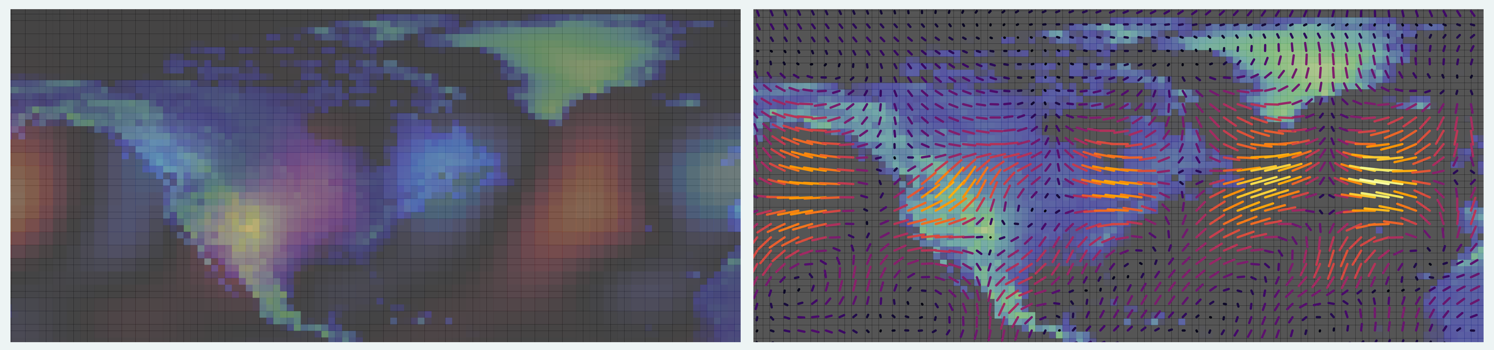
Clifford Neural Layers for PDE Modeling

Johannes Brandstetter [1] Rianne van den Berg [1] Max Welling [1] Jayesh K. Gupta [2]

[1]Microsoft Research AI4Science [2]Microsoft Autonomous Systems and Robotics Research

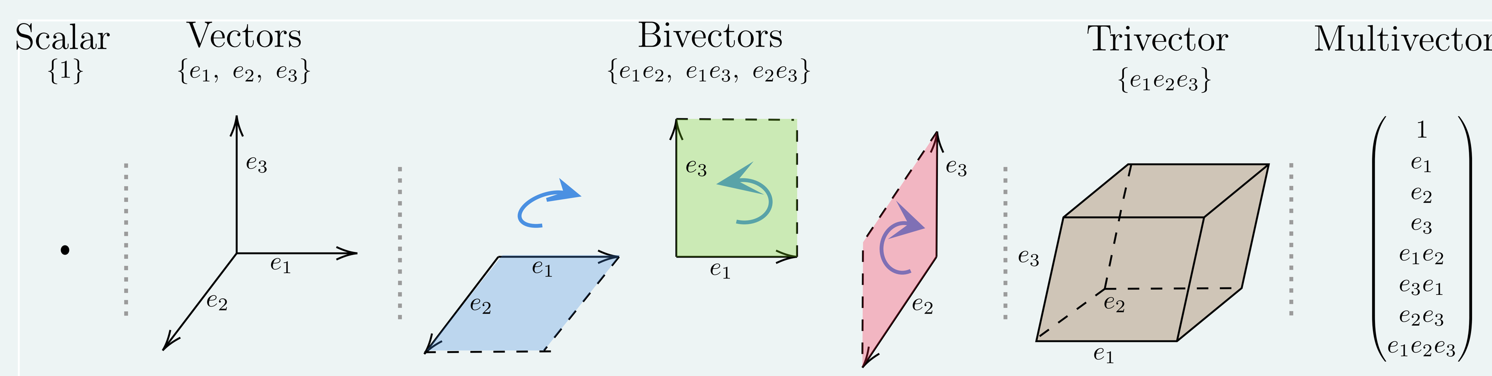
PDEs - the language of science / simulation

- We model Partial Differential Equations (PDEs) of the form $\partial_t \mathbf{u} = F(t, \mathbf{x}, \mathbf{u}, \partial_{\mathbf{x}}\mathbf{u}, \partial_{\mathbf{xx}}\mathbf{u}, \ldots)$.
- Often, data shows relation between different fields and field components, which standard neural surrogates do not take into account.



