Structure-Preserving Deep Learning

Matthias J. Ehrhardt

Institute for Mathematical Innovation, University of Bath, UK

August 13, 2021

Joint work with:
M. Benning (Queen Mary, UK), C. Etmann, C.-B. Schönlieb, F. Sherry (all Cambridge, UK), E. Celledoni, B. Owren (both NTNU, Norway), R. McLachlan (Massey, New Zealand)

The Leverhulme Trust

Engineering and Physical Sciences Research Council

THE FARADAY INSTITUTION

Main Messages of This Talk

- Concepts from **numerical analysis** offer insight in the structure of deep learning (optimal control, numerical differential equations, constrained optimisation, . . .).
- Imposing **structure** from numerical approaches can help to design neural networks with **solution guarantees** (stability, invertibility, equivariance, manifold structure, . . .).
- Many open problems and interesting opportunities for mathematicians.

Outline

- **Neural networks inspired by differential equations**
- **Equivariant neural networks**
- Invertible neural networks and normalising flows
- Deep Learning meets optimal control
- Structure-exploiting learning

Content is based on the following papers:

[1] E. Celledoni, M. J. Ehrhardt, C. Etmann, R.I. McLachlan, B. Owren, C. B. Schönlieb, F. Sherry, *Structure-preserving deep learning*, arXiv:2006.03364. European Journal for Applied Mathematics 2021

[2] M. Benning, E. Celledoni, M. J. Ehrhardt, B. Owren, C. B. Schönlieb, *Deep learning as optimal control problems: models and numerical methods*. arXiv:1904.05657. Journal of Computational Dynamics 2019

[3] E. Celledoni, M. J. Ehrhardt, C. Etmann, B. Owren, C.-B. Schönlieb, and F. Sherry, *Equivariant neural networks for inverse problems*, arXiv:2102.11504. Inverse Problems 2021

Notation: Neural Network

Define **neural network** $\Phi_\theta : X \to Y$ recursively: $\Phi_\theta(x) = z^K$

$$\begin{aligned} z^0 &= x \in X \\ z^{k+1} &= f^k(z^k, \theta^k), \quad k = 0, \dots, K-1 \end{aligned}$$

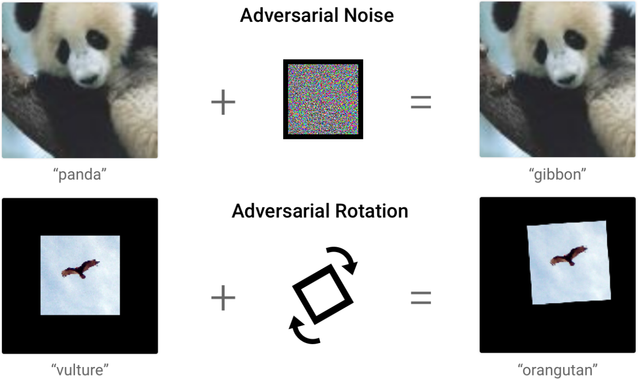
with generic **layers**

$$f^k : Z^k \times \Theta^k \to Z^{k+1}, \quad k = 0, \dots, K-1$$

- Classical, fully-connected layer defined by
$$\begin{aligned} f : \mathbb{R}^M \times (\mathbb{R}^{M' \times M} \times \mathbb{R}^{M'}) &\to \mathbb{R}^{M'} \\ (z, (A, b)) &\mapsto \sigma(Az + b), \end{aligned}$$
  where σ is an element-wise nonlinearity (ReLU, tanh etc.)
- A is often replaced by a convolutional operator
- **Training goal**: dataset $\{(x_n, y_n)\}_n$
$$\min_{\theta \in \Theta} \frac{1}{N} \sum_{n=1}^{N} L(\Phi_\theta(x_n), y_n) + R(\theta)$$

Deep Learning and Robustness

- Deep learning often is not robust (e.g. noise, rotations, ...)



https://ai.googleblog.com/2018/09/introducing-unrestricted-adversarial.html

- Data augmentation ...
- **This talk**: Design deep learning architectures with mathematical guarantees (e.g. stability, equivariance, ...)

Neural networks inspired by differential equations

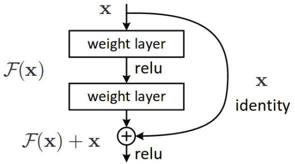
Residual networks as discretised ODEs

- "Standard" Neural Networks

$$z^{k+1} = \sigma(A^k z^k + b^k)$$

- Deep Residual Neural Networks (ResNet) He, Zhang, Ren, Sun 2015 (> 85000 citations on GoogleScholar)

$$z^{k+1} = z^k + \Delta t \, \sigma(A^k z^k + b^k)$$



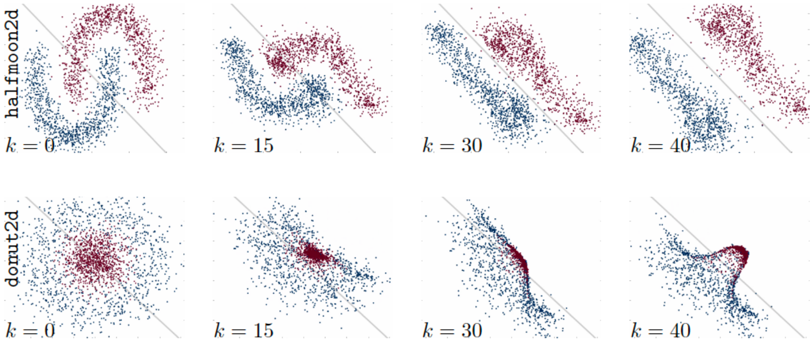
ResNet is Forward Euler discretization $\dot{z}(t) \approx \frac{z(t+\Delta t) - z(t)}{\Delta t}$ of

$$\dot{z}(t) = \sigma(A(t)z(t) + b(t)), \quad t \in [0, T]$$

with continuous-time mappings A, b. $z^k := z(k\Delta t)$ ...

Haber and Ruthotto 2018, Li et al. 2018, Benning et al. 2019, ...

ResNet in action



Interpretation as discrete optimal control

The **deep learning problem** can be seen as the discretization of

Optimal control problem

$$\min_{\theta} \frac{1}{N} \sum_{n=1}^{N} L(z_n(T), y_n)$$

subject to

$$\dot{z}_n = f(z_n, \theta), \quad z_n(0) = x_n \in X.$$

Why is the optimal control point of view useful:

- it states the deep learning problem in two lines
- can be used to create new architectures
- continuous models are useful simplifications of reality, amenable for analysis
- what ODE properties carry over to discrete neural networks?

Haber and Ruthotto 2017; Li, Chen, Tai, E 2018

Notions of Stability - What makes sense?

Notions of **ODE stability**:

- Stability of equilibrium points (e.g. Lyapunov/asymptotic stability of autonomous systems)
- How does $z(t)$ change if initial value $x = z(0)$ changes?
- Statements for all $t \in [0, \infty)$ or just $t \in [0, T]$?

Notions of **NN stability**:

- (Uniform) continuity of output w.r.t. input of network: "Always" fulfilled with standard architectures but constants can be arbitrary large
- Enforcing a specific e.g. Lipschitz constant, i.e. "Train this architecture and a certain stability is guaranteed".

ODE Stability 1

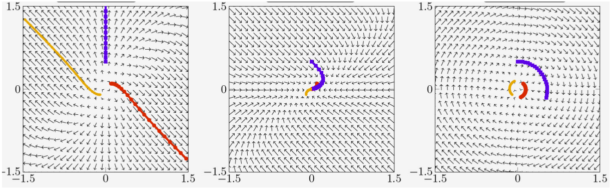
Theorem (very old): The **autonomous** ODE $\dot{z} = f(z)$ is asymptotically stable if the real parts of the eigenvalues of the Jacobian Df are non-positive.

Corollary: Let $\dot{\sigma} \geq 0$. Then forward propagation is **asymptotically stable** if $\mathrm{Re}(\lambda(A)) \leq 0$.

- Examples. $\sigma(y) = y, b = 0$

$$A_+ = \begin{pmatrix} 2 & -2 \\ 0 & 2 \end{pmatrix}, \quad A_- = \begin{pmatrix} -2 & 0 \\ 2 & -2 \end{pmatrix}, \quad A_0 = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

$$\lambda(A_+) = (2, 2), \quad \lambda(A_-) = (-2, -2), \quad \lambda(A_0) = (i, -i)$$

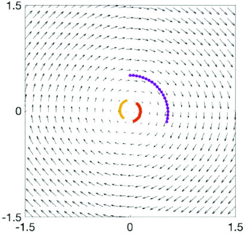


Haber and Ruthotto 2018

New **Unconditionally Stable** Architectures

- ResNet with **antisymmetric** transformation matrix

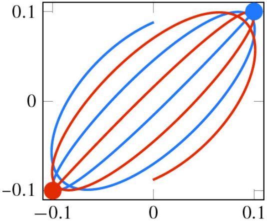
$$\dot{z} = \sigma\left((A - A^T)z + b\right)$$



- **Hamiltonian inspired Network**: ResNet with auxiliary variable and antisymmetric matrix

$$\begin{pmatrix} \dot{z} \\ \dot{w} \end{pmatrix} = \sigma\left(\begin{pmatrix} 0 & A \\ -A^T & 0 \end{pmatrix}\begin{pmatrix} z \\ w \end{pmatrix} + b\right)$$

$$z(0) = z_0, \quad w(0) = 0$$



Haber and Ruthotto 2018

Problem: this statement is **only true** for autonomous systems!

If the vector-field f depends on time, then similar statements are true but the theory is **rather weak**.

ODE stability 2

Consider $\Phi(z) = z(T)$ with z solving $\dot{z}(t) = f_t(z(t)), t \in [0, T]$

Definition We call a neural network Φ **stable** if there exists $C > 0$ such that for all u, v we have

$$\|\Phi(u) - \Phi(v)\| \leq C\|u - v\|.$$

- With **Lipschitz continuity** of f_t:

  e.g. $f_t(u) = \sigma(A(t)u + b(t))$ with σ being S-Lipschitz and A continuous

  $$C = \exp(T \cdot L) \qquad \left(= \exp(T \cdot S \max_{t \in [0,T]} \|A(t)\|)\right)$$

- With **"one-sided" Lipschitz continuity** of f_t:

  $$\langle f_t(u) - f_t(v), u - v \rangle \leq \nu \|u - v\|^2, \quad \nu \in \mathbb{R}$$

  If f is L-Lipschitz, then f is "one-sided" Lipschitz with $\nu = L$

  $$C = \exp(T \cdot \nu)$$

Celledoni et al. 2021, Zhang and Schaeffer 2020

Sufficient Conditions for Stability

Recall, "one-sided" Lipschitz continuity of f_t

$$\langle f_t(u) - f_t(v), u - v \rangle \leq \nu \|u - v\|^2 \tag{OL}$$

Theorem Celledoni et al. 2021

- Let V_t be twice differentiable and convex. Then $f_t(u) = -\nabla V_t(u)$ satisfies (OL) for some $\nu \leq 0$.
- Let $0 \leq \sigma' \leq 1$ almost everywhere. Then

$$f_t(u) = -A(t)^* \sigma(A(t)u + b(t))$$

satisfies (OL) with $-\mu_*^2 \leq \nu \leq 0$ where $\mu_* := \inf_t \mu(t)$ and $\mu(t)$ is the smallest singular value of $A(t)$.

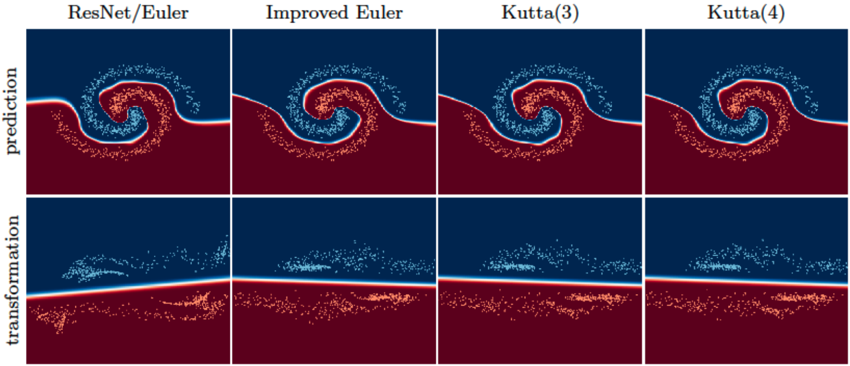
- Note that this does not require smoothness in time of A and b
- Discretized systems (e.g. Runge-Kutta methods)
  "Circle contractivity" Dahlquist 1979

$$\langle f_t(u) - f_t(v), u - v \rangle \leq \nu \|f_t(u) - f_t(v)\|^2$$

Examples: Different Runge–Kutta methods

$$
\begin{array}{c|c}
0 & \\
\hline
 & 1
\end{array}
\qquad
\begin{array}{c|cc}
0 & & \\
1 & 1 & \\
\hline
 & \frac{1}{2} & \frac{1}{2}
\end{array}
\qquad
\begin{array}{c|ccc}
0 & & & \\
\frac{1}{2} & \frac{1}{2} & & \\
1 & -1 & 2 & \\
\hline
 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6}
\end{array}
\qquad
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
\frac{1}{2} & 0 & \frac{1}{2} & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6}
\end{array}
$$

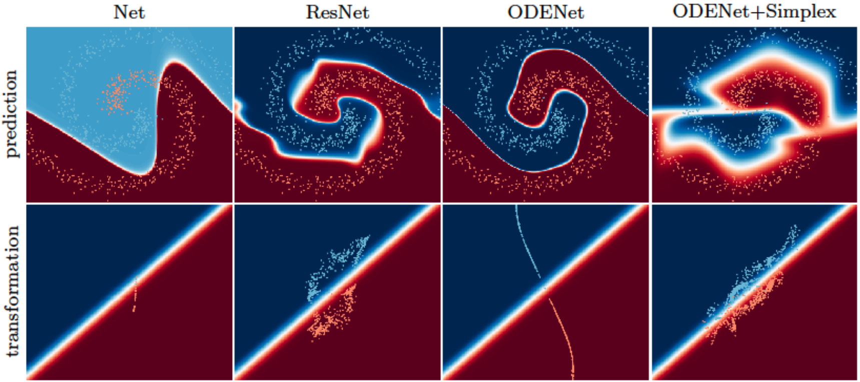
TABLE 1. Four explicit Runge–Kutta methods: ResNet/Euler, Improved Euler, Kutta(3) and Kutta(4).



Examples: Learn time steps

$$z^{k+1} = z^k + \Delta t^k \sigma(A^k z^k + b^k)$$

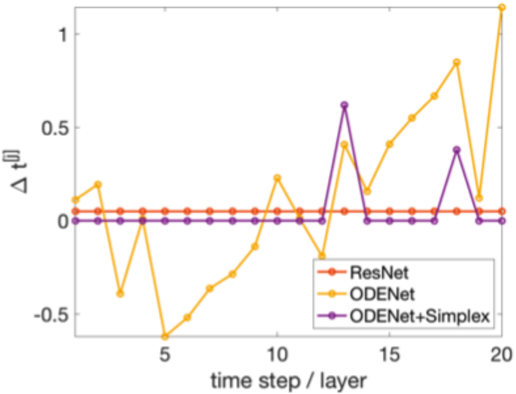
- ResNet: Choose $\Delta t^k = T/K$
- ODENet: Estimate $(\Delta t^k, A^k, b^k)$
- Simplex constraint: $\Delta t^k \geq 0, \sum_k \Delta t^k = T$



Examples: Learn time steps

$$z^{k+1} = z^k + \Delta t^k \sigma(A^k z^k + b^k)$$

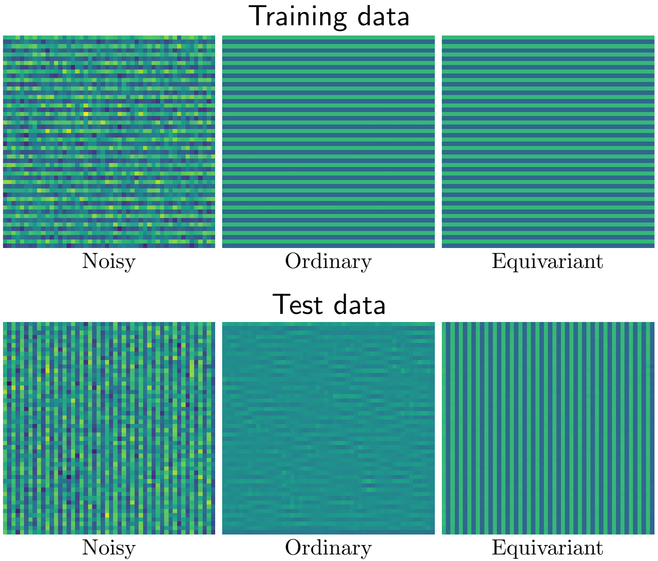
- ResNet: Choose $\Delta t^k = T/K$
- ODENet: Estimate $(\Delta t^k, A^k, b^k)$
- Simplex constraint: $\Delta t^k \geq 0, \sum_k \Delta t^k = T$



Equivariant neural networks

What happens when images are rotated?

$$\Phi(y) = x$$



Equivariance and Invariance

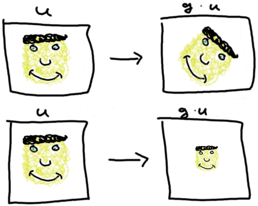
Definition: Group G "acts" on spaces X and Y denoted by $g_X \circ u$ and $g_Y \circ v$. We call $\Phi : X \to Y$ G-equivariant if for all $g \in G, u \in X$

$$\Phi(g_X \circ u) = g_Y \circ \Phi(u).$$

If Φ is G-equivariant and G acts trivially on Y, then we call Φ G-invariant, i.e. for all $u \in X$ and $g \in G$ $\Phi(g_X \circ u) = \Phi(u)$.

Examples of interesting groups:

- translations
- rotations
- scaling
- roto-translations $\overline{G} : g_X = (R, t)$
  $(g_X \circ u)(x) = \pi_X(R)u(R^{-1}x + t)$



Equivariant neural networks have been studied a lot for segmentation, classification, denoising etc

Cohen and Welling '16, Dieleman et al. '16, Worall et al. '17, Bekkers et al. '18, Weiler and Cesa '19, Sosnovik et al. '19, Worall and Welling '19, Cohen et al. '19 ...

How to get Equivariant Networks?

Proposition The following are equivariant:

- the **composition** of equivariant operators
- the **sum** of equivariant operators
- the **identity operator**

Proposition (linearity) There are non-trivial $\overline{G}$-equivariant linear operators.

Proposition (bias) Let $\Phi : X \to X, (\Phi u)(x) = u(x) + b(x)$. For any group G, Φ is equivariant if b is invariant, i.e. $g \cdot b = b$.

Proposition (nonlinearity) There are $\overline{G}$-equivariant nonlinearities.

We can construct $\overline{G}$-equivariant neural networks in the usual way:

- layers $\Phi = \Phi_n \circ \cdots \circ \Phi_1$
- $\Phi(u) = \sigma(Au + b)$
- ResNet $\Phi(u) = u + \sigma(Au + b)$

Equivalent Linear Functions ($\pi_X \equiv id$)

In a nutshell: Linear $\overline{G}$-equivariant operators are convolutions with a kernel satisfying an additional constraint.

Theorem paraphrasing e.g. Weiler and Cesa 2019

Let X, Y be function spaces, e.g. $X = L^2(\mathbb{R}^n, \mathbb{R}^m)$, $Y = L^2(\mathbb{R}^n, \mathbb{R}^M)$. The linear operator $A : X \to Y$,

$$Au(x) = \int K(x, y)u(y)dy$$

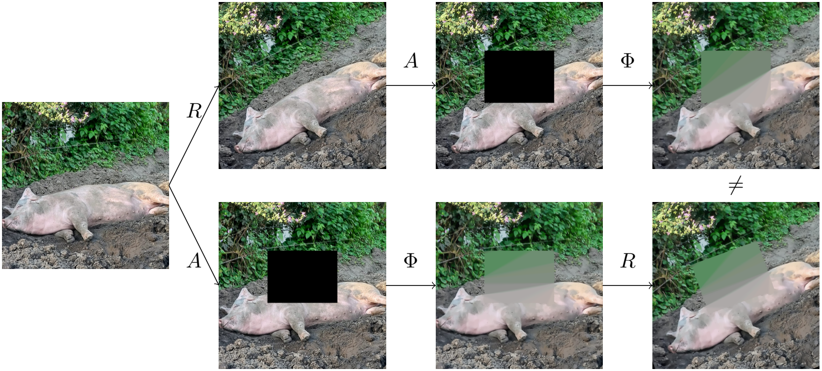
with $K : \mathbb{R}^n \to \mathbb{R}^{M \times m}$ is $\overline{G}$-**equivariant** iff there is a k such that

$$K(x, y) = k(x - y)$$

and k is rotational invariant, i.e. for all $R \in H$, $x \in \mathbb{R}^n$: $k(Rx) = k(x)$.

Equivariance and Inverse Problems

- inverse problem $Ax = y$, solution operator: $\Phi : Y \to X$
- **Hope** $\Phi \circ A$ is equivariant, e.g. $R_\theta \circ \Phi \circ A = \Phi \circ A \circ R_\theta$
- $\Phi \circ A$ is **not generally equivariant**
- Example: TV and inpainting



Proximal Operators and Equivariance

$$\operatorname{prox}_J(z) := \arg\min_x \left\{ \frac{1}{2}\|x - z\|^2 + J(x) \right\}$$

Theorem Celledoni et al. 2021

Let g_X be unitary, J G-invariant and prox_J be well-defined and single-valued. Then prox_J is **equivariant**.

- Proof does **generalize** to variatial regularization with squared L^2-datafit **if** A **is equivariant**
- For **example** the total variation (and higher order variants) is invariant to rigid motion

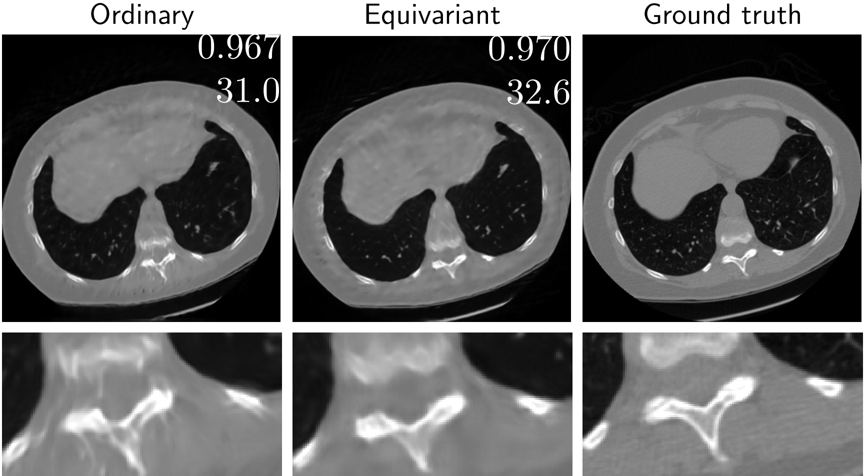
This theorem motivates iterative unrolling for image reconstruction with equivariant neural networks in place of the prox of a variational regulariser!

CT Results

Equivariant = roto-translations; Ordinary = translations

Equivariant improves upon Ordinary:

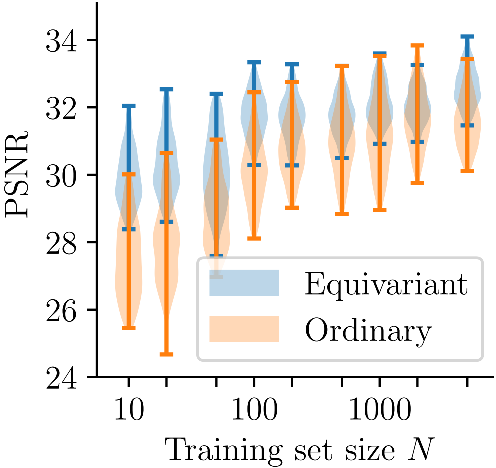
- **higher** SSIM and PSNR
- **fewer** artefacts and **finer** details



CT Results

Equivariant = roto-translations; Ordinary = translations

- Equivariant improves upon Ordinary on **small** training sets



Take Away Messages

- **Continuum modelling** of neural networks opens the toolbox of mathematical and numerical analysis
- **Connections** of deep learning to ODEs, optimal control, group theory ...
- Design of neural networks with certain **structure**: stability, equivariance
- **Many open questions** where mathematicians can help

E. Celledoni, M. J. Ehrhardt, C. Etmann, R.I. McLachlan, B. Owren, C. B. Schönlieb, F. Sherry, *Structure preserving deep learning*, arXiv:2006.03364, EJAM 2021