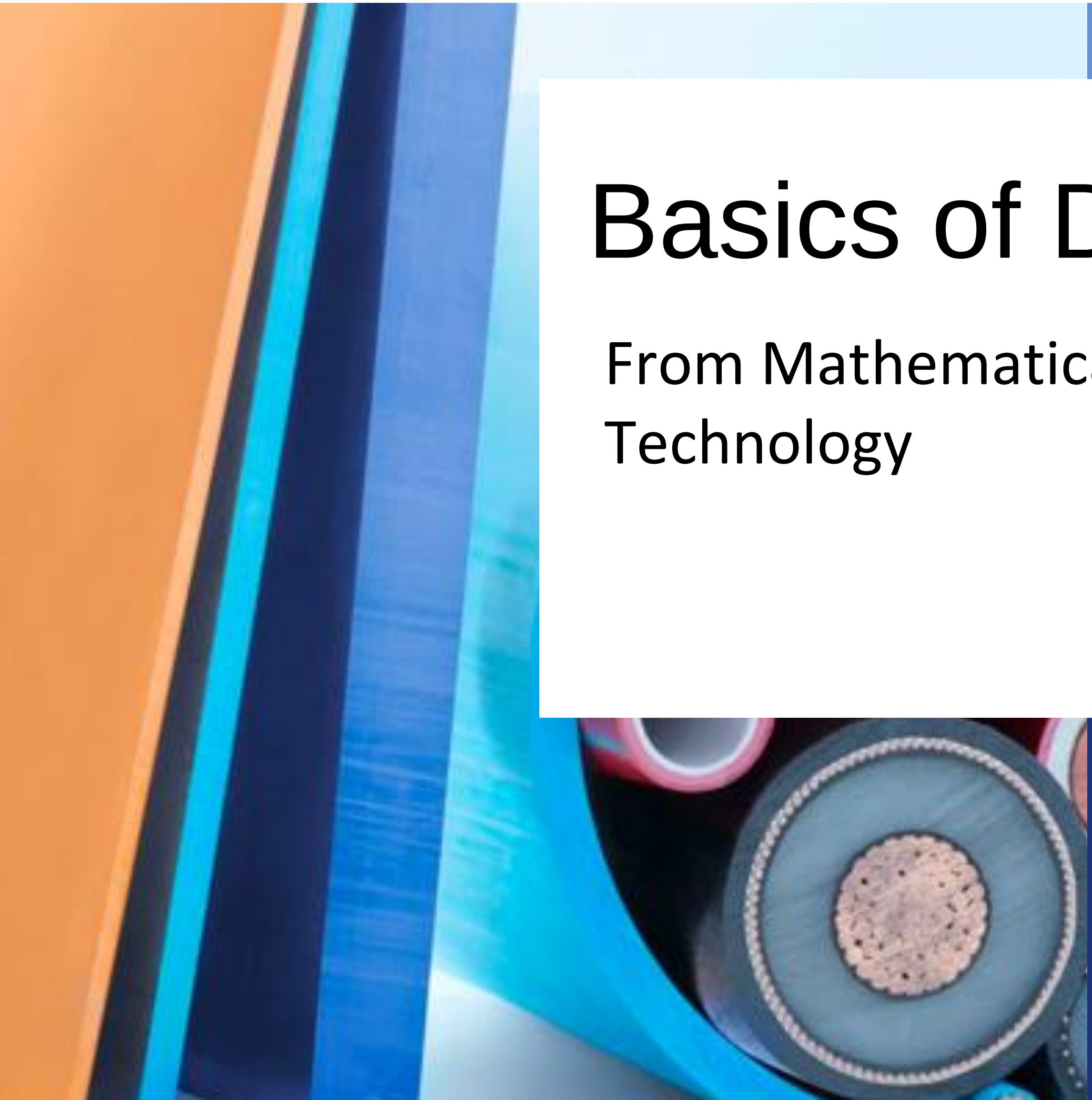
Universität Bremen

Zentrum für Technomathematik

Prof. Dr. Dr. h.c. Peter Maass



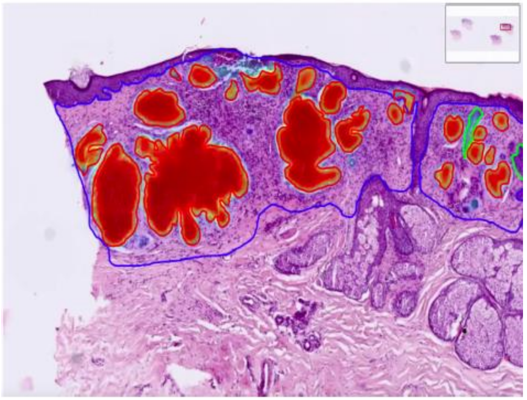
Basics of Deep Learning

From Mathematical Research to Technology

KoMSO Academy 2023
Heidelberg, Nov. 7 2023

AI beyond Computer Vision

Industrial applications
VW, Ariane Airbus, Siemens, DB, ….



Digital pathology
1st clinial installation

CT tomography
Academic example

PDE models $$u_t + v^t \nabla u + \mathrm{div}((1 - D)\nabla u) = f$$

given measured u, determine (v, D, f)

Outline

1 Neural networks

- Network architecture
- Data manifold
- Regularization by architecture

2 An industrial CT application

3 PDE based applications

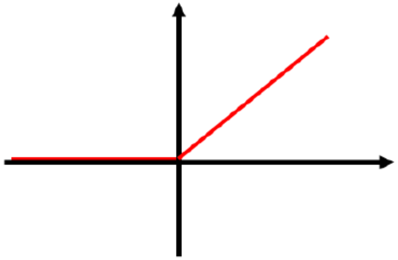
- Limitations of present DL concepts
- Topology optimization, Ariane

Neural networks for function approximation

Parameterized function systems: trigonometric functions, splines, wavelets, neural nets...

$$\Phi_\Theta(x) = \varphi(W_3\ \varphi(W_2\ \varphi(W_1 x + b_1) + b_2) + b_3)$$

- Parameters: $W_j, b_j,\ j = 1, M$
  e.g. $W_j \in \mathbb{R}^{n_j \times m_j}$ fully connected, MLP
  Toeplitz matrices, CNN
- Activation function $\varphi : \mathbb{R} \to \mathbb{R}$
  e.g. φ chosen as tanh, ReLu, ...
- Accuracy, complexity
  Number of parameters Unet: $\sim 8\ 10^6$



Neural networks for function approximation

Parameterized function systems: trigonometric functions, splines, wavelets, neural nets...

$$\varphi_\Theta(x) = \varphi(W_3\ \varphi(W_2\ \varphi(W_1 x + b_1) + b_2) + b_3)$$

- Least squares regression for given data $(x_i, y_i), i = 1, .., N$

$$\Theta_{opt} = argmin_\Theta \sum_i^N \|\varphi_\Theta(x_i) - y_i\|^2$$

  Training: costly minimization, stochastic gradient descent
- Application of trained network with fixed Θ_{opt}

$$\varphi_{\Theta_{opt}}(x_{new}) = \varphi(W_3\ \varphi(W_2\ \varphi(W_1 x_{new} + b_1) + b_2) + b_3)$$

  Linear operations, pointwise non-linearities, very efficient

Feedforward Neural Network

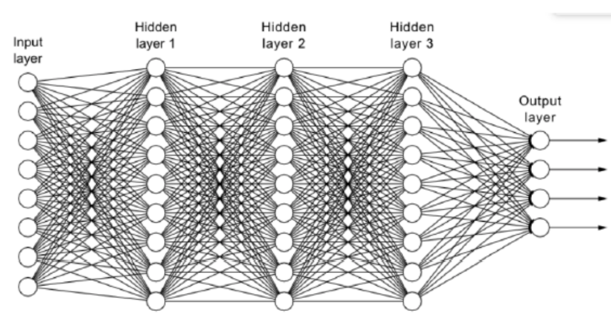
$$x_L = x_3 = \varphi(z_3)$$
$$\uparrow$$
$$z_3 = W_3 x_2 + b_3$$
$$\uparrow$$
$$x_2 = \varphi(z_2)$$
$$\uparrow$$
$$z_2 = W_2 x_1 + b_2$$
$$\uparrow$$
$$x_1 = \varphi_1(z_1)$$
$$\uparrow$$
$$z_1 = W_1 x_0 + b_1$$
$$\uparrow$$
$$x^0 = x$$