Clifford algebras have multivector structure

- The elements of a Clifford algebra are called **multivectors**, containing elements of subspaces, i.e. scalars, vectors, bivectors, trivectors etc.



- A real Clifford algebra $Cl_{p,q}(\mathbb{R})$ with **signature** (p, q) is generated through relations that define how the bilinear product of the algebra operates on the basis elements of $\mathbb{R}^n$:

$$e_i^2 = +1 \quad \text{for } 1 \le i \le p\,,$$
$$e_j^2 = -1 \quad \text{for } p < j \le p+q\,,$$
$$e_ie_j = -e_je_i \quad \text{for } i \ne j\,.$$

- $Cl_{0,1}$ with base vectors $(1, e_1)$ is isomorph to the complex numbers with $e_1 = i$, $Cl_{0,2}$ with base vectors $(1, e_1, e_2, e_1e_2)$ is isomorph to the quaternions with $e_1 = i$, $e_2 = j$, $e_1e_2 = k$.
- The vector space G^{p+q} of a Clifford algebra $Cl_{p,q}$ can be written as the direct sum of all of these subspaces:

$$G^{p+q} = M_0 \oplus M_1 \oplus \ldots \oplus M_{p+q}\,.$$

Dual representation, geometric product

- The **dual** $\boldsymbol{a}^*$ of a multivector $\boldsymbol{a}$ is defined as:

$$\boldsymbol{a}^* = \boldsymbol{a}i_{p+q}\,,$$

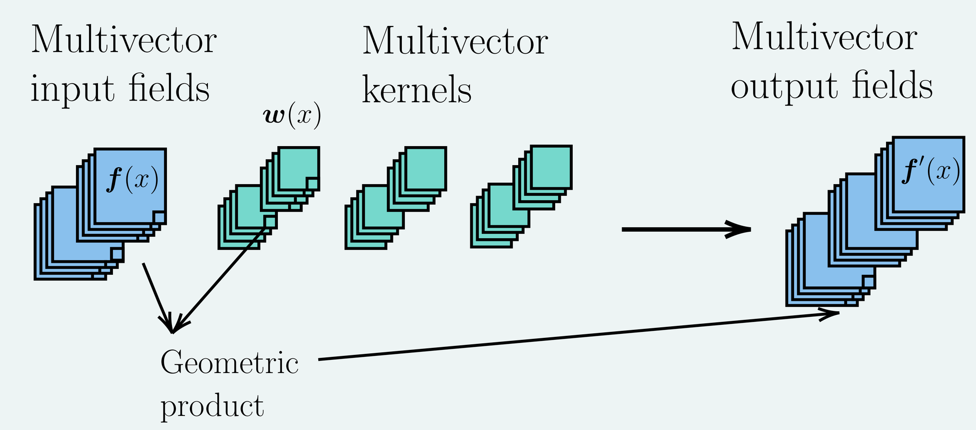
where i_{p+q} represents the respective **pseudoscalar** (highest grade object) of the Clifford algebra.
- **Geometric product, example 2D**: The 4-dimensional vector spaces of 2D Clifford algebras have the basis vectors $\{1, e_1, e_2, e_1e_2\}$ where e_1, e_2 square to $+1$ for $Cl_{2,0}(\mathbb{R})$ and to -1 for $Cl_{0,2}(\mathbb{R})$. For $Cl_{2,0}(\mathbb{R})$, the bilinear geometric product of two multivectors $\boldsymbol{a} = a_0 + a_1e_1 + a_2e_2 + a_{12}e_1e_2$ and $\boldsymbol{b} = b_0 + b_1e_1 + b_2e_2 + b_{12}e_1e_2$ is given by:

$$\boldsymbol{ab} = (a_0b_0 + a_1b_1 + a_2b_2 - a_{12}b_{12})\mathbf{1} + (a_0b_1 + a_1b_0 - a_2b_{12} + a_{12}b_2)\mathbf{e_1} + (a_0b_2 + a_1b_{12} + a_2b_0 - a_{12}b_1)\mathbf{e_2} + (a_0b_{12} + a_1b_2 - a_2b_1 + a_{12}b_0)\mathbf{e_1e_2}\,.$$

Clifford convolution

- Clifford CNN layers convolve **multivector feature maps** $\boldsymbol{f}: \mathbb{Z}^2 \to (G^2)^{c_{\text{in}}}$ with a set of c_{out} **multivector filters** $\{\boldsymbol{w}^i\}_{i=1}^{c_{\text{out}}} : \mathbb{Z}^2 \to (G^2)^{c_{\text{in}}}$:

$$[\boldsymbol{f} \star \boldsymbol{w}^i](x) = \sum_{y\in\mathbb{Z}^2}\sum_{j=1}^{c_{\text{in}}} \underbrace{\boldsymbol{f}^j(y)\boldsymbol{w}^{i,j}(y-x)}_{\boldsymbol{f}^j\boldsymbol{w}^{i,j}\,:\,G^2\times G^2 \to G^2}\,.$$



- **Rotational Clifford CNN layers (2D)**: Filters $\{\boldsymbol{w}^i_{\text{rot}}\}_{i=1}^{c_{\text{out}}} : \mathbb{Z}^2 \to (G^2)^{c_{\text{in}}}$ act on the feature map $\boldsymbol{f}^j$ through a rotational (quaternion) transformation $\boldsymbol{R}^{i,j}(w^{i,j}_{\text{rot},0}, w^{i,j}_{\text{rot},1}, w^{i,j}_{\text{rot},2}, w^{i,j}_{\text{rot},12})$ of vector and bivector parts $\boldsymbol{f}: \mathbb{Z}^2 \to (G^2)^{c_{\text{in}}}$, and an additional scalar response of the multivector filters:

$$[\boldsymbol{f} \star \boldsymbol{w}^i_{\text{rot}}](x) = \sum_{y\in\mathbb{Z}^2}\sum_{j=1}^{c_{\text{in}}} \underbrace{\left[\boldsymbol{f}^j(y)\boldsymbol{w}^{i,j}_{\text{rot}}(y-x)\right]_0}_{\text{scalar output}} + \underbrace{\boldsymbol{R}^{i,j}(y-x)}_{\text{quaternion rotation matrix}} \cdot \begin{pmatrix} f_1^j(y) \\ f_2^j(y) \\ f_{12}^j(y) \end{pmatrix},$$

