FreeCG: Free the Design Space of Clebsch–Gordan Transform for Machine Learning Force Fields

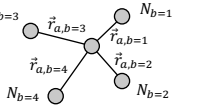
Shihao Shao[1], Haoran Geng[1], Zun Wang[2], Qinghua Cui[1]

[1] [2]

ICLR

Motivations

Let us revisit the single message construction in equivariant graph neural networks:



$$\mathcal{L}^{l_o p_o l_n p_n \mapsto l_o p_o}_{acm_o}(\boldsymbol{X}, \boldsymbol{N}) = \mathbb{1}_{(p_o = p_e p_n)} \sum_{m_e m_n} C^{l_o l_e l_n}_{m_o m_e m_n}$$

$$\sum_{b \in N(a)} (R(\|\vec{r}_{ab}\|)^{l_o l_e l_n}_{c} Y^{l_e}_{m_e}(\frac{\vec{r}_{ab}}{\|\vec{r}_{ab}\|}) N^{l_n p_n}_{bcm_n}$$

There exist two main limitations:

1. The CG transform layer scales as $O(n)$, where n is the number of neighbouring atoms for a given atom. One cannot arbitrarily remove calculations for a specific neighboring atom for the sake of permutation equivariance.

2. The design space is limited for maintaining permutation equivariance. In the above equation, formulation and the parameters of R should be the same across different neighboring atoms, thus forbidding the design for complicated CG transform layers.

Methods

Given a CG layer h,

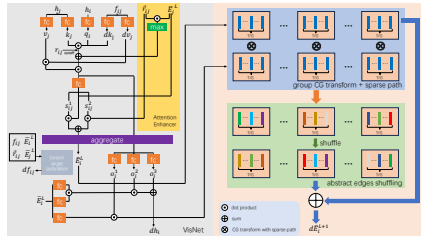
$h(x) = h'(h_1(x), h_2(x), \dots)$,

we utilize the invariance transivity trick:

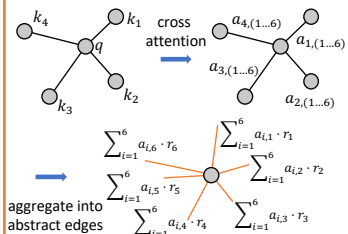
$h'(h_1(P_X(g)x), h_2(P_X(g)x), \dots)$
$= P_h(e)h'(h_1(x), h_2(x), \dots)$,

if we force each h_* to be a permutation invariant function, then h is naturally a permutation invariant function.

Thus, the key is to design each h_*, upon which, we can also freely construct CG layer h.



Abstract Edges (functions h_*)



Designs of the Function h

Group CG transform. We divide the abstract edges into different groups. The group CG transform decreases the time complexity from $O(T^2)$ to $O\left(\frac{T^2}{G}\right)$, where T and G numbers of abstract edges, and groups, respectively.

Abstract edges shuffling. Inspired by ShuffleNet, we further shuffle the abstract edges to better mix the information.

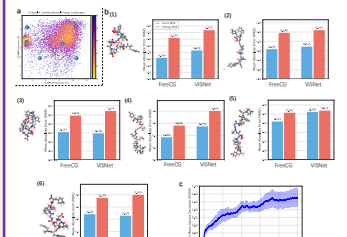
Sparse Path. Only a limited number of weight-parity pairs is permitted to achieve O(3) equivariance, while decreasing the complexity to be even lower than cheaper SO(3).

Results

On MD17:

Molecule	SchNet	DimeNet	PaiNN	SpookyNet	ET	GemNet	NequIP	SO3KRATES	ViSNet	QuinNet	FreeCG
Energy Prediction											
Aspirin	0.37	0.204	0.167	0.151	0.123	-	0.131	0.139	0.116	0.119	**0.110**
Ethanol	0.08	0.064	0.064	0.052	0.052	-	0.051	0.052	0.051	0.050	**0.049**
Malondialdehyde	0.13	0.104	0.091	0.079	0.077	-	0.076	0.077	**0.075**	0.078	0.094
Naphthalene	0.16	0.122	0.116	0.116	0.085	-	0.113	0.115	0.085	0.101	**0.083**
Salicylic acid	0.20	0.134	0.136	0.114	0.093	-	0.106	0.016	0.092	0.101	**0.090**
Toluene	0.12	0.102	0.095	0.094	**0.074**	-	0.092	0.095	**0.074**	0.080	0.076
Uracil	0.14	0.115	0.106	0.105	**0.095**	-	0.104	0.103	**0.095**	0.096	0.097
Force Prediction											
Aspirin	1.35	0.499	0.338	0.258	0.253	0.217	0.184	0.236	0.155	0.145	**0.122**
Ethanol	0.39	0.230	0.224	0.094	0.109	0.085	0.071	0.096	0.060	0.060	**0.053**
Malondialdehyde	0.66	0.383	0.319	0.167	0.169	0.155	0.129	0.147	0.100	0.097	**0.095**
Naphthalene	0.58	0.215	0.077	0.088	0.061	0.031	0.058	0.074	0.036	0.038	**0.034**
Salicylic acid	0.85	0.374	0.195	0.180	0.129	0.125	0.090	0.145	0.084	0.080	**0.070**
Toluene	0.57	0.216	0.094	0.083	0.060	0.046	0.053	0.098	0.039	0.039	**0.038**
Uracil	0.56	0.301	0.139	0.119	0.095	0.097	0.076	0.111	0.062	0.062	**0.059**

On Chig-MD:



Check out our paper for details and more experimental results: