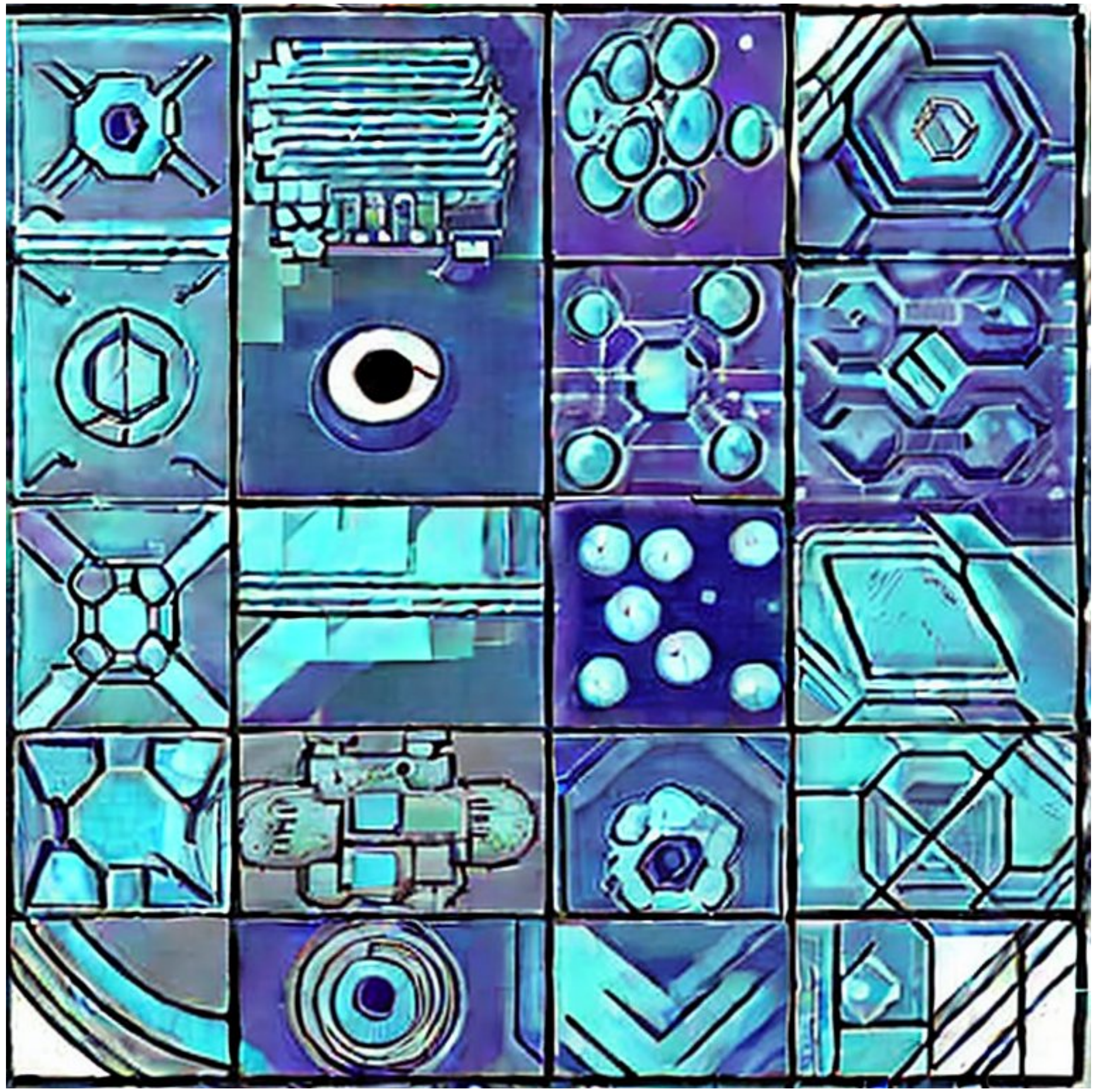


Machine learning for physicists

https://github.com/wangleiphy/ml4p

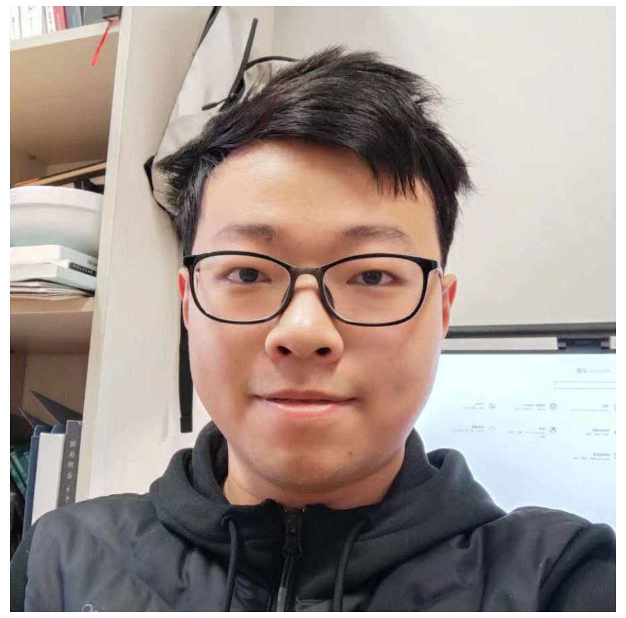
每周四上午10点

2.23	Overview
3.2	Machine learning practices
3.9	A hitchhiker's guide to deep learning
3.16	Research projects hands-on
3.23	Symmetries in machine learning
3.30	Differentiable programming
4.6	Generative models-I
4.13	Generative models-II
4.20	Research projects presentation
4.27	AI for science: why now ?

课程微信群

助教

李子航



李扬帆

考核方式：project + presentation （1学分）

Symmetries in machine learning

Lei Wang (王磊)
Institute of Physics, CAS
https://wangleiphy.github.io

Why symmetry ?

Symmetry dictates interaction.

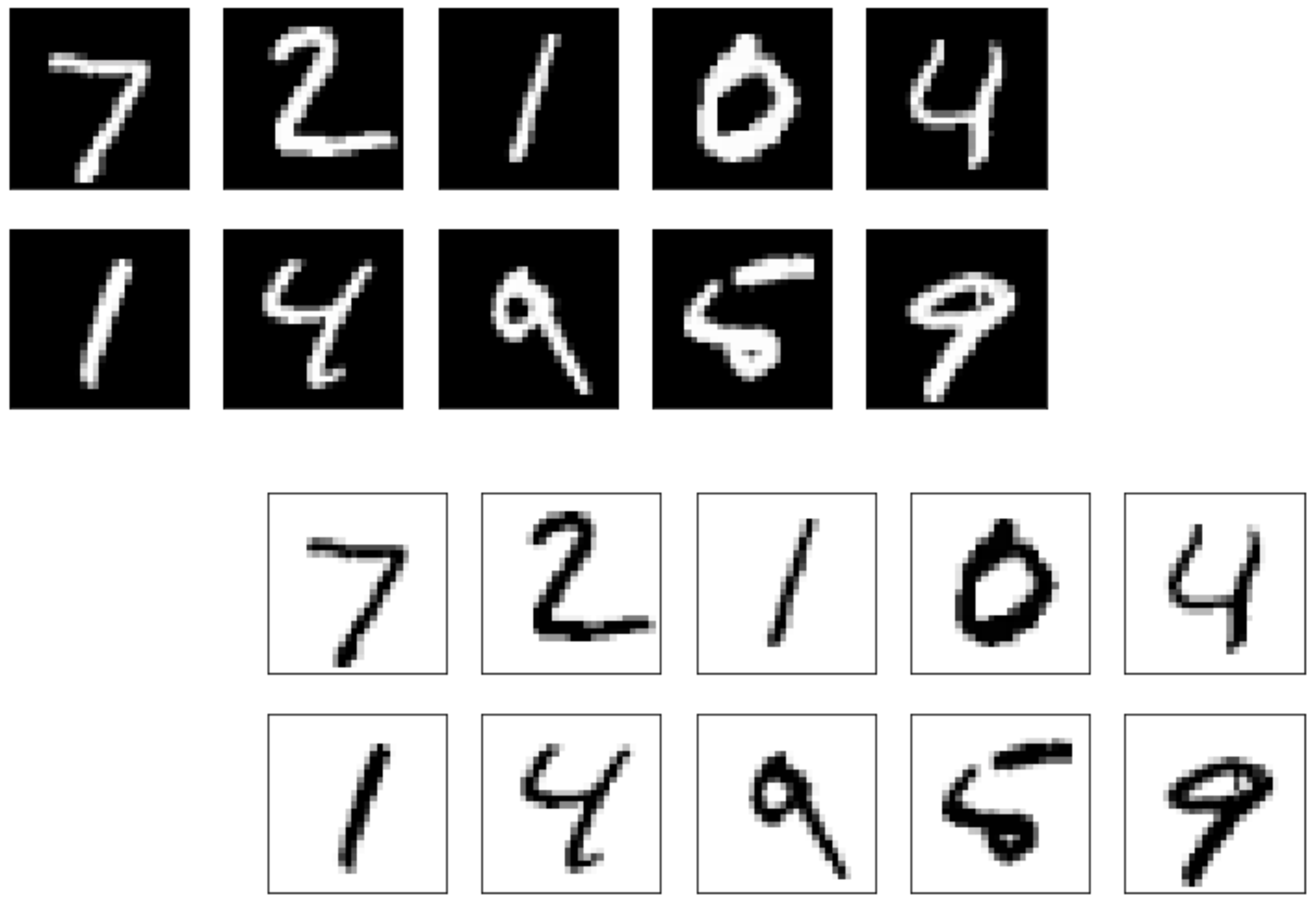
— *Symmetry and physics,* CN Yang

It is only slightly overstating the case to say that physics is the study of symmetry.

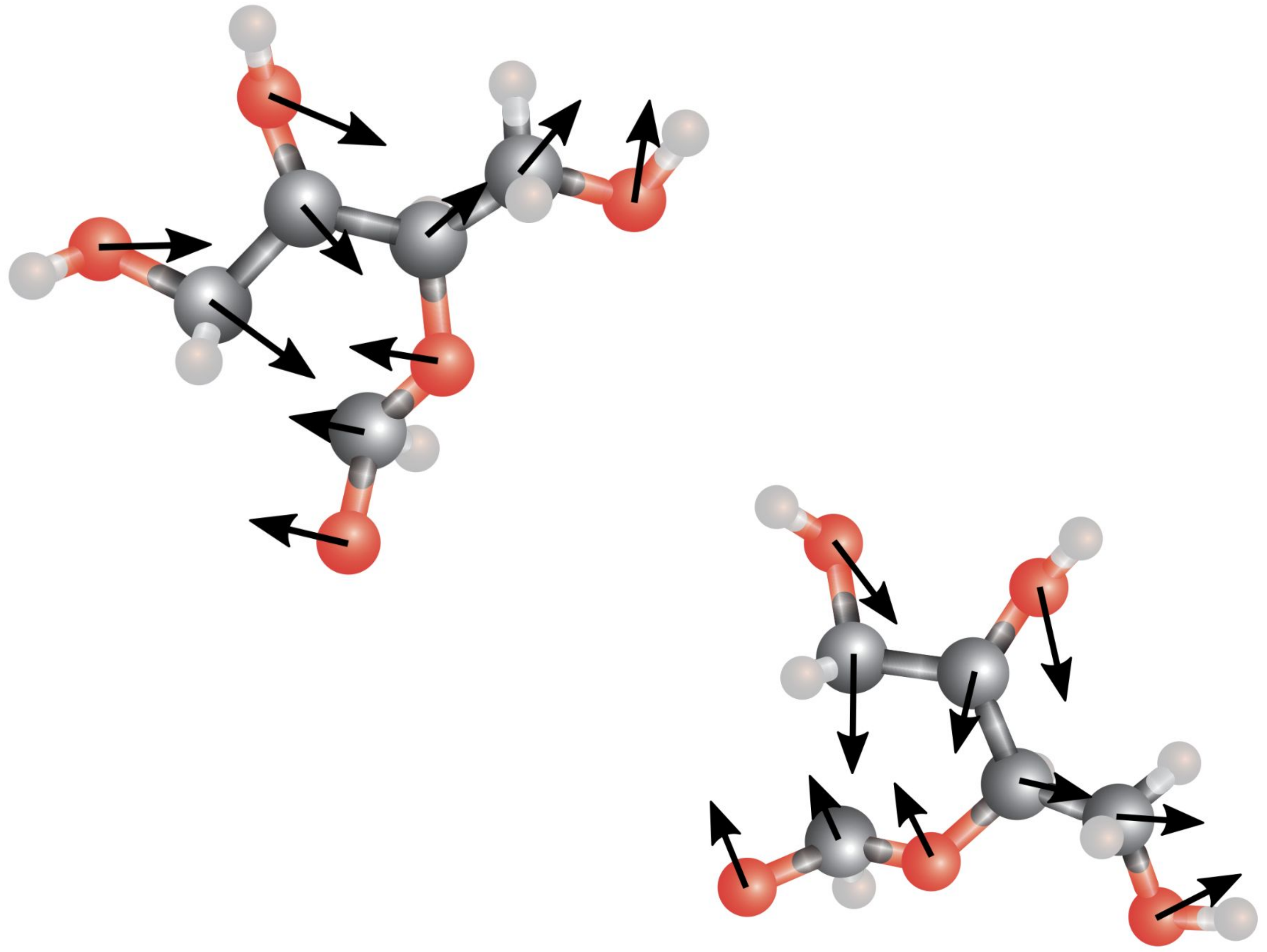
— *More is Different*, Philip Anderson

Machine learning learns about regularities of Nature

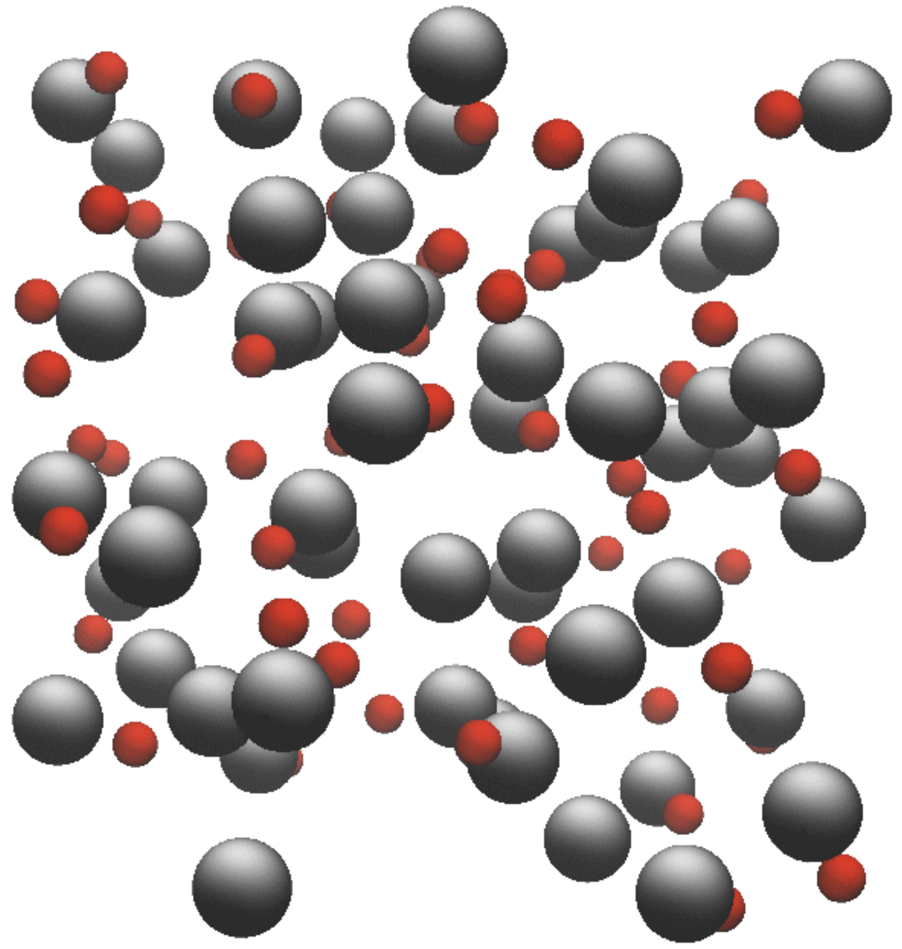
Examples of symmetry group



Z_2



$E(3) = T(3) \rtimes O(3)$



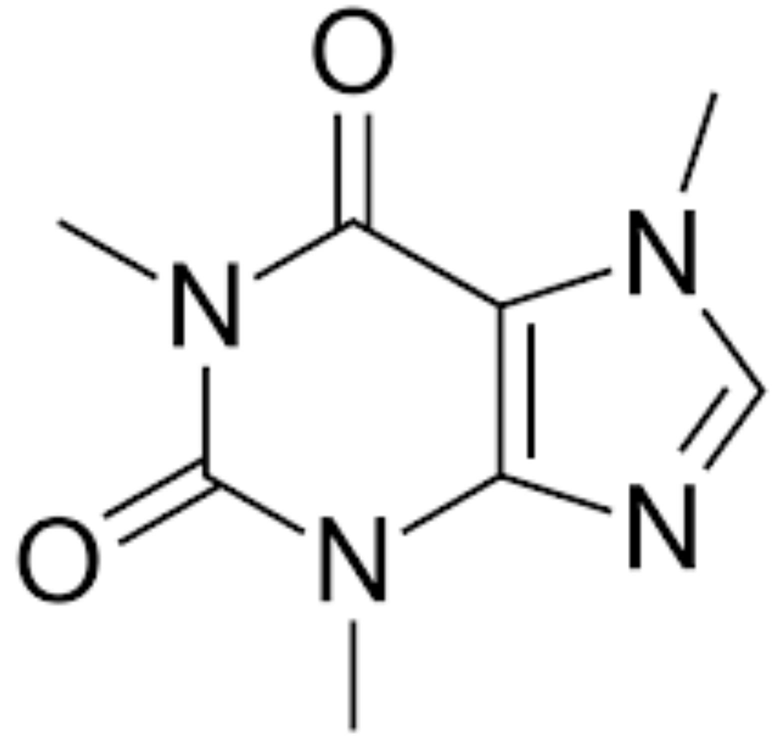
S_n

Symmetry is an increasingly stronger inductive bias for more physical representation