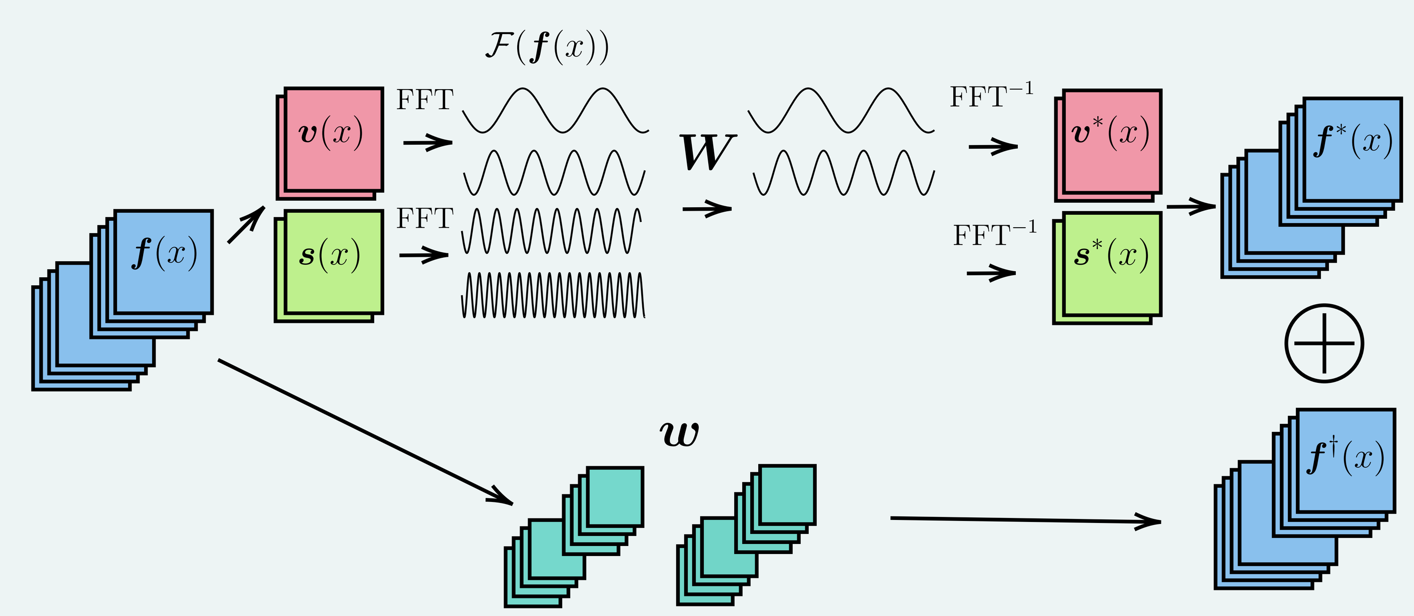
Clifford Fourier transform

- Using the duality relation $\boldsymbol{a}^* = \boldsymbol{a}i_{p+q}$, the 2-dimensional dual pairs of the base vectors are $1 \leftrightarrow e_1e_2$ and $e_1 \leftrightarrow e_2$:

$$\boldsymbol{a} = a_0 + a_1e_1 + a_2e_2 + a_{12}e_1e_2 = 1\underbrace{\left(a_0 + a_{12}i_2\right)}_{\text{spinor part}} + e_1\underbrace{\left(a_1 + a_2i_2\right)}_{\text{vector part}}\,.$$

- This duality allows us to define **2D Clifford Fourier transform layers**:

$$\mathcal{F}\{\boldsymbol{f}\}(\xi) = \frac{1}{2\pi}\int_{\mathbb{R}_2} \boldsymbol{f}(x)e^{-2\pi i_2\langle x,\xi\rangle}\,dx\,,\ \forall \xi \in \mathbb{R}^2\,,$$
$$= \frac{1}{2\pi}\int_{\mathbb{R}_2}\left[1\underbrace{\left(f_0(x) + f_{12}(x)i_2\right)}_{\text{spinor part}} + e_1\underbrace{\left(f_1(x) + f_2(x)i_2\right)}_{\text{vector part}}\right]e^{-2\pi i_2\langle x,\xi\rangle}\,dx$$
$$= 1\left[\mathcal{F}\left(f_0(x) + f_{12}(x)i_2\right)(\xi)\right] + e_1\left[\mathcal{F}\left(f_1(x) + f_2(x)i_2\right)(\xi)\right].$$



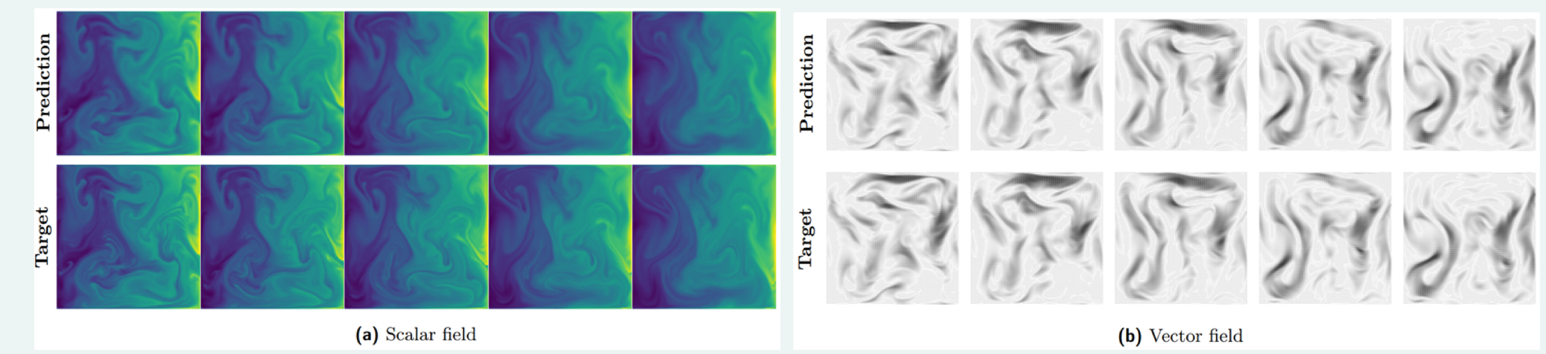
- Point-wise geometric product in the Clifford Fourier space.
- **3-dimensional Clifford Fourier transforms** follow a similar elegant construction, where we apply four separate Fourier transforms to

