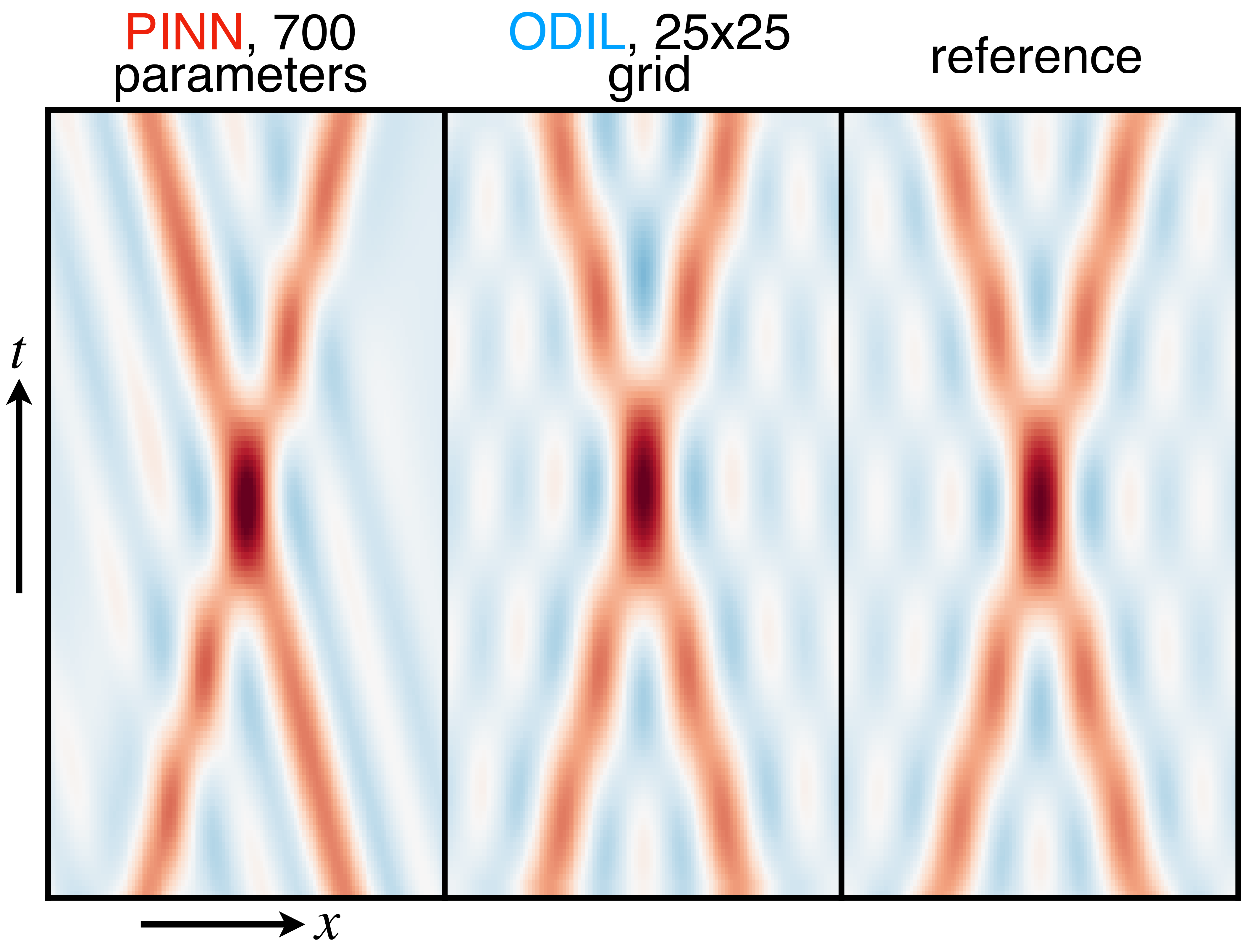


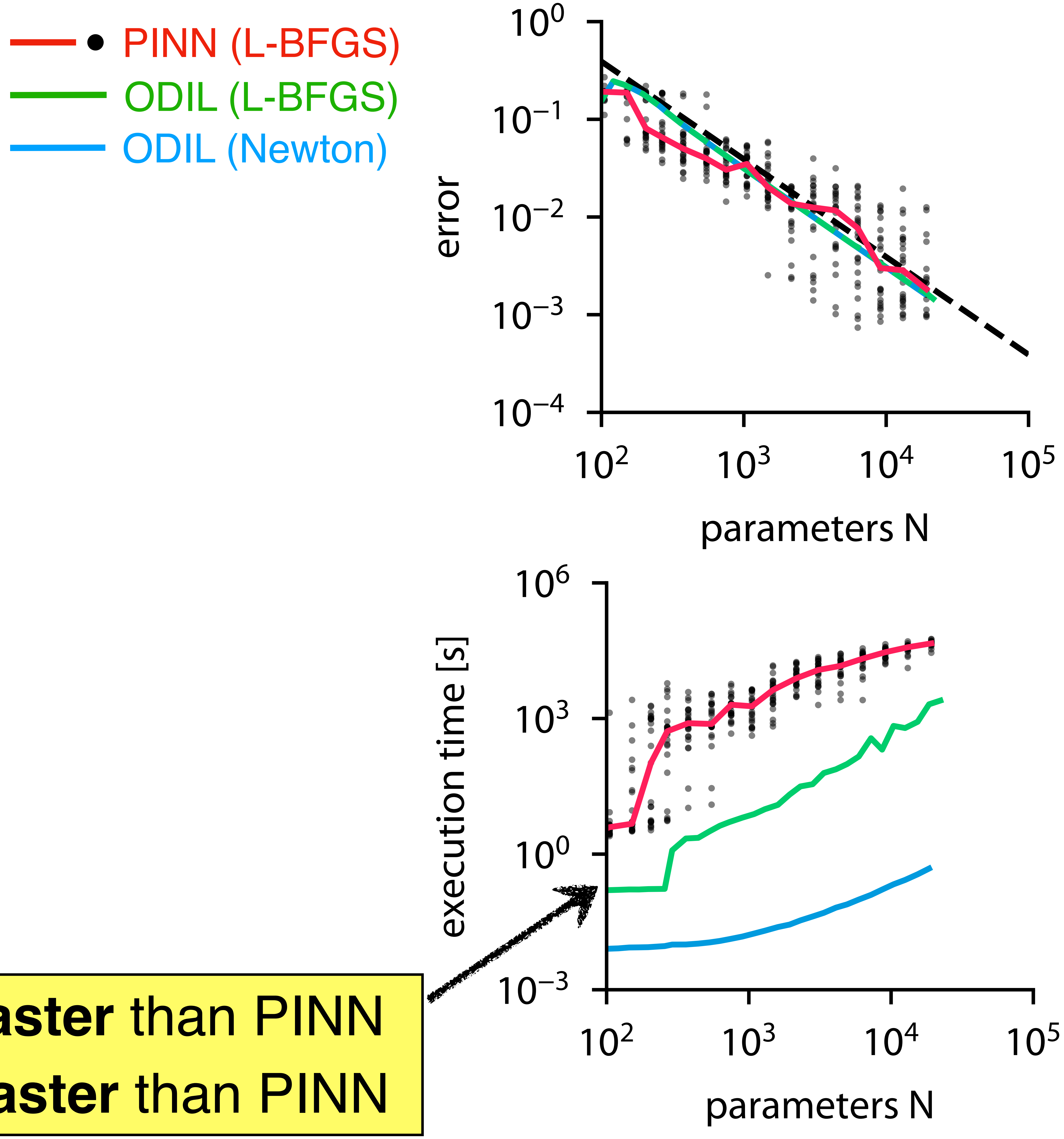
$$L(\theta) = \sum_{(x,t)\in\Omega_{\text{coll}}} \left(\frac{\partial^2 u_\theta}{\partial t^2} - \frac{\partial^2 u_\theta}{\partial x^2} \right)^2 + \sum_{(x,t)\in\Gamma_{\text{coll}}} \left(u_\theta(x,t) - g(x,t) \right)^2 \to \min_\theta$$

ODIL: Faster alternative to PINN (in 1D)

- Wave equation
  initial-value problem

$$\frac{\partial^2 u}{\partial t^2} - \frac{\partial^2 u}{\partial x^2} = 0$$





ODIL (L-BFGS) **x100 faster** than PINN
ODIL (Newton) **x100'000 faster** than PINN

FOR
MORE COMPARISONS AND DETAILS

arXiv:2205.04611

arXiv > math > arXiv:2205.04611

Search...

Help | Advanced S

Mathematics > Numerical Analysis

[Submitted on 10 May 2022]

Optimizing a DIscrete Loss (ODIL) to solve forward and inverse problems for partial differential equations using machine learning tools

Petr Karnakov, Sergey Litvinov, Petros Koumoutsakos

mODIL: Multiresolution ODIL

- ODIL loss function on a grid of $N \times N$ points

$$L(u) = \sum_{(i,n)\in\Omega_h} \left(\frac{u_i^{n-1} - 2u_i^n + u_i^{n+1}}{\Delta t^2} - \frac{u_{i-1}^n - 2u_i^n + u_{i+1}^n}{\Delta x^2} \right)^2 + \sum_{(i,n)\in\Gamma_h} \left(u_i^n - g_i^n\right)^2$$

- Use a hierarchy of M levels, e.g. $N = N_1 = 17,\ N_2 = 9,\ \dots,\ N_M = 3$
- **Multigrid decomposition** of the unknown field

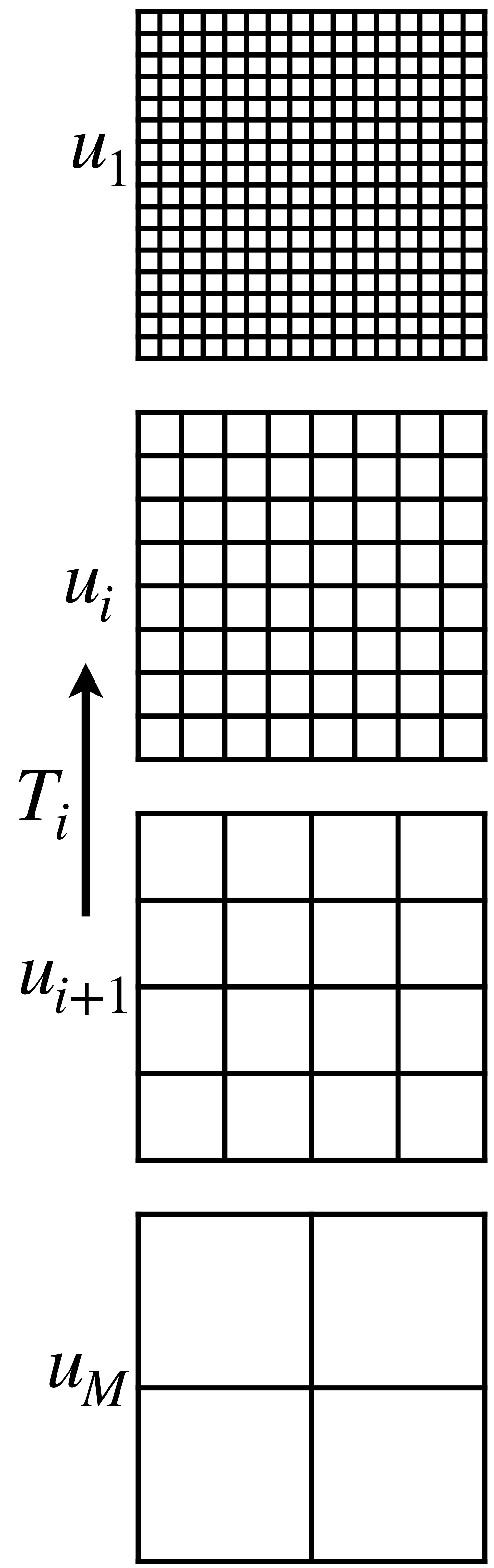
$$u = u_1 + T_1 u_2 + T_1 T_2 u_3 + \dots + T_1 \dots T_{M-1} u_M$$

with interpolation operators T_i

- Instead of $L(u)$, **mODIL** minimizes

$$L_M(u_1, \dots, u_M) := L(u_1 + T_1 u_2 + T_1 T_2 u_3 + \dots + T_1 \dots T_{M-1} u_M)$$

- Same as in ODIL: **keep using automatic differentiation**

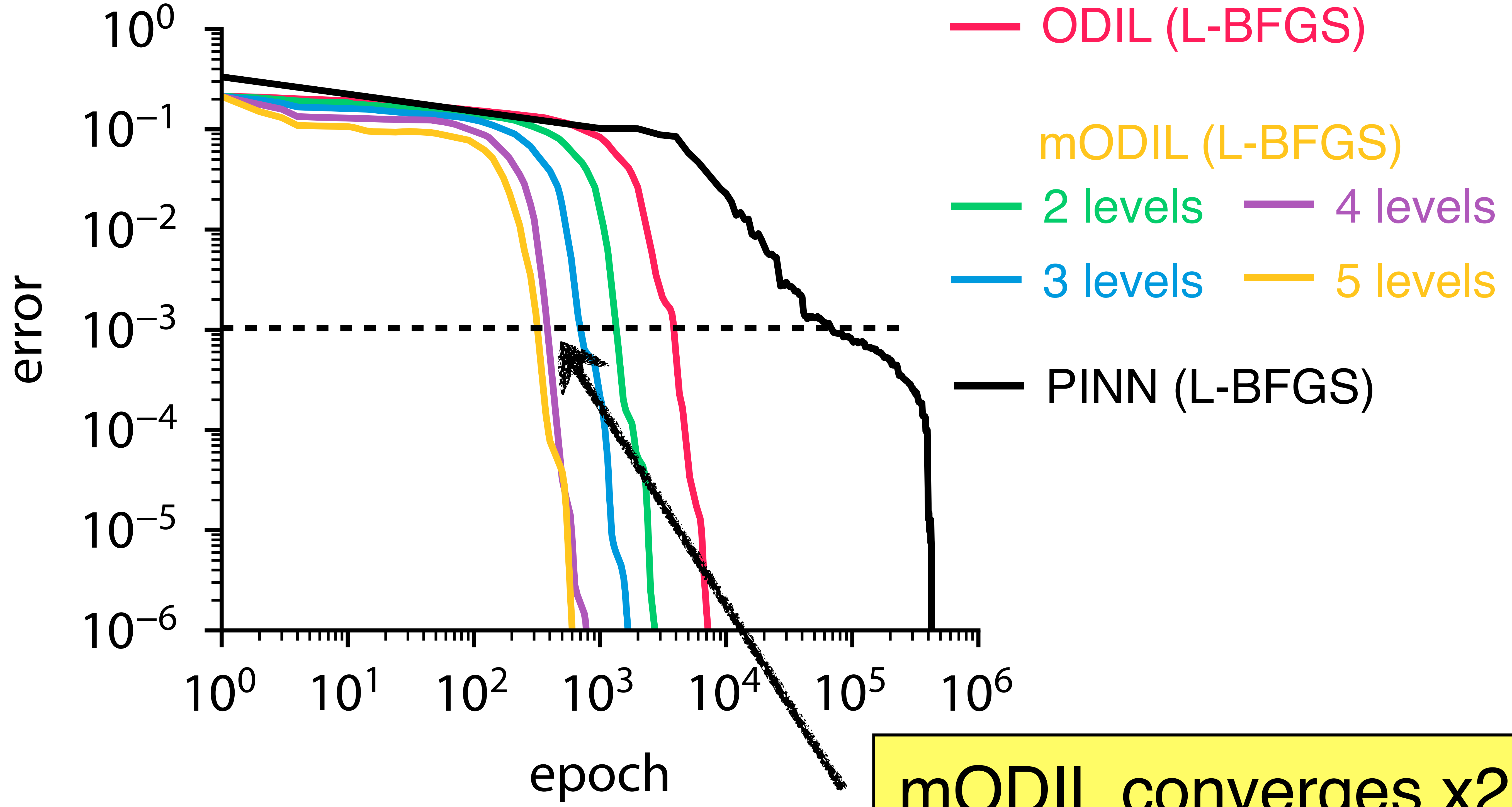


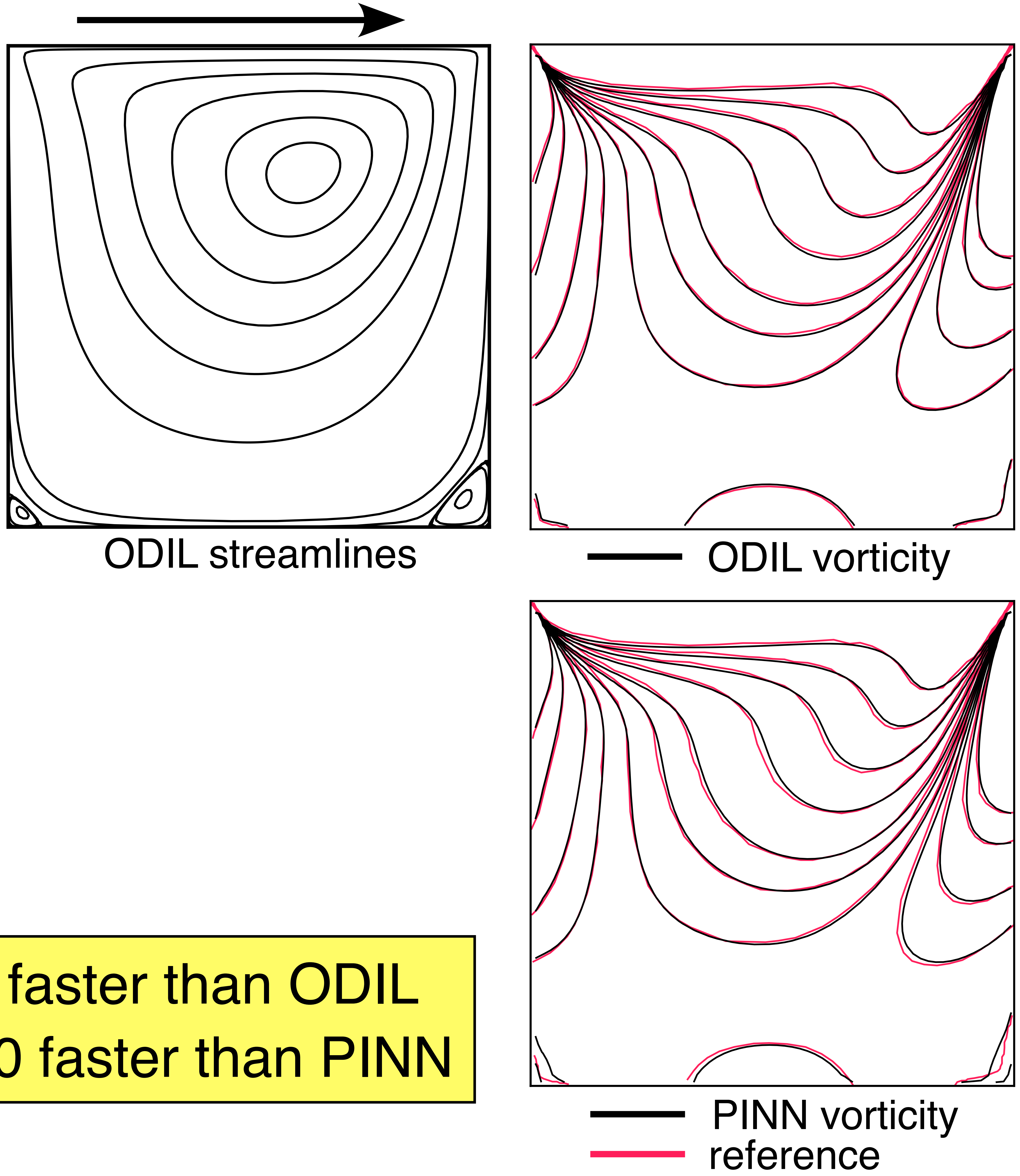
Lid-driven cavity Re=100

- 2D steady Navier-Stokes

$$\nabla \cdot \mathbf{u} = 0$$

$$(\mathbf{u} \cdot \nabla)\mathbf{u} = -\nabla p + \frac{1}{\mathrm{Re}} \nabla^2 \mathbf{u}$$





mODIL converges x20 faster than ODIL
mODIL converges x200 faster than PINN

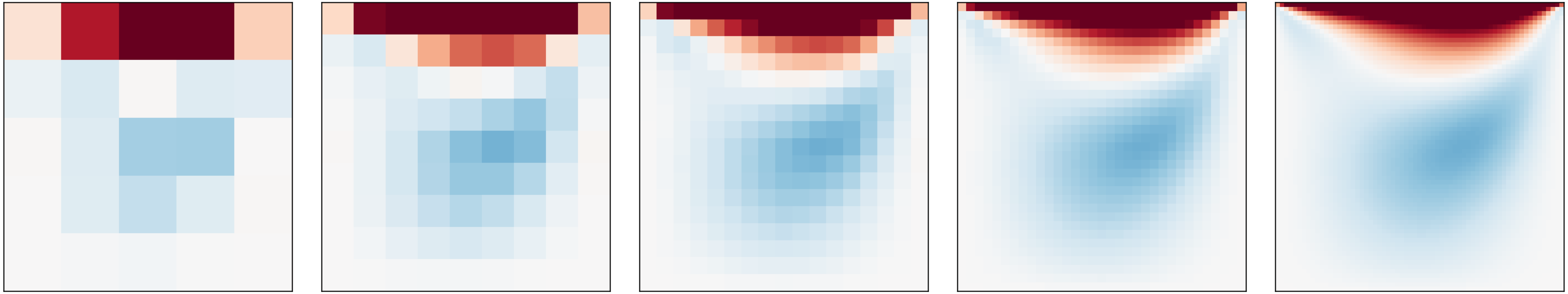
Ghia UK, Ghia KN, Shin CT. JCP. 1982

Lid-driven cavity Re=100

- mODIL: Multigrid decomposition of velocity u using 5 levels

$$u = u_1 + T_1(u_2 + T_2(u_3 + T_3(u_4 + T_4 u_5)))$$

with interpolation operators T_i



u_5 | $u_4 + T_4 u_5$ | $u_3 + T_3(u_4 + T_4 u_5)$ | $u_2 + T_2(u_3 + T_3(u_4 + T_4 u_5))$ | $u_1 + T_1(u_2 + T_2(u_3 + T_3(u_4 + T_4 u_5))) = u$