CN1C=NC2=C1C(=O)N(C(=O)N2C)C

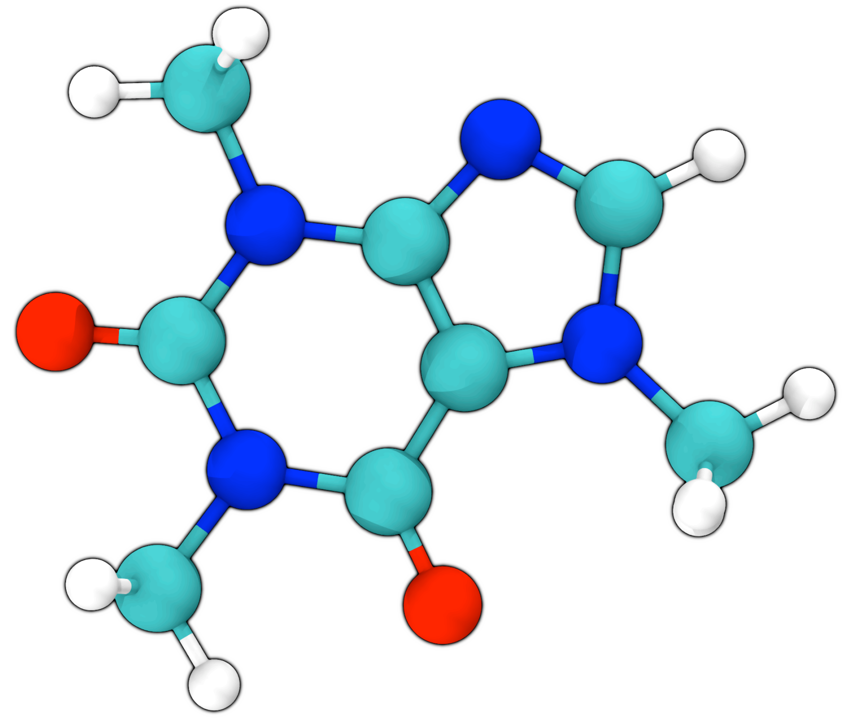
SMLIES

Permutation



Node/edge features of molecule graph

Permutation



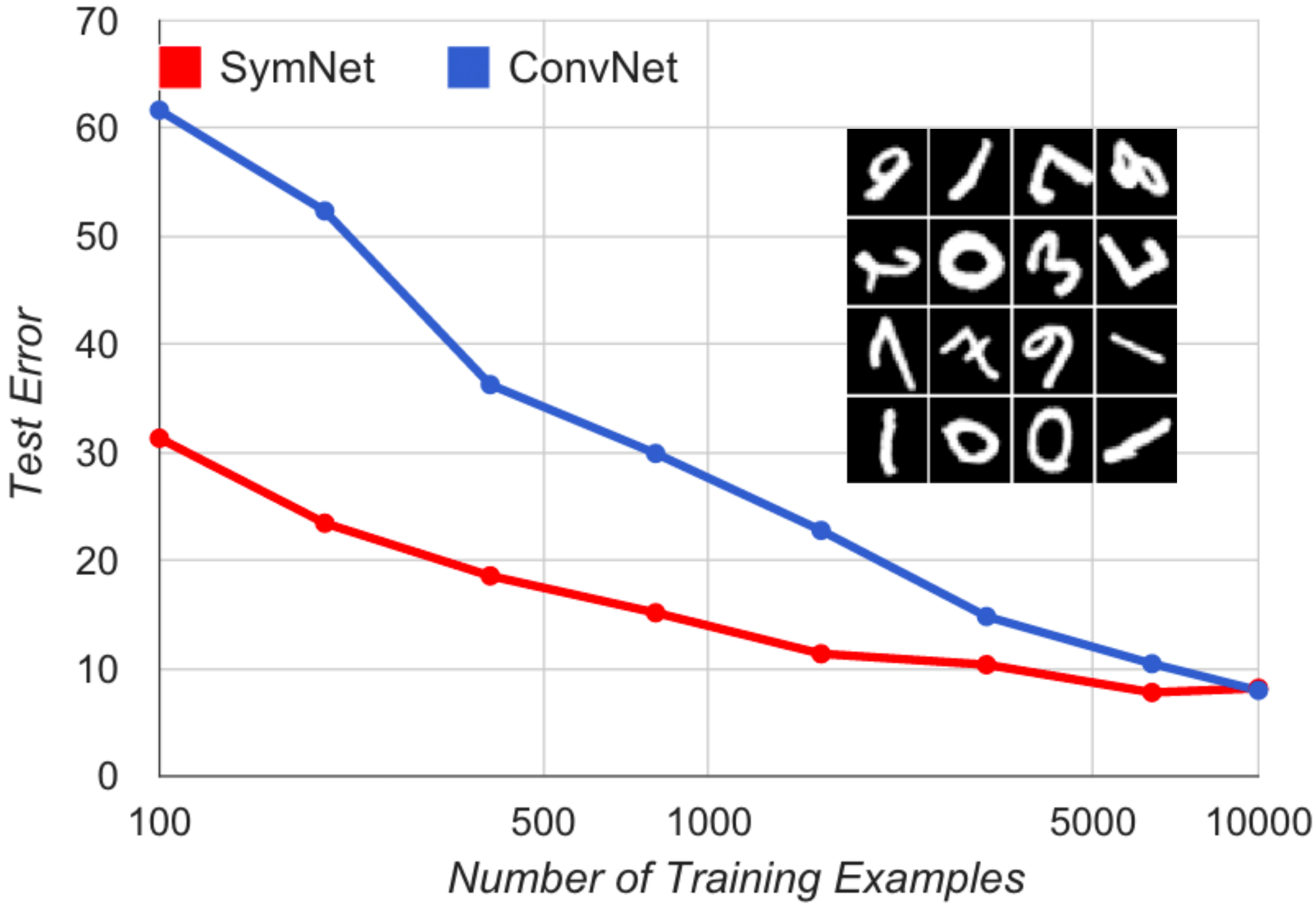
3-dimensional atom coordinates

Translation, Rotation, Permutation

Benefit of using symmetries

- More data efficient -> better generalization ability
- Provably lower variational loss

Gens, Domingos, Deep Symmetry Networks 2014



Claim:

For an invariant energy function $E(g\boldsymbol{x}) = E(\boldsymbol{x})$ and a variational density $p(\boldsymbol{x})$ the symmetric model $p_{\mathrm{sym}}(\boldsymbol{x}) = \frac{1}{N_g}\sum_g p(g\boldsymbol{x})$ will always yields lower variational objective

Proof:

$$\mathscr{L} = \int d\boldsymbol{x}\, p(\boldsymbol{x})\left[\beta E(\boldsymbol{x}) + \ln p(\boldsymbol{x})\right] \geq -\ln Z = -\ln \int dx e^{-\beta E}$$

$$\mathscr{L}_{\mathrm{sym}} = \int d\boldsymbol{x}\, p_{\mathrm{sym}}(\boldsymbol{x})\left[\beta E(\boldsymbol{x}) + \ln p_{\mathrm{sym}}(\boldsymbol{x})\right]$$

$$= \int d\boldsymbol{x}\, p(\boldsymbol{x})\left[\beta E(\boldsymbol{x}) + \ln p_{\mathrm{sym}}(\boldsymbol{x})\right]$$

$$\mathscr{L} - \mathscr{L}_{\mathrm{sym}} = \int d\boldsymbol{x}\, p(\boldsymbol{x})\left[\ln p(\boldsymbol{x}) - \ln p_{\mathrm{sym}}(\boldsymbol{x})\right] \geq 0$$

Moore, Symmetrized variational inference, NIPS 2016, Shuo-Hui Li, LW, PRL 2018 Appendix D

Ways to exploit symmetry

Canonical form

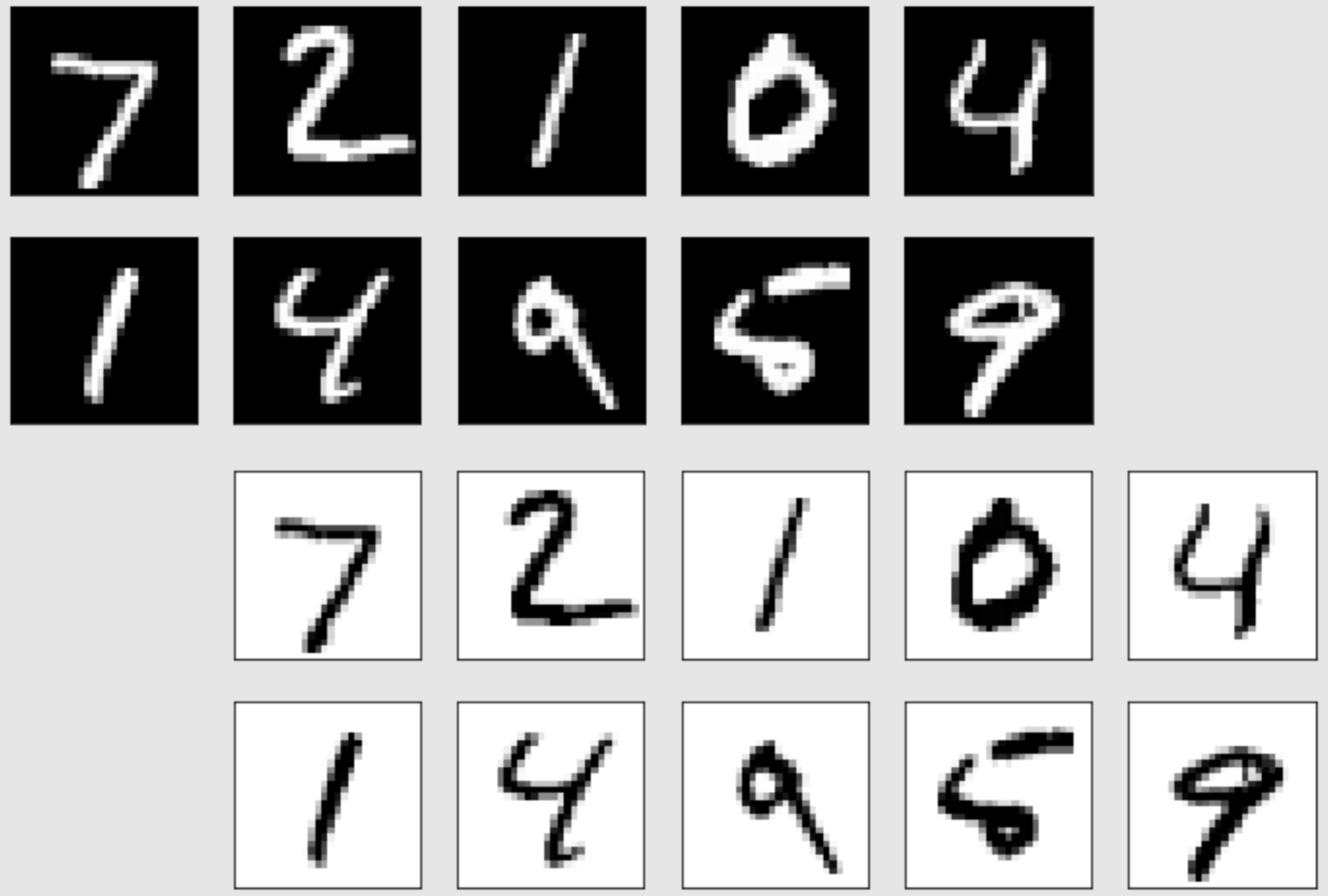
- Black/white background
- Translation: shift center-of-mass
- Rotation: fix principal axis

Explicit symmetrization

$$\text{logits} = f(\boldsymbol{x}) + f(-\boldsymbol{x})$$

Ways to exploit symmetry

Data augmentation



Invariant feature

$$\boldsymbol{h} = \sigma(W\boldsymbol{x})$$

σ: even
activation function

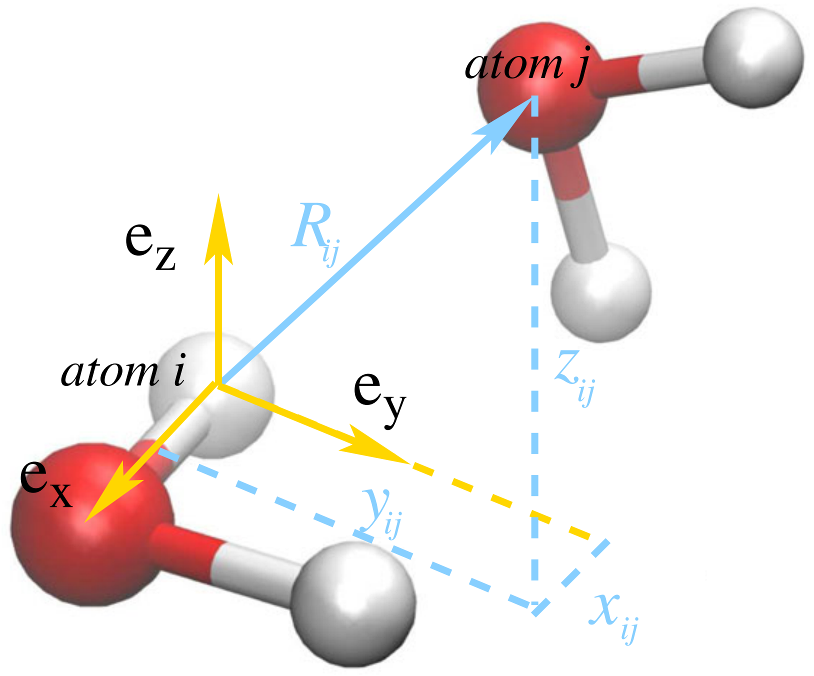
Equivariant transformation

$$\boldsymbol{h} = \sigma(W\boldsymbol{x})$$

σ: odd
activation function

Invariant pooling in the end

Invariant features in DeepMD



$$E = \sum_i f_i\left(D_i(\boldsymbol{r}_i, \{\boldsymbol{r}_j\}_{j\in\mathcal{N}(i)})\right)$$

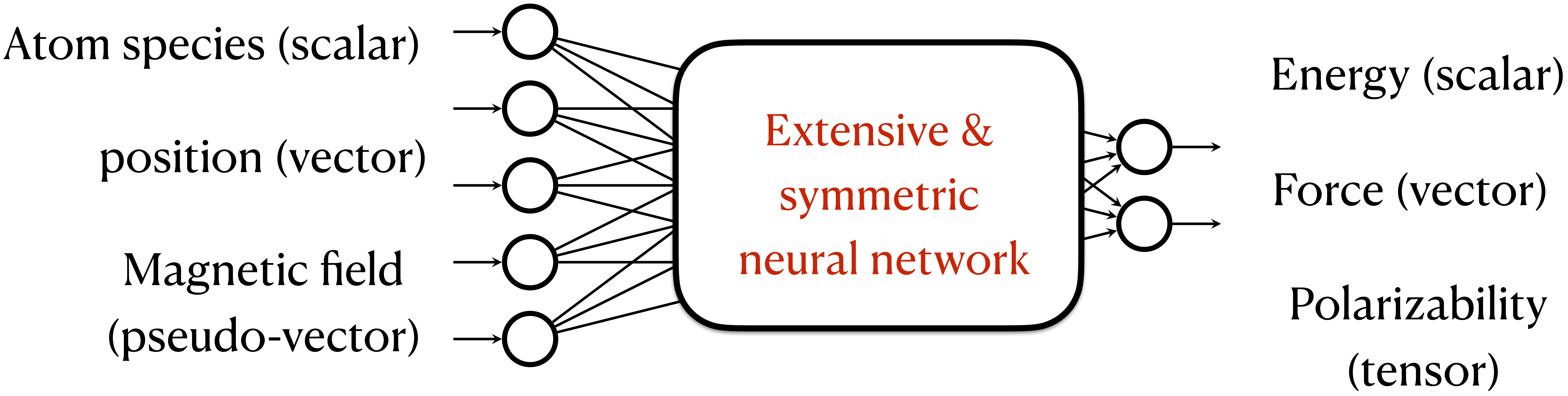
Linfeng Zhang et al, NIPS 2018

$$\boldsymbol{h}_i = \overset{4\times|\mathcal{N}(i)|}{\boxed{\begin{matrix}|\boldsymbol{r}_i-\boldsymbol{r}_j| \\ x_i-x_j \\ y_i-y_j \\ z_i-z_j\end{matrix}\quad \forall j\in\mathcal{N}(i)}} \times \overset{|\mathcal{N}(i)|\times F}{\boxed{\begin{matrix}g_k(|\boldsymbol{r}_i-\boldsymbol{r}_j|) \\ \forall k\in 1\cdots F\end{matrix}}}$$

$D_i = \boldsymbol{h}_i^T\boldsymbol{h}_i$ is translational, rotational, and permutational invariant descriptor of the local environment of the i-th atom

This construction also takes care of extensibility of total energy

Transformation of geometry tensors



Sommers et al, 2020

Group theory of neural networks

$$
\begin{tikzcd}
X \arrow[r, "f"] \arrow[d, "g"'] & Y \arrow[d] \\
X' \arrow[r] & Y'
\end{tikzcd}
$$

$f : X \mapsto Y$ is equivariant with respect to group G if

$$D_Y(g)f(\boldsymbol{x}) = f(D_X(g)\boldsymbol{x}) \quad \forall g \in G, \forall \boldsymbol{x} \in X$$

$D_X(g)$ and $D_Y(g)$: group representations in vector spaces X and Y

Invariance is a special case of equvariance when $D_Y(g) = I$

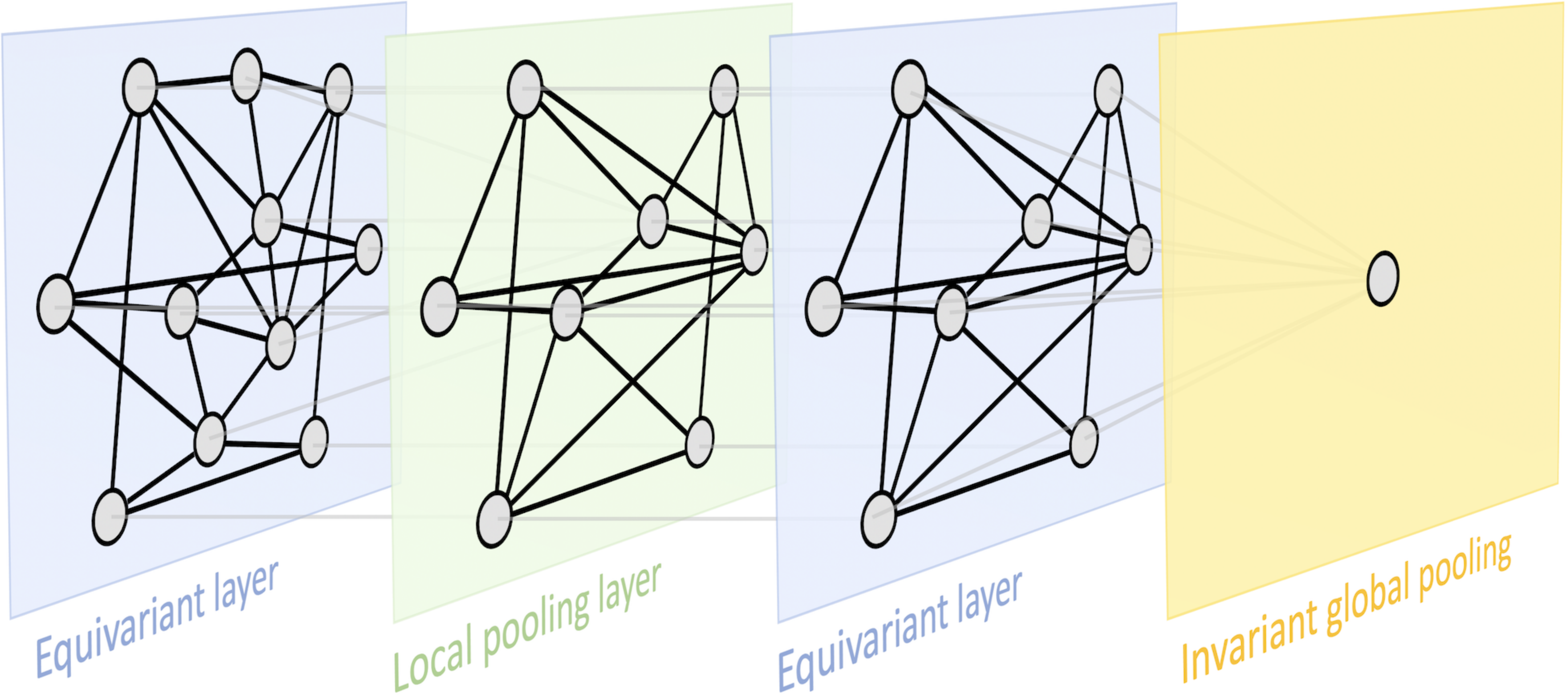
Composition of equivariant functions is also equivariant

Equivariant neural network = composition of equivariant layers

The geometric deep learning

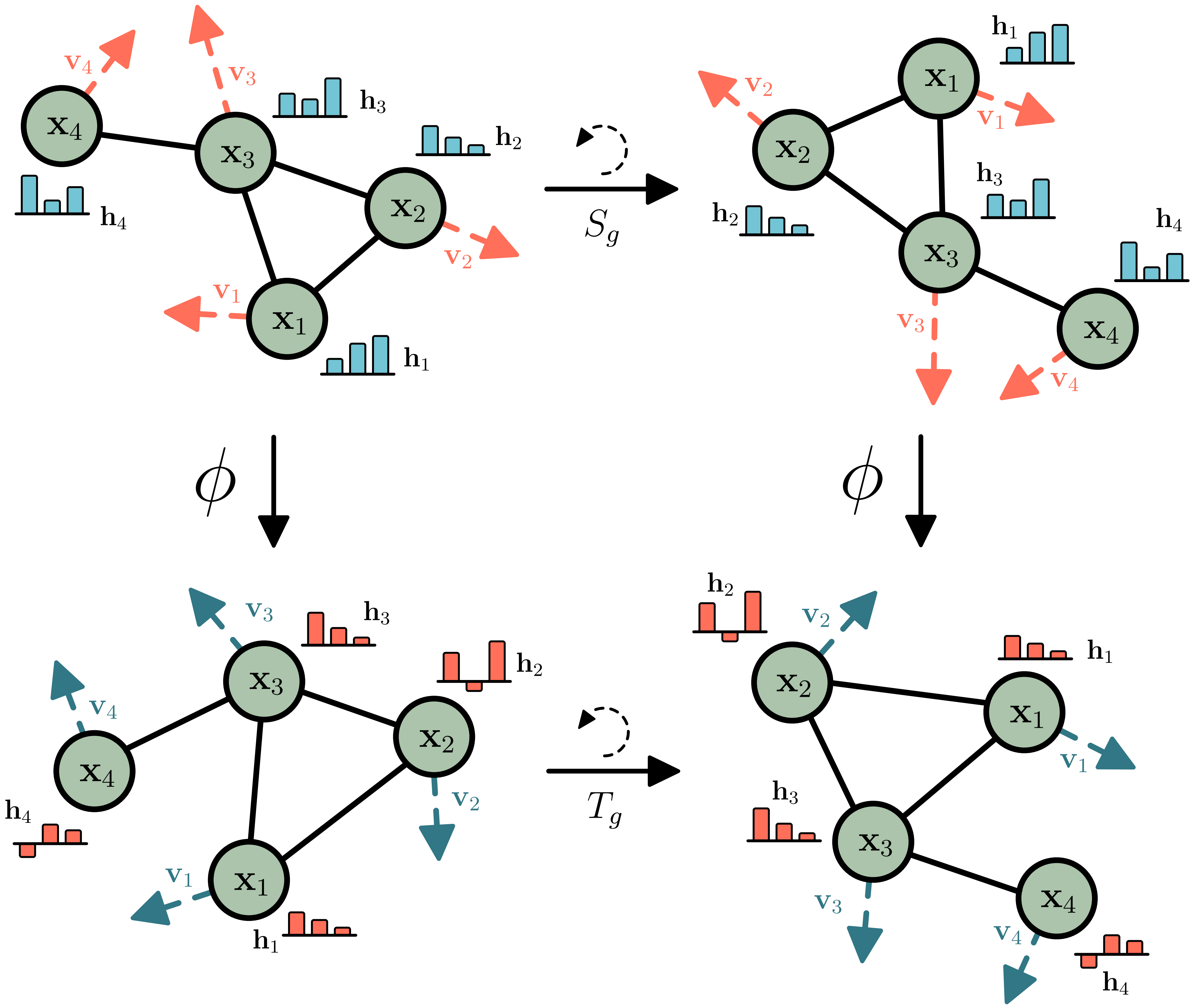
Bronstein et al, 2104.13478

> In this text, we make a modest attempt to apply the Erlangen Programme mindset to the domain of deep learning, with the ultimate goal of obtaining a systematisation of this field and 'connecting the dots'. We call this geometri-



Equivariant layers are key players in the game

Euclidian Equivariant Graph NN



A GNN layer

$$
\begin{aligned}
\mathbf{m}_{ij} &= \phi_e(\mathbf{h}_i^l, \mathbf{h}_j^l, a_{ij}) \\
\mathbf{m}_i &= \sum_{j \in \mathcal{N}(i)} \mathbf{m}_{ij} \\
\mathbf{h}_i^{l+1} &= \phi_h(\mathbf{h}_i^l, \mathbf{m}_i)
\end{aligned}
$$

A EGNN layer

$$
\begin{aligned}
\mathbf{m}_{ij} &= \phi_e\left(\mathbf{h}_i^l, \mathbf{h}_j^l, \left\|\mathbf{x}_i^l - \mathbf{x}_j^l\right\|^2, a_{ij}\right) \\
\mathbf{x}_i^{l+1} &= \mathbf{x}_i^l + C \sum_{j \neq i} \left(\mathbf{x}_i^l - \mathbf{x}_j^l\right) \phi_x\left(\mathbf{m}_{ij}\right) \\
\mathbf{m}_i &= \sum_{j \neq i} \mathbf{m}_{ij} \\
\mathbf{h}_i^{l+1} &= \phi_h\left(\mathbf{h}_i^l, \mathbf{m}_i\right)
\end{aligned}
$$

Satorras et al, 2102.09844

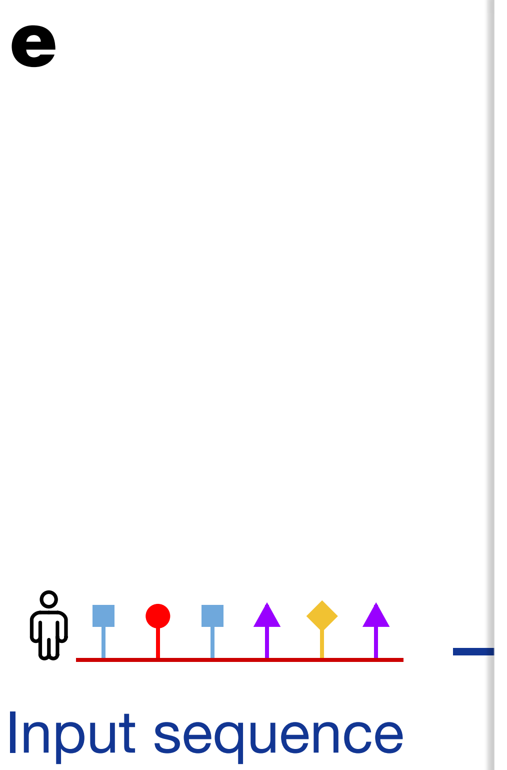
Euclidean equivariance

$$Qx' + g, h' = f(Qx + g, h)$$

Let's prove it

See it for real in code, and write test

1. Translation equivariance. Translating the input by $g \in \mathbb{R}^n$ results in an equivalent translation of the output. Let $\mathbf{x}+g$ be shorthand for $(\mathbf{x}_1+g, \ldots, \mathbf{x}_M+g)$. Then $\mathbf{y}+g = \phi(\mathbf{x}+g)$
2. Rotation (and reflection) equivariance. For any orthogonal matrix $Q \in \mathbb{R}^{n\times n}$, let $Q\mathbf{x}$ be shorthand for $(Q\mathbf{x}_1, \ldots, Q\mathbf{x}_M)$. Then rotating the input results in an equivalent rotation of the output $Q\mathbf{y} = \phi(Q\mathbf{x})$.
3. Permutation equivariance. Permuting the input results in the same permutation of the output $P(\mathbf{y}) = \phi(P(\mathbf{x}))$ where P is a permutation on the row indexes.



Algorithm 22 Invariant point attention (IPA)

def InvariantPointAttention($\{\mathbf{s}_i\}, \{\mathbf{z}_{ij}\}, \{T_i\}, N_{\text{head}} = 12, c = 16, N_{\text{query points}} = 4, N_{\text{point values}} = 8$) :

1: $\mathbf{q}_i^h, \mathbf{k}_i^h, \mathbf{v}_i^h = \text{LinearNoBias}(\mathbf{s}_i)$ $\quad \mathbf{q}_i^h, \mathbf{k}_i^h, \mathbf{v}_i^h \in \mathbb{R}^c,\ h \in \{1, \dots, N_{\text{head}}\}$

2: $\vec{\mathbf{q}}_i^{hp}, \vec{\mathbf{k}}_i^{hp} = \text{LinearNoBias}(\mathbf{s}_i)$ $\quad \vec{\mathbf{q}}_i^{hp}, \vec{\mathbf{k}}_i^{hp}, \in \mathbb{R}^3,\ p \in \{1, \dots, N_{\text{query points}}\}$, units: nanometres

3: $\vec{\mathbf{v}}_i^{hp} = \text{LinearNoBias}(\mathbf{s}_i)$ $\quad \vec{\mathbf{v}}_i^{hp} \in \mathbb{R}^3,\ p \in \{1, \dots, N_{\text{point values}}\}$, units: nanometres

4: $b_{ij}^h = \text{LinearNoBias}(\mathbf{z}_{ij})$

5: $w_C = \sqrt{\frac{2}{9 N_{\text{query points}}}}$,

6: $w_L = \sqrt{\frac{1}{3}}$

7: $a_{ij}^h = \text{softmax}_j \left(w_L \left(\frac{1}{\sqrt{c}} \mathbf{q}_i^{h\top} \mathbf{k}_j^h + b_{ij}^h - \frac{\gamma^h w_C}{2} \sum_p \left\| T_i \circ \vec{\mathbf{q}}_i^{hp} - T_j \circ \vec{\mathbf{k}}_j^{hp} \right\|^2 \right) \right)$

8: $\tilde{\mathbf{o}}_i^h = \sum_j a_{ij}^h \mathbf{z}_{ij}$

9: $\mathbf{o}_i^h = \sum_j a_{ij}^h \mathbf{v}_j^h$

10: $\vec{\mathbf{o}}_i^{hp} = T_i^{-1} \circ \sum_j a_{ij}^h \left(T_j \circ \vec{\mathbf{v}}_j^{hp} \right)$

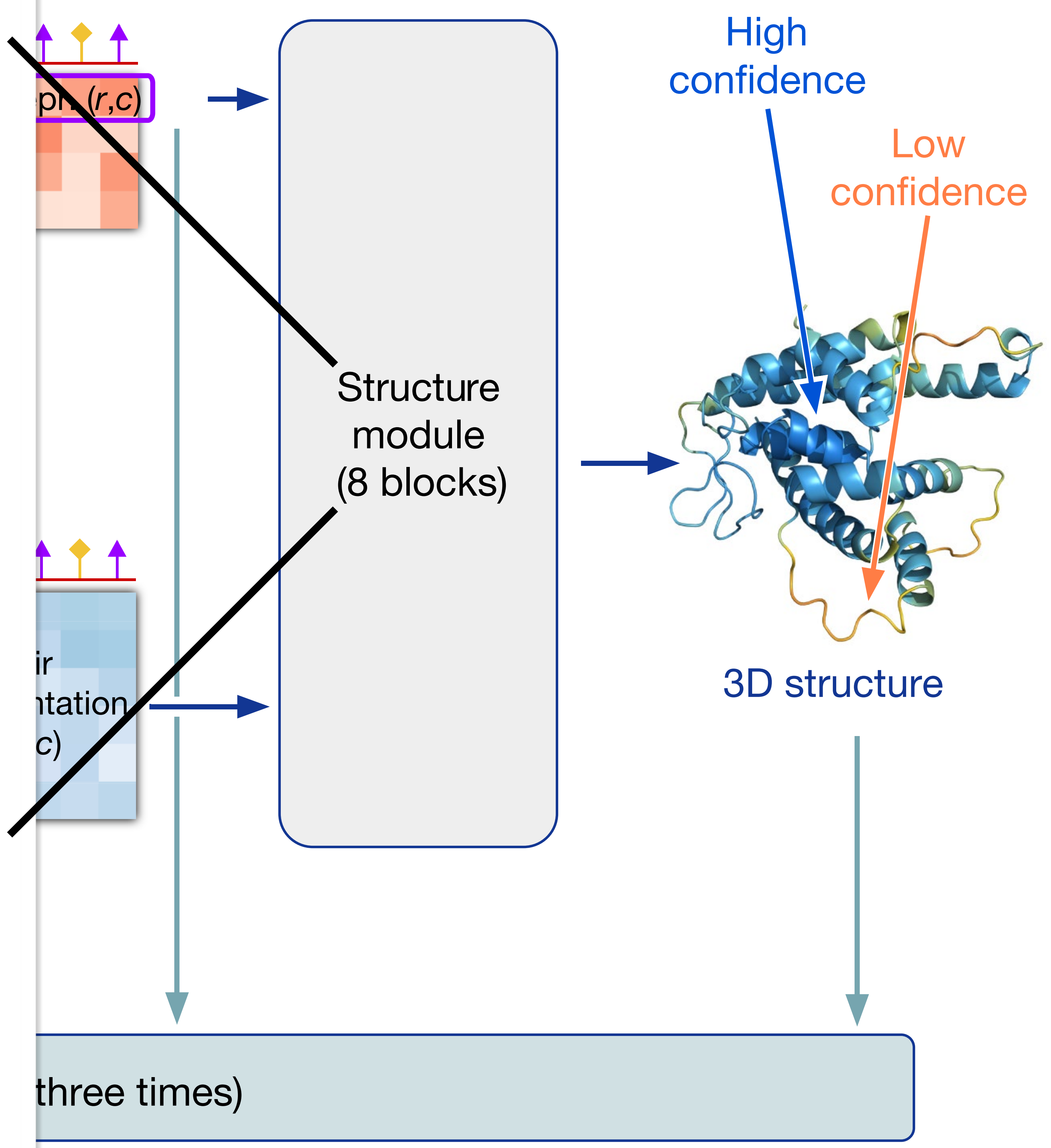
11: $\tilde{\mathbf{s}}_i = \text{Linear}\left(\text{concat}_{h,p}(\tilde{\mathbf{o}}_i^h, \mathbf{o}_i^h, \vec{\mathbf{o}}_i^{hp}, \left\| \vec{\mathbf{o}}_i^{hp} \right\|) \right)$

12: **return** $\{\tilde{\mathbf{s}}_i\}$

The proof for invariance is straight-forward: The global transformation cancels out in the affinity computation (Algorithm 22 line 7), because the L2-norm of a vector is invariant under rigid transformations:

$$\left\| \left(T_{\text{global}} \circ T_i\right) \circ \vec{\mathbf{q}}_i^{hp} - \left(T_{\text{global}} \circ T_j\right) \circ \vec{\mathbf{k}}_j^{hp} \right\|^2 = \left\| T_{\text{global}} \circ \left(T_i \circ \vec{\mathbf{q}}_i^{hp} - T_j \circ \vec{\mathbf{k}}_j^{hp} \right) \right\|^2 \tag{3}$$

$$= \left\| T_i \circ \vec{\mathbf{q}}_i^{hp} - T_j \circ \vec{\mathbf{k}}_j^{hp} \right\|^2 . \tag{4}$$



Similar bit of math in AlphaFold2's structure module

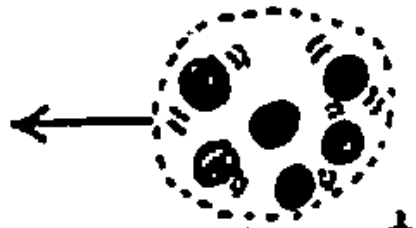
Satorras et al, 2102.09844

	GNN	Radial Field	TFN	Schnet	EGNN
Edge	$\mathbf{m}_{ij} = \phi_e(\mathbf{h}_i^l, \mathbf{h}_j^l, a_{ij})$	$\mathbf{m}_{ij} = \phi_{\text{rf}}(\Vert\mathbf{r}_{ij}^l\Vert)\mathbf{r}_{ij}^l$	$\mathbf{m}_{ij} = \sum_k \mathbf{W}^{lk}\mathbf{r}_{ji}^l\mathbf{h}_i^{lk}$	$\mathbf{m}_{ij} = \phi_{\text{cf}}(\Vert\mathbf{r}_{ij}^l\Vert)\phi_{\text{s}}(\mathbf{h}_j^l)$	$\mathbf{m}_{ij} = \phi_e(\mathbf{h}_i^l, \mathbf{h}_j^l, \Vert\mathbf{r}_{ij}^l\Vert^2, a_{ij})$ $\hat{\mathbf{m}}_{ij} = \mathbf{r}_{ij}^l\phi_x(\mathbf{m}_{ij})$
Agg.	$\mathbf{m}_i = \sum_{j\in\mathcal{N}(i)} \mathbf{m}_{ij}$	$\mathbf{m}_i = \sum_{j\neq i} \mathbf{m}_{ij}$	$\mathbf{m}_i = \sum_{j\neq i} \mathbf{m}_{ij}$	$\mathbf{m}_i = \sum_{j\neq i} \mathbf{m}_{ij}$	$\mathbf{m}_i = \sum_{j\neq i} \mathbf{m}_{ij}$ $\hat{\mathbf{m}}_i = C\sum_{j\neq i} \hat{\mathbf{m}}_{ij}$
Node	$\mathbf{h}_i^{l+1} = \phi_h(\mathbf{h}_i^l, \mathbf{m}_i)$	$\mathbf{x}_i^{l+1} = \mathbf{x}_i^l + \mathbf{m}_i$	$\mathbf{h}_i^{l+1} = w^{ll}\mathbf{h}_i^l + \mathbf{m}_i$	$\mathbf{h}_i^{l+1} = \phi_h(\mathbf{h}_i^l, \mathbf{m}_i)$	$\mathbf{h}_i^{l+1} = \phi_h\left(\mathbf{h}_i^l, \mathbf{m}_i\right)$ $\mathbf{x}_i^{l+1} = \mathbf{x}_i^l + \hat{\mathbf{m}}_i$
	Non-equivariant	$\mathrm{E}(n)$-Equivariant	$\mathrm{SE}(3)$-Equivariant	$\mathrm{E}(n)$-Invariant	$\mathrm{E}(n)$-Equivariant

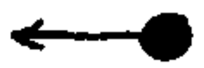
Feynman's backflow

$$\zeta_i = x_i + \sum_{j\neq i} \eta(|x_i - x_j|)(x_j - x_i)$$

Feynman & Cohen 1956



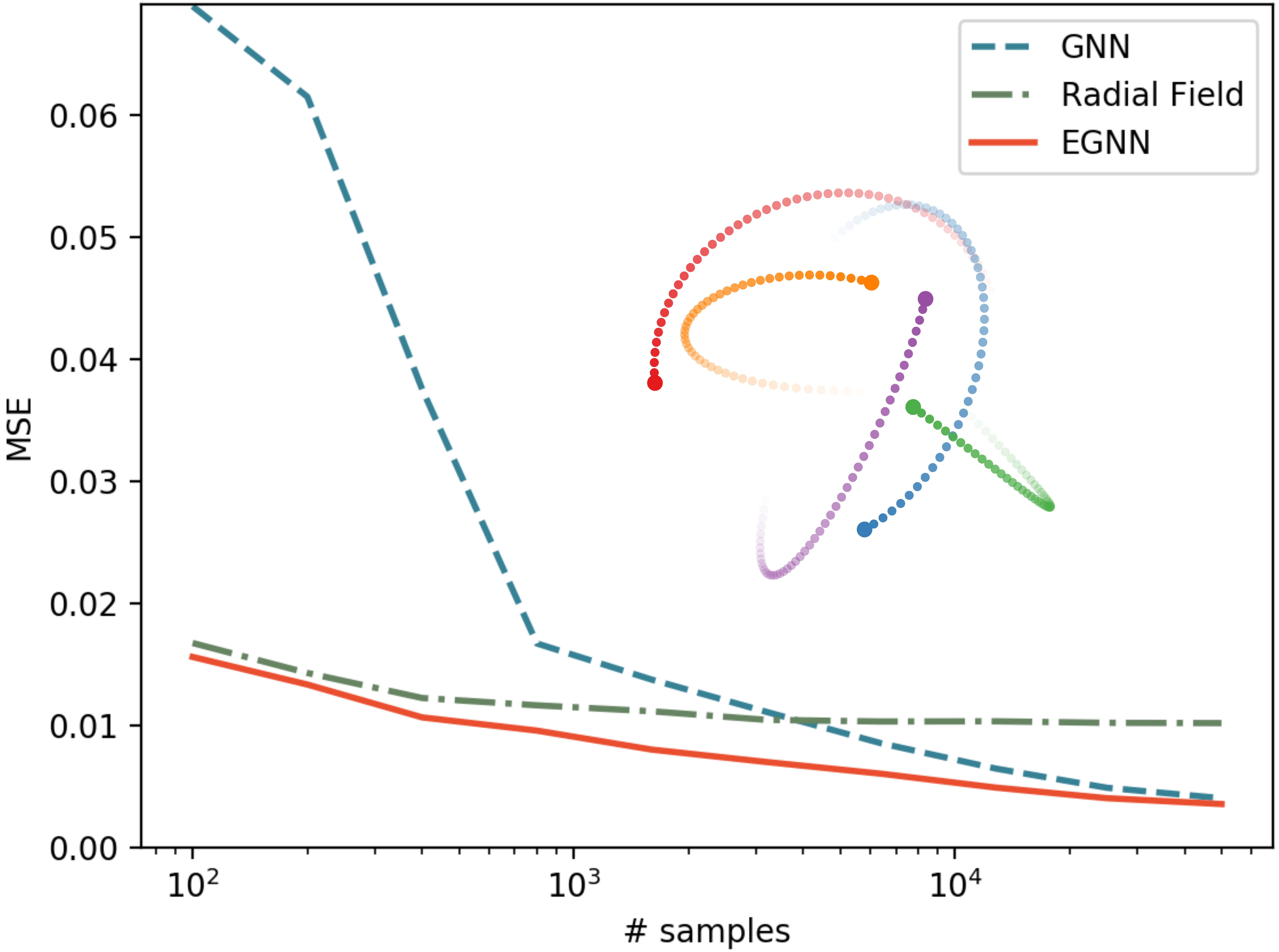
quasi particle



real particle

Predict positions of interacting particles

Satorras et al, 2102.09844



Equivariance from invariance

Claim: The force $\boldsymbol{f} = [\nabla_{\boldsymbol{x}}\varphi(\boldsymbol{x})]^T$ given a invariance scalar function $\varphi(R\boldsymbol{x}) = \varphi(\boldsymbol{x})$ is rotational equvariance, i.e., $\boldsymbol{f}(R\boldsymbol{x}) = R\boldsymbol{f}(\boldsymbol{x})$

Proof: Derivative on both side $$\nabla_{\boldsymbol{x}}\varphi(R\boldsymbol{x}) = \nabla_{\boldsymbol{x}}\varphi(\boldsymbol{x})$$

Chain rule $$\nabla_{R\boldsymbol{x}}\varphi(R\boldsymbol{x})R = \nabla_{\boldsymbol{x}}\varphi(\boldsymbol{x})$$

$$\Rightarrow \nabla_{R\boldsymbol{x}}\varphi(R\boldsymbol{x}) = \nabla_{\boldsymbol{x}}\varphi(\boldsymbol{x})R^{-1} = \nabla_{\boldsymbol{x}}\varphi(\boldsymbol{x})R^T$$

Papamakarios et al, 1912.02762

Take home message: Invariance easier than covariance

Given a Lagrangian, we have to vary it to obtain the equation of motion. For normal everyday use, we might as well stick with the equation of motion. But when symmetry comes to the fore and assumes the leading role, it is significantly easier to render the Lagrangian invariant than to render the equation of motion covariant. For the familiar rotation group, the advantage is minimal; however, when we come to more involved symmetries, finding the correct Lagrangian is usually much easier than finding the correct equation of motion. Indeed, as an example,[9] when Einstein searched for his theory of gravity, he opted for the equation of motion. He could have saved himself a considerable amount of travail if he had determined the relevant Lagrangian instead.

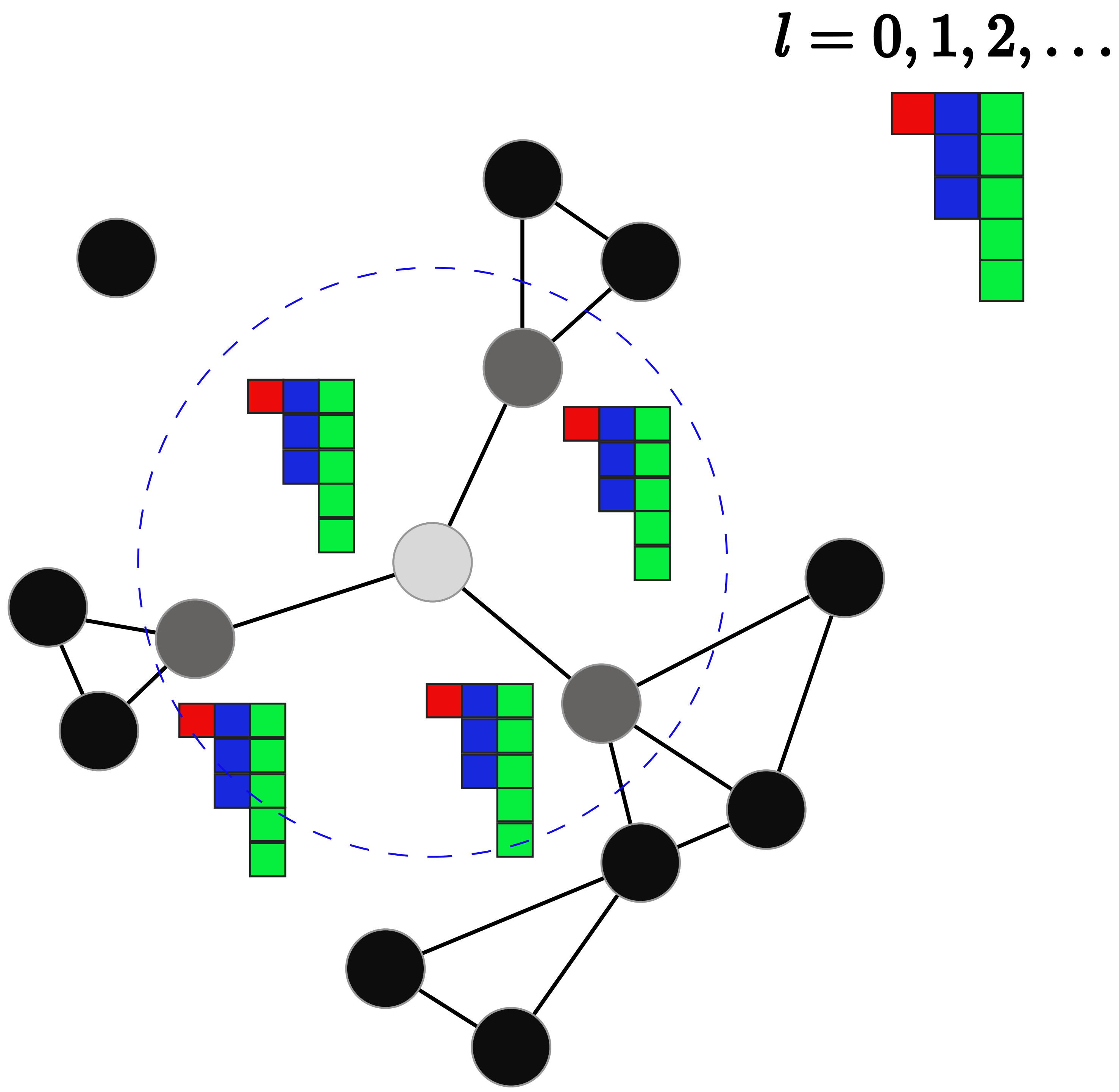
Zee, Group theory in a nutshell for physicists

$$S_{\mathrm{EH}} = \frac{1}{2\kappa}\int d^4x R\sqrt{-g}$$

General Relativity is a model of Nature

Tensor field networks

Thomas et al, 1802.08219



$$\boldsymbol{h}'_{i\mu} = \sum_{j \in \mathcal{N}(i)} \boldsymbol{h}_{j\nu} \overset{W_{\mu\nu}(|\boldsymbol{x}_{ij}|)}{\otimes} \boldsymbol{Y}(\hat{\boldsymbol{x}}_{ij})$$

Sum over neighboring atoms

$$\boldsymbol{x}_{ij} = \boldsymbol{x}_i - \boldsymbol{x}_j$$

$$\hat{\boldsymbol{x}}_{ij} = \boldsymbol{x}_{ij} / |\boldsymbol{x}_{ij}|$$

Order

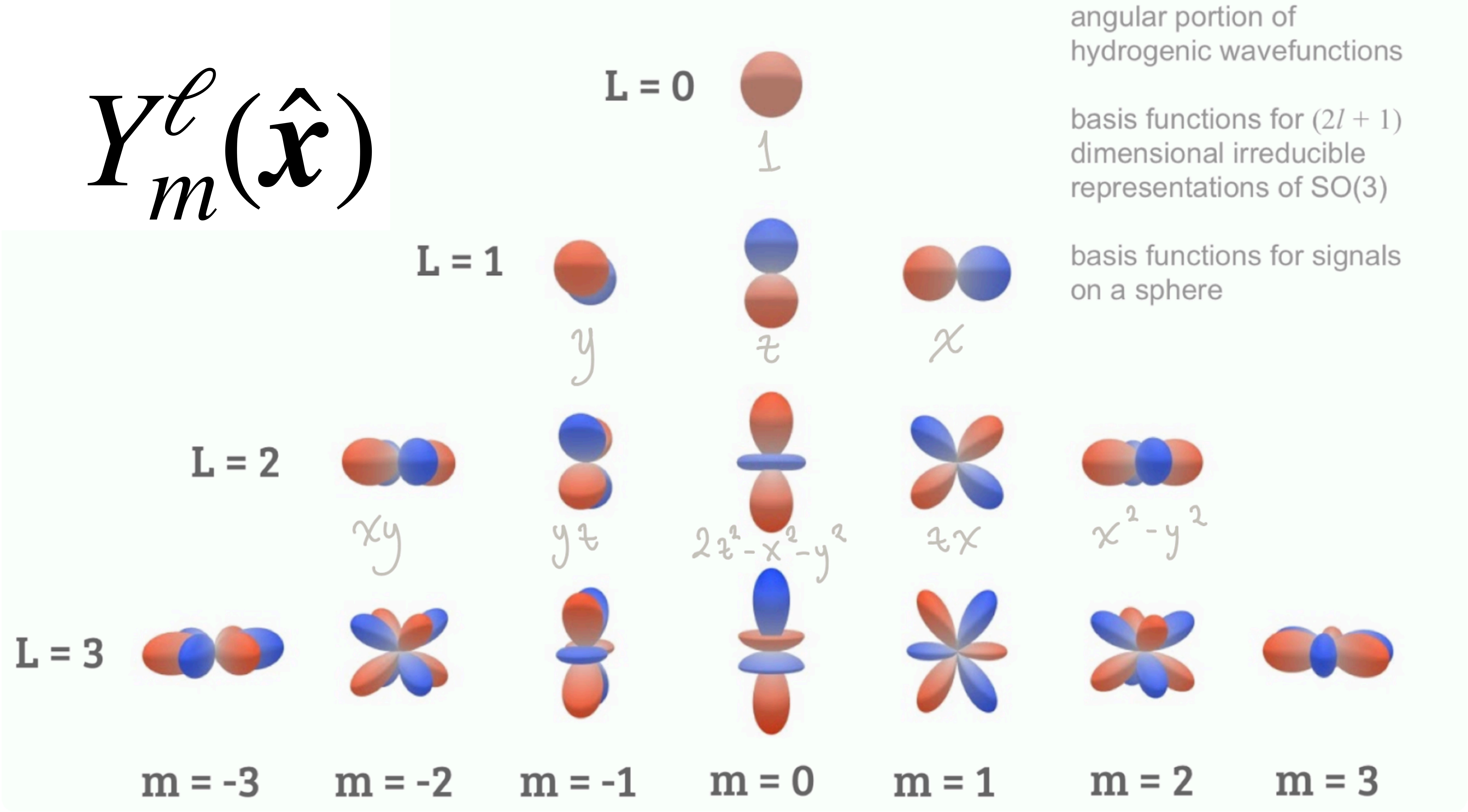
$$[h^{\ell}_{m}]_{i\mu}$$

Channel

Degree Atom

Spherical Harmonics

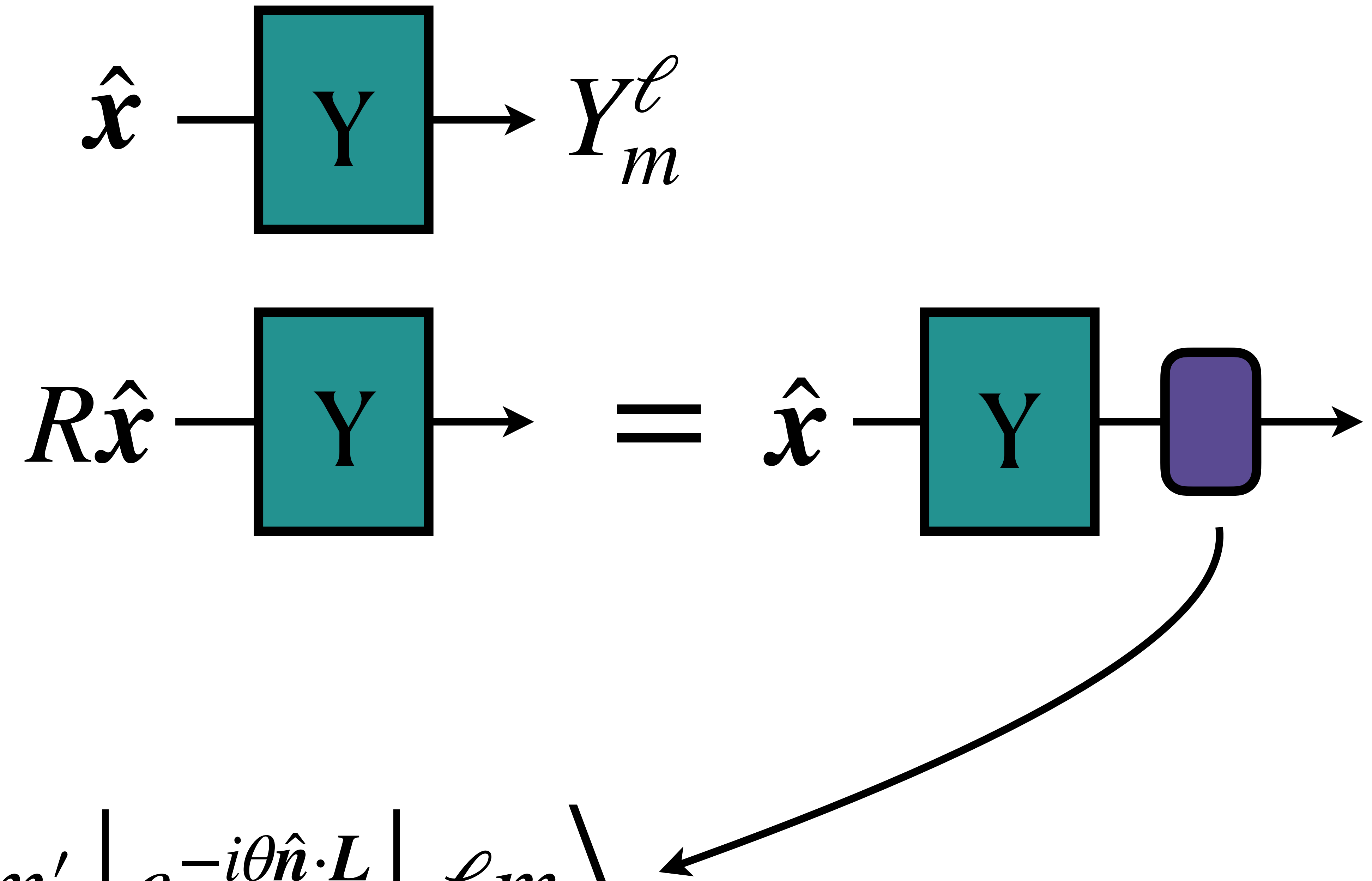
$$Y^{\ell}_{m}(\hat{\boldsymbol{x}})$$



Transformation of spherical harmonics

$$Y^\ell_m(\hat{\boldsymbol{x}}) = \langle \hat{\boldsymbol{x}} \,|\, \ell m \rangle$$

$$Y^\ell_m(R\hat{\boldsymbol{x}}) = \sum_{n=-\ell}^{\ell} D^\ell_{mn}(R)\, Y^\ell_n(\hat{\boldsymbol{x}})$$



Wigner D-matrix $\quad D^\ell_{m'm}(R) \equiv \left\langle \ell m' \middle| e^{-i\theta \hat{\boldsymbol{n}}\cdot \boldsymbol{L}} \middle| \ell m \right\rangle$

D^ℓ is a $(2\ell + 1) \times (2\ell + 1)$ unitary matrix

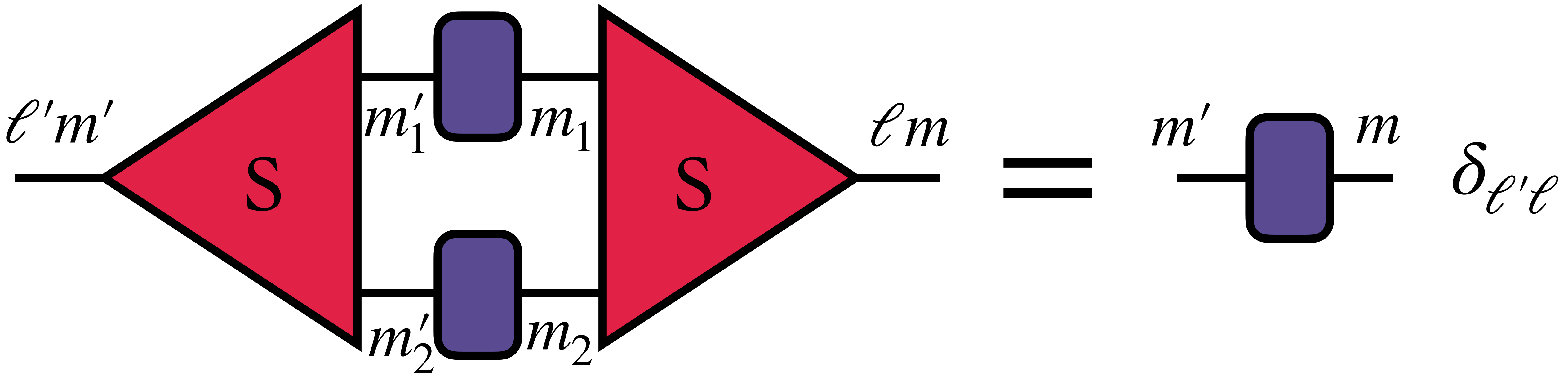
Tensor product

$$[\boldsymbol{u} \otimes \boldsymbol{v}]^{\ell}_{m} = \sum_{m_1=-\ell_1}^{\ell_1} \sum_{m_2=-\ell_2}^{\ell_2} S^{\ell_1 \ell_2}_{m_1 m_2 \ell m} u^{\ell_1}_{m_1} v^{\ell_2}_{m_2}$$

Clebsch-Gordan coefficients $\left(\langle \ell_1 m_1 | \otimes \langle \ell_2 m_2 \right) | \ell m \rangle \equiv \left[S^{\ell_1 \ell_2}\right]_{m_1 m_2, \ell m}$

$$\left[[S^{\ell_1 \ell_2}]^{-1} \left(D^{\ell_1} \otimes D^{\ell_2}\right) S^{\ell_1 \ell_2}\right] = \bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} D^{\ell}$$

Orthogonal
rotation to irreps



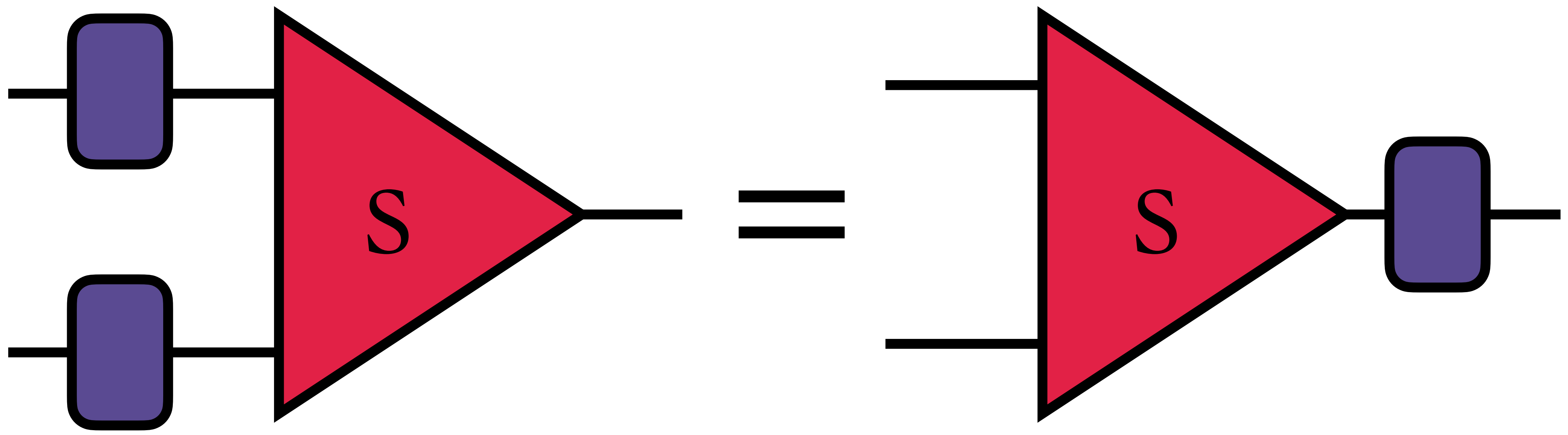
Tensor product

$$[\boldsymbol{u} \otimes \boldsymbol{v}]^{\ell}_{m} = \sum_{m_1=-\ell_1}^{\ell_1} \sum_{m_2=-\ell_2}^{\ell_2} S^{\ell_1 \ell_2}_{m_1 m_2 \ell m} u^{\ell_1}_{m_1} v^{\ell_2}_{m_2}$$

Clebsch-Gordan coefficients $\left(\langle \ell_1 m_1 | \otimes \langle \ell_2 m_2 \right) | \ell m \rangle \equiv \left[S^{\ell_1 \ell_2}\right]_{m_1 m_2, \ell m}$

$$\left[[S^{\ell_1 \ell_2}]^{-1} \left(D^{\ell_1} \otimes D^{\ell_2}\right) S^{\ell_1 \ell_2}\right] = \bigoplus_{\ell=|\ell_1-\ell_2|}^{\ell_1+\ell_2} D^{\ell}$$

Orthogonal
rotation to irreps



$$u_i v_j = \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{3}\delta_{ij} + \frac{u_i v_j - u_j v_i}{2} + \left(\frac{u_i v_j + u_j v_i}{2} - \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{3}\delta_{ij} \right)$$

$[\boldsymbol{u} \otimes \boldsymbol{v}]^{\ell=0}_{m=0}$ — Dot product

$[\boldsymbol{u} \otimes \boldsymbol{v}]^{\ell=1}_{m=-1,0,1}$ — Cross product

$[\boldsymbol{u} \otimes \boldsymbol{v}]^{\ell=2}_{m=-2,-1,0,1,2}$ — Symmetrical traceless part

$$3 \otimes 3 = 1 \oplus 3 \oplus 5$$

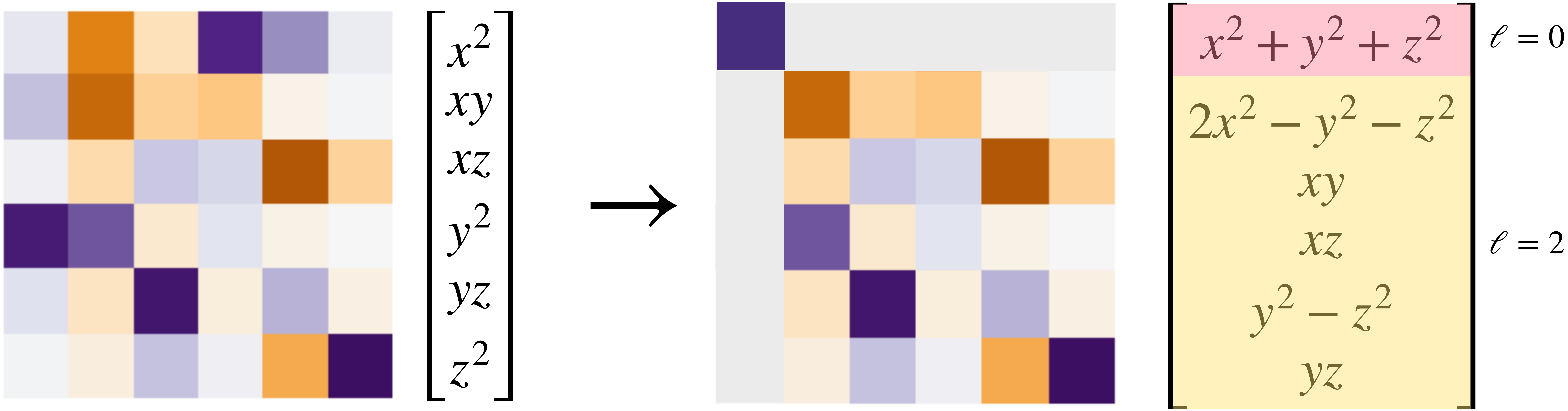
表 23.5 $S^{11}_{m_1 m_2 j m}$

$m_1\ m_2$ \ $j\ m$	0 0	1 1	1 0	1, −1	2 2	2 1	2 0	2, −1	2, −2
1 1					1				
1 0		$\sqrt{1/2}$				$\sqrt{1/2}$			
1 −1	$\sqrt{1/3}$		$\sqrt{1/2}$				$\sqrt{1/6}$		
0 1		$-\sqrt{1/2}$				$\sqrt{1/2}$			
0 0	$-\sqrt{1/3}$						$\sqrt{2/3}$		
0 −1				$\sqrt{1/2}$				$\sqrt{1/2}$	
−1 1	$\sqrt{1/3}$		$-\sqrt{1/2}$				$\sqrt{1/6}$		
−1 0				$-\sqrt{1/2}$				$\sqrt{1/2}$	
−1 −1									1

$$\begin{bmatrix} u_1 \\ u_0 \\ u_{-1} \end{bmatrix} = \begin{bmatrix} -(u_x + iu_y)/\sqrt{2} \\ u_z \\ (u_x - iu_y)/\sqrt{2} \end{bmatrix}$$

Spherical tensors — Cartesian tensors

$$u_i u_j = \frac{\boldsymbol{u}\cdot\boldsymbol{u}}{3}\delta_{ij} + \left(u_i u_j - \frac{\boldsymbol{u}\cdot\boldsymbol{u}}{3}\delta_{ij}\right)$$



Spell out the weighed tensor product

$$\boldsymbol{h}'_{i\mu} = \sum_{j \in \mathcal{N}(i)} \boldsymbol{h}_{j\nu} \overset{W_{\mu\nu}(|\boldsymbol{x}_{ij}|)}{\bigotimes} \boldsymbol{Y}(\hat{\boldsymbol{x}}_{ij})$$

$$[h'^{\ell}_{m}]_{i\mu} = \sum_{j \in \mathcal{N}(i)} \sum_{m_1, m_2} S^{\ell_1 \ell_2}_{m_1 m_2 \ell m} [h^{\ell_1}_{m_1}]_{j\nu} Y^{\ell_2}_{m_2}(\hat{\boldsymbol{x}}_{ij}) [W^{\ell_1 \ell_2}_{\ell}]_{\mu\nu}(|\boldsymbol{x}_{ij}|)$$

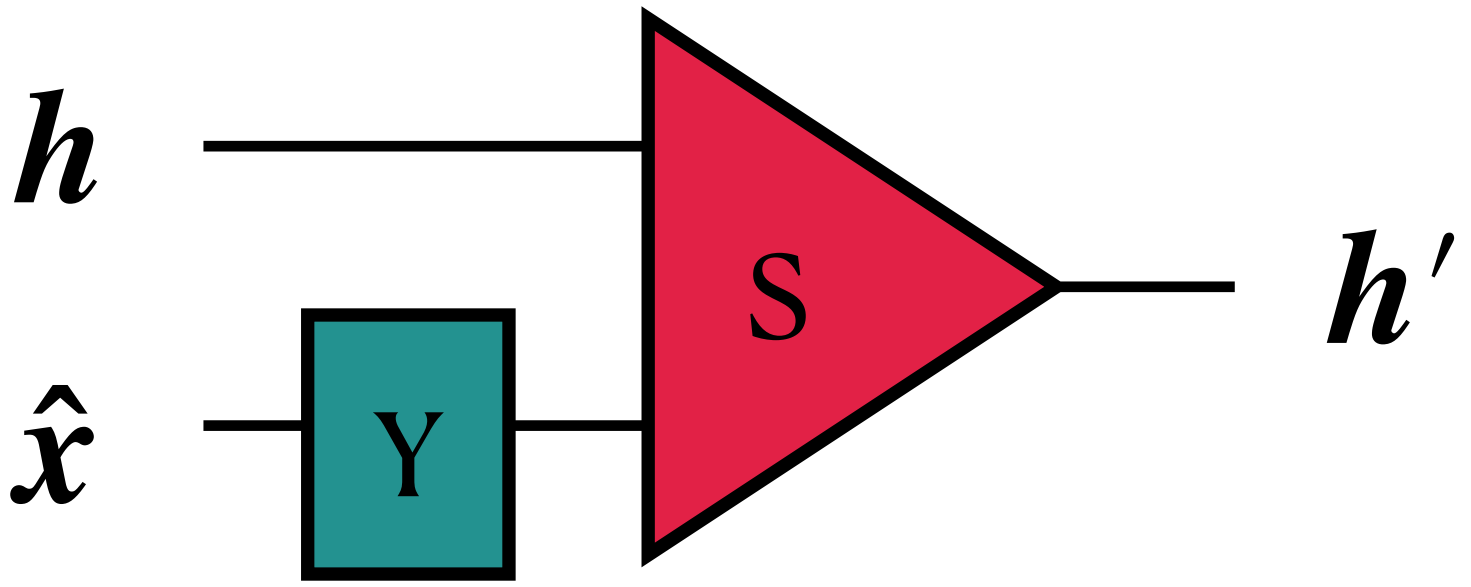
A learnable radial
function

Denote this layer as $$\boldsymbol{h}' = f(\boldsymbol{h}, \boldsymbol{x})$$

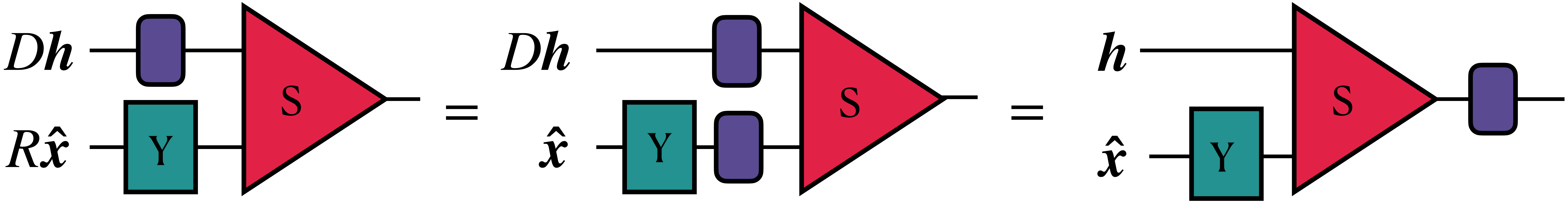
One has $$D(g)\boldsymbol{h}' = f(D(g)\boldsymbol{h}, D(g)\boldsymbol{x})$$

A visual proof of equivariance

$$\boldsymbol{h}' = f(\boldsymbol{h}, \boldsymbol{x})$$



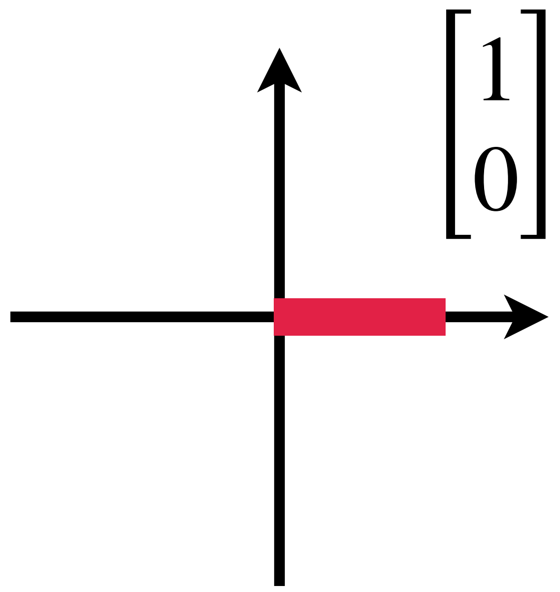
$f(D(g)\boldsymbol{h}, D(g)\boldsymbol{x})$ $D(g)\boldsymbol{h}'$



Equivariant activations

$$\sigma(D(g)\boldsymbol{h}) = D(g)\sigma(\boldsymbol{h})$$

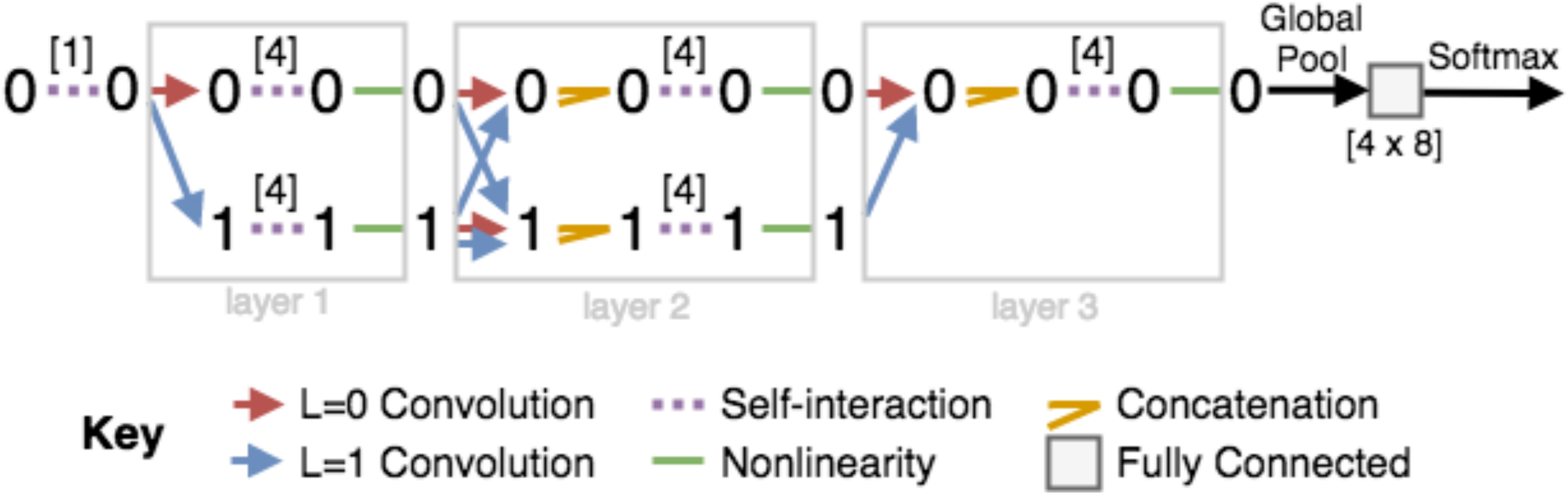
$$D(R_\pi)\mathrm{ReLU}\left(\begin{bmatrix}1\\0\end{bmatrix}\right) = \begin{bmatrix}-1\\0\end{bmatrix} \quad \neq \quad \mathrm{ReLU}\left(D(R_\pi)\begin{bmatrix}1\\0\end{bmatrix}\right) = \begin{bmatrix}0\\0\end{bmatrix}$$

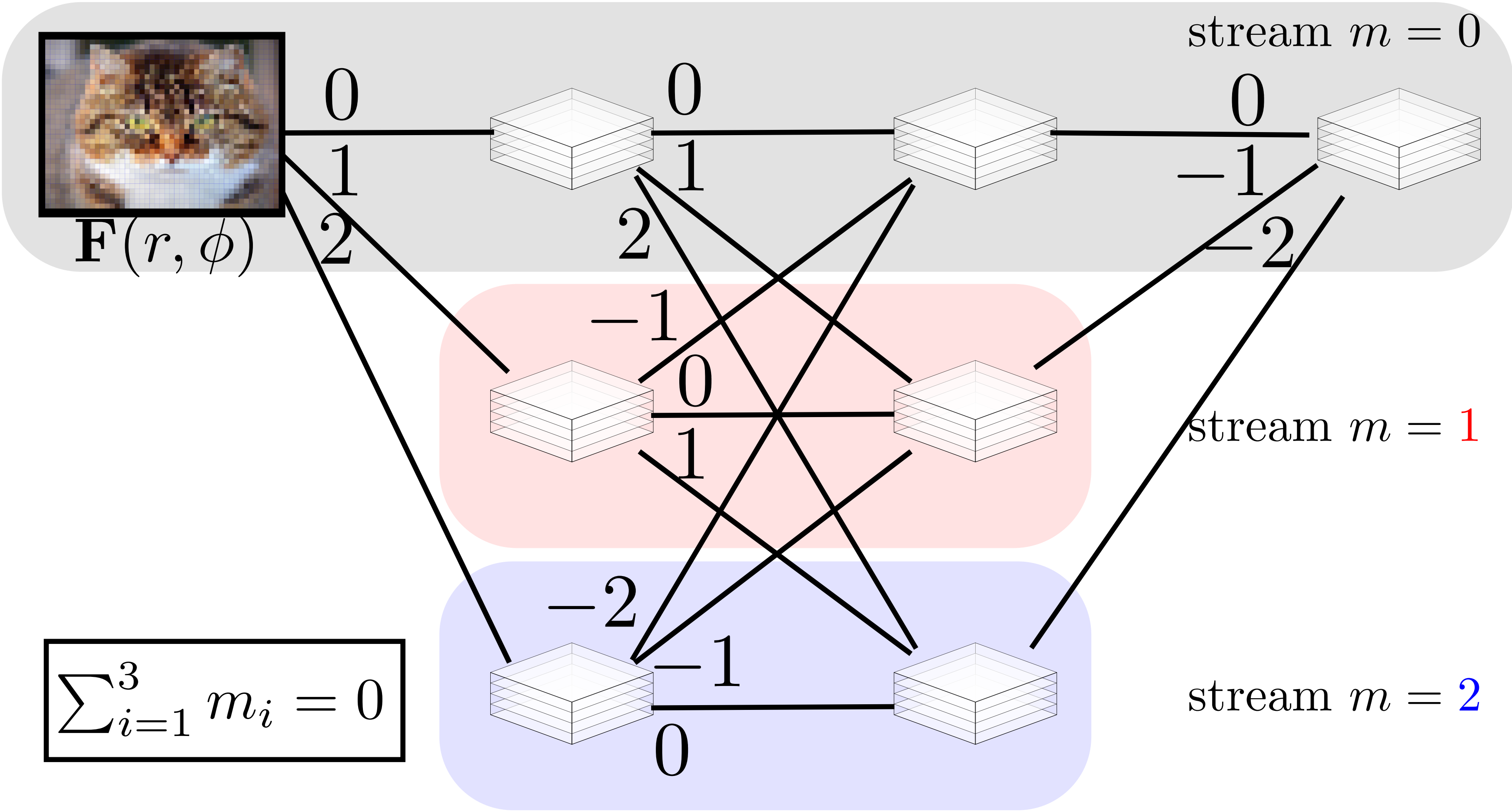


- Norm nonlinearity Thomas et al, 1802.08219 $\quad \sigma(\,|\boldsymbol{h}^{\ell>0}|\,)$
- Gated nonlinearity Weiler et al, 1807.02547 $\quad \sigma(\boldsymbol{h}^{\ell=0})\boldsymbol{h}^{\ell}$
- Tensor product Kondor 1803.01588

Put it all together

Thomas et al, 1802.08219



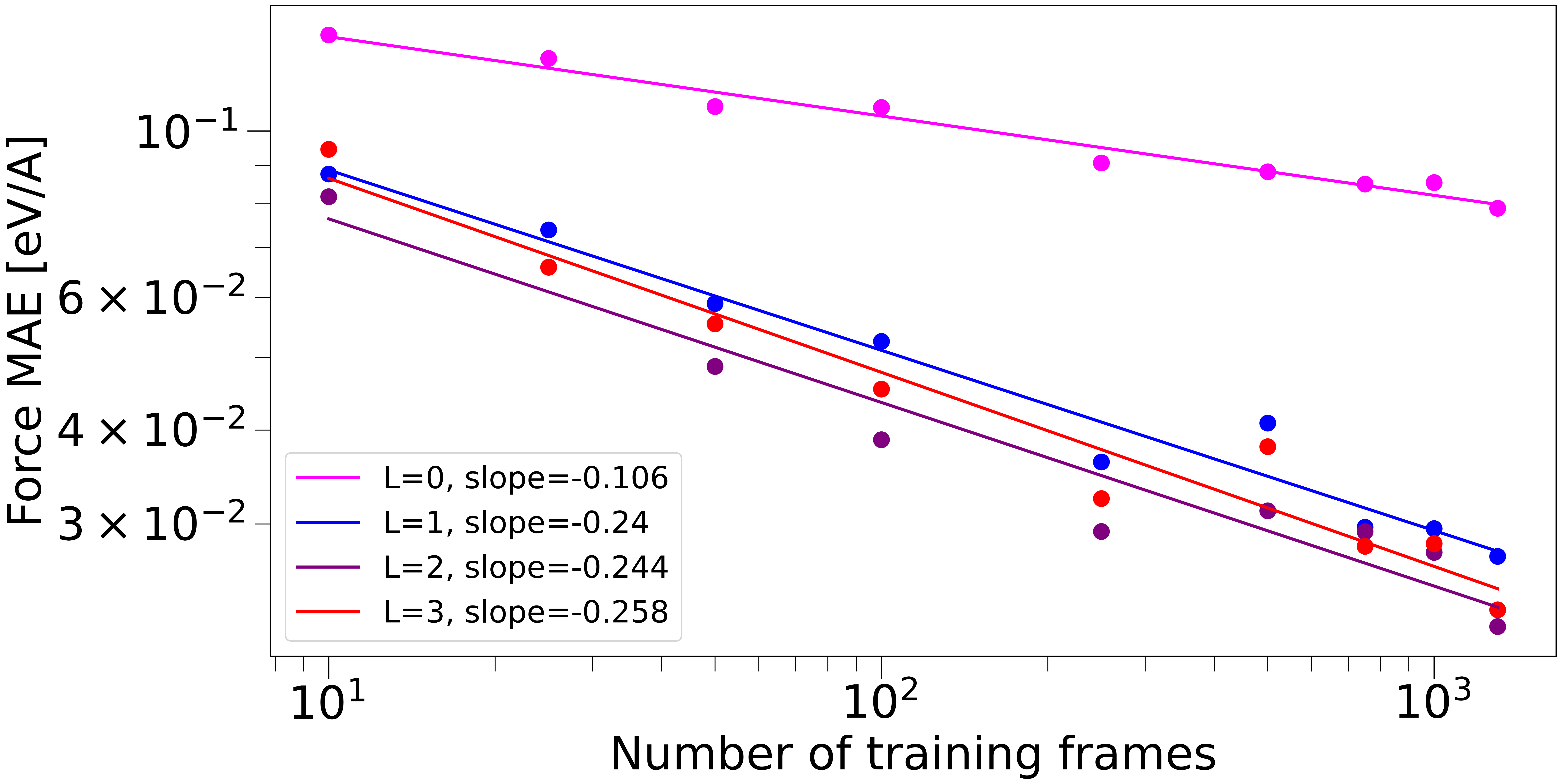


SO(2) version: harmonic networks

Worrall et al, 11612.04642

Effects of higher order equivariant hidden features

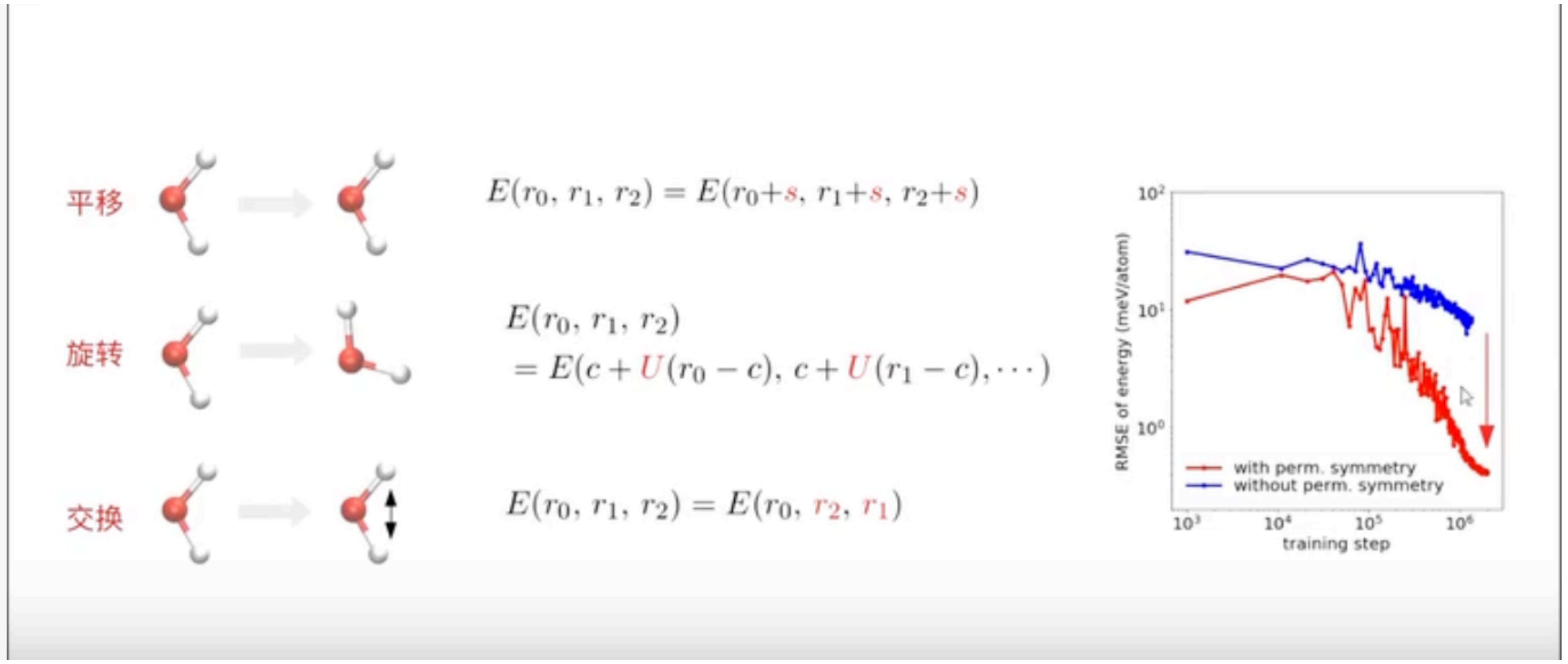
Batzner et al, Nature Comm. 2022



Linfeng Zhang et al, NIPS 2018

End-to-end Symmetry Preserving Inter-atomic Potential Energy Model for Finite and Extended Systems

Invariant features



王涵 基于深度学习的分子动力学模拟

https://www.bilibili.com/video/BV14L411E7nf/?spm_id_from=333.999.0.0

Batzner et al, Nature Comm. 2022

E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials

Equivariant transformations

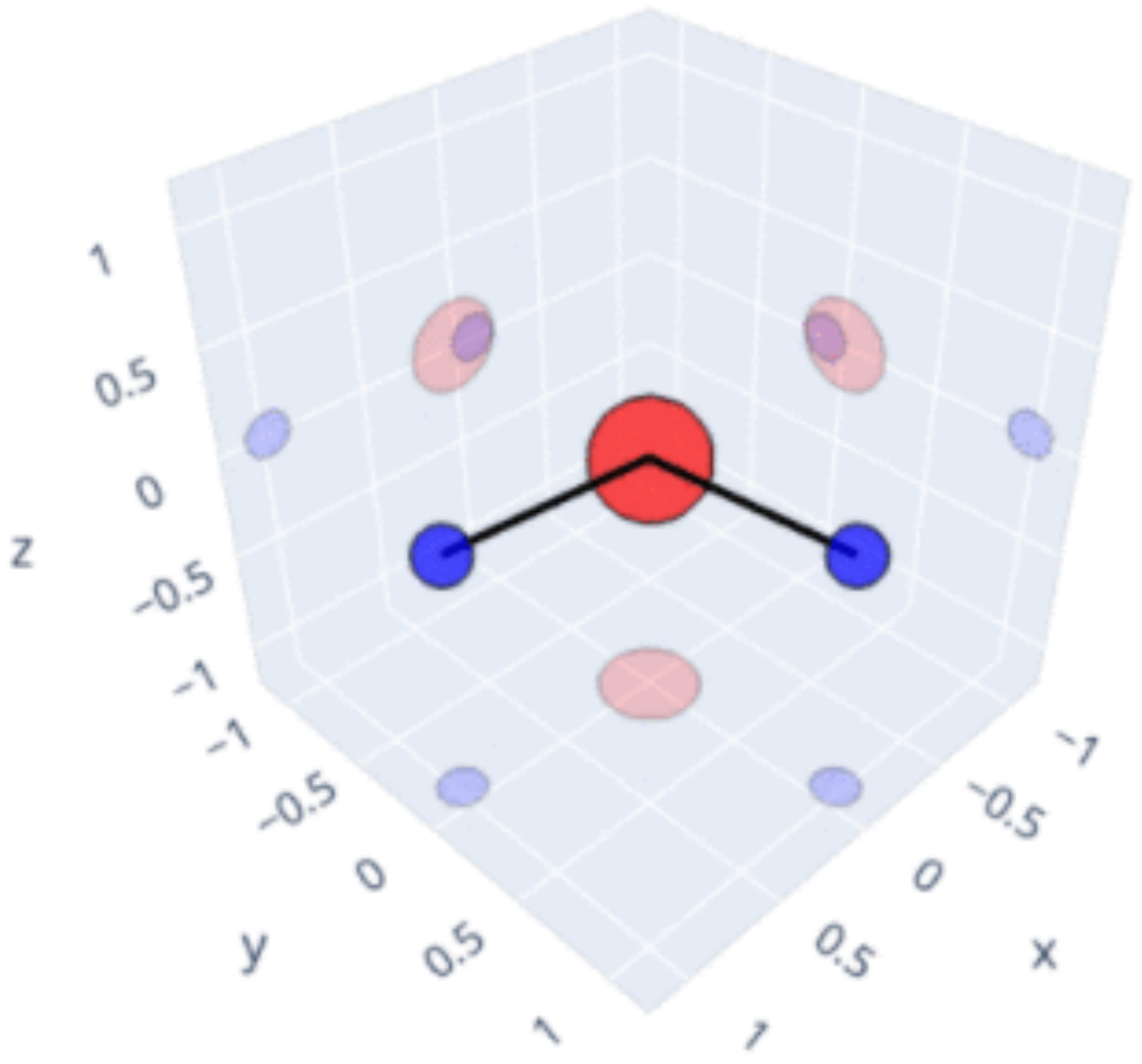
tories that were used in the earlier work[15]. Table 3 compares the energy and force errors of NequIP trained on the 133 structures vs DeepMD trained on 133,500 structures. We find that with 1000x fewer training data NequIP significantly outperforms DeepMD on all four parts of the data set in the error on the force components. We note that there are $3N$ force components for

However, these 133,500 are correlated samples

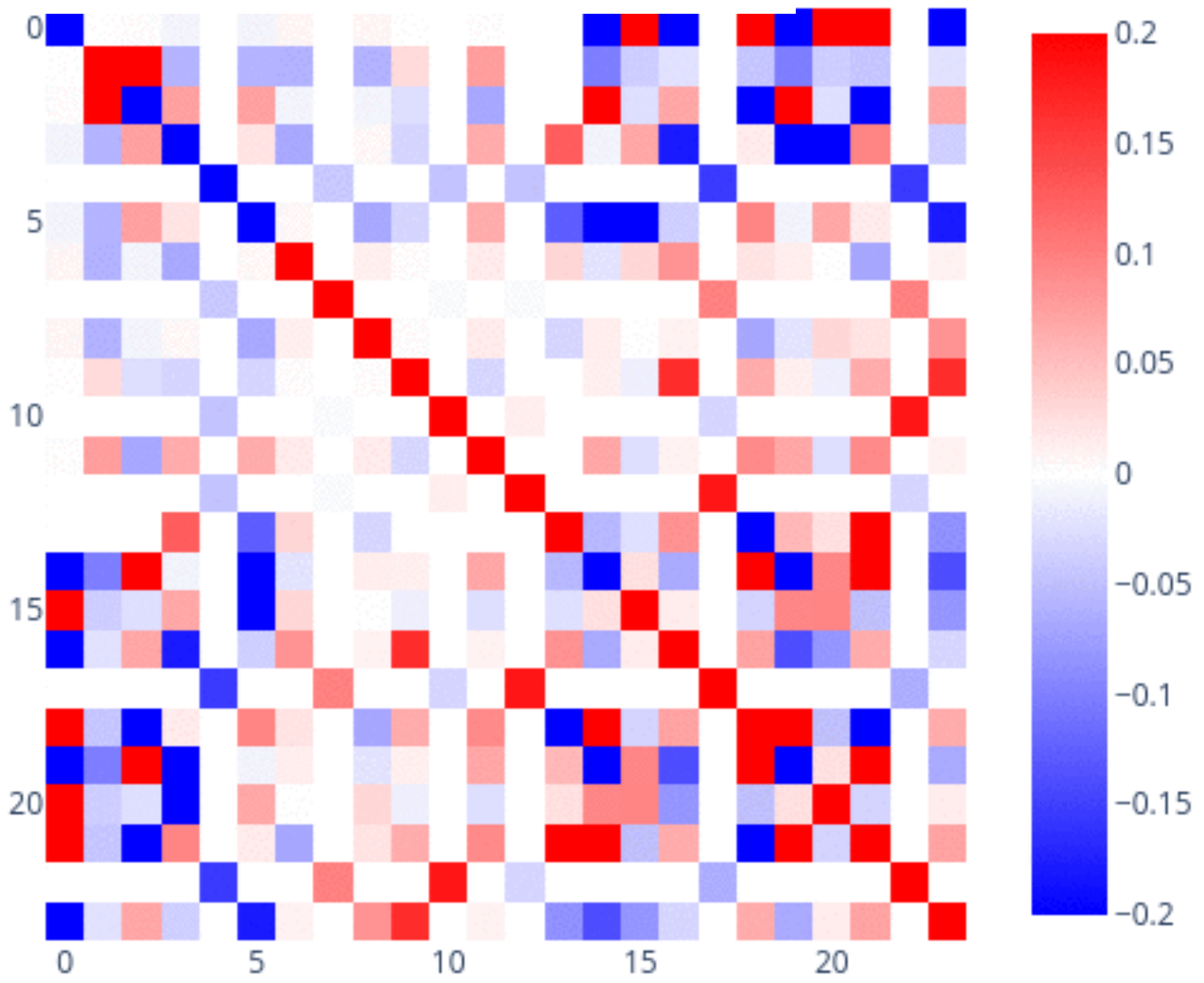
DeepMD may not need so many samples to train either

E(3) equivariant learning of Hamiltonian matrix

Rx



$D(g) H D^\dagger(g)$



Unke et al, 2106.02347, Gong et al, 2210.13955, Zhong et al, 2210.16190, Yu et al, 2211.11403

Table 4 Prediction metrics for Hamiltonian and overlap matrices: mean absolute error, less is better

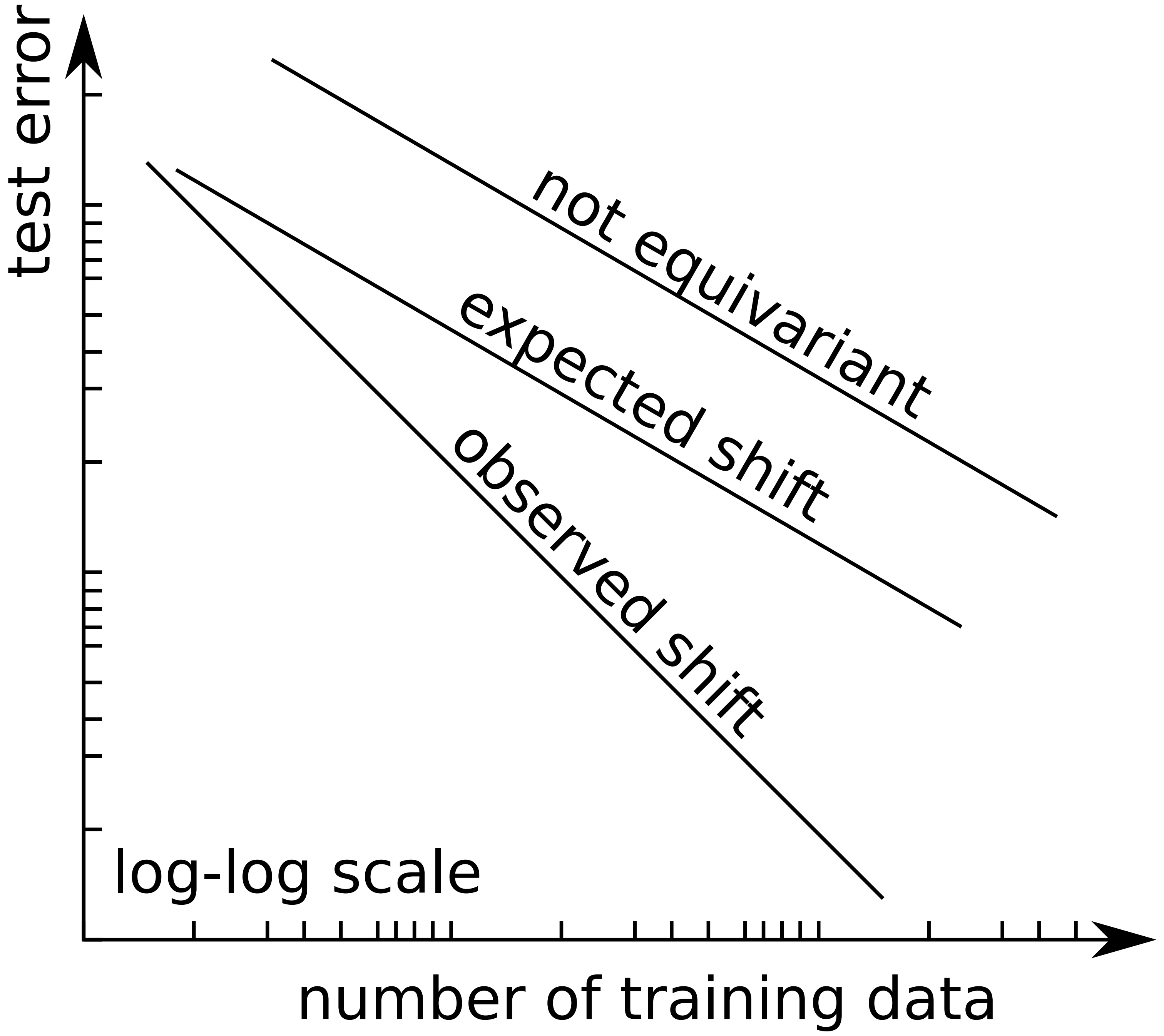
	MAE for Hamiltonian matrix prediction, $\times 10^{-3}$ E_h			MAE for overlap matrix prediction, $\times 10^{-5}$		
Model	2k	5k	10k	2k	5k	10k
Structure test split						
SchNOrb[a]	386.5	383.4	382.0	1550	1571	3610
PhiSNet	7.4	3.2	2.9	5.1	4.3	3.5
Scaffolds test split						
SchNOrb[a]	385.3	380.7	383.6	1543	1561	3591
PhiSNet	7.2	3.2	2.9	5.0	4.3	3.5
Conformations test split						
SchNOrb[a]	385.0	384.8	392.0	1544	1596	3576
PhiSNet	6.5	3.2	2.8	5.1	4.6	3.6

[a] While the relative difference between metrics for *SchNOrb* and *PhiSNet* is similar to the one reported by phisnet2021, we believe that there are still some problems with *SchNOrb* training in the multi-molecule setup, *e.g.*, gradient explosion.

Khrabrov et al, Phys. Chem. Chem. Phys., 2022

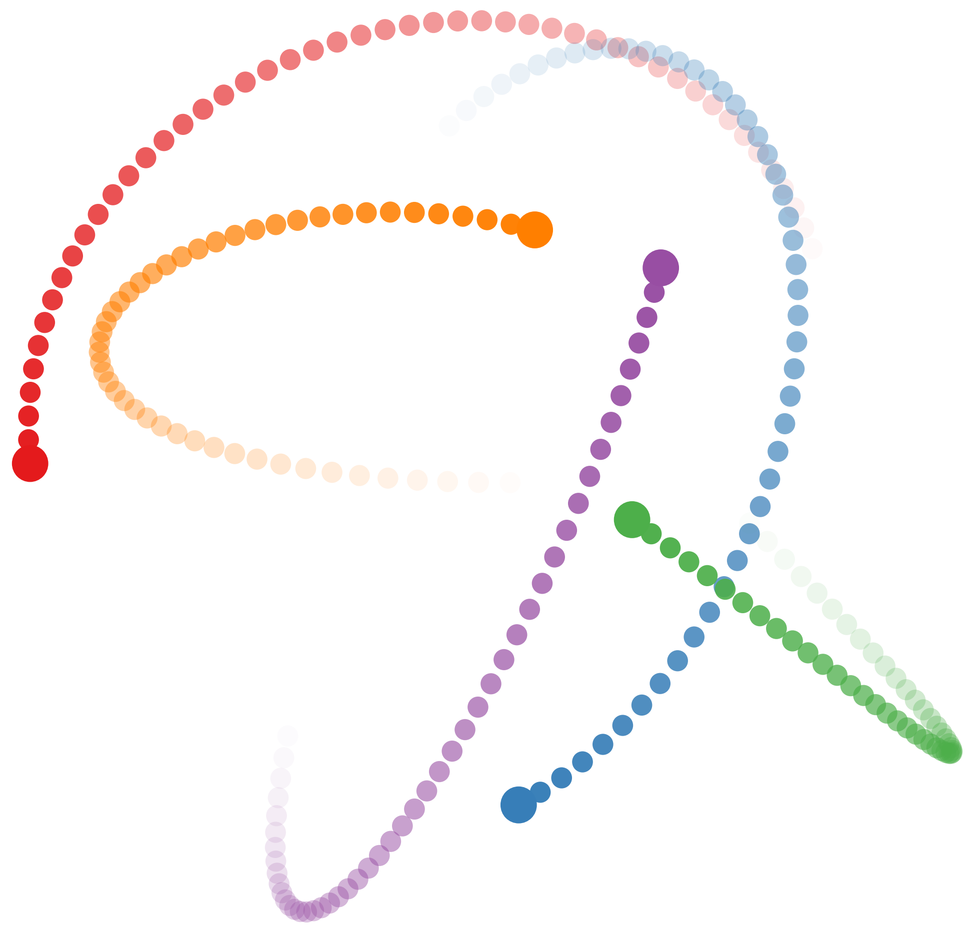
Practical recommendations

- Equivariant NN appears to show improved scaling with respect to training samples 2207.09453



- Equvariant NN may cause significant overhead due to contraction with sparse CG tensors. Update: Passaro, 2302.03655 SO(3) -> SO(2)

- However, for tasks related the Hamiltonian or density matrices written in local orbitals, Euclidean NN seems to be a natural choice



Method	MSE	Forward time (s)
Linear	0.0819	.0001
SE(3) Transformer	0.0244	.1346
Tensor Field Network	0.0155	.0343
Graph Neural Network	0.0107	.0032
Radial Field	0.0104	.0039
EGNN	**0.0071**	.0062

Table 2. Mean Squared Error for the future position estimation in the N-body system experiment, and forward time in seconds for a batch size of 100 samples running in a GTX 1080Ti GPU.

E. Sometimes invariant features are all you need.

Perhaps surprisingly we find our EGNNs outperform other equivariant networks that consider higher-order representations. In this section we prove that when only positional information is given (i.e. no velocity-type features) then the geometry is completely defined by the invariant distance norms in-between points, without loss of relevant information. As a consequence, it is not necessary to consider higher-order representation types of the relative distances, not even the relative differences as vectors. To be precise, note that these invariant features still need to be *permutation* equivariant, they are only $\mathrm{E}(n)$ invariant.

To be specific, we want to show that for a collection of points $\{\mathbf{x}_i\}_{i=1}^M$ the norm of in-between distances $\ell_2(\mathbf{x}_i, \mathbf{x}_j)$ are a *unique* identifier of the geometry, where collections separated by an $\mathrm{E}(n)$ transformations are considered to be identical. We want to show *invariance* of the norms under $\mathrm{E}(n)$ transformations and *uniqueness*: two point collections are identical (up to $\mathrm{E}(n)$ transform) when they have the same distance norms.

MLP for arbitrary matrix group

Discrete and continuous groups

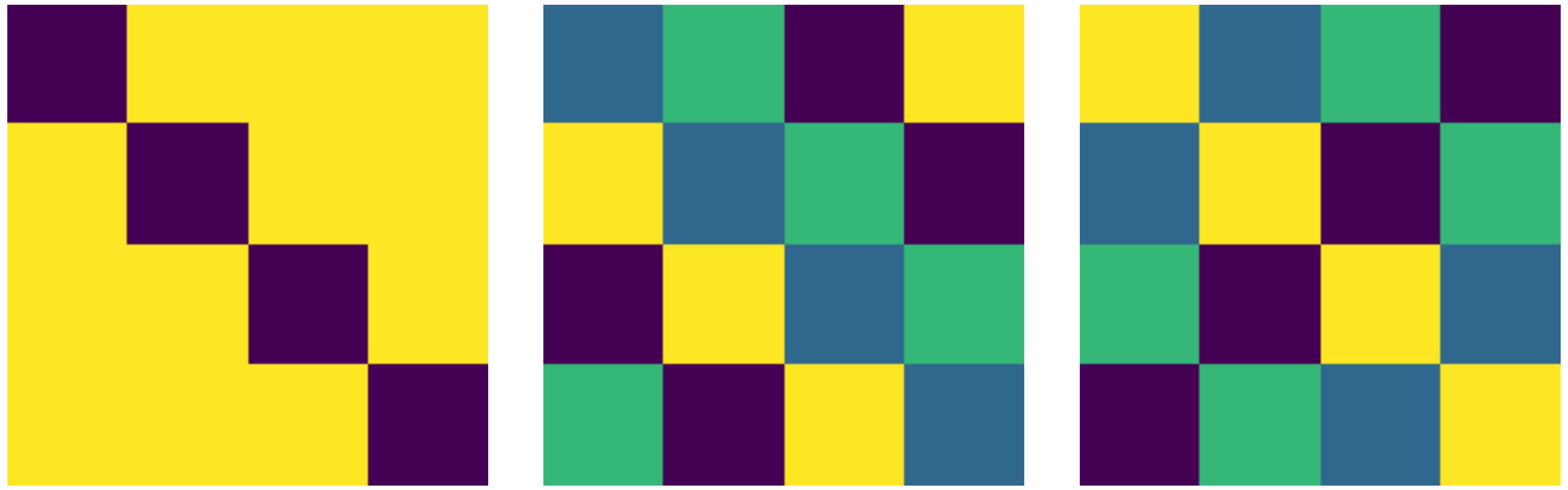
$$\forall g \in G: \quad D_Y(g) W D_X(g)^{-1} = W$$

$$\Downarrow$$

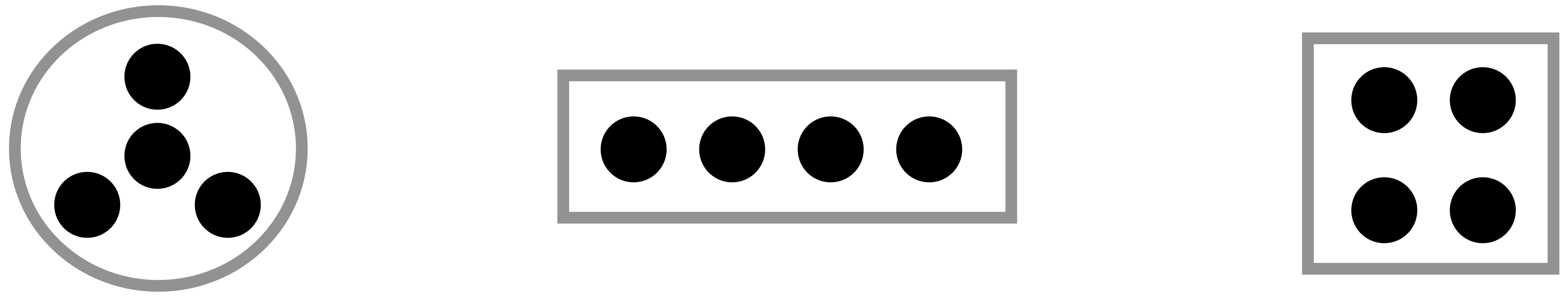
$$D_Y(g) \otimes D_X(g^{-1})^T \mathrm{vec}(W) = \mathrm{vec}(W)$$

Finzi et al 2104.09459

Solve W



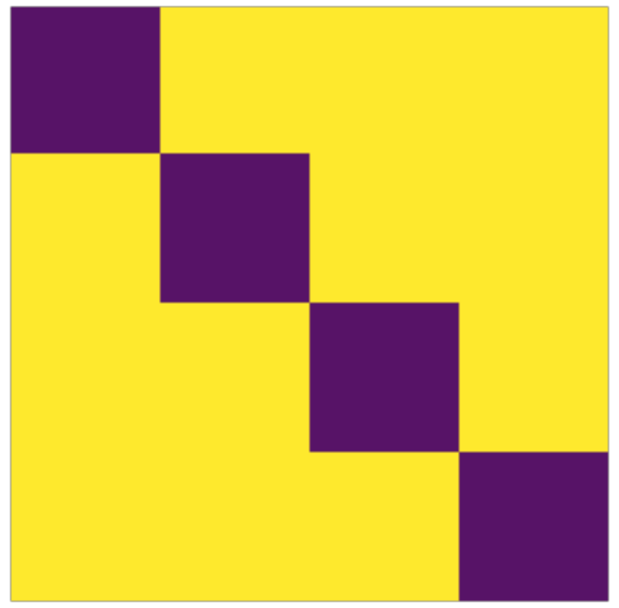
(a) S_4 (b) $\mathbb{Z}_4$ (c) $\mathbb{Z}_2^2$



Permutation equivariant layers

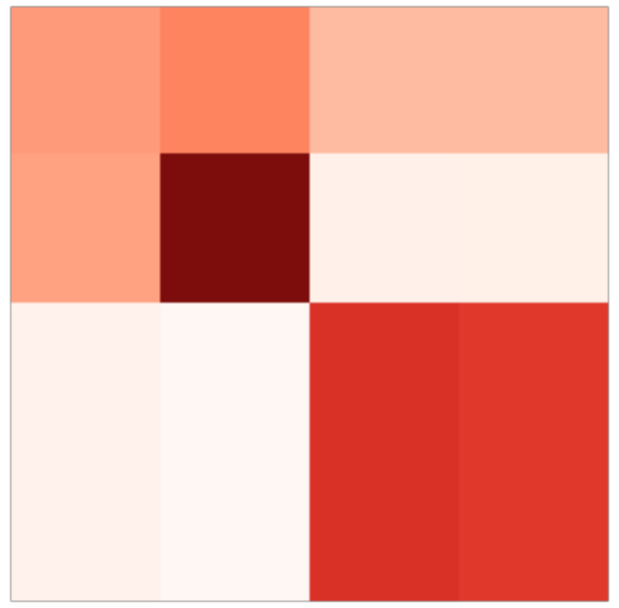
Deep Set 1703.06114

$$\boldsymbol{y} = \sigma(W\boldsymbol{x})$$
$$W = \lambda I + \gamma(11^T)$$



Self-attention 1706.03762

$$\boldsymbol{y} = \mathrm{softmax}(\boldsymbol{x}\boldsymbol{x}^T)\boldsymbol{x}$$



By global pooling equivariant features (e.g. sum), one can obtain a permutational invariant function, e.g. boson wavefucntion 2112.11957

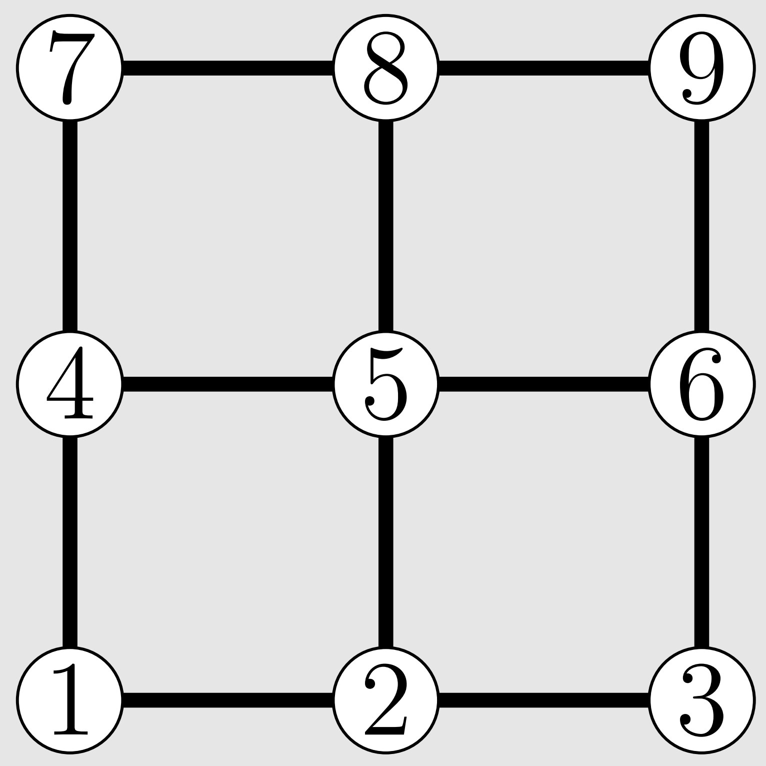
Permutation antisymmetries

Fermion wavefunctions

$$\Psi(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n) = (-1)^\pi f(\boldsymbol{x}_{\pi 1}, \boldsymbol{x}_{\pi 2}, \ldots, \boldsymbol{x}_{\pi n})$$

Canonical order

Choo et al NC 2020, Inui et al PRR 2021



Lattice model or 2nd quantization

DeepWF, Vandermont

Han et al, 2019, Acevedo et al, 2020, Pang et al, 2022

$$\prod_{i<j} \phi(x_i, x_j)$$

$$\phi(\boldsymbol{x}_i, \boldsymbol{x}_j) = -\phi(\boldsymbol{x}_j, \boldsymbol{x}_i)$$

Limited representational power

O(n²) complexity

FermiNet, PauliNet, PsiFormer

Pfau et al, PRR 2020, Hermann et al, 2020, von Glehn 2022

$$\det \begin{bmatrix} \boldsymbol{h}_1 \\ \boldsymbol{h}_2 \\ \vdots \\ \boldsymbol{h}_n \end{bmatrix} \qquad \text{pfaffian} \begin{bmatrix} \boldsymbol{h}_1 \\ \boldsymbol{h}_2 \\ \vdots \\ \boldsymbol{h}_n \end{bmatrix}$$

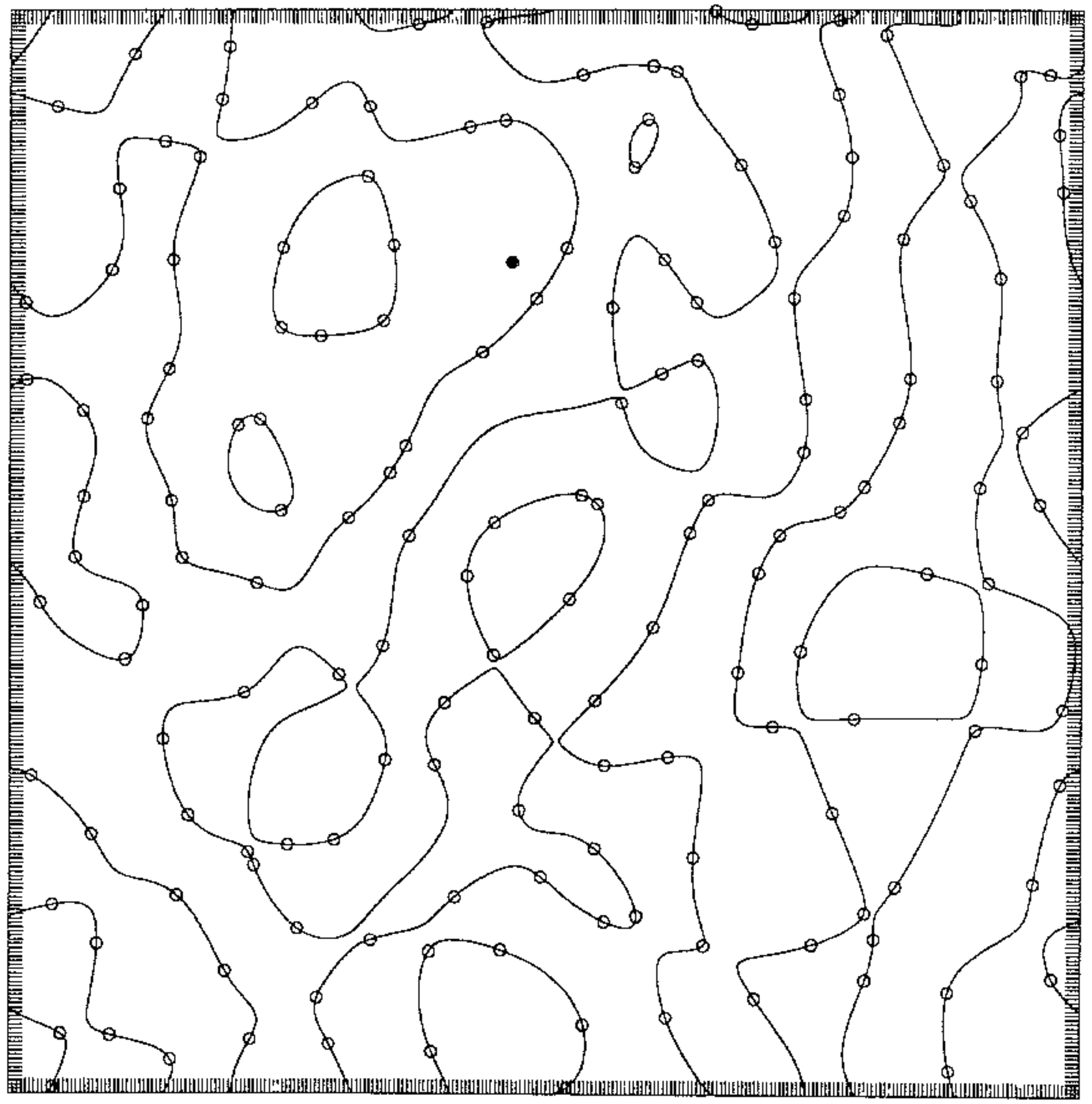
$\boldsymbol{h}$ is a perm-equivariant transformation of $\boldsymbol{x}$

O(n3) complexity

The mystery of fermion node

- 3n-1 dimensional hyper surface
- (3n-3)-dimensional Pauli node does not exhaust the nodes
- Knowing the exact nodes would allow solving fermionic ground state
- It was conjectured that the fermion ground state contains only two tiles

$$\Psi(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n) = 0$$



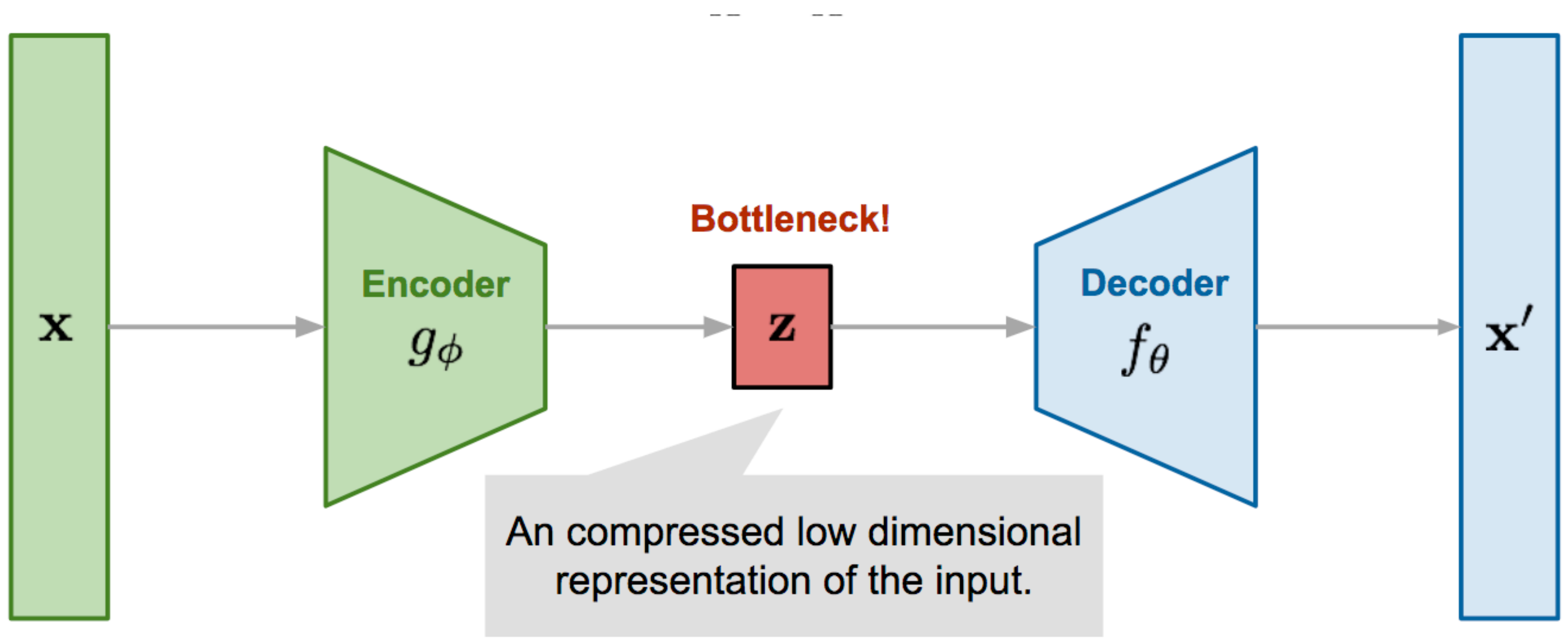
Ceperley 1991

1812.02230

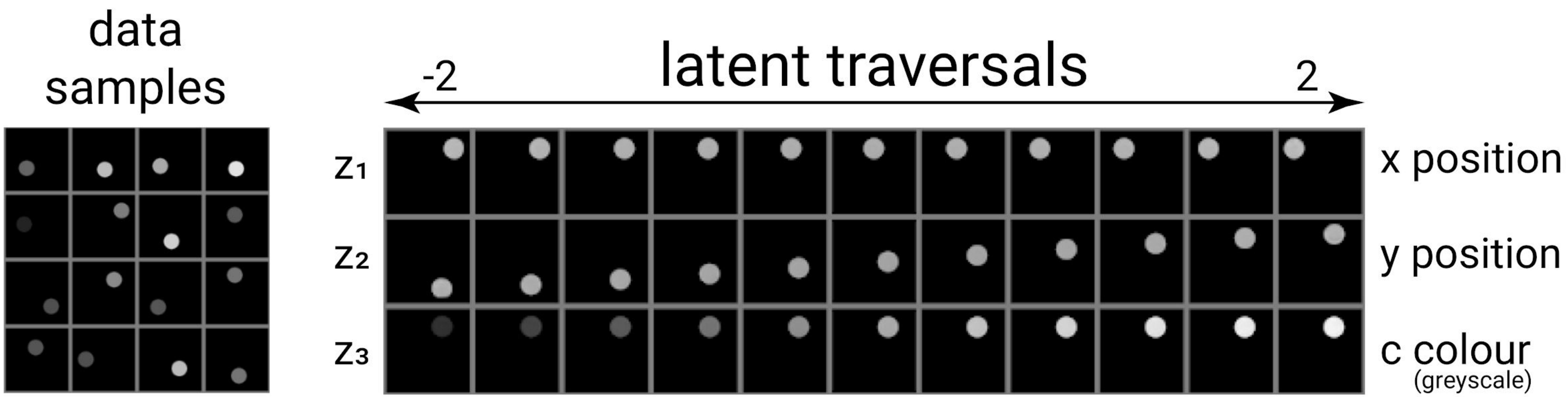
Towards a Definition of Disentangled Representations

Irina Higgins*, David Amos*, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, Alexander Lerchner
DeepMind

By connecting symmetry transformations to vector representations using the formalism of group and representation theory we arrive at the first formal definition of disentangled representations. Our new definition is in agreement



$$G = G_x \times G_y \times G_c$$



Machine learning symmetries (and its breaking)

Noether theorem 1915

Symmetry => Conversation law

Krippendorf et al, 2003.13679
Liu et al, PRL 2021
Desai et al, PRD '22...

Curie's principle 1894

The symmetries of the causes are to be found in the effects

Let f be equivariant to G, $\mathrm{Sym}(x) = \{g \in G : D(g)x = x\}$

Smidt et al, PRR 2021

$$\mathrm{Sym}(x) \subseteq \mathrm{Sym}[f(x)]$$