$$\boldsymbol{f}_0(x) = f_0(x) + f_{123}(x)i_3 \qquad \boldsymbol{f}_1(x) = f_1(x) + f_{23}(x)i_3$$
$$\boldsymbol{f}_2(x) = f_2(x) + f_{31}(x)i_3 \qquad \boldsymbol{f}_3(x) = f_3(x) + f_{12}(x)i_3\,.$$

Experiments

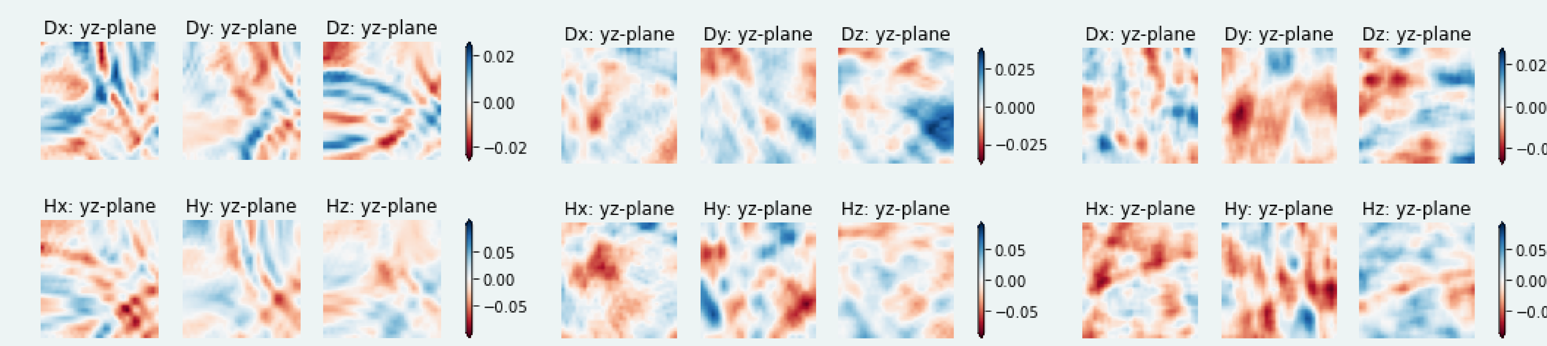
- We look at **Navier-Stokes** and **shallow water** equations which are coupled (non-linear) 2-dimensional PDEs (scalar + vector field):

$$\text{Navier-Stokes:}\quad \frac{\partial v}{\partial t} = -\underbrace{v\cdot\nabla v}_{\text{convection}} + \underbrace{\mu\nabla^2 v}_{\text{diffusion}} - \underbrace{\nabla p}_{\text{pressure gradient}} + \underbrace{f}_{\text{external force}}\,,\quad \nabla\cdot v = 0\,.$$



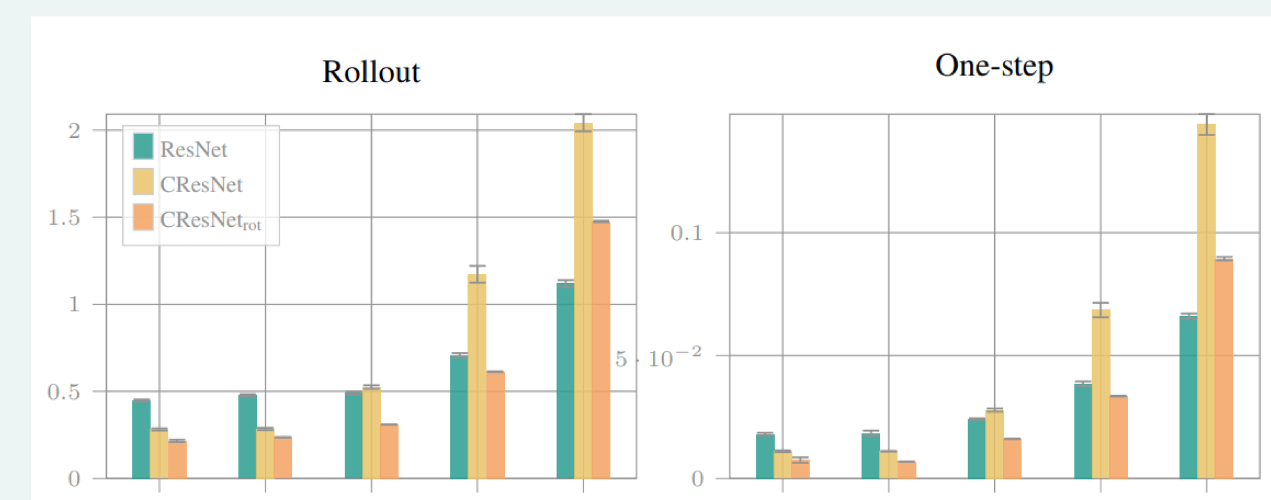
- We look at the 3-dimensional **Maxwell's equations**, which couple electric (E) and magnetic (B) field in an intriguing way:

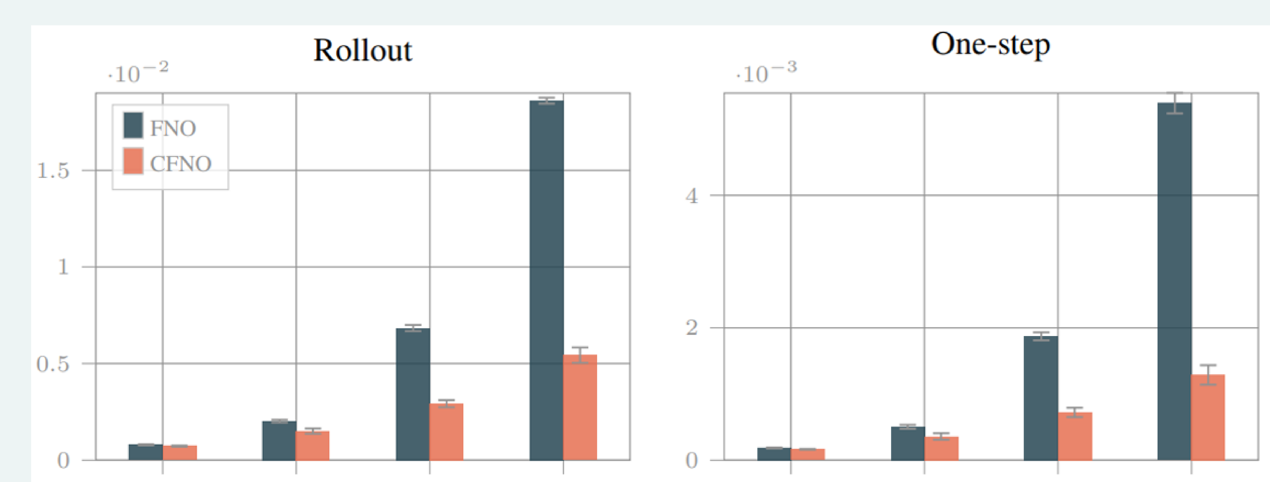
$$\mathbf{F} = E + Bi_3\,.$$



Results

- We test for different equations, for different architectures, and for different number of training trajectories:
  - Convolution vs. **(rotational) Clifford convolution**, i.e., ResNet vs. CResNet.
  - Fourier Neural Operators (FNO) vs. **Clifford Fourier Neural Operators** (CFNO).
- We introduce **Clifford normalization** and **initialization** schemes.
- We measure MSE for next step and 5-step rollout predictions.





- For similar parameter count, Clifford layers consistently improve generalization capabilities of the tested neural PDE surrogates:
  - Strongest improvement for 3D Maxwell's equations.

https://arxiv.org/abs/2209.04934