SOLVING STOKES FLOW WITH HYBRID ML-SIMULATION METHODS

Franziska Griese, Philipp Knechtges, Alexander Rüttgers
German Aerospace Center (DLR), Germany
CFC 2023

DLR

Why Hybrid ML-methods?

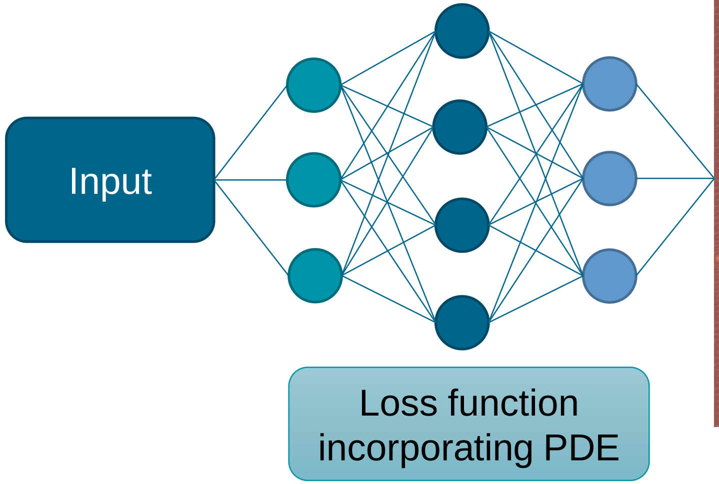
- Modeling of complicated technical systems needs surrogate modeling
- By data-driven ML approaches some natural laws are not or only poorly considered

Conservation of

Energy | Mass | Momentum

How to make ML safer and more robust?

- By teaching it the physical laws



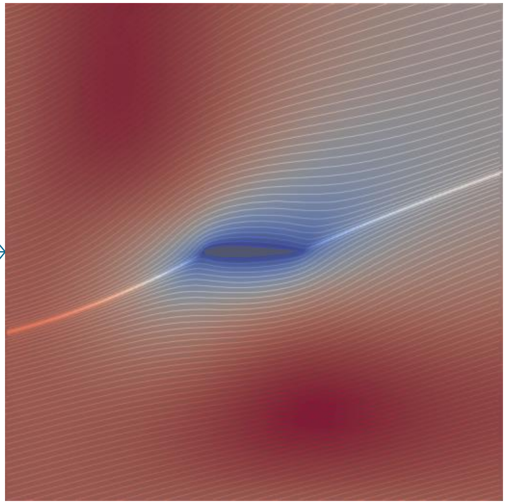


Figure: Velocity field around an airfoil with an angle of attack of 15°.

Franziska Griese, German Aerospace Center

2D Stokes flow around an airfoil

- PDE:

$$\nabla \cdot u = 0,$$
$$\nabla p - \Delta u = 0$$

- with velocity u and pressure p
- BC: Dirichlet at inflow, Neumann at outflow, no-slip at airfoil

- Saddle point problem
- Even for FEM no straight-forward problem

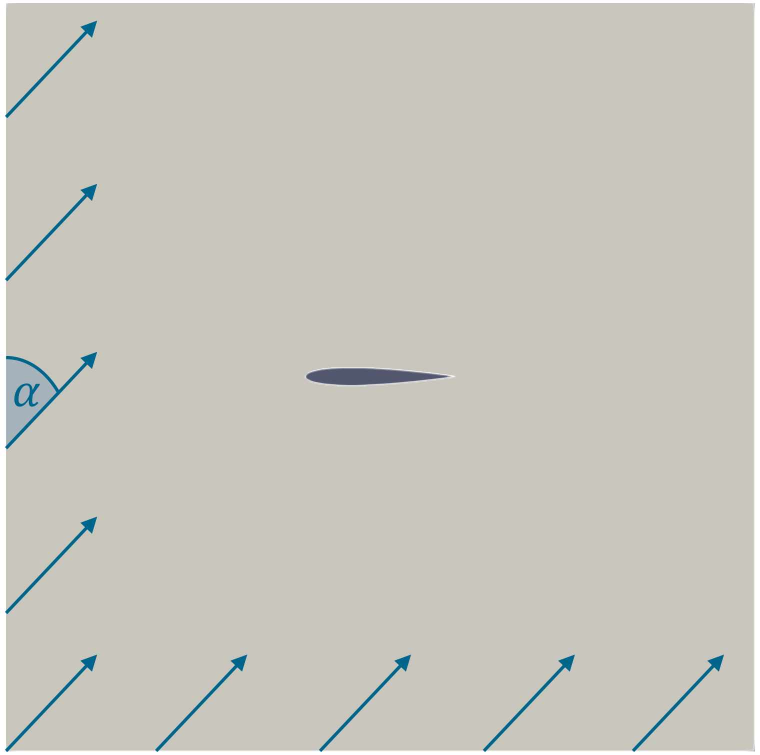
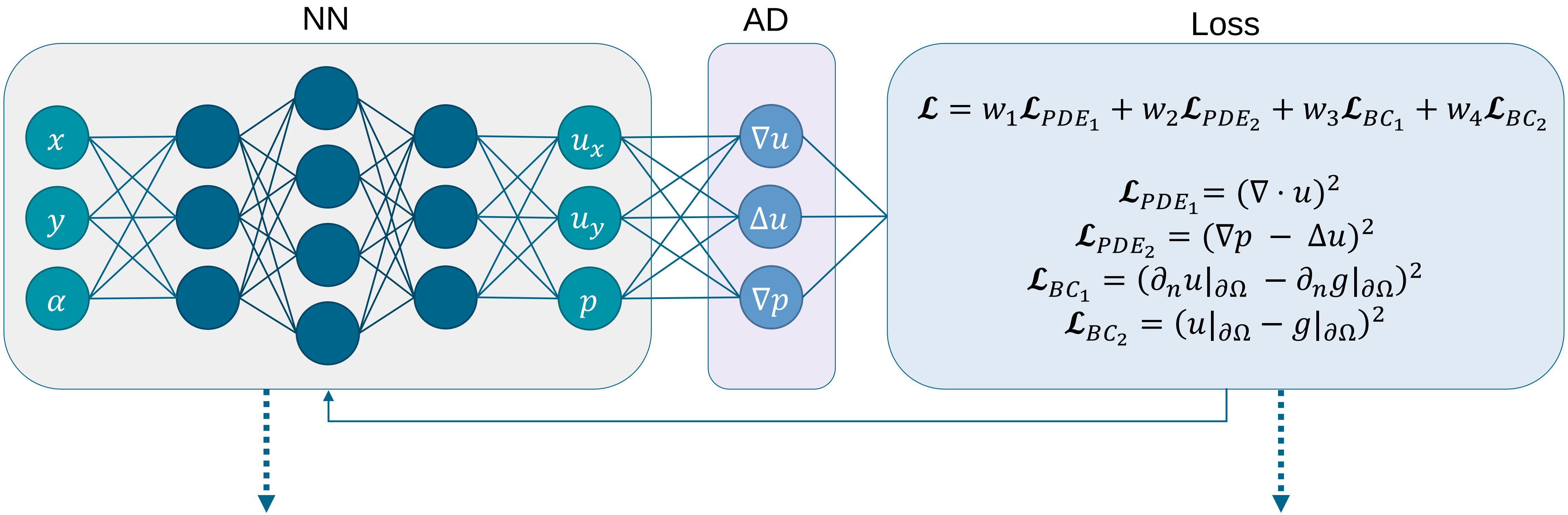


Figure: Domain around NACA 0012 airfoil with an angle of attack α.

Physics-Informed Neural Networks (PINNs [1])



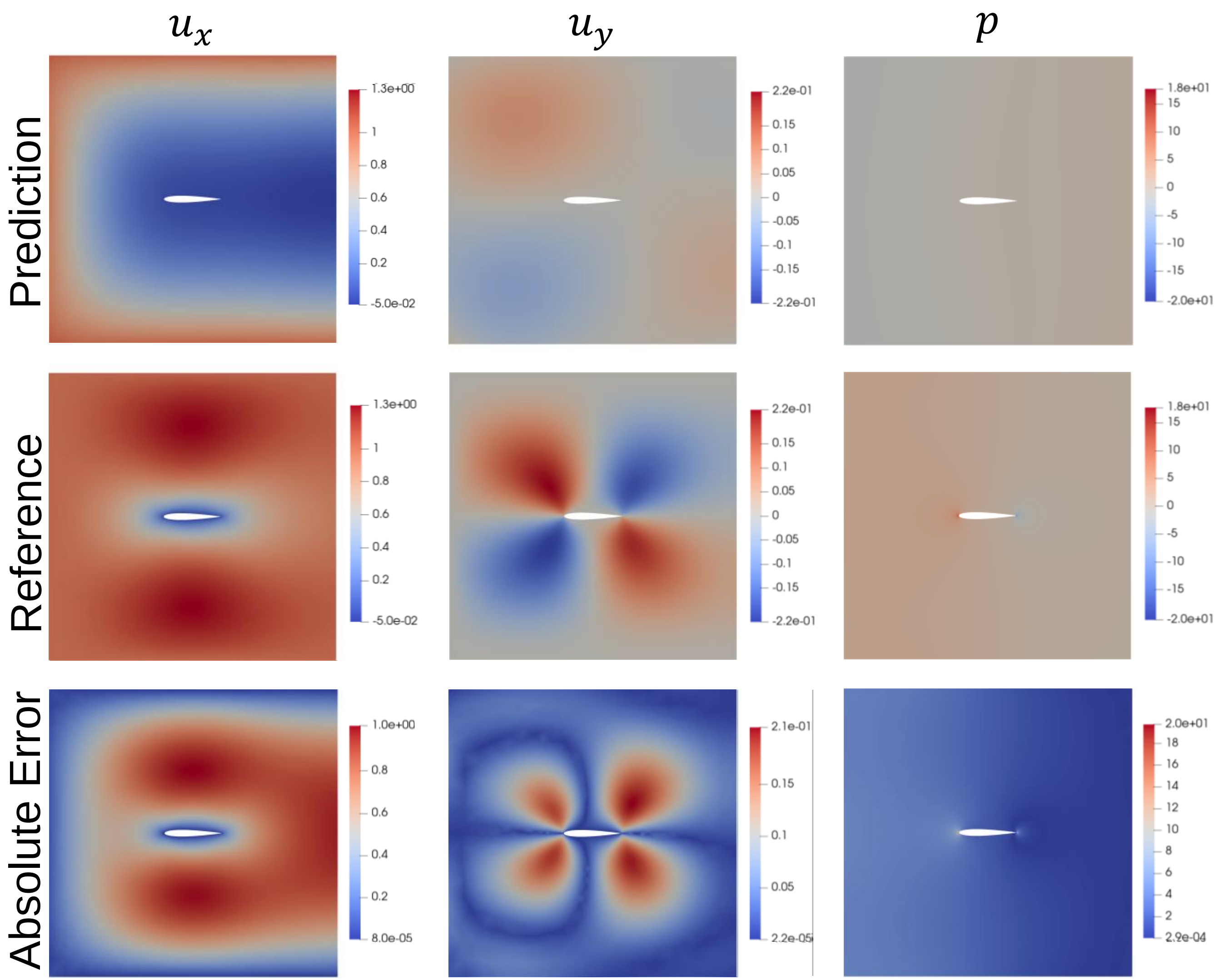
- Universal function approximator
- Fast prediction after training
- Parameterizable

- Physics-driven learning process
- More interpretability
- Expensive training due to higher order derivatives
- Multi-objective optimization

[1] M. Raissi et al., Physics-Informed Neural Networks (2019)

PINNs - Stokes

- Training with $\alpha = 0$:
- Yields unphysical results
- No guarantee for finding a global minimum of the loss



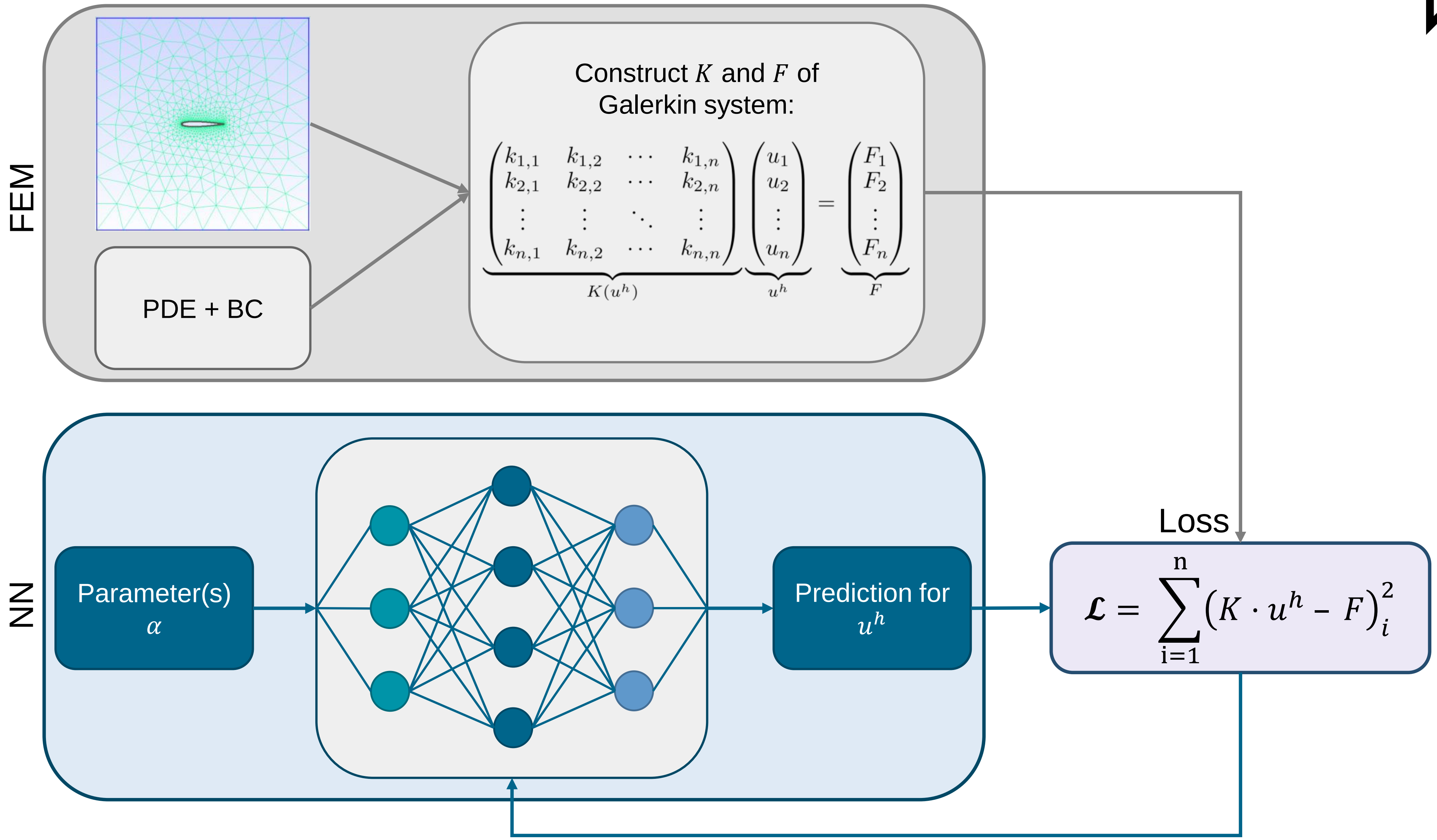
L^2-error: 3.3119 0.3645 12.9112

Finite Element Neural Networks (FENNs)

- Combination of classical finite element method and NN
- Use best of both worlds

FEM	NN
Sparsity patterns	Universal function approximators
No multi-objective optimization	Fast prediction after training
Numerical theory of errors	Parameterizable

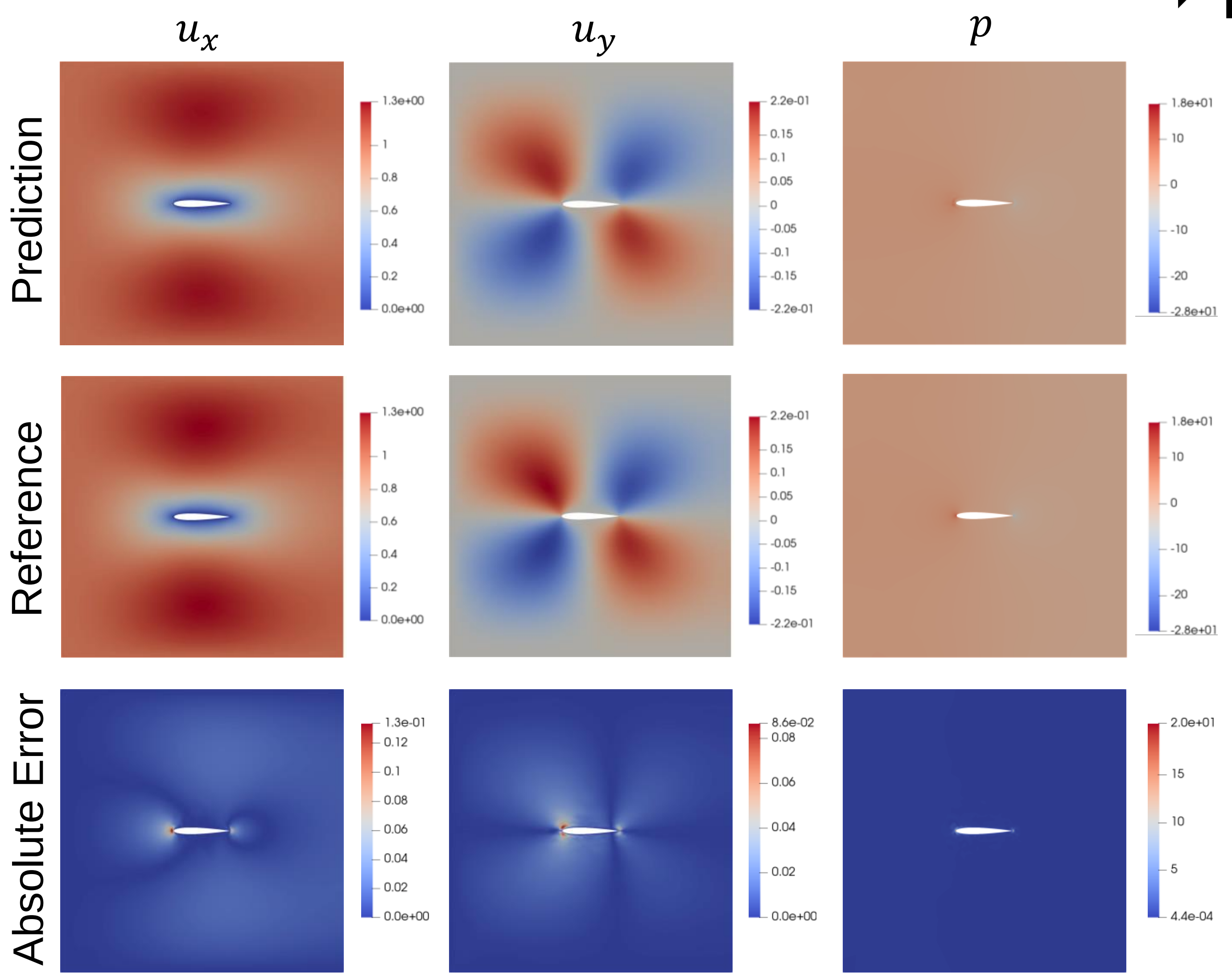
FENN



FEM
PDE + BC
Construct K and F of
Galerkin system:
$\underbrace{\begin{pmatrix} k_{1,1} & k_{1,2} & \cdots & k_{1,n} \\ k_{2,1} & k_{2,2} & \cdots & k_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ k_{n,1} & k_{n,2} & \cdots & k_{n,n} \end{pmatrix}}_{K(u^h)} \underbrace{\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix}}_{u^h} = \underbrace{\begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_n \end{pmatrix}}_{F}$
NN
Parameter(s)
α
Prediction for
u^h
Loss
$\mathcal{L} = \sum_{i=1}^{n} \left(K \cdot u^h - F\right)_i^2$