Feedforward neural network with L layers:

$$\begin{aligned} \text{input:}\quad & x_0 \\ & z_k = W_k x_{k-1} + b_k \\ & x_k = \varphi(z_k) \\ \text{ouput:}\quad & x_L \end{aligned}$$

φ non-linear, componentwise, network parameters $\Theta = \{W_k, b_k\}$

$$\varphi_\Theta(x_0) = x_L = \varphi(W_3\ \varphi(W_2\ \varphi(W_1 x + b_1) + b_2) + b_3)$$



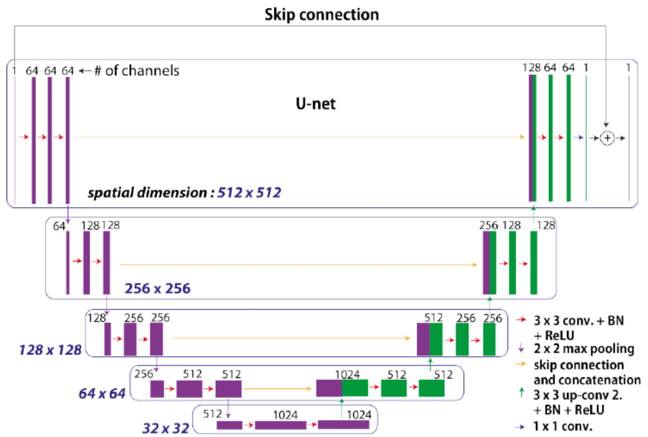
Feedforward Neural Network

$$x_L = x_3 = \varphi(z_3)$$
$$z_3 = W_3 x_2 + b_3$$
$$x_2 = \varphi(z_2)$$
$$z_2 = W_2 x_1 + b_2$$
$$x_1 = \varphi_1(z_1)$$
$$z_1 = W_1 x_0 + b_1$$
$$x^0 = x$$

Feedforward neural network with L layers:

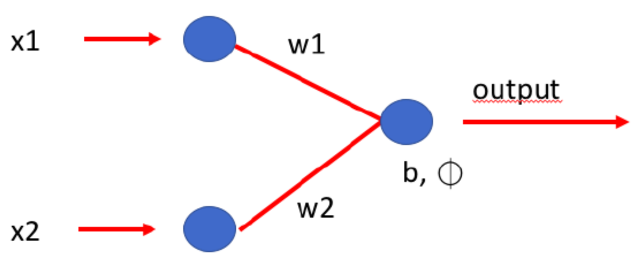
$$\begin{aligned} \text{input:} \quad & x_0 \\ & z_k = W_k x_{k-1} + b_k \\ & x_k = \varphi(z_k) \\ \text{ouput:} \quad & x_L \end{aligned}$$

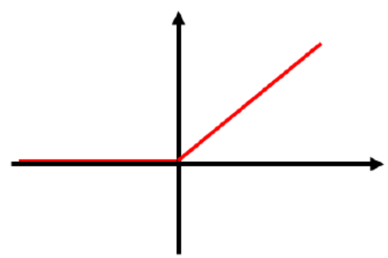
$$\varphi_\Theta(x_0) = x_L = \varphi(W_3\ \varphi(W_2\ \varphi(W_1 x + b_1) + b_2) + b_3)$$

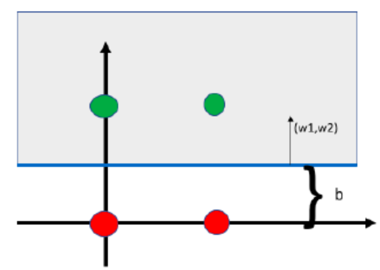


Trivial Network

- Network without inner layer, *ReLU*- activation
- Multiplikationsnetzwerk, Input (x, y), Output xy (Produkt)
- Matrix-Vektor-Multiplikation





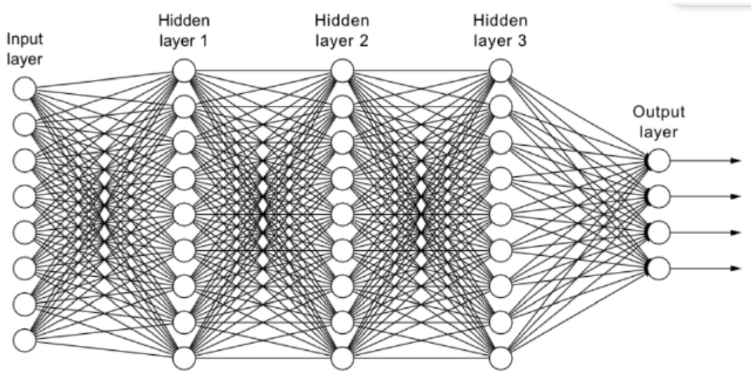


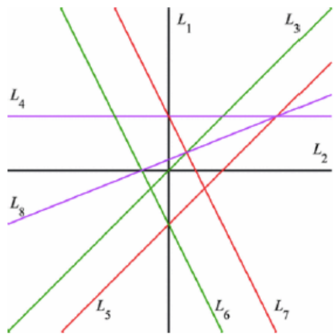
$$\text{output} \quad z = \varphi(w1 * x1 + w2 * x2 - b)$$

$$w1 * x1 + w2 * x2 - b \;\geq\; 0$$

Arrangement of hyperplanes

- Adaptive segmentation of input space
- Exponentially increasing complexity
- piecewise linear function

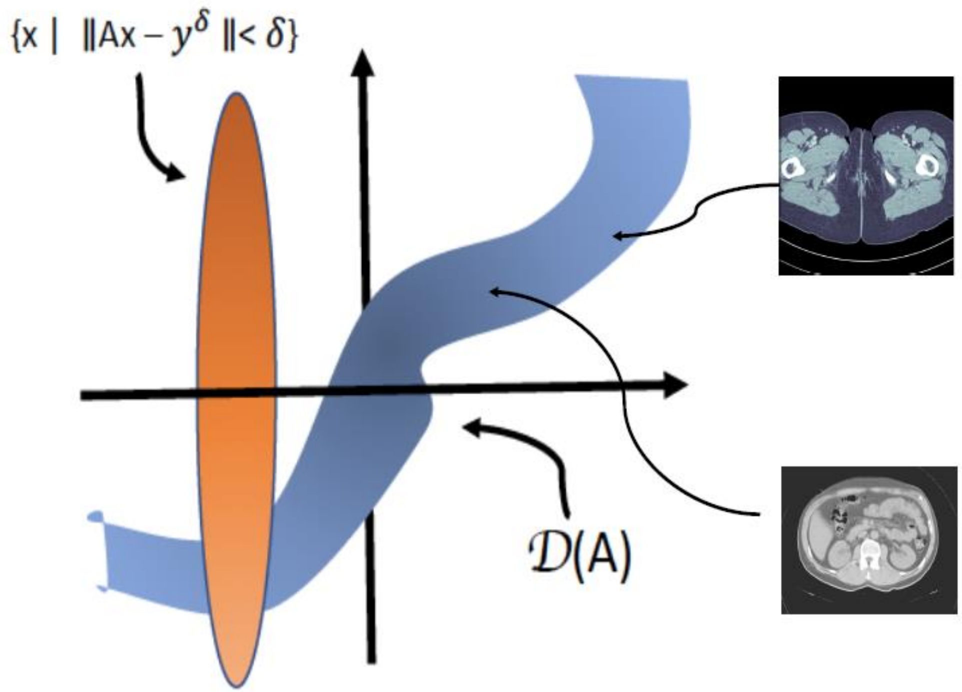




Approximation of complex functions by adaptive, linear cells

Exploiting the data manifold

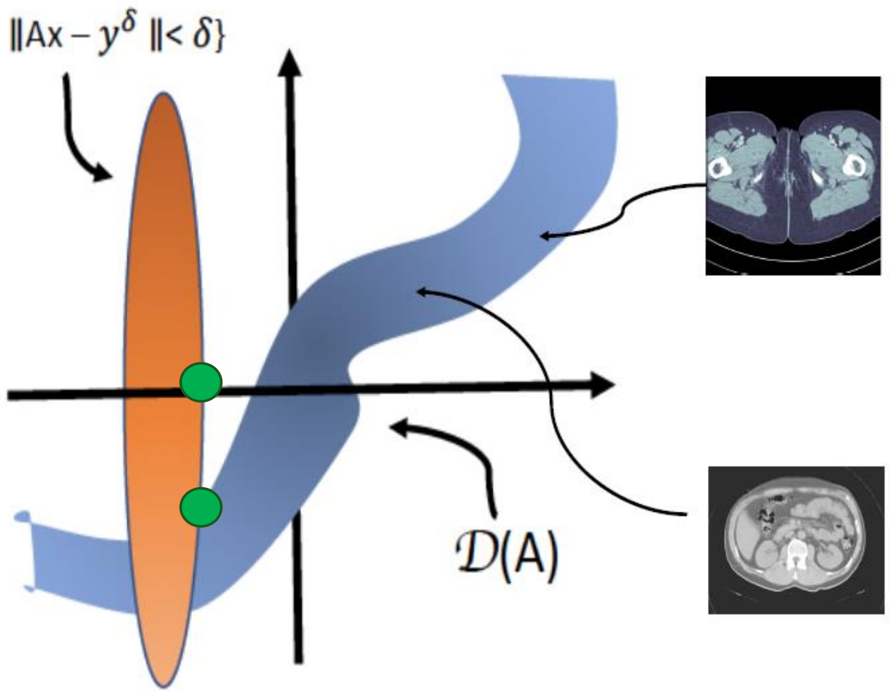
$Ax \sim y^{\delta}$, "No model is perfect", "Not every matrix is an image"



Mallat, Haltmeier, Adler, Öktem, Lunz, Schönlieb, Arridge, Hauptmann, Grasmaier, Harrach, Dittmer, Otero, ...

Exploiting the data manifold

$Ax \sim y^\delta$, "No model is perfect", "Not every matrix is an image"



Mallat, Haltmeier, Adler, Öktem, Lunz, Schönlieb, Arridge, Hauptmann, Grasmaier, Harrach, Dittmer, Otero, ...

Advantages of deep learning concepts

"No model is perfect", "Not every matrix is an image"

Postprocessing of analytic reconstructions

- Learned projection

$$\varphi : X \to \mathcal{D}(A)$$

$$y^\delta \to x \to \varphi(x)$$

Learned prior distribution

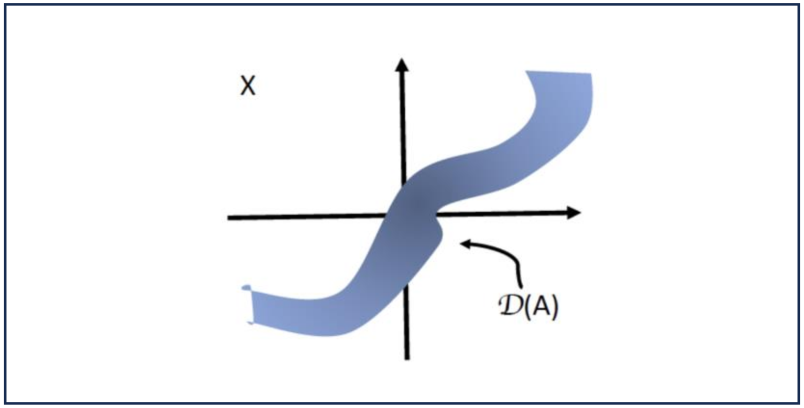
- probability for $x \in \mathcal{D}(A)$

$$min_x \|Ax - y^\delta\|^2 - log\left(\varphi(x)\right)$$

Learning operator updates

- Residual data $\left(x^{(d)}, y^{(d)} - Ax^{(d)}\right)$

$$\min_x \|Ax + \varphi(x) - y^\delta\|^2 + \alpha R(x)$$



Theoretical results

- Complexity and Approximation

  Cybenko (1989), Hornik, Stinchcombe, White (1989), Chen, Chen (1995), Yarotsky (2017), Grohs, Petersen, et al. (2019), Karniadakis et al. (2021), Gribonval, Kutyniok, Nielsen, Voigtländer (2021), Langer, Ay (2021), Shen, Yang, Zhang (2021)....

- Stability of network architectures

  Gregor, LeCun (2010), Kunisch, Pock (2013), Chen, Liu, Wang, Yin (2018), Li, Schwab, Antholzer, Haltmeier (2018), Jin, Lu (2018), Lunz, Schönlieb (2018), Arridge, PM, Öktem, Schönlieb (2019), Jacobsen, Behrmann, Zemel, Bethge (2020), Ruiz, Gama, Ribeiro (2020), Baguer, Leuschner, et al (2020), Ruthotto, Steidl (2023)....

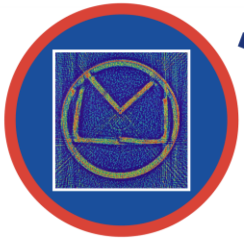
- Neural networks and PDEs

  Weinan E (2020), Kutyniok, Petersen, Raslan, Schneider (2019), Aarset, Holler, Nguyen (2022), Bhattacharya, Hosseini, Kovachi, Stuart (2020), Lanthaler, Mishra (2021), Jiang, Willet (2021), Lu, Jin, Karniadakis (2019), De Hoop, Huang, Qian, Stuart (2022), Tanyu, PM et al. (2023)....

Ill-posed inverse problem

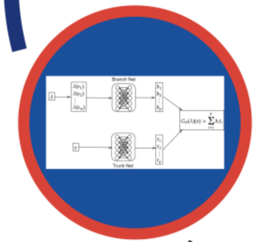
given $y^\delta = F(x) + \eta$

$$\min_x \|F(x) - y^\delta\|\|^2 + \alpha R(x)$$



Function-operator approximation
Regularization – stability

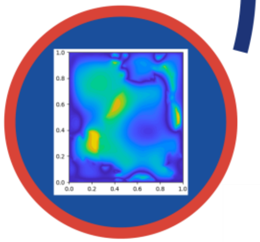
Parameter identification
F parameter-to-state map



Data driven neural network

Θ network parameters

$$\min_\Theta \sum_i \|\varphi_\Theta(x_i) - y_i\|^2 + \alpha R(\Theta)$$



Partial differential equation

$$-\nabla(\lambda \nabla u) = f$$

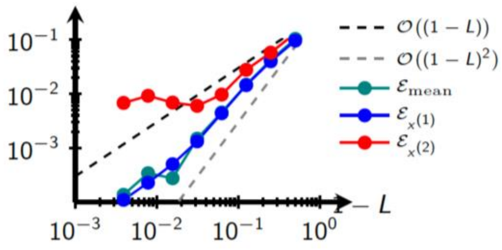
given u^δ determine λ

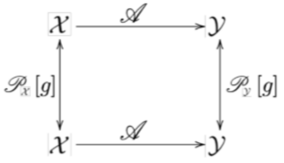
Efficiency/complexity – accuracy
Parametric studies, approximation

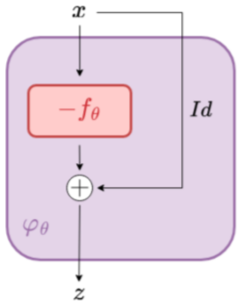
Regularization by architecture

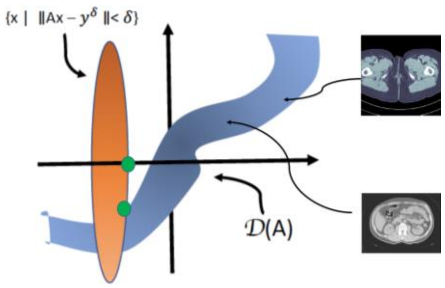
Network architectures for inverse problems

Operator approximation	Equivariance	Invertible networks	Deep prior concepts









Janek Gödeke
quantifiable approx.
general activation

generalized CNN
group transform
stability

iResNet architecture
$y_{j+1} = y_j + \Phi_\Theta(y_j)$
Optimal convergence

$min\|A\Phi_\Theta(z) - y^\delta\|^2$
no training data
bi-level optimization

D. Nganyu Tanyu, J. Ning, T. Freudenberg, N. Heilenkötter, A. Rademacher, U. Iben, P. Maaß. Deep learning methods for partial differential equations and related parameter identification problems. Inverse Problems, 39(10), 2023.

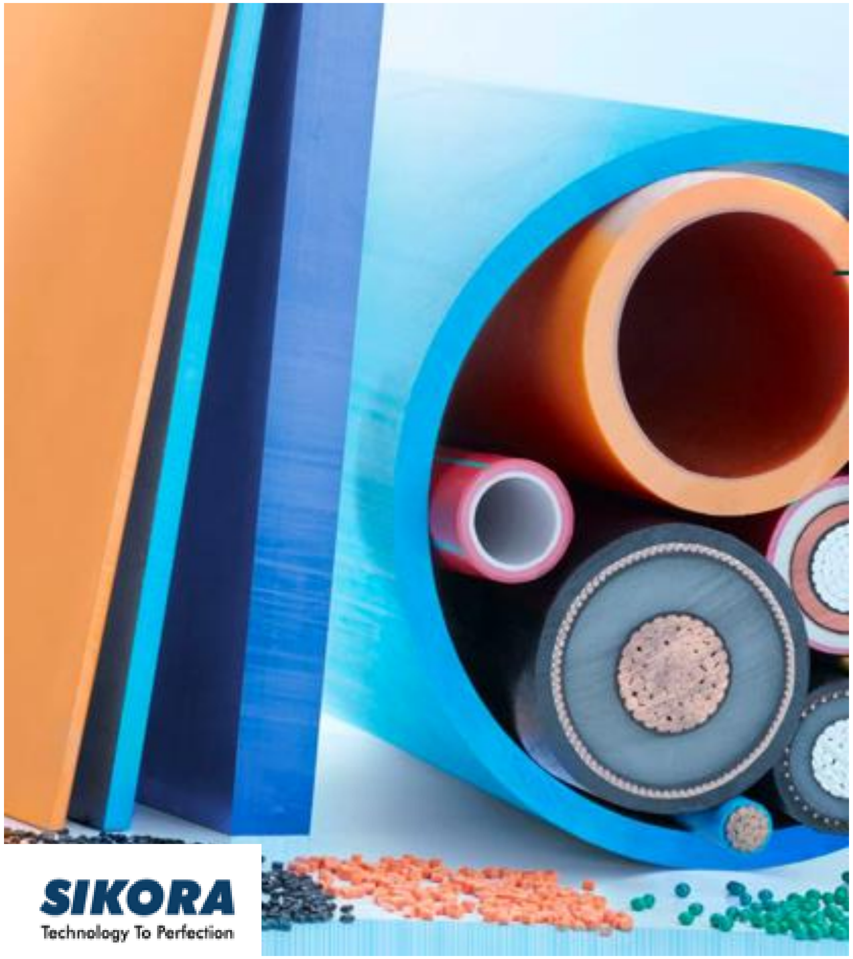
C. Arndt, A. Denker, S. Dittmer, N. Heilenkötter, M. Iske, T. Kluth, P. Maaß, J. Nickel. Invertible residual networks in the context of regularization theory for linear inverse problems. Submitted for publication, 2023.

M. Beckmann, N. Heilenkötter. Equivariant Neural Networks for Indirect Measurements. Submitted for publication, 2023.

Outline

1. Neural networks
   - Network architecture
   - Data manifold
   - Regularization by architecture
2. An industrial CT application
3. PDE based applications
   - Limitations of present DL concepts
   - Topology optimization, Ariane

X-Ray technology for in-process quality monitoring



03/2021 – 09/2022

Simulation of measurement process, off-line computation
Co-operation funded by WFB (local business development fund, EFRE)

10/2022-12/2022

Sikora in-house evaluation

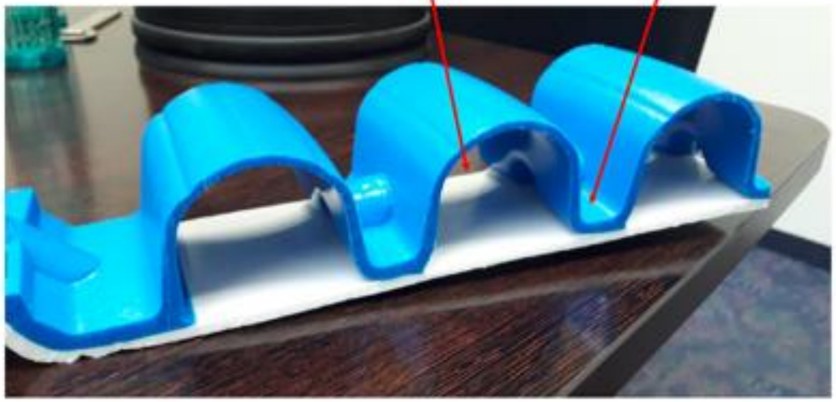
03/2023-07/2023

DLL development (WFB)

11/2023

product release

WELLROHRE

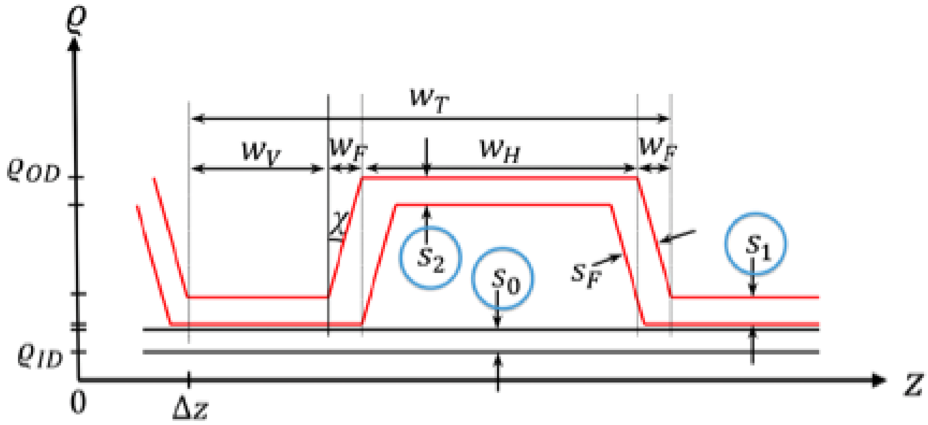


SIKORA
Technology To Perfection

Structure

- Innenhaut (weiß)
- Wellschicht (blau)

- Mono-Layer
- Single-Layer
- Multi-Layer

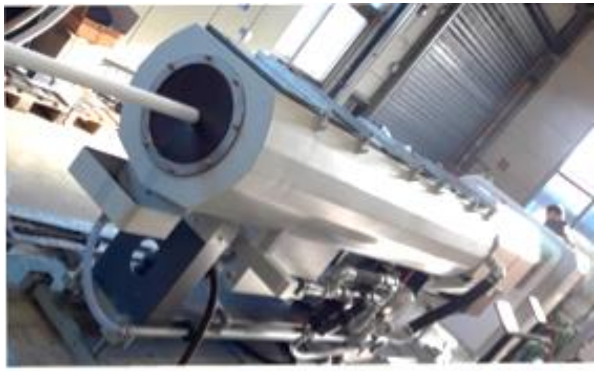


Parameters

- S0
- S2
- S0+S1

Online production monitoring

- Inprecise positioning
- Motion blur (< 9 m/s)



Kühlung & Trocknung

Abschneider

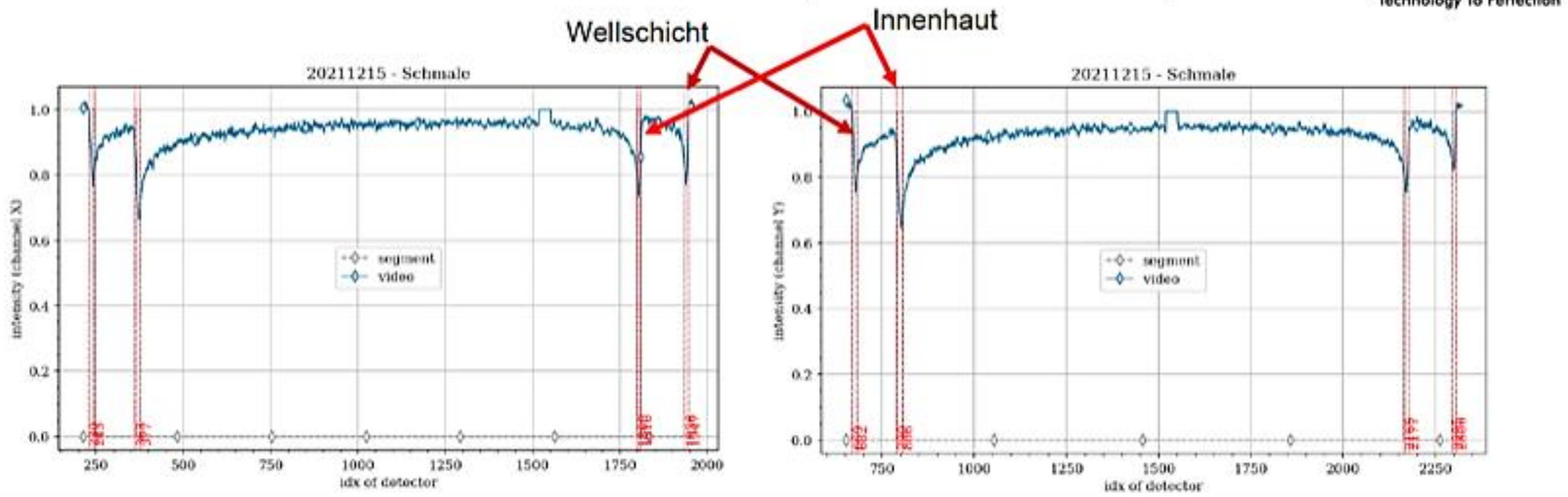
SIKORA X-RAY 6200 PRO

Basic X-Ray technology

- 2 directions
- Limited angle
- 3 μs for each scan
- Significant ray widening
- Low compute power

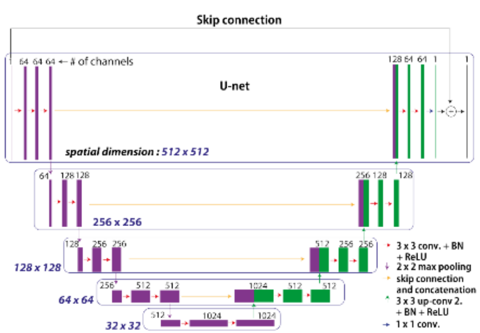
AI-MODELL ZUR FEATURE-DETEKTION (SEGMENTIERUNG)

SIKORA
Technology To Perfection



Math Theorie:
Range characterization for data calibration
Papers on equivariance

Machine Learning
Standard U-Net for segmentation



Implementation
Forward model for data generation
Dynamic link library

3-4 month mathematical research
(mainly reading papers)

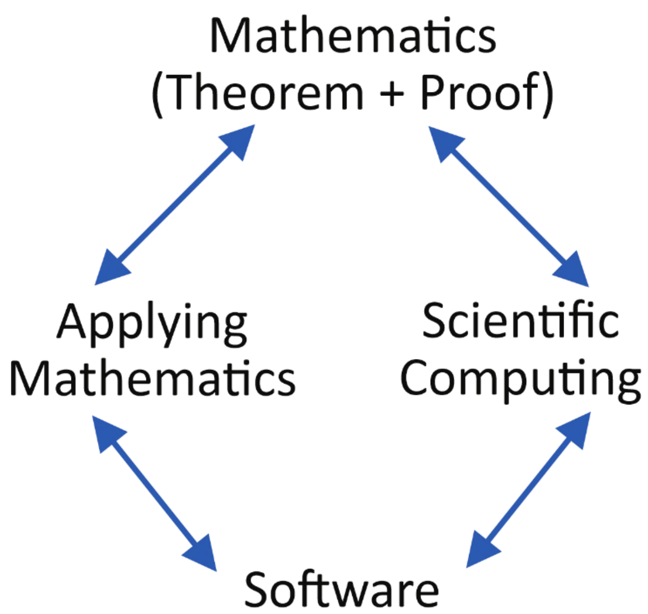
6 month math-inspired implementation
6 month testing, adaptation

6 month DLL programming

24 month at industrial partner

From Mathematical Research to Innovation

Novelty with relevance for society/economy



University

Research Papers

Patent

Open IP

Innovation Office

Law Office

Non-profit Qualification

Steinbeis

GmbH

Company

Problem solving

Patent

Proprietary IP

Software

Warranty

Updates

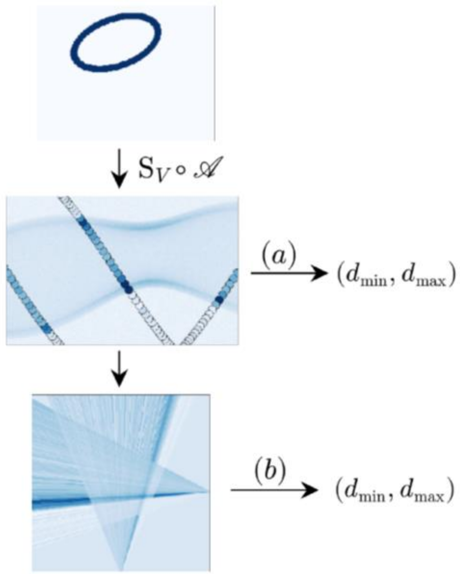
Time to market

TRL: Mathematics needed for TRL 1-2, Mathematicians are needed on all levels

Equivariant NN for Indirect Measurements

Numerical Insights

- experiments on regression and classification tasks, e.g. tube thickness measurements
- NN with equivariance towards the induced symmetries of rotation and translation on $\mathcal{Y}$ $\rightsquigarrow$ (a)
- comparison to classical group-equivariant CNN $\rightsquigarrow$ (b)
- first results:
  - improved generalization for small training data sets
  - increased robustness towards noise



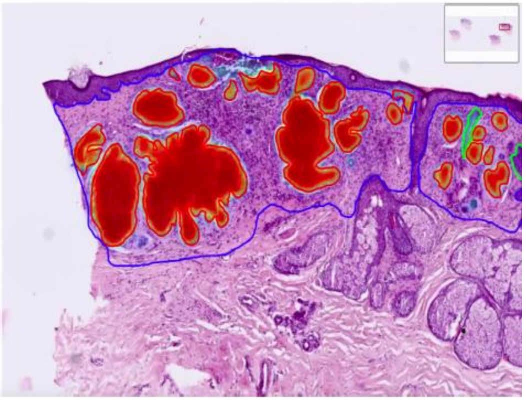
Beckmann, N. Heilenkötter. Equivariant Neural Networks for Indirect Measurements. Submitted for publication, 2023.

Outline

1. Neural networks
   - Network architecture
   - Data manifold
   - Regularization by architecture
2. An industrial CT application
3. PDE based applications
   - Limitations of present DL concepts
   - Topology optimization, Ariane

AI beyond Computer Vision

Industrial applications
VW, Ariane Airbus, Siemens, DB, ….



Digital pathology
1st clinial installation

CT tomography
Academic example

PDE models $$u_t + v^t \nabla u + \mathrm{div}((1 - D)\nabla u) = f$$

given measured u, determine (v, D, f)

Poisson problem

$$\Delta u = \lambda \quad \text{on} \quad \Omega \subset \mathbb{R}^2$$

$$u = g \quad \text{on} \quad \partial\Omega$$

Parameter-to-state operator

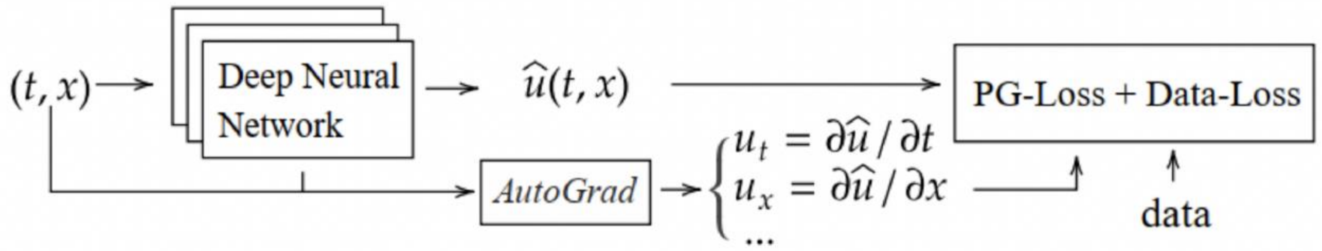
$$F : \mathcal{X} \to \mathcal{Y}$$

$$\lambda \mapsto u$$

- Forward problem: evaluation of F for a single parameter λ
- Parametric studies: evaluation of F for many parameters λ
- Inverse problem: given measured u^δ or Pu, solve $F(\lambda) \sim u^\delta$

Deep learning concepts for PDE forward solvers, e.g. PINN

- function evaluation, neural network $\Phi_\Theta(t,x)$



- operator evaluation, neural network $\Phi_\Theta(\lambda)$, e.g. PCANN

$$\begin{tikzcd} \mathcal{X} \arrow[r, "F_{\mathcal{X}}"] \arrow[d, "\Psi"'] & \mathbb{R}^{d_{\mathcal{X}}} \arrow[r, "G_{\mathcal{X}}"] \arrow[d, "\varphi"'] & \mathcal{X} \arrow[d, "\Psi"'] \\ \mathcal{Y} \arrow[r, "F_{\mathcal{Y}}"] & \mathbb{R}^{d_{\mathcal{Y}}} \arrow[r, "G_{\mathcal{Y}}"] & \mathcal{Y} \end{tikzcd}$$

$$\lambda = \sum c_k b_k \in \mathcal{X}, \quad \phi_\Theta : c \mapsto \tilde{c}, \quad u = \sum_\ell \tilde{c}_\ell \tilde{b}_\ell$$

Numerical Results – PCANN

Resolution	33×33	65×65	129×129	257×257	513×513
Poisson (μ_G)	4.02×10^{-2}	3.43×10^{-2}	4.62×10^{-2}	5.43×10^{-2}	4.32×10^{-2}
Darcy Flow (μ_P)	2.40×10^{-2}	1.63×10^{-2}	1.05×10^{-2}	1.08×10^{-2}	1.04×10^{-2}

Table: Relative L2 Errors for PCANN

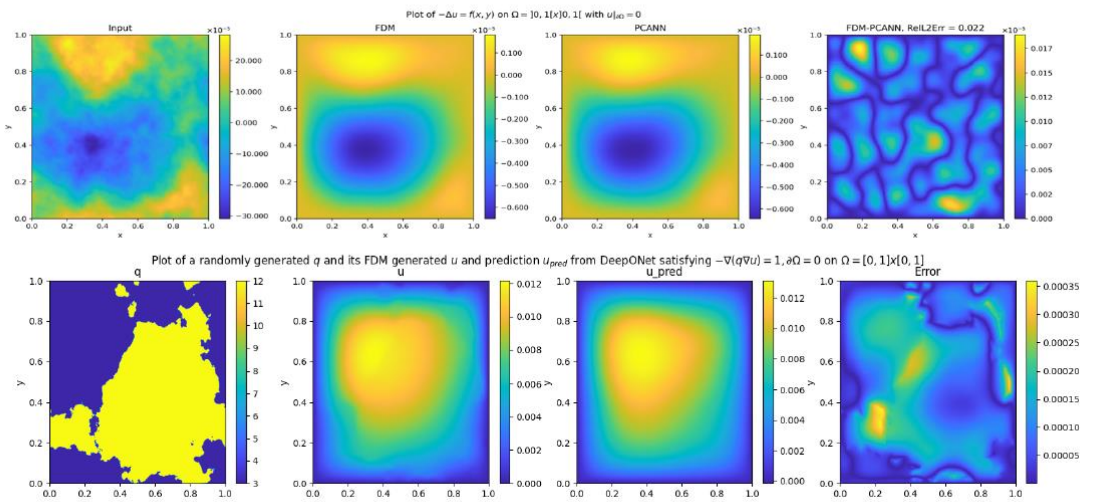


Figure: Parameter functions are Gaussian fields $\mu_G = \mathcal{N}(0, (-\Delta + 9I)^{-2})$ and $\mu_P = T_\sharp \mu_G$, details see K. Bhattacharya, A.M. Stuart, et al. , 2020

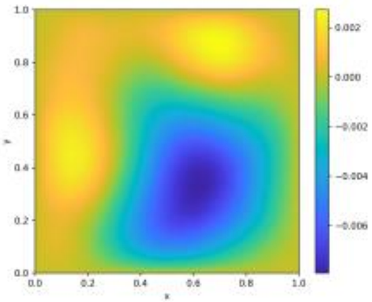
Numerical results inverse Darcy problem

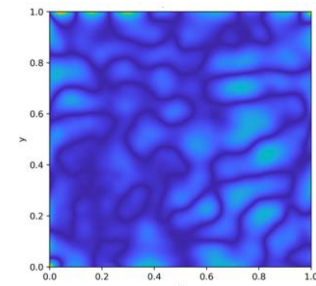
$$-\nabla \cdot (\lambda(s)\nabla u(s)) = f(s)$$
$$u(s) = 0$$

- Embedded Tikhonov
- Operator evaluation
- Very efficient
- Sufficient accuracy

- Parameter studies
- Rapid prototyping

	0%	0.1%	1%	5%	10%
PCANN	0.4684	0.4692	0.4680	0.4828	0.5148
PCALin	0.3157	0.3157	0.3159	0.3163	0.3183
FNO	0.1764	0.1764	0.1823	0.2143	0.2402
U-FNO	0.2140	0.2205	0.2017	0.2371	0.2734
MWT	0.1287	0.1250	0.1435	0.1880	0.2209
DeepONet	0.2484	0.2483	0.2469	0.2547	0.2691
PINO	0.1559	0.1549	0.1607	0.1932	0.2244
PI-DeepONet	0.2792	0.2801	0.2820	0.2822	0.2861

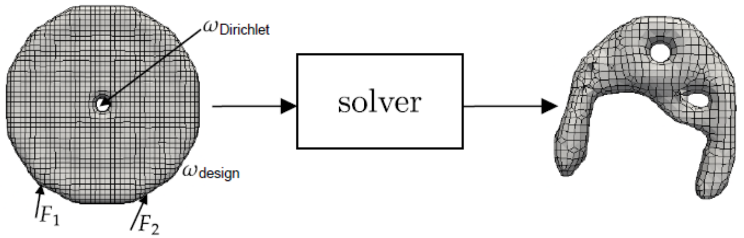




Deep Learning Methods for Partial Differential Equations and Related Parameter Identification Problems
Derick Nganyu Tanyu, Jianfeng Ning, Tom Freudenberg, Nick Heilenkötter, Andreas Rademacher, Uwe Iben, PM

Inverse design

Figure: Ariane 6 (Source ESA)



Problem:
forces F,
Dirichlet boundary $\omega_{\text{Dirichlet}}$,
design space ω_{Design},
material properties

Solution:
optimized structure
density distribution

$$\min_{\text{density}} \quad \text{compliance}$$

$$\text{subject to} \quad \text{physics,}$$
$$\text{weight constraints,}$$
$$\text{binary density.}$$

Iterative method + PDE solver = ⌛

$$\min_{\rho} L(\rho)$$

$$L(K(\rho)) = \langle f, u(\rho) \rangle + \lambda \|\rho\|_1$$

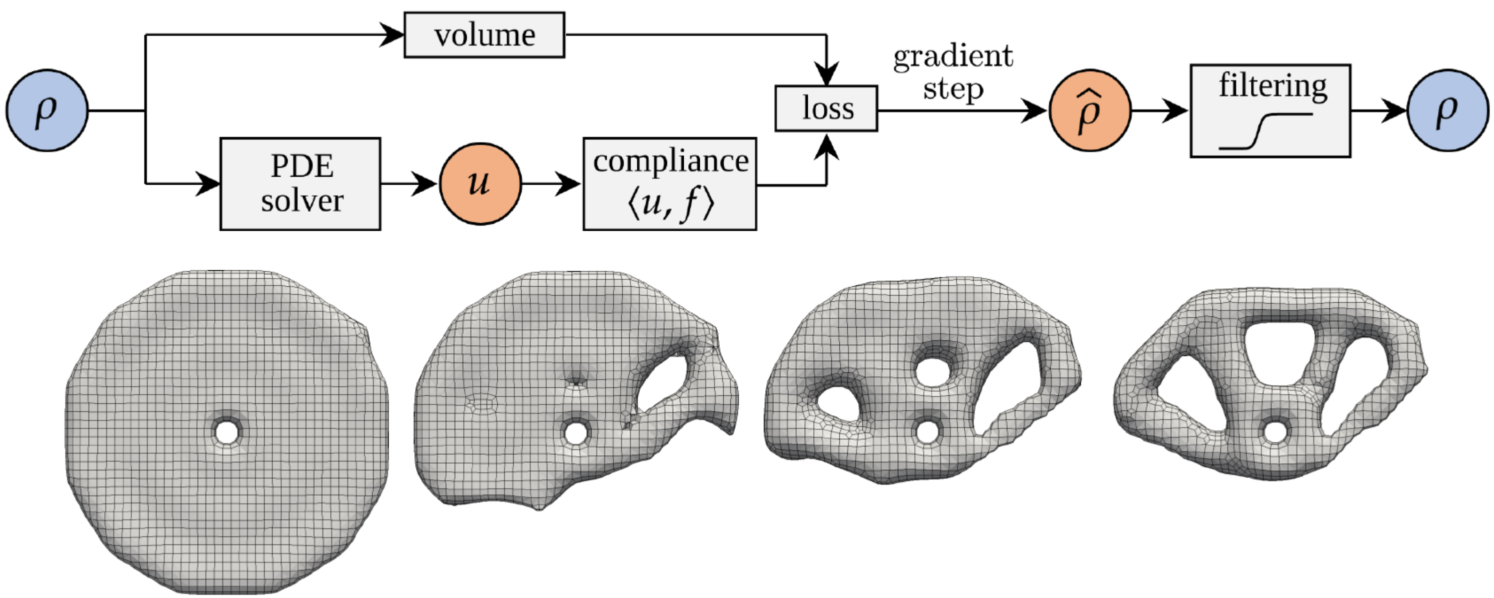
- Density: $\rho(\cdot) \in [0, 1]$
- Displacement: $u(\cdot) \in \mathbb{R}^3$
- Compliance: $\langle f, u(\rho) \rangle \in \mathbb{R}_{\geq 0}$

Linear elasticity:

$$\epsilon = \frac{1}{2}[\nabla u + \nabla u^T] \qquad \nabla \cdot \sigma = -f \qquad \sigma = C(\rho(\cdot)) : \epsilon$$

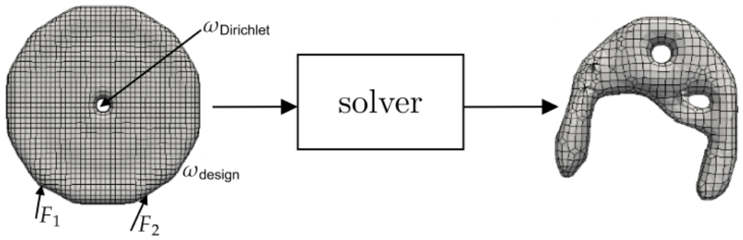
Iterative method + PDE solver = ⌛

SIMP method (sketch):

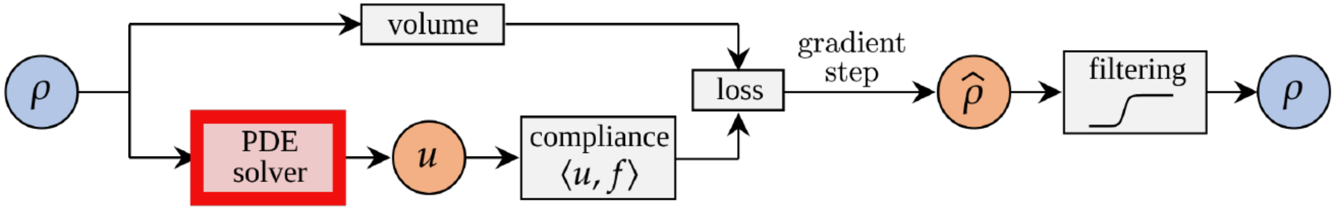


Deep learning approaches

1. End-to-end learning

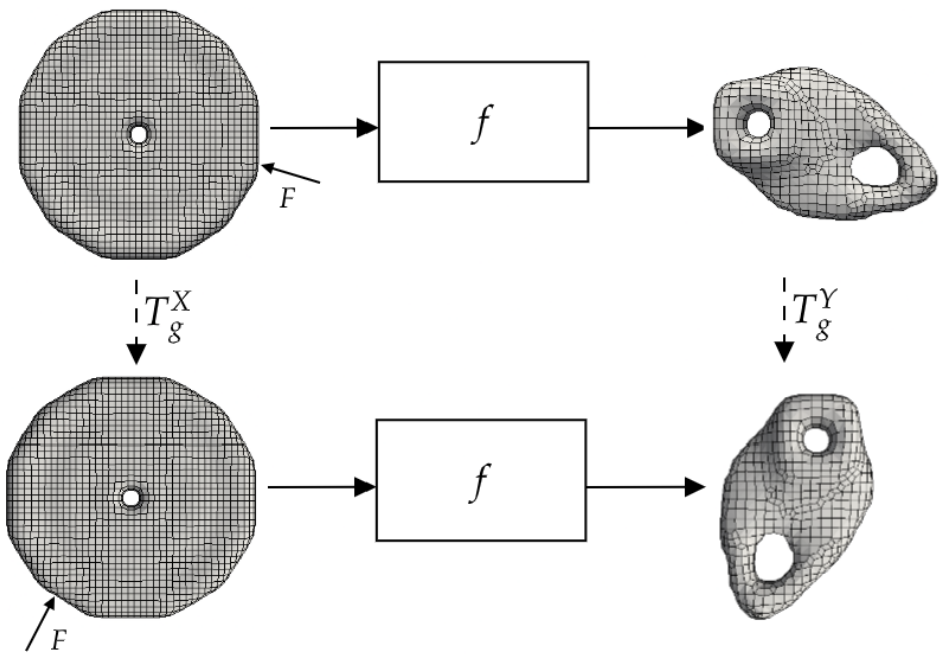


2. Learn substitute for PDE solver



3. Neural reparameterization, ∞-resolution

Equivariance



$$f : X \to Y$$

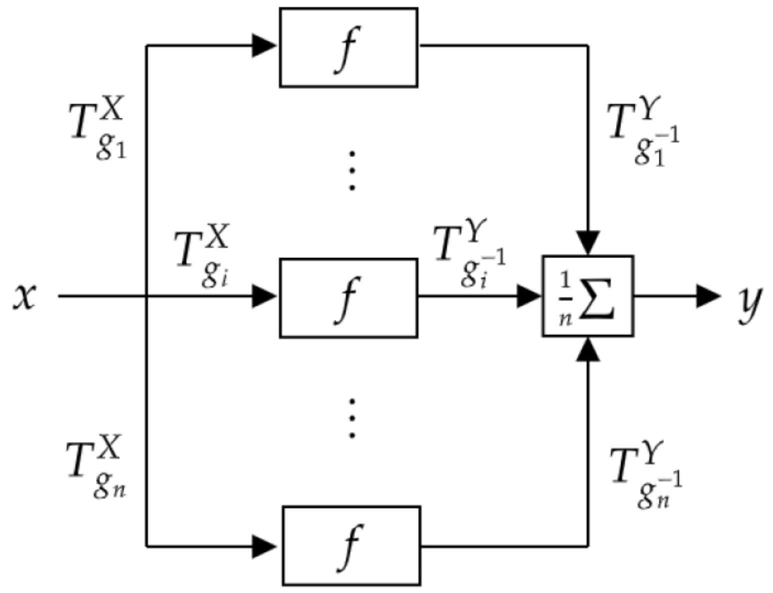
is *G-equivariant* [1] if

$$f\left(T_g^X x\right) = T_g^Y f(x)$$

$$\forall g \in G, x \in X.$$

[1] Cohen and Welling (2016) – Group equivariant convolutional networks

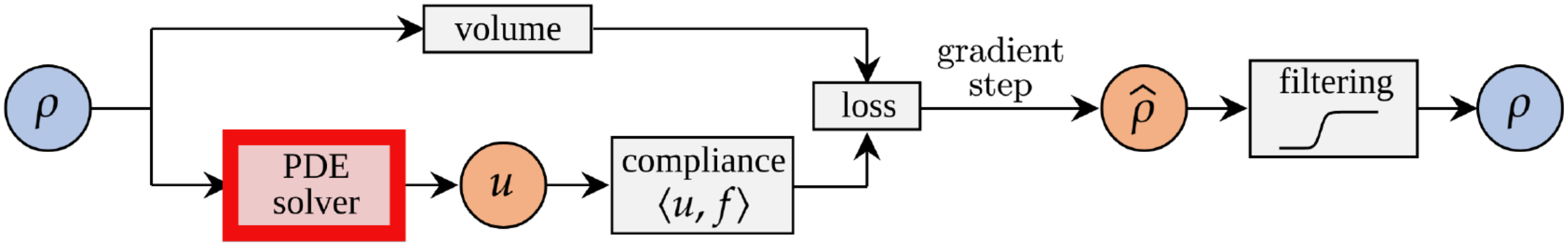
Group averaging[1]



$$F^f_G(x) := \frac{1}{|G|} \sum_{g \in G} T^Y_{g^{-1}} \left[f\left(T^X_g(x) \right) \right]$$

[1] Puny et al. (2021) – Frame averaging for invariant and equivariant network design

SIMP trajectory learning



- Learn PDE solver

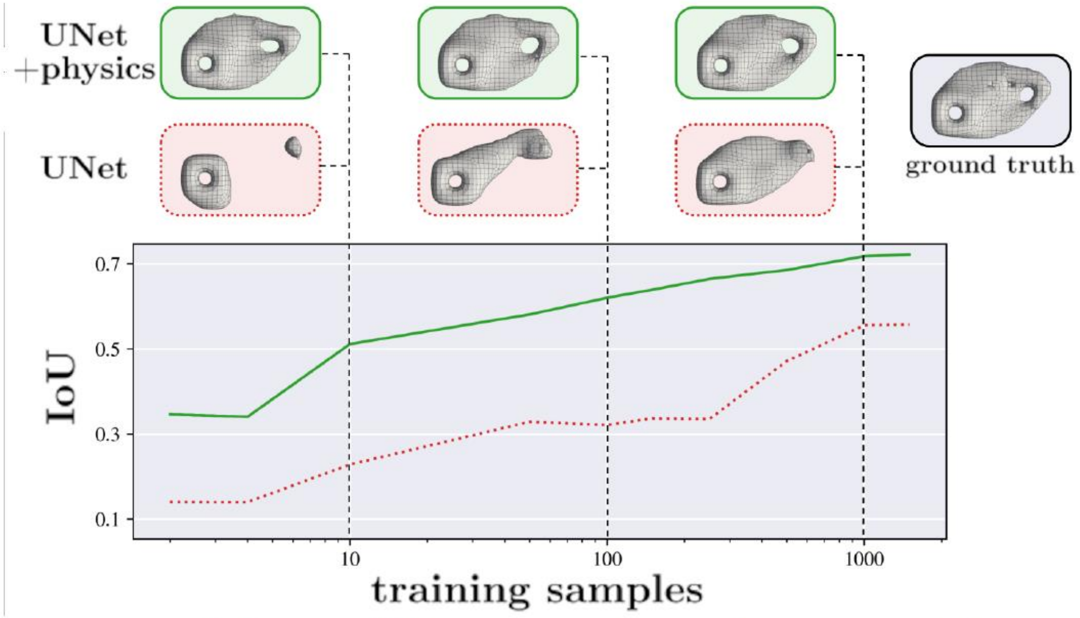
$$(\rho, f) \longmapsto u$$

- Supervised training via

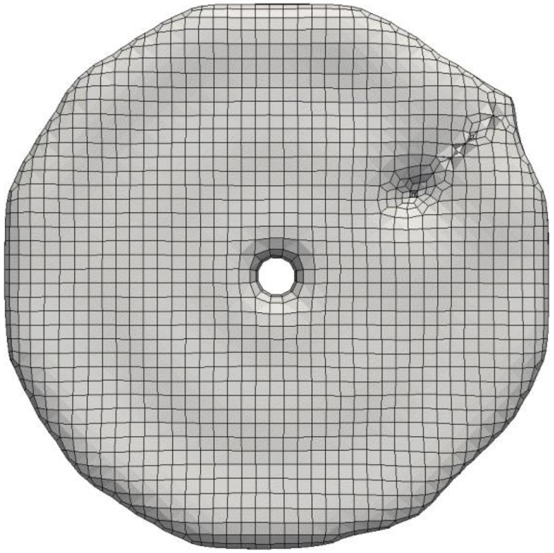
$$\mathcal{L}(u_{\text{pred}}, u) := \|u_{\text{pred}} - u\|_2^2 + \alpha \left\|\partial_\rho \langle u_{\text{pred}}, f\rangle - \partial_\rho \langle u, f\rangle\right\|_2^2$$

compliance

Results



- "Physics:" Preprocessing includes von Mises stresses as input channel + Equivariance

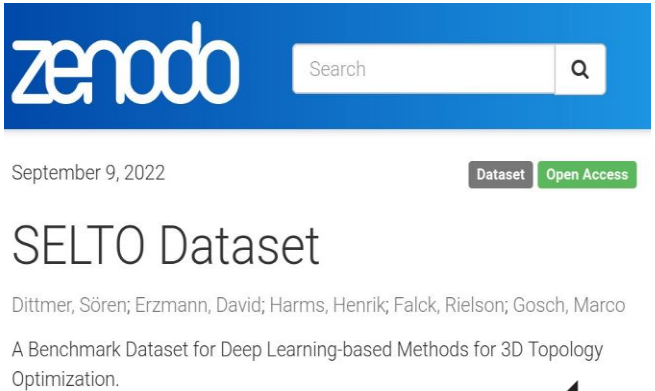


Inverse design of mechanical structures

Sören Dittmer, David Erzmann

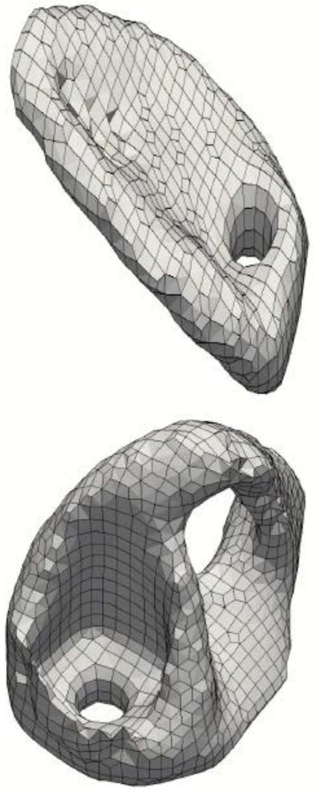
arianeGroup

Center for Industrial Mathematics



ELISE

DLR



Working group