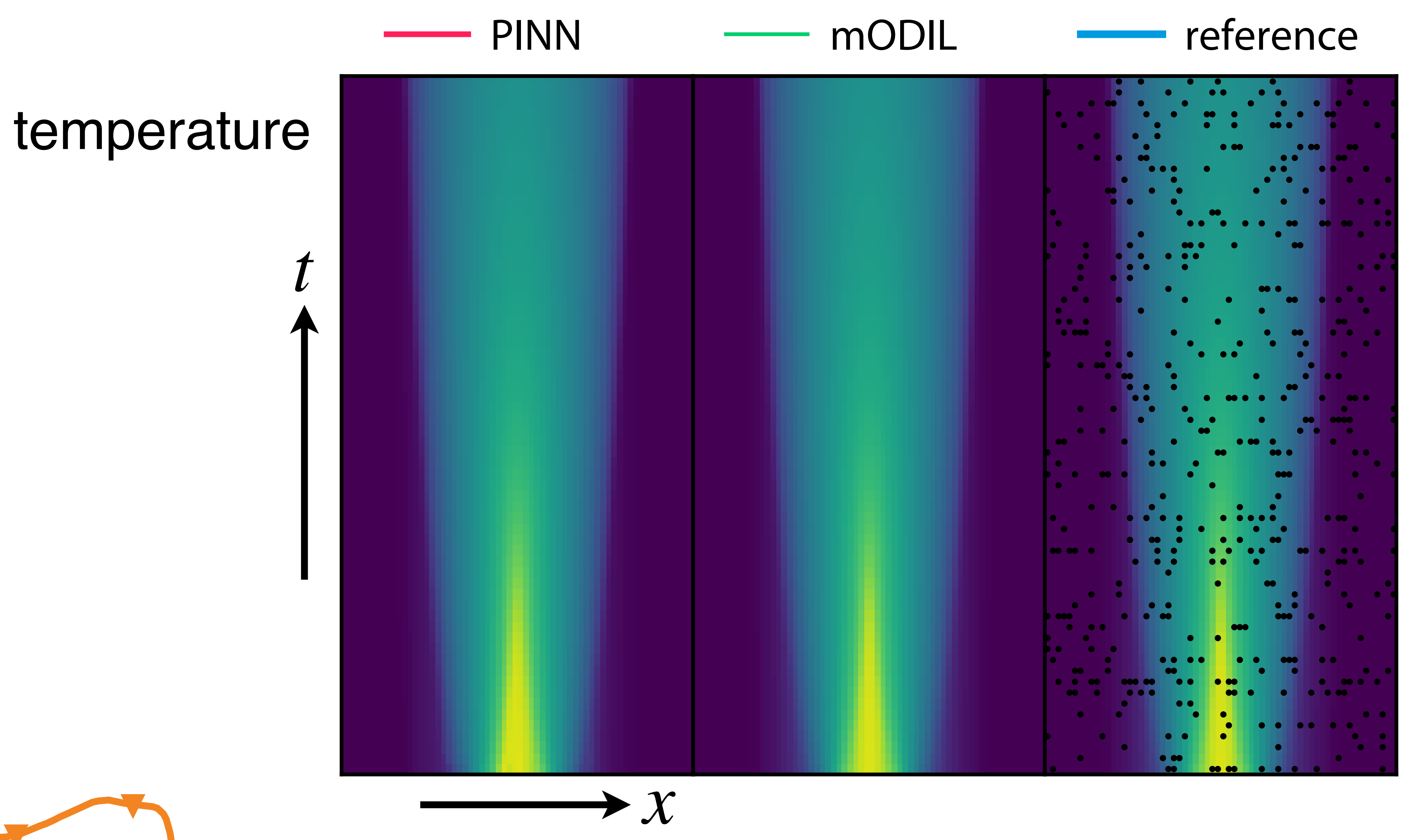
Conductivity from temperature

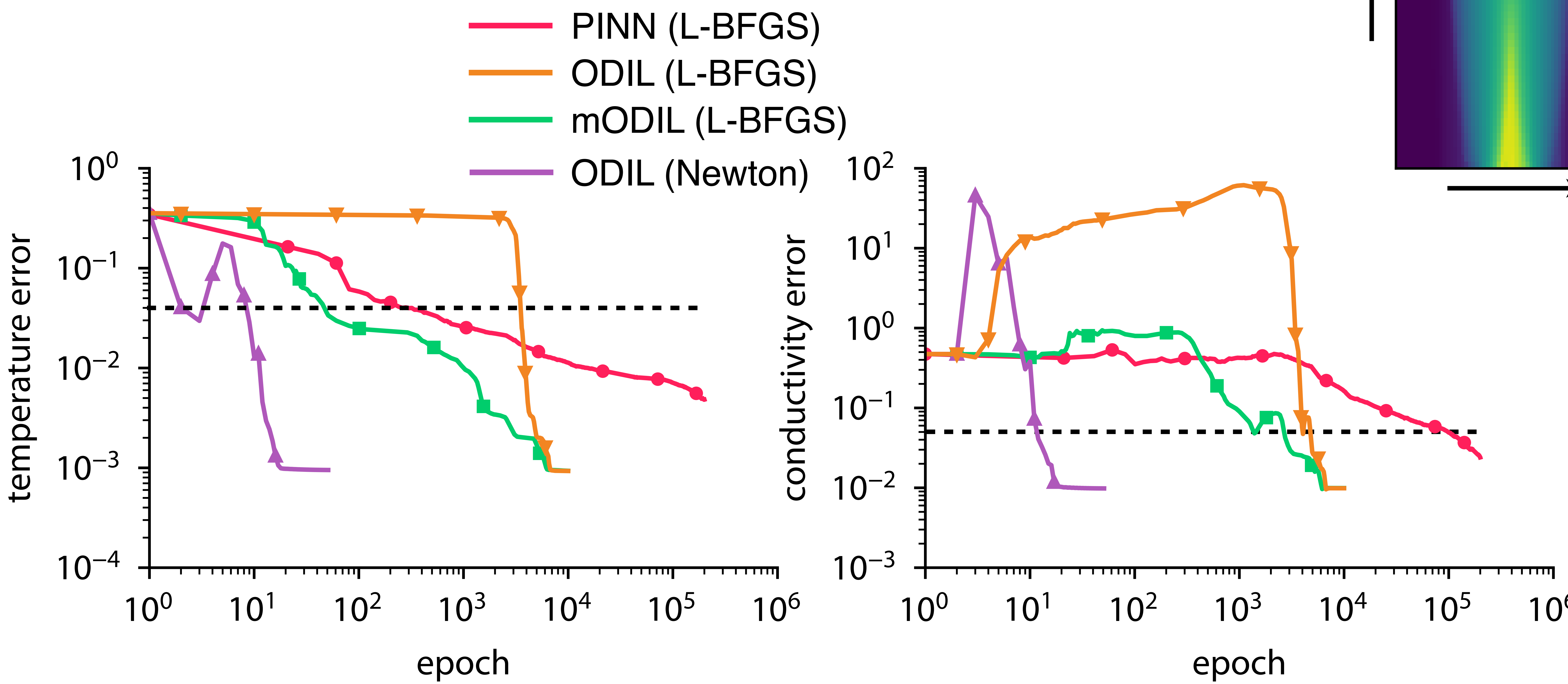
- Infer conductivity $k(u)$ in the heat equation

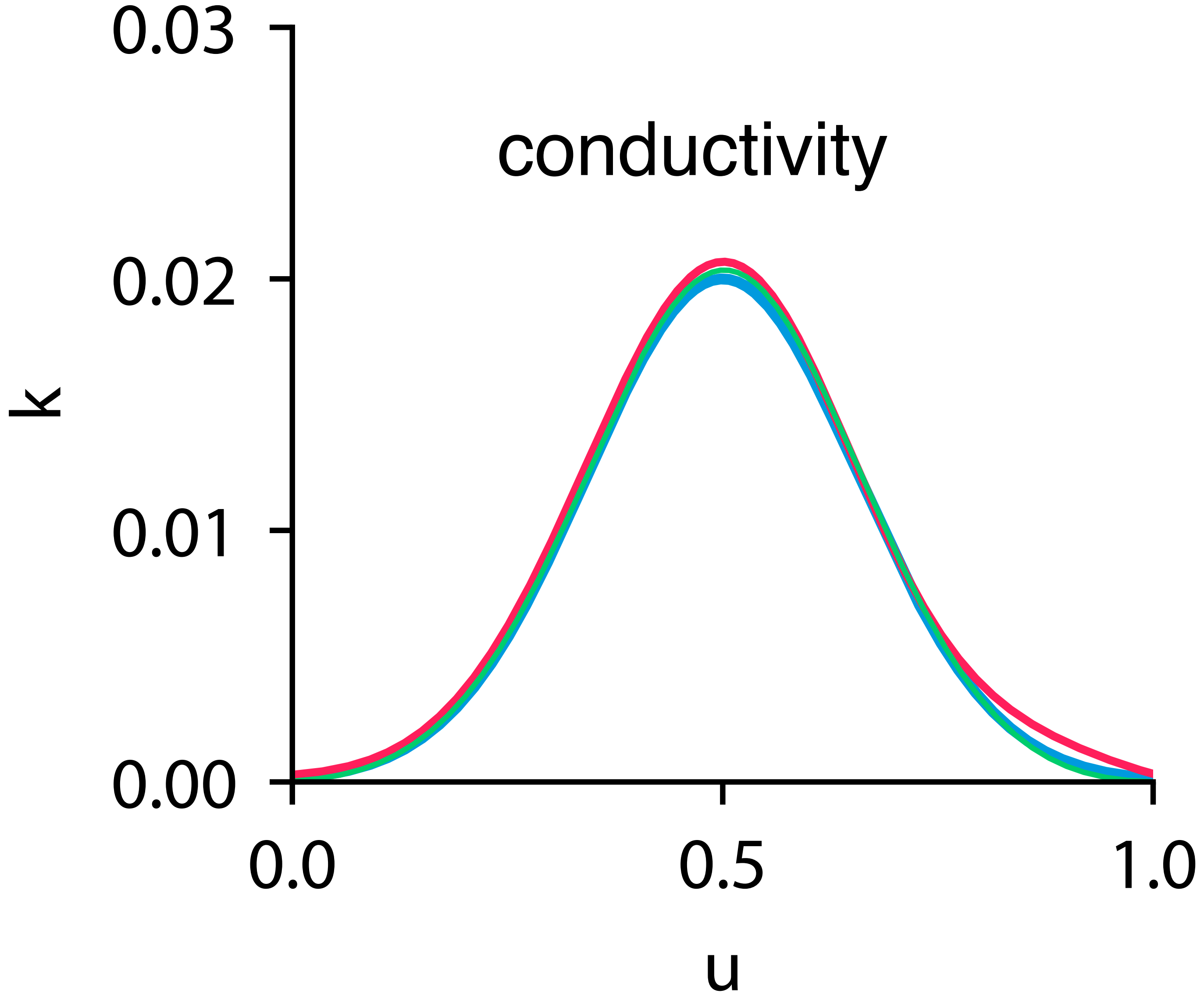
$$\frac{\partial u}{\partial t} - \frac{\partial}{\partial x}\left(k(u)\frac{\partial u}{\partial x}\right) = 0$$

$k(u)$ as neural network

given temperature measurements (dots)





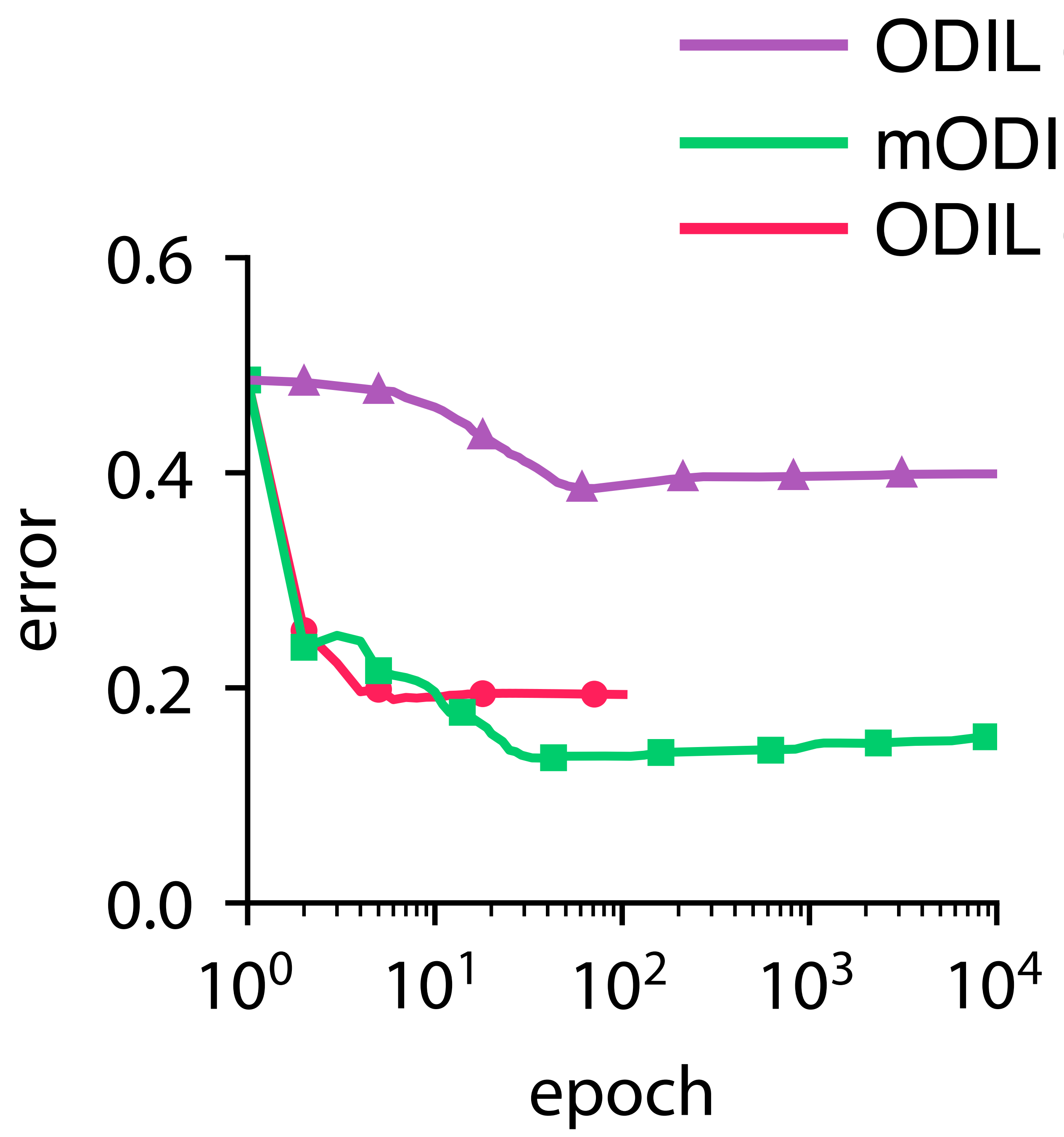


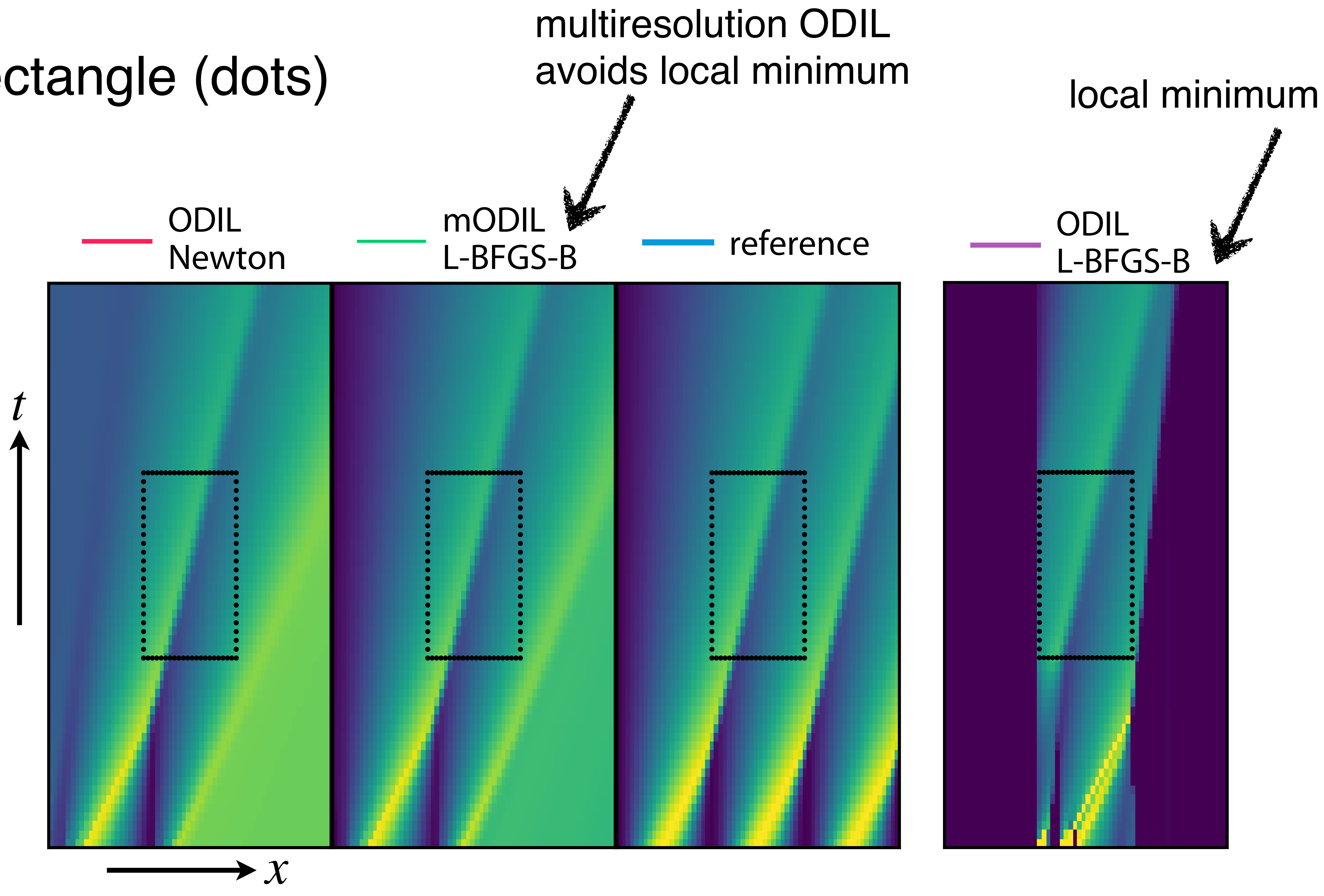
Reconstruction for Burgers equation

- Reconstruct solution of the Burgers equation

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0$$

given measurements on edges of rectangle (dots)





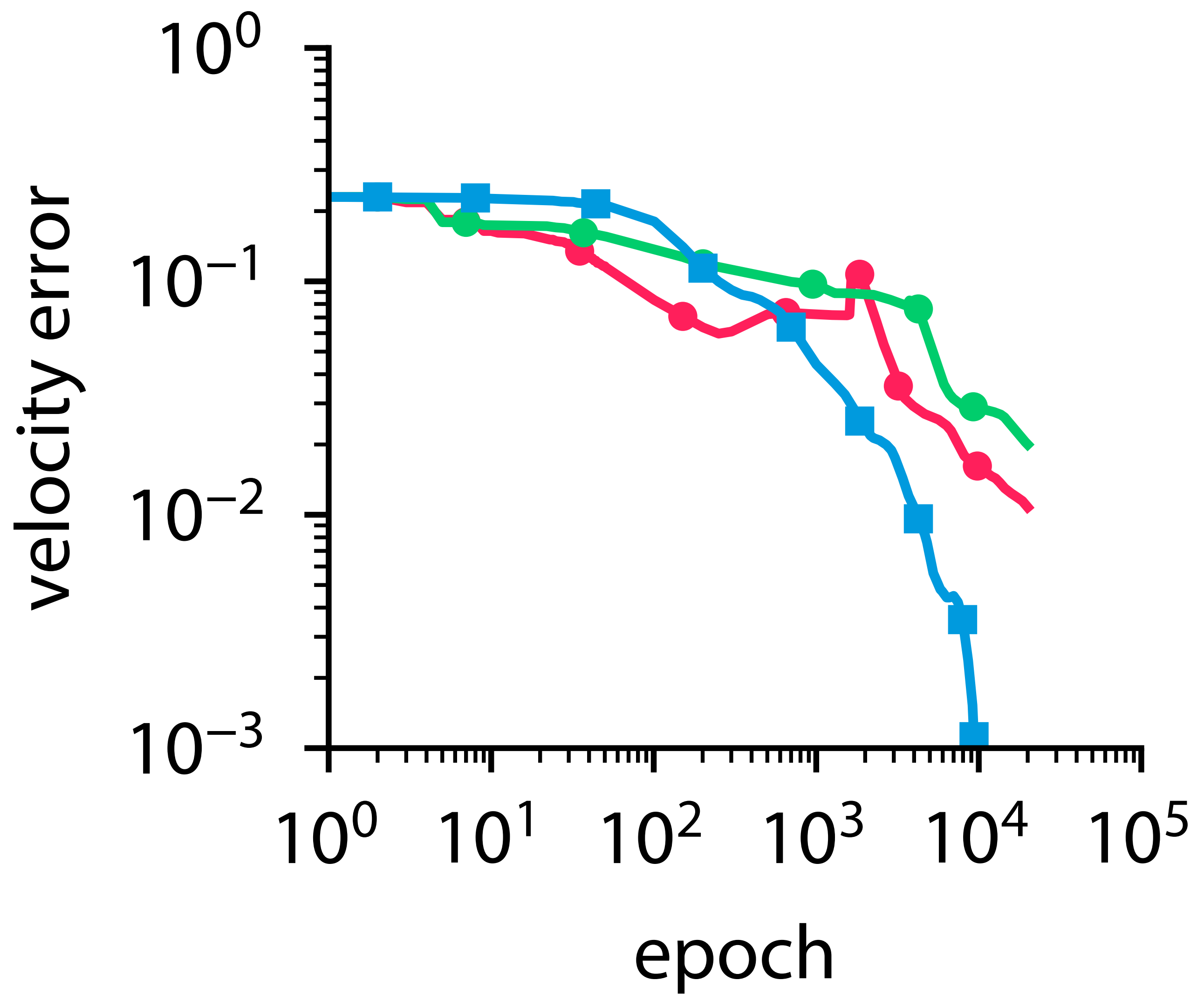
Body shape from velocity

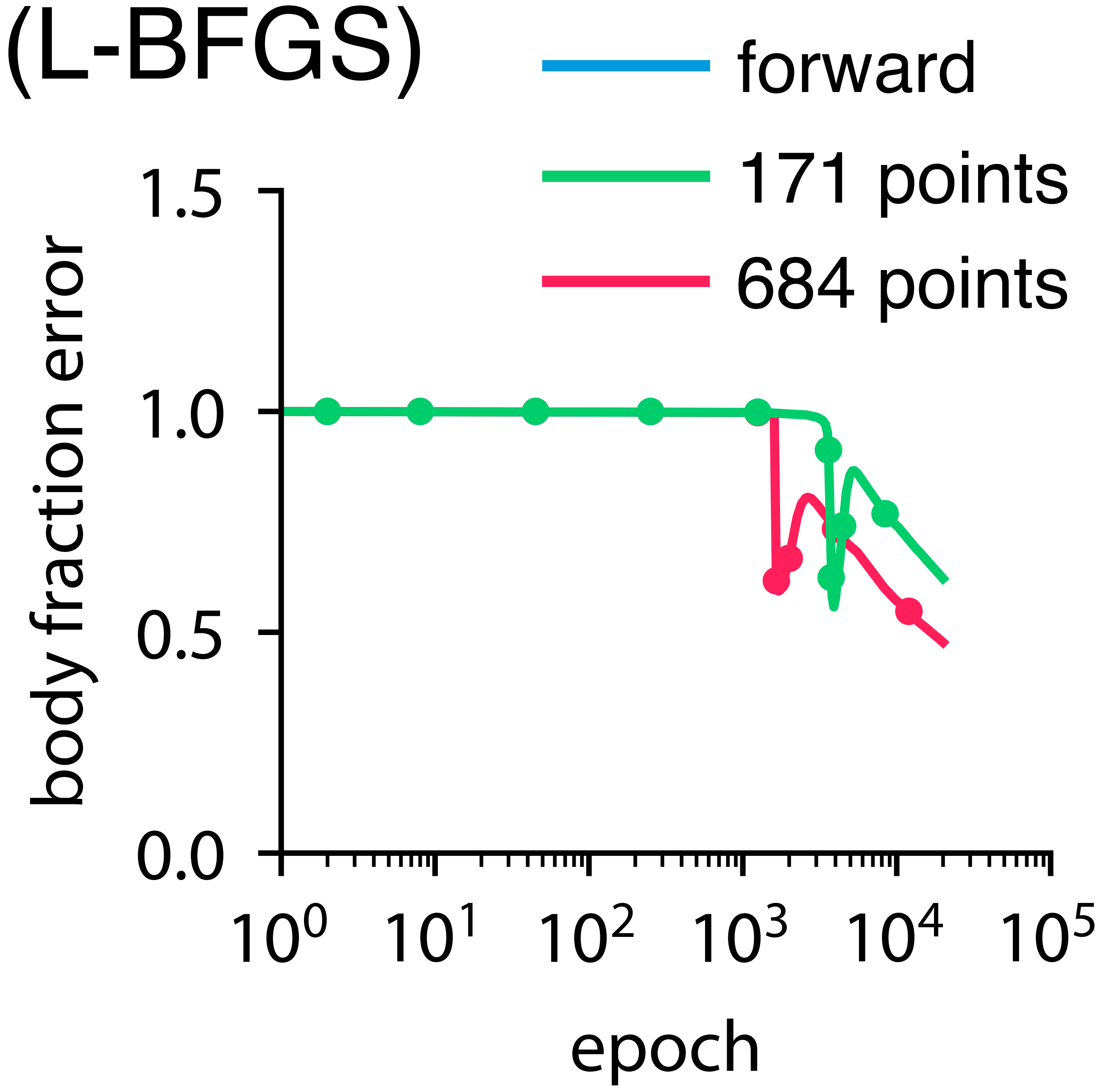
- Infer body fraction $\chi(\mathbf{x})$ given
  3D steady Navier-Stokes with penalization

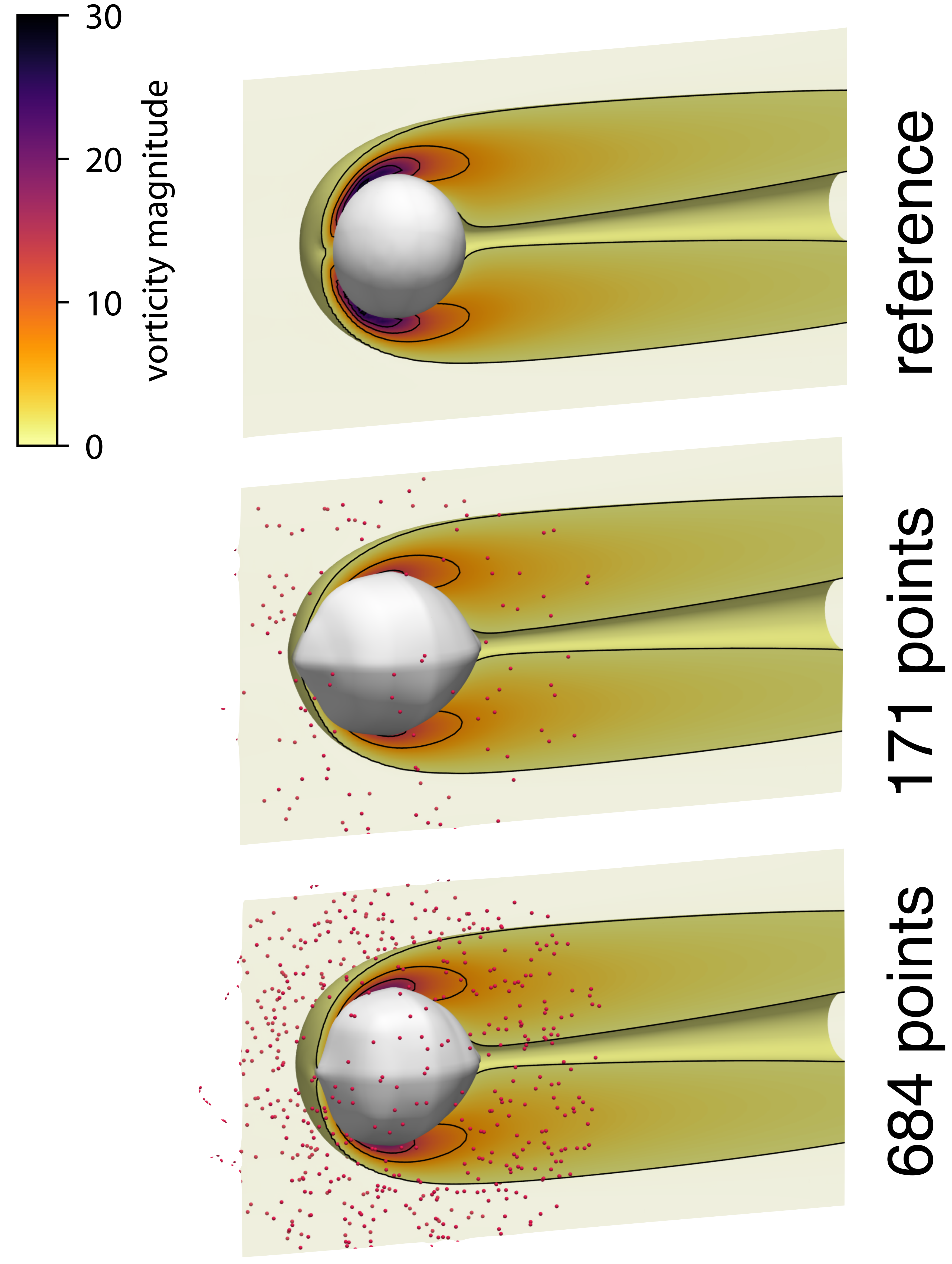
$$\nabla \cdot \mathbf{u} = 0$$
$$(1-\chi)\Big((\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p - \frac{1}{\mathrm{Re}}\nabla^2\mathbf{u}\Big) + \lambda\chi\mathbf{u} = 0$$

  and velocity measurements (red dots)

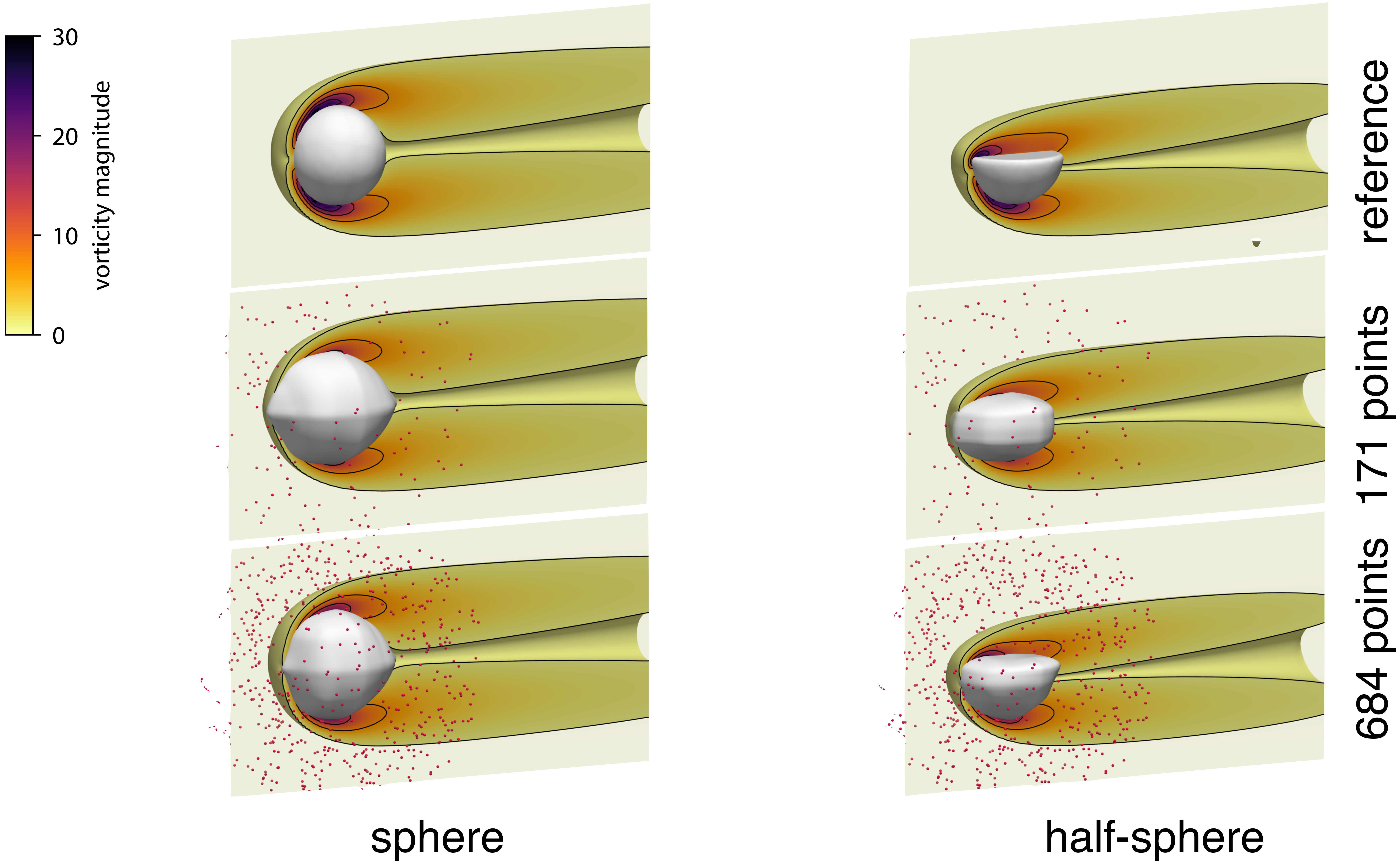
- Convergence history of mODIL (L-BFGS)







Body shape from velocity



Conclusion

1. ODIL is orders of magnitude faster than PINN
2. mODIL with multigrid decomposition accelerates convergence of standard optimizers

Thank you!

arXiv:2303.04679