FENNs – Stokes

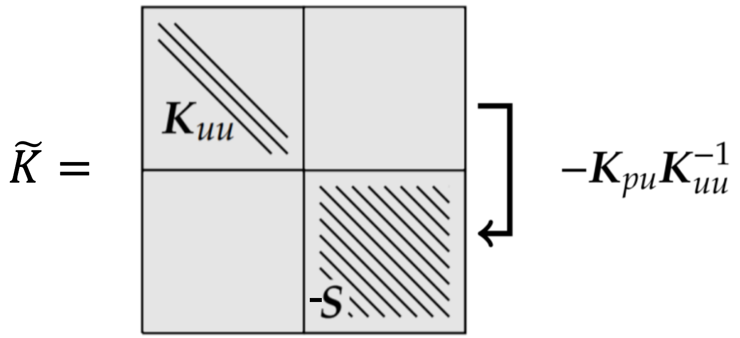
- Taylor-Hood elements to construct Galerkin system → avoid instabilities
- Training with $\alpha = 0$ and 2000 epochs of LBFGS
- Prediction similar to reference!



L^2-error: 0.0523 0.0317 0.6065

FENNs – Stokes Preconditioned

- Improve condition of indefinite stiffness matrix K
- Preconditioning with:
  - Schur complement
  - Cholesky decomposition $\widetilde{K} = L \cdot L^T$



$$S = -K_{up}K_{uu}^{-1}K_{pu}$$

- Training with $\alpha = 0$ and 2 epochs of LBFGS:

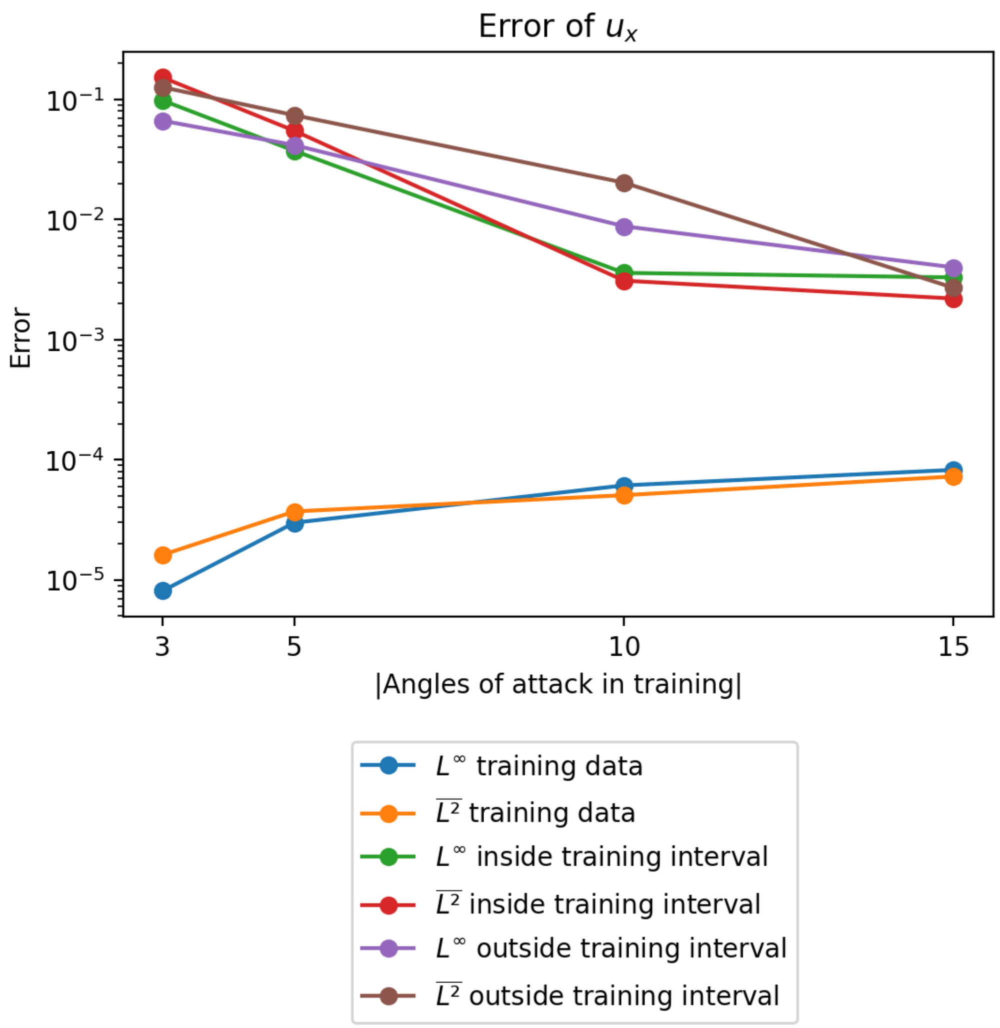
	$\boldsymbol{u_x}$	$\boldsymbol{u_y}$	$\boldsymbol{p}$
L^∞	1.1108e-08	1.7542e-09	1.5471e-07
L^2	4.4982e-08	3.4358e-09	2.8849e-08

Parameterizabilty

- Training with various numbers $\alpha \in [1, 45]$

Generalizability

- Testing inside training interval with random numbers
- Testing outside training interval with $\alpha \in \{47.5, 50, 55\}$



Conclusion and Outlook

Results

- PINN yields unphysical Stokes results
- Preconditioned FENN shows the ability to parameterize and generalize well

Next steps for FENNs

- Inverse problems
- Uncertainty Quantification

Contact: Franziska.Griese@dlr.de

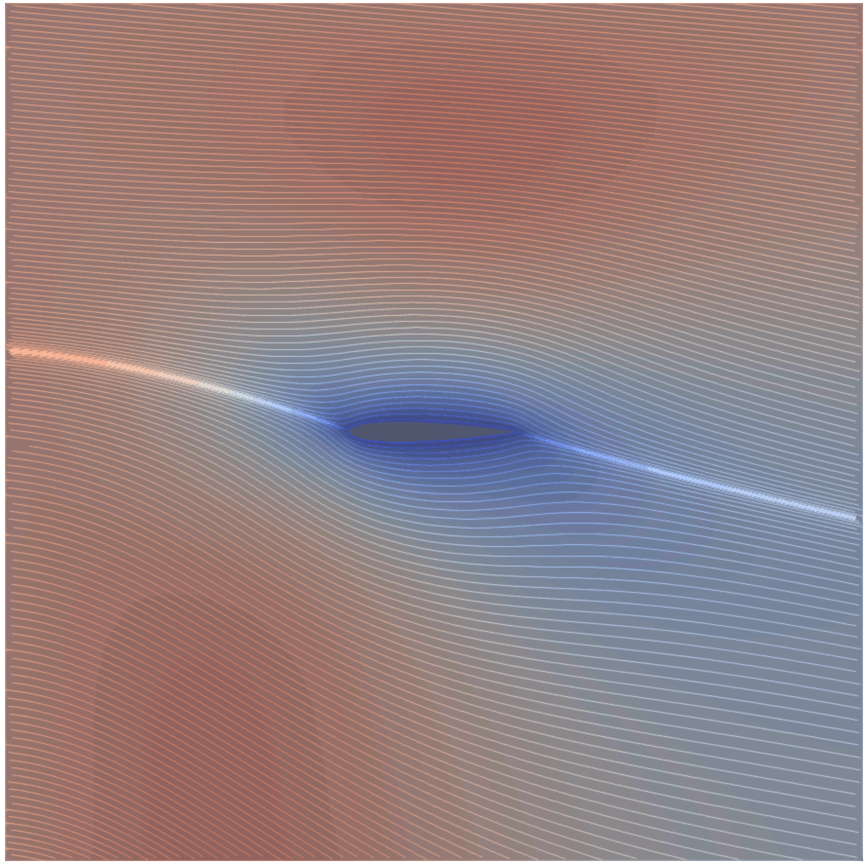


Figure: Velocity field around an airfoil with various angles of attack calculated from a FENN.