

Equivariant Networks

Applied to Crystal Structures

Alex Heilman

May 13, 2025

- What does equivariance mean in general?
- Examples of Equivariant Convolutions for $SO(3)$ and Crystal Structures
- Project Ideas

An equivariant function $f : X \rightarrow Y$ is one that satisfies the following equality:

$$f(D_X(g) \cdot x) = D_Y(g) \cdot f(x)$$

Equivariant Convolution

Equivariant
Networks

Alex Heilman

Equivariance

NEquIP

Crystal Networks

Project

Ideas/Extensions

Enforcing equivariance often restricts our choice of function; while also guaranteeing a natural condition for certain tasks.

Hence, equivariance conditions may often restrict our trainable parameter space (requiring less data in training) while providing similar expressibility of the model (giving similar, or sometimes better performance).

Example: Convolutional Networks and Translation

Traditional convolutional neural networks (CNNs) are actually equivariant under the group action of translation (in 2D for the usual input of images).

Special Orthogonal Group in 3D

The special orthogonal group of 3 dimensions, $SO(3)$, is the group describing 3 dimensional rotations.

It has irreducible representations indexed by a rotational order ℓ and a harmonic order m , termed the spherical harmonics Y_m^ℓ .

Note that products of spherical harmonics can be decomposed in terms of another linear superposition of spherical harmonics via Clebsch-Gordon Coefficients $C_{\ell_1 m_1 \ell_2 m_2}^{\ell_f m_f}$

$$Y_{\ell_1}^{m_1}(\Omega) Y_{\ell_2}^{m_2}(\Omega) = \sum_{\ell_3, m_3} \sqrt{\frac{(2\ell_1 + 1)(2\ell_2 + 1)}{4\pi(2\ell_3 + 1)}} C_{\ell_1 m_1 \ell_2 m_2}^{\ell_3 m_3} C_{\ell_1 0 \ell_2 0}^{\ell_3 0} Y_{\ell_3 m_3}(\Omega)$$

NEquiP utilizes a convolutional structure that is equivariant under the $SO(3)$ group, following that introduced in the Tensor Field Networks paper

With a layer-to-layer convolution defined as:

$$\mathcal{L}_{acm_o}^{(l_o)} = \sum_{m_f, m_i} C_{l_f m_f, l_i m_i}^{l_o m_o} \sum_{b \in S} F_{cm_f}^{l_f l_i}(\vec{r}_{ab}) V_{bcm_i}^{l_i}$$

Essentially, different atomic positions have different local site symmetries and these different site symmetries allow us to define different classes of atomic positions.

The idea then is to have a different set of weights (parameter sharing) for each 'class' of atomic position. So, we effectively have a new index for the update and message functions, as below

$$\begin{aligned}m_v^{\ell+1} &\rightarrow m_v^{\ell+1,c} \\ h_v^{\ell+1} &\rightarrow h_v^{\ell+1,c}\end{aligned}$$

where the upper index c denotes the class or flavor of message passing function, which acts only on certain subsets of atomic positions

So what do we do?

Old Models, New Targets: Something that's actually equivariant! (Elasticity/Compliance Tensor, Dielectric Tensor, ...)

Old Targets, New Models: Maybe try to apply $SO(3)$ convolution in a way similar to the crystal group approach

New Target: Elasticity Tensor

Material strain (how it stretches) can be modeled as a linear transformation on the material stress

$$T^{ij} = C_{ijkl} E^{kl}$$

where **T** is the strain tensor and **E** is the stress tensor.

Materials have corresponding compliance/elasticity tensors **C**, perhaps we could use this as a target?

New Target: Dielectric Tensor

Applied electric fields induce electric displacements in materials according to

$$D^i = \epsilon_{ij} E^j$$

where **D** is the electric displacement field and **E** is the electric field.

Materials have corresponding permittivity/dielectric tensors ϵ , perhaps we could use this as a target?

New Targets cont.

Both of these tensors are available for certain materials in the materials project database

Note that these tensors have many restrictions that often make them simpler

These tensors should also be equivariant under rotations of the input crystal structure. This would utilize equivariance in a natural way, as opposed to the invariant quantities usually used as targets

Concerns: data may be lacking, do people care about these quantities?

$SO(3)$ convolution/traditional equivariant networks tend to promise smaller parameter spaces for similar performance

Crystal group's approach seems to increase the parameter space (likely due to a relaxation of total permutation invariance)

Perhaps we could utilize $SO(3)$ convolution in an approach similar to the crystal group paper, in which we have different classes for different point groups, each with their own convolutional weights